Package: file2meco (via r-universe)

July 1, 2024

Type Package

Title Transform Files to 'microtable' Object with 'microeco' Package

Version 0.8.0

Author Chi Liu [aut, cre]

Maintainer Chi Liu <liuchi0426@126.com>

Description Transform output files of some tools to the 'microtable' object of 'microtable' class in 'microeco' package. The 'microtable' class is the basic class in 'microeco' package and is necessary for the downstream microbial community data analysis.

URL https://github.com/ChiLiubio/file2meco

Depends R (>= 3.5.0)

Imports microeco (>= 1.8.0), ape, magrittr, dplyr, tidyr, yaml, rhdf5, Matrix

Suggests Biostrings, seqinr, phyloseq, readxl, SummarizedExperiment, TreeSummarizedExperiment

License GPL-3

LazyData true

Encoding UTF-8

NeedsCompilation no

Repository CRAN

Date/Publication 2024-06-30 08:20:02 UTC

RoxygenNote 7.3.1

Contents

check_match_table 2
check_sample_table 3
CHOCOPhlAn_taxonomy 3
file2meco 4

humann2meco 4
meco2phyloseq 7
meco2tse 8
MetaCyc_pathway_map 8
metacyc_pathway_website 9
mpa2meco 9
ncyc2meco 11
ncyc_map 13
pcyc_map 13
phyloseq2meco 13
qiime1meco 14
qiime2meco 15
tse2meco 17
vs2meco 17

Index **19**

`check_match_table` *Replace the names use match table*

Description

Replace the names use match table

Usage

```
check_match_table(match_table = NULL, abund_new = NULL)
```

Arguments

`match_table`	default NULL; character or data.frame; matching table used.
`abund_new`	default NULL; data.frame; the abundance table used.

Value

new abundance table.

check_sample_table *Read sample table*

Description

Read sample table

Usage

```
check_sample_table(sample_table = NULL)
```

Arguments

sample_table	default NULL; character or data.frame; matching table used.

Value

sample information table.

CHOCOPhlAn_taxonomy *The CHOCOPhlAn_taxonomy data*

Description

The CHOCOPhlAn_taxonomy data is used for the parsing the 'HUMAaN' metagenomic results and add the taxonomy hierarchical information to the 'tax_table'.

Usage

```
data(CHOCOPhlAn_taxonomy)
```

file2meco *Introduction to file2meco package (Rhrefhttps://github.com/ChiLiubio/file2mecohttps://github.com/ChiLiubio/file2meco)*

Description

For the detailed tutorial on the file2meco package, please follow the tutorial link in the github repository (https://github.com/ChiLiubio/file2meco)

Please open the help document by using `help` function or by clicking the following links collected:
`qiime1meco`
`qiime2meco`
`humann2meco`
`mpa2meco`
`ncyc2meco`
`phyloseq2meco`
`meco2phyloseq`
`vs2meco`
`tse2meco`

To report bugs or discuss questions, please use Github Issues (https://github.com/ChiLiubio/file2meco/issues). Before creating a new issue, please read the guideline (https://chiliubio.github.io/microeco_tutorial/notes.html#github-issues).

To cite file2meco package in publications, please run the following command to get the reference: `citation("file2meco")`

Reference:
Liu, C., Li, X., Mansoldo, F.R.P., An, J., Kou, Y., Zhang, X., Wang, J., Zeng, J., Vermelho, A.B., Yao, M., 2022. Microbial habitat specificity largely affects microbial co-occurrence patterns and functional profiles in wetland soils. Geoderma 418, 115866.

humann2meco *Transform 'HUMAnN' metagenomic results to 'microtable' object.*

Description

Transform 'HUMAnN' metagenomic results to microtable object, reference: Franzosa et al. (2018) <doi:10.1038/s41592-018-0176-y>.

Usage

```
humann2meco(
  feature_table,
  db = c("MetaCyc", "KEGG", "gene")[1],
  sample_table = NULL,
```

```
  match_table = NULL,
  ...
)
```

Arguments

feature_table	file path of 'HUMAnN' output abundance table; Please see the example.
db	default "MetaCyc"; one of "MetaCyc", "KEGG" or "gene"; "MetaCyc" or "KEGG" means the input feature table is pathway abundance. "gene" represents the abundance of genes, such as 'eggNOG', 'KO' and 'EC'. When using "gene", the generated tax_table has only taxonomic lineages and gene name, no higher functional levels.
sample_table	default NULL; sample metadata table; If provided, must be one of the several types of formats: 1) comma seperated file with the suffix csv or tab seperated file with suffix tsv or txt; 2) Excel type file with the suffix xlsx or xls; require `readxl` package to be installed; 3) `data.frame` object from R.
match_table	default NULL; a two column table used to replace the sample names in feature table; Must be two columns without column names; The first column must be raw sample names same with those in feature table, the second column must be new sample names same with the rownames in sample_table; Please also see the example files. If provided, must be one of the several types of formats: 1) comma seperated file with the suffix csv or tab seperated file with suffix tsv/txt; 2) Excel type file with the suffix xlsx or xls; require `readxl` package to be installed; 3) `data.frame` object from R.
...	parameter passed to `microtable$new` function of `microeco` package, such as `auto_tidy` parameter.

Value

`microtable` object.

Examples

```
library(file2meco)
library(microeco)
library(magrittr)
sample_file_path <- system.file("extdata", "example_metagenome_sample_info.tsv",
  package="file2meco")
match_file_path <- system.file("extdata", "example_metagenome_match_table.tsv", package="file2meco")
# MetaCyc pathway examples
# use the raw data files stored inside the package for MetaCyc pathway database based analysis
abund_file_path <- system.file("extdata", "example_HUMAnN_MetaCyc_abund.tsv", package="file2meco")
# the default db is "MetaCyc"
humann2meco(abund_file_path, db = "MetaCyc")
```

```
humann2meco(abund_file_path, db = "MetaCyc", sample_table = sample_file_path,
  match_table = match_file_path)
test <- humann2meco(abund_file_path, db = "MetaCyc", sample_table = sample_file_path,
  match_table = match_file_path)
test$tidy_dataset()
# rel = FALSE sum original abundance instead of relative abundance
test$cal_abund(select_cols = 1:3, rel = FALSE)
test$taxa_abund$Superclass1 %<>% .[!grepl("unclass", rownames(.)), ]
# use_percentage = FALSE disable percentage for relative abundance
test1 <- trans_abund$new(test, taxrank = "Superclass1", ntaxa = 10, use_percentage = FALSE)
# reassign ylab title instead of default 'Relative Abundance'
test1$ylabname <- "Abundance (RPK)"
# bar_full = FALSE show original abundance instead of normalized 0-1
test1$plot_bar(facet = "Group", bar_full = FALSE)
# select both function and taxa
test$cal_abund(select_cols = c("Superclass1", "Phylum", "Genus"), rel = TRUE)
test1 <- trans_abund$new(test, taxrank = "Phylum", ntaxa = 10, delete_taxonomy_lineage = TRUE)
test1$plot_bar(facet = "Group")
test$taxa_abund$Phylum %<>% .[!grepl("unclass", rownames(.)), ]
test1 <- trans_abund$new(test, taxrank = "Phylum", ntaxa = 10, delete_taxonomy_lineage = FALSE)
test1$plot_bar(facet = "Group")
# functional biomarker
test$cal_abund(select_cols = 1:3, rel = TRUE)
test$taxa_abund$Superclass1 %<>% .[!grepl("unclass", rownames(.)), ]
test$taxa_abund$Superclass2 %<>% .[!grepl("unclass", rownames(.)), ]
test$taxa_abund$pathway %<>% .[!grepl("unclass", rownames(.)), ]
test1 <- trans_diff$new(test, method = "lefse", group = "Group")
test1$plot_diff_bar(use_number = 1:20)
# taxa biomarker
test$cal_abund(select_cols = 4:9, rel = TRUE)
test$taxa_abund$Phylum %<>% .[!grepl("unclass", rownames(.)), ]
test1 <- trans_diff$new(test, method = "lefse", group = "Group", p_adjust_method = "none")
test1$plot_diff_bar(threshold = 2)
############################################################
# KEGG pathway examples
abund_file_path <- system.file("extdata", "example_HUMAnN_KEGG_abund.tsv", package="file2meco")
humann2meco(abund_file_path, db = "KEGG")
test <- humann2meco(abund_file_path, db = "KEGG",
  sample_table = sample_file_path, match_table = match_file_path)
test$tax_table %<>% subset(Level.1 != "unclassified")
test$tidy_dataset()
test$cal_abund(select_cols = 1:3, rel = FALSE)
# use_percentage = FALSE disable percentage for relative abundance
test1 <- trans_abund$new(test, taxrank = "Level.2", ntaxa = 10, use_percentage = FALSE)
# or use ggplot2::ylab to change ylab title
test1$ylabname <- "Abundance (RPK)"
test1$plot_bar(facet = "Group", bar_full = FALSE)
# select both function and taxa
test$cal_abund(select_cols = c("Level.1", "Phylum", "Genus"), rel = TRUE)
test1 <- trans_abund$new(test, taxrank = "Phylum", ntaxa = 10, delete_taxonomy_lineage = FALSE)
test1$plot_bar(facet = "Group")
# functional biomarker
test$cal_abund(select_cols = 1:3, rel = TRUE)
```

```
test1 <- trans_diff$new(test, method = "lefse", group = "Group")
test1$plot_diff_bar(threshold = 3)
# taxa biomarker
test$cal_abund(select_cols = 4:9, rel = TRUE)
test1 <- trans_diff$new(test, method = "lefse", group = "Group", p_adjust_method = "none")
test1$plot_diff_bar(threshold = 2)
```

`meco2phyloseq` *Transform 'microtable' object of 'microeco' package to the 'phyloseq' object of 'phyloseq' package <doi:10.1371/journal.pone.0061217>.*

Description

Transform 'microtable' object of 'microeco' package to the 'phyloseq' object of 'phyloseq' package <doi:10.1371/journal.pone.0061217>.

Usage

```
meco2phyloseq(meco)
```

Arguments

`meco`	a microtable object.

Value

phyloseq object.

Examples

```
## Not run:
library(microeco)
data("dataset")
meco2phyloseq(dataset)

## End(Not run)
```

`meco2tse` *Transform 'microtable' object of 'microeco' package to the 'TreeSummarizedExperiment' object of 'TreeSummarizedExperiment' package <doi:10.12688/f1000research.26669.2>.*

Description

Transform 'microtable' object of 'microeco' package to the 'TreeSummarizedExperiment' object of 'TreeSummarizedExperiment' package <doi:10.12688/f1000research.26669.2>.

Usage

```
meco2tse(meco, ...)
```

Arguments

`meco`	a microtable object.
`...`	parameter passed to `TreeSummarizedExperiment` function of TreeSummarizedExperiment package, e.g. `colTree`, `rowNodeLab` and `colNodeLab`.

Value

TreeSummarizedExperiment object.

Examples

```
## Not run:
library(microeco)
data("dataset")
meco2tse(dataset)

## End(Not run)
```

`MetaCyc_pathway_map` *The MetaCyc_pathway_map data*

Description

The MetaCyc_pathway_map data is a manually curated 'MetaCyc' pathway hierarchical structure data. It is used for the parsing the 'HUMAaN' metagenomic abundance table associated with 'MetaCyc' database. Currently, only superclass 1, 2 and the pathway are used in this data.

Usage

```
data(MetaCyc_pathway_map)
```

`metacyc_pathway_website` *Get the website for a 'MetaCyc' pathway name*

Description

Get the website for a 'MetaCyc' pathway name

Usage

```
metacyc_pathway_website(pathway = NULL)
```

Arguments

`pathway`	default NULL; character vector; one or more MetaCyc pathway names.

Value

character vector.

Examples

```
metacyc_pathway_website("FOLSYN-PWY")
```

`mpa2meco` *Transform metagenomic classification results of 'mpa' format to 'microtable' object.*

Description

Transform the classification results of mpa (MetaPhlAn) format to microtable object, such as MetaPhlAn and Kraken2/Bracken results. Kraken2/Bracken results can be obtained by merge_metaphlan_tables.py from MetaPhlAn or combine_mpa.py from KrakenTools (https://ccb.jhu.edu/software/krakentools/). The algorithm of Kraken2 determines that the abundance of a taxon is not equal to the sum of abundances of taxa in its subordinate lineage. So the default tables in taxa_abund of return microtable object are extracted from the abundances of raw file. It is totally different with the return taxa_abund of cal_abund function, which sums the abundances of taxa at different taxonomic levels based on the taxonomic table and the otu_table (i.e., taxa abundance table at a specified level, e.g., 's__').

Usage

```
mpa2meco(
  feature_table,
  sample_table = NULL,
  match_table = NULL,
  use_level = "s__",
  rel = FALSE,
  sel_same = 1,
  ...
)
```

Arguments

feature_table	'mpa' format abundance table, see the example.
sample_table	default NULL; sample metadata table; If provided, must be one of the several types of formats: 1) comma seperated file with the suffix csv or tab seperated file with suffix tsv/txt; 2) Excel type file with the suffix xlsx or xls; require `readxl` package to be installed; 3) `data.frame` object from R.
match_table	default NULL; a two column table used to replace the sample names in abundance table; Must be two columns without column names; The first column must be raw sample names same with those in feature table, the second column must be new sample names same with the rownames in sample_table; Please also see the example files.
use_level	default "s__"; the prefix parsed for the otu_table and tax_table; must be one of 'd__', 'k__', 'p__', 'c__', 'o__', 'f__', 'g__' and 's__'.
rel	default FALSE; Whether convert the original abundance to relative abundance in generated taxa_abund list. If TRUE, all the data.frame objects in taxa_abund list have relative abundance (0-1).
sel_same	default 1; How to select the taxonomic information in `tax_table` when multiple taxonomic information have same prefix (e.g., "k__Eukaryota|k__Fungi"). 1 represents the first one. 2 denotes the second one. 3 means selecting both. If `sel_same = 3`, the "|" in the taxonomic information will be replaced with ":". For the `taxa_abund` list, both names are remained to facilitate subsequent filtering, and the "|" will be also replaced with ":".
...	parameter passed to microtable$new function of microeco package, such as auto_tidy parameter.

Value

microtable object.

Examples

```
library(microeco)
library(file2meco)
library(magrittr)
# use Kraken2 file stored inside the package
```

```
abund_file_path <- system.file("extdata", "example_kraken2_merge.txt", package="file2meco")
mpa2meco(abund_file_path)
# add sample information table
sample_file_path <- system.file("extdata", "example_metagenome_sample_info.tsv",
  package="file2meco")
# sample names are different between abund_file_path and sample_file_path;
# use a matching table to adjust them
match_file_path <- system.file("extdata", "example_metagenome_match_table.tsv", package="file2meco")
test <- mpa2meco(abund_file_path, sample_table = sample_file_path,
  match_table = match_file_path, use_level = "s__", rel = TRUE)
# make the taxonomy standard for the following analysis
test$tax_table %<>% tidy_taxonomy
test$tidy_dataset()
# calculate taxa_abund with specified level instead of raw kraken results
test1 <- clone(test)
test1$cal_abund(rel = TRUE)
identical(test$taxa_abund$Kingdom, test1$taxa_abund$Kingdom)
```

`ncyc2meco` *Transform 'NCycDB' or 'PCycDB' metagenomic abundance to 'microtable' object.*

Description

Transform 'NCycDB' or 'PCycDB' metagenomic abundance to microtable object. The function can identify the mapping database according to the gene names of input feature abundance table. Reference: Qichao et al. (2019) <doi: 10.1093/bioinformatics/bty741> and Zeng et al. (2022) <doi: 10.1186/s40168-022-01292-1>.

Usage

```
ncyc2meco(feature_table, sample_table = NULL, match_table = NULL, ...)
```

Arguments

`feature_table`	'NCycDB' or 'PCycDB' output abundance table, see the example file.
`sample_table`	default NULL; sample metadata table; If provided, must be one of the several types of formats: 1) comma seperated file with the suffix csv or tab seperated file with suffix tsv/txt; 2) Excel type file with the suffix xlsx or xls; require `readxl` package to be installed; 3) `data.frame` object from R. A file path must be tab or comma seperated file, generally, a file with suffix "tsv" or "csv".
`match_table`	default NULL; a two column table used to replace the sample names in abundance table; Must be two columns without column names; The first column must be raw sample names same with those in feature table, the second column must be new sample names same with the rownames in sample_table; Please also see the example files. A file path must be tab or comma seperated file, e.g. a file with suffix "tsv" or "csv".

... parameter passed to microtable$new function of microeco package, such as auto_tidy parameter.

Value

microtable object.

Examples

```
# NCycDB
abund_file_path <- system.file("extdata", "example_Ncyc_table.tsv", package="file2meco")
sample_file_path <- system.file("extdata", "example_metagenome_sample_info.tsv",
  package="file2meco")
match_file_path <- system.file("extdata", "example_metagenome_match_table.tsv", package="file2meco")
library(microeco)
library(file2meco)
library(magrittr)
ncyc2meco(abund_file_path)
test <- ncyc2meco(abund_file_path, sample_table = sample_file_path,
  match_table = match_file_path)
test$tidy_dataset()
# use split_group = TRUE to calculate the pathway abundance with multipe map correspondance
test$cal_abund(select_cols = 1, rel = TRUE, split_group = TRUE, split_column = "Pathway")
test$taxa_abund$Pathway %<>% .[!grepl("unclass", rownames(.)), ]
test1 <- trans_abund$new(test, taxrank = "Pathway")
test1$plot_bar(bar_full = FALSE)
# for gene abundance, no splitting on the Pathway
test$cal_abund(select_cols = 1:2, rel = TRUE, split_group = FALSE)
test$taxa_abund$Gene %<>% .[!grepl("unclass", rownames(.)), ]
test1 <- trans_abund$new(test, taxrank = "Gene")
test1$plot_bar(bar_full = FALSE)

# PCycDB
abund_file_path <- system.file("extdata", "example_Pcyc_table.tsv", package="file2meco")
test <- ncyc2meco(abund_file_path)
test$tidy_dataset()
# show pathway abundance
test$cal_abund(select_cols = 1, rel = TRUE, split_group = TRUE, split_by = "&&",
    split_column = "Pathway")
test$taxa_abund$Pathway %<>% .[!grepl("unclass|Others", rownames(.)), ]
test1 <- trans_abund$new(test, taxrank = "Pathway")
test1$plot_bar(bar_full = FALSE)
# show gene abundance
test$cal_abund(select_cols = 2, rel = TRUE, split_group = FALSE)
test$taxa_abund$Gene %<>% .[!grepl("unclass", rownames(.)), ]
test1 <- trans_abund$new(test, taxrank = "Gene")
test1$plot_bar(bar_full = FALSE)
```

ncyc_map *The ncyc_map data*

Description

The ncyc_map data is used for the parsing the 'NCycDB' metagenomic results and add the N cycle pathway information to the 'tax_table' of 'microtable' object.

Usage

```
data(ncyc_map)
```

pcyc_map *The pcyc_map data*

Description

The pcyc_map data is used for the parsing the 'PCycDB' metagenomic results and add the P cycle pathway information to the 'tax_table' of 'microtable' object.

Usage

```
data(pcyc_map)
```

phyloseq2meco *Transform the 'phyloseq' object of 'phyloseq' package to 'microtable' object of 'microeco' package.*

Description

Transform the 'phyloseq' object of 'phyloseq' package to 'microtable' object of 'microeco' package.

Usage

```
phyloseq2meco(physeq, ...)
```

Arguments

physeq	a phyloseq object <doi:10.1371/journal.pone.0061217>.
...	parameter passed to microtable$new function of microeco package, such as auto_tidy parameter.

Value

microtable object.

Examples

```
## Not run:
library(phyloseq)
data("GlobalPatterns")
phyloseq2meco(GlobalPatterns)

## End(Not run)
```

`qiime1meco` *Transform 'QIIME' results to 'microtable' object.*

Description

Transform 'QIIME' results to microtable object. The QIIME results refer in particular to the files of qiime1 software.

Usage

```
qiime1meco(
  feature_table,
  sample_table = NULL,
  match_table = NULL,
  phylo_tree = NULL,
  rep_fasta = NULL,
  ...
)
```

Arguments

`feature_table`	the otu table generated from 'QIIME'. Taxonomic information should be in the end of the file.
`sample_table`	default NULL; sample metadata table; If provided, must be one of the several types of formats: 1) comma seperated file with the suffix csv or tab seperated file with suffix tsv/txt; 2) Excel type file with the suffix xlsx or xls; require `readxl` package to be installed; 3) `data.frame` object from R.
`match_table`	default NULL; a two column table used to replace the sample names in feature table; Must be two columns without column names; The first column must be raw sample names same with those in feature table, the second column must be new sample names same with the rownames in sample_table; Please also see the example files. If provided, must be one of the several types of formats: 1) comma seperated file with the suffix csv or tab seperated file with suffix tsv/txt;

	2) Excel type file with the suffix xlsx or xls; require `readxl` package to be installed; 3) `data.frame` object from R.
`phylo_tree`	default NULL; the phylogenetic tree; generally, a file with suffix "tre".
`rep_fasta`	default NULL; the representative sequences; a fasta file, generally with suffix "fasta" or "fna" or "fa".
`...`	parameter passed to microtable$new function of microeco package, such as `auto_tidy` parameter.

Value

`microtable` object.

Examples

```
## Not run:
# use the raw data files stored inside the package
otu_file_path <- system.file("extdata", "otu_table_raw.txt", package="file2meco")
sample_file_path <- system.file("extdata", "sample_info.csv", package="file2meco")
phylo_file_path <- system.file("extdata", "rep_phylo.tre", package="file2meco")
rep_fasta_path <- system.file("extdata", "rep.fna", package="file2meco")
qiime1meco(otu_file_path, sample_table = sample_file_path)
qiime1meco(otu_file_path, sample_table = sample_file_path,
  phylo_tree = phylo_file_path)
qiime1meco(otu_file_path, sample_table = sample_file_path,
  phylo_tree = phylo_file_path, rep_fasta = rep_fasta_path)

## End(Not run)
```

---

qiime2meco *Transform 'QIIME2' results to 'microtable' object.*

Description

Transform 'QIIME2' qza results to microtable object.

Usage

```
qiime2meco(
  feature_table,
  sample_table = NULL,
  match_table = NULL,
  taxonomy_table = NULL,
  phylo_tree = NULL,
  rep_fasta = NULL,
  ...
)
```

Arguments

feature_table	the ASV abundance data with qza format, such as the 'data2_table.qza' in the example.
sample_table	default NULL; the sample metadata table; four types of formats are available: 1) q2-type tab seperated file of QIIME2, such as the 'sample-metadata.tsv' in the example; 2) comma seperated file with the suffix csv or tab seperated file with suffix tsv or txt; 3) Excel type file with the suffix xlsx or xls; require readxl package to be installed; 4) data.frame object from R.
match_table	default NULL; a two column table used to replace the sample names in feature table; Must be two columns without column names; The first column must be raw sample names same with those in feature table, the second column must be new sample names same with the rownames in sample_table; Please also see the example files. If provided, must be one of the several types of formats: 1) comma seperated file with the suffix csv or tab seperated file with suffix tsv or txt; 2) Excel type file with the suffix xlsx or xls; require readxl package to be installed; 3) data.frame object from R.
taxonomy_table	default NULL; the taxonomy assignment data with qza format, such as the 'taxonomy.qza' in the example.
phylo_tree	default NULL; the phylogenetic tree with qza format, such as the 'tree.qza' in the example.
rep_fasta	default NULL; the representative sequences with qza format, such as the 'dada2_rep_set.qza' in the example.
...	parameter passed to microtable$new function of microeco package, such as auto_tidy parameter.

Value

microtable object.

Examples

```
## Not run:
# The data files is downloaded from https://docs.qiime2.org/2020.8/tutorials/pd-mice/
#   and stored inside the package.
abund_file_path <- system.file("extdata", "dada2_table.qza", package="file2meco")
sample_file_path <- system.file("extdata", "sample-metadata.tsv", package="file2meco")
taxonomy_file_path <- system.file("extdata", "taxonomy.qza", package="file2meco")
qiime2meco(abund_file_path)
qiime2meco(abund_file_path, sample_table = sample_file_path)
qiime2meco(abund_file_path, sample_table = sample_file_path,
  taxonomy_table = taxonomy_file_path)

## End(Not run)
```

tse2meco — *Transform the 'TreeSummarizedExperiment' object to 'microtable' object of 'microeco' package.*

Description

Transform the 'TreeSummarizedExperiment' object to 'microtable' object of 'microeco' package.

Usage

```
tse2meco(tse, ...)
```

Arguments

tse	a TreeSummarizedExperiment object <doi:10.12688/f1000research.26669.2>.
...	parameter passed to microtable$new function of microeco package, such as auto_tidy parameter.

Value

microtable object.

vs2meco — *Transform viromescan results to 'microtable' object.*

Description

Transform the results of viromescan software to microtable object. The output of viromescan is single file for each sample. All the results are needed to be merged and adjusted (for several chaotic taxonomy). The input should be the 'count' tables at Species level, i.e. Species_level_results-Counts.txt. For more details, please see the reference <DOI: 10.1186/s12864-016-2446-3>.

Usage

```
vs2meco(input_dir, sample_table = NULL, match_table = NULL, ...)
```

Arguments

input_dir	the input directory, containing all the result folders for each sample. Each folder should be named by the sample name.
sample_table	default NULL; sample metadata table; If provided, must be one of the several types of formats: 1) comma seperated file with the suffix csv or tab seperated file with suffix tsv/txt; 2) Excel type file with the suffix xlsx or xls; require `readxl` package to be installed; 3) `data.frame` object from R.

match_table	default NULL; a two column table used to replace the sample names in abundance table; Must be two columns without column names; The first column must be raw sample names same with those in feature table, the second column must be new sample names same with the rownames in sample_table; Please also see the example files.
...	parameter passed to microtable$new function of microeco package, such as auto_tidy parameter.

Value

microtable object.

Examples

```
library(microeco)
library(file2meco)
# use viromescan directory inside the package
dir_path <- system.file("extdata", "viromescan", package="file2meco")
d1 <- vs2meco(dir_path)
d1$cal_abund()
# d1$taxa_abund$Family is same with the percentage output of viromescan at
# Family level, i.e. Family_level_results-%.txt file
d1$cal_abund(rel = FALSE)
# d1$taxa_abund$Family is same with the count output of viromescan at
# Family level, i.e. Family_level_results-Counts.txt file
```

Index

∗ **Description**
file2meco, 4
∗ **data.frame**
CHOCOPhlAn_taxonomy, 3
MetaCyc_pathway_map, 8
ncyc_map, 13
pcyc_map, 13

check_match_table, 2
check_sample_table, 3
CHOCOPhlAn_taxonomy, 3

file2meco, 4

humann2meco, *4*, 4

meco2phyloseq, *4*, 7
meco2tse, 8
MetaCyc_pathway_map, 8
metacyc_pathway_website, 9
mpa2meco, *4*, 9

ncyc2meco, *4*, 11
ncyc_map, 13

pcyc_map, 13
phyloseq2meco, *4*, 13

qiime1meco, *4*, 14
qiime2meco, *4*, 15

tse2meco, *4*, 17

vs2meco, *4*, 17