



Faculty of Medicine

Systems Biology

Department of Human and Medical Genetics

Robertas Tomas Tupikas, 2nd year, group 1

Master's Thesis

**Atskaitos tašku normalizuotas dviejų dalių regresijos modelis mikrobinės
sudėties vertinimui**

**A Reference-Normalized Two-Part Regression Model for Microbial
Composition Assessment**

Supervisor

Dr. Sigitas Šulčius

Head of the department or clinic

Prof. Dr. Algirdas Utkus

Vilnius, 2026

Robertas.tupikas@mf.stud.vu.lt

Table of Contents

ABBREVIATIONS	4
1. INTRODUCTION	5
LAY DESCRIPTION	7
2. LITERATURE REVIEW	8
2.1. Statistical challenges and approaches in microbial data.....	8
2.1.1. Compositionality	8
2.1.2. Zero inflation and sparse data.....	9
2.1.3. Sequencing depth differences and variability.....	9
2.1.4. Misspecification of statistical distribution.....	10
2.1.5. Complex experimental designs.....	10
2.2. Available tool performance	10
2.2.1. MaAsLin2	11
2.2.2. ANCOM-BC2.....	11
2.2.3. LinDA	12
2.2.4. LDM	13
2.2.5. LOCOM.....	13
2.2.6. ALDEx2.....	14
2.2.7. Corncob	14
2.2.8. ZicoSeq.....	15
2.3. Summary.....	16
3. METHODS	18
3.1. PURSUE procedure	18
3.1.1. Pre-filtering.....	18
3.1.2. Reference selection.....	18
3.1.3. Two-part model	26
3.1.4. Permutation-based inference	29
3.1.5. Two-part model p-value aggregation and independent filtering	33
3.2. DAA tool performance evaluation	35
4. RESULTS	39
4.1. Calibration under null settings.....	39
4.1.1. Mean false positive rate across null settings	39
4.1.2. False positive rate across individual null settings	39
4.2. Detection performance under non-null settings.....	41
4.2.1. Aggregate detection performance across non-null settings	41
4.2.2. Detection performance in individual non-null settings	43

5. DISCUSSION.....	51
6. CONCLUSIONS	57
ACKNOWLEDGMENTS	58
REFERENCES	59
SUMMARY.....	65
SANTRAUKA.....	66
APPENDIX 1. PACKAGE DESIGN.....	67
APPENDIX 2. RUNTIME AND EMPIRICAL DEFAULTS.....	75

ABBREVIATIONS

16S rRNA - 16S ribosomal RNA

ACAT - Aggregated Cauchy Association Test

BH - Benjamini-Hochberg

CLR - Centered log-ratio

DAA - Differential abundance analysis

FDR - False discovery rate

FN - False negative

FP - False positive

FPR - False positive rate

GLM - Generalized linear model

NA - Not available, missing value

OFAT - One factor at a time

OLS - Ordinary least squares

OTU - Operational taxonomic unit

SD - Standard deviation

SE - Standard error

SS filter - Sensitivity filter

TN - True negative

TP - True positive

TSS - Total sum scaling

WMS - Whole-metagenome shotgun sequencing

1. INTRODUCTION

The microbiome profoundly influences both human health and ecological processes. In humans, the gut microbiome has vast gene content and crucial roles in physiology (Hurst, 2017; O'Hara & Shanahan, 2006). Microbes help regulate metabolism and immune function and even modulate brain activity via the gut-brain axis (Park et al., 2025). Disruptions in microbiome composition have been linked to diseases ranging from obesity (Turnbaugh et al., 2008) to Parkinson's (Elfil et al., 2020). In ecological contexts, environmental microbial communities influence nutrient cycling, organic matter decomposition, and climate regulation, and changes in microbial abundance can reflect shifts in soil fertility, plant productivity, and ecosystem health (Iqbal et al., 2025). To elucidate the microbial processes and potential biomarkers underlying these conditions and ecological states, differential abundance analysis (DAA) aims to identify the microbial features whose abundance shifts are statistically associated with a given condition or measurement. The most widely used method for profiling microbial communities in DAA is 16S rRNA gene amplicon sequencing (Nearing et al., 2022). In this approach, one or more hypervariable regions of the 16S rRNA gene are amplified by PCR and sequenced, providing taxonomic profiling of bacteria across samples (Durazzi et al., 2021). The reads may be clustered into operational taxonomic units (OTUs) (Edgar, 2013) or single nucleotide resolved amplicon sequence variants (Amir et al., 2017), which represent sequence-derived microbial features used as proxies for taxa. This processing yields a sample-by-taxon count matrix, accompanied by sample metadata, which forms the basic input for DAA (Lin & Peddada, 2020). This technology offers an efficient way to generate large, comparative microbiome datasets across many conditions, making it the basis of most of the DAA literature. Unfortunately, due to intrinsic biases of PCR and technical variation in DNA sequencing, 16S amplicon data has troublesome characteristics, such as compositionality, zero inflation, and library size heterogeneity (Lin & Peddada, 2020; Nearing et al., 2022). This means that DAA methods must be specifically adapted to this kind of data to correctly infer shifts in microbial abundance. Currently, there is a plethora of DAA tools available, each with distinct statistical assumptions and inference targets, however, none of them perform well under any and all datasets (Yang & Chen, 2022). In the presence of differing sample sizes, diversity and proportions of shifting taxa, and confounding variation, the same tools exhibit different performance metrics. In their review, Yang & Chen (2022) previously argued that these observations necessitate a DAA procedure that would balance scalability, design flexibility, and power/calibration across a broad range of realistic microbiome data settings. Therefore, the aim of this study was to design a method exhibiting those properties.

To achieve this aim, four objectives were defined:

1. To develop PURSUE (**P**revalence-abundance **U**nified **R**eference-normalized **S**ignificance testing **U**sing **E**mpirical permutation), a reference-normalized two-part differential abundance analysis method for 16S ribosomal RNA microbiome count data, combining stable reference selection, two-part modeling, and permutation-based inference.
2. To evaluate the calibration of PURSUE under null simulation settings.
3. To evaluate the detection performance of PURSUE under non-null settings.
4. To compare its performance against established differential abundance analysis methods and identify the data settings in which it performs well or poorly.

PURSUE is available at the <https://github.com/RTTnrcBioInf/pursue> repository.

LAY DESCRIPTION

Microbes live in and on humans, animals, plants, soil, and water, and changes in these microbial communities can be linked to disease, health, and environmental conditions. Researchers often study these communities using DNA sequencing, where unique DNA sequences are used to identify different microbes in a mixed sample. The amount of DNA detected from each microbe can then be used as an estimate of how much of that microbe is present. However, this is difficult because the detected DNA does not always reflect the true microbial amount. For example, some microbes are harder to break open and provide less DNA to sequence. As a result, apparent microbial differences can arise from variation that is not what we are trying to study. There are currently many tools that attempt to calculate true microbe amount changes, but none of them avoid mistakes in every data scenario. This thesis developed PURSUE, a new tool for detecting microbial abundance differences while accounting for the troublesome microbe data variation. The method was tested on simulated microbiome datasets where the truly changed microbes were already known and was compared with eight existing analysis tools. PURSUE was especially good at avoiding false discoveries when no microbes were truly changing in amount. It also performed well when total DNA amounts differed between samples - a condition that can falsely appear as microbial change. However, it also sometimes missed true changes. Overall, PURSUE appears useful as a high-precision tool, while also showing where further development is needed.

2. LITERATURE REVIEW

2.1. Statistical challenges and approaches in microbial data

Because sequencing produces relative, confounded, and sparse counts, DAA is essentially an inference problem under measurement constraints.

2.1.1. Compositionality

Several biological and technical factors distort the relationship between true cell counts and observed read counts. Because total microbial load is not identifiable from 16S sequencing alone, observed counts are treated as compositional. As a result, taxon counts are interpreted relative to the total read count within each sample (Lin & Peddada, 2020; Nearing et al., 2022). There are several factors that necessitate this: sequencing counts intrinsically depend on sequencing depth, which means that normalization is not optional. Taxon-specific extraction and lysis efficiencies can also introduce distortion, as some organisms yield DNA readily while others resist disruption under the same protocols (Elie et al., 2023; Teng et al., 2018). Primer mismatches and differences in primer binding and denaturation can also cause preferential amplification of some taxa and under-representation of others (Kanagawa, 2003; Klindworth et al., 2012; Laursen et al., 2017). Lastly, sequencing platform specifics can add technical variation to the observed counts (Schloss et al., 2011). Compositionality creates several statistical complications for differential abundance analysis. First, compositions are non-identifiable with respect to absolute abundance, meaning multiple absolute abundance configurations can yield the same observed relative proportions, for example when all microbes increase or decrease proportionally so that their ratios remain unchanged (Morton et al., 2019). Second, taxa become negatively correlated by mathematical necessity rather than biological effect, so a relative increase in one component can induce apparent decreases in others even without true absolute change. Third, effect sizes are inherently relative to the rest of the community, meaning an observed increase reflects a shift in ratios rather than an absolute change. Consequently, applying naive tests that do not account for compositionality can yield greatly inflated false findings (Nearing et al., 2022). Use of ratios is a common way to circumvent the need to know the total microbial load. However, the interpretation of results becomes dependent on the reference used. Both single taxon and compound references (such as the geometric mean) must be approximately stable across conditions to anchor the shifts of all other taxa (Morton et al., 2019).

2.1.2. Zero inflation and sparse data

Microbiome count tables are often highly sparse. In a typical dataset, more than half of entries can be zeros (Yang & Chen, 2022). These zeros arise for different reasons: some are structural zeros, meaning the taxon is truly absent from that environment or group. For instance, a pathogen present in diseased subjects but completely absent in all healthy controls would constitute structural zeros in the healthy group (Nearing et al., 2022; Yang & Chen, 2022). Other zeros are sampling zeros, which can occur when a taxon is present at very low abundance but was missed by sequencing due to insufficient depth (Yang & Chen, 2022). Distinguishing between these types matters because structural zeros carry information, whereas sampling zeros reflect stochasticity and technical detectability (Yang & Chen, 2022). Zeros also pose a problem when employing ratios and logarithms, because the outputs are undefined. They must therefore be filtered or imputed in such cases. However, prevalence filtering biases inference by preferentially removing rare but potentially group-specific taxa (Morton et al., 2019). When structural zeros exist, filtering can eliminate the biological signal itself, and when sampling zeros dominate, filtering artificially inflates the relative proportions of retained taxa (Morton et al., 2019). Pseudocount addition (for example, adding a 1 to all counts) introduces an arbitrary, non-biological mass to zero counts, disproportionately inflating rare taxa and distorting log-ratios (Weiss et al., 2017).

2.1.3. Sequencing depth differences and variability

Even when microbes are detected, their observed abundances can vary over several orders of magnitude between samples (Yang & Chen, 2022). This large within-group variability reduces statistical power (probability to reject the null hypothesis when it is truly false) to detect between-group differences, because true differential signals can be masked by heterogeneity (Yang & Chen, 2022). Such heterogeneity can reflect biological and environmental variation as well as technical factors. Samples often have widely varying sequencing depths, and deeper sequencing increases observed counts for most taxa irrespective of the variable of interest. As a result, raw count comparisons can confound true microbial shifts with technical sampling effort. Moreover, when sequencing depth differs between groups, rare taxa may appear differentially abundant purely because one group has more reads to detect them (Yang & Chen, 2022). In such settings, normalization leaves residual bias because depth-dependent detection determines which taxa are observed as non-zero. Taxa undetected in shallow samples remain zero after normalization, while the same taxa may be detected and rescaled in deeper samples.

2.1.4. Misspecification of statistical distribution

The question of which statistical distribution best describes microbiome counts also remains contentious (Nearing et al., 2022). Different DAA methods make different assumptions, spanning count generalized linear models (GLMs) such as Poisson or negative binomial, mixture and semi-continuous models such as zero-inflated log-normal and beta-binomial, and compositional or non-parametric approaches based on ranks (Nearing et al., 2022). If the chosen model is a poor fit to the true data generating process, inference can be misleading. A major consequence of model misspecification is false positive findings and loss of power, because an inappropriate model may fail to detect real shifts (Nearing et al., 2022; Hawinkel et al., 2020). Ultimately, no single distribution fits all datasets.

2.1.5. Complex experimental designs

Microbiome studies often involve complex designs with many potential confounders and technical sources of variation. Cohort factors like diet, age, genetics, medication, and environment can all influence the microbiome independently of the condition of interest (Vujkovic-Cvijin et al., 2020). Unaccounted confounding can lead to spurious associations, for example, a microbiome difference between two patient groups might be due to dietary differences rather than the disease being studied. It is therefore necessary to adjust for relevant covariates to reach valid conclusions (Yang & Chen, 2022). Microbiome datasets are also affected by batch effects arising from DNA extraction kits, PCR and library preparation runs, sequencing platforms, repeated measures, or systemic cohort differences. If these technical factors align with the biological groups of interest, they can produce differential signals that are not biological. However, not all DAA tools support multi-covariate modeling (Yang & Chen, 2022).

2.2. Available tool performance

There are many DAA tools available for microbiome data, each built on distinct statistical assumptions and therefore exhibiting different performance characteristics across study designs and data settings. In this overview and benchmark, we cover the most widely used and extensively benchmarked DAA tools, differing by their underlying assumptions and inferential targets (e.g., transformed abundance linear models, permutation-based tests, and reference frame approaches), including Microbiome Multivariable Associations with Linear Models 2 (MaAsLin2) (Mallick et al., 2021); Analysis of Composition of Microbiomes with Bias Correction 2 (ANCOM-BC2) (Lin & Peddada, 2023); Linear Models for Differential Abundance (LinDA) (Zhou et al., 2022); Linear Decomposition Model (LDM) (Hu & Satten, 2020); ANOVA-Like Differential Expression 2

(ALDEx2) (Fernandes et al., 2014); Count Regression for Correlated Observations with the Beta-binomial (corncob) (Martin et al., 2020); Logistic Compositional Analysis (LOCOM) (Hu et al., 2022); and ZicoSeq (Yang & Chen, 2022). It is important to note that different benchmarks test DAA tools under different data settings, and no benchmark evaluates every single tool mentioned, therefore the strengths and weaknesses of each one may not be directly comparable.

2.2.1. MaAsLin2

MaAsLin2 is a multivariable, per-feature association framework in which microbiome features are tested against variables using regression models, with optional random-effects to accommodate repeated measures designs (Mallick et al., 2021). In applied benchmarking settings, MaAsLin2 is typically evaluated as a pipeline using total sum scaling (TSS) to relative abundances, followed by a pseudocount with log transformation and fitting of linear models on log-transformed abundances (Cappellato et al., 2022; Mallick et al., 2021; Yang & Chen, 2022). In repeated-measures simulations with subject random effects, its false discovery rate (FDR) is comparatively low compared to several alternatives; mild inflation is reported in a univariate setting but not in multivariable simulations (Mallick et al., 2021). Under relatively balanced compositional changes, FDR can be close to nominal (researcher-specified target level, usually 5%) (Yang & Chen, 2023). MaAsLin2 can appear conservative under small sample size and low differential taxa abundance, showing comparatively favorable FDR control while suffering markedly low recall (probability of detecting an effect when it exists) (Cappellato et al., 2022). However, calibration (control of nominal FPR under the global null) deteriorates under strong or unbalanced compositional effects, and TSS-based pipelines degrade in FDR as the fraction of truly differential taxa increases (Yang & Chen, 2022; Yang & Chen, 2023). Performance worsens in confounded low diversity settings (Yang & Chen, 2022). Alternative normalization can improve FDR relative to default TSS, but not to the level of LinDA under the same challenging settings (Yang & Chen, 2023). During the preparation of this thesis, a newer version, MaAsLin3, had been published (Nickols et al., 2026). Instead of a single linear model, it separately models prevalence via a logistic model and abundance via a linear model. In the debut benchmark, MaAsLin3 improved precision (proportion of positive findings that are actually correct) over MaAsLin2 while maintaining similar overall recall, although at small sample sizes this gain in precision came with some loss of recall (Nickols et al., 2026). Because of its recency, independent benchmarks of MaAsLin3 have yet to come out.

2.2.2. ANCOM-BC2

ANCOM-BC2 is a regression-based differential abundance framework intended to estimate associations while accounting for taxon-specific bias factors, such as the ones discussed in section

2.1.1 (Lin & Peddada, 2023). In the ANCOM-BC2 debut study, Lin & Peddada (2023) describe a zero-handling procedure comprising structural zero identification, a default prevalence filter that removes taxa present in fewer than 10% of samples, and an optional pseudocount sensitivity analysis (SS filter) in which a sensitivity score is computed across different pseudocounts and taxa, whose abundance shift significance changes with pseudocount choice are declared non-significant (Lin & Peddada, 2023). In the debut evaluation, two modes are assessed: with or without the SS filter. Across simulated settings, including dense global shifts, covariate-adjusted and multi-group designs, and repeated-measures structures, the SS-filtered variant maintains FDR close to nominal but with power loss, whereas the no-filter variant is more powerful yet exhibits inflated error rates that worsen with increasing sample size and in repeated-measures settings (Lin & Peddada, 2023). Both variants can lose power under extreme dense shift settings (Lin & Peddada, 2023). Independent covariate-adjusted benchmarks show that under balanced covariates with overlapping causal taxa, ANCOM-BC2 is reported to control FDR worse than ANCOM-BC (He et al., 2024). Additional simulations report inflated type I error, including under the global null, across varying effect sizes, bias structures, prevalence filtering, and sample sizes (Satten et al., 2025). A more conservative ANCOM-BC2-Robust variant, which applies an alternative FDR adjustment to the same p-values, is described as achieving improved calibration but at the cost of reduced power in some settings (Satten et al., 2025).

2.2.3. LinDA

LinDA is a regression-based differential abundance framework that runs taxon-wise linear models on centered log-ratio (CLR) transformed outputs, with an explicit reference effect correction (estimated across taxa under the assumption that fewer than half of taxa are truly differential) to map coefficients to absolute log-fold changes (Zhou et al., 2022). LinDA includes an imputation for zeros and a pseudocount pathway, and it switches to imputation when library size is associated with the covariate, as a guard against depth confounding (Zhou et al., 2022). In its debut evaluation, LinDA is reported to achieve high recall with competitive error control across a range of simulation settings, including confounding, within-subject correlation, zero inflation, and library size heterogeneity, and is described as effective for both cross-sectional and longitudinal studies (Zhou et al., 2022). In an independent correlated data benchmark, LinDA is repeatedly identified as the most FDR robust method for time-effect inference in both high and low diversity simulations, controlling FDR to target without sacrificing power and outperforming MaAsLin2 and LDM in this setting (Yang & Chen, 2023). However, LinDA's calibration is less stable across all settings. In the same correlated data study, LinDA shows moderately inflated FDR under the global null in low diversity simulations, despite controlling FDR under the global null in high diversity simulations

(Yang & Chen, 2023). Under interaction bias tests, where one taxon's abundance changes because of the presence of another taxon, LinDA exhibits inflated FDR even under the null, with miscalibration worsening as the number of taxa and effect sizes increase, and with less stringent rare taxa filtering further exacerbating error inflation (Hu et al., 2023). In covariate-adjusted null benchmarks, LinDA is reported to show good FDR control when run with prevalence filtering, ranking behind LOCOM (He et al., 2024). In contrast, in another signal imputation benchmark, LinDA does not meet high recall with controlled false discovery highlighted by the study's top-performing methods (Wirbel et al., 2024).

2.2.4. LDM

LDM is a regression-based differential abundance framework that supports both global testing and OTU-specific inference with covariate adjustment, using permutation-based procedures for significance testing (Hu & Satten, 2020). In its debut evaluation, LDM's global test is reported to achieve near-nominal type I error when confounders are adjusted for, whereas omitting confounder adjustment leads to substantial error inflation (Hu & Satten, 2020). For OTU-specific inference, LDM is reported to control FDR while detecting associated taxa across simulated settings, and an omnibus variant is recommended when signals may concentrate in either abundant or rare taxa (Hu & Satten, 2020). However, LDM's calibration deteriorates as compositional effects become strong, unbalanced, or dense. In low diversity simulations, LDM shows FDR inflation in some settings, particularly when differential taxa are abundant (Yang & Chen, 2022). Under strong and unbalanced shift tests, FDR is reported as acceptable only at low signal density and worsens as the fraction of truly differential taxa increases (Yang & Chen, 2022; Yang & Chen, 2023). In covariate-adjusted benchmarks, the original LDM exhibits inflated FDR, whereas an LDM variant incorporating a centered log-ratio transformation (LDM-CLR) is described as markedly better calibrated and second only to LOCOM (He et al., 2024).

2.2.5. LOCOM

LOCOM is a differential abundance framework that tests covariate associations using a logistic regression framework designed to respect compositional constraints and taxon-specific experimental bias, with inference based on permutations (Hu et al., 2022). In its debut evaluation, LOCOM is reported to have the highest recall among methods that control FDR (Hu et al., 2022). LOCOM is also described as robust in experimental bias stress tests, remaining essentially unaffected when bias factors differ between causal and non-causal taxa, while competing methods show substantial FDR inflation (Hu et al., 2022). In a study incorporating interaction bias, LOCOM is reported to maintain FDR control across tested interaction bias magnitudes, whereas several

alternative DAA procedures show marked inflation even in the absence of experimental bias, highlighting robustness to compositional distortion and model mismatch (Hu et al., 2023). Independent covariate-adjusted benchmarks similarly report LOCOM as having the strongest FDR control, followed by LDM-CLR and LinDA (He et al., 2024). However, performance is less favorable in some settings. In continuous and binary exposure simulations reported by Lin & Peddada (2023), LOCOM's FDR is described as increasing with sample size, and power is reported to be very low at small sample sizes. LOCOM is also noted as not being evaluated in multi-group comparisons in that study due to lack of support for multi-group designs (Lin & Peddada, 2023).

2.2.6. ALDEx2

ALDEx2 is a compositional differential abundance framework, where the procedure is: sample many plausible compositions, transform each sample with the centered log-ratio (CLR), then do per-feature inference on CLR values rather than raw counts (Fernandes et al., 2014). ALDEx2 reports within- and between-condition CLR-based summaries along with an effect size, and a p-value from a test (commonly Welch's t-test or Wilcoxon), with the intuition that low count features are dominated by noise (Fernandes et al., 2014). A practical constraint of this design is that inference is inherently relative (log-ratio): inference is on a feature's abundance relative to the geometric mean of the community, not absolute microbial load changes. In practice, ALDEx2 is often treated as limited for covariate-adjusted association testing unless its GLM variant is used. Accordingly, some studies distinguish ALDEx2(Wilcox) for two group comparisons and ALDEx2(glm) when confounder adjustment is required (Yang & Chen, 2022). Across independent evaluations, ALDEx2 is consistently characterized by conservative error control, typically maintaining Type I error or FDR close to nominal levels but at the cost of reduced recall (Calgaro et al., 2020; Cappellato et al., 2022). ALDEx2 is grouped among methods that reliably control false discoveries, yet it is also flagged as having low recall in a meta-analysis where few methods achieved both good calibration and high recall (Wirbel et al., 2024). In small sample size settings, ALDEx2 can retain reasonable precision and recall relative to other conservative methods (Cappellato et al., 2022). However, ALDEx2 shows inflated FDR and poor recall when effect sizes are large in certain causal set settings, indicating that its calibration can depend on both signal strength and which taxa are truly affected (Hu et al., 2023).

2.2.7. Corncob

Corncob is a regression framework that models observed counts using a beta-binomial distribution to accommodate overdispersion, with covariates being modeled through link functions for the mean (differential abundance) and, optionally, the overdispersion and variability component

(Martin et al., 2020). Inference is implemented via multiple hypothesis testing options, including Wald and likelihood-ratio tests as well as parametric bootstrap versions of each (Martin et al., 2020). In its debut evaluation, corncob's large sample inference is reported to be anti-conservative at very small sample sizes, whereas parametric bootstrap testing recovers near-uniform null p-values and controls type I error in small sample size settings. Adequate calibration is reported for all procedures at sample sizes larger than 30, with bootstrap-based inference recommended below this threshold (Martin et al., 2020). Independent benchmarks indicate that corncob's calibration and power are data setting dependent. In multi-criteria evaluations spanning 16S and whole metagenome shotgun sequencing (WMS) data, corncob is described as conservative in WMS configurations but more liberal in 16S configurations, with additional practical issues including NA p-values and moderate computational cost relative to the fastest methods (Calgaro et al., 2020). In other benchmarks, corncob can achieve high precision-recall and recall in some settings but is also reported to fail FDR control in others, which indicates setting-dependent calibration (Cappellato et al., 2022). Benchmarks stressing dense or highly unbalanced compositional shifts report FDR inflation for corncob, particularly as the fraction of truly differential taxa increases (Lin & Peddada, 2023; Yang & Chen, 2022). In other studies, corncob is grouped among methods exhibiting excess false positives and reduced power at smaller sample sizes (Wirbel et al., 2024).

2.2.8. ZicoSeq

ZicoSeq is a regression-based differential abundance framework that combines reference-taxa normalization with permutation-based inference (Yang & Chen, 2022). The method defines taxon-specific responses relative to a data-driven reference set assumed to be largely stable across conditions. Association testing is performed using a linear model-based omnibus statistic, with significance assessed via permutation. ZicoSeq incorporates an empirical-Bayes beta-mixture model to infer underlying taxon proportions and handle zero inflation and depth-related artifacts (Yang & Chen, 2022). In its debut benchmark, ZicoSeq is reported as broadly robust and high power across a range of settings, with FDR control and recall described as comparable to or better than the best performing comparators in the tested settings (Yang & Chen, 2022). A highlighted strength is sequencing depth confounding: ZicoSeq is reported as the only evaluated method that simultaneously controls FDR at the target level and maintains high power under depth confounding (Yang & Chen, 2022). However, the authors report residual limitations, including mild FDR inflation under strong compositional effects (Yang & Chen, 2022). Independent evaluation suggests performance is simulation dependent. In a signal-imputation benchmark, ZicoSeq ranks among top methods under negative-binomial simulations, but this ranking is reported to be largely discordant

with rankings under other simulation methods and implanted signal settings, indicating sensitivity to the assumed data generation distribution (Wirbel et al., 2024).

2.3. Summary

Microbes are valuable biomarkers in both clinical fields and ecology. However, because microbial 16S rRNA sequencing data has troublesome characteristics and is heavily confounded, statistical tools must be specifically adapted in order to validly infer abundance shifts. The literature indicates that DAA tool performance is condition-dependent: methods show different power and error profiles across settings of zero inflation, effect size, library size heterogeneity, and design complexity (Table 1).

Table 1. Comparison of DAA tools' success and failure modes

Method	Success modes	Failure modes
MaAsLin2	Acceptable FDR under balanced and sparse signals.	Inflated FDR under strong or dense compositional shifts; power loss with dense signals
ANCOM-BC2	SS-filtered mode shows near-nominal FDR across many settings	SS-filtered mode loses power; no-filter mode inflates error with increasing n
LinDA	High recall with covariate adjustment; robust to library-size heterogeneity	Inflated FDR under null settings in low diversity settings; interaction-bias recall
LDM	Near-nominal type I with confounder adjustment; CLR variant improves calibration	Error inflation without confounder adjustment; low diversity failures
LOCOM	Strong FDR control under compositional null; high recall among calibrated methods	Low power at small n; limited evaluation in multi-group settings
ALDEx2	Conservative type I and FDR control; stable in small samples	Low recall; power loss under realistic effects
corncob	Flexible mean-variance modeling; good recall in some settings	Anti-conservative at very small n; FDR inflation under dense shifts
ZicoSeq	High power with FDR control under sequencing-depth confounding	Mild FDR inflation under strong compositional effects

Tools that control false discoveries often have low recall, particularly in small sample or dense signal settings, whereas more powerful approaches can suffer from miscalibration under strong or unbalanced compositional shifts. This becomes especially visible when moving beyond simple two-group comparisons into covariate or repeated-measures settings, where several frameworks are not even applicable. Therefore, to reduce the burden of selecting an appropriate DAA tool for every dataset, we designed PURSUE, a reference-normalized two-part prevalence and abundance model

with permutation-based inference that aims to overcome troublesome microbial data characteristics and retain calibration and power across different dataset settings.

3. METHODS

3.1. PURSUE procedure

The workflow of PURSUE is divided into five modules: pre-filtering of low information taxa, selection of a stable reference set used to construct log-ratios, construction of two regression model responses that separately model presence and abundance and fitting of each response, Freedman-Lane permutation testing, and p-value combination of the two model arms. Each stage is described in detail in the following subsections. The novelty of PURSUE lies in applying reference normalization within the abundance arm of the two-part model and using bootstrap-stability reference selection with adequacy checks and expansion limits. A high level overview of the package architecture and the order in which these stages are orchestrated is provided in Appendix 1, Figure A1.1.

3.1.1. Pre-filtering

Prior to reference selection and testing, taxa may be removed using two criteria: minimum proportion of samples the taxon is detected in and minimum total count summed across samples. Filtering rare taxa can decrease noise, improve reproducibility, and lower multiple testing burden, and has been shown to increase DAA tool performance in some scenarios (Cao et al., 2021; Nearing et al., 2022). An important detail of filtering is that it should be independent of the tested variables. Conceptually, in empirical p-value generation, the null distribution is obtained by permuting the labels of the variable of interest, relative to the observed data points, so that the tested variable has no true effect. If features are kept based on their association with the tested variable, the null distribution is no longer fairly sampled and further testing is likely to see inflated false positives and poor calibration (Bourgon et al., 2010; Nearing et al., 2022). In PURSUE, filtering is disabled by default, and users may supply the criteria thresholds.

3.1.2. Reference selection

In DAA, a reference frame refers to a baseline taxon or set of taxa against which changes in other taxa are calculated and interpreted (Morton et al., 2019). This is particularly useful for compositional data, because ratios cancel out the unknown total abundance bias:

$$\log\left(\frac{X_{ij}}{X_{i,ref}}\right) = \log\left(\frac{p_{ij}A_i}{p_{i,ref}A_i}\right) = \log\left(\frac{p_{ij}}{p_{i,ref}}\right) = \log(p_{ij}) - \log(p_{i,ref})$$

Where X_{ij} and $X_{i,ref}$ are count abundances of taxon j and the reference taxon in sample i , respectively, A_i is a per sample multiplicative factor (biomass, sequencing depth, or any other

unknown scaling), and p are the taxon proportions. Reference frame reasoning has previously been applied in many DAA methods (Brill et al., 2022; Fernandes et al., 2014; Hu et al., 2022; Morton et al., 2019; Yang & Chen, 2022; Zhou et al., 2022).

In PURSUE, reference taxa are selected using a DACOMP-inspired stability score (Brill et al., 2022). First, a candidate set of taxa is defined. This is usually every taxon retained after pre-filtering, unless a maximum number of taxa is specified, in which case a subset of taxa with the most total counts across samples is selected. Then, all unique pairwise log ratios between candidate taxa are constructed (Fig. 1), such that for taxa j and k in sample i

$$y_{i,jk} = \log(X_{ij} + c) - \log(X_{ik} + c)$$

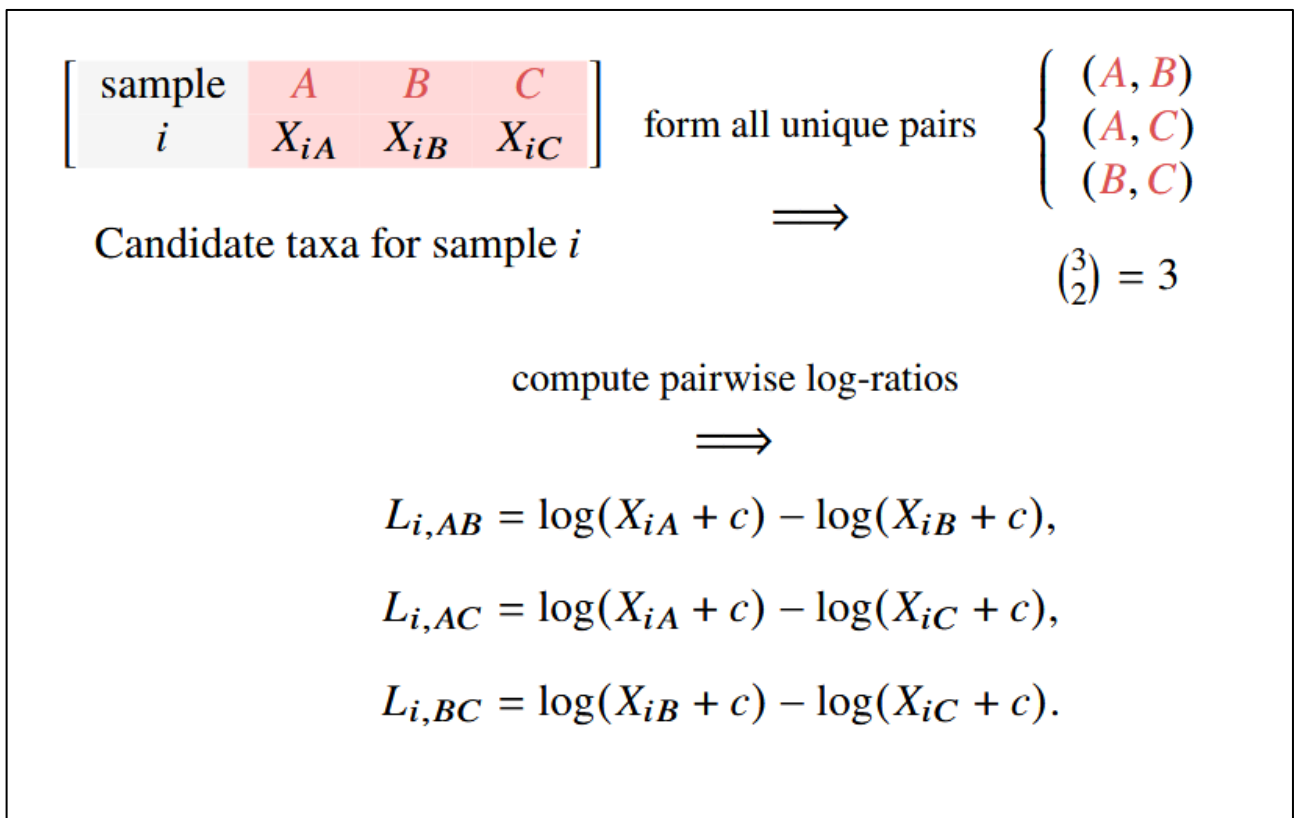


Figure 1. Formation of unique pairwise log ratios between candidate taxa.

Then, a nuisance-only design matrix is constructed by removing the tested term from the model formula. A design matrix is a sample-by-variable matrix whose columns represent the covariates included in the analysis, as represented by X_{full} in Fig. 2. If the analysis contains additional adjustment variables, these are retained in the nuisance matrix (Fig. 2a). Conversely, if the tested term is the only variable in the formula, the nuisance matrix is reduced to an intercept-only column of ones (Fig. 2b). To account for differing sequencing depth, PURSUE computes the logarithm of

the total library size N_i and includes it as an additional covariate, so that it enters the design matrix X_{full} as a nuisance variable. This is enabled by default but may be disabled.

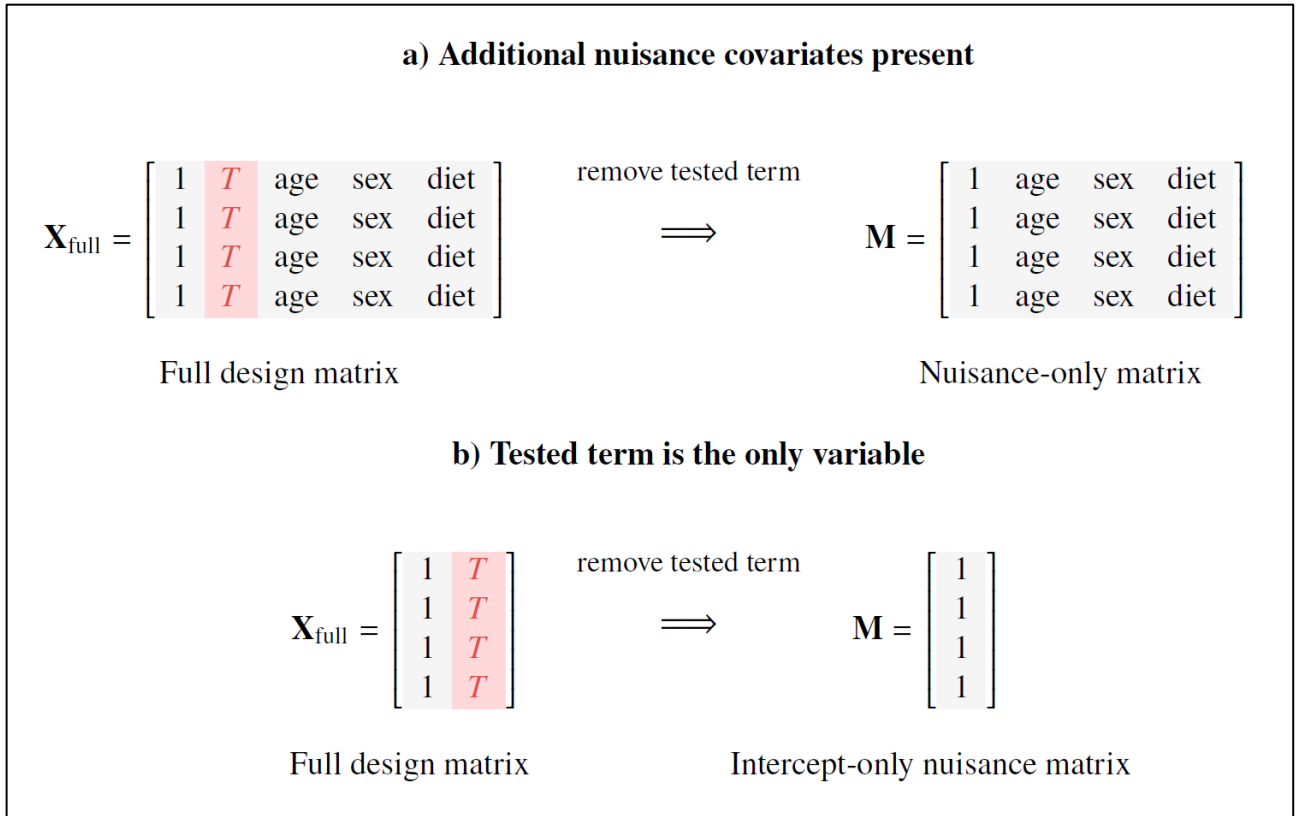


Figure 2. Construction of the nuisance-only design matrix. T refers to the tested term.

What constitutes the tested term or a nuisance variable depends on the experimental design. The tested term is the variable whose association with microbial abundance is of primary interest, whereas nuisance variables are covariates that may influence microbial abundance but are not themselves of interest and are included to control for potential confounding. For example, in a drug-control trial assessing the effect of a treatment on the gut microbiome, the variable of interest would be treatment assignment (drug versus placebo), while cohort factors such as age, sex, or diet would be treated as nuisance variables, as they may influence microbiome composition but are not the targets of inference. Conversely, in an experiment assessing age-related microbiome composition changes, age would constitute the tested term.

When the nuisance-only matrix is constructed, each pairwise log-ratio vector is then residualized with respect to this nuisance matrix by ordinary least squares (OLS) (Fig. 3).

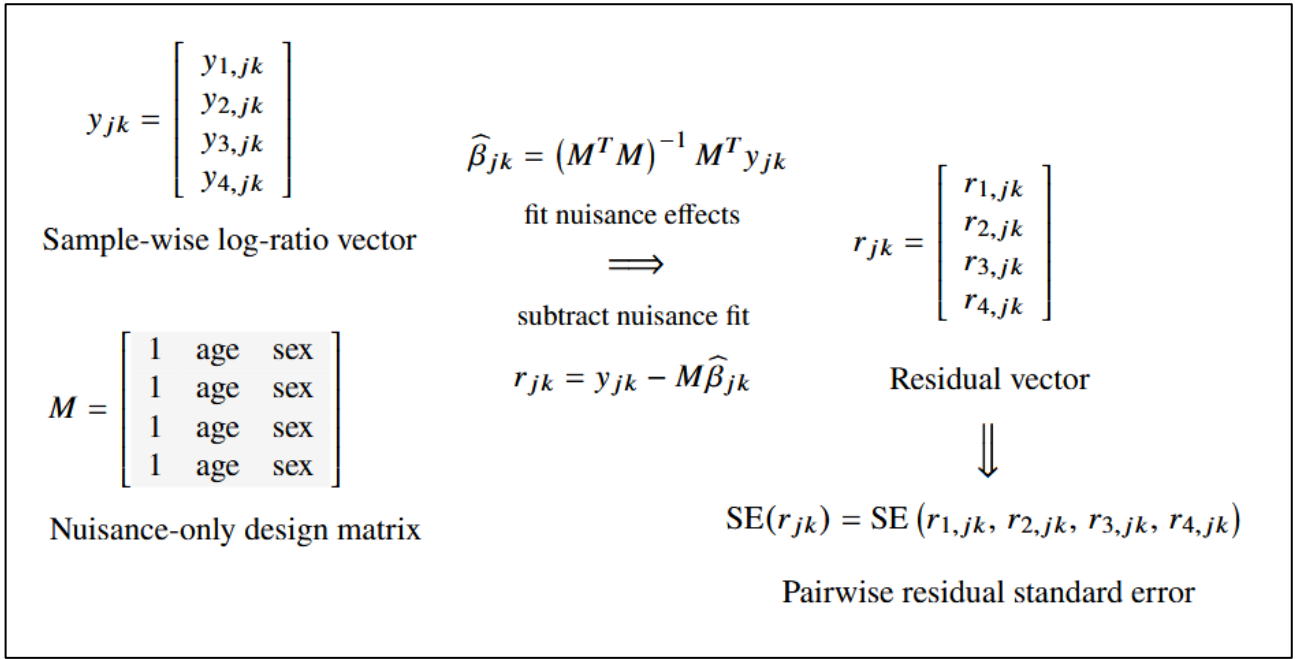


Figure 3. Obtaining pairwise residual standard error after nuisance adjustment by OLS. Note that the final stability metric is the median of standard errors across all pairs involving the particular taxon.

If y_{jk} denotes the sample-wise log-ratio vector for the taxa pair j and k , and M denotes the nuisance-only design matrix, the fitted nuisance effects are obtained from

$$\widehat{\beta}_{jk} = (M^T M)^{-1} M^T y_{jk}$$

and the residual vector across all samples is

$$r_{jk} = y_{jk} - M\widehat{\beta}_{jk}$$

When no nuisance variables are present, this reduces to intercept-only regression, so the fitted value is the sample mean of the log-ratio vector and the residual vector across all samples is simply

$$r_{jk} = y_{jk} - \bar{y}_{jk}$$

where $\bar{y}_{jk}$ is the mean of y_{jk} across samples. When nuisance covariates are present, the regression step removes nuisance associated variation. For each unique taxon pair, the residual standard error is then computed:

$$\sigma_{jk} = \sqrt{\frac{\sum_{i=1}^n r_{i,jk}^2}{n - q}},$$

where n is the number of samples, and q is the number of columns in the nuisance-only design matrix. Then, for each taxon, the median of residual standard error across all pairs that have that

specific taxon in the numerator or denominator is taken. Taxa are then ranked in ascending order of this median, so that smaller values correspond to more stable taxa.

Upon obtaining this ranking, the stability of subsets of top-ranking taxa is evaluated using bootstrap resampling, with the intention of selecting the smallest stable subset (Fig. 4).

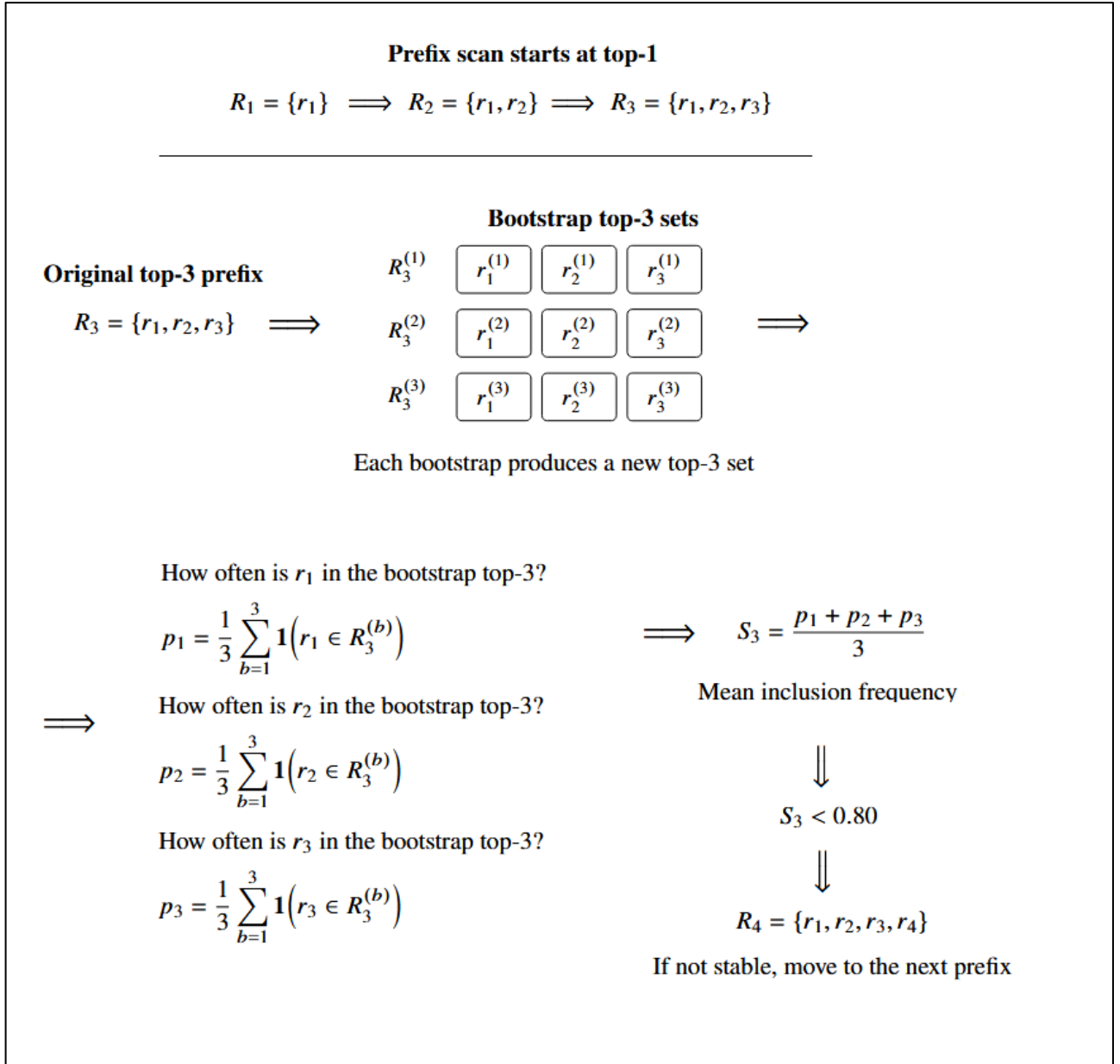


Figure 4. The bootstrapped top- k taxa overlap procedure, which culminates in a mean inclusion score, indicating stability of the prefix.

Top- k prefixes are defined as the first k taxa in the full-data median stability ranking, starting from $k = 1$. If the full data ranked candidate list, containing every taxon in the analysis, is denoted as

$$r_1, r_2, \dots, r_m$$

then the top- k prefix, containing only the top- k taxa from the original ranking is

$$R_k = \{r_1, r_2, \dots, r_k\}$$

A bootstrap distribution of reference taxon stability rankings is then constructed by repeatedly resampling the original samples with replacement, meaning a particular sample can enter the set multiple times. For each bootstrap replicate, the entire scoring procedure (pairwise log-ratios, nuisance residualization, and per-taxon median residual SD) is recalculated using the resampled count table, and candidate taxa are ranked again according to their bootstrap median instability scores. This produces one bootstrap ranking $R^{(b)}$ per replicate b .

For a given prefix size k , the full data top- k reference set R_k is compared against the top- k reference set obtained in each bootstrap replicate. If $R_k^{(b)}$ denotes the top- k reference set from bootstrap replicate b , then the overlap between the full-data and bootstrap top- k sets is

$$O_k^{(b)} = \frac{|R_k \cap R_k^{(b)}|}{k}$$

This value represents the proportion of taxa selected in the original top- k prefix that also appear in the bootstrap top- k prefix. Values close to 1 indicate that the same taxa are repeatedly recovered among the most stable candidates, whereas lower values indicate that the composition of the top- k set is unstable.

The overall bootstrap stability of prefix R_k is then calculated as the average overlap across all B bootstrap replicates:

$$S_k = \frac{1}{B} \sum_{b=1}^B O_k^{(b)} = \frac{1}{B} \sum_{b=1}^B \frac{|R_k \cap R_k^{(b)}|}{k}$$

A prefix is considered sufficiently stable when its bootstrap stability reaches a defined threshold, 0.8 by default, chosen empirically (Fig. A2.1):

$$S_k \geq S_{\min}$$

Later, PURSUE evaluates whether each top- k prefix provides sufficient denominator mass (Fig. 5).

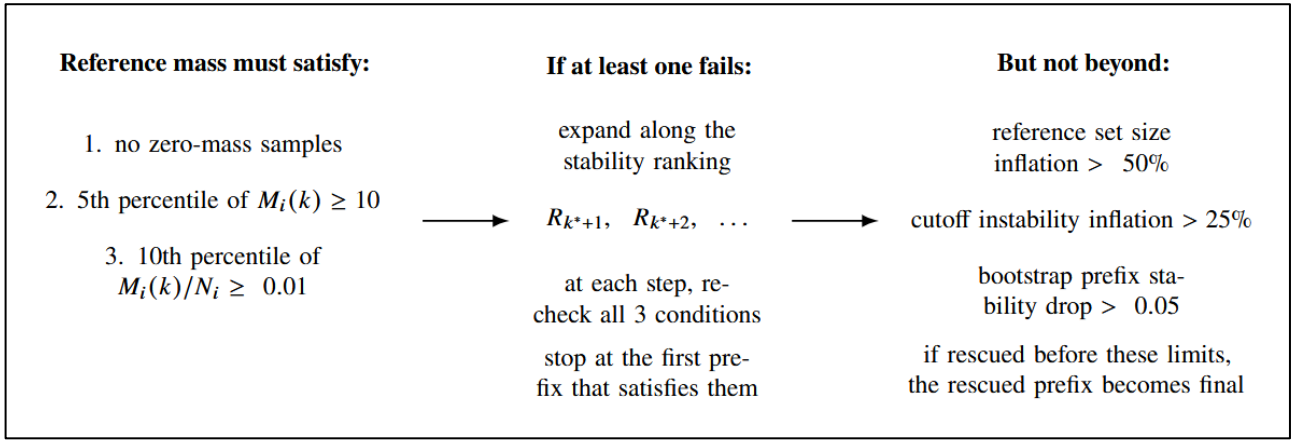


Figure 5. Minimum reference mass conditions and expansion limits when these conditions are not satisfied.

For a given prefix R_k , the sample-wise reference mass is computed as

$$M_i(k) = \sum_{r \in R_k} X_{ir}$$

and the total library size is

$$N_i = \sum_{j=1}^m X_{ij}$$

over taxa retained after filtering. For each prefix, the first condition is that no sample may have zero reference mass. Second, it calculates a lower-tail absolute count quantile: by default, the 5th percentile of the sample-wise reference masses must be at least 10 reads. Third, it calculates a lower-tail relative reference-mass quantile: by default, the 10th percentile of the sample-wise ratios $M_i(k)/N_i$ must be at least 0.01. These were chosen as pragmatic defaults for small (absolute mass) and large (relative mass) library sizes.

If at least one of these conditions fails, PURSUE tries to expand the reference set by moving forward along the stability ranking:

$$R_{k^*+1}, R_{k^*+2}, \dots$$

At each expanded prefix $R_{k'}$, the three adequacy conditions are reevaluated. Expansion stops successfully when it finds the first expanded prefix satisfying all three conditions.

However, expansion is limited so that the reference set may not grow excessively, and its stability must not decrease substantially relative to the base prefix k^* . For an expanded prefix $k' > k^*$, PURSUE requires that the reference set not expand by more than 50%, that the per-taxon instability score at the prefix cutoff (the score of the worst ranked taxon currently in the reference set) not

increase by more than 25%, and that bootstrap prefix stability does not drop by more than 0.05. If all three adequacy conditions are rescued before these limits are exceeded, the rescued expanded prefix becomes the final reference set.

Condition 1 is treated more strictly than the two minimum mass conditions. Because zero reference mass would make the downstream log-ratios undefined, PURSUE is allowed to continue expanding past the limits. Expansion continues until every sample has nonzero reference mass.

If the expansion limits are reached after condition 1 has already been satisfied, but the other two conditions remain unsatisfied, PURSUE retains the largest expanded prefix that did not exceed expansion limits. In the worst case, if condition 1 still cannot be satisfied even after exhausting all candidate reference taxa, PURSUE stops, because at least one sample has zero counts across all candidate reference taxa and cannot be supported by any expanded reference set.

Finally, PURSUE applies a reference cleanup step for binary tested variables (Fig. 6).

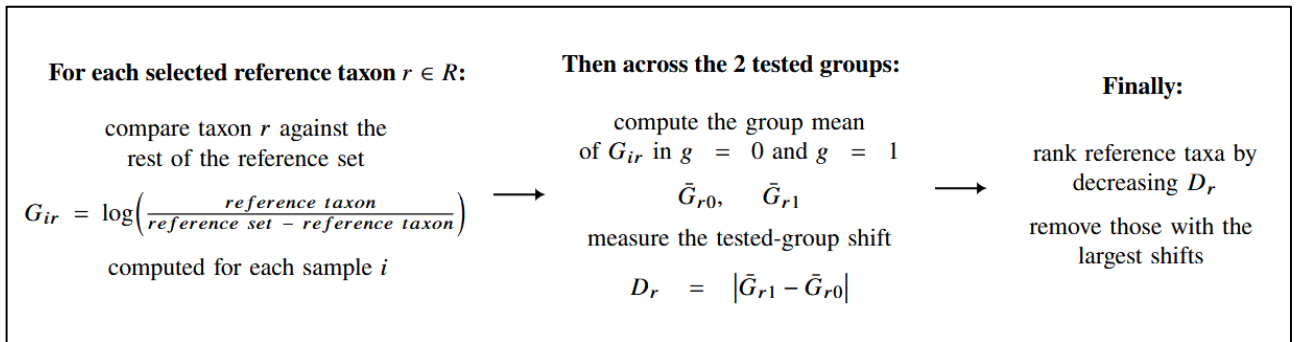


Figure 6. Reference cleanup procedure: scoring and removal of taxa with the largest binary-group shift in relative abundance.

This step is applied after the primary reference set has been selected and is intended to remove reference taxa that still exhibit the strongest apparent tested group shift relative to the rest of the reference set.

For each selected reference taxon $r \in R$, PURSUE compares that taxon against the mass of the remaining selected references. For each reference taxon r in sample i , this log-ratio is computed as the ratio of the abundance of taxon r and the rest of the reference set R that excludes that taxon r :

$$G_{ir} = \log\left(\frac{X_{ir} + c}{\sum_{q \in R, q \neq r} X_{iq} + c}\right)$$

where X_{ir} is the count of reference taxon r , R is the selected reference set, and c is a pseudocount.

If the binary tested variable defines two groups, $g = 0$ and $g = 1$, PURSUE then computes the mean of this log-ratio within each group:

$$\bar{G}_{rg} = \frac{1}{n_g} \sum_{i:T_i=g} G_{ir}$$

where T_i is the tested term group label for sample i , and n_g is the number of samples in group g . The tested-group shift of reference taxon r is then defined as the absolute difference between the two group means:

$$D_r = | \bar{G}_{r1} - \bar{G}_{r0} |$$

Selected references are ranked by decreasing D_r . The references with the largest values are then removed up to the configured cleanup fraction, which is 0.15 by default, chosen empirically (Fig. A2.2). This means that cleanup preferentially removes reference taxa whose abundance relative to the remaining reference mass differs most strongly between the two tested groups.

3.1.3. Two-part model

Taking inspiration from the MaAsLin3 two-part model approach (Nickols et al., 2026), after the final reference set has been selected, the count table is converted into two responses, corresponding to prevalence and positive abundance (Fig. 7). In regression, a response or dependent variable refers to a variable that the model is trying to explain or predict given independent variables, in this case user-provided tested terms and nuisance variables.

Linear regression model responses for only non-reference taxa	
Expected-rarefied prevalence	Reference-normalized abundance
$p_{ij}^{\text{ER}} = 1 - \frac{\binom{N_i - X_{ij}}{d}}{\binom{N_i}{d}}$ Probability that taxon j would still be detected in sample i if all samples were reduced to the same minimum depth d	$A_{ij} = \log(X_{ij}) - \log(M_i)$ Log abundance of taxon j in sample i, normalized by the sample's selected reference mass M_i Defined only for $X_{ij} > 0$

Figure 7. The responses for both the prevalence and abundance arm regression models.

Let $\mathcal{R}$ denote the final selected reference set. For each sample i , the final reference mass is computed as

$$M_i = \sum_{r \in \mathcal{R}} X_{ir}$$

where X_{ir} is the count of reference taxon r in sample i . The total library size is:

$$N_i = \sum_{j=1}^m X_{ij}$$

For both prevalence and abundance responses, reference taxa are excluded because their selection for the reference set already implies the assumption that they are stable. Furthermore, they are excluded to not self-normalize (appear in both sides of the ratio) in the abundance response.

The prevalence response is expected rarefied prevalence, rather than binary prevalence (Hu et al., 2021). As covered in section 2.1.3, a taxon present at low relative abundance is more likely to be observed at high depth than at low depth, even when its true relative abundance is identical. Rarefying raw count tables is a controversial practice, as it results in substantial information loss and adds stochastic sampling noise (McMurdie & Holmes, 2014). However, in PURSUE, these disadvantages are bypassed by analytically calculating the expected probability of detection if samples were rarefied to a common depth and using them as the response in a linear regression model. This means that the original count table is not transformed, nor is sampling noise a concern when probabilities are calculated in closed form. A common rarefaction depth is defined as the minimum total library size across samples in the original unfiltered count table,

$$d = \min_i N_i$$

For each non-reference taxon $j \notin \mathcal{R}$ in sample i , PURSUE computes the expected probability that taxon j would remain detectable after rarefying sample i to depth d . This response can be written as

$$P_{ij}^{\text{ER}} = 1 - \frac{\binom{N_i - X_{ij}}{d}}{\binom{N_i}{d}},$$

The prevalence arm is fitted taxon-wise using linear regression:

$$P_{ij}^{\text{ER}} = \beta_{0j} + \beta_{1j}Z_{i1} + \dots + \beta_{qj}Z_{iq} + \varepsilon_{ij}^P$$

where $Z_{i1}, \dots, Z_{iq}$ denote the covariates included in the prevalence model formula.

The abundance response is constructed for non-reference taxa only on the log-ratio scale relative to the selected reference mass. For each non-reference taxon $j \notin \mathcal{R}$, the abundance response is

$$A_{ij} = \log(X_{ij}) - \log(M_i)$$

A pseudocount is not used here. Instead, A_{ij} is computed only for cells where the taxon count $X_{ij} > 0$ and the reference denominator $M_i > 0$. Otherwise, the abundance response is treated as missing for that taxon-sample pair. The abundance arm is fitted taxon-wise using linear regression:

$$A_{ij} = \alpha_{0j} + \alpha_{1j}Z_{i1} + \dots + \alpha_{qj}Z_{iq} + \varepsilon_{ij}^A$$

Optional depth adjustment is handled through formula expansion with the logarithm of sample depth, just like in reference selection. This expanded formula is used for the abundance arm. In contrast, the prevalence arm does not use this formula expansion, because expected rarefaction already provides depth normalization.

After fitting both arms, Wald statistics are calculated (Fig. 8).

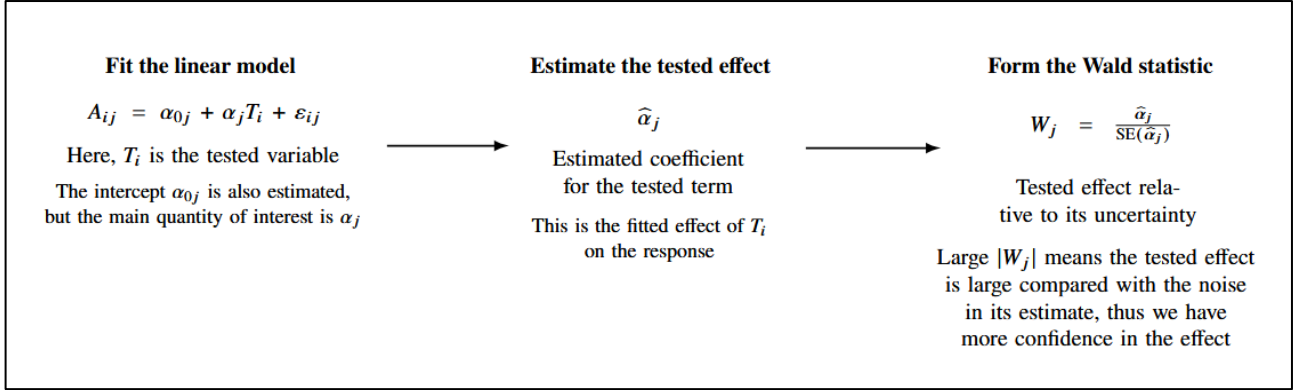


Figure 8. Obtaining the 1-df scalar Wald statistic from a regression model fit.

For a one-degree-of-freedom tested term, the component statistic is the signed Wald statistic,

$$W = \frac{\hat{\theta}}{SE(\hat{\theta})}$$

where $\hat{\theta}$ denotes the tested-term coefficient from the corresponding linear model. Specifically, $\hat{\theta} = \hat{\beta}$ for the prevalence arm and $\hat{\theta} = \hat{\alpha}$ for the abundance arm. Each coefficient $\alpha_{1j} \dots \alpha_{qj}$ corresponds to one column of the design matrix Z , which contains both tested and nuisance covariates, while α_{0j} is the intercept. The tested-term coefficient block is the subset of these α 's corresponding to columns of the tested variable: a single coefficient for binary or continuous tested terms, and $(categories - 1)$ coefficients for multi-category tested terms. The Wald statistics are computed on this block only, meaning only the tested term. The standard error of the tested coefficient is estimated from the residual variability left after fitting the linear model, scaled by the uncertainty structure of the model design matrix:

$$SE(\hat{\theta}) = \sqrt{\hat{\sigma}^2 [(Z^T Z)^{-1}]_{\theta\theta}}$$

where Z is the full design matrix containing the tested covariate and any nuisance covariates, the subscript $\theta\theta$ denotes the diagonal position in the product of $(Z^T Z)^{-1}$ corresponding to the tested-term coefficient. For example, if the design matrix Z contains an intercept in the first column and

the tested variable in the second column, then $\theta\theta$ refers to the (2,2) entry of $(Z^T Z)^{-1}$. $\hat{\sigma}^2$ is the residual variance estimated from the fitted model:

$$\hat{\sigma}^2 = \frac{\sum_i \hat{r}_i^2}{n - p}$$

Here, n is the number of samples used in the fit and p is the number of fitted model coefficients. For tested terms that have multiple model coefficients (multi-category factors with categories ≥ 3) PURSUE computes the corresponding quadratic Wald statistic,

$$W = \hat{\boldsymbol{\theta}}^T \hat{\boldsymbol{\Sigma}}^{-1} \hat{\boldsymbol{\theta}}$$

where $\hat{\boldsymbol{\theta}}$ is the tested coefficient block and $\hat{\boldsymbol{\Sigma}}$ is its estimated covariance matrix.

3.1.4. Permutation-based inference

Drawing inspiration from LDM (Hu & Satten, 2020), PURSUE obtains empirical p-values separately for the prevalence and abundance arms using permutation. Instead of permuting the count-metadata correspondence directly (shuffling sample labels), PURSUE uses a Freedman-Lane residual permutation scheme (Freedman & Lane, 1983) (Fig. 9). PURSUE first fits a reduced model containing the nuisance covariates but excluding the tested term, then permutes the residual variation from this model. The resulting null statistics therefore test the effect of the variable of interest after nuisance adjustment. (Freedman & Lane, 1983).

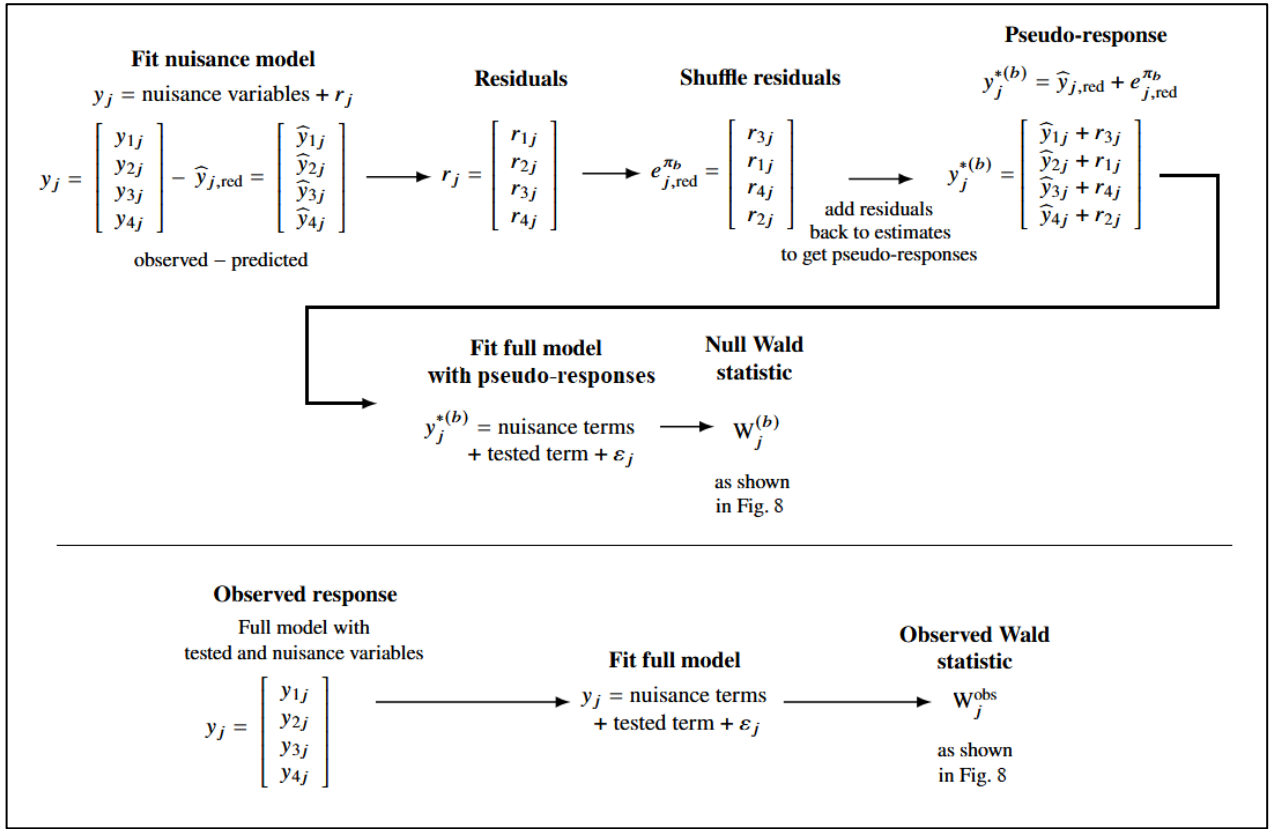


Figure 9. Top row: obtaining a single permuted null Wald statistic from permuted pseudoresponses. Bottom row: obtaining an observed Wald statistic with unpermuted observed data.

Let y_{jc} denote the response vector for taxon j in component c , meaning either the prevalence or the abundance arm. A reduced model excluding the tested term but retaining the nuisance covariates is first fitted, giving reduced model fitted values and residuals

$$\hat{y}_{jc,\text{red}} \text{ and } r_{jc,\text{red}}$$

For permutation replicate b , a null response is generated by permuting the reduced model residuals across samples within the selected permutation scheme (described below) and adding them back to the reduced model fitted values:

$$y_{jc}^{*(b)} = \hat{y}_{jc,\text{red}} + r_{jc,\text{red}}^{\pi_b}$$

where π_b denotes the permutation scheme π and permutation replicate b . Each permuted pseudoresponse is then used as a response in the full model with a full design matrix, which contains both nuisance and tested term columns. From this model fit with permuted responses, for each permutation replicate b , a null Wald statistic $W_{jc}^{(b)}$ is calculated. Conversely, the observed Wald statistic W_{jc}^{obs} is calculated from a full model fit including both nuisance and tested terms with the observed, unpermuted responses.

For empirical p-value calculation, the absolute value of these statistics is used. For tested terms represented by multiple coefficients, PURSUE uses the corresponding quadratic Wald statistic. For each taxon and model arm, the observed statistic is compared with its permuted null statistics. The proportion of permuted null statistics larger than or equal to the observed statistic over all permutations is the two-sided p-value adjusted with +1 in both the numerator and denominator, so that $p_{jc} > 0$ (Phipson & Smyth, 2016):

$$p_{jc} = \frac{1 + \sum_{b \in \mathcal{B}_{jc}} \mathbf{1}(|W_{jc}^{(b)}| \geq |W_{jc}^{\text{obs}}|)}{1 + \mathcal{B}_{jc}}$$

where $|W_{jc}^{\text{obs}}|$ is the absolute value of the signed Wald statistic or quadratic Wald statistic for taxon j in component c , $|W_{jc}^{(b)}|$ is the corresponding statistic under permutation b , and $\mathcal{B}_{jc}$ is the number of permutation replicates that produced valid statistics for that taxon-component pair.

This comparison gives a two-sided (meaning the direction of change is not evaluated) empirical p-value for the tested effect. Components that are unusable in the observed fit, or components with no valid permutation statistics, are assigned an empirical p-value of 1. Failed permutations are excluded from the valid permutation denominator.

For the abundance arm, reduced-model residuals may be winsorized before null and observed responses are generated (Fig. 10).

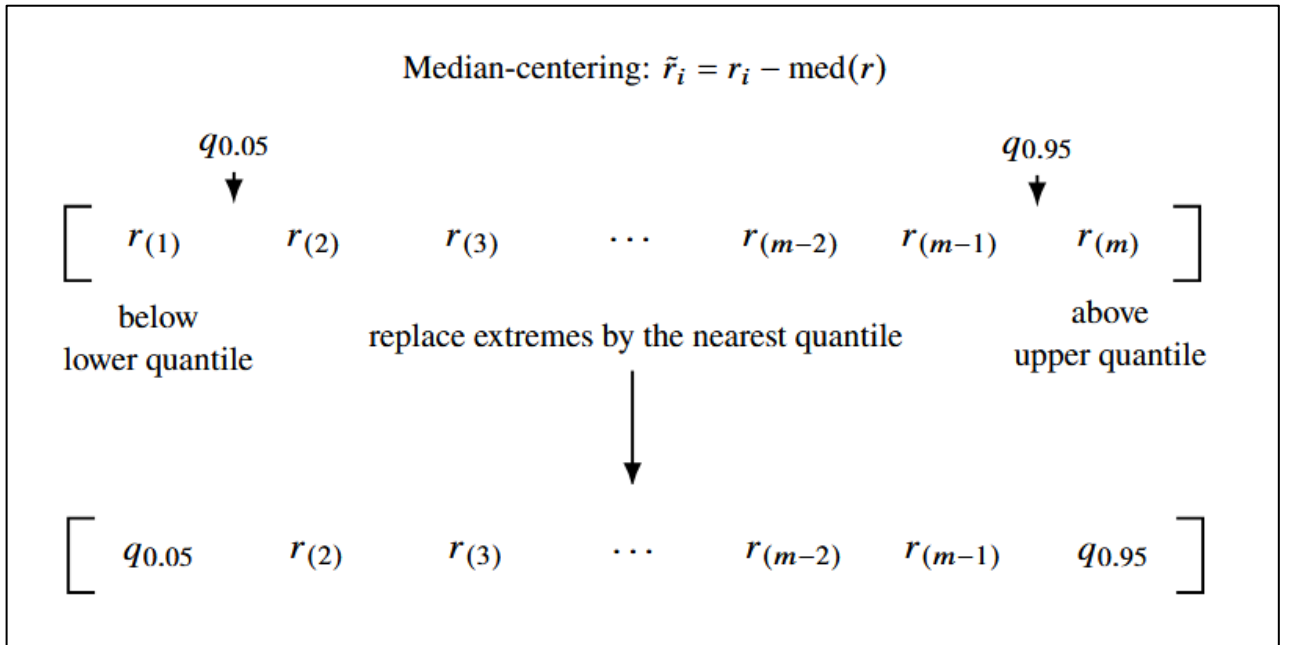


Figure 10. Before null and observed statistic generation, residuals may be winsorized by replacing outliers outside of quantile ranges with the respective quantile values on both tails.

When enabled, residuals are centered around their median and clipped at the configured lower and upper residual quantiles, 0.05 and 0.95 by default, provided that enough residuals are available, 20 by default. The winsorized residuals r_{red}^{wins} are then used in place of r_{red} in null response construction:

$$y^{*(b)} = \hat{y}_{red} + (r_{red}^{wins})^{\pi_b}$$

And the observed response is now constructed similarly to the null response: by first fitting a nuisance-only reduced model, winsorizing the residuals, and adding them back to make the response y_{obs} which is used in the full model fit:

$$y_{obs} = \hat{y}_{red} + r_{red}^{wins}$$

The difference in the observed statistic generation is that the residuals are not permuted. The prevalence arm does not apply residual winsorization, because the response is already bounded.

Exchangeability during residual permutation can be specified in PURSUE to control which residuals are allowed to move between which observations, so that the permutation scheme matches the experimental design (Winkler et al., 2014) (Fig. 11).

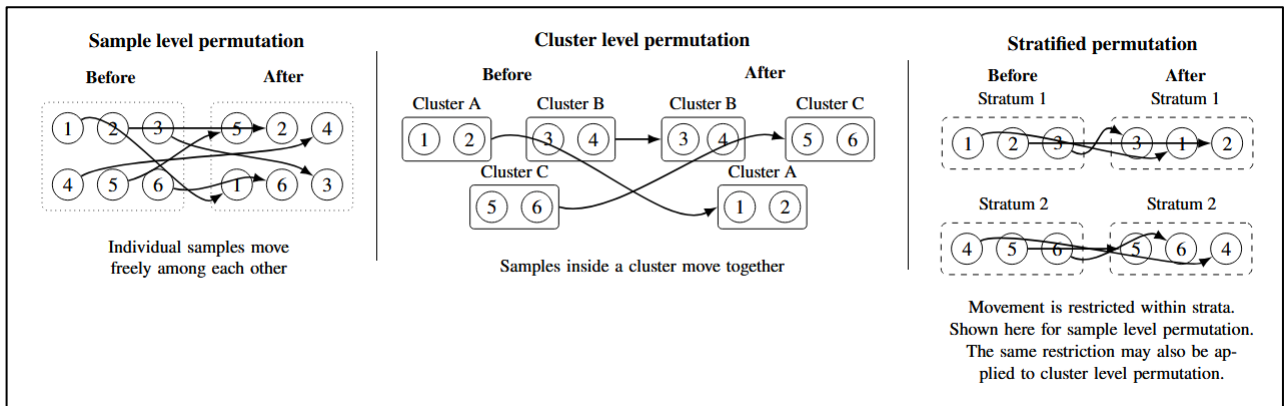


Figure 11. Sample associated residual permutation movement modes used to match the exchangeability structure of the experiment design.

This is controlled along two axes: whether residuals move at the sample level or the cluster level, and whether movement is unrestricted or constrained within exchangeability blocks.

In sample level permutation, each sample-associated residual is treated as its own exchangeable unit. If no cluster or stratum restrictions are supplied, residuals may be shuffled freely across all samples. This is the default setting and is appropriate when samples can reasonably be treated as independent and exchangeable under the null, for example in a simple two-group case-control design without repeated measurements or batch restrictions (Winkler et al., 2014).

Sample level permutation can also be restricted by clusters. In this case, residuals still move sample by sample, but only within the same cluster (Winkler et al., 2014). This mode is appropriate when the tested variable varies within clusters, because permuting whole clusters would destroy the within cluster exposure structure. For example, if several samples come from the same subject and the tested variable differs between time points within that subject, residuals may be exchanged among samples from the same subject rather than across unrelated subjects.

In cluster level permutation, residuals are not moved sample by sample. Instead, whole clusters of related observations move together as a unit (Winkler et al., 2014). A cluster may represent repeated measurements from the same individual, paired samples from the same subject, or multiple technical or biological samples from the same experimental unit. This is appropriate when observations within a cluster are not independent and should not be broken apart during permutation. For example, if several microbiome samples come from the same subject across time and the tested variable is defined at the subject level, the subject can be treated as a cluster so that all residuals from that subject move together.

Strata provide an additional restriction layer that can be supplied on top of either sample level or cluster level permutation. A stratum is a group within which exchangeability is assumed, but across which permutation should not occur (Winkler et al., 2014). Examples include sequencing batch, study site, cohort, donor group, sex, or treatment block. With sample level stratification, individual sample residuals are shuffled only within the same stratum. If both cluster and stratum restrictions are supplied under sample level permutation, residuals are exchanged only within the combined cluster by stratum blocks. For example, if sequencing batch is used as a stratum, samples from batch 1 only exchange residuals with other samples from batch 1, and samples from batch 2 only with batch 2; if subjects are also used as clusters, this exchange is further restricted to samples from the same subject within the same batch.

With cluster level stratification, whole clusters are shuffled only within their stratum (Winkler et al., 2014). For example, if subjects are clusters and study site is the stratum, all samples from a subject move together, but subjects are only permuted with other subjects from the same site. Each cluster must belong to a single stratum, since moving a cluster as a unit is incompatible with splitting its members across strata.

3.1.5. Two-part model p-value aggregation and independent filtering

The taxon-wise empirical p-values from the prevalence and abundance arms are then aggregated into a single overall p-value using the aggregated Cauchy association test (ACAT) (Liu et al., 2019). ACAT was chosen because it provides valid p-value combination under arbitrary

dependence between p-values. This is appropriate here because both prevalence and abundance arms are computed from the same count table. If both components were available for a given taxon, equal weight is assigned to the prevalence and abundance p-values. If only one component is usable, the union p-value is based entirely on that component. If neither component is usable, the union p-value is set to 1. This yields a single union p-value for each taxon, representing the association with the variable of interest through prevalence, abundance, or both.

Similarly to DESeq2 (Love et al., 2014), PURSUE can apply independent filtering to remove low abundance (and thus low information) modeled taxa from the Benjamini-Hochberg (BH) correction set (Fig. 12).

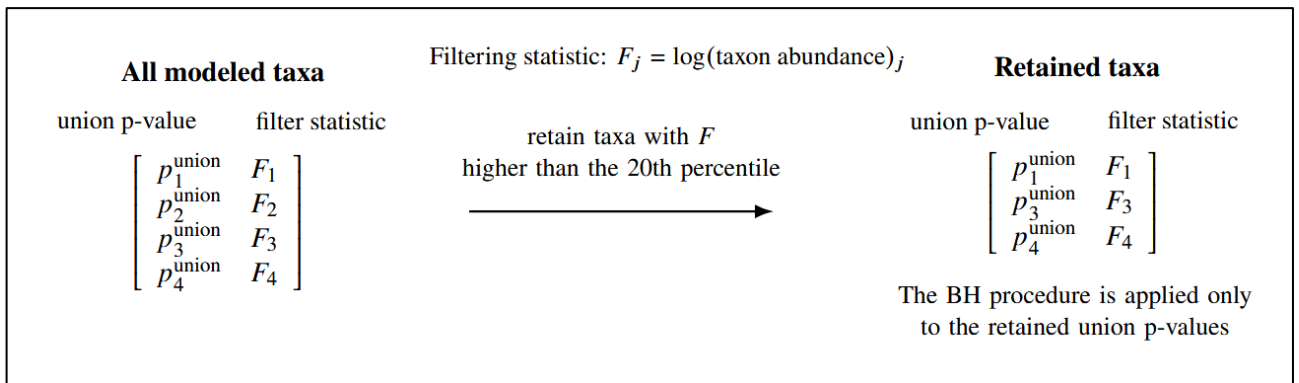


Figure 12. Independent filtering uses the log-transformed total count of each modeled non-reference taxon to exclude low abundance taxa from final BH adjustment

This is done using the log-transformed total count of each modeled non-reference taxon with +1 added to avoid zero count taxa:

$$F_j = \log \left(1 + \sum_i X_{ij} \right)$$

As in pre-filtering (see section 3.1.1), the filtering statistic must be independent of the tested term (Bourgon et al., 2010). By default, PURSUE uses the 20th percentile of this statistic as the filtering threshold, chosen empirically (Fig. A2.3). Taxa with this statistic at or above this threshold are retained for final multiple testing correction, whereas taxa below the threshold are excluded from the final BH correction set. The final q-values are then computed by applying the BH procedure to the union p-values of the retained taxa only. Rejecting all taxa with $q \leq \alpha$ then controls FDR, meaning that, under the assumptions of the BH procedure, no more than an α of rejected taxa are expected to be false positives (Benjamini & Hochberg, 1995).

3.2. DAA tool performance evaluation

To evaluate and compare the performance of the reviewed DAA methods together with PURSUE, we employed the simulation framework of Yang & Chen (2022), SimulateMSeq. This framework generates microbiome count data while allowing dataset characteristics to be tuned: Direction of abundance shifts indicates whether truly differential taxa change in a balanced manner, with mixed directions across taxa, or in an unbalanced manner, with the same direction of change across taxa. Sample size is the total number of simulated samples, and taxa number of taxa is the number of taxa included in the simulated count table. Signal density is the proportion of taxa that are truly differential. Differential mode specifies whether differential taxa are selected from the relatively abundant, rare, or mixed part of the abundance distribution. Sequencing depth controls the mean and variability of total read counts and can also be made dependent on the covariate of interest to induce depth confounding. Additional binary or continuous variables may also be included. Low and high diversity community compositions were also represented by using the Human Microbiome Project reference vaginal or stool datasets, respectively (Huttenhower et al., 2012; Schiffer et al., 2019), as the baseline datasets for parameter estimation and data generation. Because evaluation of the full combinatorial parameter grid used by Yang & Chen (2022) was computationally prohibitive, a hybrid benchmark design was used. We used a small data characteristics parameter grid for null settings (where there are no shifting taxa), in order to see whether combinations of unfavorable parameters can decrease the calibration of PURSUE more than each one alone. For the non-null settings (where a subset of taxa is truly shifted), we used a one factor at a time (OFAT) design, where we tested individual parameters. This was because the number of parameters and, consequently, unique combinations of parameters was large. The covariate of interest was simulated as binary with equal group sizes. For the global null, all unique combinations of diversity, sample size, and number of taxa were used (Table 2).

Table 2. Simulation settings used to evaluate DAA tool performance for different dataset characteristics. In null settings, each parameter separated by commas means that each value was used in constructing a unique combination of settings ($2 \times 3 \times 2$, to be exact). In non-null settings, cells in bold show which parameter was perturbed in the given setting. Cells with multiple parameters separated by commas represent multiple separate unique settings.

Parameter Setting	Shift direction	Taxa diversity	Sample size, n	Number of taxa, m	Depth confounding	Signal density, %	Differential OTU	Confounder
Global null	None	Stool, vaginal	50, 100, 200	50, 500	None	None	None	None
Baseline non-null	Balanced	Stool	100	500	None	10	Abundant	None
Unbalanced shift direction test	Unbalanced	Stool	100	500	None	10	Abundant	None
Low taxa diversity test	Balanced	Vaginal	100	500	None	10	Abundant	None
Sample size test	Balanced	Stool	50, 200	500	None	10	Abundant	None
Small taxa number test	Balanced	Stool	100	50	None	10	Abundant	None
Depth confounding test	Balanced	Stool	100	500	4-fold, 9-fold	10	Abundant	None
Signal density test	Balanced	Stool	100	500	None	5, 20	Abundant	None
Differential OTU rarity test	Balanced	Stool	100	500	None	10	Rare, mix	None
Confounder adjustment test	Balanced	Stool	100	500	None	10	Abundant	Continuous

For the non-null settings, a baseline simulation setting was first defined, where the direction of abundance shifts was balanced, community diversity was high (stool), sample size $n = 100$, number of taxa $m = 500$, no depth confounding, signal density of 10 %, differential OTUs being

selected from the most abundant, and no confounder (Table 2). All subsequent settings were tuned one at a time. For all 9 tested DAA tools, all 25 simulation settings were generated 100 times. To ensure a reproducible comparison, all methods were applied to the same simulated datasets, which were externally filtered to retain taxa with a minimum total count of 10 and a prevalence of at least 0.05 across samples. Internal abundance and prevalence filtering within each method was disabled. Despite the literature review covering MaAsLin2, the newer version, MaAsLin3, was used in the benchmark due to its novelty and relative lack of benchmarking. MaAsLin3 was run in its default configuration with TSS normalization, log transformation, augmented prevalence modeling, median-comparison abundance correction, and depth adjustment included in the model formula as per standard recommendation. ANCOM-BC2 was run in a fixed effects two group sensitivity score-filtered configuration with BH adjustment. LinDA was run in its default fixed effects count data configuration with winsorization and adaptive zero handling. LDM was run in its standard OTU level omnibus configuration. LOCOM was run using the core permutation-based model. ALDEx2 was run as Wilcoxon in unadjusted settings and GLM in the confounder adjusted setting. Corncob was run as a differential abundance only Wald test without bootstrapping. ZicoSeq was run in its standard configuration with winsorization, posterior sampling and default multistage reference-based normalization. PURSUE was run in its default configuration with bootstrapped reference selection and group cleanup, depth appended to the model formula, winsorized abundance residuals, and independent filtering of final p-values. For performance evaluation, taxa that were not assigned a final q-value in the PURSUE result table, including reference taxa and independently filtered taxa, were treated as non-significant by setting their benchmark q-values to 1. Taxa excluded by the common external prefilter were not used in truth comparison as non-significant calls but were removed entirely for every method.

For each simulated dataset, taxa declared significant after multiple-testing correction were identified using a q-value threshold of 0.05. These significance calls were then compared against the known simulation truth, that is, whether each taxon was truly shifted or non-shifted in the simulation model. Several metrics were then employed to assess performance:

$$FPR = \frac{FP}{TN + FP}$$

$$FDR = \frac{FP}{TP + FP}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

where TP is the number of true positives, FP is the number of false positives, FN is the number of false negatives, and TN is the number of true negatives. In the null scenarios, where no taxa were truly differential, calibration is assessed using the false positive rate (FPR). The nominal FPR threshold for a DAA method to be considered calibrated is 0.05. In the non-null scenarios, detection performance was assessed using FDR, precision, recall, and the F1 score. The nominal FDR threshold was 0.05. Methods were considered to control FDR if their empirical FDR did not exceed this threshold. All metrics are reported as mean $\pm$ SD. When discoveries exist, FDR is $1 - \text{Precision}$. However, when both TP and FP are 0, FDR and Precision are set to 0. This is mainly visible in the case of ALDEx2 in the low diversity vaginal setting, where mean FDR is 0 across replicates, meaning it returned no FP , but precision was 0.3400 ± 0.4761 , along with 1.2300 ± 2.3820 TP (Fig. 16; Fig. 20). This means there were replicates that contributed zeros when precision was averaged, exhibiting a $FDR \neq 1 - \text{Precision}$ scenario.

First, mean performance metrics were averaged across simulation settings, with null and non-null settings reported separately. This provides a direct assessment of the main aim of the thesis: whether PURSUE maintains calibration and power across diverse data settings. PURSUE is also compared against the other evaluated DAA methods using the same summaries. Second, performance is also reported in individual simulation settings, where mean performance metrics are averaged across simulation setting repetitions. This is used to identify the strengths and weaknesses of PURSUE, and to compare these against the other methods.

4. RESULTS

4.1. Calibration under null settings

4.1.1. Mean false positive rate across null settings

Across the 12 null simulation settings, PURSUE remained well calibrated (Fig. 13).

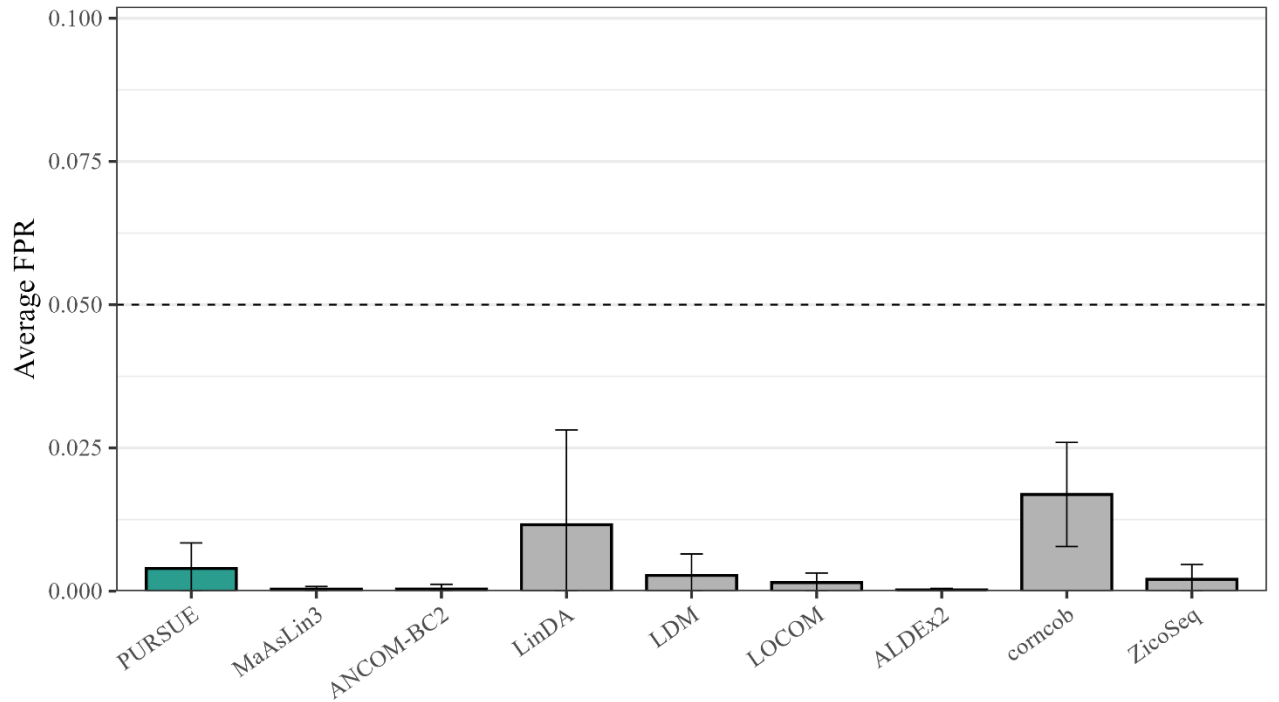


Figure 13. Average FPR across 12 null settings. Error bars show one standard deviation of the per-setting FPR across the 12 null settings.

Its average FPR across settings was 0.0040 ± 0.0045 , remaining below the nominal FPR threshold in every null setting. Compared with PURSUE, ALDEx2 (0.0002 ± 0.0003), MaAsLin3 (0.0003 ± 0.0005), ANCOM-BC2 (0.0004 ± 0.0008), LOCOM (0.0015 ± 0.0017), ZicoSeq (0.0021 ± 0.0026), and LDM (0.0027 ± 0.0037) were more conservative on average. LinDA (0.0116 ± 0.0165) and corncob (0.0169 ± 0.0091) produced more false positives than PURSUE but still remained below nominal FPR in all null settings.

4.1.2. False positive rate across individual null settings

Setting-specific null calibration was then inspected across the 12 null settings (Fig. 14).

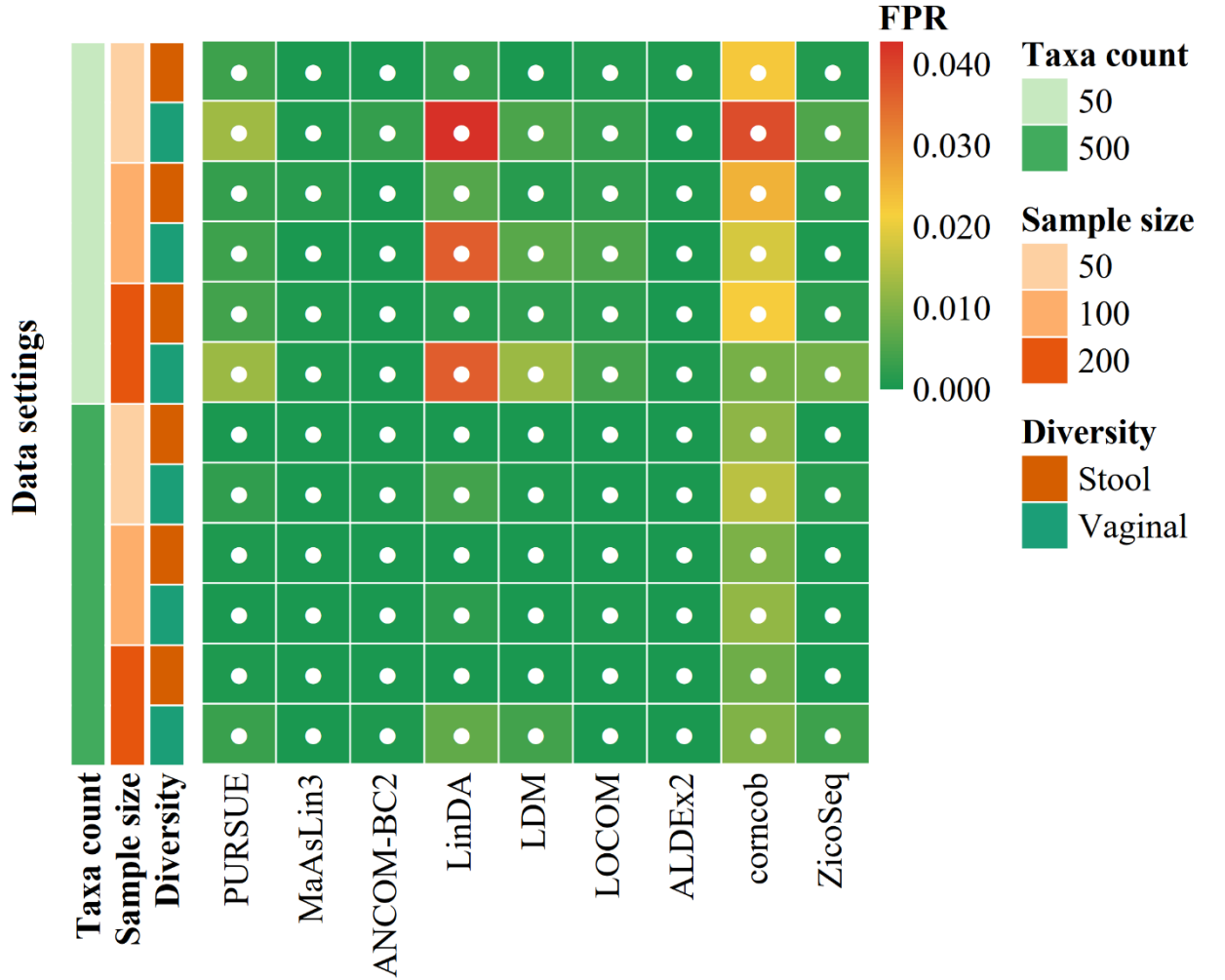


Figure 14. Average FPR across simulation replicates for each data setting and DAA method. White dots mark cells with $\text{FPR} < 0.05$.

For PURSUE, the highest mean FPR was observed in vaginal low diversity with $n = 50$ samples and $m = 50$ number of taxa, where the mean FPR was 0.0132 ± 0.0437 . The second-highest mean FPR was observed in the vaginal low diversity with $n = 200$ and $m = 50$, where the mean FPR was 0.0128 ± 0.0532 . The lowest PURSUE mean FPR was observed in the stool high diversity with $n = 50$ and $m = 500$, where the mean FPR was 0.0002 ± 0.0010 . All methods stayed below the nominal 0.05 FPR threshold in every null setting. ANCOM-BC2 had the lowest or was tied for the lowest FPR in 9 out of 12 settings, returning a mean FPR of exactly zero in those 9 settings, while ALDEx2 had the lowest or was tied for the lowest FPR in 7 out of 12 settings. Among the remaining methods, the closest to nominal FPR values were observed for LinDA and corncob, which produced the largest setting-specific FPRs overall. LinDA reached its highest value of 0.0428 ± 0.1144 in the vaginal, $n = 50$, $m = 50$ setting, and remained relatively close to nominal in the corresponding $m = 50$ and vaginal combination settings with $n = 100$ and $n = 200$, where mean

FPR was 0.0366 ± 0.0929 and 0.0366 ± 0.0955 , respectively. corncob was also near nominal in the vaginal, $n = 50$, $m = 50$ setting, with mean FPR 0.0388 ± 0.0474 .

4.2. Detection performance under non-null settings

4.2.1. Aggregate detection performance across non-null settings

Mean detection performance was first assessed across the 13 non-null simulation settings using precision and recall (Fig. 15).

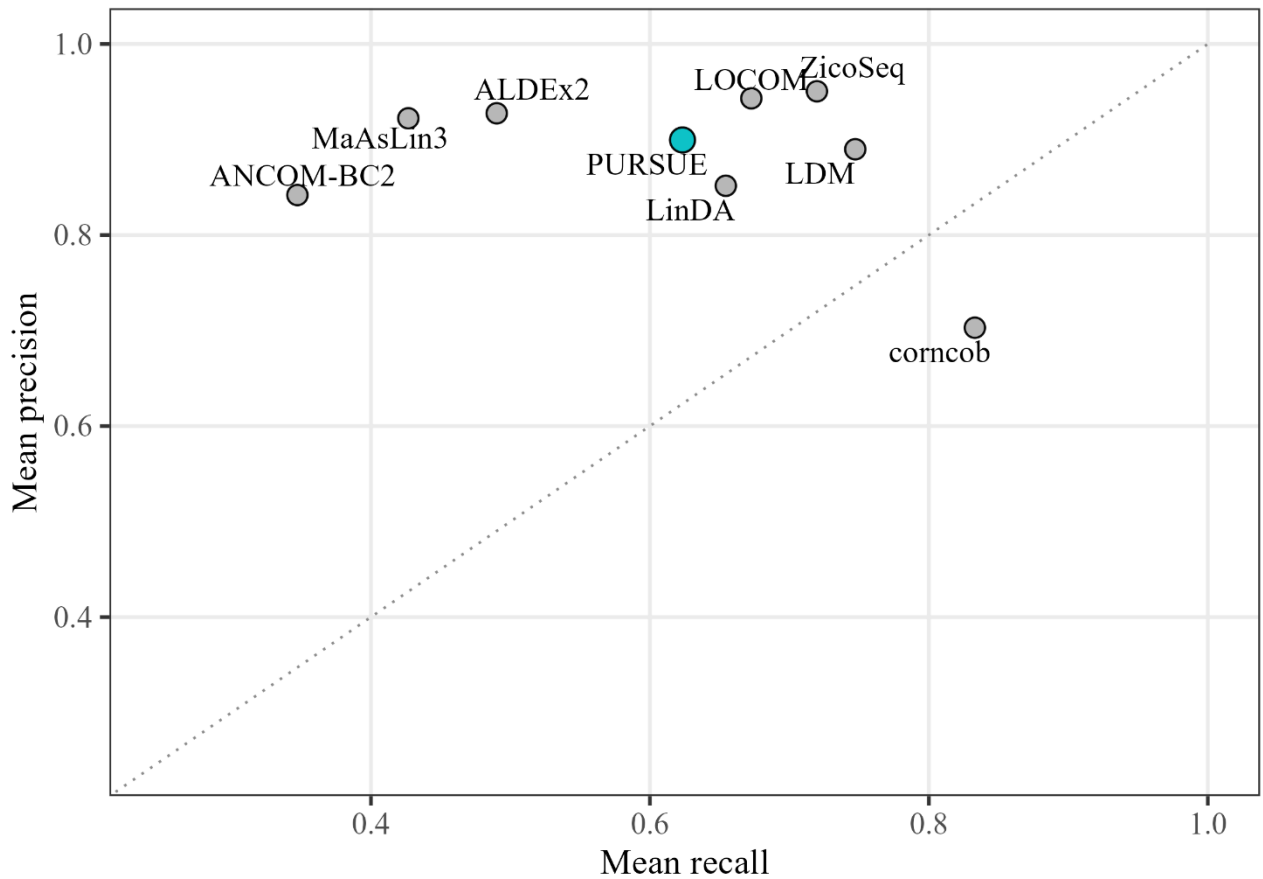


Figure 15. Mean precision and recall across 13 non-null simulation settings. Methods near the upper-right corner have the best combined performance, corresponding to better precision and recall.

Across non-null settings, PURSUE showed high mean precision and intermediate recall. Its mean precision was 0.8995 ± 0.0361 , while its mean recall was 0.6234 ± 0.1776 . PURSUE had greater precision than LDM (0.8898 ± 0.0960), LinDA (0.8516 ± 0.1752), ANCOM-BC2 (0.8419 ± 0.2559), and corncob (0.7030 ± 0.0966), but lower precision than ZicoSeq (0.9505 ± 0.0086), LOCOM (0.9431 ± 0.0211), ALDEx2 (0.9275 ± 0.1846), and MaAsLin3 (0.9222 ± 0.1426).

In terms of recall, PURSUE was lower than corncob (0.8330 ± 0.1087), LDM (0.7472 ± 0.1595), ZicoSeq (0.7198 ± 0.1664), LOCOM (0.6727 ± 0.1715), and LinDA (0.6544 ± 0.2176), but higher than ALDEx2 (0.4902 ± 0.2510), MaAsLin3 (0.4268 ± 0.2720), and ANCOM-BC2 (0.3473 ± 0.2612). On the precision axis, ZicoSeq reached the highest value while maintaining competitive recall, whereas corncob reached the lowest precision. On the recall axis, corncob reached the highest value at the cost of low precision, while ANCOM-BC2 reached the lowest recall but retained moderate precision. corncob was the more liberal outlier, trading precision for recall, whereas ANCOM-BC2, MaAsLin3, and ALDEx2 traded recall for precision.

Mean detection performance was also assessed across the 13 non-null simulation settings using the F1 score and FDR (Fig. 16).

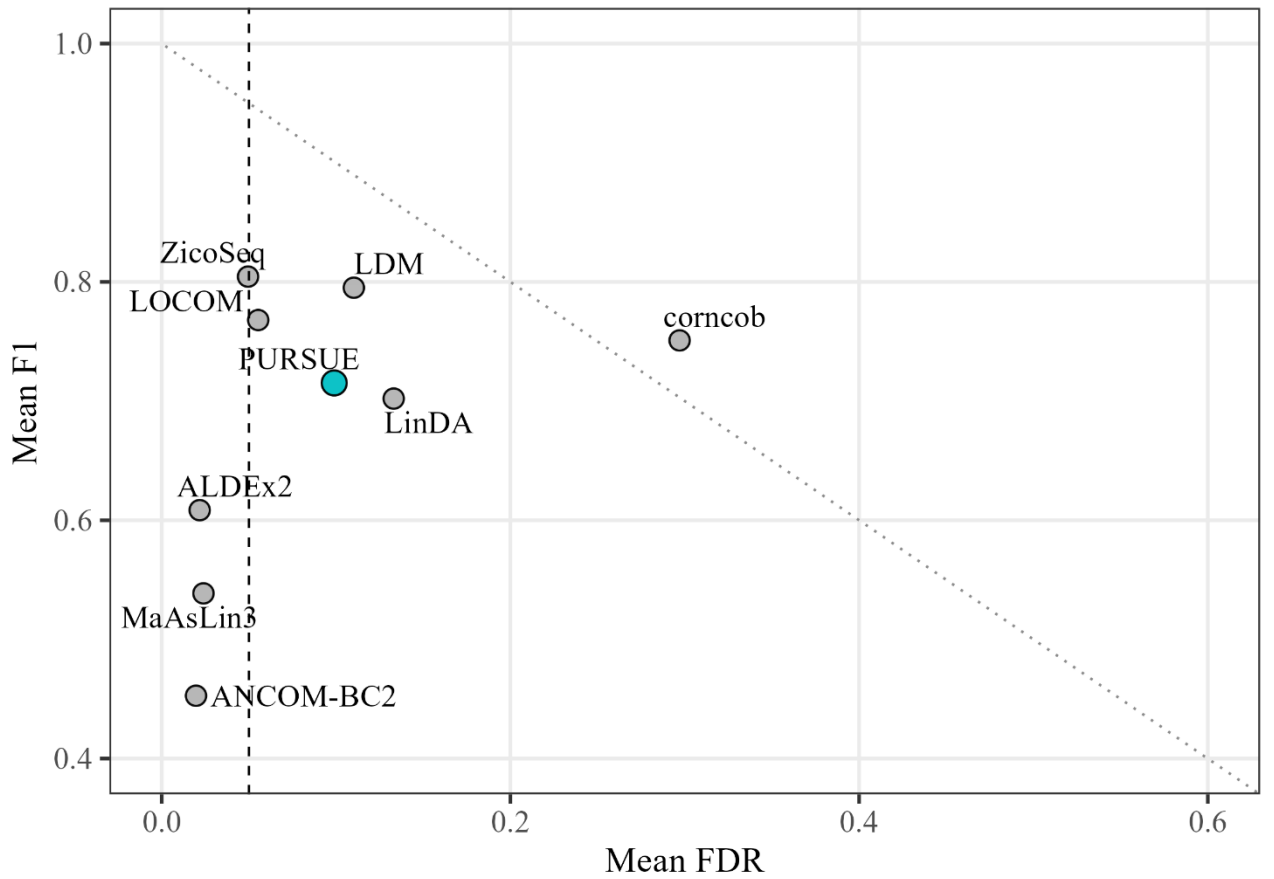


Figure 16. Mean F1 score and empirical FDR across 13 non-null simulation settings. The dashed vertical line indicates the nominal FDR threshold of 0.05. Methods near the upper-left corner have the best combined performance, corresponding to higher F1 and lower FDR.

Across non-null settings, PURSUE showed both moderate F1 and FDR, placing it between the more conservative and the more liberal methods. Its mean F1 was 0.7152 ± 0.1435 , while its mean FDR was 0.0989 ± 0.0351 . Compared with PURSUE, ZicoSeq (0.8045 ± 0.1221), LDM ($0.7951 \pm$

0.1162), LOCOM (0.7680 ± 0.1263), and corncob (0.7509 ± 0.0852) achieved higher mean F1, whereas LinDA (0.7021 ± 0.2032), ALDEx2 (0.6085 ± 0.2552), MaAsLin3 (0.5386 ± 0.2846), and ANCOM-BC2 (0.4527 ± 0.2867) achieved lower mean F1.

In terms of FDR, PURSUE was lower than LDM (0.1102 ± 0.0960), LinDA (0.1330 ± 0.1727), and corncob (0.2970 ± 0.0966), but higher than ANCOM-BC2 (0.0197 ± 0.0604), ALDEx2 (0.0218 ± 0.0545), MaAsLin3 (0.0239 ± 0.0078), ZicoSeq (0.0495 ± 0.0086), and LOCOM (0.0553 ± 0.0225). Thus, PURSUE achieved lower FDR than several higher-recall, higher-F1 methods, but did not remain at the nominal 0.05 FDR on average.

On the F1 axis, ZicoSeq reached the highest value while also remaining below the nominal FDR threshold. ANCOM-BC2 showed the lowest FDR overall but also had the lowest F1 owing to its low recall. ANCOM-BC2, ALDEx2, MaAsLin3, and ZicoSeq were the four methods that controlled FDR at the nominal level on average across settings. corncob had the highest FDR but still achieved relatively high F1, trading precision and FDR control for recall.

4.2.2. Detection performance in individual non-null settings

Precision was then inspected individually for the 13 non-null simulation settings (Fig. 17).

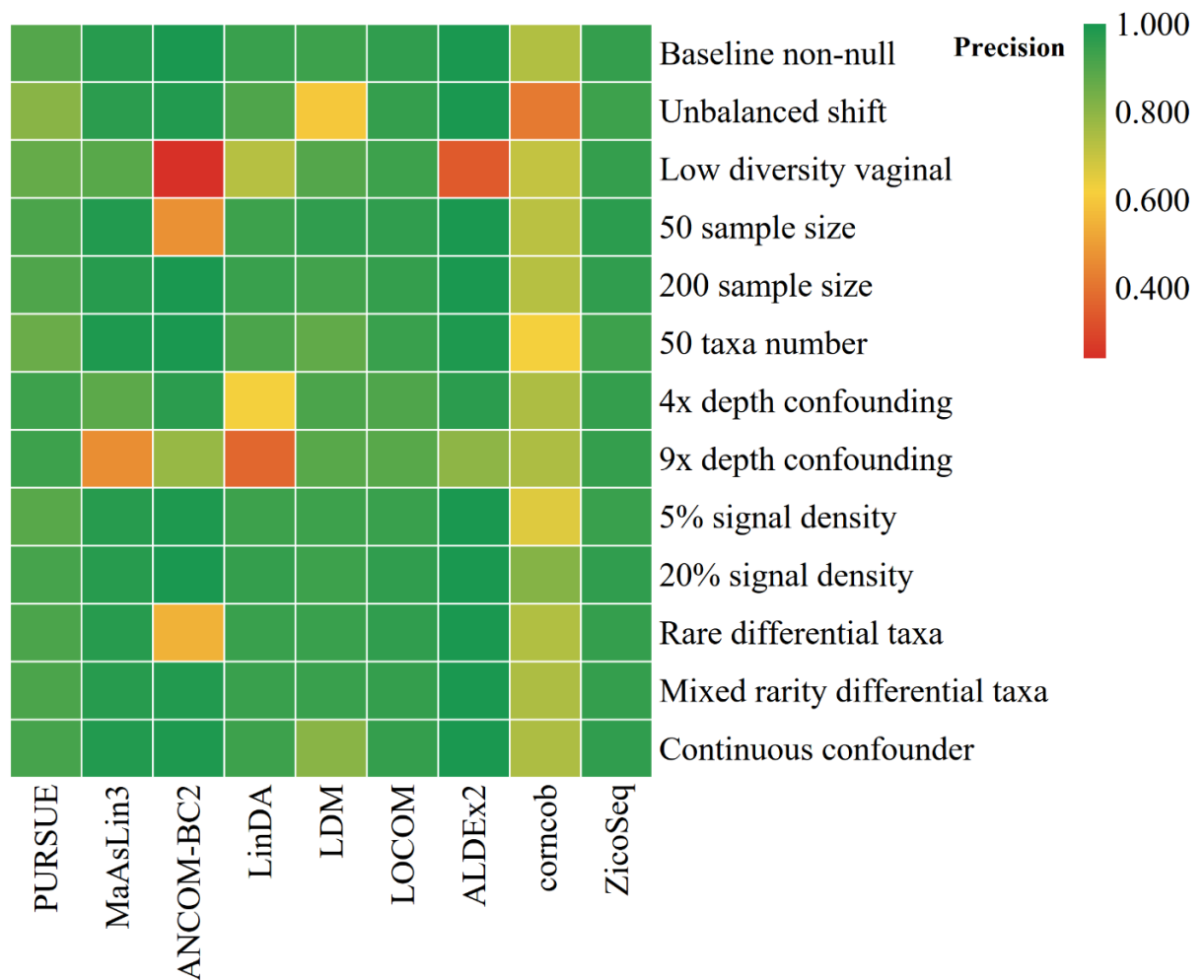


Figure 17. Mean precision across non-null simulation settings for each DAA method.

For PURSUE, mean precision remained relatively high across all non-null settings, ranging from 0.8084 to 0.9379. The lowest PURSUE precision was observed in the unbalanced shift setting, where mean precision was 0.8084 ± 0.0798 . The highest PURSUE precision was observed in the 9-fold depth confounding setting, where mean precision was 0.9379 ± 0.0639 , followed by the 4-fold depth confounding setting, where mean precision was 0.9364 ± 0.0596 .

ZicoSeq and LOCOM showed the most consistently high precision. ZicoSeq ranged from 0.9354 to 0.9641 across settings, while LOCOM ranged from 0.8915 to 0.9600. ALDEx2 also showed high precision in most settings and had the highest precision in 6 out of 13 non-null settings, but it showed a marked drop in the low diversity vaginal setting, where mean precision was 0.3400 ± 0.4761 . ANCOM-BC2 had the highest precision in 5 out of 13 non-null settings, but its precision varied widely across settings, with marked drops in the low diversity vaginal setting (0.2400 ± 0.4292), the 50 sample size setting (0.4700 ± 0.5016), and the rare differential taxa setting (0.5500 ± 0.5000). PURSUE, together with ZicoSeq and LOCOM, were the only methods to remain

approximately consistent among settings and have no obvious outliers. corncob showed the lowest precision overall, being the lowest precision method in 8 out of 13 non-null settings.

Depth confounding reduced precision particularly strongly for MaAsLin3 and LinDA. In the 9-fold depth confounding setting, MaAsLin3 exhibited its lowest mean precision of 0.4605 ± 0.4949 , while LinDA reached 0.3695 ± 0.0686 . In contrast, PURSUE, ZicoSeq, LOCOM, and LDM remained above 0.89 precision in the same setting. The low diversity vaginal setting was also unfavorable for some methods, with ANCOM-BC2 reaching 0.2400 ± 0.4292 and ALDEx2 reaching 0.3400 ± 0.4761 precision, while PURSUE, MaAsLin3, LDM, LOCOM, and ZicoSeq remained above 0.86 precision. In the unbalanced-shift setting, corncob and LDM showed precision reduction, reaching 0.4186 ± 0.0754 and 0.5982 ± 0.1427 , respectively. corncob also showed consistently lower precision than the high precision group.

Recall was also assessed individually for the 13 non-null simulation settings (Fig. 18).

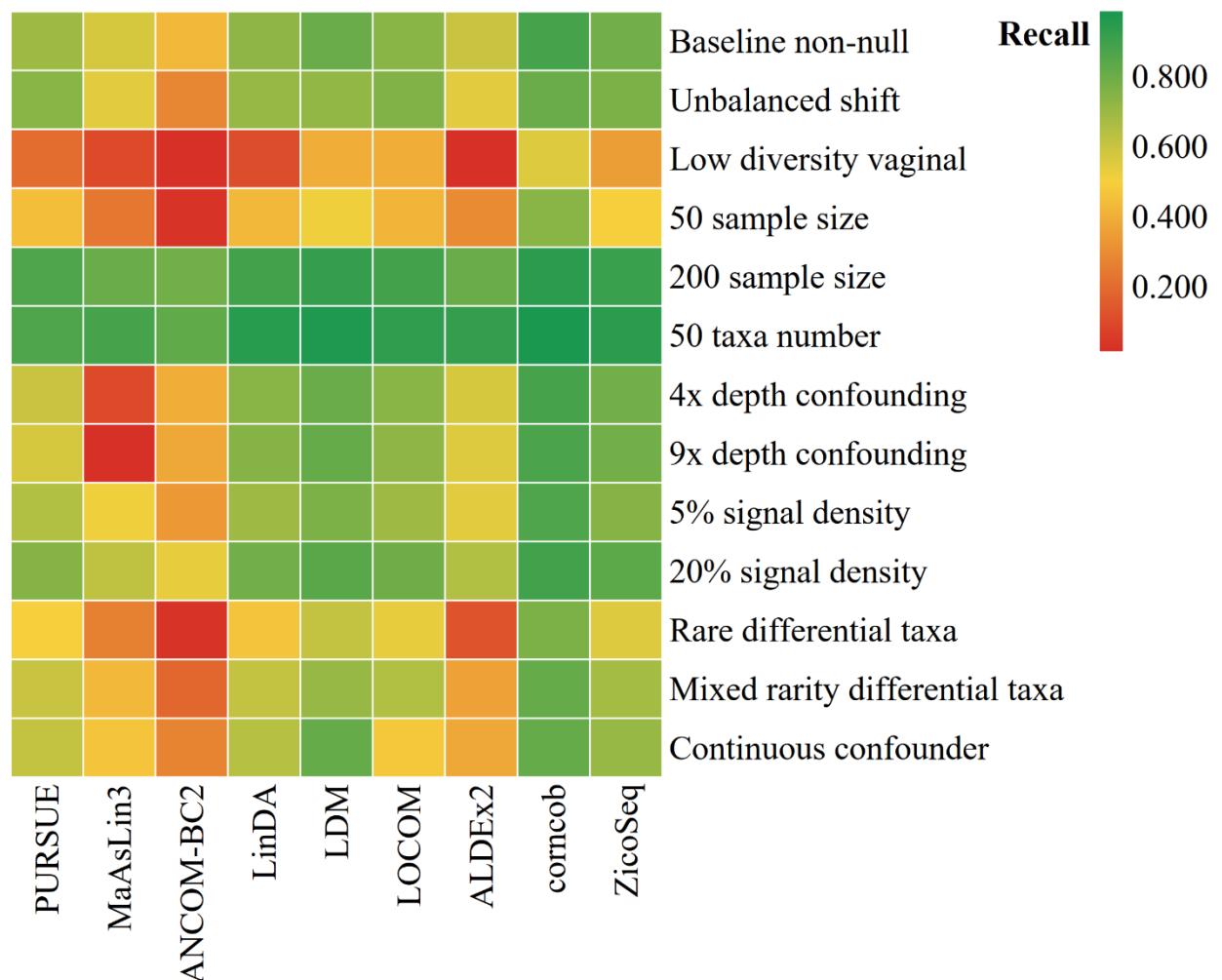


Figure 18. Mean recall across non-null simulation settings for each DAA method.

For PURSUE, mean recall varied more strongly across settings than precision, ranging from 0.2022 to 0.8680. The lowest PURSUE recall was observed in the low diversity vaginal setting, where the mean was 0.2022 ± 0.0944 . The highest PURSUE recall was observed in the 50 number of taxa setting, where mean recall was 0.8680 ± 0.1638 , followed closely by the 200 sample setting, where it was 0.8618 ± 0.0460 . In the baseline non-null setting, PURSUE reached mean recall 0.6892 ± 0.0619 .

Across methods, corncob had the highest recall, achieving the highest recall in all 13 non-null settings, ranging from 0.5525 in the low diversity vaginal setting to 0.9860 in the 50 number of taxa setting. This complemented its more liberal behavior and lower precision. ZicoSeq, LDM, and LOCOM generally had the next highest recall, while PURSUE was intermediate. ANCOM-BC2 showed the lowest recall in 11 out of 13 non-null settings, with its lowest value in the low diversity vaginal setting (0.0152 ± 0.0371). MaAsLin3 had the lowest recall in the remaining two settings, both involving depth confounding.

The low diversity vaginal setting reduced recall across most methods. Besides PURSUE, ALDEx2, and ANCOM-BC2, the strongest reductions were observed for MaAsLin3 (0.0962 ± 0.0598), LinDA (0.1122 ± 0.1044), ZicoSeq (0.3501 ± 0.0866), LOCOM (0.3953 ± 0.0911), and LDM (0.4033 ± 0.0889). In contrast, corncob retained higher recall in this setting. The 50 sample setting also reduced recall for several methods, especially ANCOM-BC2 (0.0300 ± 0.0492), MaAsLin3 (0.2312 ± 0.0847), ALDEx2 (0.2942 ± 0.0780), LOCOM (0.4172 ± 0.0780), and PURSUE. The setting where differential taxa were rare also produced strong recall losses for ANCOM-BC2 (0.0294 ± 0.0391), ALDEx2 (0.1262 ± 0.0562), and MaAsLin3 (0.2626 ± 0.0795). In contrast, the 200 sample and 50 number of taxa settings were generally favorable for recall. In the 200 sample setting, most methods exceeded 0.89 mean recall, with only ALDEx2, ANCOM-BC2, MaAsLin3, and PURSUE below that level. In the 50 number of taxa setting, all methods except ANCOM-BC2 reached at least 0.868 mean recall.

As a summary of detection performance, mean F1 was likewise computed across the 13 non-null settings (Fig. 19).

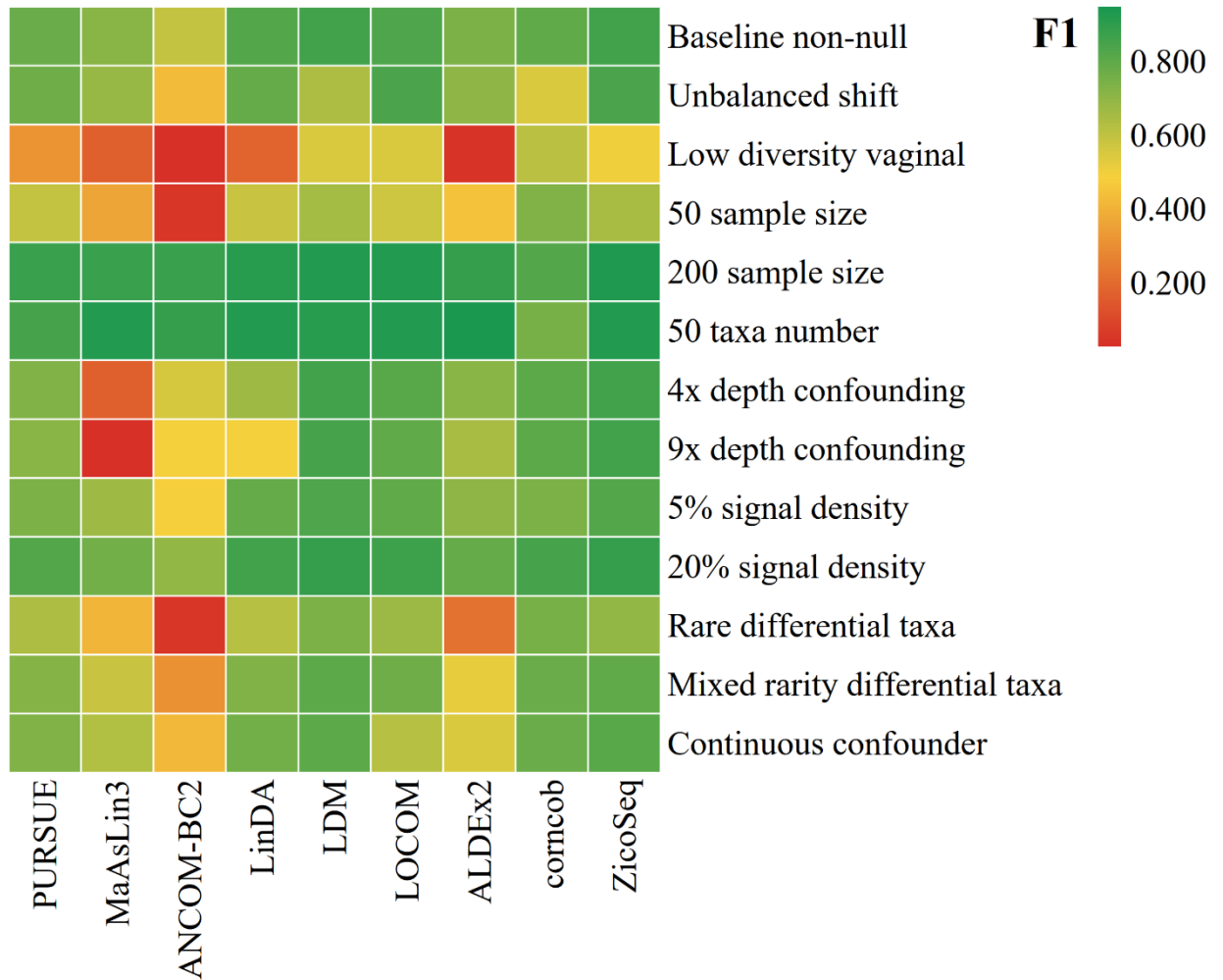


Figure 19. Mean F1 score across non-null simulation settings for each DAA method.

PURSUE's mean F1 ranged from 0.3153 to 0.8823, closely reflecting the variation observed for recall. The lowest PURSUE F1 was observed in the low diversity vaginal setting, where mean F1 was 0.3153 ± 0.1247 . The highest PURSUE F1 was observed in the 200 sample setting, where mean F1 was 0.8823 ± 0.0448 , followed by the 50 number of taxa setting (0.8506 ± 0.1499) and the 20% signal density setting (0.8259 ± 0.0290). In the baseline non-null setting, PURSUE reached mean F1 of 0.7780 ± 0.0507 .

Across methods, ZicoSeq and LDM showed the strongest overall F1 performance across individual settings. ZicoSeq had the highest F1 in 5 out of 13 settings, while LDM had the highest F1 in 3 settings. Corncob also reached the highest F1 in 3 settings, mainly because of its high recall, though at the cost of precision. LOCOM and ALDEx2 each had the highest F1 in one setting. ANCOM-BC2 was strongly affected by multiple settings, with mean F1 dropping to 0.0276 ± 0.0647 in the low diversity vaginal setting, 0.0542 ± 0.0852 in the 50 sample setting, and 0.0545 ± 0.0690 in the rare differential taxa setting. PURSUE did not reach the highest F1 in any individual setting and

was generally in an intermediate position. ALDEx2 was also strongly affected by the rare differential taxa setting, where its mean F1 dropped to 0.2195 ± 0.0882 .

The biggest F1 reductions occurred mainly in the low diversity vaginal setting. In this setting, all methods had mean F1 below 0.62. corncob had the highest F1 in this setting (0.6102 ± 0.0806), followed by LOCOM, LDM, and ZicoSeq, all of which remained around 0.50–0.55. The lowest values were observed for ANCOM-BC2, ALDEx2 (0.0442 ± 0.0833), MaAsLin3 (0.1689 ± 0.0967), and LinDA (0.1769 ± 0.1466). The 50 sample setting also reduced F1 for several methods, although less severely than the low diversity vaginal setting. In this setting, corncob had the highest F1 (0.7275 ± 0.0590), while ANCOM-BC2, MaAsLin3, ALDEx2, and LOCOM were lower.

In the 9-fold depth confounding setting, MaAsLin3 and LinDA dropped to 0.0315 ± 0.0450 and 0.4904 ± 0.0654 , respectively, reflecting the precision and recall losses observed earlier. In contrast, ZicoSeq (0.8605 ± 0.0385), LDM (0.8473 ± 0.0503), corncob (0.8044 ± 0.0530), and LOCOM (0.7976 ± 0.0518) retained higher F1 in this setting. Conversely, the 200 sample and 50 number of taxa settings were generally favorable for F1 across methods, most likely due to increased recall. In the 200 sample setting, all methods except corncob exceeded 0.88 mean F1 or were close to it, while in the 50 number of taxa setting all methods reached at least 0.75 mean F1.

Finally, setting-specific FDR was evaluated across the 13 non-null simulation settings (Fig. 20).

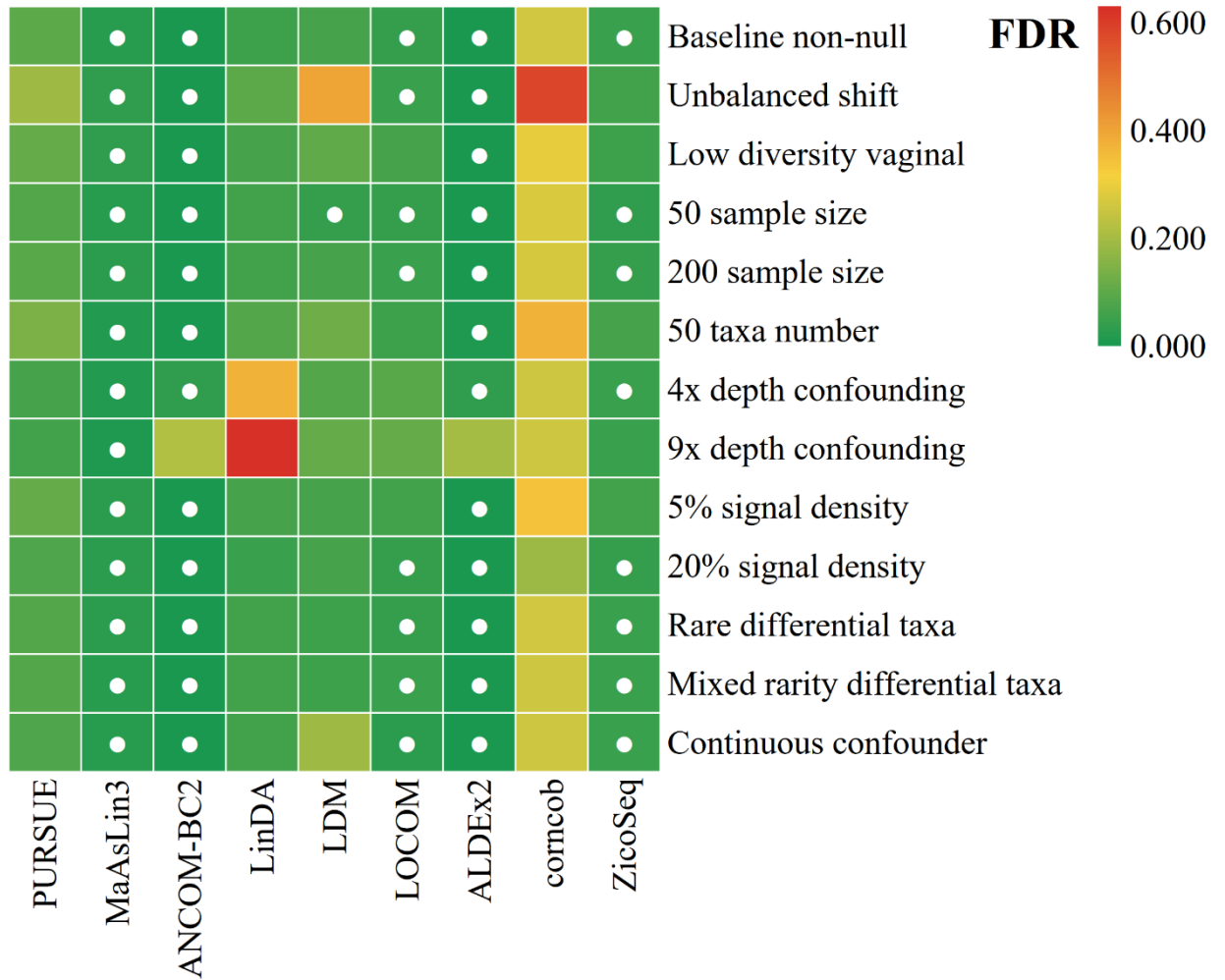


Figure 20. Mean empirical FDR across non-null simulation settings for each DAA method. White dots mark cells with $\text{FDR} < 0.05$.

PURSUE's mean empirical FDR ranged from 0.0621 to 0.1916, exhibiting moderate but consistent inflation across non-null settings. The lowest PURSUE FDR was observed in the 9-fold depth confounding setting, where mean FDR was 0.0621 ± 0.0639 . This was the second lowest FDR in that setting, behind ZicoSeq. In the 4-fold depth confounding setting, PURSUE also had near-nominal FDR (0.0636 ± 0.0596). The highest PURSUE FDR was observed in the unbalanced shift setting, where mean FDR was 0.1916 ± 0.0798 . In the baseline non-null setting, PURSUE reached mean FDR 0.1005 ± 0.0698 . PURSUE did not fall below the nominal 0.05 FDR level in any of the 13 non-null settings.

Generally, FDR control was more method-dependent than setting-dependent, as most white dots aggregated vertically. Across methods, MaAsLin3 showed the strongest overall FDR control, remaining below the nominal 0.05 level in all 13 non-null settings. ALDEx2 and ANCOM-BC2 each remained below nominal FDR in 12 out of 13 settings, with FDR near zero in most settings,

although both showed a marked increase in the 9-fold depth confounding setting, where mean FDR reached 0.2003 ± 0.1148 for ALDEx2 and 0.2181 ± 0.1545 for ANCOM-BC2. ZicoSeq and LOCOM also remained below nominal FDR in 8 out of 13 settings each. LDM fell below nominal FDR only in the 50 sample setting, while LinDA, corncob, and PURSUE did not reach below nominal FDR in any setting. corncob had the highest FDR overall.

The 9-fold depth confounding setting produced severe FDR inflation for LinDA, which reached 0.6305 ± 0.0686 . The unbalanced-shift setting also produced pronounced FDR inflation for several methods, including LDM (0.4018 ± 0.1427) and corncob (0.5814 ± 0.0754). corncob additionally showed consistently elevated FDR across all non-null settings, ranging from 0.2541 ± 0.0789 in the 9-fold depth confounding setting to 0.5814 ± 0.0754 in the unbalanced-shift setting.

Overall, PURSUE showed calibrated behavior under null simulations and high precision under non-null simulations, but its recall and F1 were intermediate relative to the other evaluated methods. In null settings, PURSUE remained below the nominal 0.05 FPR threshold in every setting. In non-null settings, PURSUE maintained high precision across individual settings, but recovered fewer true positives than the highest recall methods, particularly corncob. Its F1 score was therefore comparatively moderate. FDR was consistently lower than that of LDM, LinDA, and corncob, but PURSUE did not remain below the nominal 0.05 FDR level in any non-null setting. Across methods, MaAsLin3, ALDEx2, and ANCOM-BC2 showed the strongest FDR control but lower recall, ZicoSeq and LOCOM showed the most favorable balance of high precision and near-nominal FDR, while corncob was the most liberal.

5. DISCUSSION

In this thesis, we developed PURSUE, a reference-normalized two-part model with permutation-based inference implemented in R and evaluated it against eight other DAA methods using simulated 16S microbiome data. The benchmark included null settings for assessing calibration and non-null settings for assessing detection performance.

PURSUE was well calibrated under all null settings, achieved high precision and intermediate recall in non-null settings, and showed elevated FDR relative to the most conservative methods like ALDEx2 and ANCOM-BC2. Its strongest performance was in depth confounded settings, while low diversity and small sample size settings were its main stressors.

Turning to the other evaluated methods, ALDEx2 was conservative, with high precision, low recall, and strong FDR control (Fig. 15; Fig. 16). Its weakest settings were the low diversity vaginal setting (precision) and the 9-fold depth confounding setting, the only setting in which ALDEx2 exceeded the nominal FDR threshold (Fig. 17; Fig. 20). The conservative profile is concordant with previous benchmarks that describe ALDEx2 as reliably FDR controlling but low recall (Calgaro et al., 2020; Cappellato et al., 2022; Wirbel et al., 2024).

ZicoSeq was a top performer overall, with the highest precision, below nominal aggregate FDR, and the highest F1 (Fig. 15; Fig. 16). Its strongest setting was depth confounding, where it ranked first in precision and achieved the highest F1 among all methods while maintaining near-nominal FDR (Fig. 17; Fig. 19; Fig. 20). This is consistent with the general characterization of ZicoSeq as broadly robust and, specifically, Yang and Chen (2022) report that ZicoSeq is uniquely capable of controlling FDR while maintaining power under depth confounding.

LOCOM had the second highest precision and a near-nominal aggregate FDR (Fig. 15; Fig. 16). Recall reductions were observed in low diversity and small sample settings (Fig. 18). The same was seen in prior benchmarks of LOCOM that showed strong FDR control among compositional methods (Hu et al., 2022; Zhou et al., 2022) and with Lin and Peddada's (2023) report of reduced power at small sample sizes.

LDM had high recall and the second highest F1 but met the nominal FDR threshold in only one setting (Fig. 15; Fig. 16; Fig. 20). Its weakest setting was unbalanced shifts, where FDR rose substantially (Fig. 20). Previous studies also report that LDM's calibration deteriorates as compositional effects become strong or unbalanced (Yang & Chen, 2022; Yang & Chen, 2023).

MaAsLin3 had high aggregate precision but lower recall and F1, with calibration and FDR control met in every setting (Fig. 14; Fig. 15; Fig. 16; Fig. 20). Its most pronounced weakness was depth

confounding, where both precision and recall dropped sharply (Fig. 17; Fig. 18). FDR nevertheless remained well controlled across all settings, including depth confounding (Fig. 20). MaAsLin3 has not previously been benchmarked for depth confounding behavior in independent literature, and this finding of its weakness in that setting is novel.

LinDA had intermediate recall but did not meet the nominal FDR threshold in any non-null setting (Fig. 15; Fig. 20). Its weakest setting was 9-fold depth confounding, where precision dropped sharply and FDR became severely inflated (Fig. 17; Fig. 20). The general calibration instability is concordant with prior reports outside specific data settings (Hu et al., 2023; Yang & Chen, 2023), but Zhou and colleagues (2022) reported that their adaptive depth confounding adjustment substantially reduced false positives under their depth confounded simulations. However, in the present benchmark, this was not sufficient to maintain FDR control (Fig. 20).

corncob was the most liberal method, with the highest recall but FDR exceeding the nominal threshold in every non-null setting (Fig. 15; Fig. 16; Fig. 20). Its weakest setting was unbalanced shifts (Fig. 20). This matches the general characterization of corncob as liberal in 16S settings (Calgaro et al., 2020; Cappellato et al., 2022), with the unbalanced shift inflation specifically aligning with prior reports of FDR breakdown under dense or unbalanced compositional shifts (Lin & Peddada, 2023; Yang & Chen, 2022).

ANCOM-BC2 was well calibrated under the null and showed conservative behavior in non-null settings, with the lowest aggregate FDR but also the lowest recall and F1 (Fig. 13; Fig. 16; Fig. 20). Its weakest setting was the low diversity vaginal setting, where precision, recall, and F1 all collapsed (Fig. 17; Fig. 18; Fig. 19). FDR control was strong in all settings except 9-fold depth confounding, where mean FDR rose to 0.2181 ± 0.1545 (Fig. 20). This calibrated, conservative profile aligns with the debut characterization of the SS-filtered variant as adequately calibrated (Lin & Peddada, 2023), although it contrasts with more recent independent reports of inflated type I error and miscalibration (He et al., 2024; Satten et al., 2025).

The high precision and calibration of PURSUE could potentially be attributed to its use of permutation for p-value generation, rather than relying on distributional asymptotic p-values. In their simulations, Hu & Satten (2020) showed that LDM, which uses permutation-based p-values, was well calibrated. Likewise, ZicoSeq exhibited generally good calibration across data settings in its evaluation benchmark (Yang & Chen, 2022). In this study, both methods also had excellent FPR across null settings (Fig. 14). The main drawback of permutation is that it is computationally heavy, as seen by the longest runtime of PURSUE among all benchmarked methods - PURSUE was roughly 18× slower than ZicoSeq on the baseline non-null setting (Table A2). This means that

runtime would potentially be prohibitive when applying PURSUE to thousands of samples and features. So, a DAA method that takes time to compute empirical p-values should ideally return high confidence precise calls, which was only partially true for PURSUE, that showed good calibration but worse FDR control.

Its inflated FDR may be partially explained by two components: first, during reference selection, if unstable taxa are retained in the reference set, otherwise stable tested taxa could be, in relation, interpreted as shifting. The current group-shifting taxa cleanup procedure intends to remedy this, but it uses a simple arithmetic proxy of abundance shifts. On top of that, despite reference cleanup using the tested term to select the reference set (see section 3.1.1), it is empirically stable and calibrated (Fig. 14). Initial pairwise log ratio calculation also uses a pseudocount, which, as discussed in section 2.1.2, is suboptimal (Weiss et al., 2017). A zero-imputation method could replace the arbitrary pseudocount. However, much like with DAA methods themselves, there is a lack of consensus on best zero imputation practices (Abegaz et al., 2024). A more elegant reference cleanup, as employed by ZicoSeq, uses its final p-values from the first run, including reference and non-reference taxa, to remove the most tested term-associated taxa from the reference set, and then re-runs the analysis using the new reference set. This means the removal decision is made by a metric that more comprehensively evaluates taxa abundance shifts. However, the integration of such iterative cleanup would multiply the already long runtime of PURSUE. A potential avenue could be using a fast parametric asymptotic p-value solely for reference selection, while keeping the final p-value empirical. Another option would be to use scaling, such as TSS, instead of references. MaAsLin3 employs TSS and exhibited better FDR control than PURSUE (Fig. 16). Secondly, PURSUE is unique among the benchmarked DAA methods in that it may exclude a substantial portion of taxa from the final testing pool, owing to reference exclusion and independent filtering. This reduces the p-value vector going into BH correction, decreasing the severity of multiplicity correction, meaning that the resulting q-values will be lower and more likely to pass the significance threshold. A potential solution would be to use leave-one-out behavior, where reference taxa are excluded from the denominator only when the numerator includes themselves. That way reference taxa would remain in the testing pool, and BH on more taxa would likely reduce false positives. Stronger FDR control could potentially make PURSUE a more universal high-confidence discovery method than the ones benchmarked, as the FDR controlling, conservative methods, such as ANCOM-BC2, MaAsLin3, and ALDEx2, had notably lower precision in some settings, while PURSUE stayed consistent (Fig. 17).

The low recall of PURSUE could be attributed to its two-part model. Much like in MaAsLin3, the abundance arm is modeled only on nonzero abundance observations, to avoid using a pseudocount,

and to delegate absence modeling separately to the prevalence arm (Chen & Li, 2016; Nickols et al., 2026). This means that the abundance arm may have fewer observations to work with, and in data settings where count information is sparse, may cause reduction in recall (Hu & Satten, 2020; Yang & Chen, 2022). This benchmark also supports that claim, as PURSUE and MaAsLin3 saw notable reductions in recall in low diversity, small sample size, and rare differential taxa settings (Fig. 18).

The strongest performance of PURSUE, observed in depth confounded settings (Fig. 17), was likely due to its use of expected rarefied detection probability as the response in the prevalence arm. Hu and colleagues (2021) already demonstrated that rarefaction-based LDM is robust to depth confounding. Thus, like the permutation step, the rarefied prevalence response may be an important component of PURSUE. Another potential application of this component could be to rescue the performance of MaAsLin3 in depth confounded settings. MaAsLin3 was generally well calibrated (Fig. 13) and had excellent FDR control (Fig. 20), however, its most obvious weaknesses were low precision and recall in depth confounded settings (Fig. 17; Fig. 18). Therefore, replacing the binary prevalence response of MaAsLin3 with expected rarefied prevalence could, theoretically, improve robustness in depth confounded settings substantially, bringing it up to par with the top performing methods such as ZicoSeq and LOCOM.

A DAA method can be seen as the sum of its components or modules (such as normalization strategy, choice of model, and p-value generation scheme). Thus, even though PURSUE generally had intermediate performance, these findings point to the strengths and weaknesses of individual components of the procedure, although an ablation study would be needed to confirm this.

Moving on to limitations, the simulation framework used in this study was chosen because of its extensive parameter profile, which allowed to test DAA methods in various data settings. It is also a semiparametric simulation framework, meaning it mixes real data as a baseline and parametric effect imputation (custom data settings, such as signal density, sample size), in order to more closely represent real microbiome data (Yang & Chen, 2022). However, there are limitations of the design of this benchmark and the simulation framework used. First, as explained in section 3.2, non-null data settings were OFAT, which means that the DAA method performance was not assessed in combinations of unfavorable data settings. For example, low diversity and small sample size were detrimental for most methods, so a combination of these characteristics in one setting would likely show even further breakdown of performance. It would have also been useful to show interactions including sample size or number of taxa, as these are data characteristics that can be directly observed and assessed. Researchers could then use this information to predict DAA method behavior or scale their sampling effort appropriately. The null settings did include data characteristic combinations and indeed showed that they can be more detrimental than each

characteristic alone. In the case of LinDA, low diversity had little impact on FPR in combination with large number of taxa, but a combination of low diversity and small number of taxa brought it up to near-nominal FPR (Fig. 14). In such cases, combinations of data characteristics can also increase confidence in the effect of a characteristic if similar performance changes are observed across most settings containing that characteristic. This is also visible in the LinDA pattern itself, where the small number of taxa and low diversity combination produced consistently elevated FPR across all three sample sizes.

The choice of the simulation generator also adds a layer of complexity when assessing DAA method performance. Similarly to the discussion of which DAA method fits which data best, there is discourse on which simulation framework can best represent realistic microbiome data. Previous work suggests that DAA method performance can differ depending on the simulation generator used (Wirbel et al., 2024). ZicoSeq would be one example of this, as it is the top-performer in SimulateMSeq, but its ranking dropped when applied to data generated by Wirbel and colleagues' (2024) signal-implantation simulator. The same authors also claim that several parametric and semiparametric frameworks failed to represent real microbiome data properties. It is probably unsurprising that realistic microbiome data is difficult to simulate, as it is laden with problematic characteristics, such as those discussed in section 2.1 (He et al., 2024). On top of that, Nearing and colleagues (2022) argue that distribution-based methods can exhibit favorable performance because the simulated data can match the method's statistical assumptions. Therefore, future benchmark work on PURSUE should evaluate the method against multiple simulation frameworks, for example SparseDOSSA2 (Ma et al., 2021), MIDASim (He et al., 2024), metaSPARSim (Patuzzi et al., 2019), and SIMBA, which claims to represent real microbiome data most closely as opposed to other simulators (Wirbel et al., 2024).

The default parameters of reference bootstrap stability, reference cleanup fraction, and independent filtering quantile could have poor generalization as they were tuned on only one data setting (Fig. A2.1; Fig. A2.2; Fig. A2.3). Using multiple simulation frameworks could also apply here.

Finally, in this study, only one DAA method configuration was used. Most DAA methods have multiple configurations that exhibit different performance profiles and are tuned for specific data characteristics. For example, LDM has several variants: standard LDM omnibus (used in this study) (Hu & Satten, 2020), rarefaction-based LDM-A (Hu et al., 2021), LDM-omni3 (Zhu et al., 2022), and LDM-CLR (Hu & Satten, 2023). ANCOM-BC2 has a sensitivity score mode and an unfiltered mode that produce different precision-recall tradeoffs (Lin & Peddada, 2023). The configurations chosen for this benchmark were the defaults, intended to reflect typical user practice rather than

tuned per-dataset performance. Each individual configuration would have added to the size of this benchmark, which was out of scope for this thesis.

6. CONCLUSIONS

In this thesis:

1. PURSUE was developed and implemented as a reference-normalized two-part differential abundance analysis method for 16S ribosomal RNA microbiome count data. Its workflow combines stability-based reference selection, expected-rarefied prevalence modeling, reference-normalized positive-abundance modeling, permutation-based inference, and two-part evidence aggregation.
2. PURSUE showed strong calibration, with false positive rate below the nominal threshold in every null setting.
3. Under non-null simulation settings, PURSUE showed consistently high precision, but its recall was limited in some non-null settings, and false discovery rate was not controlled.
4. Across the full benchmark, no evaluated method simultaneously achieved optimal calibration, precision, recall, and false discovery rate control across all simulation settings. PURSUE was most competitive under sequencing-depth confounding and least competitive under low diversity or small sample size settings.

ACKNOWLEDGMENTS

Thank you to Sigitas Šulčius for supervision and guidance, and to Gediminas Alzbutas for methodological and development support throughout the whole preparation of this thesis. Thank you to the Nature Research Center for allowing us to utilize their computational resources.

Artificial intelligence tools were used during the preparation of this thesis. ChatGPT (OpenAI, 2026) and Claude (Anthropic, 2026) were used for proofreading and language refinement. GitHub Copilot (GitHub, 2026) was used for coding assistance.

REFERENCES

1. Abegaz, F., Abedini, D., White, F., Guerrieri, A., Zancarini, A., Dong, L., Westerhuis, J. A., Van Eeuwijk, F., Bouwmeester, H., & Smilde, A. K. (2024). A strategy for differential abundance analysis of sparse microbiome data with group-wise structured zeros. *Scientific Reports*, 14(1), 12433. <https://doi.org/10.1038/s41598-024-62437-w>
2. Amir, A., McDonald, D., Navas-Molina, J. A., Kopylova, E., Morton, J. T., Xu, Z. Z., Kightley, E. P., Thompson, L. R., Hyde, E. R., Gonzalez, A., & Knight, R. (2017). Deblur rapidly resolves Single-Nucleotide community sequence patterns. *mSystems*, 2(2). <https://doi.org/10.1128/msystems.00191-16>
3. Anthropic. (2026). *Claude* [Large language model]. <https://claude.ai/>
4. Benjamini, Y., & Hochberg, Y. (1995). Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society Series B (Statistical Methodology)*, 57(1), 289–300. <https://doi.org/10.1111/j.2517-6161.1995.tb02031.x>
5. Bourgon, R., Gentleman, R., & Huber, W. (2010). Independent filtering increases detection power for high-throughput experiments. *Proceedings of the National Academy of Sciences*, 107(21), 9546–9551. <https://doi.org/10.1073/pnas.0914005107>
6. Brill, B., Amir, A., & Heller, R. (2022). Testing for differential abundance in compositional counts data, with application to microbiome studies. *The Annals of Applied Statistics*, 16(4). <https://doi.org/10.1214/22-aoas1607>
7. Calgaro, M., Romualdi, C., Waldron, L., Risso, D., & Vitulo, N. (2020). Assessment of statistical methods from single cell, bulk RNA-seq, and metagenomics applied to microbiome data. *Genome Biology*, 21(1), 191. <https://doi.org/10.1186/s13059-020-02104-1>
8. Cao, Q., Sun, X., Rajesh, K., Chalasani, N., Gelow, K., Katz, B., Shah, V. H., Sanyal, A. J., & Smirnova, E. (2021). Effects of rare microbiome taxa filtering on statistical analysis. *Frontiers in Microbiology*, 11, 607325. <https://doi.org/10.3389/fmicb.2020.607325>
9. Cappellato, M., Baruzzo, G., & Di Camillo, B. (2022). Investigating differential abundance methods in microbiome data: A benchmark study. *PLoS Computational Biology*, 18(9), e1010467. <https://doi.org/10.1371/journal.pcbi.1010467>

10. Chen, E. Z., & Li, H. (2016). A two-part mixed-effects model for analyzing longitudinal microbiome compositional data. *Bioinformatics*, 32(17), 2611–2617.
<https://doi.org/10.1093/bioinformatics/btw308>
11. Durazzi, F., Sala, C., Castellani, G., Manfreda, G., Remondini, D., & De Cesare, A. (2021). Comparison between 16S rRNA and shotgun sequencing data for the taxonomic characterization of the gut microbiota. *Scientific Reports*, 11(1), 3030.
<https://doi.org/10.1038/s41598-021-82726-y>
12. Edgar, R. C. (2013). UPARSE: highly accurate OTU sequences from microbial amplicon reads. *Nature Methods*, 10(10), 996–998. <https://doi.org/10.1038/nmeth.2604>
13. Elfil, M., Kamel, S., Kandil, M., Koo, B. B., & Schaefer, S. M. (2020). Implications of the gut microbiome in Parkinson’s disease. *Movement Disorders*, 35(6), 921–933.
<https://doi.org/10.1002/mds.28004>
14. Elie, C., Perret, M., Hage, H., Sentausa, E., Hesketh, A., Louis, K., Fritah-Lafont, A., Leissner, P., Vachon, C., Rostaing, H., Reynier, F., Gervasi, G., & Saliou, A. (2023). Comparison of DNA extraction methods for 16S rRNA gene sequencing in the analysis of the human gut microbiome. *Scientific Reports*, 13(1), 10279.
<https://doi.org/10.1038/s41598-023-33959-6>
15. Fernandes, A. D., Reid, J. N., Macklaim, J. M., McMurrough, T. A., Edgell, D. R., & Gloor, G. B. (2014). Unifying the analysis of high-throughput sequencing datasets: characterizing RNA-seq, 16S rRNA gene sequencing and selective growth experiments by compositional data analysis. *Microbiome*, 2(1), 15. <https://doi.org/10.1186/2049-2618-2-15>
16. Freedman, D., & Lane, D. (1983). A nonstochastic interpretation of reported significance levels. *Journal of Business and Economic Statistics*, 1(4), 292.
<https://doi.org/10.2307/1391660>
17. GitHub. (2026). *GitHub Copilot* [Large language model]. <https://github.com/features/copilot>
18. Hawinkel, S., Rayner, J. C. W., Bijnsens, L., & Thas, O. (2020). Sequence count data are poorly fit by the negative binomial distribution. *PLoS ONE*, 15(4), e0224909.
<https://doi.org/10.1371/journal.pone.0224909>
19. He, M., Zhao, N., & Satten, G. A. (2024). MIDASim: a fast and simple simulator for realistic microbiome data. *Microbiome*, 12(1), 135. <https://doi.org/10.1186/s40168-024-01822-z>
20. Hu, Y., & Satten, G. A. (2020). Testing hypotheses about the microbiome using the linear decomposition model (LDM). *Bioinformatics*, 36(14), 4106–4115.
<https://doi.org/10.1093/bioinformatics/btaa260>

21. Hu, Y., Lane, A., & Satten, G. A. (2021). A rarefaction-based extension of the LDM for testing presence–absence associations in the microbiome. *Bioinformatics*, 37(12), 1652–1657. <https://doi.org/10.1093/bioinformatics/btab012>
22. Hu, Y., Satten, G. A., & Hu, Y. (2022). LOCOM: A logistic regression model for testing differential abundance in compositional microbiome data with false discovery rate control. *Proceedings of the National Academy of Sciences*, 119(30), e2122788119. <https://doi.org/10.1073/pnas.2122788119>
23. Hu, Y., & Satten, G. A. (2023). Compositional analysis of microbiome data using the linear decomposition model (LDM). *Bioinformatics*, 39(11). <https://doi.org/10.1093/bioinformatics/btad668>
24. Hu, Y., Satten, G. A., & Hu, Y.-J. (2023). Impact of Experimental Bias on Compositional Analysis of Microbiome Data. *Genes*, 14(9), 1777. <https://doi.org/10.3390/genes14091777>
25. Hurst, G. D. D. (2017). Extended genomes: symbiosis and evolution. *Interface Focus*, 7(5), 20170001. <https://doi.org/10.1098/rsfs.2017.0001>
26. Huttenhower, C., Sathirapongsasuti, J. F., Segata, N., Huttenhower, C., Lennon, N., Hepburn, T. A., Griggs, A., Ward, D. V., Yandava, C., Friedrich, D. C., Fisher, S., Priest, M. E., Shenoy, N., Kells, C., Tabbaa, D. G., Huttenhower, C., Alm, E. J., Gevers, D., Earl, A. M., . . . Conlan, S. (2012). Structure, function and diversity of the healthy human microbiome. *Nature*, 486(7402), 207–214. <https://doi.org/10.1038/nature11234>
27. Iqbal, S., Begum, F., Nguchu, B. A., Claver, U. P., & Shaw, P. (2025). The invisible architects: microbial communities and their transformative role in soil health and global climate changes. *Environmental Microbiome*, 20(1), 36. <https://doi.org/10.1186/s40793-025-00694-6>
28. Kanagawa, T. (2003). Bias and artifacts in multitemplate polymerase chain reactions (PCR). *Journal of Bioscience and Bioengineering*, 96(4), 317–323. [https://doi.org/10.1016/s1389-1723\(03\)90130-7](https://doi.org/10.1016/s1389-1723(03)90130-7)
29. Klindworth, A., Pruesse, E., Schweer, T., Peplies, J., Quast, C., Horn, M., & Glöckner, F. O. (2012). Evaluation of general 16S ribosomal RNA gene PCR primers for classical and next-generation sequencing-based diversity studies. *Nucleic Acids Research*, 41(1), e1. <https://doi.org/10.1093/nar/gks808>
30. Laursen, M. F., Dalgaard, M. D., & Bahl, M. I. (2017). Genomic GC-Content Affects the Accuracy of 16S rRNA Gene Sequencing Based Microbial Profiling due to PCR Bias. *Frontiers in Microbiology*, 8, 1934. <https://doi.org/10.3389/fmicb.2017.01934>

31. Lin, H., & Peddada, S. D. (2020). Analysis of microbial compositions: a review of normalization and differential abundance analysis. *Npj Biofilms and Microbiomes*, 6(1), 60. <https://doi.org/10.1038/s41522-020-00160-w>
32. Lin, H., & Peddada, S. D. (2023). Multigroup analysis of compositions of microbiomes with covariate adjustments and repeated measures. *Nature Methods*, 21(1), 83–91. <https://doi.org/10.1038/s41592-023-02092-7>
33. Liu, Y., Chen, S., Li, Z., Morrison, A. C., Boerwinkle, E., & Lin, X. (2019). ACAT: A Fast and Powerful p Value Combination Method for Rare-Variant Analysis in Sequencing Studies. *The American Journal of Human Genetics*, 104(3), 410–421. <https://doi.org/10.1016/j.ajhg.2019.01.002>
34. Love, M. I., Huber, W., & Anders, S. (2014). Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology*, 15(12), 550. <https://doi.org/10.1186/s13059-014-0550-8>
35. Ma, S., Ren, B., Mallick, H., Moon, Y. S., Schwager, E., Maharjan, S., Tickle, T. L., Lu, Y., Carmody, R. N., Franzosa, E. A., Janson, L., & Huttenhower, C. (2021). A statistical model for describing and simulating microbial community profiles. *PLoS Computational Biology*, 17(9), e1008913. <https://doi.org/10.1371/journal.pcbi.1008913>
36. Mallick, H., Rahnavard, A., McIver, L. J., Ma, S., Zhang, Y., Nguyen, L. H., Tickle, T. L., Weingart, G., Ren, B., Schwager, E. H., Chatterjee, S., Thompson, K. N., Wilkinson, J. E., Subramanian, A., Lu, Y., Waldron, L., Paulson, J. N., Franzosa, E. A., Bravo, H. C., & Huttenhower, C. (2021). Multivariable association discovery in population-scale meta-omics studies. *PLoS Computational Biology*, 17(11), e1009442. <https://doi.org/10.1371/journal.pcbi.1009442>
37. Martin, B. D., Witten, D., & Willis, A. D. (2020). Modeling microbial abundances and dysbiosis with beta-binomial regression. *The Annals of Applied Statistics*, 14(1). <https://doi.org/10.1214/19-AOAS1283>
38. McMurdie, P. J., & Holmes, S. (2014). Waste not, want not: Why rarefying microbiome data is inadmissible. *PLoS Computational Biology*, 10(4), e1003531. <https://doi.org/10.1371/journal.pcbi.1003531>
39. Morton, J. T., Marotz, C., Washburne, A., Silverman, J., Zaramela, L. S., Edlund, A., Zengler, K., & Knight, R. (2019). Establishing microbial composition measurement standards with reference frames. *Nature Communications*, 10(1). <https://doi.org/10.1038/s41467-019-10656-5>

40. Nearing, J. T., Douglas, G. M., Hayes, M. G., MacDonald, J., Desai, D. K., Allward, N., Jones, C. M. A., Wright, R. J., Dhanani, A. S., Comeau, A. M., & Langille, M. G. I. (2022). Microbiome differential abundance methods produce different results across 38 datasets. *Nature Communications*, 13(1), 342. <https://doi.org/10.1038/s41467-022-28034-z>
41. Nickols, W. A., Kuntz, T., Shen, J., Maharjan, S., Mallick, H., Franzosa, E. A., Thompson, K. N., Nearing, J. T., & Huttenhower, C. (2026). MaAsLin 3: refining and extending generalized multivariable linear models for meta-omic association discovery. *Nature Methods*, 23(3), 554–564. <https://doi.org/10.1038/s41592-025-02923-9>
42. O’Hara, A. M., & Shanahan, F. (2006). The gut flora as a forgotten organ. *EMBO Reports*, 7(7), 688–693. <https://doi.org/10.1038/sj.embor.7400731>
43. OpenAI. (2026). *ChatGPT* [Large language model]. <https://chat.openai.com/>
44. Park, J. C., Chang, L., Kwon, H., & Im, S. (2025). Beyond the gut: decoding the gut–immune–brain axis in health and disease. *Cellular and Molecular Immunology*, 22(11), 1287–1312. <https://doi.org/10.1038/s41423-025-01333-3>
45. Patuzzi, I., Baruzzo, G., Losasso, C., Ricci, A., & Di Camillo, B. (2019). metaSPARSim: a 16S rRNA gene sequencing count data simulator. *BMC Bioinformatics*, 20(S9), 416. <https://doi.org/10.1186/s12859-019-2882-6>
46. Phipson, B., & Smyth, G. K. (2016). Permutation p-values should never be zero: calculating exact p-values when permutations are randomly drawn. *arXiv (Cornell University)*. <https://doi.org/10.48550/arxiv.1603.05766>
47. Satten, G. A., Li, M., & Zhao, N. (2025). CAFT: A Compositional Log-Linear Model for Microbiome Data with Zero Cells. *bioRxiv (Cold Spring Harbor Laboratory)*. <https://doi.org/10.1101/2025.11.26.690468>
48. Schiffer, L., Azhar, R., Shepherd, L., Ramos, M., Geistlinger, L., Huttenhower, C., Dowd, J. B., Segata, N., & Waldron, L. (2019). HMP16SDATA: Efficient Access to the Human Microbiome Project through Bioconductor. *American Journal of Epidemiology*, 188(6), 1023–1026. <https://doi.org/10.1093/aje/kwz006>
49. Schloss, P. D., Gevers, D., & Westcott, S. L. (2011). Reducing the effects of PCR amplification and sequencing artifacts on 16S rRNA-Based studies. *PLoS ONE*, 6(12), e27310. <https://doi.org/10.1371/journal.pone.0027310>
50. Scott, J., & Tuma, M. (2025). Sparse linear least-squares problems. *Acta Numerica*, 34, 891–1010. <https://doi.org/10.1017/s0962492924000059>

51. Teng, F., Nair, S. S. D., Zhu, P., Li, S., Huang, S., Li, X., Xu, J., & Yang, F. (2018). Impact of DNA extraction method and targeted 16S-rRNA hypervariable region on oral microbiota profiling. *Scientific Reports*, 8(1), 16321. <https://doi.org/10.1038/s41598-018-34294-x>
52. Turnbaugh, P. J., Hamady, M., Yatsunencko, T., Cantarel, B. L., Duncan, A., Ley, R. E., Sogin, M. L., Jones, W. J., Roe, B. A., Affourtit, J. P., Egholm, M., Henrissat, B., Heath, A. C., Knight, R., & Gordon, J. I. (2008). A core gut microbiome in obese and lean twins. *Nature*, 457(7228), 480–484. <https://doi.org/10.1038/nature07540>
53. Vujkovic-Cvijin, I., Sklar, J., Jiang, L., Natarajan, L., Knight, R., & Belkaid, Y. (2020). Host variables confound gut microbiota studies of human disease. *Nature*, 587(7834), 448–454. <https://doi.org/10.1038/s41586-020-2881-9>
54. Weiss, S., Xu, Z. Z., Peddada, S., Amir, A., Bittinger, K., Gonzalez, A., Lozupone, C., Zaneveld, J. R., Vázquez-Baeza, Y., Birmingham, A., Hyde, E. R., & Knight, R. (2017). Normalization and microbial differential abundance strategies depend upon data characteristics. *Microbiome*, 5(1), 27. <https://doi.org/10.1186/s40168-017-0237-y>
55. Winkler, A. M., Ridgway, G. R., Webster, M. A., Smith, S. M., & Nichols, T. E. (2014). Permutation inference for the general linear model. *NeuroImage*, 92, 381–397. <https://doi.org/10.1016/j.neuroimage.2014.01.060>
56. Wirbel, J., Essex, M., Forslund, S. K., & Zeller, G. (2024). A realistic benchmark for differential abundance testing and confounder adjustment in human microbiome studies. *Genome Biology*, 25(1), 247. <https://doi.org/10.1186/s13059-024-03390-9>
57. Yang, L., & Chen, J. (2022). A comprehensive evaluation of microbial differential abundance analysis methods: current status and potential solutions. *Microbiome*, 10(1), 130. <https://doi.org/10.1186/s40168-022-01320-0>
58. Yang, L., & Chen, J. (2023). Benchmarking differential abundance analysis methods for correlated microbiome sequencing data. *Briefings in Bioinformatics*, 24(1). <https://doi.org/10.1093/bib/bbac607>
59. Zhu, Z., Satten, G. A., & Hu, Y. (2022). Integrative analysis of relative abundance data and presence–absence data of the microbiome using the LDM. *Bioinformatics*, 38(10), 2915–2917. <https://doi.org/10.1093/bioinformatics/btac181>
60. Zhou, H., He, K., Chen, J., & Zhang, X. (2022). LinDA: linear models for differential abundance analysis of microbiome compositional data. *Genome Biology*, 23(1), 95. <https://doi.org/10.1186/s13059-022-02655-5>

**VILNIUS UNIVERSITY
FACULTY OF MEDICINE**

Systems Biology

Robertas Tomas Tupikas, 2nd year

Master's Thesis

**A Reference-Normalized Two-Part Regression Model for Microbial Composition
Assessment**

SUMMARY

Differential abundance analysis identifies microbial features whose abundance shifts are statistically associated with conditions of interest. However, sequencing count data has strong technical confounding, compositionality, zero inflation, and library size heterogeneity, which complicate inference. Numerous differential abundance analysis tools have been developed, but no single method controls error and maintains power across realistic data settings, motivating new procedures. This thesis presents PURSUE, a reference-normalized two-part regression framework with permutation-based inference. PURSUE was benchmarked against eight DAA methods (ALDEx2, ANCOM-BC2, corncob, LDM, LinDA, LOCOM, MaAsLin3, ZicoSeq) using the SimulateMSeq simulation framework across null and non-null simulation data settings spanning sample size, taxa diversity, number of taxa, signal density, sequencing depth confounding, differential taxa rarity, and confounder adjustment. Under null settings, PURSUE remained well calibrated, with mean false positive rate well below the nominal 0.05 threshold in every setting. Under non-null settings, PURSUE showed high mean precision and intermediate recall, making it precision-favored, with a moderate F1 score and elevated false discovery rate relative to the most conservative methods. PURSUE performed strongest in depth confounded settings, ranking second after ZicoSeq for precision. Low diversity and small sample size settings were its main weaknesses. These findings suggest that PURSUE can serve as a high precision, depth-specialized tool, while highlighting which design components contribute to its strengths and weaknesses, providing a basis for further development.

Keywords:

microbiome; differential abundance analysis; 16S rRNA data; compositional data; DAA method benchmark.

**VILNIAUS UNIVERSITETAS
MEDICINOS FAKULTETAS**

Sistemų biologija

Robertas Tomas Tupikas, 2 kursas

Magistro baigiamasis darbas

**Atskaitos tašku normalizuotas dviejų dalių regresijos modelis mikrobinės sudėties
vertinimui**

SANTRAUKA

Diferencinio gausumo analizė identifikuoja mikrobinius taksonus, kurių gausos pokyčiai statistiškai susiję su tiriamomis sąlygomis. Tačiau sekoskaitos duomenims būdingas techninis iškreipiantysis poveikis, kompoziciškumas, pertekliniai nuliai ir bibliotekos dydžio heterogeniškumas, todėl statistinė inferencija tampa sudėtinga. Nors sukurta daug diferencinio gausumo analizės įrankių, nė vienas iš jų neužtikrina klaidų kontrolės ir pakankamos galios visuose realistiškuose duomenų scenarijuose. Šiame darbe pristatomas PURSUE - atskaitos tašku normalizuotas dviejų dalių modelis, taikantis permutacijomis grįstą inferenciją. PURSUE buvo palygintas su aštuoniais metodais: ALDEx2, ANCOM-BC2, corncob, LDM, LinDA, LOCOM, MaAsLin3 ir ZicoSeq. Lyginimas atliktas naudojant simuliacijos sistemą, apimančią duomenų scenarijus, kurie skyrėsi imties dydžiu, taksonų įvairove, taksonų skaičiumi, signalo tankiu, sekoskaitos gylio iškreipiančiuoju poveikiu, diferencinių taksonų retumu ir papildomais kintamaisiais. Nuliniuose scenarijuose PURSUE išliko gerai kalibruotas: vidutinis klaidingai teigiamų rezultatų dažnis buvo mažesnis už nominalią ribą. Nenuliniuose režimuose PURSUE pasižymėjo dideliu preciziškumu ir vidutiniu jautrumu, todėl buvo labiau orientuotas į patikimus atradimus nei į maksimalų aptikimą. Jo F1 įvertis buvo vidutinis, o klaidingų atradimų dažnis - didesnis nei konservatyviausių metodų. PURSUE pasirodė geriausiai sekoskaitos gylio iškreipimo scenarijuose, kur pagal preciziškumą užėmė antrą vietą po ZicoSeq. Pagrindinės silpnybės buvo mažos taksonų įvairovės ir mažo imties dydžio scenarijai. Rezultatai rodo, kad PURSUE gali būti taikomas kaip aukšto preciziškumo, sekoskaitos gylio iškreipymams pritaikytas įrankis, kartu išryškinant metodo komponentus, kuriuos verta toliau tobulinti.

Raktažodžiai:

mikrobiomas; diferencinio gausumo analizė; 16S rRNA duomenys; kompoziciniai duomenys; diferencinio gausumo analizės metodų palyginimas.

APPENDIX 1. PACKAGE DESIGN

PURSUE is implemented as an R package organized around the main entry point `run_pursue()`. The package source is divided into the top-level runner, reference selection, response construction, permutation machinery, results orchestration, and utilities. The pipeline passes a number of objects between modules: a filtered count matrix, a reference selection result, a responses object containing the prevalence and abundance matrices, a permutation result, and a final union layer. These are then assembled into the results table at the end. Figure A1.1 provides a schematic overview of the workflow.

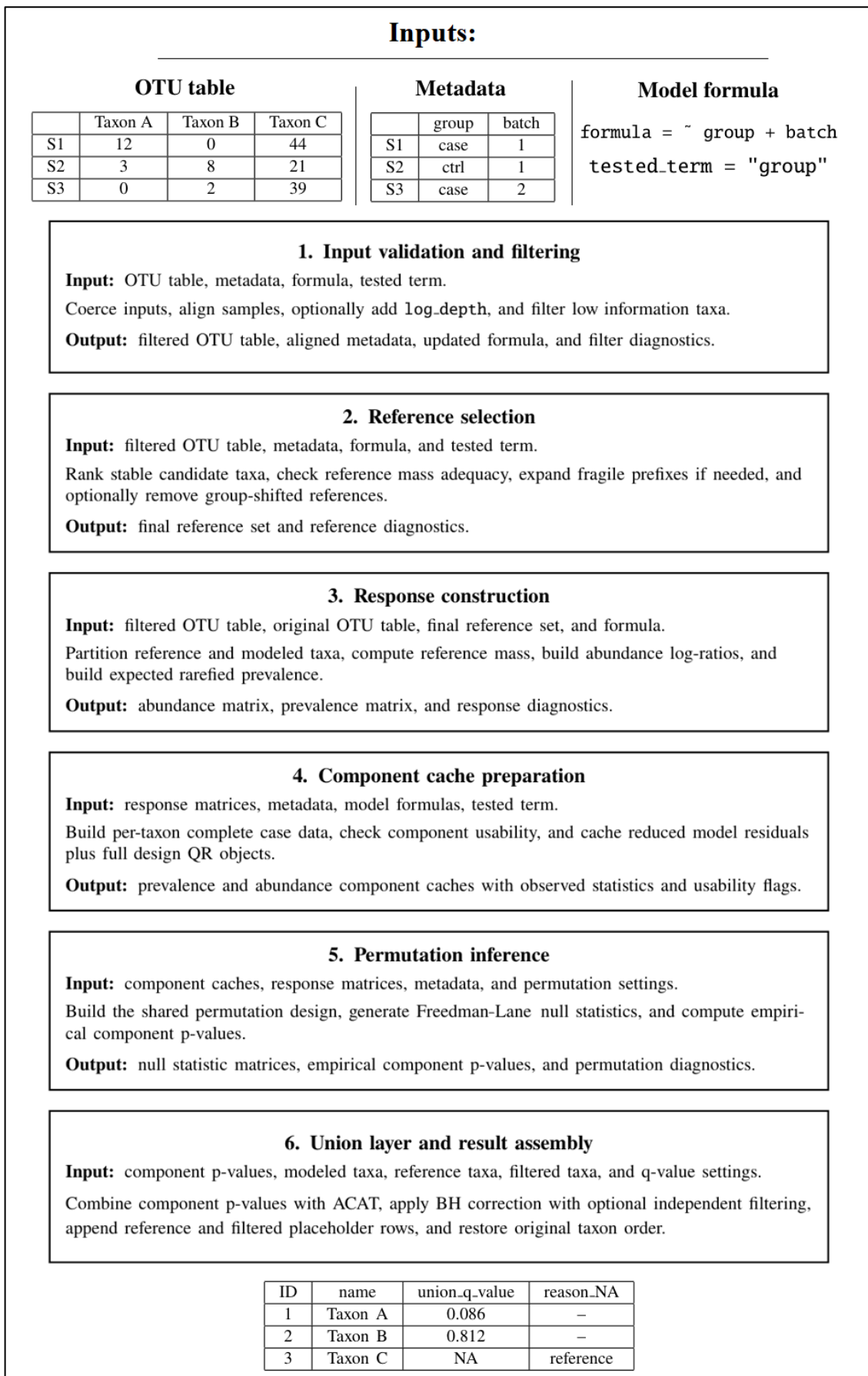


Figure A1.1. Overview of the PURSUE workflow, showing the required inputs, sequential processing steps, and final taxon level results table.

The package relies mainly on base R and `stats`. Additional dependencies are `parallel` for permutation parallelization, `permute` for restricted permutation generation, and `ACAT` for union p-value combination.

Inputs and filtering

The workflow begins inside `run_pursue()`. The function receives all inputs and parameters. Minimum required inputs are sample by taxon OTU count table; sample by variable metadata table; formula, which specifies the variables from metadata to be included in the analysis; and the variable from the formula that will be the tested term.

`as_numeric_matrix()` turns the OTU table into a numeric matrix, and `as_plain_data_frame()` coerces metadata to a base `data.frame`. Then the OTU matrix and metadata are checked to have the same number of samples, matching sample names, and order of samples.

When depth adjustment is enabled, `add_log_depth()` computes each sample's library size from the original count matrix and appends it to the metadata, and `add_term_if_missing()` inserts the resulting `log_depth` term into the model formula if it is not already present. Taxon filtering is then performed by `filter_taxa()` using the supplied minimum prevalence and minimum total count thresholds. Indices of dropped taxa are retained so that they can be reinserted as placeholder rows in the final results table.

Reference selection

Reference selection is orchestrated by `select_reference_bootstrap_prefix()`, which returns the final selected reference indices and names along with diagnostics. Its component functions are described below.

Candidate definition is handled by `select_reference_candidates()`. By default, every filtered taxon enters reference selection; if a maximum candidate count is supplied, only the highest abundance subset of that size is retained.

Stability scoring is performed by `score_reference_candidates()`. The function builds the nuisance design matrix via `make_nuisance_matrix()` by removing the tested term from the formula, then converts candidate counts to the log scale with a pseudocount. Pairwise log-ratios are formed between each unique pair of candidates and residualized against the nuisance matrix using OLS, implemented with matrix operations. The resulting residual standard errors are then put in a symmetric pairwise matrix. The per-taxon stability score is the median of the corresponding row.

Bootstrap re-ranking is performed by `bootstrap_reference_rankings()`. Each replicate resamples the OTU table and metadata jointly with replacement, regenerating row names where samples are drawn more than once and re-applies `score_reference_candidates()` to obtain a bootstrap ranking. `compute_prefix_stability()` then compares each bootstrap ranking against the original data ranking by computing the top-*k* overlap and averaging across replicates. The initial reference cutoff is chosen by `find_stable_prefix()`, which selects the smallest prefix whose bootstrap stability reaches the configured stability threshold.

Reference denominator adequacy is evaluated by `compute_prefix_adequacy()`, which precomputes, for all prefix sizes, the sample-wise reference mass, the number of zero mass samples, and the configured lower-tail absolute and relative mass quantiles. If the initial stable prefix fails the mass conditions, `extend_fragile_reference_prefix()` walks forward along the stability ranking, calling `evaluate_adequacy_conditions()` at each step, and accepts the first prefix that satisfies all three conditions within the tunable expansion limits. If the expansion budget is exhausted after zero mass samples have been resolved but lower-tail absolute or relative mass thresholds remain unsatisfied, the selected prefix is retained as a quality compromise and the failed conditions and affected samples are reported. If the budget is exhausted while zero reference mass remains, the function enters an override mode and continues expanding solely to rescue the zero mass condition. If even exhausting all candidates fails to resolve this, the pipeline stops with an error identifying the affected samples. `diagnose_prefix_mass_problem_samples()` returns the samples that triggered expansion and those that remain problematic in the accepted prefix, and `format_reference_mass_problem_samples()` renders these into a console output.

Once a final prefix is fixed, `summarize_reference_mass()` computes the per-sample reference denominator used downstream.

Reference cleanup is performed by `cleanup_reference_group_shift()`. This is currently limited to binary tested terms. `score_reference_group_shift()` constructs the tested grouping variable, and computes for each selected reference taxon the leave-one-reference-out log-ratio against the remaining reference mass and the absolute difference between the two group means. The highest scoring fraction is removed, subject to a configurable minimum reference size.

Response construction

Response construction is orchestrated by `build_reference_normalized_responses()`. `check_response_inputs()` validates the count matrix and partitions it into reference and modeled non-reference taxa. `summarize_reference_mass()` then computes the per-sample

reference mass and summary statistics for diagnostics and as the denominator for the abundance response.

The abundance response is constructed by `build_abundance_matrix()` on the non-reference taxa. Cells where both the taxon count and the reference mass are positive get the log-ratio response, and all other cells remain NA.

The prevalence response is constructed separately from the wrapper, because it requires depth computed from the full input OTU table rather than after filtering.

`build_prevalence_inputs_by_engine()`, called directly from `run_pursue()`, fixes the rarefaction depth as the minimum sample library size, removes any covariate named `log_depth` from the model formula with `remove_depth_from_prevalence_formula()`, and computes the rarefied response with `compute_expected_rarefied_prevalence()`. The probabilities are evaluated via `lchoose()` for numerical stability and clipped away from exactly 0 and 1 for present taxa; absent taxa stay at 0.

Permutations

The permutation module builds the design and index list which are used by both abundance and prevalence components in the same order, so each null replicate represents the same rearrangement of the samples in both. Both components go through the same downstream cache and null machinery, differing only in which response matrix they receive and component-specific defaults, like winsorization being disabled for the prevalence component.

Design construction is performed by `build_permutation_design()`. It validates the cluster and strata vectors and returns either a sample level permutation design that includes a block ID (cluster, stratum, or both) or a cluster level design carrying a cluster ID with at most one stratum per cluster. The single stratum restriction is because a cluster spanning multiple strata cannot be transported as a unit without splitting its rows apart. Permutation indices are then generated by `generate_permutation_indices()`, which delegates the actual permuting to `permute::shuffleSet`. Sample level draws are stored as integer row permutations. Cluster level draws are stored as named target-to-donor label maps, which the cluster kernel reads when transplanting residual blocks.

`prepare_component_fl_cache()` prepares, per taxon, the objects that every permutation refit will reuse. For each taxon, it assembles a per-taxon data frame: the metadata table with that taxon's response vector attached as a new column. Rows missing in any formula variable are then dropped via complete case filtering. For the abundance arm this drops samples where the taxon was absent.

Different taxa lose different samples this way, so each taxon ends up with its own valid row set, and downstream cache fields are built against this local row set. This means that the reduced model fit and its residuals are computed only on each taxon's valid samples. The residuals are later mirrored into a full sample length view (with NAs at absent rows) so the cluster kernel can index them using the same sample positions its cluster permutations work in.

Whether the taxon can actually be tested is then checked: too few usable samples, a constant response, or a tested term that collapsed to a single level after row deletion. These taxa are flagged as not usable for testing and given a status string that the null kernel reads to skip them.

For taxa that pass, the cache fits the reduced model via QR factorization once and stores everything the per-permutation refit will need: reduced model fitted values via `qr.fitted()` and residuals via `qr.resid()`, the full design QR, and $(Z^T Z)^{-1}$ (Scott & Tuma, 2025). Here, QR factorization is used as an alternative to full linear model fits and caching refers to storing reduced model fitted values and residuals for reuse in every permutation (Fig. A1.2). Once $Z = QR$ is factorized, recomputing coefficients and residuals for a new response vector is reduced to substitution.

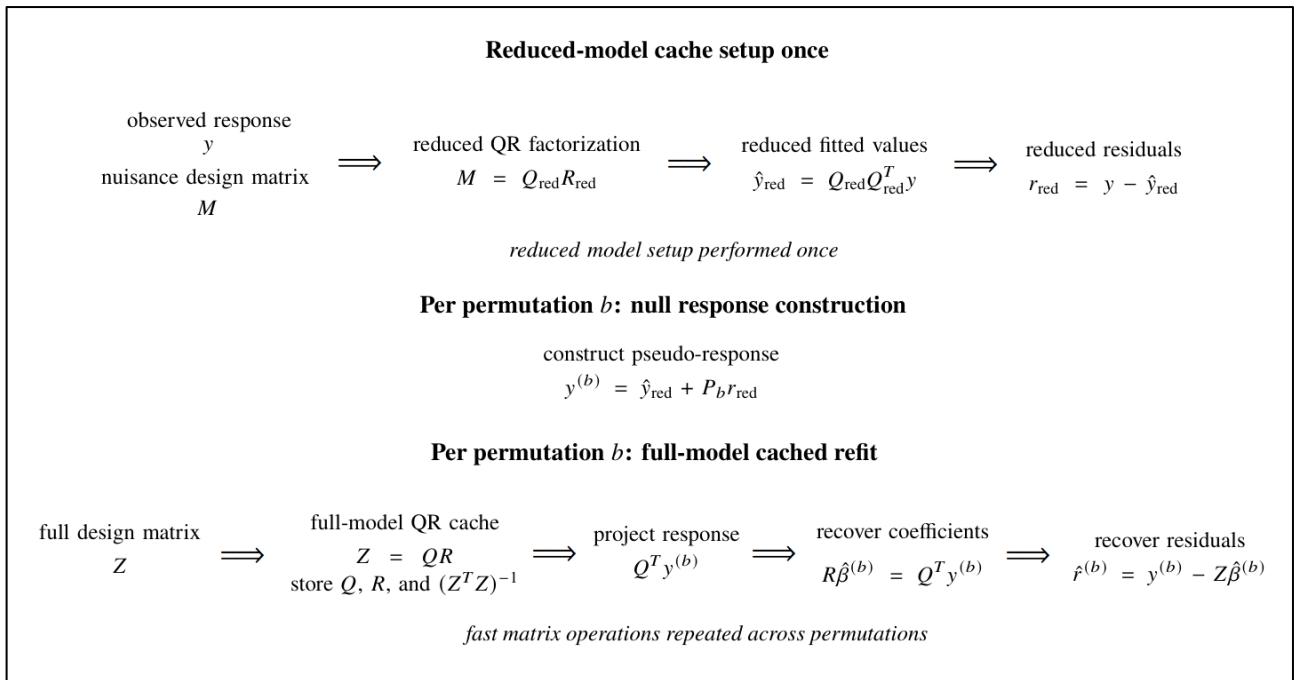


Figure A1.2. QR-based construction of Freedman-Lane null responses and cached full model refits used to obtain permutation level coefficients and residuals.

The observed Wald statistics are then computed from these cached components. When residual winsorization is enabled, `winsorize_residuals()` is applied to the residuals before they are added back to the reduced model fitted values to form the observed response, so both the observed statistic and every null pseudoresponse are constructed from the same residual transformation.

`prepare_component_permutation_cache()` extends each cached entry so it can be indexed in full sample coordinates rather than the taxon's local row set. If a taxon's residual vector is of length 50 when the taxon was fitted on 50 of the 200 samples, it is expanded to length 200 with NAs at the absent rows. `build_taxon_row_context()` and `build_taxon_unit_context()` record which full sample row each local row corresponds to and which local rows share a permutation unit. The cluster kernel can then see which of a taxon's rows belong to a given cluster. `linear_block_wald_from_precomp()` is the only function called inside the permutation loop, and it is also what the cache uses to compute observed statistics. It reuses the cached QR to recover the coefficient vector and residual sum of squares with matrix operations, extracts the tested coefficient block out of the cached $(Z^T Z)^{-1}$, and passes the components to `wald_stat()`, which returns either the scalar or quadratic Wald statistic.

Null statistic computation is done by `compute_component_permutation_null()` based on the permutation design. The sample level kernel `compute_component_fl_null_sample_one_perm()` induces a row order from the global sample permutation via `induce_local_sample_permutation()`, then permutes the residuals, adds them back to the cached fitted values, and computes the null Wald statistic. The cluster level kernel `compute_component_fl_null_cluster_one_perm()` instead transplants whole cluster residual blocks via `apply_cluster_map_to_residual_blocks()`. If a cluster size mismatch is encountered for a particular taxon under a particular permutation, that single cell is neutralized. Each kernel takes the absolute value of its output before storing it, so the comparison against the observed $|W|$ is two-sided. Both kernels are run by a shared orchestrator, `compute_component_fl_null_run()`, which handles preallocation of the statistic and status matrices, parallelization, and the status diagnostics.

When parallelization is enabled, permutations are computed by multiple PSOCK (parallel socket) workers via `parallel::parLapplyLB`. Each worker receives the QR-derived cache once at startup, then takes one permutation index at a time and computes the null Wald statistics for every taxon under that permutation.

Results orchestration

The results orchestration module takes the per-component caches and null matrices produced by the permutation step and turns them into the final union p-values, q-values, and null hypothesis rejection flags that end up in the results table.

`run_component_permutations()` is the main permutation orchestrator. It builds the shared permutation design and draws once, runs the cache preparation for each component, runs the null computation for each component, and assembles the wide observed table from the two caches via `combine_component_cache_tables()`. Each taxon's row carries its observed Wald statistic, estimate, standard error, degrees of freedom, sample count, usability flag, and status string.

Empirical p-values are computed by `compute_empirical_component_pvalues()`. For each taxon and component, only permutations that produced a valid statistic for that taxon contribute to the denominator. Failed or neutralized cells are excluded rather than counted as ties. Taxa whose observed fit was already neutralized by the cache are assigned a p-value of one regardless of how the null comparison turns out.

The per-component p-values are combined into a single union p-value per taxon by `combine_observed_components()`, which behaves differently depending on which components are usable: with both usable it calls `ACAT::ACAT()` on the two p-values, with only one usable it returns that component's p-value, and with neither usable it returns a p-value of one.

Final adjustment and rejection are handled by `compute_union_layer()`. A raw BH adjustment is always computed over all modeled taxa, regardless of whether independent filtering is enabled. When independent filtering is enabled, the function additionally takes the filter statistic as $\log(1 + \text{total count})$ per modeled taxon, thresholds it at the configured percentile, applies BH only to the taxa at or above the threshold, and leaves the filtered out taxa with a placeholder q-value.

APPENDIX 2. RUNTIME AND EMPIRICAL DEFAULTS

Table A2. Runtime of benchmarked methods in seconds, averaged across 50 repetitions of the baseline non-null setting (500 taxa, 100 samples). At 19999 permutations, PURSUE was $\sim 18\times$ slower than ZicoSeq (2nd slowest method), and $\sim 6500\times$ slower than LinDA (fastest method). No method was internally parallelized.

Method	Runtime, seconds $\pm$ SD
PURSUE	2181.573 $\pm$ 166.319
MaAsLin3	14.352 $\pm$ 0.669
ANCOM-BC2	100.943 $\pm$ 22.008
LinDA	0.333 $\pm$ 0.046
LDM	80.775 $\pm$ 3.282
LOCOM	7.737 $\pm$ 0.579
ALDEx2	8.939 $\pm$ 0.741
Corncob	18.532 $\pm$ 0.706
ZicoSeq	119.345 $\pm$ 4.417

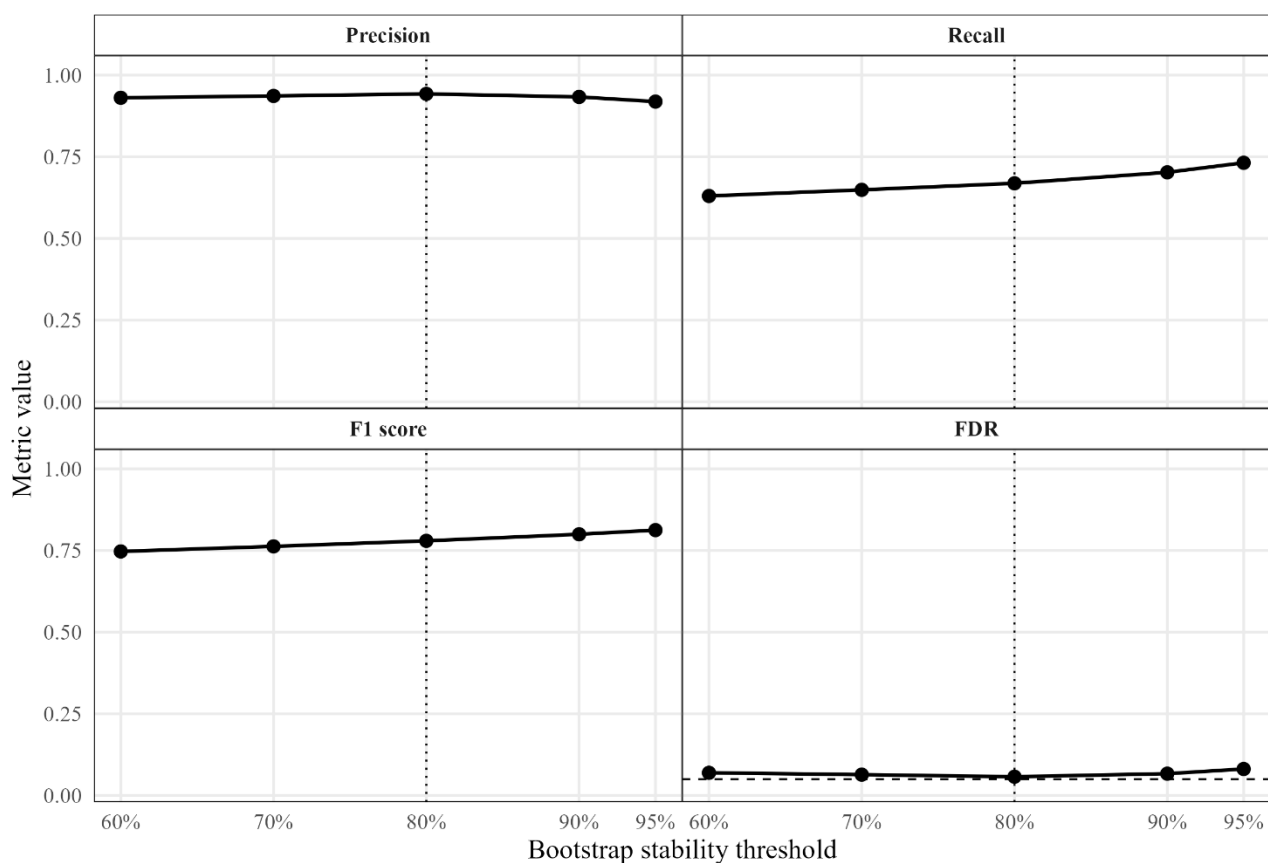


Figure A2.1. PURSUE performance sensitivity to the bootstrap stability threshold. Mean precision, recall, F1 score, and FDR were calculated across repeated simulations of the baseline non-null setting while varying only the bootstrap stability threshold. All other PURSUE parameters were held at their baseline values. The dotted vertical line marks the chosen parameter value, and the dashed horizontal line in the FDR panel marks the nominal 0.05 level. 0.8 was chosen as the last value before precision and FDR started worsening.

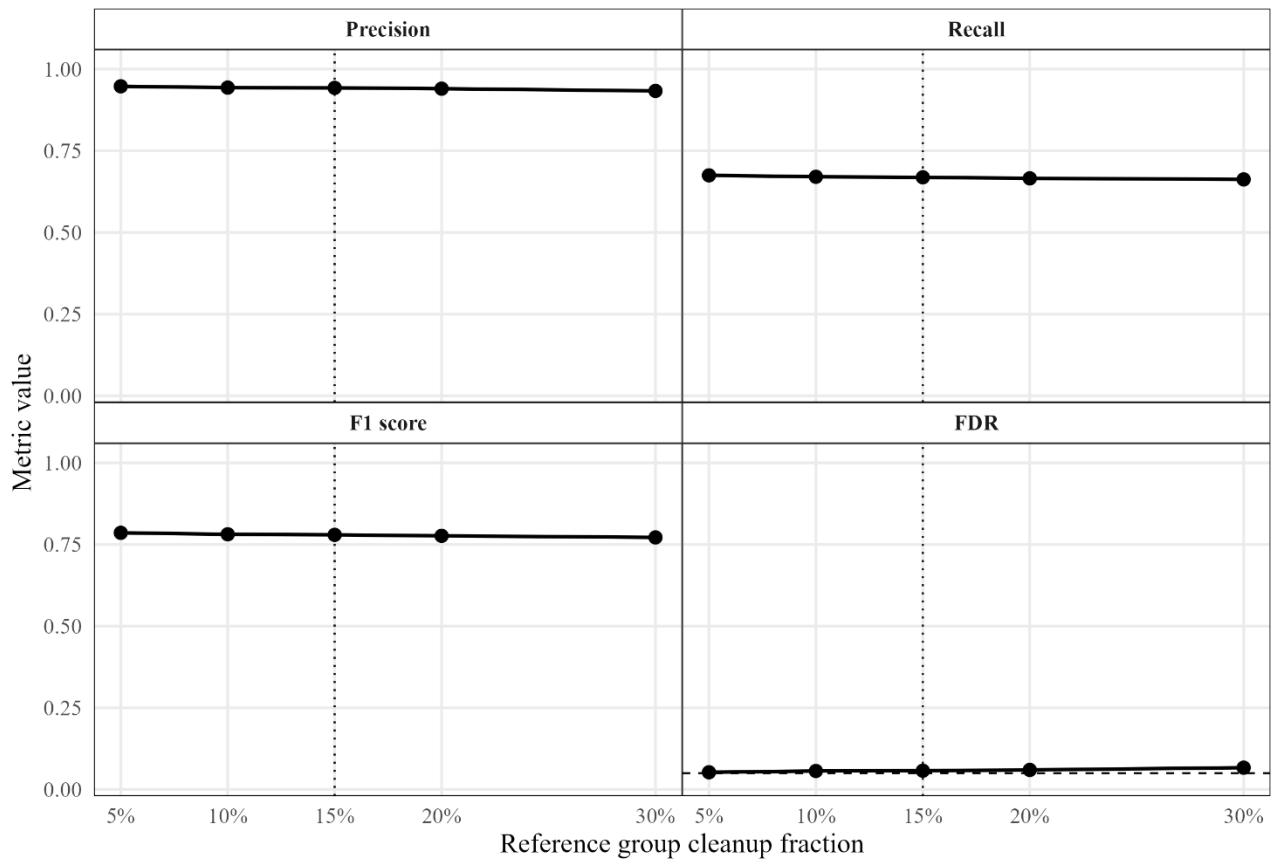


Figure A2.2. PURSUE performance sensitivity to the reference group cleanup. Mean precision, recall, F1 score, and FDR were calculated across repeated simulations of the baseline non-null setting while varying only the fraction of reference candidates removed during group-aware reference cleanup. All other PURSUE parameters were held at their baseline values. The dotted vertical line marks the chosen parameter value, and the dashed horizontal line in the FDR panel marks the nominal 0.05 level. There was marginal difference between different values, therefore a middle ground was chosen.

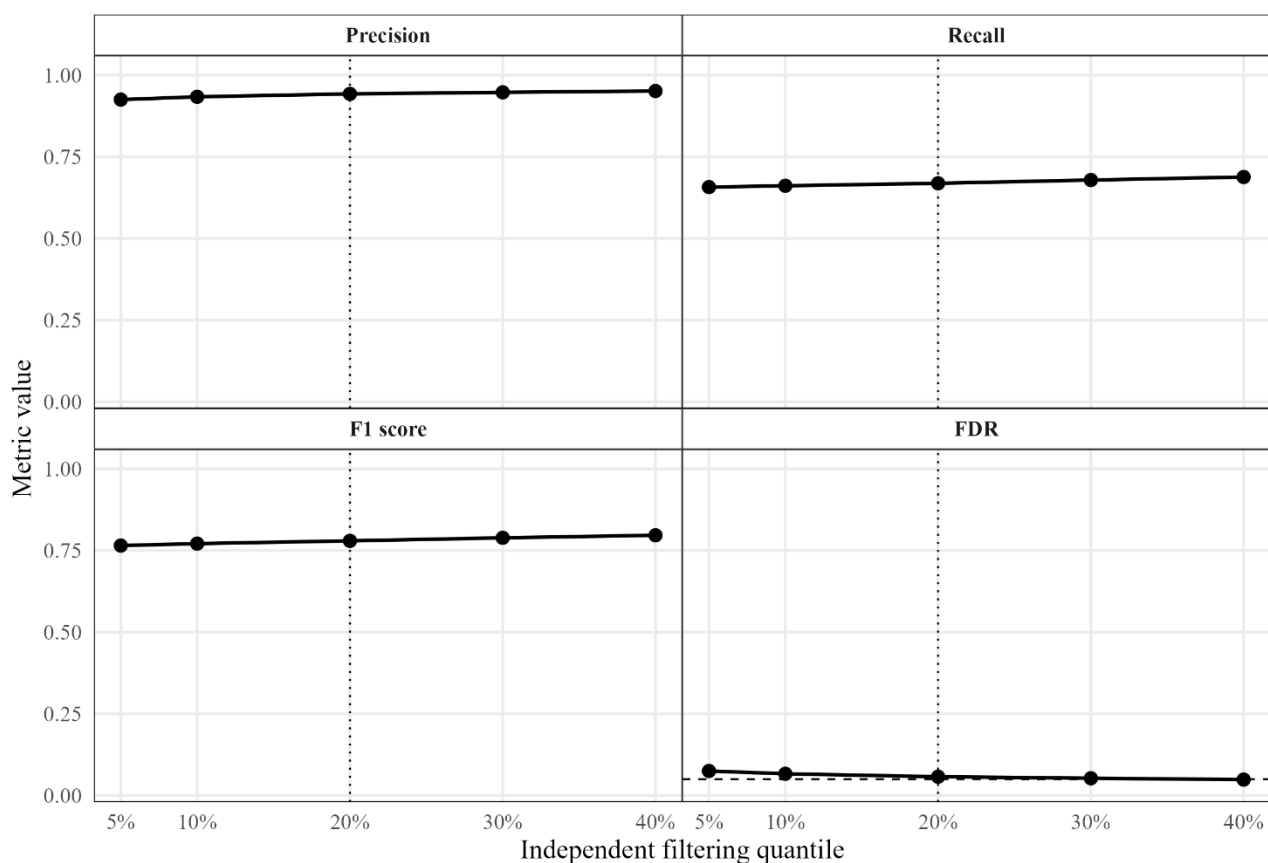


Figure A2.3. PURSUE performance sensitivity to the independent filtering quantile. Mean precision, recall, F1 score, and FDR were calculated across repeated simulations of the baseline non-null setting varying only the independent filtering quantile. All other PURSUE parameters were held at their baseline values. The dotted vertical line marks the chosen parameter value, and the dashed horizontal line in the FDR panel marks the nominal 0.05 level. The 20th percentile was chosen because it was the first value where precision and FDR stabilized; however, at the 30th percentile and onward, a large portion of the p-values would be filtered out, removing substantial amounts of information.