MicrobiotaProcess: A comprehensive R package for managing and analyzing microbiome and other ecological data within the tidy framework

Shuangbin Xu

Southern Medical University <https://orcid.org/0000-0003-3513-5362>

Zhan Li

Southern Medical University

Wenli Tang

Southern Medical University

Zehan Dai

Southern Medical University

Lang Zhou

Southern Medical University

Tingze Feng

Southern Medical University

Meijun Chen

Southern Medical University

Shanshan Liu

Southern Medical University

Xiaocong Fu

Southern Medical University

Tianzhi Wu

Southern Medical University

Erqiang Hu

Southern Medical University

Guangchuang Yu (✉ gcyu1@smu.edu.cn)

Southern Medical University <https://orcid.org/0000-0002-6485-8781>

Methods Brief Communication

Keywords:

Posted Date: February 4th, 2022

DOI: <https://doi.org/10.21203/rs.3.rs-1284357/v1>

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
[Read Full License](#)

***MicrobiotaProcess*: A comprehensive R package for managing and analyzing microbiome and other ecological data within the tidy framework**

Shuangbin Xu¹, Li Zhan¹, Wenli Tang¹, Zehan Dai¹, Lang Zhou^{1,2}, Tingze Feng¹, Meijun Chen¹, Shanshan Liu¹, Xiaocong Fu¹, Tianzhi Wu¹, Erqiang Hu¹ and Guangchuang Yu^{1,2,*}

¹ Department of Bioinformatics, School of Basic Medical Sciences, Southern Medical University, Guangzhou 510515, China

² Division of Laboratory Medicine, Microbiome Medicine Center, Zhujiang Hospital, Southern Medical University, Guangzhou, China

* Correspondence: Guangchuang Yu <gcyul@smu.edu.cn>

Abstract

The data output from microbiome research is growing at an accelerating rate, yet mining the data quickly and efficiently remains difficult. There is still a lack of effective data structure to represent and manage data, as well as flexible and composable analysis methods. In response to these two issues, we designed and developed the *MicrobiotaProcess* package. It provides a comprehensive data structure, *MPSE*, to better integrate the primary and intermediate data, which improves the integration and exploration of the downstream data. Around this data structure, the downstream analysis tasks are decomposed and a set of functions are designed under the tidy framework. These functions independently perform simple tasks and can be combined to perform complex tasks. This gives users the ability to explore data, conduct personalized analysis and develop analysis workflows. Moreover, *MicrobiotaProcess* can interoperate with other packages in the R community, which further expands its analytical capabilities. This article demonstrates the *MicrobiotaProcess* for analyzing microbiome data as well as other ecological data through several examples. It connects upstream data, provides flexible downstream analysis components, and provides visualization methods to assist in presenting and interpreting results.

Introduction

A wide array of important roles of the microbiota in diverse environments have been investigated and explored substantially^{1,2}, largely due to the development of high-throughput sequencing technologies and bioinformatics. During the last decades, many bioinformatics algorithms and tools for the exploration and the analysis of microbiome data have been built in the scientific community, such as *qiime2*³, *dada2*⁴, *usearch*⁵, *mothur*⁶, *MetaPhlan*^{7,8}. These pipelines or tools conduce initial bioinformatics analysis of microbiome data, but the relevant data to a microbiome experiment became heterogeneous consisting of feature-oriented (operational taxonomic unit or exact sequence variants) data frames, sample-oriented data frames, the representative sequences, and the phylogenetic tree of the sequences after the initial analysis, which brings new challenges for the downstream data analysis and reproducibility. There are often many important links (such as the features of the feature table and the nodes of the phylogenetic tree, feature table and metadata, *etc.*) among the diverse data that need to be preserved throughout an analysis. If a researcher failed to integrate these data comprehensively, the information might be lost and thus generate errors. In

addition, many intermediate steps that may confuse may be repeatedly performed in the downstream statistical analysis. For instance, the dissimilarity indices (such as *Bray-Curtis*, *(Un)Weighted Unifrac* and *Jaccard*, etc.) for the communities of microbes might need to be calculated and reused in hierarchical cluster analysis, *PCoA* (Principal Coordinate Analysis), and *PERMANOVA* (Permutational multivariate analysis of variance), etc. If the intermediate data can be effectively integrated and stored, it will improve the efficiency of analysis, enhance reproducibility and avoid errors⁹. The R programming language has become one of the most popular tools for biomedical data analysis¹⁰. Some efforts have been made to build common representations and infrastructures for complex, highly interdependent data sets in R^{10,11}. For example, *SummarizedExperiment*¹² class is widely used to integrate matrix-like objects of feature abundance, the normalized feature abundance, a sample- and feature-oriented metadata data frames as a standardized data structure across many *Bioconductor*¹³ packages. To integrate the phylogenetic tree structure and heterogeneous associated data, we defined the *treedata*^{14,15} class, which has also been widely used in several packages, such as *tidytree*¹⁴, *treeio*¹⁵, and *ggtree*¹⁶. However, these data structures did not cover all the heterogeneous data of a microbiome experiment, and the existing tools¹⁷⁻²² for the downstream statistical analysis of microbiome also did not well integrate the primary heterogeneous data and the useful intermediate data. For instance, the *phyloseq*¹⁷ class is used in the *phyloseq*¹⁷, *microViz*¹⁹, and *MicrobiomeAnalyst*²² packages, but the data structure can only store the primary input datasets, it cannot integrate the normalized data and the intermediate data such as the alpha diversity, dissimilarity indices, and the result of differential analysis, etc. The *animalcules* package was developed using the *MultiAssayExperiment*²³ class, and it cannot integrate intermediate data and the phylogenetic tree which is often needed in the calculation of specified dissimilarity ((Un)Weighted UniFrac). These data structures for the microbiome cannot take into account the diversified needs of downstream analysis, which makes it more convenient for some specific needs, while other needs may be troublesome. Moreover, the differences in these data structures are too large, which restricts downstream integrated analysis.

Downstream statistical, visual, and functional analyses of microbiome data are complex since the appropriate analysis workflow often needs to be explored and adjusted according to the research design. Many methods and parameters, such as the methods of dissimilarity, the clustering, and the ordination, must be considered properly during analysis. A simple, unified, and human-readable analysis workflow will improve the efficiency of data analysis, avoid errors, and enable researchers to focus more on biological problems⁹. Recently, a concise data structure (tidy data)²³ has become popular in the R¹⁰ data analysis community with tidy data tools, such as *dplyr*²⁴, *tidyr*²⁴, and *ggplot2*²⁵. This tidy concept has been applied to different disciplines, including genomic, transcriptomic, and phylogenetic data analyses. The *plyranges*²⁶ package provided a *dplyr*-like interface for interacting with some of the most common data structures containing genomic coordinates including *Ranges* and *GenomicRanges*²⁷. The *tidybulk*²⁸ package introduced a tidy transcriptomic data structure paradigm and analysis grammar for the bulk transcriptional analyses to bridge transcriptional data analysis with the *tidyverse* ecosystem²⁴. The *clusterProfiler*²⁹ package developed by us introduced a tidy interface to facilitate users to operate and visualize enrichment results. We also developed the *ggtree*¹⁶ and *ggtreeExtra*³⁰ packages to provide a general framework for the annotation and visualization of phylogenetic trees with heterogeneous associated data based on a concise data structure, the *treedata*^{14,15} class. Through the human-readable data structure and

analysis grammar, these resources make it easier for the community to develop modular manipulation, visualization, and analysis methods, decrease the learning curve for users, and facilitate reproducibility for the related studies^{9,23,24}. However, this principle of tidiness has not been implemented in the existing tools^{17–22} for microbiome data analysis. This greatly hinders the flexibility and ease of use of downstream data analysis in this discipline.

To fill these gaps, we developed the *MicrobiotaProcess* package. We defined the *MPSE* class for storing the microbiome experiment or related ecological dataset and the related intermediate data of downstream analysis. The *MPSE* class defined in *MicrobiotaProcess* inherits both the *SummarizedExperiment*¹² class and the *treedata*^{14,15} class, which are popular and widely used in the *Bioconductor*¹³ ecosystem. It takes merits from *Bioconductor* packages that are based on these two data structures. The importance of data structure is that downstream analysis is based on it. A good data structure helps unify downstream analysis. When operating a certain data, the associated data in the data structure can be updated synchronously, which can reduce many errors caused by improper operations. In order to make downstream data analysis modular and easy to use, we introduce a user-friendly grammar (i.e., tidy interface) to process, analyze and visualize microbiome data stored in the *MPSE* data structure. This framework allows users to build human-readable and flexible analysis workflows. We believe it can remove a major obstacle for scientists to explore and analyze the microbiome and other ecological data.

Results

A container improving the exploration of the downstream data of microbiome

MicrobiotaProcess introduces the *MPSE* class, which inherits the *SummarizedExperiment*¹², *treedata*^{14,15}, and *XStringSet*³¹ classes, for storing microbiome or other related assay data, metadata, and phylogenetic tree data (**Fig.1A**). The primary matrices data, such as count matrices and standardized matrices, are stored in the *assays*, where rows represent features (such as ASVs/OTUs and genes, etc.) and columns represent samples. The samples characteristics are stored in the *colData*. To store the phylogenetic tree and associated data, we defined the *otutree* slot, which inherits a *treedata*^{14,15} class defined in *tidytree*¹⁴ and *treeio*¹⁵. The phylogenetic tree file that contained the evolutionary statistics inferences and other associated data can be parsed using *treeio*¹⁵ and stored in the *otutree* slot. We also defined a *taxatree* slot which is also a *treedata*^{14,15} class to store hierarchical relationships among the taxa and their lineages. The *taxatree* and *otutree* components can be processed and visualized directly via the in-house developed *ggtree* package suite (*tidytree*¹⁴, *treeio*¹⁵, *ggtree*¹⁶, and *ggtreeExtra*³⁰). Moreover, the results of the downstream analysis can also be stored in the *MPSE* class, which is also different from existing related packages. For example, we used *mp_rarefy* to rarefy the primary matrices data and added the results into *assays* slot, then used *mp_cal_alpha* to calculate the alpha diversity of the samples. The results of alpha diversity were added into *colData* slot, which can be visualized using *mp_plot_alpha* further.

```
library(SummarizedExperiment)
```

```
library(MicrobiotaProcess)
```

```

data(mouse.time.mpse)
assays(mouse.time.mpse)

## List of length 1
## names(1): Abundance

mouse.time.mpse %<>% mp_rrarefy()
assays(mouse.time.mpse)

## List of length 2
## names(2): Abundance RareAbundance

# the alpha index will add into colData when action = 'add'
mouse.time.mpse %<>% mp_cal_alpha(.abundance = RareAbundance, action = "
add")
print(mouse.time.mpse, n=4)

## # A MPSE-tibble (MPSE object) abstraction: 4,142 × 18
## # OTU=218 | Samples=19 | Assays=Abundance, RareAbundance | Taxonomy=K
kingdom, Phylum, Class, Order, Family, Genus, Species
##   OTU   Sample Abundance RareAbundance time   Observe Chao1
##   <chr> <chr>      <int>         <int> <chr>   <dbl> <dbl>
## 1 OTU_1 F3D0        579           214 Early    104  104.
## 2 OTU_2 F3D0        345           116 Early    104  104.
## 3 OTU_3 F3D0        449           179 Early    104  104.
## 4 OTU_4 F3D0        430           167 Early    104  104.
## # ... with 4,138 more rows, and 11 more variables: ACE <dbl>,
## #   Shannon <dbl>, Simpson <dbl>, Pielou <dbl>,
## #   Kingdom <chr>, Phylum <chr>, Class <chr>, Order <chr>,
## #   Family <chr>, Genus <chr>, Species <chr>

```

Each component of the *MPSE* class can be extracted by the self-explanatory function names, prefix with ‘mp_extract_’ and followed by a specific explanatory term (see also *Accessors to fetch internal data* in the Methods session) (**Fig.1A**). In the following example, we used *mp_extract_assays* to extract the *assays* component and used *mp_extract_sample* to extract the *colData* component which contained sample metadata.

```

rare.tb <- mouse.time.mpse %>% mp_extract_assays(.abundance = RareAbunda
nce)
sample.da <- mouse.time.mpse %>% mp_extract_sample()

```

Through the *MPSE* class, all related data and results relevant to a microbiome experiment can be stored in a single instance, which enables to improve the exploration of the downstream data, facilitate data sharing and enhance reproducibility.

Bridging the upstream analysis tools and MPSE constructor

The output files of the upstream analysis of the microbiome are various and usually not human-friendly. How to parse the output files is often the first obstacle that researchers should face. The *phyloseq*¹⁷ had provided some functions to parse the output of common upstream analysis tools for 16s rRNA dataset, but not for metagenomics or the output of commonly used tools (*qiime2*²³, *dada2*²⁴). To address the need for multiple types of microbiome data (16s rRNA, metagenomics, and other related ecological data) as well as output obtained from commonly used tools, *MicrobiotaProcess* provides several functions to parse the output of the upstream analysis of microbiome (**Fig.2A**). To enhance the interoperability with other R packages, *MicrobiotaProcess* also provides *as.MPSE* to convert common *S4* objects defined in the *Bioconductor*¹³ ecosystem, such as *phyloseq*¹⁷, *biom*³², *SummarizedExperiment*¹², and *TreeSummarizedExperiment*¹⁸. All of the parse functions and the convert functions output the same object type, *MPSE*, making them consistent and robust for downstream manipulation. In addition, An *MPSE* class also can be constructed by a call of the function of the same name with required parameters (see also *parser functions and the MPSE constructor* in the Methods session). Here, we used *as.MPSE* to convert a *biom*³² object which is the output of *read_biom* of *biomformat*³² to an *MPSE* object, and used the *MPSE* function to build an *MPSE* object from scratch.

```
library(MicrobiotaProcess)
library(biomformat)
biom_file <- system.file("extdata", "rich_sparse_otu_table.biom", package = "biomformat")
xx <- read_biom(biom_file)
# convert to an MPSE object using as.MPSE function
mpse2 <- xx %>% as.MPSE()
```

```
library(MicrobiotaProcess)
library(vegan)
data(varespec, varechem)
# building an MPSE object using MPSE function
mpse1 <- MPSE(assays=list(Abundance=t(varespec)), colData=varechem)
```

Through these functions, *MicrobiotaProcess* can bridge the upstream analysis tools and address the need for downstream analysis based on the *MPSE* class (**Fig.2C1 and C2**). It supports multiple types of data and we will describe some case studies using *MicrobiotaProcess* to analyze 16s rRNA, metagenomics, KEGG gene datasets, and related ecological data in the following sessions.

The unified tidy framework analysis grammar

To facilitate data manipulation and exploration of the microbiome or other ecological data based on the *MPSE*, *MicrobiotaProcess* defined a tidy-like format output for the *MPSE* class (**Fig.1B**). The *assays* of *MPSE* are converted to a tidy format, in which three columns (feature identifiers, sample,

and feature abundance) are fixed, each row represents the abundance of each feature of each sample. Optional columns include the taxonomy annotation and sample metadata information or newly calculated information. *MicrobiotaProcess* provides a unified analysis grammar to express all the steps of a typical microbiome community analysis workflow using self-explanatory function names, which are composed of ‘*mp_*’, a specific verb, and one or two explanatory terms (**Fig.2 C1 and C2**). Moreover, *MicrobiotaProcess* extends the *dplyr*²⁴ verbs to support processing data stored in an *MPSE* object (**Fig.2B**). Following the concept of tidiness, these verbs provide robust and standardized operations for data analysis and transformation, and can be assembled into a workflow using the pipe operator (%>%). This enables users to effectively analyze ecological data such as metagenomic data and easily build reproducible and human-readable pipelines.

To illustrate the flexibility and versatility of *MicrobiotaProcess*, we reanalyzed three public datasets as examples. Each example addresses a different problem and has a different focus on demonstrating the features of *MicrobiotaProcess*. The details of the analysis and the corresponding code are presented in the Supplementary file.

Example 1: Characterization of the gut microbiota using 16s rRNA data

The 16s rRNA dataset of pediatric stool was obtained from the iHMP (Integrative Human Microbiome Project Consortium)³³ and downloaded from the *MicrobiomeAnalyst*²² website, which contained 23 CD (Crohn’s Disease) and 20 control samples. We covered the calculation of the alpha and beta diversity indices, the analysis of the diversity measures with the related test method (subsequent Mann–Whitney U test and Permutational Multivariate Analysis of Variance), and visualization of patterns in the microbiota community using *MicrobiotaProcess*. The result of alpha diversity showed that the richness of the CD’s microbiota community was significantly lower than the control’s (**Fig.3A and 3B**), the evenness of the microbiota species between the CD and control group did not have significant difference but the median of the diversity of the CD group was higher than the control group (**Fig.3B**). Then the PCoA (principal coordinate decomposition) and the *adonis* (Permutational Multivariate Analysis of Variance) results revealed that the microbiota communities between the CD and control group had significant difference (**Fig.3C**). We also found the Proteobacteria might be more abundant in CD by combining the hierarchical relationships and the phyla abundance presented as the horizontal stacked bar charts (**Fig.3D**). To further investigate the relationship between the microbial communities and CD, we used *mp_cal_abundance* and *mp_plot_abundance* to display the abundance of class level, this result showed that the abundance of Gammaproteobacteria belonged to Proteobacteria might be enriched in the CD group (**Fig.3E and 3F**). Next, we used *mp_diff_analysis* to identify the different OTUs, the result also revealed that the OTUs significantly enriched in the CD group mainly belong to Proteobacteria, whereas the OTUs significantly enriched in the Control group mainly belonged to Firmicutes (**Fig.3G**). Through the tidy-like unified grammar of *MicrobiotaProcess*, the alpha and beta diversity of the stool microbial communities between the Crohn’s Disease patients and control can be explored rapidly. The differential microbial organisms between the Crohn’s Disease patients and the control groups can also be identified and visualized intuitively.

Example 2: Differential abundance analysis and functional characterization of metagenomic

data

The metagenomics taxa abundance and KEGG gene abundance datasets were obtained from another pediatric Crohn's disease study³⁴. We used the *mp_diff_analysis* function to identify the differential species and genes, followed by KEGG pathway enrichment analysis of differential genes using *clusterProfiler*²⁹ and *MicrbiomeProfiler*³⁵. We found the *Clostridium symbiosum* and *Faecalibacterium prausnitzii* were nominally (uncorrected p-value ≤ 0.05 , $\text{fdr} > 0.05$) significant-enriched in the CD group compared to the control group (**Fig.S20** and **Fig.S21**). Interestingly, the KEGG enrichment results showed that the KEGG pathways of the CD stool group were significantly enriched in the Biosynthesis of amino acids and Glycine, serine, and threonine metabolism, and Pyruvate metabolism (**Fig.4**). This result is consistent with a recent study, which found that Crohn's Disease microbiomes had an increased potential to synthesize amino acids³⁶. However, the differential species result is inconsistent with the findings in example 1. This may be due to the differences of samples and sequencing platforms between the studies, and the relationships between the different species and the pediatric CD need to be further investigated. In this example, we use *MicrobiotaProcess* for differential abundance analysis, and then use the in-house developed packages, *clusterProfiler*²⁹ and *MicrbiomeProfiler*³⁵, for functional enrichment analysis. The combination of these packages facilitates the functional interpretation of metagenomic data and the discovery of functional signatures that leads to disease progression.

Example 3: Analyzing environmental-trait interactions in ecological communities

The dataset is from a study on the landscape ecology of mosquito communities in North Carolina for determining the risk for vector-borne disease³⁷. Three agricultural/mixed-use landscapes (regions) were selected. Within each region, three 200m transects that spanned a field-forest habitat gradient were identified, and five traps were set for each transect to collect mosquitoes³⁷. Here, we used *MicrobiotaProcess* to calculate alpha diversity and visualize the patterns of the community composition to evaluate whether the mosquito community structure was influenced by spatial variability. We found the richness of the mosquito species was significantly associated with the habitat with mean richness increasing from field to forest (**Fig.5A**). The partial constrained correspondence analysis (pCCA) result showed that the mosquito communities were significantly associated with some environment landscape variables, such as grassland (Grassland), evergreen tree canopy (EvergreenForest), and deciduous tree canopy (DeciduousForest) (**Fig.5B**). The mosquito species showed a clear preference for forested or field habitat (**Fig.S23**). These results were consistent with the findings of the original study³⁷. This example shows that *MicrobiotaProcess* can be used not only for analyzing microecological data but also for general ecological data, helping us to discover the impact of environmental factors on community structure and functions.

Example 4: Interact with existing tools

Since the *MPSE* class inherits *SummarizedExperiment*¹², the methods developed for *SummarizedExperiment*¹² can also be applied to an *MPSE* object. For example, the *tidybulk*²⁸ package provides a *test_differential_abundance* to perform differential transcriptome analysis using

*edgeR*³⁸ quasi-likelihood (QLF), *edgeR* likelihood-ratio (LRT), limma-voom, limma-voom-with-quality-weights, or DESeq2³⁹. These methods can be directly applied to an *MPSE* object within the tidy framework for differential abundance analysis and indeed they have been reported for analyzing microbiome data^{40,41}. Here, we re-analyze example 1 by using the *test_differential_abundance* function to perform the differential analysis based on the *edgeR* quasi-likelihood³⁸. Then the result was extracted by the *mp_extract_tree* function and can be manipulated and visualized using the *ggtree* package suite. Compared with the results of example 1, the number of the differential OTUs identified by *edgeR*³⁸ is more (**Fig.S16**) and the main conclusions are consistent. We found the abundance of OTUs belonged to *Bifidobacterium*, *Faecalibacterium*, *Roseburia*, and *Coprobacillus* were also significantly decreased in the CD group compared to the control group and the abundance of several OTUs belonged to *Escherichia*, *Klebsiella* and *Haemophilus*, which belonged to Gammaproteobacteria, were significantly enriched in the CD group (**Fig.6**). *MicrobiotaProcess* is able to interoperate with existing tools, including *tidybulk*, *dplyr*, *ggtree* package suite, *clusterProfiler* and *MicrobiomeProfiler* that we have demonstrated earlier. This greatly expands the functionalities of *MicrobiotaProcess*. It can connect upstream data, unify downstream analysis, and easily integrate into existing analysis pipelines.

Conclusion and Discussion

Microbiomics technologies have become increasingly popular methods for exploring the relationship between the microbial communities and the host or the environment (e.g., intestine, skin, soil, and the ocean)^{1,33,42–46}. Data analysis is still one of the bottlenecks in this field, especially in downstream analysis, the integration of heterogeneous data and the need for personalized analysis have brought new challenges. *MicrobiotaProcess* provides a comprehensive data structure, the *MPSE* class, to store heterogeneous microbiome data, including feature (e.g., species, OTUs, etc.) abundance, sample data (e.g., clinical information), and feature data (e.g., taxonomic relationships, functional profiles, etc.). Moreover, simultaneously with the analysis, intermediate results can be stored in an *MPSE* object. For example, the normalized or rarefied data, the diverse dissimilarity metrics, and the results of the differential analysis can be integrated with the microbiome data into an *MPSE* object. These intermediate results can be further processed, visualized, and re-used. This can prevent repeated calculation, facilitate data sharing, and enhance analytic reproducibility. *MicrobiotaProcess* implements a set of functions within the tidy principles to unveil the characteristics and biological function of microbial communities in a diverse environment. The returned values of these functions are predictable and consistent. Each function is designed to accomplish a simple task and these functions can be combined to accomplish complex tasks. In this way, it has better flexibility, and the development of workflow through function series can meet most of the personalized analysis needs. Considering the interoperation between *MicrobiotaProcess* and other tools, some functions developed by other software can be applied to *MPSE* objects, making the downstream analysis function more comprehensive and having wider application scenarios. For example, differential analysis using *tidybulk*²⁸ and functional analysis using *clusterProfiler*²⁹ can be integrated into the analysis of microbiome and other ecological data through *MPSE* objects.

MicrobiotaProcess provides several functions to parse outputs obtained from upstream tools and

allows converting commonly used objects to *MPSE* objects (**Fig.2A**). This enables the functionalities provided by *MicrobiotaProcess* to be well connected to upstream analysis. The downstream analysis results need to be interpreted through visualization. *MicrobiotaProcess* provides several functions for visualization. More importantly, the tidy interface extracts relevant data and connects to the visualization functions developed by the R community, allowing users to have more options for visual exploration and presentation of data. For example, differential analysis results can be extracted by *mp_extract_tree*, followed by a visual exploration of the result via the *ggtree* package suit (**Fig.3G and Fig.6**). Although *MicrobiotaProcess* was mainly designed for analyzing microbiome data, some of its methods, such as alpha, beta diversity, and (partial) constrained correspondence analysis (pCCA or CCA) and redundancy analysis (RDA), etc.⁴⁸, are also applicable to other ecological data (**Fig.5**). We believe that *MicrobiotaProcess* will be a valuable resource to support the analysis of the microbiome and other ecological data. This package is freely available at <https://www.bioconductor.org/packages/MicrobiotaProcess>. The development version of *MicrobiotaProcess* is hosted on GitHub (<https://github.com/YuLab-SMU/MicrobiotaProcess>).

Methods

The MPSE class

To better store the input data (abundance data, sequence data, phylogenetic tree data) and the result of downstream analysis, *MPSE* class was implemented in the *MicrobiotaProcess* package. This class inherits the *SummarizedExperiment*¹² class. In which, the *assays* slot was designed to store the rectangular abundance matrices of features (microbiota profiling or function profiling) for microbiome experiment results. The *colData* slot was designed to store the meta-data of features and results about the features generated in the downstream analysis. Compared to the *SummarizedExperiment*¹² class, *MPSE* introduces the following additional slots, 1) the *otutree* slot is a *treedata* object and was designed to store the phylogenetic tree and the associated data, including the results about the features in the downstream analysis and the evolutionary statistics inferred by the software building the tree; 2) the *taxatree* slot is also a *treedata*^{14,15} object and was designed to store hierarchical taxonomy relationships and the associated data, such as the relative abundances and the results from different analysis; 3) the *refseq* slot is a *XStringSet*³¹ object and was designed to store the reference sequences, with names corresponding to the rows of the *assays* slot. In addition, an internal attribute *internal_attr* (a list object) was introduced to store the raw results of *pca* (Principal Components Analysis), *pcor* (Principal Coordinate Analysis), and hierarchical cluster analysis.

Overview of the design of the MicrobiotaProcess package

The overall design of the *MicrobiotaProcess* package was illustrated in **Fig.2**. It presents multiple parser functions to read the outputs of upstream analysis tools, such as *qiime2*³ or *qiime*⁴⁹, *dada2*⁴, and *MetaPhlAn*^{7,8}. After parsing, the abundance of microbiota (or other features), the metadata of sample (optional), and the phylogenetic tree information (optional) are extracted from the outputs and stored as an *MPSE* object. Other objects designed to store microbiome data, such as *phyloseq*¹⁷, *TreeSummarizedExperiment*¹⁸, and *SummarizedExperiment*¹² can be converted to an *MPSE* object.

This enables *MPSE* to serve as a standardized entry point for downstream analysis while being compatible with existing analysis software and pipelines. *MicrobiotaProcess* provides a wide variety of microbiome analysis procedures to work with *MPSE* objects. These procedures were designed to follow the tidy data principles and thus are human-friendly, consistent, and composable to solve complicated problems. The results of the analysis can be returned in three modes via the *action* argument. The (intermediate) result can be stored in the *MPSE* object if the action is ‘add’. If the action is ‘only’, it returns a tidy data frame with non-redundant sample or OTU (features) information with the result. While the action is ‘get’ returns the analysis result only. *MicrobiotaProcess* also extends the *dplyr*²⁴ package to offer *dplyr*-verbs for data operation.

Parser functions and the MPSE constructor

The *MicrobiotaProcess* package provides *mp_import_qiime2* to load the output of *qiime2*³, which is a common tool for the analysis of amplicon data. The feature abundance table of the output (i.e., a *qza* format file) of *qiime2* is required while the taxonomy information, phylogenetic tree, and representative sequences are optional. The *mp_import_dada2* function was designed to parse the output of *dada2*⁴. The output of *removeBimeraDenovo* of *dada2*⁴ is required while the taxonomy information, representation phylogenetic tree, and the representative sequences are optional. The *mp_import_metaphlan* function was designed to parse the output of *MetaPhlAn3*⁵⁰, which is a common tool for profiling the composition of microbial communities. The microbiota abundance output of *MetaPhlAn3* is required, while the phylogenetic tree and metadata are also optional. In addition, An *MPSE* object can be constructed from scratch by calling the *MPSE* function with the following key parameters:

- 1) *assays* (required): A *list* or *SimpleList* of matrix-like objects or a matrix-like object (rows represent the features and columns represent the samples) providing abundance data for all samples.
- 2) *colData* (optional): A *DataFrame* object storing the characteristics of the samples.
- 3) *otutree* (optional): A *treedata* object storing a phylogenetic tree with/without associated data. Any tree file formats as well as commonly used software outputs that can be parsed by *treeio*¹⁵ are supported.
- 4) *taxatree* (optional): A *treedata* object storing the taxonomy information (or other hierarchical data). *MicrobiotaProcess* provides the *convert_to_treedata* function to convert taxonomy data (a *data.frame* object) to a *treedata* object.
- 5) *refseq* (optional): A *XStringSet* object storing the representative sequences. Both nucleic acid sequences and amino acid sequences are supported via the *readDNAStringSet* or *readAAStringSet* functions provided by the *Biostrings* packages.

Process MPSE object using dplyr-verbs

To facilitate data manipulation and exploration of the microbiome data, *MicrobiotaProcess* defined a tidy-like formatted output for the *MPSE* object and extended a subset of the *dplyr*-verbs to support the *MPSE* object. The extended *dplyr*-verbs include:

- 1) *filter* function: subset *MPSE* class, retaining the data that satisfy the provided conditions.
- 2) *select* function: select the data according to the provided variables.
- 3) *group_by* function: return a *grouped tbl_df-like* data frame according to groups defined by the provided variables, then some data operations can be done on the groups.
- 4) *mutate* function: create new columns according to the provided variables and conditions.

5) *left_join* function: add columns from 'y' (a *data.frame* object) to 'x' (an *MPSE* object), by matching all rows of 'x' based on the keys. The keys only should be one or all of the *Sample* or *OTU*.

6) *rename* function: rename the column names of an *MPSE* object, except the *OTU*, *Sample*, and *Abundance* column names which cannot be renamed.

Data preprocessing

The microbiome features (OTU or ASV) with very low abundance and rare occurrence (exists in few samples) are difficult to distinguish from the sequencing error or other experimental technical error and are usually uninformative. It is better to improve the statistical power of multiple testing in the downstream analysis by filtering these features. *MicrobiotaProcess* presents *mp_filter_taxa* to filter the features based on their abundance (default *Abundance* count) and sample prevalence. By default, this function will screen out the features with zeros counts in 0.05% sample prevalence and users can reset the criteria via the parameters of *min.abun* (minimum abundance in a sample) and *min.prop* (the minimum sample prevalence). To make the data more meaningful for the downstream analysis after filtering, *MicrobiotaProcess* provides the *mp_decostand* and *mp_rrarefy* functions for the standardization of community data by inheriting the *decostand* and *rrarefy* functions from *vegan*⁴⁸. The *mp_rrarefy* function allows users to estimate expected diversity (e.g., taxonomic richness) for a reduced sampling size, while the *mp_decostand* function provides several standardization methods for community data, such as *total* (divide by total abundance of each sample (relative abundance)), *max* (divide by the max abundance of the feature in all samples), *frequency* (divide by total abundance of each sample and multiply the number of non-zero features), *hellinger* (square root for the result of *total*)⁵¹, *log* (logarithmic transformation $\log_b(x > 0) + 1$)⁵². These functions are developed to follow the tidiness concept and the results will be added to the *assays* slot of the *MPSE* object automatically to enhance reproducibility and reuse in the follow-up analysis.

Alpha diversity

Alpha diversity measures the species richness and evenness within a community. *MicrobiotaProcess* provides the *mp_cal_alpha* function to calculate the community diversity. There are six commonly used methods to calculate alpha diversity, including *Observe* (calculates the total species per sample), *Chao1*, *ACE* (estimate species richness by considering the low abundance of species), *Pielou* (measures the species evenness), *Shannon* and *Simpson* (take both of the richness and evenness of species into account). By default, the *mp_cal_alpha* will rarefy the abundance before calculating the alpha diversity. Users can specify the *force* argument to TRUE to calculate the diversity directly without performing rarefaction. This will be useful for taxonomic profiling data since they are usually relative abundance and cannot be rarefied. By default, the results are added to the *colData* slot that stored the metadata information of samples and returns an updated *MPSE* object. The *mp_plot_alpha* function is designed to visualize alpha diversity and it allows comparing different communities that were specified via the *.group* parameter.

Taxonomy composition

To compare the difference of OTUs (features) composition between different communities, *MicrobiotaProcess* provides the *mp_cal_upset* and *mp_cal_venn* to calculate the conjunct OTUs (features) or specific OTUs (features) of different groups (specified by the *.group* parameter). The

result can be visualized by the *mp_plot_upset* and *mp_plot_venn* functions respectively. The microbiome OTUs (features) are often annotated to different taxonomy levels in upstream analyses, and to survey the species profile of different samples, it is often necessary to calculate abundances of different taxonomy levels. *MicrobiotaProcess* implements the *mp_cal_abundance* function to calculate the abundances of all taxonomy levels. Similar to the calculation of alpha diversity, the *mp_cal_abundance* will rarefy the raw abundance and then calculate the relative abundance by default. Users can specify the *force* argument to TRUE to disable rarefaction. And the *relative* argument controls whether to calculate the relative abundance (total is 100% for the same taxonomy level). The results will be added to the associated data of the *taxatree* (*treedata* object) slot by default. The abundance of a selected taxonomy level can be extracted by the *mp_extract_abundance* function with the *taxa.class* parameter specified. The results of the *mp_cal_abundance* can be visualized by the *mp_plot_abundance* function.

Beta diversity

The beta diversity has been applied in a broad sense to measure variation or changes in community composition. It can assess how microbiota composition changes across spatial and temporal scales. Some distance indexes, such as Bray-Curtis index, Jaccard index, UniFrac (weighted or unweighted) index, are useful and popular to measure the degree of community differentiation. These distances can be further subjected to ordination which aims to capture essential information in a lower-dimensional representation and is commonly used to visualize sample dissimilarities. *MicrobiotaProcess* implements the *mp_cal_dist* function to compute the common distances (dissimilarity) and provides the *mp_plot_dist* function to visualize the result. It also provides several commonly-used ordination methods, such as Principal Components Analysis (PCA: *mp_cal_pca*), Principal Coordinate Analysis (PCoA: *mp_cal_pcoa*), Nonmetric Multidimensional Scaling (NMDS: *mp_cal_nmds*), Detrended Correspondence Analysis (DCA: *mp_cal_dca*), Redundancy Analysis (RDA: *mp_cal_rda*), and (Constrained) Correspondence Analysis (CCA: *mp_cal_cca*). To fit environmental vectors or factors onto an ordination, this package provides the *mp_envfit* function to perform this analysis. All the ordination results can be visualized by the *mp_plot_ord* function. In addition, it also wraps several statistical analyses for the distance matrices, such as permutational multivariate analysis of variance (*mp_adonis*), analysis of similarities (*mp_anosim*), multi-response permutation procedure (*mp_mrpp*), and mantel tests for dissimilarity matrices (*mp_mantel*). All these functions are developed based on a tidy-like framework. These functions can be assembled into linear workflows with the pipe operator (*%>%*).

Differential analysis and biomarker discovery

MicrobiotaProcess implements the *mp_diff_analysis* function for identifying differentially abundant genera as biomarkers based on the tidy-like framework. Similar to *LEfSe*⁵³, there are three steps to perform this analysis. First, all features are tested to determine whether values (e.g., abundance) in different groups of samples are differentially distributed via the Kruskal-Wallis rank-sum test (default, other option is *oneway.test*, *glm* or *glm.nb*). Then, the resulting features infringing the null hypothesis are further tested by a second round of test using Wilcoxon rank-sum test (default, other option is *t.test*, *glm* or *glm.nb*) to keep the features that in all pairwise comparisons between the sub-groups are significantly consistent with the group level trend. Finally, the linear discriminant analysis (LDA) or random forest model was built to rank all the features based on the relative

difference among different groups. Compare to *LEfSe*, *mp_diff_analysis* is more flexible. Not only the test method but also the test value (using generalized fold change⁵⁴ by default, other option is comparing the median or mean value of different groups by setting to `compare_median` or `compare_mean`) can be set by users. The result is integrated into the *taxatree* component (default) or *rowData* component depending on whether the taxonomy is provided or not. The result can be extracted via the *mp_extract_tree* or *mp_extract_feature* respectively. Then it can be processed and displayed via *treeio*¹⁵, *tidytree*¹⁴, *ggtree*¹⁶, *ggtreeExtra*³⁰, and *ggplot2* (**Fig.3G** and **Fig.6**). To decrease the coding burden, we also developed *mp_plot_diff_res* to visualize the result of differential analysis (**Fig.S13**).

Accessors to fetch internal data

The *MPSE* object is composed of several objects to store different data including primary data and analysis results. To extract the components of the data, *MicrobiotaProcess* provides several accessors started with ‘*mp_extract*’, including:

1) *mp_extract_sample* function: to return the sample characteristics in a tidy data table, similar to the *colData* function for the *SummarizedExperiment* class.

2) *mp_extract_assays* function: to extract the *assays* from an *MPSE* object, similar to the *assay* function for the *SummarizedExperiment* class.

3) *mp_extract_feature* function: to extract the features characteristics (optional with taxonomy information by setting *addtaxa* argument to TRUE) and return tidy data, similar to the *rowData* function for the *SummarizedExperiment* class.

4) *mp_extract_tree* function: to extract the *taxatree* (by default) or *otutree* (by specifying the *type* argument to *otutree*) from an *MPSE* object.

5) *mp_extract_taxonomy* function: to extract the taxonomy information from an *MPSE* object.

6) *mp_extract_refseq* function: to extract the representative sequences from an *MPSE* object.

7) *mp_extract_dist* function: to extract distances in a matrix (as a *dist* object, by default) or a tidy data frame with comparison among the groups (by specifying the *.group* argument).

8) *mp_extract_rarecurve* function: to extract rarefaction in a *rarecurve* object, which can be visualized by *ggrarecurve* function.

9) *mp_extract_internal_attr* function: to extract the raw result of *mp_cal_pca*, *mp_cal_pcoa*, *mp_cal_rda*, *mp_cal_cca*, *mp_cal_clust*, *mp_envfit*, *mp_adonis*, *mp_anosim*, *mp_mrpp* and *mp_mantel*.

References

1. Turnbaugh, P. J. *et al.* The Human Microbiome Project. *Nature* **449**, 804–810 (2007).
2. Lloyd-Price, J., Abu-Ali, G. & Huttenhower, C. The healthy human microbiome. *Genome Medicine* **8**, 51 (2016).
3. Bolyen, E. *et al.* Reproducible, interactive, scalable and extensible microbiome data science using QIIME 2. *Nature Biotechnology* **37**, 852–857 (2019).

4. Callahan, B. J. *et al.* DADA2: High-resolution sample inference from Illumina amplicon data. *Nature Methods* **13**, 581–583 (2016).
5. Edgar, R. C. UPARSE: highly accurate OTU sequences from microbial amplicon reads. *Nature Methods* **10**, 996–998 (2013).
6. Schloss, P. D. *et al.* Introducing mothur: Open-Source, Platform-Independent, Community-Supported Software for Describing and Comparing Microbial Communities. *Applied and Environmental Microbiology* **75**, 7537–7541 (2009).
7. Segata, N. *et al.* Metagenomic microbial community profiling using unique clade-specific marker genes. *Nature Methods* **9**, 811–814 (2012).
8. Truong, D. T. *et al.* MetaPhlAn2 for enhanced metagenomic taxonomic profiling. *Nature Methods* **12**, 902–903 (2015).
9. Lowndes, J. S. S. *et al.* Our path to better science in less time using open data science tools. *Nature Ecology & Evolution* **1**, 0160 (2017).
10. Ihaka, R. & Gentleman, R. R: A Language for Data Analysis and Graphics. *Journal of Computational and Graphical Statistics* **5**, 299–314 (1996).
11. Huber, W. *et al.* Orchestrating high-throughput genomic analysis with Bioconductor. *Nature Methods* **12**, 115–121 (2015).
12. Morgan, M., Obenchain, V., Hester, J. & Pagès, H. *SummarizedExperiment: SummarizedExperiment container.* (2021).
13. Gentleman, R. C. *et al.* Bioconductor: open software development for computational biology and bioinformatics. *Genome Biology* **5**, R80 (2004).
14. Yu, G. *tidytree: A Tidy Tool for Phylogenetic Tree Data Manipulation.* (2021).
15. Wang, L.-G. *et al.* treeio: an R package for phylogenetic tree input and output with richly annotated and associated data. *Molecular Biology and Evolution* **37**, 599–603 (2020).
16. Yu, G., Smith, D. K., Zhu, H., Guan, Y. & Lam, T. T.-Y. ggtree: an r package for visualization and annotation of phylogenetic trees with their covariates and other associated data. *Methods in Ecology and Evolution* **8**, 28–36 (2017).
17. McMurdie, P. J. & Holmes, S. phyloseq: An R Package for Reproducible Interactive Analysis and Graphics of Microbiome Census Data. *PLOS ONE* **8**, 1–11 (2013).
18. Huang, R. *et al.* TreeSummarizedExperiment: a S4 class for data with hierarchical structure. *F1000Research* **9**, 1246 (2021).
19. Barnett, D. J., Arts, I. C. & Penders, J. microViz: an R package for microbiome data visualization and statistics. *Journal of Open Source Software* **6**, 3201 (2021).
20. Liu, C., Cui, Y., Li, X. & Yao, M. microeco: an R package for data mining in microbial community ecology. *FEMS Microbiology Ecology* **97**, fiae255 (2021).
21. Zhao, Y. *et al.* animalcules: interactive microbiome analytics and visualization in R. *Microbiome* **9**, 76 (2021).
22. Dhariwal, A. *et al.* MicrobiomeAnalyst: a web-based tool for comprehensive statistical, visual and meta-analysis of microbiome data. *Nucleic Acids Research* **45**, W180–W188 (2017).
23. Wickham, H. Tidy Data. *Journal of Statistical Software* **59**, 1–23 (2014).
24. Wickham, H. *et al.* Welcome to the tidyverse. *Journal of Open Source Software* **4**, 1686 (2019).
25. Wickham, H. ggplot2. *WIREs Computational Statistics* **3**, 180–185 (2011).
26. Lee, S., Cook, D. & Lawrence, M. plyranges: a grammar of genomic data transformation.

Genome Biology **20**, 4 (2019).

27. Lawrence, M. *et al.* Software for Computing and Annotating Genomic Ranges. *PLoS Computational Biology* **9**, (2013).
28. Mangiola, S., Molania, R., Dong, R., Doyle, M. A. & Papenfuss, A. T. tidybulk: an R tidy framework for modular transcriptomic data analysis. *Genome Biology* **22**, 42 (2021).
29. Wu, T. *et al.* clusterProfiler 4.0: A universal enrichment tool for interpreting omics data. *The Innovation* **2**, 100141 (2021).
30. Xu, S. *et al.* ggtreeExtra: Compact Visualization of Richly Annotated Phylogenetic Data. *Molecular Biology and Evolution* **38**, 4039–4042 (2021).
31. Pagès, H., Aboyoun, P., Gentleman, R. & DebRoy, S. *Biostrings: Efficient manipulation of biological strings*. (2021).
32. McMurdie, P. J. & Paulson, J. N. *biomformat: An interface package for the BIOM file format*. (2021).
33. Consortium, I. H. (iHMP) R. N. The Integrative Human Microbiome Project: Dynamic Analysis of Microbiome-Host Omics Profiles during Periods of Human Health and Disease. *Cell Host & Microbe* **16**, 276–289 (2014).
34. Douglas, G. M. *et al.* Multi-omics differentially classify disease state and treatment outcome in pediatric Crohn's disease. *Microbiome* **6**, 13 (2018).
35. Chen, M. & Yu, G. *MicrobiomeProfiler: An R/shiny package for microbiome functional enrichment analysis*. (2021).
36. Heinken, A., Hertel, J. & Thiele, I. Metabolic modelling reveals broad changes in gut microbial metabolism in inflammatory bowel disease patients with dysbiosis. *npj Systems Biology and Applications* **7**, 19 (2021).
37. REISKIND, M. H., GRIFFIN, R. H., JANAIRO, M. S. & HOPPERSTAD, K. A. Mosquitoes of field and forest: the scale of habitat segregation in a diverse mosquito assemblage. *Medical and Veterinary Entomology* **31**, 44–54 (2017).
38. Robinson, M. D., McCarthy, D. J. & Smyth, G. K. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics* **26**, 139–140 (2010).
39. Love, M. I., Huber, W. & Anders, S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology* **15**, 550 (2014).
40. Weiss, S. *et al.* Normalization and microbial differential abundance strategies depend upon data characteristics. *Microbiome* **5**, 27 (2017).
41. Ventero, M. P. *et al.* Evolution of intestinal microbiome in a process of faecal microbiota transplantation in a patient with *Clostridioides difficile* infection: NGS analysis with different bioinformatics software programs. *Enfermedades infecciosas y microbiología clínica (English ed.)* **39**, 184–187 (2021).
42. Shreiner, A. B., Kao, J. Y. & Young, V. B. The gut microbiome in health and in disease. *Curr Opin Gastroenterol* **31**, 69–75 (2015).
43. He, J. *et al.* Gut microbiota and metabolite alterations associated with reduced bone mineral density or bone metabolic indexes in postmenopausal osteoporosis. *Aging (Albany NY)* **12**, 8583–8604 (2020).
44. Sunagawa, S. *et al.* Structure and function of the global ocean microbiome. *Science* **348**, 1261359 (2015).

45. Grice, E. A. & Segre, J. A. The skin microbiome. *Nature Reviews Microbiology* **9**, 244–253 (2011).
46. Fierer, N. Embracing the unknown: disentangling the complexities of the soil microbiome. *Nature Reviews Microbiology* **15**, 579–590 (2017).
47. Ritchie, M. E. *et al.* limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Research* **43**, e47–e47 (2015).
48. Oksanen, J. *et al.* *vegan: Community Ecology Package*. (2020).
49. Caporaso, J. G. *et al.* QIIME allows analysis of high-throughput community sequencing data. *Nature Methods* **7**, 335–336 (2010).
50. Beghini, F. *et al.* Integrating taxonomic, functional, and strain-level profiling of diverse microbial communities with bioBakery 3. *eLife* **10**, e65088 (2021).
51. Legendre, P. & Gallagher, E. D. Ecologically meaningful transformations for ordination of species data. *Oecologia* **129**, 271–280 (2001).
52. Anderson, M. J., Ellingsen, K. E. & McArdle, B. H. Multivariate dispersion as a measure of beta diversity. *Ecology letters* **9**, 683–693 (2006).
53. Segata, N. *et al.* Metagenomic biomarker discovery and explanation. *Genome Biology* **12**, R60 (2011).
54. Wirbel, J. *et al.* Meta-analysis of fecal metagenomes reveals global microbial signatures that are specific for colorectal cancer. *Nature Medicine* **25**, 679–689 (2019).

Fig.1 The structure and formatted output of the *MPSE* class. **(A)** *MPSE* class instantiates an object capable of storing various data types generated from a microbiome study. In the assays component, the rows represent features such as OTUs, species, or genes (horizontal), and the columns represent sample (vertical). The *rowData* and *colData* components can hold information (such as metadata) about those features and samples, respectively. The *otutree* and *taxatree* components can store the phylogenetic tree of the features and the species tree of the features, respectively. The *refseq* component can store the sequences of features. The component can also be extracted via the corresponding extraction functions (composed of ‘*mp_extract_*’ and a specific explanatory term). **(B)** A tidy-like output of the *MPSE* class (via ‘*options(MPSE_show_tbl=TRUE)*’). The rows represent the values (abundance, annotation, or other metadata) of each feature in each sample.

Fig.2 Overview of the design of the *MicrobiotaProcess* package. **(A)** The package provides several functions to import upstream analysis results (*mp_import_qiime2*, *mp_import_metaphlan*, and *mp_import_dada2*, etc), several converters (*as.MPSE* methods) to import data from other objects, and a method (*MPSE*) for building an *MPSE* object from scratch. Unifying downstream data processing and data analysis through *MPSE* objects. **(B)** Some functions provided in the *dplyr* and *tidybulk* packages are compatible to work with *MPSE* objects. **(C1 and C2)** *MicrobiotaProcess* provides a wide variety of functions for routine microbiome analysis. The functions were designed to follow the tidy principle.

Fig.3 Characterizing gut microbiota in patients with CD (Crohn’s Disease). **(A)** The rarefaction

curves of the sample from CD and control groups (visualized by *mp_plot_rarecurve*). **(B)** The alpha diversity boxplot between the CD and control groups, and labeled with the p.value calculated by Mann–Whitney *U* test (visualized by *mp_plot_alpha*). **(C)** The PCoA (principal coordinate decomposition) plot and the result of Adonis (Permutational Multivariate Analysis of Variance analysis) (visualized by *mo_plot_ord*). **(D)** A cladogram displayed the hierarchical clustering result of the samples based on the OTU abundance, and a bar chart displayed the relative abundance at the phyla level (visualized by *ggtree* and *ggtreeExtra*). **(E)** The relative abundance bar chart at the class level for each group. **(F)** The relative abundance heatmap at the class level of each sample (visualized by *mp_plot_abundance*). **(G)** The taxa tree of the community was visualized with associated data, including the relative abundance and the LDA (Linear discriminant analysis) value of differential OTUs. The background color in the tree represents the phyla of the clade. The external point layer represents the relative abundance of each OTU on each sample. The external bar layer represents the mean LDA of the differential OTUs. The colored points represent the differential taxa, and the sizes were scaled by the *fdr* (*kruskal.test*) (visualized by *mp_plot_diff_res*).

Fig.4 Comparing functional profiles between the CD and control groups. **(A)** The Category-Gene Network was employed to visualize KEGG enrichment results between the CD and control groups. **(B)** The dot plot of the KEGG enrichment results. (visualized by *cnetplot* and *dotplot* provided by the *enrichplot* package)

Fig.5 Characterizing mosquito community structure with the raincloud plot of the alpha diversity and the pCCA plot of the mosquito communities. **(A)** The alpha diversity result showed that the mosquito species richness gradually increases from field to forest (visualized by *mp_plot_alpha*). **(B)** The pCCA result showed that the mosquito communities were significantly associated with some environment landscape variables, such as grassland (Grassland), evergreen tree canopy (EvergreenForest), and deciduous tree canopy (DeciduousForest). Each point represents one sample and the corresponding data of the samples are encoded as visual characteristics of the points (observe species as to size, habitat as to color, and the region as to shape). The arrows represent the environmental factors and the asterisks indicate significant levels related to the Mosquito communities in the study (* 0.05, ** 0.01, ***0.001) (visualized by *mp_plot_ord*).

Fig.6 The results of differential OTUs based on the *edgeR_quasi_likelihood* method using *tidybulk*. **(A)** The relative abundance heatmap of the different OTUs (visualized by the *mp_plot_abundance* function). **(B)** The hierarchical cluster of samples is based on the relative abundance of the different OTUs (visualized by *ggtree*). **(C)** The hierarchical cluster of OTUs based on their relative abundance across different samples, the differential OTUs were labeled with their names (visualized by *ggtree*).

Figures

Figure 1

The structure and formatted output of the MPSE class. (A) MPSE class instantiates an object capable of storing various data types generated from a microbiome study. In the assays component, the rows represent features such as OTUs, species, or genes (horizontal), and the columns represent sample (vertical). The rowData and colData components can hold information (such as metadata) about those features and samples, respectively. The otutree and taxatree components can store the phylogenetic tree of the features and the species tree of the features, respectively. The refseq component can store the sequences of features. The component can also be extracted via the corresponding extraction functions (composed of 'mp_extract_' and a specific explanatory term). (B) A tidy-like output of the MPSE class (via 'options(MPSE_show_tbl=TRUE)'). The rows represent the values (abundance, annotation, or other metadata) of each feature in each sample.

Figure 2

Overview of the design of the MicrobiotaProcess package. (A) The package provides several functions to import upstream analysis results (mp_import_qiime2, mp_import_metaphlan, and mp_import_dada2, etc), several converters (as.MPSE methods) to import data from other objects, and a method (MPSE) for building an MPSE object from scratch. Unifying downstream data processing and data analysis through MPSE objects. (B) Some functions provided in the dplyr and tidybulk packages are compatible to work with MPSE objects. (C1 and C2) MicrobiotaProcess provides a wide variety of functions for routine microbiome analysis. The functions were designed to follow the tidy principle.

Figure 3

Characterizing gut microbiota in patients with CD (Crohn's Disease). (A) The rarefaction curves of the sample from CD and control groups (visualized by mp_plot_rarecurve). (B) The alpha diversity boxplot between the CD and control groups, and labeled with the p.value calculated by Mann-Whitney U test (visualized by mp_plot_alpha). (C) The PCoA (principal coordinate decomposition) plot and the result of Adonis (Permutational Multivariate Analysis of Variance analysis) (visualized by mo_plot_ord). (D) A cladogram displayed the hierarchical clustering result of the samples based on the OTU abundance, and a bar chart displayed the relative abundance at the phyla level (visualized by ggtree and ggtreeExtra). (E) The relative abundance bar chart at the class level for each group. (F) The relative abundance heatmap at the class level of each sample (visualized by mp_plot_abundance). (G) The taxa tree of the community

was visualized with associated data, including the relative abundance and the LDA (Linear discriminant analysis) value of differential OTUs. The background color in the tree represents the phyla of the clade. The external point layer represents the relative abundance of each OTU on each sample. The external bar layer represents the mean LDA of the differential OTUs. The colored points represent the differential taxa, and the sizes were scaled by the $\text{fdr}(\text{kruskal.test})$ (visualized by `mp_plot_diff_res`).

Figure 4

Comparing functional profiles between the CD and control groups. (A) The Category-Gene Network was employed to visualize KEGG enrichment results between the CD and control groups. (B) The dot plot of the KEGG enrichment results. (visualized by `cnetplot` and `dotplot` provided by the `enrichplot` package)

Figure 5

Characterizing mosquito community structure with the raincloud plot of the alpha diversity and the pCCA plot of the mosquito communities. (A) The alpha diversity result showed that the mosquito species richness gradually increases from field to forest (visualized by `mp_plot_alpha`). (B) The pCCA result showed that the mosquito communities were significantly associated with some environment landscape variables, such as grassland (Grassland), evergreen tree canopy (EvergreenForest), and deciduous tree canopy (DeciduousForest). Each point represents one sample and the corresponding data of the samples are encoded as visual characteristics of the points (observe species as to size, habitat as to color, and the region as to shape). The arrows represent the environmental factors and the asterisks indicate significant levels related to the Mosquito communities in the study (* 0.05, ** 0.01, ***0.001) (visualized by `mp_plot_ord`).

Figure 6

The results of differential OTUs based on the `edgeR_quasi_likelihood` method using `tidybulk`. (A) The relative abundance heatmap of the different OTUs (visualized by the `mp_plot_abundance` function). (B) The hierarchical cluster of samples is based on the relative abundance of the different OTUs (visualized by `ggtree`). (C) The hierarchical cluster of OTUs based on their relative abundance across different samples, the differential OTUs were labeled with their names (visualized by `ggtree`).

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [supplementalfile.pdf](#)