Research plan: Enhancing metabolome analysis with modern data containers in the R/Bioconductor ecosystem

Vilhelm Suksi, 41856

vsuksi@abo.fi

Biochemistry major, bioscience program at Åbo Akademi

Supervisor: Leo Lahti, Turun Yliopisto

2023

Abstract

Insight in how the microbiome relates to human biology is increasingly data-driven, where new data science methodologies need to account for hierarchical, heterogenous and multimodal data. The planned contribution is aimed at improving metabolome and multimodal microbiome analysis by using the modern R/Bioconductor TreeSummarizedExperiment and MultiAssayExperiment data containers applied to the metabolomics workflow suggested by the Notame R package. The work spans implementation of the Notame workflow in the TreeSummarizedExperiment context, integration with microbial abundance data using MultiAssayExperiment and a multimodal analysis contrasting the workflow with other approaches. This will explore and extend the metabolomics support of the R/Bioconductor ecosystem for extracting insight from metabolomic and multimodal microbiome analyses in a user-friendly, reproducible workflow. Using the workflow, substantive research efforts will be better equipped to tackle questions relating the microbiome to physiology and pathology alike.

Contents

1. Introduction 4
2. Research objectives 6
3. Research plan 7
   3.1 Metabolomics pipeline 7
      3.1.1 Data collection 7
      3.1.2 Drift correction and flagging of low-quality features 8
      3.1.2 Quality control 8
      3.1.3 Metaboset to TreeSummarizedExperiment 10
      3.1.4 Missing value imputation 11
      3.1.5 Transformation, normalization and scaling 12
      3.1.6 Clustering molecular features originating from the same metabolite 12
      3.1.7 Univariate analysis 13
      3.1.8 Multivariate analysis 14
      3.1.9 Ranking and filtering for features 15
      3.1.10 Feature-wise visualization 15
      3.1.11 Multivariate visualization 16
      3.1.12 Annotation of metabolites 17
   3.2 Integration with microbiome analysis 17
      3.2.1 MultiAssayExperiment 17
      3.2.2 Cross-correlation analysis 18
      3.2.3 Multimodal factor analysis 18
   3.3 Comparison with Notame and extensive scripting 18
4. Research schedule 19
5. Research synopsis 20
6. References 21

Abbreviations:

LC-MS – liquid chromatography-mass spectrometry

LC-MS/MS – tandem liquid chromatography-mass spectrometry

TSE – TreeSummarizedExperiment

MAE – MultiAssayExperiment

m/z – mass-to-charge ratio

RT – retention time

QC – quality control

1. Introduction

With the advent of new experimental techniques, insight in life science has become increasingly reliant on sophisticated data science methodologies. This also holds true for microbiome analysis, where new data science methodologies need to account for hierarchical, heterogenous and multimodal data[1]. The manipulation, analysis and reproducible reporting of such data is well developed in the R/Bioconductor ecosystem (hence referred to as Bioconductor)[1], focused on high-quality open research software for life science[2]. Bioconductor can be conceptualized as consisting of data containers, R packages and a community of users and developers, who contribute to the ecosystem in an interoperable and modular fashion[1]. The Bioconductor package repository delivers releases consisting of a set of compatible R package versions intended for compatibility only with a certain version of R, allowing for reproducible and reportable analysis.

The notion of data containers arises from the need to organize biological data including assay matrices and metadata such as sample descriptions and feature annotations into a single instance, facilitating the development and usage of complex analysis workflows. For example, it is possible to exclude a sample from both the metadata and assay data in one operation, keeping the metadata and assay data synchronized. In Bioconductor, the SummarizedExperiment family of classes provides data containers

for various research needs[3]. In microbiome research, the TreeSummarizedExperiment (TSE) derivative of SummarizedExperiment allows for storing taxonomical information as a hierarchical tree structure[4]. TSE and other SummarizedExperiment derivatives also come with functions for making efficient use of the container for the research at hand. MultiAssayExperiment (MAE) is used to include and manipulate data from multiple modalities in a single instance[5].

Orchestration of microbiome research in Bioconductor using the TSE lineage of containers and MAE has been explored thoroughly, including basic data manipulation, transformation, exploration and quality control, taxonomic-focused tasks and machine learning [1]. On the other hand, metabolomics support is underdeveloped, as evidenced by the lack of metabolomics packages that support the TSE lineage of containers. The Notame package, detailed in "the Metabolomics Data Processing and Data Analysis—Current Best Practices" special issue of the Metabolites journal presents a contemporary analysis workflow for liquid chromatography-mass spectrometry (LC-MS) research[6]. Few of the analysis steps detailed therein are available in Bioconductor. The maplet package[7], which has been included on/off in Bioconductor repository, supports the TSE lineage of containers and provides some relevant functionalities. Shortcomings include LC/MS data quality control, drift correction, retention time-incorporating metabolite clustering, some univariate methods and multivariate models for different downstream analyses. As such, maplet is not sufficient to implement the Notame workflow. An overview of multimodal microbiome research reflects this lack of metabolomics support in Bioconductor, showing extensive use of scripting to manage all aspects of the analyses.

The proposed work aims to streamline metabolome and multimodal microbiome analyses by implementing metabolomics functionalities as per the Notame workflow using the modern TSE container. The Notame-inspired workflow is showcased in a multimodal and microbiome analysis. Finally, the multimodal analysis workflow is reviewed and contrasted with alternative data analysis approaches. The workflow is expected to benefit metabolome and multimodal microbiome research efforts, ultimately knowledge of biological functions at large.

2. Research objectives

The proposed workflow aims to overcome the above shortcomings in Bioconductor metabolomics support by using the modern TSE container to implement the Notame workflow. Capitalizing on recent advances in Bioconductor data analysis, can complex metabolome and multimodal microbiome research be more streamlined? How does using the modern TSE and Bioconductor compare in terms of reproducibility and reportability? Available methods? What about the margin for user error? And execution time?

Much of what matters here can be summed up in terms of friction at the level of an individual researcher and science collective enterprise. For the individual researcher, friction can be reduced by using sets of compatible packages, performing operations in a single action across assays, and allowing for changes in the analysis workflow without having to rebuild the code around the change. Such simplification of code further reduces friction in the form of easier developing, commenting and documentation. The shorter execution time also reduces friction; a researcher is more likely to re-run the code with different pseudo-random number generator seeds to make sure findings are robust, for example.

From the perspective of science as a collective enterprise, reducing friction makes the work more accessible. For the research project, this makes for a less arduous process for collaborators and peer-reviewers alike. After publishing, more people grasping and re-running the code results in verification and error-correction at scale. Perhaps a seed was cherry picked, resulting in further wasted research efforts?

In other words, minimizing friction in code is not only in line with good practice, but translates to reproducibility and the advancement of open science at large. To highlight the reproducibility aspect of the proposed workflow, the thesis is written using Quarto[8], a computational document system which generates an output file of desired format based on text and code input. To explore writing of a thesis as a computational document is a research objective in itself, in an age where the lack of reproducibility undermines the pursuit of science.

3. Research plan

3.1 Metabolomics pipeline

3.1.1 Data collection

Mass spectrometry-driven metabolomics research requires expertise in analytical chemistry, biochemistry, bioinformatics and data analysis[6]. Biological samples are typically analysed using tandem liquid chromatography-mass spectrometry (LC-MS/MS), in which analytes in the liquid phase are separated in a chromatography column and ionized for detection in a tandem mass spectrometry system, where analytes are fragmented and further ionized for detection for improved specificity[6]. LC-MS/MS instrumentation and analysis conditions can differ substantially among laboratories and experiments, necessitating careful data collection to achieve reproducible results[6]. There are several approaches to data collection, but the general picture is as follows. First, the mass-to-charge ratio of ionized analytes is detected, and the resulting raw data undergoes algorithmic peak-picking to sort out the signal from each analyte[9]. Second, the retention time (RT), or the time an analyte takes to pass through the chromatography column, is calculated for each analyte based on computed analyte peak areas or the raw data from the non-fragmented LC-MS spectra[9]. Finally, since the RT for each analyte varies between samples, RTs are aligned to identify corresponding analytes across samples[9]. These data collection steps are implemented using the MS-DIAL software[10], (version 3.70), using the parameters detailed in the Notame workflow.

The data is now ready for preprocessing, annotation and analysis, where the areas of the peaks reflect the relative abundancies of analytes in the sample, while the identities of the peaks are teased apart by the m/z and RT of analytes (hence called features). RT and LC-MS/MS spectra are also included in analyte identification as per the most robust identification level as specified by the Metabolomics Standards Initiative[11].

3.1.2 Drift correction and flagging of low-quality features

LC-MS measurements suffer from systematic intensity drift, the removal of which increases the quality of the data by minimizing variance introduced by the experimental methodology while conserving the biological variance of interest[12]. Quality control (QC) samples included at regular intervals in the LC-MS measurement, consisting of aliquots from each sample, are used to remove the drift by subtracting values according to a smoothed cubic spline fit to QC samples by injection order. Features are temporarily log-transformed to better meet the assumptions of the smoothed cubic spline model. Low-quality features, as identified by detection in less than 70% of the QC samples or having an RSD < 0.2 and D-ratio F < 0.4 [13], are flagged and monitored by visualizations before and after drift correction. The D-ratio compares the standard deviation of the QC samples to relative to the standard deviation of the samples for each feature[13]. Features identified as low-quality after drift correction are flagged and not included in downstream analysis. These steps are implemented as per the Notame workflow.

3.1.2 Quality control

Linear models relating each feature to injection order are fit to visualize the effect of drift correction on individual features by drawing histograms of the p-values for the regression coefficient of the models. The p-values should follow a uniform distribution represented by a horizontal line under the null hypothesis, namely that p-values across features are normally distributed. This would indicate that systematic drift has been removed. Such histograms are also drawn for QC samples and biological samples. To visualize the effect of drift correction, these histograms are drawn before and after drift correction, as is the case with subsequent quality control visualizations. For example, a comparison of histograms for QC samples shows how the p-values of the features tend towards zero before drift correction, whereas after drift correction the p-values tend towards one. Before drift correction feature p-values do not follow a normal distribution because of systemic drift. In other words, we can reject the null hypothesis, namely that sample injection order has no effect on the features' intensities. After drift correction, the within-group variance of features in QC samples

is very small because the predictor, injection order, is no more associated with the response, intensity. This indicates that drift correction was successful.

To visualize systematic drift in global feature intensities across samples, boxplots representing the distribution of feature intensities in each sample are drawn. The median is represented by a line, with the interquartile range as boxes and whiskers at values max 1.5x of the interquartile range. Drawing these boxplots before and after drift correction typically shows a systematic decrease in signal intensity as a function of injection order.

Features are then normalized by subtraction of mean and division by standard deviation. This allows for visualization of Euclidean pairwise distances between features across samples using a density plot. Drawing such density plots before and after drift correction hopefully shows how features' intensities are more similar after drift correction, especially the for the QC samples which should group independently of the biological samples.

A dimensionality reduction technique such as principal component analysis (PCA) or, if the data is non-linearly separable, uniform manifold approximation and projection (UMAP) is then applied to visualize patterns in the data according to study group and QC sample membership. Trends in the biological samples may not be apparent before or after drift correction, but the QC samples should group tightly after drift correction. Samples can also be colored by gradient according to injection order, where after drift correction, any trends should be dissipated. If too many samples make the dimensionality reduction plot hard to interpret, hexbin versions are drawn such that each hexagon is colored by the mean of the injection orders of the points in the hexagon.

Finally, hierarchical clustering using Ward's criterion on Euclidean distances between samples is used to visualize the sample clusters in a dendrogram, where the QC samples should cluster together early after drift correction. Using the same clustering methodology, a heatmap is drawn to represent pairwise distances between sample clusters on the axes. With quality control finished, QC samples are discarded.

All of the above quality control steps are implemented as per the Notame workflow.

3.1.3 Metaboset to TreeSummarizedExperiment

Abstraction, in general, is the process of paring something down to a set of essential elements for the work at hand. Data containers meet this description in simplifying the representation of data, while hiding its complexities and associated operations. For example, instead of storing metadata in a separate table and accessed by extensive scripting, metadata is stored in the same instance and accessed by user-friendly operations.

Bioinformatics projects typically require a data container with a count matrix, sample descriptions and annotation functionality. The MetaboSet container used for the above parts of the analysis workflow is a Bioconductor base package ExpressionSet container derivative. ExpressionSet was designed for array-based experiments and can store only one count matrix per instance[2]. This results in back-and-forth or creation of new MetaboSet instances to handle transformations and other data manipulation steps, complicating the workflow.

Although many packages interface with ExpressionSet and are largely compatible with MetaboSet, new developments in the fast-moving field of reproducible computation in the R/Bioconductor ecosystem increasingly leans on the modern SummarizedExperiment family of containers. SummarizedExperiment is also based on the ExpressionSet class, but is more flexible, highly optimized and can store multiple count matrices in a single instance[3]. The TSE derivative adds to the functionality of SummarizedExperiment by facilitating storage of hierarchical structure of the data as per the phyloseq package for exploring microbiome profiles[4]. Although multimodal microbiome analysis is a downstream concern, storing the hierarchical structure of data can also be used to track clustering of samples and features in metabolome analysis. Moreover, since TSE is derived from SummarizedExperiment via RangedSummarizedExperiment and SingleCellExperiment, it also includes functionality for representing ranges, adding low-dimensional representations, adding alternative features sets, storing data pairings and addition of further metadata fields. To top it off TSE is compatible with packages that interface with any of the above containers in the SummarizedExperiment family[4], further extending the functionality and relevance of TSE into the future.

An overview of alternative data containers for metabolomics, chiefly a suite of data containers included in the RForMassSpectrometry initiative, show promise but do not match the flexibility of TSE. The RForMassSpectrometry suite of containers does not support RT data, used for clustering of features and identification of metabolites as per the most robust identification level as specified by the Metabolomics Standards Initiative.[14] Another concern is that the RForMassSpectroMetry suite of containers rely on the legacy eSet data container[14].

Regarding multimodal support, the MSexperiment integrative data container from the RForMassSpectrometry initiative shows promise in being based on MAE, but only one slot is available for storage of data from other modalities[14]. Thus, the MSexperiment container is rejected in favor of MAE, which allows for differing numbers of samples and features for data from different modalities[5].

To this end, the MetaboSet instance used in the above parts of the analysis is converted into a TSE instance, which is included in a MultiAssayExperiment instance for downstream multimodal microbiome analysis. The converter is based on the makeSummarizedExperimentFromExpressionSet() function from the SummarizedExperiment package.

3.1.4 Missing value imputation

Values missing due to being below an instrument's limit of detection are often referred to as missing not at random (MNAR)[15]. Missing values caused by processing errors are often referred to as missing completely at random (MCAR), because they are uniformly distributed across the dataset and are not missing due to any property of the metabolite or measurement itself[15]. The Notame workflow deals with MNARs and MCARs by flagging values that are not detected in > 50% of the samples. This is a heuristic method that should flag all MCARs but doesn't explicitly deal with the detection threshold and probably flags features that are barely detectable. The starting point for imputation in the Notame workflow thus involves features that are detected in over > 50% of samples, which probably removes features that are at the detection threshold of the instrument. However, if any MNARs from around the detection threshold remain unflagged, they may impact the random forest model used for imputation in the Notame workflow. Moreover, the Notame workflow suggests

imputing flagged features after imputation of non-flagged features. This risks imputing MNARs as MCARs.

Thus, imputation is performed using the MAI (Mechanism-Aware Imputation) package, which features a two-step approach; first missing values are classified as MCAR or MNAR, after which random forest imputation is applied to predict MCARs and single imputation or No-Skip k-nearest neighbors is applied to predict MNAR values[15]. Random forest imputation has been shown to outperform other methods in LC-MS data[16]. A small α parameter value from the model, indicating that MCAR values are predominant, can be expected because of the filtering of values that are not detected in >50% of the samples. Conveniently, the MAI function accepts TSE as an argument. A potential downside is that imputation parameters can not be adjusted.

3.1.5 Transformation, normalization and scaling

In the Notame workflow, data is next transformed using the natural logarithm or a weighted logarithm if the data is very skewed. In favor of a simplified workflow, the weighted logarithm option is ignored, and the natural logarithm applied using the mia package. Alternatively, the weighted logarithm is contributed to a Bioconductor package that supports TSE. The data is then normalized using probabilistic quotient normalization using the lipidr package. Finally, for multivariate analysis, the data is standardized using the mia package.

3.1.6 Clustering molecular features originating from the same metabolite

MS-DIAL combines isotopes, most common adducts and in-source fragments into a single feature[10]. Still, redundant representation of the same metabolite due to unpredictable adduct behavior can not be ruled out[6]. Thus, features need to be combined by clustering to represent unique features[6]. This is done separately for each mode in the dataset. In the Notame workflow, features are clustered and combined anew based on correlated feature pairs within a specified retention time window and correlation threshold. This clustering method is bespoke to the Notame workflow, but similar functionality is provided by the cliqueMS Bioconductor package. However, cliqueMS is very tightly integrated with the xcms package for processing MS data.

This means that using cliqueMS in the TSE context involves extensive manipulation and conversion of data, if at all possible. Due to the substantial amount of work involved to use cliqueMS as is, other options are explored with help from my supervisor. Options include contributing the Notame feature clustering methodology to a Bioconductor package supporting TSE, perhaps the maplet package, or opting for another feature clustering methodology, such as provided by the MsFeatures package.

After feature clustering, the modes are merged. This results in a dataset with some redundancy as many features are detected in multiple modes. The dataset is now ready for analysis.

3.1.7 Univariate analysis

For case versus control studies with two groups and no covariates, Welch's t-test is used as it allows for unequal variances between groups. The Mann-Whitney U-test can be used as a non-parametric alternative. Welch's t-test and the Mann-Whitney U-test are available in the POMA package and can take TSE as an argument.

For studies with multiple groups, Welch's ANOVA, which allows for unequal variances, is used to identify interesting features based p-value. To investigate differences between multiple groups, pairwise Welch's t-tests are used. Welch's ANOVA is available in the POMA package.

In the case of two categorical study factors, two-way ANOVA is applied to examine the main effect of each factor and their interaction. If factors have multiple levels, interesting features are selected based on overall p-values and features can be examined using pairwise Welch's t-tests. Friedman test can be used as a non-parametric alternative to two-way ANOVA. Two-way ANOVA is available in the POMA package, while Friedman test can be applied using the stats package, although it doesn't support TSE.

If the study design includes multiple time points, a linear mixed effects model is used with the time point, group and interaction factors as fixed effects and subjects as random effect. To assess the significance of effects between no more than two groups or time points, t-tests are applied on the regression coefficients of the mixed effects model. If several groups and/or time points are used, type III ANOVA is used,

returning p-values from an F-test. Linear mixed effects model is available in lmerTest package, including type III ANOVA.

To investigate the association between molecular features or between molecular features and other variables, Pearson correlation or the non-parametric Spearman correlation is used. These are available in the POMA package.

Finally, p-values are adjusted using the Benjamin-Hochberg false discovery rate (FDR) approach for multiple testing. This is done to correct for the chance of obtaining significant results in a situation where, on average, a significance threshold of 0.05 for twenty tests would give a false positive for one test[17]. FDR is available in the POMA package.

3.1.8 Multivariate analysis

The Notame workflow specifies prediction and feature selection using the MUVR package, featuring an unbiased feature selection using random forest or partial least squares. Selection bias is introduced when features are selected based on the training set used to estimate prediction error in a cross-validation scheme, overfitting the model to the samples used in the training set[17]. MUVR minimizes bias by repeatedly and randomly sampling the entire dataset to a training set and a test set. For each outer repetition, a validation set is randomly sampled from the training set to select the optimum model parameters from models fit in a cross-validation scheme on the remaining training set. A proportion of low-ranking features are eliminated, and optimum model parameters and average feature ranking is obtained by repeating model fitting and validation on the reduced and anew randomly sampled training and validation set. The optimum model is then trained on a combined validation and training set, used to obtain the optimum model and feature ranking for the test set in the given outer repetition. With several outer repetitions, this methodology arrives at optimally ranked features and model parameters for the entire dataset. The above was for relative simplicity, as MUVR actually returns a “min”, a “mid” and a “max” model tailored to the analytical problem.

This methodology is not available in any Bioconductor package. For the proposed workflow, options include contributing the MUVR package methodology to a

Bioconductor package or opting for an optimal model from a cross-validation scheme, from which feature ranks can be obtained in different ways. This is worked out with help from my supervisor.

3.1.9 Ranking and filtering for features

Feature ranking from multivariate and univariate analyses are combined to determine the most biologically relevant features for identification. First, the ranks from the multivariate analysis are sorted such that the most predictive feature comes first. Features from each univariate model are sorted similarly based on the FDR-adjusted p-values, called q-values. Ranks are then combined and sorted across multivariate and univariate analyses to create a final ranking. This is done using the R base package.

3.1.10 Feature-wise visualization

Feature-wise visualizations are then used to facilitate broad comprehension of differences between study groups and/or time points. Data visualization will inevitably shape interpretation of the data.

Differences in mean feature abundance between groups are visualized by separate beeswarm plots for each group. The distribution of p-values from univariate analysis between groups or time points is depicted in a histogram, with the uniform distribution expected under the null hypothesis represented by a horizontal line. To relate p-values from univariate tests to fold change or other effect size measure, volcano plots are used with a horizontal line representing the significance threshold for q-values. The relation of p-values to biochemical properties of the features, such as m/z or RT, is visualized using a Manhattan plot, again with a horizontal line representing the significance threshold for q-values. The Manhattan plot is drawn separately for each mode. To visualize the relation of p-values to m/z and RT, a scatter plot with point sizes reflecting p-value is drawn, again separately for each mode. To assess the impact of intervention or time point and the number or proportion of metabolites behaving in a certain manner, a heat map is used with k-means clustered features using Pearson's correlation and average linkage clustering. The optimal number of clusters for

visualization is arrived at by sequentially increasing the number of clusters until no more clusters with a unique pattern emerge.

In case of multiple time points, the abundance of a given feature is visualized with a line plot featuring one line per sample and a thicker line representing the mean of the feature across samples. A similar plot featuring lines colored by groups and group-wise means is also included. Least square means from mixed effects models are plotted with whiskers representing 95% confidence intervals or some other measure of variability.

Many of the visualizations are implemented in the maplet package, which currently isn't included in the Bioconductor repository. The maplet package fails to build, so the relevance of the visualizations can't be assessed thoroughly. In case the maplet package is reintroduced to the Bioconductor repository, some visualizations are contributed. Alternatively, the Notame visualizations could be drawn using the tidySummarizedExperiment package which provides plotting functionalities for containers in the SummarizedExperiment family.

3.1.11 Multivariate visualization

Unsupervised dimensionality reduction, such as PCA or UMAP in case of non-linearly separable data, is applied and visualized to reveal patterns in the data. In the case of PCA, principal component loadings are plotted to reveal the contribution of features to the principal components. To visualize global changes in samples across multiple time points, PCA and UMAP plots with arrows connecting the time points for each sample are drawn.

If PLS-DA is used to assess to what extent features can predict study group or time point, the samples are visualized in a PLS score plot with differently colored areas representing class label.

The maplet package implements standard dimensionality reduction visualizations, but the more intricate plots are best created using the tidySummarizedExperiment package.

3.1.12 Annotation of metabolites

MS-DIAL is used for semi-automated metabolite annotation as per the Notame workflow. In short, experimental characteristics of metabolites, including m/z and RT, are compared with those in reference databases. Metabolites with a similarity score of 80% or above are annotated and manually curated by assessing the similarity of spectra between the work at hand and the reference database. Alternative annotations proposed by MS-DIAL are also explored. Remaining unknown metabolites are annotated by manual, additional searches of reference databases specializing in molecular groups like lipids.

Metabolites annotated by matching m/z, MS/MS spectra and RT are considered identified in accordance with the Metabolomics Standards Initiative. Metabolites with matching m/z and MS/MS spectra are considered as putatively identified if the MS/MS spectra only matches one reference metabolite. Finally, putative characterization status is given to metabolites for which only compound class can be established.

3.2 Integration with microbiome analysis

3.2.1 MultiAssayExperiment

Since microbiome analysis is already highly developed in Bioconductor using TSE, details of microbiome analysis are not relevant herein. In short, amplicon sequence variant tables from paired-end Illumina sequencing demultiplexed by sample are processed to identify sequences with taxa, for which relative abundancies are calculated. The amplicon sequence variant table is a higher-resolution analogue of the traditional operational taxonomic unit table and records the number of times each exact amplicon sequence variant was observed in each sample.[1]

Relative abundancies of microbial taxa are included in a MAE instance along with identified metabolic feature abundancies. This allows for easy data manipulation across modalities and a tidy workflow.

3.2.2 Cross-correlation analysis

To assess the association between microbial taxa and metabolites abundancies, cross-correlation with heatmap visualization is used, as implemented by the mia package.

3.2.3 Multimodal factor analysis

Multimodal factor analysis is an unsupervised method that can be seen as a generalization of PCA, as it can infer a low-dimensional representation from multiple modalities[18]. Visualized results include box plots featuring factor loadings, that is how much of the variance is explained by microbial and metabolite abundancies. Factor loadings are also visualized for each modality to assess the contribution of individual taxa or metabolites. Multimodal factor analysis is carried out using the MOFA package.

3.3 Comparison with Notame and extensive scripting

The proposed workflow is assessed for ease-of-use, reproducibility and reportability, execution time and other metrics of interest. Workflow methodologies are unlikely to be identical between Bioconductor, Notame and extensive scripting, and differences in methodology are used to map the space of metabolomics functionality in R. Since methods to potentially contribute for the proposed workflow are evaluated on a case-by-case basis depending on the plausibility of contributing a method given the state of Bioconductor, a sense of metabolomics support and future directions in the Bioconductor is gained. Methods specified in the Notame workflow which are outside the scope of this work to contribute are addressed as limitations in Bioconductor. For example, if contributing the unbiased multivariate analysis method mentioned above is deemed unpractical because there are no suitable packages in Bioconductor to contribute to, this presents a shortcoming in the package interplay for metabolome analysis in Bioconductor. Such situations may also arise from the usage of the TreeSummarizedExperiment data container, as much of metabolomics support in Bioconductor leans on other data containers. Many metabolomics packages in

Bioconductor are perhaps too tightly integrated with a specific workflow, lacking modularity.

The above concerns can elicit discussion about package structure and how to advance metabolomics support in Bioconductor. To approach the concerns systematically, parallel analyses are performed using the proposed workflow, the Notame workflow and extensive scripting. The latter workflow would not be restricted to Bioconductor, perhaps even incorporating methods from other programming languages. Alternatively, select aspects of the analysis approaches are compared without support from parallel analyses.

4. Research schedule

June	Conceptualize the project, gain oversight and write research plan
July	Implement workflow data collection and quality control, write thesis
August	Implement workflow steps leading up to data analysis, write thesis
September	Implement workflow statistical analyses, write thesis
October	Implement visualizations, write thesis
November	Implement multimodal microbiome analysis, write thesis

The project will be done over a half-year period so as to graduate at the end of 2023. The thesis itself is written in parallel with computational work because of the nature of the work, where writing helps in seeing the big picture and staying on track. Moreover, much of the work proceeds chronologically, which is convenient for parallel writing of the thesis.

There are several steps in the workflow which can be implemented in multiple ways. Such situations are addressed in chronological order with my supervisor. Performing parallel analyses using different analysis approaches is a putative component, the feasibility of which becomes clear when working on the project.

5. Research synopsis

The proposed work aims at exploring and expanding metabolomics functionality in the Bioconductor ecosystem using TSE, as inspired by the Notame workflow. The workflow presented therein is expected to showcase reproducible analysis and its integration in multimodal microbiome research. This sets the stage for substantive metabolome and multimodal microbiome research efforts in Bioconductor, resulting in the advancement of open science and insight in complex, biological systems, which in practice may translate into health outcomes.

6. References

1. Lahti et al. (2021). Orchestrating Microbiome Analysis with R and Bioconductor. https://microbiome.github.io/OMA/intro.html (accessed in July 2023)
2. Gentleman et al. (2004). Bioconductor: open software development for computational biology and bioinformatics. Genome biology 5:80.
3. Morgan et al. (2021). SummarizedExperiment: SummarizedExperiment container, version 1.24.0.
4. Huang et al. (2021). TreeSummarizedExperiment: a S4 class for data with hierarchical structure. F1000Research, 9:1246.
5. Ramos et al. (2017). Software For The Integration Of Multiomics Experiments In Bioconductor. Cancer Research, 77:21.
6. Klåvus et al. (2020). "Notame": Workflow for Non-Targeted LC–MS Metabolic Profiling. Metabolites, 10: 135.
7. Chetnik et al. (2021). Maplet: an extensible R toolbox for modular and reproducible metabolomics pipelines. Bioinformatics, 38:4.
8. Allaire et al. (2022). Quarto, version 1.2. https://doi.org/10.5281/zenodo.5960048 (accessed in July 2023)
9. Basics of Liquid Chromatograph-Mass Spectrometry. Shimadzu. https://www.shimadzu.com/an/service-support/technical-support/liquid-chromatograph-mass-spectrometry/index.html (accessed in July 2023)
10. Tsugawa et al. (2015) SWATH-MS/MS and DIA-MS: MS-DIAL: data independent MS/MS deconvolution for comprehensive metabolome analysis. Nature Methods, 12.
11. Sumner et al. (2007). Proposed minimum reporting standards for chemical analysis. PMC Metabolomics, 3:3.
12. Märtens et al. (2023). Instrumental Drift in Untargeted Metabolomics: Optimizing Data Quality with Intrastudy QC Samples. Metabolites, 13:5.
13. Broadhurst et al. (2018). Guidelines and considerations for the use of system suitability and quality control samples in mass spectrometry assays applied in untargeted clinical metabolomic studies. Metabolomics, 14:72.
14. Gatto et al. (2023) R For Mass Spectrometry. https://rformassspectrometry.github.io/docs/ (accessed in July 2023)
15. Dekermanjian et al. (2022). Mechanism-aware imputation: a two-step approach in handling missing values in metabolomics.BMC Bioinformatics, 23: 179.

16. Kokla et al. (2019). Random forest imputation outperforms other methods for imputing LC-MS metabolomics data: a comparative study. BMC Bioinformatics, 20:492.
17. Jafari et al. (2019). Why, When and How to Adjust Your P Values? Cell Journal, 20:4.
18. Argelaguet et al. (2018). Multi-Omics Factor Analysis—a framework for unsupervised integration of multi-omics data sets. Molecular Systems Biology, 20:14