metaboPipe: a Modular Pipeline for Metabolomic Data Preprocessing

Eduard Pérez Méndez

Statistical Bioinformatics and Machine Learning

Master's degree in Bioinformatics and Biostatistics

Name of the tutor:
Alexandre Sánchez Pla

Name of the SRP:
Carles Ventura Royo

May 23, 2024

Universitat Oberta de Catalunya

Universitat de Barcelona

Final Work Card

Title of the work:	metaboPipe: a Modular Pipeline for Metabolomic Data Preprocessing
Name of the author:	Eduard Pérez Méndez
Name of the tutor:	Alexandre Sánchez Pla
Name of the SRP:	Carles Ventura Royo
Date of delivery:	May 23, 2024
Studies or Prgram:	Master's degree in Bioinformatics and Biostatistics
Area or the Final Work:	Statistical Bioinformatics and Machine Learning
Language of the work:	English
Keywords:	targeted metabolomics, preprocessing, pipeline

Abstract

A maximum of 250 words, detailing the purpose, contexto of application, methodology, results and conclusiones of the work.

"Never let your sense of morality stop you from doing the right thing"

Isaac Asimov

Contents

List of Figures

List of Tables

1. Introduction

1.1 Problem description

Metabolomics, a powerful and evolving field within the realm of systems biology, plays a pivotal role in unraveling the intricate web of biochemical processes occurring within living organisms. As we delve into the molecular intricacies of biological systems, the generation of vast and complex datasets poses a significant challenge. Challenges in standardizing nutritional metabolomics include experimental design, sample preparation, and data analysis, which impact result validity and reproducibility. Efforts by the international community aim to establish standard procedures and infrastructure for advancing nutritional metabolomics research. This master thesis project aims for the creation of a modular pipeline designed to streamline the processing of targeted metabolomics data to a usable and meaningful dataset for further analysis and biological interpretation.

1.2 Context and justification

Metabolomics is a rapidly evolving field within biology that focuses on the comprehensive study of the metabolite composition of cell types, tissues, organs, or organisms [1–3]. It aims to measure, identify and (semi-)quantify those metabolites. Metabolites are chemical compounds that undergo analysis through conventional chemical assessment methods like Mass Spectrometry (MS) and Nuclear Magnetic Resonance (NMR) spectrometry. MS approaches are commonly integrated with Gas Cromatography (GC) and Liquid Cromatography (LC), leading to the development of two advanced techniques known as Gas Cromatography-Mass Spectrometry (GC-MS) and Liquid Cromatography-Mass Spectrometry (LC-MS). All of these analytical platforms and methodologies generate large amounts of high-dimensional and complex experimental raw data.

However, the statistical analysis of metabolomics data presents significant challenges, attributable not only to the inherent complexity of metabolomics as a research discipline but also to the intricate nature of the data itself. Notwithstanding that numerous studies have explored various methodologies for metabolomic data management, the field still lacks an accepted standard for preprocessing and pretreatment of such data.

One of the obstacles the field encounters is the lack of well defined terminology, as the terms “data preprocessing” and “data pretreatment” have not been used consistently in metabolomics literature [4].

The objectives of data preprocessing/pretreatment encompass two primary aims: firstly, to rectify or mitigate instrumental artifacts and extraneous biological variance, thereby amplifying the Signal-to-Noise Ratio (SNR); and secondly, to effectively transform the data into interpretable spectral profiles through processes such as centering, scaling, and dimensionality reduction [4, 5]. The choice of preprocessing and pretreatment methods can signifi-

cantly impact the downstream analysis and interpretation of metabolomic data [6] so the steps should be carefully selected based on the specific characteristics of the data and the research.

By establishing a standardized approach to preprocess and pretreat metabolomic data, the field can improve the quality, comparability, and reproducibility of metabolomic studies. This would facilitate data integration, enable the development of robust statistical models, and enhance our understanding of the complex metabolic processes underlying health and disease.

1.2.1 Preprocessing of data

Given the inherent dissimilarities in data acquisition techniques, unique preprocessing procedures are imperative before embarking on statistical analyses in metabolomics investigations. NMR spectra, for instance, often exhibit signal shifts along the axis due to factors like pH fluctuations [7]. Thus, meticulous preprocessing is indispensable to ensure robust statistical analyses and facilitate inter-spectral signal comparisons. This involves techniques such as binning, peak fitting with spectral databases, and exclusion of unstable or non-informative spectral regions (e.g., water peaks) [3, 4, 8]. By refining the dataset to a subset of relevant metabolites, statistical methods can effectively discern variations in signal intensity among sample groups [9].

The preprocessing workflows diverge between MS-based and NMR-based metabolomic analyses. In MS-based profiling, data are presented as three-dimensional (3D) tables, in contrast to the two-dimensional (2D) tables derived from GC-MS data preprocessing [4, 8]. GC-MS preprocessing entails deconvolution and peak integration to generate intensity profiles for each sample feature corresponding to RT/m/z pairs. Notably, metabolite identification strategies differ between GC-MS and LC-MS methodologies. While GC-MS relies on reproducible mass spectra and extensive databases for metabolite identification based on characteristic fragment ions, MS-based methods prioritize automation, accuracy, peak identification, integration, and annotation [10, 11].

While the primary objective of preprocessing is to render data comparable across samples despite instrumental discrepancies, the strategies employed in MS-based methodologies differ from those in NMR-based approaches. Moreover, variations exist between preprocessing methodologies utilized in GC-MS and LC-MS metabolomic analyses, underscoring the intricate nature of metabolomics data preprocessing.

MS-based data preprocessing

MS-based analysis involves the measurement of Mass-to-Charge Ratio (m/z). When combined with either LC or GC, the resulting raw GC/LC-MS data encompass three measured variables: m/z, chromatographic Retention Time (RT), and intensity count, thereby constituting a three-dimensional (3D) data structure. To streamline the data and eliminate spectral noise and irrelevant biological variability, a two-dimensional (2D) features table is generated through peak picking. This table encompasses all quantified metabolic features from the analyzed samples, with rows corresponding to samples and columns representing variables such as peak areas or intensities, characterized by m/z and retention time in minutes or scan

number (m/z-RT pairs). The preprocessing of MS data involves several steps: 1) denoising and baseline correction; 2) alignment across all samples; 3) peak picking; 4) merging the peaks; and 5) creating a data matrix [3, 4, 10, 12–17].

NMR-based data preprocessing

Similar to MS-based analysis, NMR-based analysis generates a 2D structure of feature data matrix with the samples in the rows and the spectral data points in the columns. Also similar to MS-based analysis, the NMR-based analysis (e.g., 1H NMR analysis) requires data preprocessing to mitigate non-biologically relevant effects. The following data preprocessing steps could be performed: 1) baseline correction; 2) peak binning; 3) peak alignment; 4) quality control; 5) create a data matrix [4, 5, 15–20]. Preprocessing by either MS or NMR constructs a data matrix containing the relative abundances of a set of mass spectra for a group of samples or subjects under different conditions. The metabolomics data matrix are typically constructed in such a way that each row of the data matrix represents a subject and each column represents the mass spectra (metabolite intensities or metabolite relative abundances, peak or peak intensities).

1.2.2 Pretreatment of Data

Handling Missing Values

Within datasets, missing values or zeros can arise due to a variety of factors, both biological and technical in nature. Categorizations by Sun Xia delineate these zeros into four distinct categories: 1) Structural zeros, 2) Sampling zeros, 3) Values below the limit of detection (LOD), and 4) Zeros derived from negative values that are automatically transformed.

1. **Structural zeros** pertain to peaks absent from a sample or chromatogram due to genuine biological absence rather than technical errors. For instance, if a compound is not present in a biological sample, the corresponding peak for that compound is deemed a structural zero.

2. **Sampling zeros** refer to peaks present in samples but missed during peak picking.

3. **Values below LOD** represent intensities or abundances falling below the detection limit of the mass spectrometer.

4. **Negative value zeros** result from negative intensity or abundance values, considered spectral artifacts or noise, and subsequently transformed to zero.

Identifying the origins of these zeros poses a challenge, and their prevalence presents a significant obstacle for statistical analyses [4, 21]. Hence, practical approaches for managing zeros include:

1. **Filtering** based on a threshold, such as the 80% rule.

2. **Imputation** techniques, which can involve substituting zeros with the mean, minimum (or half of the minimum) of non-missing values, or simply zero.

3. Utilizing **missing data estimation algorithms** to employ various methods for handling missing values.

However, it's crucial to recognize that valuable biological insights may be embedded within peaks containing missing values.

Managing Outliers

Various methods exist for addressing outliers, including:

1. Assessing metabolite peak areas and comparing the ratio of mean to median, with the median often considered more robust in the presence of outliers.
2. Employing Principal Component Analysis (PCA) to identify outliers, followed by techniques such as Principal Component Partial R-square (PCPr2) and Analysis of Variance (ANOVA).
3. Recent advancements have introduced specialized algorithms for outlier identification in metabolomic data, such as cellwise outlier diagnostics using robust pairwise log ratios (cell-rPLR) and a kernel weight function-based biomarker identification technique.

Normalization

Normalization is a crucial step in data preprocessing that seeks to eliminate unwanted variations between samples. By doing so, it ensures that samples can be directly compared to each other by eliminating or reducing systematic errors, biases, and experimental variance [22].

Normalization of data within metabolomic workflows can occur either during sample analysis (preanalytical normalization) or during postanalytical data processing. Normalization of samples is essential due to variations in composition influenced by factors like time of day, health status, and dietary intake.

For instance, blood samples may not require normalization due to the body's control over blood volume and composition. However, urine samples may necessitate normalization due to potential concentration variations [23].

Centering and Scaling

Centering aims to shift metabolite concentrations to fluctuate around zero, while scaling adjusts for fold-change differences between metabolites. Both steps are crucial in data preprocessing.

Transformation

Transformation becomes necessary to address data variance after scaling, aiming to correct for heteroscedasticity, convert multiplicative relations into additive ones, and normalize skewed distributions.

1.3 State of the art

Metabolomic data arrives to the researcher in different shapes and forms depending on the method, the instrument used and the company that analyses. Most of the time those companies do a preprocessing of the data, adjusting for baseline correction, peak picking, and alignment. The preprocessing of this data is a crucial step in the analysis, as it can significantly impact the results and the conclusions drawn from the data. For MS-based analysis, tools like XCMS [24] and MZmine [25] facilitate preprocessing, while for NMR data, packages like BATMAN [26] and RAMSY [27] offer robust preprocessing capabilities . Pretreatment techniques include handling missing values, outlier detection, imputation and normalization.

Nevertheless the field lacks a standardized approach to metabolomic data preprocessing, with inconsistencies in terminology and methodologies. Stanstrup *et al.* in their "The metaRbolomics Toolbox in Bioconductor and Beyond" made an extensive revision of both the scientific literature and the R landscape for packages relevant for metabolomic research.

Since there is no consensus on the order those pretreatment techinques should be applied, the researcher has to decide which steps to take and in which order. This can lead to inconsistencies in the results and the conclusions drawn from the data. Furthermore, the amount of packages that there are for performing the preprocessing and pretreatment of metabolomic data can be overwhelming. And thus preparing the data for analysis, changing the order of the steps, or even changing the parameters of the steps can be a time-consuming task.

The project aims to develop a modular pipeline for targeted metabolomic data pretreatment, implemented in R. The pipeline is designed to enhance efficiency and modularity compared to existing solutions. As a gift to the scientific community, this project is free and open source, with detailed documentation and code available in a public repository. It's encouraged continuous community efforts to improve and expand the package.

Add hypothesis

2. Objectives

2.1 Main Objective

1. Develop a new pipeline for the preprocessing of targeted metabolomic data with the aim of improving efficiency and modularity compared to existing pipelines. This new pipeline will be implemented in R.

2.2 Specific Objectives

1. Design a new pipeline for the preprocessing of targeted metabolomic data.
2. Implement the preprocessing pipeline for targeted metabolomic data using the "targets" package to ensure replicability and efficient management of computational resources.
3. Select various targeted metabolomic datasets to validate and optimize the performance of the new pipeline, analyzing its quality and consistency.
4. Make the development accessible to the scientific community by creating detailed documentation and publishing the code in a public repository.

En els objectius específics és on hauries de parlar de la forma en que ho faras (per exemple, fent servir el paquet "targets") i també introduir la selecció i anàlisi d'alguns datasets ilustratius. En aquest camp sovint els datasets ho son tot. Ja he vist que després parles deL "approach taken to achieve this objective", però es que això és el que haurien de ser les tasques i la metodologia.

Resumint, valdria la pena incloure lo del targets i els datasets en les tasques i deliverables i sobretot tenir en compte que caldrà agafar que ja estigui fet o que caldria fer, i per això serà bo que en la revisió de la literatura" hi incloguis, no només, treballs de metabolòmica sino paquets de R que ja estan implementant moltes d'aquestes coses.

3. Sustainable development goals

Our project aligns with multiple crucial Sustainable Development Goals (SDGs) set by the United Nations, fostering global sustainability and development. The primary objectives of our project focus on developing a pipeline to modulate the pretreatment of metabolomics data and creating an R implementation. This solution holds significant potential to support the following SDGs:

SDG 3: Good Health and Well-being

The use of our pipeline has the potential to reduce the time required for metabolomic data research, accelerating the investigation of rare diseases, cancer, and other medical conditions. By expediting research processes, our project contributes to advancing medical science, improving healthcare outcomes, and ultimately enhancing global health and well-being.

SDG 9: Industry, Innovation, and Infrastructure:

Our focus on developing an open-source, well-documented, and user-friendly pipeline fosters innovation and infrastructure development. By opening access to metabolomics research tools, our project empowers individuals from diverse backgrounds to engage in scientific inquiry and innovation, thus promoting inclusive economic growth and technological progress.

SDG 10: Reduced Inequalities:

Through our implementation, we prioritize inclusivity and accessibility, ensuring that individuals regardless of sex, gender, race, wealth, or ability can utilize, learn from, and contribute to our pipeline. By reducing barriers to entry and promoting equal opportunities for participation in scientific endeavors, our project contributes to reducing inequalities and promoting social inclusion.

While our project aims to bring about positive change, it is essential to consider potential negative impacts and ethical considerations. These may include concerns about data privacy and security, particularly in handling sensitive information. Additionally, there may be unintended consequences such as exacerbating existing inequalities in access to technology or inadvertently reinforcing biases in data analysis. Therefore, it is imperative to approach the development and implementation of our pipeline with careful consideration of ethical principles, transparency, and accountability to mitigate potential risks and maximize societal benefits.

4. Approach and methodology

4.1 Methodology

Mención de cuáles son las posibles estrategias para llevar a cabo el trabajo y cuál es la estrategia elegida (desarrollar un producto nuevo, adaptar un producto existente...). Hay que incluir una valoración de por qué esta es la estrategia más apropiada para conseguir los objetivos.

In the context of enhancing the field of metabolomics and improving the efficiency and accuracy of metabolomics reports, the choice of creating a tool for targeted metabolomic data pretreatement is a strategic decision. This tool will be developed in R, a widely used programming language in the field of bioinformatics and biostatistics. The tool will be designed to streamline the pretreatement of targeted metabolomic data, ensuring that the data is ready for further analysis and interpretation.

The tool will be modular, allowing users to select and apply specific pretreatement steps according to their needs and preferences. This modularity design makes the tool flexible, adaptable and more important expandable, as new pretreatement methods can be easily added to the pipeline and thus to the package using existent methods and tools from other packages.

In order to be modular, the package modules should take as inputs the same object type that outputs. Making it easier to chain the modules in any specific order. To achieve this objective, the package should use a main data structure to store the data. In omics data, the most used data structure is the `SummarizedExperiment` from the Bioconductor project [28]. However this data structure is not the most efficient for the pretreatement of the metabolomics data, as the `SummarizedExperiment` dataframes are designed to store the data features in a row-wise manner and the samples in a column-wise manner. Though this may fit well for other omics data, is not the most intuitive way to manipulate the data in the context of metabolomics.

Since targeted metabolomic data uses multiple dataframes to store the diferent types of data, the most intuitive and already developed data structure will be the `DatasetExperiment` from the `structToolbox` package [29]. This data structure is designed to store the data features in a column-wise manner and the samples in a row-wise manner, making it more intuitive to manipulate the data in the context of metabolomics. It also incorporate multiple methods for the manipulation of the data, making it easier to develop the package.

A deployment controller will be used to manage the order of the modules and the data flow between them. This controler will be created using the `targets` package [30], a pipeline toolkit for reproducible research. The `targets` package will ensure that the pipeline is reproducible, efficient and easy to manage.

The `targets` package only deploys the steps needed to complete the pipeline as previous

steps that do not change its output will not be re-run. This makes the pipeline more efficient and faster, as only the steps that need to be re-run will be re-run.

The integration of modularity with a deployment controler creates 2 usefull features:

- The first one is that the pipeline can be run in parallel, as the steps that do not depend on each other can be run at the same time. This will make the pipeline faster and more efficient.

In order to select the methods and tools that will be used in the pipeline, a deep review of the literature will be performed. This review will explore recent publications of metabolomics data and also comparative reviews of the most used methods and tools for the pretreatement of targeted metabolomic data to select the most used methods for metabolomic data pretreatement. This review will also explore and select datasets to validate and optimize the performance of the pipeline.

4.2 Planning and calendar

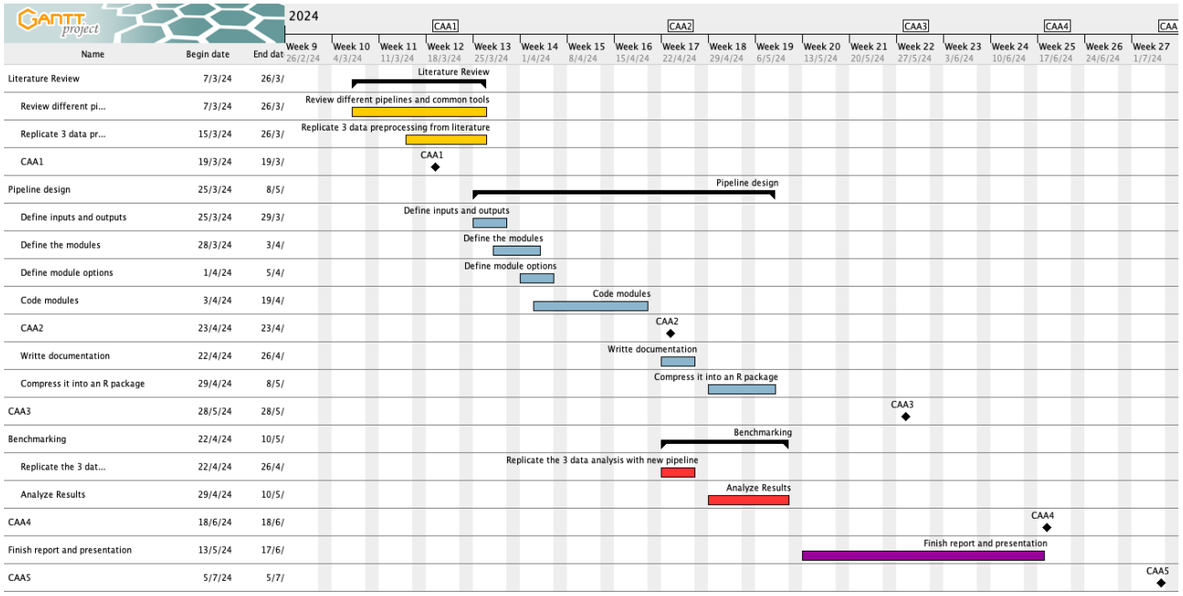

Figure 4.1: Gantt chart showing the project timeline and milestones.

4.2.1 Tasks

Main Tasks and prioritization

1. **Definition of the work plan:** Define the project's scope, objectives, methodology and expected outcomes. Create a project charter outlining the project's purpose and goals as well as a calendar with milestones and dates.

2. **Literature review:** Review multiple publications and replicate the data processing from 3 of them. Review most used methods and tools for metabolomics data processing like normalization, scaling, filtering, transformation, batch effect
3. **Pipeline design:** Propose and code in R a new pipeline to manage the processing of targeted metabolomics data.
4. **Benchmarking:** Replicate again the data processing of the 3 publications using the new pipeline and compare the results.

Extra tasks

1. **Heavy-workload:** Optimize the pipeline to enable parallelization of the processing.
2. **Documentation:** Write a documentation for the package so its accessibility.
3. **R package implementation:** Pack the code for the pipeline in an R package to easy distribution.

4.3 Risk analysis

Risk	Severity	Likelihood	Mitigation
Resource constraints	Moderate	Moderate	Develop a clear project timeline, incorporating milestones and allocating adequate time for each phase. Ensure contingency measures are in place to address unforeseen challenges or changes.
Technical challenges	Moderate	High	Perform proper exploration of packages and software and seek guidance and mentorship from professors or experts in relevant fields.
User adoption and awareness	High	Moderate	Be sure to incorporate appropriate cautions regarding the correct application of the chosen modules and data.
Pipeline branching	Low	Moderate	Adopt new methods to interactively select the branching

Table 4.1: Risk analysis. This table presents various risks associated with the project, along with their severity, likelihood, and potential mitigation measures.

4.4 Final products

- A **pipeline** for targeted metabolomic data preprocessing designed to streamline data pretreatment tasks, optimize workflows, and ensure reproducibility in metabolomic research endeavors.
- A **R package** crafted to facilitate the seamless integration and modular implementation of the pipeline within the R environment, empowering researchers with flexible and efficient tools for metabolomic data analysis.
- A detailed **documentation** accompanying the pipeline and R package.
- A user-friendly **Shiny app** that enables researchers with varying levels of computational expertise to effortlessly pretreat targeted metabolomic data.
- A **PDF report** detailing the project's processes, including investigation, development, results, conclusions, and discussions.
- A **virtual presentation** providing a comprehensive overview of the project. This includes a video recording with explanatory narration to further enhance understanding.

5. Materials and methods

Metodes: Com Desenvolupo // Materials: Amb que ho fas

5.1 Literature review

The literature selected for the review was obtained from multiple journals. The search was conducted using the following keywords: "metabolomics", "data preprocessing", "data pretreatment", "metabolomics pipeline", "metabolomics tools", "metabolomics R packages", "targeted metabolomics", "nutrimetabolomics", "metabolomics proce". The search was limited to articles published in the last 20 years, with a focus on metabolomics data preprocessing and pretreatment methodologies. The review aimed to identify the most commonly used methods and tools for metabolomic data preprocessing and pretreatment, as well as to explore recent advancements in the field. The review also sought to identify gaps in the existing literature and to inform the development of the new pipeline.

Incloure taula amb els articles revisats?

5.2 Pipeline design

The methods selected to be included in the pipeline were based on the results of the literature review. The pipeline was designed to be modular, allowing users to select and apply specific pretreatment steps according to the data caracteristics and user needs and preferences.

5.3 Datasets

The datasets employed for the purpose of this study were obtained from public repositories. The datasets were selected based on the following criteria:

- Aim of the dataset
- Availability of raw data
- Targeted metabolomics data
- Diverse biological samples

The datasets selected were 2:

- **MTBLS79:** [31] This dataset represents a systematic evaluation of the reproducibility of a multi-batch direct-infusion mass spectrometry (DIMS)-based metabolomics study of cardiac tissue extracts. It comprises twenty biological samples (cow vs. sheep) that were analysed repeatedly, in 8 batches across 7 days, together with a concurrent set of quality control (QC) samples. Data are presented from each step of the data processing

workflow and are available through MetaboLights. This dataset was selected due to its importance in the field of metabolomics and its data aviability.

- **ST000284:** [32] This dataset from MetaboWorkbench includes a study on colorectal cancer (CRC) using targeted liquid chromatography-tandem mass spectrometry. It examines 158 metabolites across 25 pathways in 234 serum samples (66 CRC patients, 76 polyp patients, 92 healthy controls). Blood samples were collected after fasting and bowel preparation. This dataset was selected due to its data aviability, its diversity (3 groups) and amount of samples.

5.4 Packages

The pipeline was developed using the R programming language [33] and the packages described in 5.1.

Also a brief description of the most relevant ones

add a summary of 1: Number of functions and 2: Number of lines of code

Package	Version	Ref
arrow	*15.0.1*	[34]
base	*4.3.3*	[33]
BiocStyle	*2.30.0*	[35]
caret	*6.0.94*	[36]
cowplot	*1.1.3*	[37]
crew	*0.9.2*	[38]
datasets	*4.3.3*	[33]
dplyr	*1.1.4*	[39]
DT	*0.33*	[40]
fst	*0.9.8*	[41]
ggforce	*0.4.2*	[42]
graphics	*4.3.3*	[43]
grDevices	*4.3.3*	[33]
HotellingEllipse	*1.1.0*	[44]
impute	*1.76.0*	[45]
imputeLCMD	*2.1*	[46]
knitr	*1.46*	[47]
MetaboAnalystR	*4.0.0*	[48]
methods	*4.3.3*	[33]
missForest	*1.5*	[49]
pcaMethods	*1.94.0*	[50]
pmp	*1.14.1*	[51]
purrr	*1.0.2*	[52]
renv	*1.0.7*	[53]
reshape2	*1.4.4*	[54]
rmarkdown	*2.27*	[55]
shiny	*1.8.1.1*	[56]
shinyFiles	*0.9.3*	[57]
stats	*4.3.3*	[33]
structToolbox	*1.14.0*	[58]
SummarizedExperiment	*1.32.0*	[28]
tarchetypes	*0.9.0*	[59]
targets	*1.7.0*	[60]
tidyverse	*2.0.0*	[61]
tinytex	*0.51*	[62]
tools	*4.3.3*	[33]
usethis	*2.2.3*	[63]
utils	*4.3.3*	[33]
VIM	*6.2.2*	[64]
withr	*3.0.0*	[65]

Table 5.1: List of R packages with their versions used to develop `metaboPipe`

6. Results

In this project, an R package named `metaboPipe` was developed to preprocess metabolomic data efficiently and accessibly. Metabolomic data preprocessing is a crucial step in ensuring the quality and reliability of downstream analyses. The transformations applied during preprocessing are primarily intended to remove unwanted effects unrelated to the study, but these tools can inadvertently delete meaningful biological data if used without caution. The `metaboPipe` package leverages the `targets` and `structToolbox` packages to create a modular and reproducible pipeline for preprocessing. This section details the components and functionality of `metaboPipe`.

6.1 Package Overview

The `metaboPipe` package is designed to streamline the preprocessing of metabolomic data by creating a series of user-accessible modules that translate into interconnected steps. These steps can be easily added, modified, and rearranged. The package utilizes the `targets` package to orchestrate task deployment and dependencies, ensuring that each step is executed in the correct order. The `structToolbox` package provides the `DatasetExperiment` object, which serves as the primary data structure throughout the pipeline, along with multiple functions for data transformation.

6.2 Functions and Processing Steps

Although the `metaboPipe` package comprises over 56 functions, there are six main preprocessing functions that users will primarily engage with, each responsible for a specific data transformation step. These functions are designed to operate sequentially on a `DatasetExperiment` object, transforming it at each stage to prepare it for subsequent steps.

6.2.1 Data Format

Data must be imported in `.csv` format, consisting of two types of files:

Matrix Data

The matrix data is a table where each row represents a unique sample and each column represents a metabolite. The first row must contain the metabolite names. This table has the concentration data, peak intensity tables or MS/NMR spectral bins

Sample Metadata

The sample metadata is a table where each row represents a unique sample and each column represents a variable. The samples must be in the same order as the matrix data. The first row must contain the variable names, and there must be a column named `sample_id` specifying a unique identifier for each sample. The number of variables depends on the study and preprocessing needs.

Feature Metadata

The feature metadata is an optional table where each row represents a feature and each column information related to the feature such as the chemical formula, retention time, and mass-to-charge ratio. The first row contains the column names and the first column needs to be called `annotation` containing the feature names. This table is optional as if not provided the pipeline will create a minimum table with only the annotation column using those provided in the matrix data.

6.2.2 Data loading

The data loading function takes the data matrix, sample metadata and optionally a feature metadata files and creates a `DatasetExperiment` object providing a structured representation of the metabolomic dataset. Loading the data is the first step in the preprocessing pipeline, enabling the subsequent transformations to be applied to the data.

6.2.3 Filtering

Filtering is usually the first step in the pipeline. It removes irrelevant or low-quality features from the dataset based on user-defined criteria. This step is essential to reduce noise and enhance the accuracy of downstream analyses. The module generates target nodes that apply the filtering criteria to the `DatasetExperiment` object, producing a refined dataset. Those criteria are the minimum percentage of samples and metabolites with non-zero values and the removing of outliers based on the Hotelling's T2 distribution ellipse.

6.2.4 Imputation

Following filtering, the imputation function addresses missing values in the dataset. Missing data can significantly affect the results of metabolomic analyses; hence, accurate imputation is critical. The imputation function offers various methods for estimating missing values, such as mean substitution, k-nearest neighbors, and multiple imputation, and updates the `DatasetExperiment` object accordingly.

6.2.5 Batch Correction

Batch effects are common in metabolomic studies and can confound results if not properly corrected. The batch correction function in `metaboPipe` identifies and adjusts for batch effects, ensuring that the data is comparable across different experimental batches. This step

generates target nodes that modify the `DatasetExperiment` object to remove batch-related variability.

6.2.6 Normalization

Normalization is performed to account for differences in sample concentration and ensure that the metabolite intensities are comparable across samples. The normalization function offers several methods, including total area normalization, probabilistic quotient normalization, and variance stabilization. The normalized `DatasetExperiment` object is then passed to the next processing step.

6.2.7 Scaling

Scaling is used to adjust the range and distribution of metabolite intensities. The scaling function in `metaboPipe` provides options such as standard scaling (z-score), min-max scaling, and Pareto scaling. This step standardizes the data, facilitating meaningful comparisons between metabolites. The scaled `DatasetExperiment` object is prepared for the final transformation step.

6.2.8 Transformation

The transformation function applies mathematical transformations to stabilize the variance and make the data more normally distributed. Common transformations include logarithmic, square root, and Box-Cox transformations. The transformed `DatasetExperiment` object is the final output of the preprocessing pipeline, ready for downstream analysis.

6.3 Implementation Details

Each preprocessing step in `metaboPipe` is implemented as a function that creates target nodes using the `targets` package. These nodes encapsulate the necessary operations and dependencies, ensuring that the pipeline is both modular and reproducible.

The `DatasetExperiment` object from the `structToolbox` package is used throughout the pipeline, providing a consistent and flexible data structure for all preprocessing steps.

The `metaboPipe` package offers a comprehensive and modular approach to preprocessing metabolomic data. By integrating the `targets` and `structToolbox` packages, `metaboPipe` ensures that each preprocessing step is executed efficiently and reproducibly. This pipeline enhances the quality of metabolomic data, facilitating robust and reliable downstream analyses. Future work may involve extending the pipeline to include additional preprocessing steps or integrating advanced machine learning techniques for enhanced data processing.

6.4 Documentation

6.5 Shiny app

7. Discussion

8. Conclusion and future vision

The second one is that it enables the pipeline to be used as a pretreatement method comparable tool. As the easy of use and order can be easily changed, the user can test different pretreatement methods and orders to compare the changes in the data.

Este capítulo tiene que incluir:

- Una descripción de las conclusiones del trabajo:
  - Una vez se han obtenido los resultados, ¿qué conclusiones se extraen?
  - ¿Estos resultados son los esperados? ¿O han sido sorprendentes? ¿Por qué?
- Una reflexión crítica sobre el logro de los objetivos planteados inicialmente:
  - ¿Hemos logrado todos los objetivos? Si la respuesta es negativa, ¿por qué motivo?
- Un análisis crítico del seguimiento de la planificación y metodología a lo largo del producto:
  - ¿Se ha seguido la planificación?
  - ¿La metodología prevista ha sido suficientemente adecuada?
  - ¿Ha habido que introducir cambios para garantizar el éxito del trabajo? ¿Por qué?
- De los impactos previstos en 3, ético-sociales, de sostenibilidad y de diversidad, evaluar/mencionar si se han mitigado (si eran negativos) o si se han conseguido (si eran positivos).
- Si han aparecido impactos no previstos a 3, evaluar/mencionar cómo se han mitigado (si eran negativos) o que han aportado (si eran positivos).
- Las líneas de trabajo futuro que no se han podido explorar en este trabajo y han quedado pendientes.
- Glossary test: LaTeX

Glossary

LaTeX A typesetting system used for document preparation. 27

Acronyms

ANOVA Analysis of Variance. 12

GC Gas Cromatography. 9, 10

GC-MS Gas Cromatography-Mass Spectrometry. 9, 10

LC Liquid Cromatography. 9, 10

LC-MS Liquid Cromatography-Mass Spectrometry. 9, 10

m/z Mass-to-Charge Ratio. 10

MS Mass Spectrometry. 9–11, 13

NMR Nuclear Magnetic Resonance. 9–11, 13

PCA Principal Component Analysis. 12

PCPr2 Principal Component Partial R-square. 12

RT Retention Time. 10

SNR Signal-to-Noise Ratio. 9

Bibliography

1. Patti, G. J., Yanes, O. & Siuzdak, G. Metabolomics: The Apogee of the Omics Trilogy. *Nature Reviews Molecular Cell Biology* **13,** 263–269. ISSN: 1471-0080 (Apr. 2012).
2. Zhang, A., Sun, H. & Wang, X. Serum Metabolomics as a Novel Diagnostic Approach for Disease: A Systematic Review. *Analytical and Bioanalytical Chemistry* **404,** 1239–1245. ISSN: 1618-2650 (Sept. 1, 2012).
3. Chen, Y., Li, E.-M. & Xu, L.-Y. Guide to Metabolomics Analysis: A Bioinformatics Workflow. *Metabolites* **12,** 357. ISSN: 2218-1989 (4 Apr. 2022).
4. Sun, J. & Xia, Y. Pretreating and Normalizing Metabolomics Data for Statistical Analysis. *Genes & Diseases* **11,** 100979. ISSN: 2352-3042 (May 1, 2024).
5. Martin, M. *et al.* PepsNMR for 1H NMR Metabolomic Data Pre-Processing. *Analytica Chimica Acta* **1019,** 1–13. ISSN: 1873-4324. pmid: 29625674 (Aug. 17, 2018).
6. Karaman, I. in *Metabolomics: From Fundamentals to Clinical Applications* (ed Sussulini, A.) 145–161 (Springer International Publishing, Cham, 2017). ISBN: 978-3-319-47656-8.
7. Bhinderwala, F., Roth, H., Noel, H., Feng, D. & Powers, R. Chemical Shift Variations in Common Metabolites. *Journal of magnetic resonance (San Diego, Calif. : 1997)* **345,** 107335. ISSN: 1090-7807. pmid: 36410060 (Dec. 2022).
8. Stanstrup, J. *et al.* The metaRbolomics Toolbox in Bioconductor and Beyond. *Metabolites* **9,** 200. ISSN: 2218-1989 (10 Oct. 2019).
9. Qiu, S. *et al.* Small Molecule Metabolites: Discovery of Biomarkers and Therapeutic Targets. *Signal Transduction and Targeted Therapy* **8,** 1–37. ISSN: 2059-3635 (Mar. 20, 2023).
10. Xiao, J. F., Zhou, B. & Ressom, H. W. Metabolite Identification and Quantitation in LC-MS/MS-based Metabolomics. *Trends in analytical chemistry : TRAC* **32,** 1–14. ISSN: 0165-9936. pmid: 22345829 (Feb. 1, 2012).
11. Kiseleva, O., Kurbatov, I., Ilgisonis, E. & Poverennaya, E. Defining Blood Plasma and Serum Metabolome by GC-MS. *Metabolites* **12,** 15. ISSN: 2218-1989. pmid: 35050137 (Dec. 24, 2021).
12. Defernez, M. & Le Gall, G. in *Advances in Botanical Research* (ed Rolin, D.) 493–555 (Academic Press, Jan. 1, 2013).
13. Troisi, J., Troisi, G., Scala, G. & Richards, S. M. in *Metabolomics Perspectives* (ed Troisi, J.) 287–379 (Academic Press, Jan. 1, 2022). ISBN: 978-0-323-85062-9.
14. Burton, L. *et al.* Instrumental and Experimental Effects in LC–MS-based Metabolomics. *Journal of Chromatography B. Hyphenated Techniques for Global Metabolite Profiling* **871,** 227–235. ISSN: 1570-0232 (Aug. 15, 2008).

15. Trygg, J., Gabrielsson, J. & Lundstedt, T. in *Comprehensive Chemometrics* (eds Brown, S. D., Tauler, R. & Walczak, B.) 1–8 (Elsevier, Oxford, Jan. 1, 2009). ISBN: 978-0-444-52701-1.

16. Alonso, A., Marsal, S. & Julià, A. Analytical Methods in Untargeted Metabolomics: State of the Art in 2015. *Frontiers in Bioengineering and Biotechnology* **3.** ISSN: 2296-4185 (Mar. 5, 2015).

17. Bloemberg, T. G., Gerretzen, J., Lunshof, A., Wehrens, R. & Buydens, L. M. C. Warping Methods for Spectroscopic and Chromatographic Signal Alignment: A Tutorial. *Analytica Chimica Acta* **781,** 14–32. ISSN: 0003-2670 (June 5, 2013).

18. Bork, C., Ng, K., Liu, Y., Yee, A. & Pohlscheidt, M. Chromatographic Peak Alignment Using Derivative Dynamic Time Warping. *Biotechnology Progress* **29,** 394–402. ISSN: 1520-6033 (2013).

19. Veselkov, K. A. *et al.* Recursive Segment-Wise Peak Alignment of Biological (1)h NMR Spectra for Improved Metabolic Biomarker Recovery. *Analytical Chemistry* **81,** 56–66. ISSN: 1520-6882. pmid: 19049366 (Jan. 1, 2009).

20. Sawall, M. *et al.* Multi-Objective Optimization for an Automated and Simultaneous Phase and Baseline Correction of NMR Spectral Data. *Journal of Magnetic Resonance* **289,** 132–141. ISSN: 1090-7807 (Apr. 1, 2018).

21. Martín-Fernández, J. A., Palarea-Albaladejo, J. & Olea, R. A. in *Compositional Data Analysis* 43–58 (John Wiley & Sons, Ltd, 2011). ISBN: 978-1-119-97646-2.

22. Zacharias, H. U., Altenbuchinger, M. & Gronwald, W. Statistical Analysis of NMR Metabolic Fingerprints: Established Methods and Recent Advances. *Metabolites* **8,** 47. ISSN: 2218-1989 (3 Sept. 2018).

23. Ulaszewska, M. M. *et al.* Nutrimetabolomics: An Integrative Action for Metabolomic Analyses in Human Nutritional Studies. *Molecular Nutrition & Food Research* **63,** 1800384. ISSN: 1613-4133 (2019).

24. Smith, C. A. *et al. xcms: LC-MS and GC-MS Data Analysis* R package version 4.0.2 (2024). https://bioconductor.org/packages/xcms.

25. Schmid, R. *et al.* Integrative Analysis of Multimodal Mass Spectrometry Data in MZmine 3. *Nature Biotechnology* **41,** 447–449. ISSN: 1546-1696 (Apr. 2023).

26. Keyes, O. *et al. batman: Convert Categorical Representations of Logicals to Actual Logicals* R package version 0.1.0 (2015). https://CRAN.R-project.org/package=batman.

27. Gu, H., Gowda, G. A. N., Neto, F. C., Opp, M. R. & Raftery, D. RAMSY: Ratio Analysis of Mass Spectrometry to Improve Compound Identification. *Analytical Chemistry* **85,** 10771–10779. ISSN: 0003-2700 (Nov. 19, 2013).

28. Morgan, M., Obenchain, V., Hester, J. & Pagès, H. *SummarizedExperiment: Summarized-Experiment container* R package version 1.32.0 (2023). https://bioconductor.org/packages/SummarizedExperiment.

29. Lloyd, G. R., Jankevics, A. & Weber, R. J. M. struct: an R/Bioconductor-based framework for standardized metabolomics data analysis and beyond. *Bioinformatics* **36,** 5551–5552. https://doi.org/10.1093/bioinformatics/btaa1031 (2020).

30. Landau, W. M. The targets R package: a dynamic Make-like function-oriented pipeline toolkit for reproducibility and high-performance computing. *Journal of Open Source Software* **6,** 2959. https://doi.org/10.21105/joss.02959 (2021).

31. Kirwan, J. A., Weber, R. J. M., Broadhurst, D. I. & Viant, M. R. Direct Infusion Mass Spectrometry Metabolomics Dataset: A Benchmark for Data Processing and Quality Control. *Scientific Data* **1,** 140012. ISSN: 2052-4463 (June 10, 2014).

32. Zhu, J. *et al.* Colorectal Cancer Detection Using Targeted Serum Metabolic Profiling. *Journal of Proteome Research* **13,** 4120–4130. ISSN: 1535-3907. pmid: 25126899 (Sept. 5, 2014).

33. R Core Team. *R: A Language and Environment for Statistical Computing* R Foundation for Statistical Computing (Vienna, Austria, 2024). https://www.R-project.org/.

34. Richardson, N. *et al. arrow: Integration to Apache 'Arrow'* R package version 15.0.1 (2024). https://github.com/apache/arrow/.

35. Oleś, A. *BiocStyle: Standard styles for vignettes and other Bioconductor documents* R package version 2.30.0 (2023). https://bioconductor.org/packages/BiocStyle.

36. Kuhn, M. *caret: Classification and Regression Training* R package version 6.0-94 (2023). https://github.com/topepo/caret/.

37. Wilke, C. O. *cowplot: Streamlined Plot Theme and Plot Annotations for ggplot2* R package version 1.1.3 (2024). https://wilkelab.org/cowplot/.

38. Landau, W. M. *crew: A Distributed Worker Launcher Framework* R package version 0.9.2 (2024). https://wlandau.github.io/crew/.

39. Wickham, H., François, R., Henry, L., Müller, K. & Vaughan, D. *dplyr: A Grammar of Data Manipulation* R package version 1.1.4 (2023). https://dplyr.tidyverse.org.

40. Xie, Y., Cheng, J. & Tan, X. *DT: A Wrapper of the JavaScript Library DataTables* R package version 0.33 (2024). https://github.com/rstudio/DT.

41. Klik, M. *fst: Lightning Fast Serialization of Data Frames* R package version 0.9.8 (2022). http://www.fstpackage.org.

42. Pedersen, T. L. *ggforce: Accelerating ggplot2* R package version 0.4.2, https://github.com/thomasp85/ggforce (2024). https://ggforce.data-imaginist.com.

43. Gentleman, R., Whalen, E., Huber, W. & Falcon, S. *graph: A package to handle graph data structures* R package version 1.80.0 (2023). https://bioconductor.org/packages/graph.

44. Goueguel, C. L. *HotellingEllipse: Hotelling T-Square and Confidence Ellipse* R package version 1.1.0 (2022). https://github.com/ChristianGoueguel/HotellingEllipse.

45. Hastie, T., Tibshirani, R., Narasimhan, B. & Chu, G. *impute: Imputation for microarray data* R package version 1.76.0 (2023). https://bioconductor.org/packages/impute.

46. Lazar, C. & Burger, T. *imputeLCMD: A Collection of Methods for Left-Censored Missing Data Imputation* R package version 2.1 (2022). https://CRAN.R-project.org/package=imputeLCMD.

47. Xie, Y. *knitr: A General-Purpose Package for Dynamic Report Generation in R* R package version 1.46 (2024). https://yihui.org/knitr/.

48. Xia, J., Chong, J. & Pang, Z. *MetaboAnalystR: An R Package for Comprehensive Analysis of Metabolomics Data* R package version 4.0.0 (2024).
49. Stekhoven, D. J. *missForest: Nonparametric Missing Value Imputation using Random Forest* R package version 1.5 (2022). https://www.r-project.org.
50. Stacklies, W., Redestig, H. & Wright, K. *pcaMethods: A collection of PCA methods* R package version 1.94.0 (2023). https://bioconductor.org/packages/pcaMethods.
51. Jankevics, A., Lloyd, G. R. & Weber, R. J. M. *pmp: Peak Matrix Processing and signal batch correction for metabolomics datasets* R package version 1.14.1 (2024).
52. Wickham, H. & Henry, L. *purrr: Functional Programming Tools* R package version 1.0.2 (2023). https://purrr.tidyverse.org/.
53. Ushey, K. & Wickham, H. *renv: Project Environments* R package version 1.0.5 (2024). https://rstudio.github.io/renv/.
54. Wickham, H. *reshape2: Flexibly Reshape Data: A Reboot of the Reshape Package* R package version 1.4.4 (2020). https://github.com/hadley/reshape.
55. Allaire, J. *et al. rmarkdown: Dynamic Documents for R* R package version 2.26, https://pkgs.rstudio.com/rmarkdown/ (2024). https://github.com/rstudio/rmarkdown.
56. Chang, W. *et al. shiny: Web Application Framework for R* R package version 1.8.1.1 (2024). https://shiny.posit.co/.
57. Pedersen, T. L., Nijs, V., Schaffner, T. & Nantz, E. *shinyFiles: A Server-Side File System Viewer for Shiny* R package version 0.9.3 (2022). https://github.com/thomasp85/shinyFiles.
58. Lloyd, G. R. & Weber, R. J. M. *structToolbox: Data processing & analysis tools for Metabolomics and other omics* R package version 1.14.0 (2023). https://bioconductor.org/packages/structToolbox.
59. Landau, W. M. *tarchetypes: Archetypes for Targets* R package version 0.9.0, https://github.com/ropensci/tarchetypes (2024). https://docs.ropensci.org/tarchetypes/.
60. Landau, W. M. *targets: Dynamic Function-Oriented Make-Like Declarative Pipelines* R package version 1.6.0, https://github.com/ropensci/targets (2024). https://docs.ropensci.org/targets/.
61. Wickham, H. *tidyverse: Easily Install and Load the Tidyverse* R package version 2.0.0, https://github.com/tidyverse/tidyverse (2023). https://tidyverse.tidyverse.org.
62. Xie, Y. *tinytex: Helper Functions to Install and Maintain TeX Live, and Compile LaTeX Documents* R package version 0.51 (2024). https://github.com/rstudio/tinytex.
63. Wickham, H., Bryan, J., Barrett, M. & Teucher, A. *usethis: Automate Package and Project Setup* R package version 2.2.3 (2024). https://usethis.r-lib.org.
64. Templ, M., Kowarik, A., Alfons, A., de Cillia, G. & Rannetbauer, W. *VIM: Visualization and Imputation of Missing Values* R package version 6.2.2 (2022). https://github.com/statistikat/VIM.
65. Hester, J. *et al. withr: Run Code With Temporarily Modified Global State* R package version 3.0.0 (2024). https://withr.r-lib.org.

Listado de apartados que son demasiado extensos para incluir dentro de la memoria y tienen un carácter autocontenido (por ejemplo, manuales de usuario, manuales de instalación, etc.)

Dependiendo del tipo de trabajo, es posible que no haya que añadir algún anexo.