

Package ‘miaDash’

September 24, 2025

Version 1.1.5

Title Dashboard for the interactive analysis and exploration of
microbiome data

Description miaDash provides a Graphical User Interface for the exploration of
microbiome data. This way, no knowledge of programming is required to
perform analyses. Datasets can be imported, manipulated, analysed and
visualised with a user-friendly interface.

biocViews Microbiome, Software, Visualization, GUI, ShinyApps,
DataImport

License Artistic-2.0

Encoding UTF-8

Depends R (>= 4.4.0), iSEE (>= 2.19.4), shiny

Imports ape, bluster, htmltools, iSEEtree (>= 1.1.4), mia, rintrojs,
scater, scuttle, shinydashboard, shinyjs, shinyWidgets,
S4Vectors, SingleCellExperiment, SummarizedExperiment,
TreeSummarizedExperiment, utils, vegan

Suggests BiocStyle, knitr, philr, remotes, rmarkdown, testthat (>= 3.0.0)

URL <https://github.com/microbiome/miaDash>

BugReports <https://github.com/microbiome/miaDash/issues>

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

VignetteBuilder knitr

Config/testthat/edition 3

git_url <https://git.bioconductor.org/packages/miaDash>

git_branch devel

git_last_commit 6b11cd5

git_last_commit_date 2025-09-09

Repository Bioconductor 3.22

Date/Publication 2025-09-23

Author Giulio Benedetti [aut, cre] (ORCID:
<<https://orcid.org/0000-0002-8732-7692>>),
Akwak Jeba [ctb] (ORCID: <<https://orcid.org/0009-0007-1347-7552>>),
Leo Lahti [aut] (ORCID: <<https://orcid.org/0000-0001-5537-637X>>)

Maintainer Giulio Benedetti <giulio.benedetti@utu.fi>

Contents

constants	2
create_observers	3
landing_page	4
miaDash	4
render_output	5
utils	5
Index	7

constants	<i>Constants</i>
-----------	------------------

Description

Constant values used throughout the miaDash app.

Usage

- .miaDashDefaultPanels
- .miaDashOtherPanels
- .transformMethods
- .qualityMetrics
- .alphaMetrics
- .betaMetrics
- .betaMethods
- .clustMethods
- .DmmCriteria

Format

- An object of class character of length 7.
- An object of class character of length 9.
- An object of class character of length 19.
- An object of class list of length 4.
- An object of class character of length 23.
- An object of class character of length 4.
- An object of class character of length 4.
- An object of class character of length 4.
- An object of class character of length 3.

Panel layout

.miaDashDefaultPanels List of panel names in the default layout of miaDash.

.miaDashOtherPanels List of panel names not in the default layout of miaDash.

.transformMethods List of transformations applicable to assays.

.alphaMetrics List of alpha diversity metrics.

.betaMetrics List of beta diversity metrics.

.betaMethods List of beta diversity methods.

.qualityMetrics List of metrics for quality control.

Author(s)

Giulio Benedetti

create_observers	<i>Observers</i>
------------------	------------------

Description

.create_observers and .create_launch_observers define the server to import and build TreeSE objects and track the state of the Build and Launch buttons.

Usage

```
.create_import_observers(input, rObjects)
.create_manipulate_observers(input, rObjects)
.create_estimate_observers(input, rObjects)
.update_observers(input, session, rObjects)
.create_launch_observers(FUN, input, session, rObjects)
```

Arguments

input	The Shiny input object from the server function.
rObjects	A reactive list of values generated in the landing page.

Value

Observers are created in the server function in which this is called. A NULL value is invisibly returned.

landing_page	<i>Landing page</i>
--------------	---------------------

Description

.landing_page creates the landing page of miaDash, where TreeSE objects can be built and iSEE can be launched.

Usage

```
.landing_page(FUN, input, output, session)
```

Value

The UI is defined by the function. A NULL value is invisibly returned.

miaDash	<i>miaDash</i>
---------	----------------

Description

miaDash is a web app that provides an interface to build and explore [TreeSummarizedExperiment](#) (TreeSE) objects by means of [iSEE](#).

Usage

```
miaDash()
```

Value

An [iSEE](#) app with a custom landing page to build TreeSE objects and explore [mia datasets](#).

See Also

[iSEE](#) [mia](#) [miaViz](#)

Examples

```
app <- miaDash()

if (interactive()) {
  shiny::runApp(app)
}
```

render_output	<i>Outputs</i>
---------------	----------------

Description

.render_overview and .render_download create the output of the UI, which consists of the overview of the TreeSE object and the download object, respectively.

Usage

```
.render_overview(output, rObjects)
```

```
.render_download(output, rObjects)
```

Arguments

output The Shiny output object from the server function.

Value

Adds a rendered item to output. A NULL value is invisibly returned.

utils	<i>miaDash utilities</i>
-------	--------------------------

Description

Helper functions to support the app functionality.

Usage

```
.import_datasets(selection)
```

```
.update_tse(tse, fun, fun.args = list())
```

```
.print_message(..., title = "Invalid input:")
```

```
.set_optarg(item, loader = NULL, alternative = NULL, ...)
```

```
.check_formula(form, tse)
```

```
.importBIOM(file, col.data = NULL, tree.file = NULL, ...)
```

```
.rownames2taxa(x)
```

Arguments

<code>selection</code>	Numeric vector. A list of indices for the mia datasets to return.
<code>tse</code>	a TreeSummarizedExperiment object.
<code>fun</code>	Function scalar. Function to apply to tse.
<code>fun.args</code>	Named list. A list of arguments to pass to fun.
<code>...</code>	Either a series of strings to form the message returned by <code>.print_message</code> or named arguments for loader.
<code>title</code>	Character scalar. The title of the error message to print. (Default: "Invalid input:")
<code>item</code>	Character scalar. The file path from which the object should be loaded.
<code>loader</code>	Function scalar. The function to load item. (Default: NULL)
<code>alternative</code>	an alternative output to return when at least one <code>item</code> and <code>loader</code> are not defined. (Default: NULL)
<code>form</code>	Character scalar The formula to be checked.

Value

- `.import_datasets`: returns the list of available mia datasets.
- `.update_tse`: returns a `TreeSE` object after applying `fun` with arguments `fun.args`. Eventual messages and errors are also printed.
- `.print_message`: returns a `modalDialog` with the error message specified with `...` and titled `title`.
- `.set_optarg`: returns an optional element for a `TreeSE` constructor. The output is either an object located at the file path `item` and loaded with `loader` or `alternative`.
- `.check_formula`: returns `TRUE` or `FALSE` depending whether or not all variables included in `form` are present in `se.colData`.

Index

- * **datasets**
 - constants, [2](#)
- * **internal**
 - constants, [2](#)
 - create_observers, [3](#)
 - landing_page, [4](#)
 - render_output, [5](#)
 - utils, [5](#)
 - .DmmCriteria (constants), [2](#)
 - .alphaMetrics (constants), [2](#)
 - .betaMethods (constants), [2](#)
 - .betaMetrics (constants), [2](#)
 - .check_formula (utils), [5](#)
 - .clustMethods (constants), [2](#)
 - .create_estimate_observers
 - (create_observers), [3](#)
 - .create_import_observers
 - (create_observers), [3](#)
 - .create_launch_observers
 - (create_observers), [3](#)
 - .create_manipulate_observers
 - (create_observers), [3](#)
 - .importBIOM (utils), [5](#)
 - .import_datasets (utils), [5](#)
 - .landing_page (landing_page), [4](#)
 - .miaDashDefaultPanels (constants), [2](#)
 - .miaDashOtherPanels (constants), [2](#)
 - .print_message (utils), [5](#)
 - .qualityMetrics (constants), [2](#)
 - .render_download (render_output), [5](#)
 - .render_overview (render_output), [5](#)
 - .rownames2taxa (utils), [5](#)
 - .set_optarg (utils), [5](#)
 - .transformMethods (constants), [2](#)
 - .update_observers (create_observers), [3](#)
 - .update_tse (utils), [5](#)
 - constants, [2](#)
 - create_observers, [3](#)
 - iSEE, [4](#)
 - landing_page, [4](#)
 - mia, [4](#)
 - mia datasets, [4](#)
 - miaDash, [4](#)
 - miaViz, [4](#)
 - render_output, [5](#)
 - TreeSummarizedExperiment, [4](#), [6](#)
 - utils, [5](#)