

MetaDAVis: An R shiny application for metagenomic data analysis and visualization

Authors: Sankarasubramanian Jagadesan and Chittibabu Guda

University of Nebraska Medical Center

Tutorial for MetaDAVis

The interactive Metagenome Data Analysis and Visualization (MetaDAVis) application analyzes 16S rRNA and whole genome sequencing (WGS) data at various taxonomic levels from kingdom to species. It is developed to function both as a standalone version for local installation and a web-based R Shiny application for researchers to analyze and visualize without programming proficiency. It comprises of six functional analysis modules.

1. Data Summary and Distribution
2. Diversity analysis
3. Dimension reduction
4. Correlation analysis
5. Heatmap
6. Differential abundance (Two and multiple groups)

In this tutorial, we will go through the installation and usage of each functional module using an example dataset. MetaDAVis is publicly available at (<https://github.com/GudaLab/MetaDAVis>) and <https://www.gudalab-rtools.net/MetaDAVis>. The example dataset is provided on the GitHub page (https://github.com/GudaLab/MetaDAVis/tree/main/www/example_data).

How to start MetaDAVis locally

Download the MetaDAVis application locally from the GitHub page (<https://github.com/GudaLab/MetaDAVis>).

Requirements:

- R ($\geq 4.4.1$), available at (<https://www.r-project.org/>)
- RStudio ($\geq 2024.09.0$) available at (<https://posit.co/download/rstudio-desktop/>)
- Bioconductor (≥ 3.19) and
- Shiny ($\geq 1.9.1$)

This Application was tested in Linux (Red Hat) and Windows 10

Start an R session using RStudio and run the following commands to install the shiny package:
install.packages("shiny")

To run MetaDAVis by the following commands in R:

```
library(shiny)  
shiny::runGitHub("MetaDAVis","GudaLab")
```

Or

Alternatively, download the source code from GitHub and run the following command in the R session using RStudio:

```
library(shiny)  
runApp('/path/to/the/MetaDAVis-master', launch.browser=TRUE)
```

The Interface of MetaDAVis will pop up. See **Figure S1**

MetaDAVis (interactive Metagenome Data Analysis and Visualization) is a browser-based and user-friendly R Shiny application for researchers without programming proficiency to analyze and visualize metagenomics results from kingdom to species level. It comprises six functional analyses.

The package includes the following:

- Data summary and abundance distribution
 - The data can be visualized in the stacked bar plot for abundance percentage, value, and relative frequency from 2 to 100 Taxa.
 - Group: Samples are grouped by given conditions based on the metadata.
 - Individual: Sample-based plots.
- Diversity analysis
 - Alpha: Seven different methods were used from the phyloseq package. The results were displayed in a box and violin plot with a summary table.
 - Beta: A total of 42 different diversity metrics were integrated from the phyloseq (unlist(distanceMethodList)) package with six selection methods. Results are visualized by bar and ordination with a summary table.
- Dimension reduction
 - PCA: The ggfortify and plotly package was used to display plots in 2D (with and without labels and frame) and 3D with their summary table.
 - t-SNE: Six different methods were used from the scatter package. The samples were displayed in a t-SNE plot (2 and 3 dimensions) with a summary table.
 - UMAP: Six different methods were used from the scatter package. Displays the UMAP plot in sample and cluster-based with a summary table.
- Correlation analysis
 - Taxa-based: The ggfortify package was used to display plot (with and without labels and frame) and their summary table.
 - Sample-based: Six different methods were used from the scatter package. The result was displayed in 2 and 3 dimensions t-SNE plots with a summary table.
- Heatmap: It was integrated with the ComplexHeatmap package. Display heatmap with and without row and column dendograms and names.
- Differential abundance
 - Two groups: Six different analyses were provided using the Wilcoxon Rank Sum test, t-test, metagenomeSeq, DESeq2, Limma-Voom, and edgeR. These will perform statistical analysis and generate plots and summary tables based on the significant taxa.
 - Multiple groups comparison: Two different analyses, such as Kruskal-Wallis test and ANOVA was used for more than multiple group comparisons. These will perform statistical analysis and generate plots and summary tables based on the significant taxa.

*It provides publication quality plots in seven formats: JPG, TIFF, PDF, SVG, BMP, EPS, and PS and summary tables (.csv format) to visualize and download.

use MetaDAVis online

MetaDAVis is deployed at: <https://www.gudalab-rtools.net/MetaDAVis>

Figure S1: Interface of MetaDAVis application

Table S1. List of R packages used to develop MetaDAVis application

R Packages	Used for	Citation	Web link
shiny	To develop the web and interactive application	(Chang et al., 2022)	https://github.com/rstudio/shiny
DT	Interface to the data tables	(Xie et al., 2022)	https://github.com/rstudio/DT
shinyFiles	A server-side file system viewer for shiny	(Pedersen et al., 2022)	https://github.com/thomasp85/shinyFiles
shinythemes	To use the shiny themes	(Chang 2021)	https://github.com/rstudio/shinythemes
ggplot2	To create plots and graphics	(Wickham 2016)	https://github.com/tidyverse/ggplot2
phyloseq	To explore microbiome profiles for alpha and beta diversity	(McMurdie and Holmes, 2013)	https://github.com/joey711/phyloseq
ggpubr	Do the graphics for the correlation plot	(Kassambara 2022)	https://github.com/kassambara/ggpubr
vegan	The beta diversity orientation methods	(Oksanen et al., 2017)	https://github.com/vegandeys/vegan
ggfortify	To plot PCA in 2D	(Tang et al., 2016)	https://github.com/sinhrks/ggfortify
plotly	To plot PCA in 3D	(Sievert 2020)	https://github.com/plotly/plotly.R
ggplotify	Convert plot to ggplot object	(Yu 2021)	https://github.com/GuangchuangYu/ggplotify
reshape2	To transform data into a different structure	Wickham H (2007)	https://github.com/hadley/reshape
tibble	To convert row names to column	(Müller and Wickham 2022).	https://github.com/tidyverse/tibble
scales	Scale functions visualization in a heatmap	(Wickham and Seidel, 2022)	https://github.com/r-lib/scales
dunn.test	Multiple comparisons using rank sums (used in the Kruskal-Wallis test)	(Dinno 2017)	https://github.com/cran/dunn.test
tidyR	Creating tidy data, where each column is a variable, each row is an observation	(Wickham and Girlich 2022)	https://github.com/tidyverse/tidyr
dplyr	Data manipulation: adds new variables that are functions of existing variables	(Wickham et al., 2022)	https://github.com/tidyverse/dplyr
devtools	To install several R packages	(Wickham et al., 2022)	
patchwork	Adding multiple plots together	(Pedersen 2022)	https://github.com/thomasp85/patchwork
GGally	Creating correlation plots	(Schloerke et al., 2022)	https://github.com/ggobi/ggally
BiocManager	To install Bioconductor packages	(Morgan 2022)	https://bioconductor.org/packages/BiocVersion/
ComplexHeatmap	Creating heatmap	(Gu 2022)	https://bioconductor.org/packages/ComplexHeatmap/
qvalue	Estimates for false discovery used in statistical analysis	(Storey et al., 2022)	https://bioconductor.org/packages/qvalue/
DESeq2	Statistical analysis for two groups or sets	(Love et al., 2014)	https://bioconductor.org/packages/DESeq2/
edgeR	Statistical analysis for two groups or sets	(Robinson et al., 2010)	https://bioconductor.org/packages/edgeR/
limma	Statistical analysis for two groups or sets	(Ritchie et al., 2015)	https://bioconductor.org/packages/limma/

metagenomeSeq	Statistical analysis for two groups or sets	(Paulson et al., 2013)	https://bioconductor.org/packages/metagenomeSeq/
bluster	Used in UMAP for creating k-means and graph-based clustering	(Lun 2022)	https://bioconductor.org/packages/bluster/
mia	Used for data wrangling in t-SNE and UMAP	(Ernst et al., 2022)	https://bioconductor.org/packages/mia/
scater	Creating t-SNE and UMAP plots	(McCarthy et al., 2017)	https://bioconductor.org/packages/scater/
microbiome	Utilities for microbiome analysis	(Lahti et al., 2019)	https://bioconductor.org/packages/microbiome/
microbiom utilities	Pairwise comparison using a non-parametric test (Wilcoxon test) in alpha diversity	(Shetty and Lahti, 2022)	https://github.com/microbiome/microbiome/

Data preparation

This section will introduce how to prepare input data sets: read counts with complete taxonomy (kingdom to species level) and corresponding metadata.

Counts and metadata input formats

This application will accept input files in .txt, .tsv, or .csv format. User can directly upload level 7 of Qiime2 results generated using Greengenes or Silva. Likewise, it will support MEGAN data from the whole metagenome sequence (remove the metadata column if it is included in the level7.csv file from Qiime2). If the user has a different output format, they need to prepare their data count file and metadata for analysis. For file preparation, please refer to our example count data and metadata files from the upload files pages shown in **Figure S2** or example datasets from the GitHub repository ([\(\)](#)). The metadata files contain two columns. The first column contains the Samples which need to match the sample IDs in the count data input (**Figure S2**). The second column is Condition which indicates any user-specified categorical variable, such as "case" and "control" (two or multiple groups).

The screenshot shows the MetaDAVis web application interface. At the top, there is a navigation bar with tabs: MetaDAVis, Upload files, Distribution ▾, Diversity ▾, Dimension reduction ▾, Correlation ▾, Heatmap, and Differential abundance ▾. Below the navigation bar, there are two main sections: 'Upload files' on the left and a summary section on the right.

Upload files: This section includes fields for selecting the input format (Qiime2), uploading count files (Browse...), and uploading meta-data files (Browse...). It also allows users to choose the level to display (Kingdom, Phylum, Class, Order, Family, Genus, Species, with Order selected), and a 'Submit' button.

Summary: This section contains tabs for Summary, Taxonomy table, Metadata table, No. of conditions, and Counts in samples. A red box highlights the 'Example data' link under the 'Taxonomy table' tab, which points to a list of supported formats: Qiime2 format, Qiime2 Greengenes Output format, Qiime2 metadata for greengenes, Qiime2 Silva Output format, Qiime2 metadata for Silva, MEGAN output format, Megan WGS output format, Megan metadata, Taxa count file (Prepare your input according to our count and metadata format), Count files format, and Metadata.

Text at the bottom: After the data is uploaded and checked, it will be displayed in the table summary below.

Figure S2: Example data were provided for Qiime2, MEGAN output formats. If users have a different output format, prepare the files according to the taxa count file format

Using MetaDAVis

This section will introduce step-by-step instructions for each functional analysis using the example dataset provided at the MetaDAVis GitHub repository ([\(\)](#)). Using our application, users can download the plot in publication quality in seven different formats: JPG, TIFF, PDF, SVG, BMP, EPS, and PS. The summary tables were displayed using the DT package to visualize up to 100 rows (default 10) and download the table in .csv files.

This tutorial is described with the MEGAN output files (Megan WGS output format and Megan metadata) available on the Upload files page.

Data upload and summary tables

In the "Upload files" tab, user must select the file format and click the browse button to upload the count and metadata. Then, select the taxonomy level and click the submit button to analyze the metagenomic data. The summary page will provide the following details.

Summary of count data and metadata, Taxonomy tables, metadata tables, No. of conditions, count in samples (**Figure S3 – S7**).

The screenshot shows the MetaDAVis interface divided into two main sections: 'Input files & format (txt, tsv, csv)' on the left and 'Summary tables' on the right.

Input files & format (txt, tsv, csv)

- Upload files:** A red box highlights the 'Select Input format' dropdown set to 'Megan'. Below it are 'Upload count file' and 'Upload meta-data' fields, both showing 'Megan_WGS_output.tsv' and 'Megan_WGS_metadata.tsv' respectively, with 'Upload complete' status.
- Fields separated by:** Two dropdown menus are shown, both set to 'tab'.
- Choose the level to display:** A red box highlights the 'Family' radio button selected from options: Kingdom, Phylum, Class, Order, Family, Genus, Species.
- Select levels for analysis:** A red box highlights the 'Submit' button.

Summary tables

- Summary:** Shows 'Number of OTUs' and 'Summary of count data and metadata'.
- Taxonomy table:** Shows 'There are 170 bacterial taxa at the Family level.'
- Metadata table:** Shows 'Metadata' and 'Number of CD: 88, Number of HC: 21, Number of UC: 48'.
- No. of Conditions:** Shows 'Counts in samples'.

A red box highlights the 'Download example data' link.

Figure S3: Data upload page and summary table display.

Summary Taxonomy table Metadata table No. of conditions Counts in samples

Display the taxonomy counts for each samples

Show 10 entries Search:

	SRR5650021	SRR5650022	SRR5650023	SRR5650024	SRR5650025	SRR5650026	SRR5650027	SRR5650028	SRR5650029
f__(Actinobacteria <phylum>)	1	1	2	2	1	4	21	10	6
f__(Actinobacteria)	5	7	9	5	3	7	30	8	5
f__(Alphaproteobacteria)	0	0	0	0	0	0	0	0	0
f__(Archaea)	0	0	0	0	0	0	0	0	0
f__(Bacillales)	0	0	0	5	7	0	0	0	0
f__(Bacilli)	2	6	9	3	3	2	4	2	2
f__(Bacteria)	906	1066	1021	1030	996	817	881	878	936
f__(Bacteroidales bacterium 43_8)	0	0	0	0	0	0	0	0	0
f__(Bacteroidales)	1318	863	837	743	844	1229	328	1254	880
f__(Bacteroidetes)	109	86	90	81	82	116	31	92	83

Showing 1 to 10 of 170 entries Previous 1 2 3 4 5 ... 17 Next

[Download as csv](#)

Figure S4: Summary of selected taxonomy table with their counts for each sample

Summary Taxonomy table Metadata table No. of Conditions Counts in samples

Display the metadata file

Show 10 entries Search:

	Condition
SRR5650021	UC
SRR5650022	HC
SRR5650023	HC
SRR5650024	HC
SRR5650025	HC
SRR5650026	HC
SRR5650027	HC
SRR5650028	HC
SRR5650029	HC
SRR5650030	HC

Showing 1 to 10 of 157 entries Previous 1 2 3 4 5 ... 16 Next

[Download as csv](#)

Figure S5: Summary of metadata table

Summary	Taxonomy table	Metadata table	No. of conditions	Counts in samples
Display the number of condition based on your metadata				
Show 10 entries				
Condition				
1	UC			
2	HC			
3	CD			
Showing 1 to 3 of 3 entries				
Download as csv				
Previous 1 Next				

Figure S6: Display the no. of conditions based on the metadata

Summary	Taxonomy table	Metadata table	No. of conditions	Counts in samples
Display the total number of counts in each samples				
Show 10 entries				
Samples				Total_counts
1	SRR5650021			12054
2	SRR5650022			12066
3	SRR5650023			12078
4	SRR5650024			12068
5	SRR5650025			12079
6	SRR5650026			12057
7	SRR5650027			12078
8	SRR5650028			12076
9	SRR5650029			12079
10	SRR5650030			12071
Showing 1 to 10 of 157 entries				
Download as csv				
Previous 1 2 3 4 5 ... 16 Next				

Figure S7: Display the total number of counts based on the count file

After uploading the count data, metadata file and pre-selected level, inputs will be automatically saved for accessing the distribution, diversity, dimension reduction, correlation, heatmap and differential abundance tabs.

Distribution Analysis

Under the distribution tab, user can visualize their taxa (two to 100) by grouped (based on the Condition) or individual (based on the individual samples).

Distribution based on groups:

Users need to select the plot types, number of bacterial taxa to display, the image output format, and click the submit button to visualize the plot (**Figure S8 – S10**).

The screenshot shows the 'Distribution' tab selected in the top navigation bar. A dropdown menu is open under 'Distribution', with 'Group' and 'Individual' options visible. The main panel displays 'Distribution of top bacterial taxa (groups)' and a message 'Selected input: No data selected! please load the data first'. Below this, a section titled 'Types of plot' contains three radio button options: 'Abundance (%) - stacked bar' (selected), 'Abundance value - stacked bar', and 'Relative frequency - stacked bar'. To the right of this section, a red box highlights the 'Different types of plots' label. Further down, there's a field 'Number of top bacterial taxa (Max = 100)' with the value '15' entered, and another red box highlights this field with the annotation 'Select upto two to 100 taxa's (default is 15)'. At the bottom, there's an 'Output image format' dropdown set to 'JPG', and a 'Submit' button. Red arrows point from the annotations to the corresponding UI elements.

Figure S8: Input selection for group distribution

Downloading images in various formats

Users can download figures with preferred dimensions up to 49 inches of height and width in multiple image formats (JPG, TIFF, PDF, SVG, BMP, EPS, and PS) to obtain in publication quality and comply with journal requirements (dpi resolution: 72 to 1000). This menu was incorporated into all the tabs, which contain figures.

The screenshot shows three input fields for download settings: 'Figure height (upto 49 inces)' with value '8', 'Figure width (upto 49 inces)' with value '8', and 'Figure resolution (dpi:72 to 300)' with value '300'. Below these fields is a 'Download plot' button with a file icon.

Figure S9: Download the plot in preferred dimensions for publication in multiple image formats.

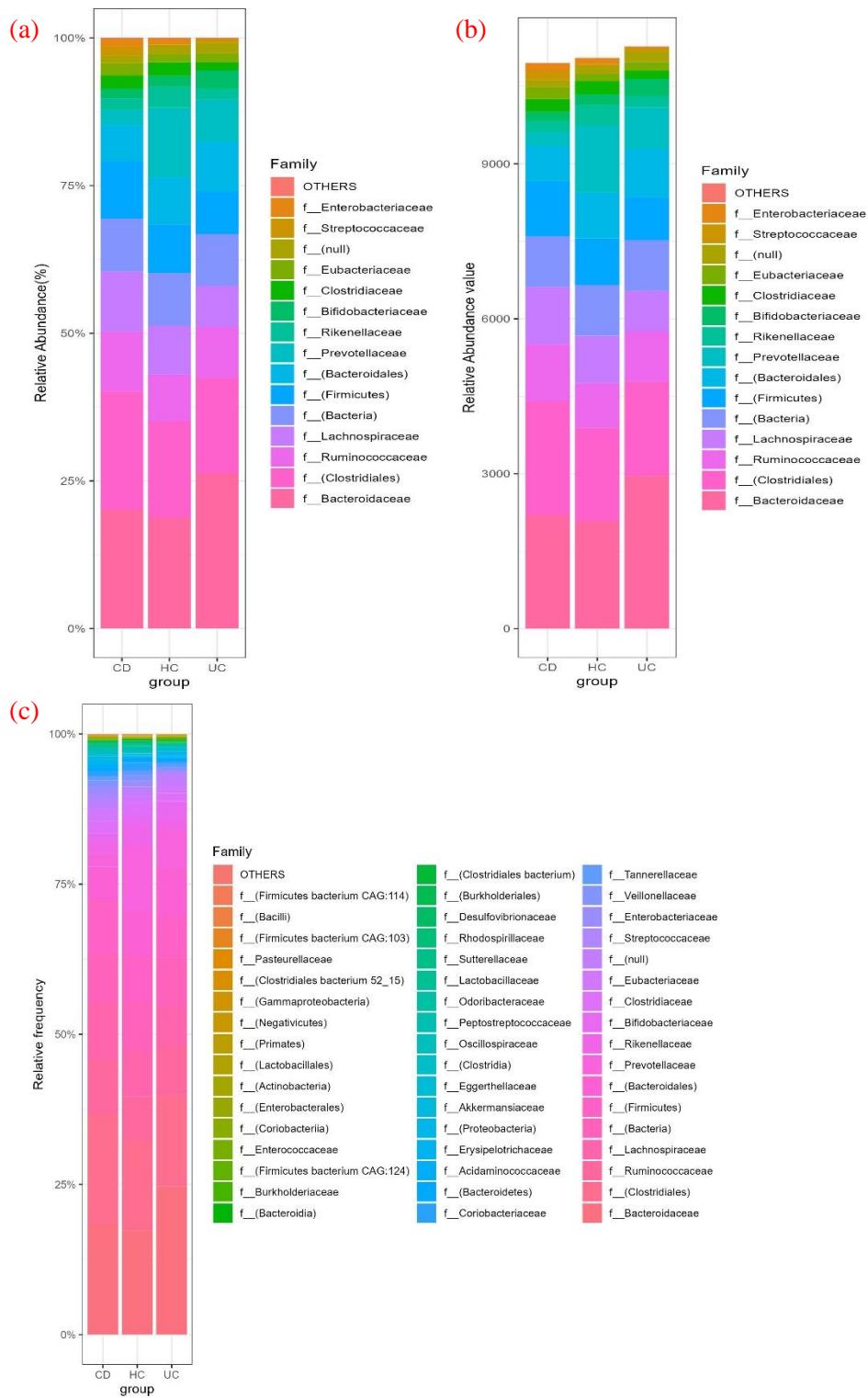


Figure S10: displays the top a) 15 relative abundance percentages, b) 15 relative abundance values, c) 50 relative abundance frequencies grouped by the list of conditions in metadata

Distribution based on individual:

Users need to select the plot types, number of bacterial taxa to display, image output format, and click the submit button to visualize the plot (**Figure S11 – S12**).

Distribution of top bacterial taxa (samples)

Selected input
Megan_WGS_output.tsv

Types of plot

Abundance (%) - stacked bar
 Abundance value - stacked bar
 Relative frequency - stacked bar

Different types of plots

Number of top bacterial taxa (Max = 100)
15

Output image format
JPG

Submit

Figure S11: Input selection for individual distribution

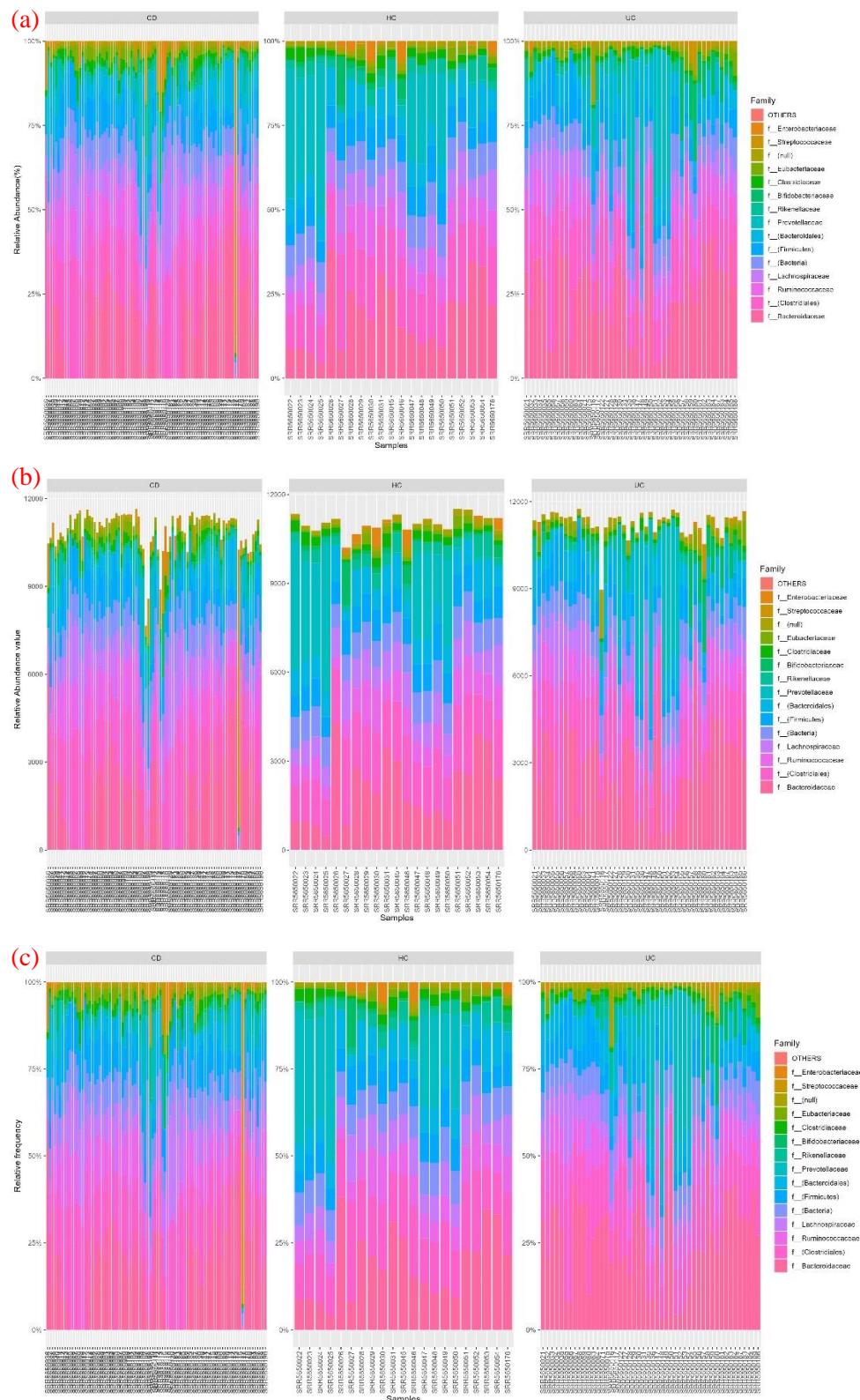


Figure S12: displays the top 15 a) relative abundance percentages, b) relative abundance values, c) relative abundance frequencies for the given samples in metadata

Diversity Analysis

Under the diversity tab, there are two subsections (**Figure S13**): 1. Alpha and 2. Beta diversity.

Alpha diversity

Alpha diversity was calculated by the phyloseq package (McMurdie and Holmes, 2013). Users can visualize the alpha diversity by choosing any one of the methods (**Figure S13**), such as Observed, Chao1 (**Figure S14 a**), ACE, Shannon (**Figure S14 b**), Simpson, Inverse Simpson, Fisher or All_combined (combined all the listed methods) (**Figure S14 c**). Users can also get the diversity plot with p-values (either values or *) (**Figure S14 d**), using Wilcoxon tests (from microbiomeutilities package) (Shetty and Lahti, 2022) considering each pair of groups. Once the output plot types and the image format are selected, then click the submit button to calculate diversity. Users can also get the alpha diversity for each sample by clicking the summary table tab (**Figure S15**).

MetaDAVis Upload files Distribution ▾ **Diversity ▾** Dimension reduction ▾ Correlation ▾ Heatmap Differential abundance ▾

Alpha **Subsection** **Beta**

Alpha diversity

Selected input: Megan_WGS_output.tsv

Select Method: Observed

Wilcoxon test

Yes (shows Pvalue) → (0, 0.0001, 0.001, 0.01, 0.05, Inf)
 No
 Show * → ("****", "****", "***", "**", "ns")

Types of plot

Box plot **Different types of plot**
 Violin plot

Output image format: JPG

Submit

Observed, Chao1, ACE, Shannon, Simpson, Inverse Simpson, Fisher or All_combined (combined all the listed methods)

Select file formats (JPG, TIFF, PDF, SVG, BMP, EPS, and PS)

Figure S13: Input selection for alpha diversity

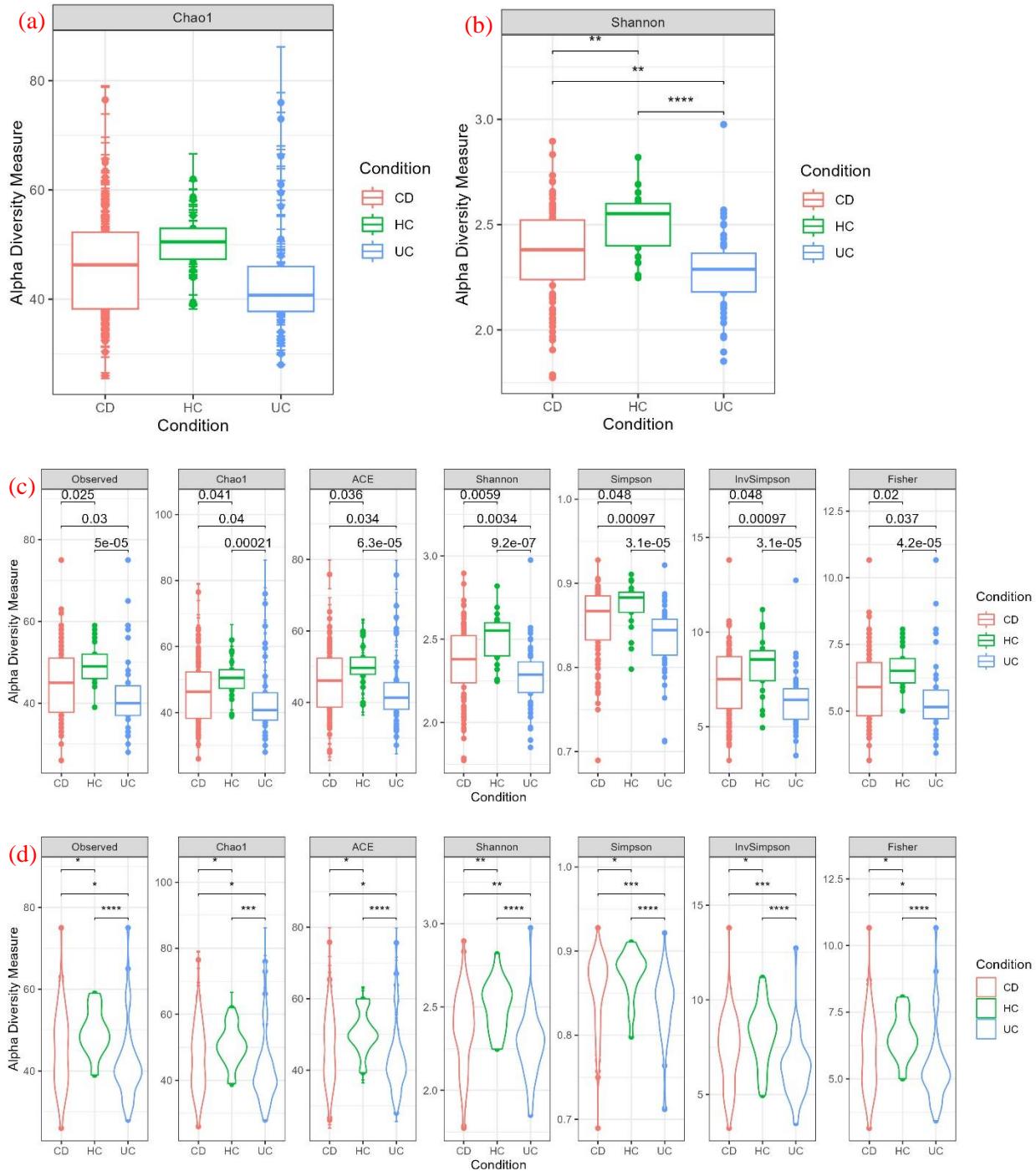


Figure S14. Box plots show the alpha diversity, which is calculated based on the a) chao1 without p-value, b) Shannon index with p-value (shows: "****", "***", "**", "*", "ns"), c) using all the methods with p-value, d) Plotted in violin with p-value (shows: "****", "***", "**", "*", "ns").

[Alpha diversity plot](#)[Summary Table](#)

Result - alpha diversity estimates for each metagenome

Show **10** entries

	Observed	Chao1	se.chao1	ACE	se.ACE	Shannon	Simpson	InvSimpson	Fisher
SRR5650036	55	55.25	0.74	55.6	3.49	2.21	0.76	4.12	7.44
SRR5650037	51	51	0.25	51.24	3.44	2.6	0.89	9.03	6.82
SRR5650038	42	42	0.16	42.28	2.81	2.39	0.85	6.64	5.45
SRR5650039	59	62.33	4.12	61.13	3.73	2.73	0.91	10.59	8.07
SRR5650040	47	48	2.33	47.42	3.19	2.6	0.89	9.39	6.2
SRR5650041	48	48	0.12	48.27	3.38	2.52	0.89	8.98	6.36
SRR5650042	58	59.5	2.23	59.86	3.39	2.49	0.87	7.87	7.91
SRR5650043	59	59.33	0.92	59.67	3.49	2.6	0.89	8.72	8.07
SRR5650044	56	59	4.17	57.73	3.56	2.36	0.85	6.7	7.6
SRR5650065	32	32.5	1.29	32.72	2.77	1.9	0.78	4.56	3.99

Showing 1 to 10 of 157 entries

Previous [1](#) [2](#) [3](#) [4](#) [5](#) ... [16](#) Next [Download as csv](#)

Figure S15: Summary table of alpha diversity for each sample

Beta diversity

Beta diversity was calculated based on phyloseq (unlist(distanceMethodList)) (McMurdie and Holmes, 2013) and vegan (Oksanen et al., 2017) packages. Users can visualize the beta diversity by choosing any one of the methods. In our application, we have integrated 42 distance metrics. Users can use any one of the following methods such as (bray, jaccard, manhattan, euclidean, canberra, kulczynski, gower, altGower, morisita, horn, mountford, raup, binomial, chao, cao, w, -1, c, wb, r, I, e, t, me, j, sor, m, -2, co, cc, g, -3, l, 19, hk, rlb, sim, gl, z, maximum, binary and minkowski) (**Figure S16**). In addition, we have incorporated six different orientation methods using the vegan package, such as (PCoA, NMDS, DCA, CCA, RDA, and MDS) (**Figure S16**). The result will be displayed in the box (**Figure S17 a**) and orientation plot (**Figure S17 b**) with the summary table (**Figure S18**), which contains distance matrices between all the samples.

Beta diversity

Selected input
Megan_WGS_output.tsv

Select diversity metrics
bray-curtis

Select method
PCoA

Output image format
JPG

Submit

Select distance metrics (bray, jaccard, manhattan, euclidean, canberra, kulczynski, gower, altGower, morisita, horn, mountford, raup, binomial, chao, cao, w, -1, c, wb, r, I, e, t, me, j, sor, m, -2, co, cc, g, -3, l, 19, hk, rlb, sim, gl, z, maximum, binary and minkowski)

► Select orientation methods (PCoA, NMDS, DCA, CCA, RDA, and MDS) (JPG, TIFF, PDF, SVG, BMP, EPS, and PS)

► Select file formats (JPG, TIFF, PDF, SVG, BMP, EPS, and PS)

Figure S16: Input selection for beta diversity

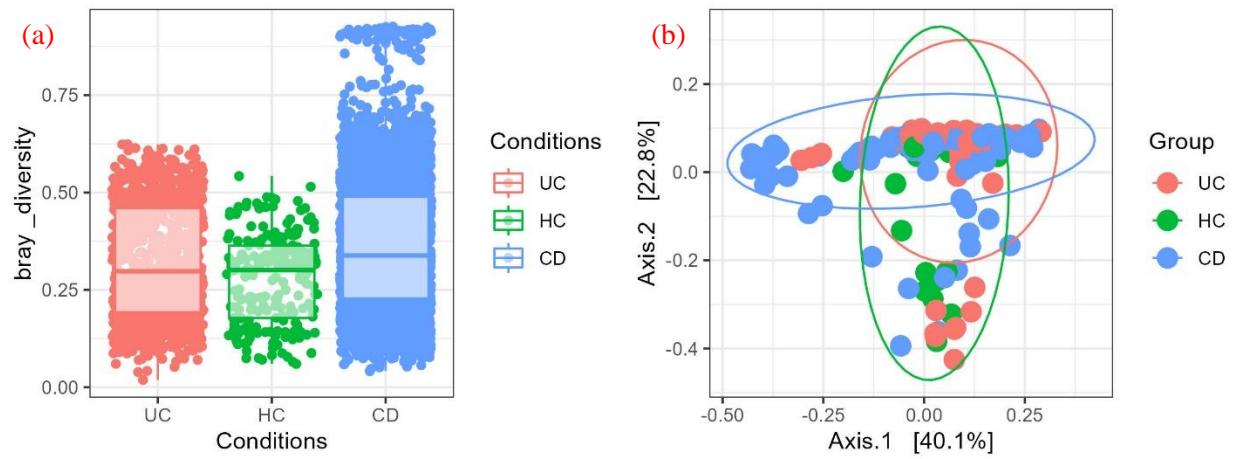


Figure S17 a) The diversity metrics (bray) were plotted in the box plot b) the diversity matrices (bray) with PCoA orientation were plotted.

Beta diversity Plot Summary Table

Result - distance between all the samples

Show 10 entries Search:

SRR5650036	SRR5650037	SRR5650038	SRR5650039	SRR5650040	SRR5650041	SRR5650042	SRR5650043	SRR5650044	SRR5650045
SRR5650036	0	0.56	0.35	0.47	0.52	0.49	0.67	0.64	0.71
SRR5650037	0.56	0	0.3	0.2	0.32	0.14	0.4	0.38	0.44
SRR5650038	0.35	0.3	0	0.23	0.22	0.21	0.41	0.39	0.45
SRR5650039	0.47	0.2	0.23	0	0.26	0.16	0.37	0.35	0.42
SRR5650040	0.52	0.32	0.22	0.26	0	0.21	0.35	0.32	0.38
SRR5650041	0.49	0.14	0.21	0.16	0.21	0	0.43	0.4	0.46
SRR5650042	0.67	0.4	0.41	0.37	0.35	0.43	0	0.1	0.1
SRR5650043	0.64	0.38	0.39	0.35	0.32	0.4	0.1	0	0.17
SRR5650044	0.71	0.44	0.45	0.42	0.38	0.46	0.1	0.17	0
SRR5650045	0.77	0.56	0.53	0.54	0.5	0.56	0.34	0.39	0.25

Showing 1 to 10 of 157 entries

Previous **1** 2 3 4 5 ... 16 Next

[Download as csv](#)

Figure S18: Summary table of beta diversity

Dimension reduction

Under this tab, there are three subsections (**Figure S19**) 1. Principal Component Analysis (PCA) 2D and 3D, 2. t-distributed Stochastic Neighbor Embedding (t-SNE) and 3. Uniform Manifold Approximation and Projection for Dimension Reduction (UMAP).

PCA-2D & 3D

The ggfortify (Tang et al., 2016) was used to plot the PCA-2D. Users must select the text label and its size, frame, and output image format and click the submit button (**Figure S19**). The output will be displayed in a PCA-2D plot (**Figure S20**) with a summary table (**Figure S21**). For the PCA-3D plot, we used plotly to create the plot. In this section, the user must click the submit button to see the PCA plot in 3D. The plotly has its function to export the image in png format (**Figure S22**). The PCA summary table contains the PC1, PC2 and PC3 coordinates (**Figure S23**).

The screenshot shows the 'Dimension reduction' dropdown menu open, revealing four options: PCA-2D (selected), PCA-3D, t-SNE, and UMAP. The 'PCA-2D' section contains the following input fields:

- Selected input:** Megan_WGS_output.tsv
- Label:** FALSE (with a note: → If true it will display sample labels)
- Label size:** 3 (with a note: → Sample label size)
- Frame:** FALSE (with a note: → If true display circular frames)
- Output image format:** JPG (with a note: → Select file formats (JPG, TIFF, PDF, SVG, BMP, EPS, and PS))

On the right, under 'Principal Component Analysis', there are two tabs: 'PCA 2D Plot' (selected) and 'Summary Table'.

Figure S19: Input selection for PCA plot.

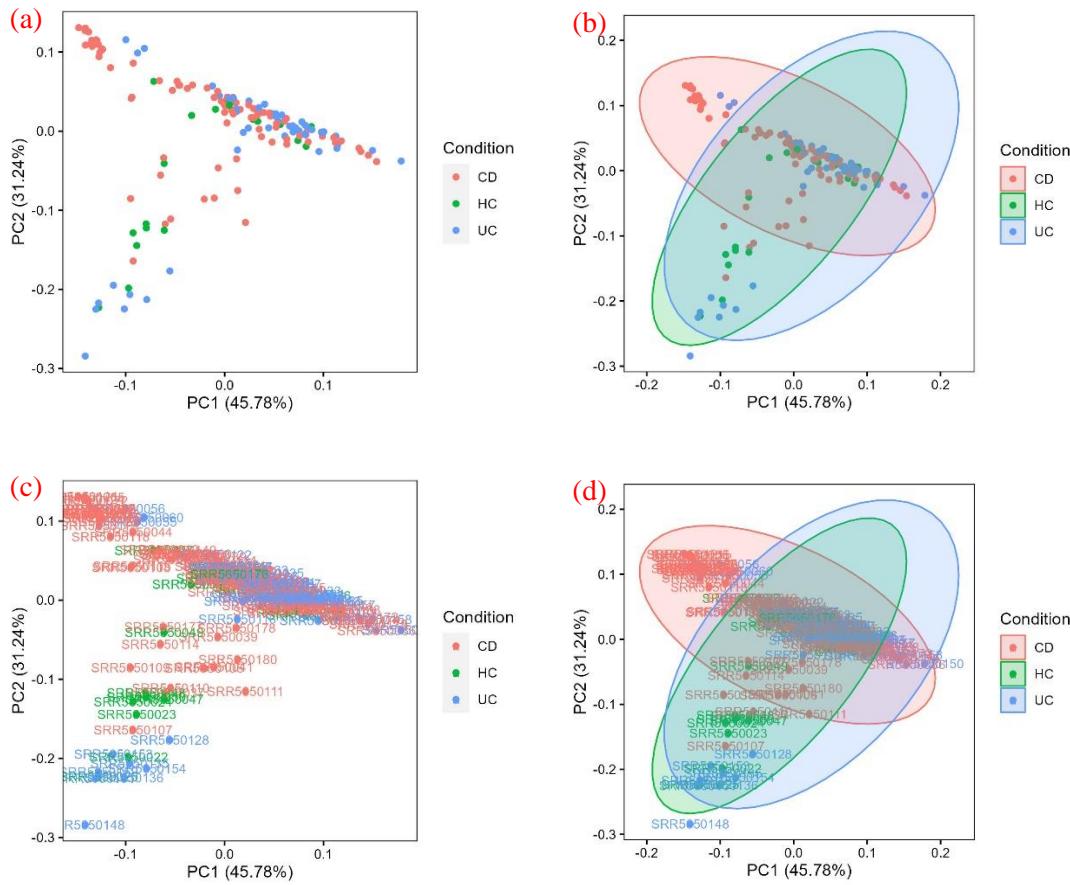


Figure S20. Displays PCA-2D plot with a) no labels and no frame, b) no labels and with a frame, c) with labels and no frame, d) with labels and frame.

PCA 2D Plot Summary Table

Show 10 entries Search:

	PC1	PC2	Condition
SRR5650036	0.15	-0.04	CD
SRR5650037	-0.06	-0.12	CD
SRR5650038	0.06	-0.02	CD
SRR5650039	-0.01	-0.05	CD
SRR5650040	0	0.01	CD
SRR5650041	-0.01	-0.08	CD
SRR5650042	-0.07	0.06	CD
SRR5650043	-0.05	0.05	CD
SRR5650044	-0.09	0.09	CD
SRR5650065	-0.14	0.13	CD

Showing 1 to 10 of 157 entries Previous 1 2 3 4 5 ... 16 Next

[Download as csv](#)

Figure S21. Summary table for PCA. Each sample coordinate position was shown in PC1 and PC2.

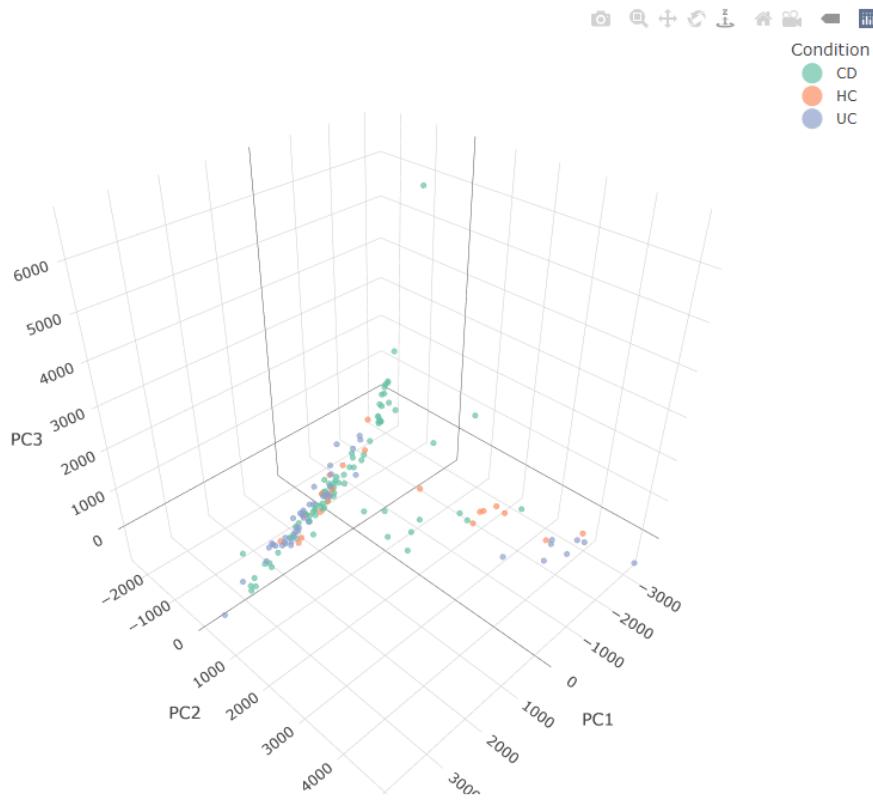


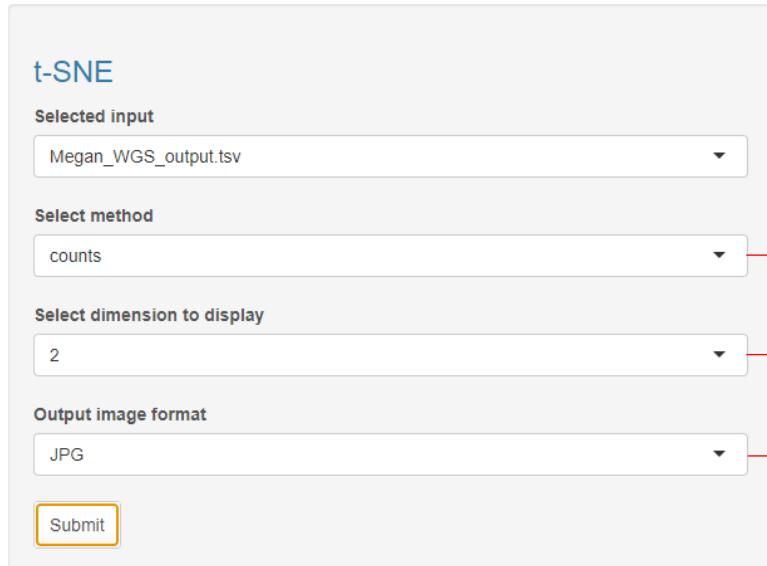
Figure S22. Displays PCA-3D plot. Plotly provides a default menu option in the top-right corner to export the plot.

PCA 3D Plot		Summary Table	
Show	10 entries	Search:	
Samples		PC1	PC2
1	SRR5650021	1441.84	-117.4
2	SRR5650022	-2127.69	3591.86
3	SRR5650023	-1947.49	2618.18
4	SRR5650024	-2028.39	2327.14
5	SRR5650025	-2789.16	4038.58
6	SRR5650026	1924.27	-104.94
7	SRR5650027	-1569.94	-1140.54
8	SRR5650028	623.74	-275.16
9	SRR5650029	179.38	-453.86
10	SRR5650030	-200.31	-501.6
Showing 1 to 10 of 157 entries		Previous	1 2 3 4 5 ... 16 Next
Download as csv			

Figure S23. Summary table for PCA-3D. Each sample coordinate position was shown in PC1, PC2 and PC3.

t-SNE

The t-SNE was plotted using a scater package (McCarthy et al., 2017). We have incorporated six methods from the scater to plot the t-SNE: counts, rclr, hellinger, pa, rank, and relabundance in two and three dimension orientations (**Figure S24**). After selecting methods, orientation and output image format, click submit to visualize the t-SNE plot (**Figure S25 a & b**) and their summary tables (**Figure S26 a & b**).



The figure shows a user interface for generating a t-SNE plot. It includes fields for 'Selected input' (Megan_WGS_output.tsv), 'Select method' (counts), 'Select dimension to display' (2 or 3), 'Output image format' (JPG), and a 'Submit' button. Red arrows point from the right side to each of these four fields, indicating they are the ones used for input selection.

- Select method (counts, rclr, hellinger, pa, rank, and relabundance)
- Select dimension (2 or 3)
- Select file formats (JPG, TIFF, PDF, SVG, BMP, EPS, and PS)

Figure S24: Input selection for t-SNE plot

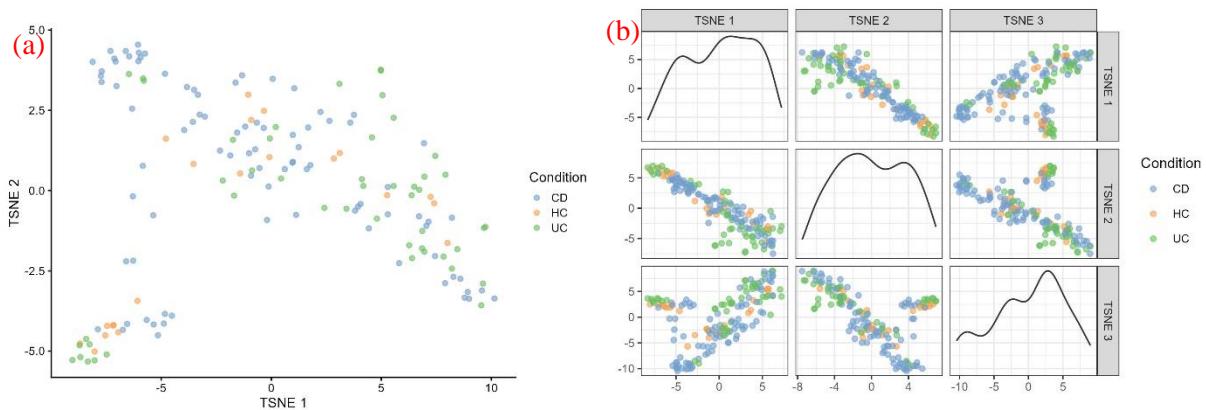


Figure S25. The t-SNE plot in a) two dimensions, b) three dimensions

(a) t-SNE Plot

Show 10 entries

Summary Table

Search:

	X	Y	colour_by	order_by
SRR5650036	7.45	-6.26	CD	1
SRR5650037	-7.02	1.91	CD	2
SRR5650038	4.32	-3.58	CD	3
SRR5650039	-5.42	1.04	CD	4
SRR5650040	-2.13	0.77	CD	5
SRR5650041	-5.96	1.17	CD	6
SRR5650042	0.37	4.6	CD	7
SRR5650043	0.43	4.33	CD	8
SRR5650044	0.16	5.47	CD	9
SRR5650045	-0.17	7.18	CD	10

Showing 1 to 10 of 157 entries

Previous 1 2 3 4 5 ... 16 Next

[Download as csv](#)

(b) t-SNE Plot

Show 10 entries

Summary Table

Search:

xvar	yvar	x	y	X1	X2	X3	colour_by	order_by
SRR5650036	TSNE 1	TSNE 2	2.91	11.81	11.81	2.91	2.98	CD
SRR5650037	TSNE 1	TSNE 2	4.93	-6.68	-6.68	4.93	1.96	CD
SRR5650038	TSNE 1	TSNE 2	1.26	6.48	6.48	1.26	2.73	CD
SRR5650039	TSNE 1	TSNE 2	3.84	-4.39	-4.39	3.84	1.22	CD
SRR5650040	TSNE 1	TSNE 2	0.62	-1.56	-1.56	0.62	-0.78	CD
SRR5650041	TSNE 1	TSNE 2	4.43	-4.98	-4.98	4.43	1.57	CD
SRR5650042	TSNE 1	TSNE 2	-4.52	-4.36	-4.36	-4.52	-2.53	CD
SRR5650043	TSNE 1	TSNE 2	-4.21	-3.96	-3.96	-4.21	-2.76	CD
SRR5650044	TSNE 1	TSNE 2	-4.98	-5.48	-5.48	-4.98	-2.4	CD
SRR5650045	TSNE 1	TSNE 2	-6.12	-7.81	-7.81	-6.12	-2.25	CD

Showing 1 to 10 of 942 entries

Previous 1 2 3 4 5 ... 95 Next

[Download as csv](#)

Figure S26. Summary table for t-SNE a) two dimensions, b) three dimensions

UMAP

The UMAP was incorporated using two packages, scater (McCarthy et al., 2017) and bluster (Lun, 2022). We have incorporated six methods from the scater to plot the UMAP: counts, rclr, hellinger, pa, rank, and relabundance. The cluster package was used to plot the graph using cluster-based with the selected k-value (**Figure 27**). After selecting methods, k-value and output image format, click submit to visualize the UMAP plot (**Figure S28 a & b**) and their summary tables (**Figure S29 a & b**).

UMAP

Selected input
Megan_WGS_output.tsv

Select method
counts → Select method (counts, rclr, hellinger, pa, rank, and relabundance)

Select k value (for graph construction)
5 → K-value (2 to 15)

Output image format
JPG → Select file formats (JPG, TIFF, PDF, SVG, BMP, EPS, and PS)

Figure S27: Input selection UMAP plot

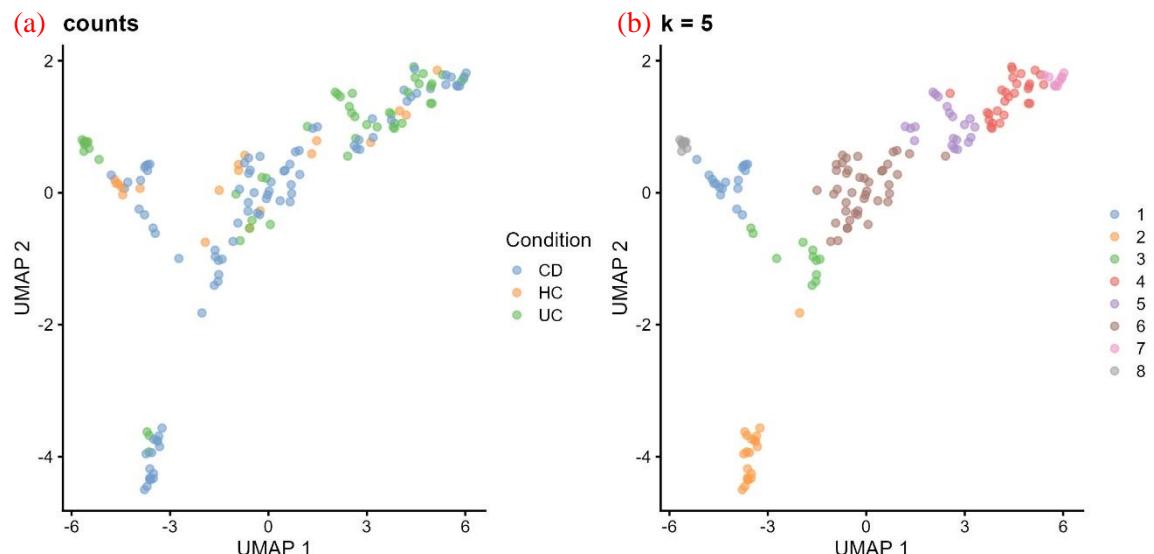


Figure S28. a) UMAP plot based on the selected method (counts), colored by Condition from metadata, b) Plot colored based on cluster-based using the selected K-value

(a) JMAP Plot Summary Table based on condition Summary Table based on cluster

Show 10 entries Search:

	X	Y	colour_by	order_by
SRR5650036	-0.7	-4.34	CD	1
SRR5650037	5.76	-0.31	CD	2
SRR5650038	-1.8	-3.04	CD	3
SRR5650039	4.94	-0.37	CD	4
SRR5650040	-1.4	1.14	CD	5
SRR5650041	5.11	-0.3	CD	6
SRR5650042	-1	2.89	CD	7
SRR5650043	-1.07	2.79	CD	8
SRR5650044	1.3	4.93	CD	9
SRR5650065	1.61	5.45	CD	10

Showing 1 to 10 of 157 entries Previous 1 2 3 4 5 ... 16 Next

Download as csv

(b) UMAP Plot Summary Table based on condition Summary Table based on cluster

Show 10 entries Search:

	X	Y	colour_by	order_by
SRR5650036	-0.7	-4.34	7	1
SRR5650037	5.76	-0.31	1	2
SRR5650038	-1.8	-3.04	4	3
SRR5650039	4.94	-0.37	1	4
SRR5650040	-1.4	1.14	6	5
SRR5650041	5.11	-0.3	1	6
SRR5650042	-1	2.89	3	7
SRR5650043	-1.07	2.79	3	8
SRR5650044	1.3	4.93	2	9
SRR5650065	1.61	5.45	2	10

Showing 1 to 10 of 157 entries Previous 1 2 3 4 5 ... 16 Next

Download as csv

Figure S29. Summary table for UMAP based on a) conditions from metadata using the selected method (counts), b) Cluster-based using K-value

Correlation analysis

Under this tab are two subsections: 1) Taxa-based and 2) Sample-based correlation (**Figure S30**).

Taxa-based correlation

The taxa-based correlation plot was incorporated using the GGally (Schloerke et al., 2022) package with the ggcrr function to call three different methods: pearson, kendall and spearman. Once the method, label size and output format are selected, click submit (**Figure S30**) to visualize the taxa plot (**Figure S31**) and summary table (**Figure S32**) for the selected taxonomy on the file upload page. We have used ggpubr (Kassambara 2022) to do our graphics.

MetaDAVis Upload files Distribution ▾ Diversity ▾ Dimension reduction ▾ Correlation ▾ Heatmap Differential abundance ▾

Subsection

Compute correlation between taxa

Selected input: Megan_WGS_output.tsv

Correlation methods:

- pearson
- kendall
- spearman

Methods used for analysis

Label size: 3

Output image format: JPG

Submit

→ Enter the font size to display the sample labels

→ Select file formats (JPG, TIFF, PDF, SVG, BMP, EPS, and PS)

Figure S30. Input selection for taxa-based correlation analysis

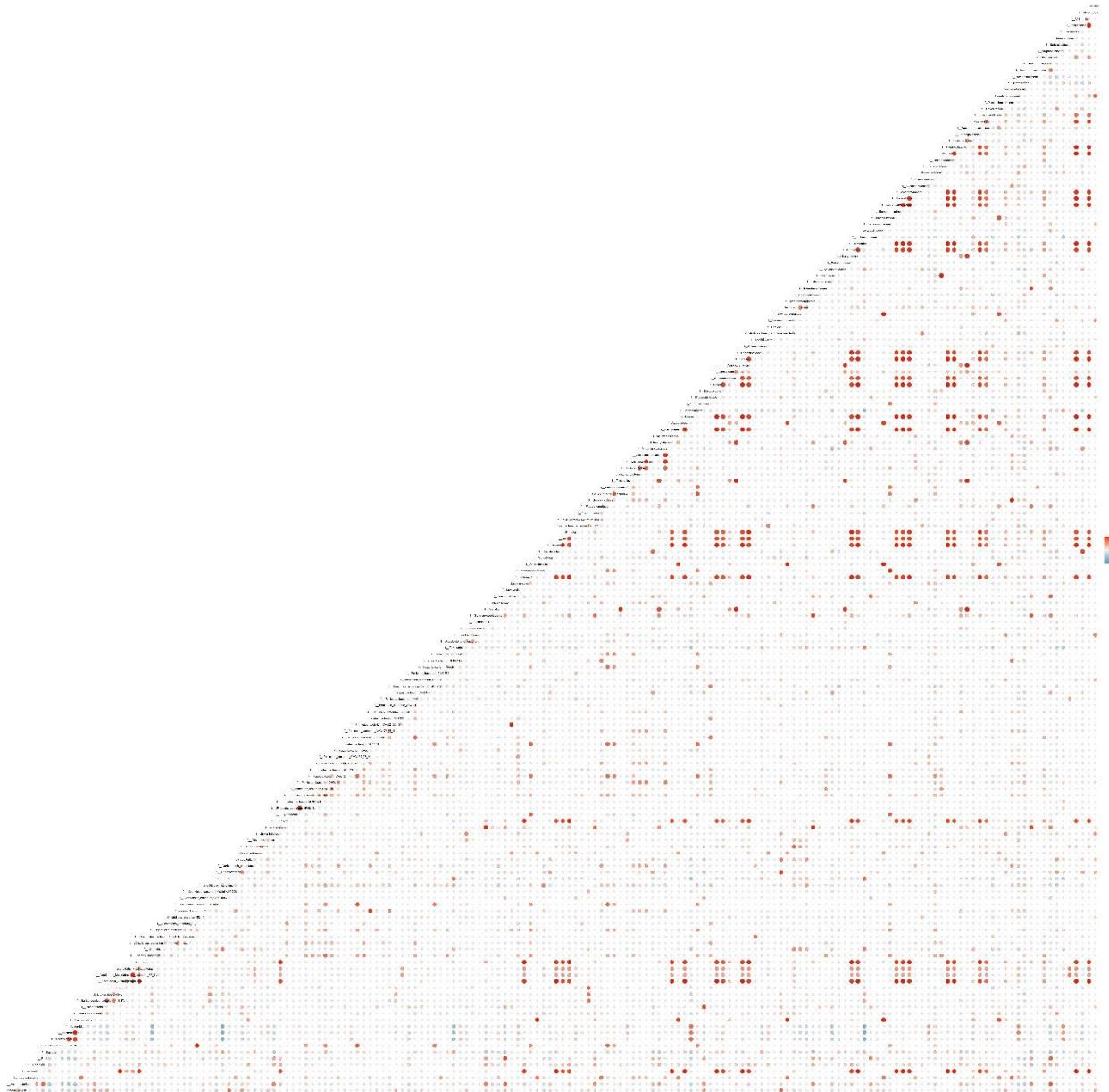


Figure S31. Taxa-based correlation plot using Pearson method

Correlation plot Summary Table

Show 10 entries Search:

x	y	coefficient	label
2 f__Actinobacteria.	f__Actinobacteria_phylum..	0.41	0.4
3 f__Alphaproteobacteria.	f__Actinobacteria_phylum..	0.06	0.1
4 f__Archaea.	f__Actinobacteria_phylum..	-0.05	-0.1
5 f__Bacillales.	f__Actinobacteria_phylum..	0.04	0
6 f__Bacilli.	f__Actinobacteria_phylum..	0.26	0.3
7 f__Bacteria.	f__Actinobacteria_phylum..	-0.22	-0.2
8 f__Bacteroidales_bacterium_43_8.	f__Actinobacteria_phylum..	-0.01	0
9 f__Bacteroidales.	f__Actinobacteria_phylum..	-0.26	-0.3
10 f__Bacteroidetes.	f__Actinobacteria_phylum..	-0.27	-0.3
11 f__Bacteroidia.	f__Actinobacteria_phylum..	-0.25	-0.3

Showing 1 to 10 of 14,365 entries

Previous 1 2 3 4 5 ... 1,437 Next

[Download as csv](#)

Figure S32. Taxa-based correlation table using Pearson method

Sample-based correlation

The sample-based correlation plot was incorporated with a similar method used for taxa-based correlation. Once the method, label size and output format are selected, click submit (**Figure S33**). It will display the correlation plot for samples provided in the metadata (**Figure S34**) and the summary table (**Figure S35**).

Compute correlation between samples

Selected input
Megan_WGS_output.tsv

Correlation methods
 pearson
 kendall
 spearman

Methods used for analysis

Label size
3

Output image format
JPG

Submit

→ Enter the font size to display the sample labels

→ Select file formats (JPG, TIFF, PDF, SVG, BMP, EPS, and PS)

Figure S33. Input selection for sample-based correlation analysis

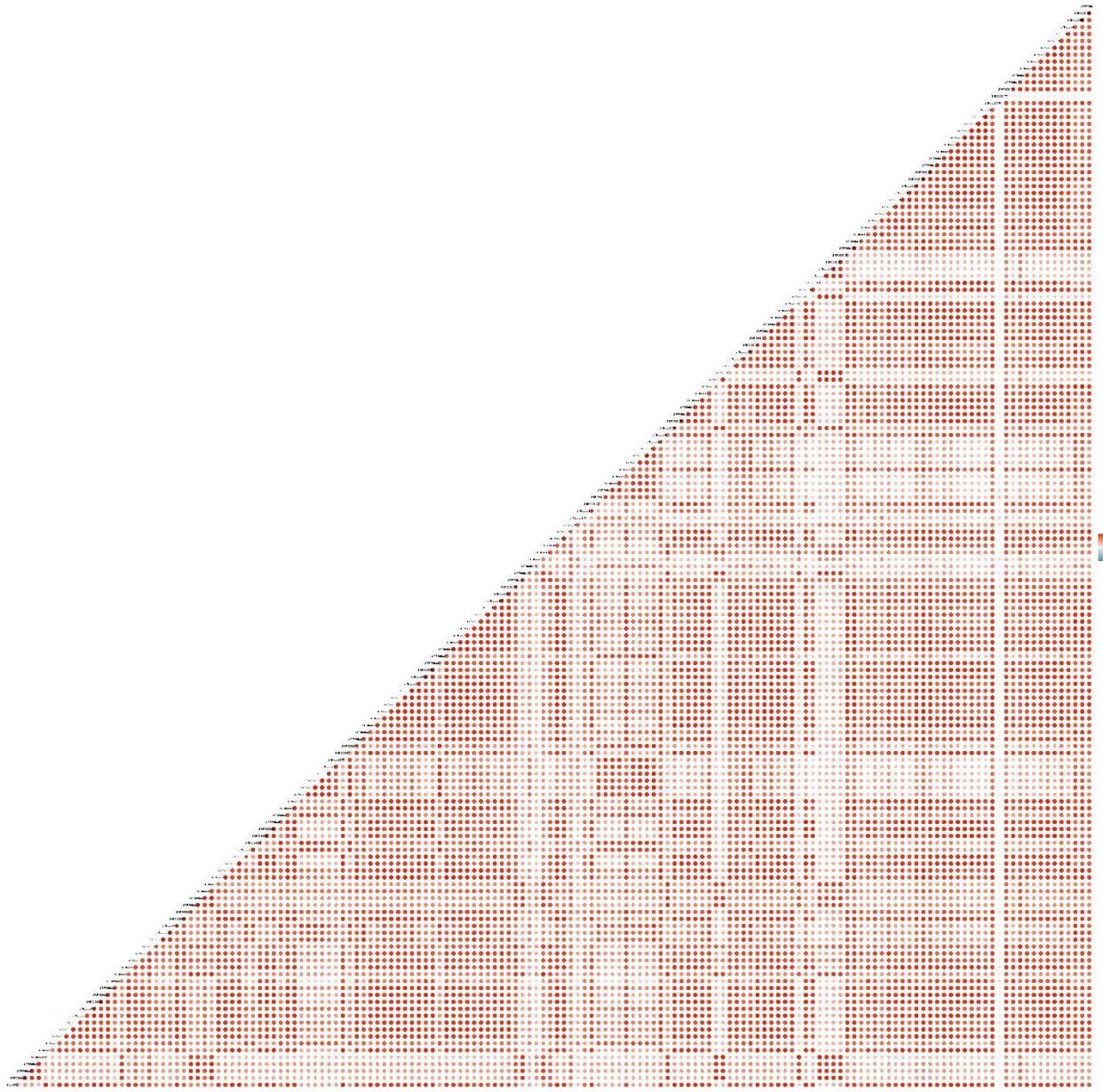


Figure S34. Sample-based correlation plot using Pearson method

Correlation plot Summary Table

Show 10 entries Search:

x	y	coefficient	label	
2	SRR5650037	SRR5650036	0.44	0.4
3	SRR5650038	SRR5650036	0.89	0.9
4	SRR5650039	SRR5650036	0.73	0.7
5	SRR5650040	SRR5650036	0.67	0.7
6	SRR5650041	SRR5650036	0.67	0.7
7	SRR5650042	SRR5650036	0.33	0.3
8	SRR5650043	SRR5650036	0.38	0.4
9	SRR5650044	SRR5650036	0.25	0.2
10	SRR5650045	SRR5650036	0.15	0.1
11	SRR5650046	SRR5650036	0.15	0.1

Showing 1 to 10 of 12,246 entries

Previous **1** 2 3 4 5 ... 1,225 Next

[Download as csv](#)

Figure S35. Sample-based correlation table using Pearson method

Heatmap analysis

The heatmap was generated with ComplexHeatmap (Gu, 2022), scales (Wickham and Seidel, 2022) and ggplotify. The user modifies the heatmap according to their needs by selecting the label names, text size, and cladogram (**Figure S36**). Then select the output format and click the submit button to visualize the heatmap (**Figure S37**).

The screenshot shows a user interface for generating a heatmap. At the top, there is a navigation bar with tabs: MetaDAVis, Upload files, Distribution ▾, Diversity ▾, Dimension reduction ▾, Correlation ▾, Heatmap (which is highlighted in blue), and Differential abundance ▾. Below the navigation bar is a form titled "Heatmap - relative abundance". The form contains the following settings:

- Selected input:** Megan_WGS_output.tsv
- Show row names:** TRUE (with a red arrow pointing to it labeled "If true display row names")
- Row name size:** 7 (with a red arrow pointing to it labeled "Size of the row names")
- Show column names:** TRUE (with a red arrow pointing to it labeled "If true display column names")
- Column name size:** 7 (with a red arrow pointing to it labeled "Size of the column names")
- Show row cladogram:** TRUE (with a red arrow pointing to it labeled "If true display row cladogram")
- Show column cladogram:** TRUE (with a red arrow pointing to it labeled "If true display column cladogram")
- Output image format:** JPG (with a red arrow pointing to it labeled "Select file formats (JPG, TIFF, PDF, SVG, BMP, EPS, and PS)")

At the bottom left of the form is a "Submit" button.

Figure S36. Input selection for heatmap analysis

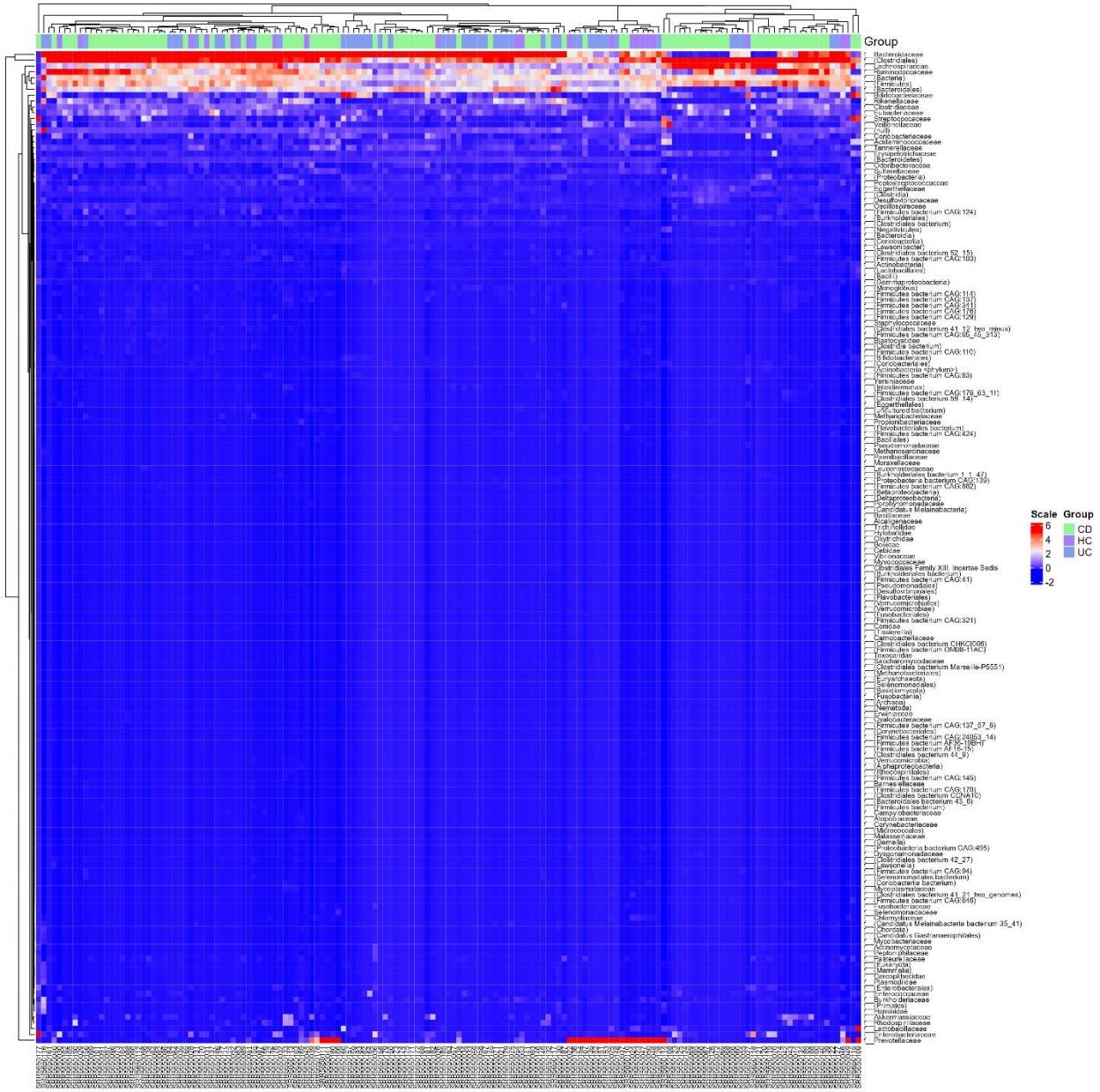


Figure S37. The heatmap for the selected taxonomy level on the upload page shows sample names in rows and family names in columns with a cladogram. Scale values represent the colors in the heatmap and groups represent the no. of conditions in the metadata file

Differential abundance analysis

In differential abundance, we have two subsections: Two group and Multiple group comparison (**Figure S38**).

Two group comparison

Two group methods analyze one set of control and case samples from the metadata. To analyze the metagenome data, we have incorporated six different methods: Wilcoxon Rank Sum test, t-test: Two sample t-test, metagenomeSeq (Paulson et al., 2013), DESeq2 (Love et al., 2016), Limma-Voom (Ritchie et al., 2015) and edgeR (Robinson et al., 2010) (**Figure S38**). For the Wilcoxon Rank Sum test (wilcox.test) and t-test (t.test) statistical analysis, we have converted the raw count value to relative frequency using the formula (**Relative Frequency = (Subgroup frequency/ Total frequency) *100**). For metagenomeSeq, DESeq2, Limma-Voom and edgeR, we have used their package algorithm to find the significant taxonomy.

Users must select two different conditions in 1 and 2 (it was a pop-up based on your metadata file, which you uploaded on the upload page). This section uses only two groups for the comparison (HC vs. CD). Then, users need to select the test correction method, either Benjamini-Hochberg FDR or P-value; also, they can adjust the FDR or P-value based on their needs (default is < 0.05). Finally, select any plot type and image format, then click the submit button (**Figure S38**) to visualize the grouped box plot (**Figure S39a**), individual box blot for each taxon (**Figure S39b**), volcano plot (**Figure S39c**) and the heatmap of significantly identified taxa (**Figure S39d**). We have similar input methods for all these six methods, and similar plots will be generated. Only the summary tables (**Figure S40**) columns will differ (**Table 2**).

MetaDAVis Upload files Distribution ▾ Diversity ▾ Dimension reduction ▾ Correlation ▾ Heatmap Differential abundance ▾

Wilcoxon Rank Sum test

Selected input
Megan_WGS_output.tsv

Select condition1
UC

Select condition2
UC

Test correction
Benjamini-Hochberg FDR

FDR or P value
0.05

Types of plot
 Grouped box plot
 Individual box plot
 Volcano plot
 Heatmap

Output image format
JPG

Submit

Result - OTUs

Download significant
[Download as csv](#)

Differential abundance

Two groups
Wilcoxon Rank Sum test (highlighted)
 t-test
 metagenomeSeq
 DESeq2
 Limma-Voom
 edgeR

Multiple groups
 Kruskal-Wallis test
 ANOVA

Subsection

Figure S38. Input selection for the Wilcoxon Rank Sum test and similar input is needed for the remaining methods

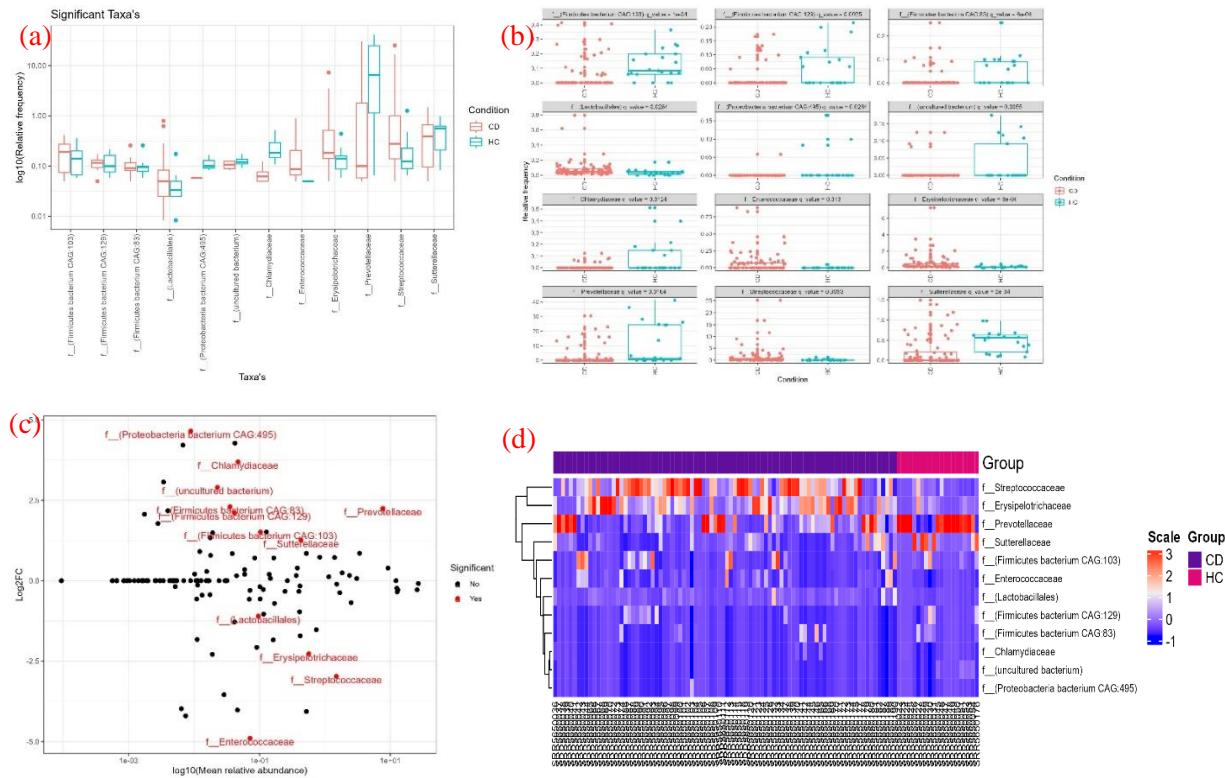


Figure S39. Taxa were identified as significant by using the Wilcoxon Rank Sum test. Results were visualized in a) the grouped box plot; the x-axis represents the taxa and the y-axis represents the $\log_{10}(\text{relative frequency})$. b) An individual box plot for each taxon; the x-axis represents the Condition, the y-axis represents relative frequency, and c) the volcano plot; the x-axis represents the $\log_{10}(\text{mean relative abundance})$ and the y-axis represents Log2FC. d) the heatmap for significantly identified taxa. Likewise, similar plots were generated for the remaining methods.

Summary Table Plot

Result - OTUs that were significantly different between two groups
Total of 12 taxa were identified as significant.

Number of significant taxa
Statistically of significant taxa

Show: 10 entries Search:

OTU Present_in_no_of_CD Present_in_no_of_HC Mean_relative_frequency_CD Mean_relative_frequency_HC All_mean_relative_frequency Difference_between_means fold_change log2FC PValue q_value

OTU	Present_in_no_of_CD	Present_in_no_of_HC	Mean_relative_frequency_CD	Mean_relative_frequency_HC	All_mean_relative_frequency	Difference_between_means	fold_change	log2FC	PValue	q_value
f_(Firmicutes bacterium CAG 103)	f_(Firmicutes bacterium CAG 103)	19	17	0.04273	0.1219	0.1036623	-0.0791382	2.85210443541593	1.51202680986669	1.666e-06 0.0001183
f_(Firmicutes bacterium CAG 129)	f_(Firmicutes bacterium CAG 129)	10	10	0.01308	0.05639	0.0412792	-0.0433109	4.31059812726925	2.10788806777713	0.0001726 0.002451
f_(Firmicutes bacterium CAG 83)	f_(Firmicutes bacterium CAG 83)	8	10	0.01026	0.05049	0.03550325	-0.0402331	4.92219579247012	2.29930204568873	2.102e-05 0.000597
f_(Lactobacillales)	f_(Lactobacillales)	88	17	0.07784	0.03629	0.0959811	0.0415488	0.466207587651117	-1.10095560998824	0.004007 0.02845
f_(Proteobacteria bacterium CAG 495)	f_(Proteobacteria bacterium CAG 495)	1	3	0.0006586	0.01656	0.008937998	-0.015900202	25.142499673549	4.65205618487486	0.00386 0.02845
f_(uncultured bacterium)	f_(uncultured bacterium)	4	6	0.004801	0.0359	0.02275026	-0.03109624	7.4777743523942	2.90260893741786	0.0005405 0.005482
f_Chlamydiaceae	f_Chlamydiaceae	8	7	0.006306	0.08168	0.04714525	-0.0753731	12.9529797953779	3.69521211899496	0.00131 0.0124
f_Enterococcaceae	f_Enterococcaceae	36	1	0.07038	0.002365	0.07155949	0.06801142	0.0336102238237601	-4.89495603962052	0.001459 0.01295
f_Erysipelotrichaceae	f_Erysipelotrichaceae	87	15	0.5142	0.1065	0.5674325	0.407708	0.207088416561161	-2.2716812358588	1.933e-05 0.000597
f_Prevotellaceae	f_Prevotellaceae	54	16	2.285	10.84	7.70489	-8.55361	4.7427357558141	2.24571937442907	0.001968 0.01644

Showing 1 to 10 of 12 entries Previous 1 2 Next

Export the summary tables in csv

Download significant Download all Download relative frequency Total counts in each samples

Download as csv Download as csv Download as csv Download as csv

Figure S40. Summary table for the Wilcoxon Rank Sum test. Likewise, similar tables were generated for the remaining methods.

Table S2. Output table column for the significant taxa by using various methods

	Wilcoxon Rank Sum test	t-test	metagenomeSeq	DESeq2	Limma-Voom	edgeR
OUT (Taxa)	✓	✓	✓	✓	✓	✓
Present_in_no_of_CD (Condition1)	✓	✓	✓			
Present_in_no_of_HC (Condition2)	✓	✓	✓			
Counts_in_HC			✓			
Counts_in_CD			✓			
Mean /Mean_relative_frequency_CD (Condition1)	✓	✓			✓	
Mean / Mean_relative_frequency_HC (Condition2)	✓	✓			✓	
All_mean / All_mean_relative_frequency	✓	✓	✓	✓	✓	
Difference_between_means	✓	✓				
fold_change	✓	✓	✓	✓	✓	✓
log2FC	✓	✓	✓	✓	✓	✓
PValue	✓	✓	✓	✓	✓	✓
FDR or q_value	✓	✓	✓	✓	✓	✓
standard_error			✓			

Multiple group comparison

The multiple-group methods analyze more than two sets of conditions from the metadata, e.g. (Control, case1 and case2). In this tutorial, we have used Healthy control (HC), Crohn's Disease (CD) and Ulcerative Colitis (UC). To analyze the metagenome data, we have incorporated the Kruskal-Wallis test and ANOVA (Analysis of variance). The counts were converted to relative frequency, as mentioned above. Then we used the Kruskal-Wallis test (`kruskal.test`) and ANOVA (`aov`) function for statistical analysis. In addition, we also incorporated the Post-hoc test used to calculate the p-value for pairwise comparison between multiple groups, e.g. (CD vs. HC, CD vs. UC and HC vs. UC). For the Post-hoc test `Dunn.test` from `dunn.test` package was used in the Kruskal-Wallis test. Likewise, TukeyHSD was used under ANOVA (**Figure S41**).

Users need to select the test correction method, either Benjamini-Hochberg FDR or P-value and the Post-hoc test; also, they can adjust the FDR or P-value based on their needs (default is < 0.05). Finally, select any plot type and image format, then click the submit button (**Figure S41**) to visualize the grouped box plot (**Figure S42a**), individual box blot for each taxon (**Figure S42b**) and heatmap (**Figure S42c**). We have similar input methods for the ANOVA methods. It also generates similar plots and summary tables (**Figure S43**).

The screenshot shows a user interface for the Kruskal-Wallis test. At the top, it says "Kruskal-Wallis test". Below that is a "Selected input" dropdown menu containing "Megan_WGS_output.tsv". Under "Test correction", there is a dropdown menu set to "Benjamini-Hochberg FDR". Next is a "FDR or Pvalue" input field with the value "0.05". A red arrow points from this field to the text "User can adjust the value based on their needs". Below these is a "Post-hoc test" section with two radio buttons: "Yes" and "No". The "Yes" button is selected, and a red box surrounds this section with the text "If yes perform pairwise comparison". Another red arrow points from this box to the text "Select Benjamini-Hochberg FDR or P-value". Further down is a "Types of plot" section with three radio buttons: "Grouped box plot" (selected), "Individual box plot", and "Heatmap". A red box surrounds this section with the text "Select plot type to display". A red arrow points from this box to the text "Select Benjamini-Hochberg FDR or P-value". At the bottom is an "Output image format" dropdown menu set to "JPG", and a "Submit" button.

Figure S41. Input selection for the Kruskal-Wallis test and similar input is needed for ANOVA

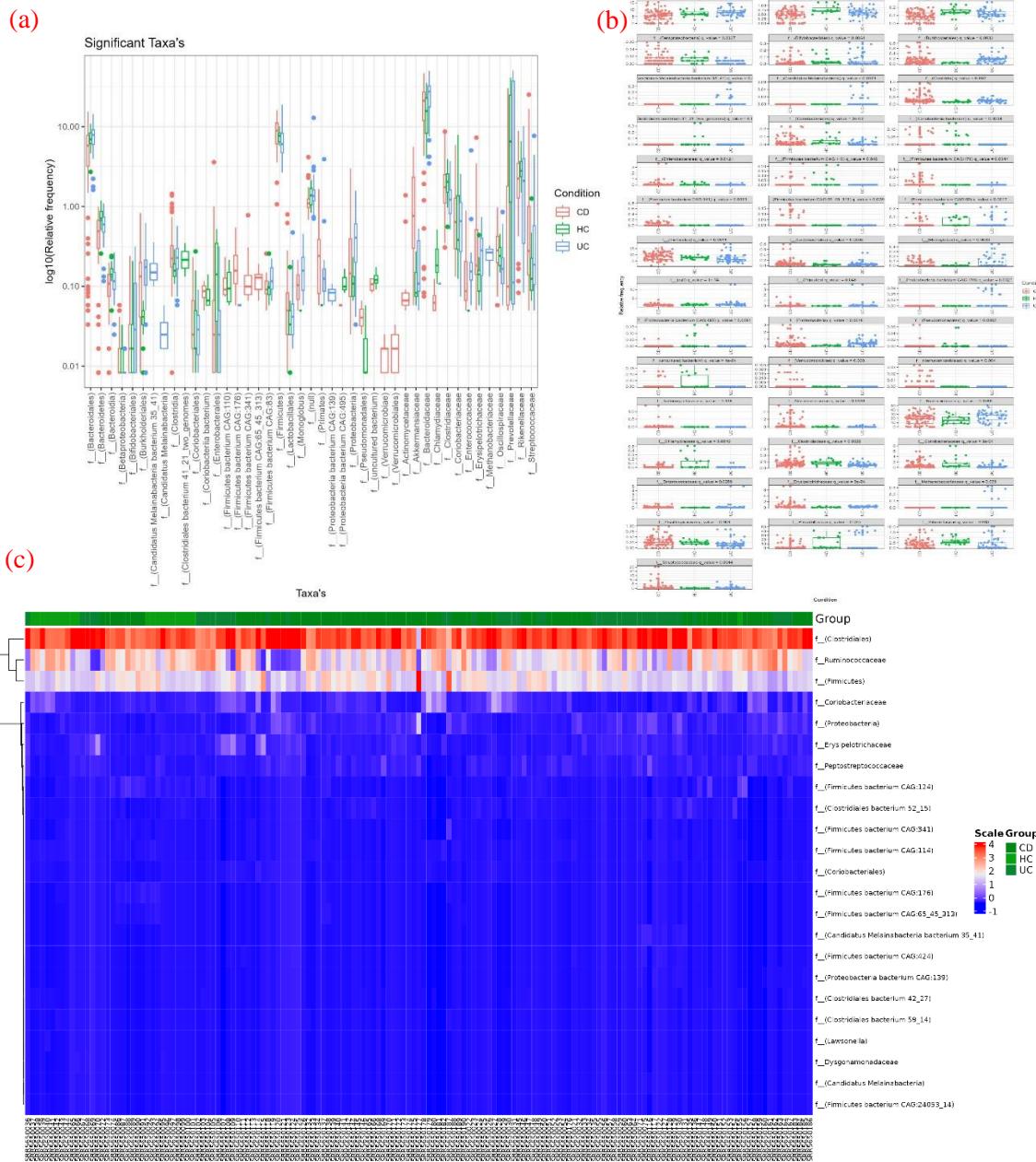


Figure S42. a) Taxa were identified as significant by using the Kruskal-Wallis test. Results were visualized in the grouped box plot in which x-axis represents the taxa and y-axis represents the $\log_{10}(\text{relative frequency})$. b) A box plot for each taxon in which x-axis represents the Condition and y-axis represents the relative frequency c) the heatmap for significantly identified taxa.

Summary Table Plot

Result - OTUs that were significantly different between multiple groups
Total of 43 taxa were identified as significant.

Number of significant taxa
Statistically of significant taxa

Show 10 entries Search:

OTU	Present_in_no_of_CD	Present_in_no_of_HC	Present_in_no_of_UC	Mean_relative_frequency_CD	Mean_relative_frequency_HC	Mean_relative_frequency_UC	PValue	q_value	CD-HC	CD-UC	HC-UC
f__(Bacteroidales)	f__(Bacteroidales)	87	21	48	5.55	7.29	7.807	0.0006911	0.004351	0.01976	0.0001376 0.2797
f__(Bacteroidetes)	f__(Bacteroidetes)	85	21	48	0.4666	0.7568	0.6098	1.187e-05	0.0002522	3.703e-06	0.003219 0.01097
f__(Bacteroidia)	f__(Bacteroidia)	74	21	48	0.09102	0.1487	0.1193	5.632e-05	0.008704	1.224e-05	0.08595 0.01122
f__(Betaproteobacteria)	f__(Betaproteobacteria)	52	17	25	0.009697	0.01183	0.005004	0.007927	0.03369	0.03326	0.02581 0.001191
f__(Bifidobacteriales)	f__(Bifidobacteriales)	45	21	26	0.02024	0.0363	0.04192	0.011133	0.00642	0.0001159	0.1957 0.002329
f__(Burkholderiales)	f__(Burkholderiales)	77	21	46	0.1427	0.03983	0.1662	0.00048	0.003264	0.004776	0.01578 5.22e-05
f__(Candidatus Melainabacteria bacterium 35_41)	f__(Candidatus Melainabacteria bacterium 35_41)	0	0	7	0	0	0.02589	0.0002596	0.002093	0.5	4.356e-05 0.003561
f__(Candidatus Melainabacteria)	f__(Candidatus Melainabacteria)	0	0	7	0	0	0.004315	0.0002586	0.002093	0.5	4.356e-05 0.003561
f__(Clostridia)	f__(Clostridia)	88	21	48	0.3274	0.1676	0.2291	0.0002048	0.002048	1.907e-05	0.11 0.001433
f__(Clostridiaceae bacterium 41_21_two_genomes)	f__(Clostridiaceae bacterium 41_21_two_genomes)	0	2	0	0	0.02287	0	0.001477	0.0081	0.0002456	0.5 0.0006072

Showing 1 to 10 of 43 entries

Export the summary tables in csv

Previous 1 2 3 4 5 Next

Download significant Download all Download relative frequency Total counts in each samples

[Download as csv](#) [Download as csv](#) [Download as csv](#) [Download as csv](#)

Figure S43. The summary table for the Kruskal-Wallis test. The last three columns contain p-values of the pairwise comparison that will display if the Post-hoc test is selected as yes. Likewise, similar tables were generated for the ANOVA.

Session information from R

Here is a list of loaded packages (with their versions) used to develop MetaDAVis (**Figure S44 & S45**).

```
R version 4.4.1 (2024-06-14 ucrt)
Platform: x86_64-w64-mingw32/x64
Running under: windows 10 x64 (Build 19045)

Matrix products: default

locale:
[1] LC_COLLATE=English_United States.utf8  LC_CTYPE=English_United States.utf8   LC_MONETARY=English_United States.utf8 LC_NUMERIC=C
[2] LC_TIME=English_United States.utf8

time zone: America/Chicago
tzcode source: internal

attached base packages:
[1] stats4      grid       stats      graphics    grDevices  utils      datasets    methods     base

other attached packages:
[1] shinyxcsloders_1.1.0   microbiomeutilities_1.00.17   mia_1.12.0          MultiAssayExperiment_1.30.3   TreeSummarizedExperiment_2.12.0   Biostrings_2.72.1
[7] xvector_0.44.0        bluster_1.14.0            metaagenomeseq_1.46.0   rcolorBrewer_1.1.3        gemit_4.1-8             Matrix_1.7-0
[13] edger_4.2.3          limma_3.60.4             DESeq2_1.44.0         scatter_1.32.1          scutte_1.14.0           SingleCellExperiment_1.26.0
[19] SummarizedExperiment_1.34.0 Biobase_2.64.0            GenomicRanges_1.56.1   GenomeInfoDb_1.40.1     IRanges_2.38.1          S4vectors_0.42.1
[25] BiocGenerics_0.50.0   MatrixGenerics_1.16.0    matrixStats_1.4.1      GenomeInfoDbData_1.40.0  patchwork_1.3.0         microbiome_1.26.0
[31] phyloseq_1.48.0      BioManager_1.30.25     plotly_4.10.4         ggridges_2.3.0          patchwork_1.3.0         tools_4.3.0
[37] usethis_3.0.0        dplyr_1.1.4              tidyverse_1.3.1        gtable_0.3.6            scales_1.3.0           devtools_2.4.5
[43] reshape2_1.4.4       ggfortify_0.1.2       ggrepel_0.1.3         dunnit_tidy_1.3.6       scales_1.3.0           tibble_3.2.1
[49] ggpubr_0.6.0        shinyFiles_0.9.3      shinythemes_1.2.0      Dunnit_0.1.3            scales_1.3.0           purrr_0.1.2
[55] gridExtra_2.3.5      shiny_1.9.1             DT_0.33              lattice_0.22-6         scales_1.3.0           purmote_0.9-7
[61] shiny_1.9.1          shiny_1.9.1             DT_0.33              shiny_1.9.1             scales_1.3.0           shiny_1.9.1

loaded via a namespace (and not attached):
[1] fs_1.6.4                bitops_1.0-8           DirichletMultinomial_1.46.0   fontawesome_0.5.2       gghalves_0.1.4          htr_1.4.7               doParallel_1.0.17
[7] profvis_0.3.8          tools_4.4.1             backports_1.5.0          utf8_1.2.4              R6_2.5.1               uwot_0.2.2             lazeval_0.2.2
[13] ngsolve_1.0.0           grid_4.4.1              gettextf_1.0.0            urlchecker_1.0.1       withr_0.1               grDevices_4.4.1         lazyeval_0.2.2
[19] tibbleshaping_0.4.0     labels_1.3.0            gettextf_1.0.5           urlchecker_1.0.1       withr_0.1               gridExtra_2.3.3       clin_1.6.1
[25] sessioninfo_1.2.2      rstudioserver_0.16.0   FNN_1.14.1              sasr_0.1.9              systemfonts_1.1.0      urlchecker_1.0.1       SingleCellExperiment_1.26.0
[31] crossstalk_1.2.1       car_3.1                 biomformat_1.32.0       genericr_0.1.3         gridGraphics_0.5-1     shape_1.4.6.1           decompr_1.24.0
[37] lifecycle_1.0.4        yaml_2.3.10             cardata_3.0-5          ggbeswarm_0.7.2        grid_4.4.1              DECIPIER_3.0.0         tools_4.3.0
[43] promises_3.1.0        crayon_1.5.3            minui_0.1.1             fansi_1.0.6             DECIPHER_3.0.0         abind_1.4-8             Rtsne_0.17
[49] rjson_0.2.22           codetools_0.2-20       cowplot_1.1.3           fansi_1.0.6             DECIPIER_3.0.0         abind_1.4-8             pillar_1.9.0
[55] treeio_1.28.0          glue_1.7.0              data.table_1.16.0       heatmap_2.20.0         knitr_1.48              pillar_1.9.0           png_0.1-8
[61] gttable_0.3.5          cachem_1.1.0            remotest_2.5.0         htr_1.4.7               knitr_1.48              pillar_1.9.0           survminer_3.6-4
[67] pheatmap_0.1.12        iterators_1.0.14       xfun_0.47              remotes_2.5.0          vctrs_0.6.5             survminer_3.6-4
[73] fortunes_1.0.14       iterators_1.0.14       ellipsis_0.3.16        s4array_1.4.1           mime_0.12              survminer_3.6-4
[79] fortunes_1.0.14       iterators_1.0.14       iterators_1.0.14       nime_3.1-164            survival_3.6-4         survminer_3.6-4
[85] BiocNeighbors_1.22.0   fastmap_1.2.0          colMeans_1.1-1          stringr_1.5.1           s4array_1.4.1           survminer_3.6-4
[91] biocgenerics_1.16.0    delayedarray_0.30.1   catools_1.8.3           stringr_1.5.1           digest_0.6.37          tibble_0.8.0             lazeval_0.2.2
[97] fastmap_1.2.0          rlang_1.1.4             GlobalOptions_0.1.2    httr_1.1.5              httr_1.1.5             clin_1.6.1
[103] bioconductor_2.34-24  grid_4.4.1              jsonlite_1.8.8          htmlwidgets_1.6.4       httr_1.1.5             gridExtra_2.3.3
[109] BiocGenerics_1.2.1    DelayedArray_0.30.1   BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[115] MASS_7.3-60.2          catools_1.8.3           BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[121] plyr_1.8.9              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[127] circos_0.4.16          BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[133] locfit_1.5.9-10        BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[139] broom_1.0.7             BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[145] memoise_2.0.1           BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[151] beeswarm_0.4.0          BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[157] cluster_2.1.6           BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[163] ggpubr_0.6.0             BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[169] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[175] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[181] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[187] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[193] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[199] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[205] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[211] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[217] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[223] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[229] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[235] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[241] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[247] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[253] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[259] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[265] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[271] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[277] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[283] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[289] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[295] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[301] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[307] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[313] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[319] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[325] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[331] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[337] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[343] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[349] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[355] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[361] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[367] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[373] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[379] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[385] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[391] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[397] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[403] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[409] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[415] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[421] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[427] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[433] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[439] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[445] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[451] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[457] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[463] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[469] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[475] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[481] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[487] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[493] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[499] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[505] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[511] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[517] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[523] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[529] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[535] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[541] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[547] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[553] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[559] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[565] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[571] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[577] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[583] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[589] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[595] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[601] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[607] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[613] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[619] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[625] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[631] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[637] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[643] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[649] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[655] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[661] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[667] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[673] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[679] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[685] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[691] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[697] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[703] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[709] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[715] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[721] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[727] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[733] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[739] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[745] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[751] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[757] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[763] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[769] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[775] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[781] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[787] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[793] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[799] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[805] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[811] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[817] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[823] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[829] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[835] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[841] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[847] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[853] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[859] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[865] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[871] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[877] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[883] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[889] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[895] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[901] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[907] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[913] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[919] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[925] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[931] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[937] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[943] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[949] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[955] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[961] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[967] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[973] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[979] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[985] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[991] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[997] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[1003] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[1009] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[1015] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[1021] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[1027] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[1033] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[1039] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[1045] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[1051] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[1057] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[1063] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[1069] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[1075] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[1081] shiny_1.9.1              BioCGenerics_1.1.1     BioCGenerics_1.1.1     BioCGenerics_1.1.1     httr_1.1.5             grid_4.4.1
[1087] shiny_1.9.1              BioCGenerics_1.1.1     Bio
```

Figure S44. MetaDAVis application is developed and tested with the listed package version in Windows 10

```

> sessionInfo()
R version 4.4.1 (2024-06-14)
Platform: x86_64-pc-linux-gnu
Running under: Ubuntu 24.04.1 LTS

Matrix products: default
BLAS:  /usr/lib/x86_64-linux-gnublas/libblas.so.3.12.0
LAPACK: /usr/lib/x86_64-linux-gnulapack/liblapack.so.3.12.0

locale:
[1] LC_CTYPE=C.UTF-8        LC_NUMERIC=C           LC_TIME=C.UTF-8
[4] LC_COLLATE=C.UTF-8     LC_MONETARY=C.UTF-8    LC_MESSAGES=C.UTF-8
[7] LC_PAPER=C.UTF-8       LC_NAME=C             LC_ADDRESS=C
[10] LC_TELEPHONE=C        LC_MEASUREMENT=C.UTF-8 LC_IDENTIFICATION=C

time zone: Etc/UTC
tzcode source: system (glibc)

attached base packages:
[1] stats4   grid      stats     graphics  grDevices utils     datasets
[8] methods  base

other attached packages:
[1] microbiomeutilities_1.00.17      mia_1.12.0
[3] MultiAssayExperiment_1.30.3      TreeSummarizedExperiment_2.12.0
[5] Biostrings_2.72.1                xVector_0.44.0
[7] bluster_1.14.0                  metagenomeSeq_1.46.0
[9] RColorBrewer_1.1-3              glnnet_4.1-8
[11] Matrix_1.7-0                   edgeR_4.2.1
[13] limma_3.60.4                   DESeq2_1.44.0
[15] scatter_1.32.1                scuttle_1.14.0
[17] SingleCellExperiment_1.26.0     SummarizedExperiment_1.34.0
[19] Biobase_2.64.0                 GenomicRanges_1.56.1
[21] GenomeInfoDb_1.40.1            IRanges_2.38.1
[23] S4Vectors_0.42.1              BiocGenerics_0.50.0
[25] MatrixGenerics_1.16.0          matrixStats_1.4.1
[27] qvalue_2.36.0                 ComplexHeatmap_2.20.0
[29] microbiome_1.26.0              phyloseq_1.48.0
[31] BiocManager_1.30.25            plotly_4.10.4
[33] GCall_2.2.1                   patchwork_1.2.0
[35] devtools_2.4.5                usethis_3.0.0
[37] dplyr_1.1.4                   tidydr_1.3.1
[39] Dunn.test_1.3.6               scales_1.3.0
[41] tibble_3.2.1                  reshape2_1.4.4
[43] ggplotify_0.1.2              ggfortify_0.4.17
[45] vegan_2.6-8                  lattice_0.22-6
[47] permute_0.9-7               ggpibr_0.6.0
[49] ggplot2_3.5.1                shinyFiles_0.9.3
[51] shinythemes_1.2.0             DT_0.33
[53] shiny_1.9.1

loaded via a namespace (and not attached):
[1] splines_4.4.1                later_1.3.2
[3] bitops_1.0-8                 DirichletMultinomial_1.46.0
[5] lifecycle_1.0.4              rstatix_0.7.2
[7] doParallel_1.0.17            MASS_7.3-61
[9] backports_1.5.0              magrittr_2.0.3
[11] remotes_2.5.0               httpuv_1.6.15
[13] Wrench_1.22.0               sessioninfo_1.2.2
[15] pkgbuild_1.4.4              DBI_1.2.3
[17] ade4_1.7-22                 abind_1.4-5
[19] pkgload_1.4.0               zlibbioc_1.50.0
[21] Rtsne_0.17                   purrr_1.0.2
[23] yulab.utils_0.1.7            circlize_0.4.16
[25] GenomeInfoDbData_1.2.12     ggrepel_0.9.6
[27] irlba_2.3.5.1               tidytree_0.4.6
[29] pheatmap_1.0.12             DelayedMatrixStats_1.26.0
[31] codetools_0.2-20             DelayedArray_0.30.1
[33] tidyselect_1.2.1             shape_1.4.6.1
[35] UCSC.utils_1.0.0             viridis_0.6.5
[37] scaledMatrix_1.12.0          jsonlite_1.8.8
[39] getoptLong_1.0.5             BiocNeighbors_1.22.0
[41] multtest_2.6.0               gghalves_0.1.4
[43] decontam_1.24.0              ellipsis_0.3.2
[45] survival_3.7-0              iterators_1.0.14
[47] foreach_1.5.2               tools_4.4.1
[49] treeio_1.28.0               Rcpp_1.0.13
[51] glue_1.7.0                  gridExtra_2.3
[53] SparseArray_1.4.8            mgcv_1.9-1
[55] withr_3.0.1                 fastmap_1.2.0
[57] rhdf5filters_1.16.0          fansi_1.0.6
[59] caTools_1.18.3              digest_0.6.37
[61] rsvd_1.0.5                  R6_2.5.1
[63] mime_0.12                   gridGraphics_0.5-1
[65] colorspace_2.1-1             gtools_3.9.5
[67] utf8_1.2.4                  generics_0.1.3
[69] DECIPHER_3.0.0              data.table_1.16.0
[71] httr_1.4.7                  htmlwidgets_1.6.4
[73] S4Arrays_1.4.1              ggstats_0.6.0
[75] pkgconfig_2.0.3              gtable_0.3.5
[77] htmitools_0.5.8.1           carData_3.0-5
[79] profvis_0.3.8               biomformat_1.32.0
[81] clue_0.3-65                 png_0.1-8
[83] rjson_0.2.22                nlme_3.1-166
[85] cachem_1.1.0                rhdf5_2.48.0
[87] GlobalOptions_0.1.2          stringr_1.5.1
[89] KernSmooth_2.23-24          viper_0.4.7

```

```
[91] parallel_4.4.1           miniUI_0.1.1.1
[93] pillar_1.9.0             vctrs_0.6.5
[95] gplots_3.1.3.1          urlchecker_1.0.1
[97] promises_1.3.0           car_3.1-2
[99] BiocSingular_1.20.0      beachmat_2.20.0
[101] xtable_1.8-4            cluster_2.1.6
[103] beeswarm_0.4.0          locfit_1.5-9.10
[105] cli_3.6.3               compiler_4.4.1
[107] rlang_1.1.4              crayon_1.5.3
[109] ggsignif_0.6.4          ggbeeswarm_0.7.2
[111] plyr_1.8.9               fs_1.6.4
[113] stringi_1.8.4            viridisLite_0.4.2
[115] BiocParallel_1.38.0       munsell_0.5.1
[117] lazyeval_0.2.2           sparseMatrixStats_1.16.0
[119] Rhdf5lib_1.26.0          statmod_1.5.0
[121] igraph_2.0.3              broom_1.0.6
[123] memoise_2.0.1            ape_5.8
```

Figure S45. MetaDAVis application is developed and tested with the listed package version in Ubuntu and RedHat

References:

- McMurdie and Holmes (2013) phyloseq: An R package for reproducible interactive analysis and graphics of microbiome census data PLoS ONE 8(4):e61217
- Oksanen, F.J., et al. (2017) Vegan: Community Ecology Package. R package Version 2.4 -3. <https://CRAN.R-project.org/package=vegan>
- Chang W, Cheng J, Allaire J, Sievert C, Schloerke B, Xie Y, Allen J, McPherson J, Dupertuis A, Borges B (2022). shiny: Web Application Framework for R. R package version 1.7.4 .9000, <https://shiny.rstudio.com/>.
- Wickham H (2016). ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York. ISBN 978-3-319-24277-4, <https://ggplot2.tidyverse.org>.
- Wickham H (2007). "Reshaping Data with the reshape Package." Journal of Statistical Software, 21(12), 1–20. <http://www.jstatsoft.org/v21/i12/>.
- Tang Y, Horikoshi M, Li W (2016). "ggefortify: Unified Interface to Visualize Statistical Result of Popular R Packages." The R Journal, 8(2), 474–485. doi:10.32614/RJ-2016-060 , <https://doi.org/10.32614/RJ-2016-060>.
- Müller K, Wickham H (2022). tibble: Simple Data Frames. <https://tibble.tidyverse.org/>, <https://github.com/tidyverse/tibble>.
- Wickham H, Seidel D (2022). scales: Scale Functions for Visualization. <https://scales.r-lib.org>, <https://github.com/r-lib/scales>.
- Gu Z (2022). "Complex Heatmap Visualization." iMeta. doi: 10.1002/imt2.43.
- Wickham H, Girlich M (2022). tidyR: Tidy Messy Data. <https://tidyR.tidyverse.org>, <https://github.com/tidyverse/tidyr>.
- Wickham H, François R, Henry L, Müller K (2022). dplyr: A Grammar of Data Manipulation. <https://dplyr.tidyverse.org>, <https://github.com/tidyverse/dplyr>.
- Pedersen T (2022). patchwork: The Composer of Plots. <https://patchwork.data-imaginist.com>, <https://github.com/thomasp85/patchwork>.
- McCarthy DJ, Campbell KR, Lun ATL, Willis QF (2017). "Scater: pre-processing, quality control, normalisation and visualisation of single-cell RNA-seq data in R." Bioinformatics, 33, 1179–1186. doi: 10.1093/bioinformatics/btw777.
- Love MI, Huber W, Anders S (2014). "Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2." Genome Biology, 15, 550. doi: 10.1186/s13059-014-0550-8.
- Robinson MD, McCarthy DJ, Smyth GK (2010). "edgeR: a Bioconductor package for differential expression analysis of digital gene expression data." Bioinformatics, 26(1), 139–140. doi: 10.1093/bioinformatics/btp616.
- Paulson JN, Olson ND, Braccia DJ, Wagner J, Talukder H, Pop M, Bravo HC (2013). metagenomeSeq: Statistical analysis for sparse high-throughput sequencing. Bioconductor package, <http://www.cbcn.umd.edu/software/metagenomeSeq>.
- Ritchie ME, Phipson B, Wu D, Hu Y, Law CW, Shi W, Smyth GK (2015). "limma powers differential expression analyses for RNA-sequencing and microarray studies." Nucleic Acids Research, 43(7), e47. doi: 10.1093/nar/gkv007.
- Lun A (2022). bluster: Clustering Algorithms for Bioconductor. R package version 1.8.0.
- Ernst F, Shetty S, Borman T, Lahti L (2022). mia: Microbiome analysis. R package version 1.6.0, <https://github.com/microbiome/mia>.

- Schloerke B, Cook D, Larmarange J, Briatte F, Marbach M, Thoen E, Elberg A, Crowley J (2022). GGally: Extension to 'ggplot2'. <https://ggobi.github.io/ggally/>, <https://github.com/ggobi/ggally>.
- Kassambara A (2022). _ggpubr: 'ggplot2' Based Publication Ready Plots_. R package version 0.5.0, <<https://CRAN.R-project.org/package=ggpubr>>.
- Shetty S, Lahti L (2022). microbiomeutilities: microbiomeutilities: Utilities for Microbiome Analytics. R package version 1.00.17.
- Xie Y, Cheng J, Tan X (2022). _DT: A Wrapper of the JavaScript Library 'DataTables'_ . R package version 0.26, <<https://CRAN.R-project.org/package=DT>>
- Dinno A (2017). _dunn.test: Dunn's Test of Multiple Comparisons Using Rank Sums_. R package version 1.3.5, <<https://CRAN.R-project.org/package=dunn.test>>.
- Pedersen T, Nijs V, Schaffner T, Nantz E (2022). _shinyFiles: A Server-Side File System Viewer for Shiny_. R package version 0.9.3, <<https://CRAN.R-project.org/package=shinyFiles>>.
- Chang W (2021). _shinythemes: Themes for Shiny_. R package version 1.2.0, <<https://CRAN.R-project.org/package=shinythemes>>.
- Wickham H, Hester J, Chang W, Bryan J (2022). _devtools: Tools to Make Developing R Packages Easier_. R package version 2.4.5, <<https://CRAN.R-project.org/package=devtools>>.
- Morgan M (2022). _BiocManager: Access the Bioconductor Project Package Repository_. R package version 1.30.19, <<https://CRAN.R-project.org/package=BiocManager>>.
- Storey JD, Bass AJ, Dabney A, Robinson D (2022). qvalue: Q-value estimation for false discovery rate control. R package version 2.30.0, <http://github.com/jdstorey/qvalue>.
- Yu G (2021). _ggplotify: Convert Plot to 'grob' or 'ggplot' Object_. R package version 0.1 .0, <<https://CRAN.R-project.org/package=ggplotify>>.
- Leo Lahti et al. microbiome R package. URL: <http://microbiome.github.io>
- Sievert C (2020). Interactive Web-Based Data Visualization with R, plotly, and shiny. Chapman and Hall/CRC. ISBN 9781138331457, <https://plotly-r.com>.