Package: microbiomeutilities (via r-universe)

November 6, 2024

Type Package

Title microbiomeutilities: Utilities for Microbiome Analytics

Version 1.00.17

Maintainer Sudarshan Shetty <sudarshanshetty9@gmail.com>

Description This is a supporting tool for extending the functionality of both phyloseq and microbiome R packages.

License BSD_2_clause + file LICENSE

Encoding UTF-8

LazyData true

Depends R (>= 3.6.0), phyloseq, microbiome, ggplot2, dplyr

Imports reshape2, stats, tidyr, utils, gghalves, vegan, ggpubr, RColorBrewer, pheatmap, ggrepel, tibble, scales, Biostrings

Suggests BiocGenerics, patchwork, data.table, plyr, methods, knitr, rmarkdown, ape, viridis

BugReports https://github.com/microsud/microbiomeutilities/issues

VignetteBuilder knitr

RoxygenNote 7.2.1

Config/pak/sysreqs cmake libglpk-dev make libicu-dev libxml2-dev libssl-dev

Repository https://metagenlab.r-universe.dev

RemoteUrl https://github.com/microsud/microbiomeutilities

RemoteRef HEAD

RemoteSha 046a9f928f35fc2a02dd6cfb9175a0c7ddf65c4b

Contents

add_refseq *Add refseq slot for dada2 based phyloseq object*

Description

Utility to add refseq slot for dada2 based phyloseq object. Here, the rownames which are unique sequences, are stored in refseq slot of phyloseq. Sequence ids are converted to ids using tag option.

Usage

```
add_refseq(x, tag = "ASV")
```

Arguments

x	phyloseq-class object with seqs as rownames.
tag	Provide name for Ids, Default="ASV".

Value

phyloseq-class object

Examples

```
# ps <- add_refseq(p0,tag="ASV")
# ps
```

aggregate_top_taxa2 *Aggregate Top Taxa v2*

Description

Summarize phyloseq: combine other than the most abundant taxa.

Usage

```
aggregate_top_taxa2(x, top, level)
```

Arguments

x	phyloseq-class object
top	Keep the top-n taxa, and merge the rest under the category 'Other'. Instead of top-n numeric this can also be a character vector listing the groups to combine.
level	Summarization level (from rank_names(pseq))

Details

Backup from microbiome R pkg. This function is replaced by aggregate_rare function.

Value

phyloseq-class object

Author(s)

Contact: Leo Lahti <microbiome-admin@googlegroups.com>

References

See citation('microbiome')

Examples

```
data(dietswap)
s <- aggregate_top_taxa2(dietswap, top = 3, "Phylum")
```

---

dominant_taxa *Dominant Taxa*

Description

Identify dominant taxa in each sample and give overview.

Usage

```
dominant_taxa(x, level = NULL, group = NULL)
```

Arguments

x	phyloseq-class object
level	Taxonomic level uses microbiome::aggregate_taxa
group	Provide overview by groups. Default=NULL

Details

Identifies the dominant taxa in each sample and gives an overview of frequency and percent sample that are dominated by each taxon. Can be group wise or overall.

Value

A list of two data frames/tibbles

Examples

```
library(microbiomeutilities)
library(dplyr)
data("zackular2014")
p0 <- zackular2014
x.d <- dominant_taxa(p0, level = "Genus", group = "DiseaseState")
head(x.d$dominant_overview)
```

`find_samples_taxa` *Find samples dominated by specific taxa*

Description

Finding the samples dominated by user provided taxa in a phyloseq object. This is useful especially if user suspects a taxa to be contaminant and wishes to identify which samples are dominated by the contaminant taxa.

Usage

```
find_samples_taxa(x, taxa = NULL, relative = FALSE)
```

Arguments

`x`	phyloseq-class object
`taxa`	this should match the rownames of otu_table(x)
`relative`	Logical. If TRUE will transform input to relative abundance. Default=FALSE

Value

A character with sample ids.

Examples

```
library(microbiomeutilities)
data("zackular2014")
p0 <- zackular2014
p0.f <- aggregate_taxa(p0, "Genus")
bac_dom <- find_samples_taxa(p0.f, taxa = "g__Bacteroides")
# get samples dominated by g__Bacteroides
ps.sub <- prune_samples(sample_names(p0.f) %in% bac_dom, p0.f)
```

format_to_besthit *Formatting the* phyloseq-class *Object advanced*

Description

Format the phyloseq object to add the best taxonomy in phyloseq object (tax_table and otu_table).

Usage

```
format_to_besthit(x, prefix = NULL)
```

Arguments

x	phyloseq-class object
prefix	Prefered prefix e.g. OTU-d__denovo161:Roseburia or ASV-d__denovo161:Roseburia

Details

Most commonly it is observed that the taxonomy file has classification until a given taxonomic level. row.names for both tax_table and otu_table have best hit, until maximun genus level (species classification with short amplicons is a myth)is made available. This code is a slight modification of the code from **ampvis** phyloseq-class. Here, we directly take the phyloseq object as input and make the necessary formatting.

Value

phyloseq-class object

Examples

```
## Not run:
# Example data
library(microbiome)
library(microbiomeutilities)
library(dplyr)
data("zackular2014")
p0 <- zackular2014
p0.f <- format_to_besthit(p0, prefix = "OTU-")

## End(Not run)
```

get_group_abundances *Taxa abundance summary by group*

Description

Taxa abundance summary by group. Useful for description of microbiome.

Usage

```
get_group_abundances(x, level, group, transform = "compositional")
```

Arguments

x	phyloseq-class object
level	Taxonomic level uses microbiome::aggregate_taxa, if NULL with return OTU/ASVs level stats.
group	Provide overview by groups. Default=NULL and will return values for entire dataset, akin to taxa_summary.
transform	Either "compositional" or "counts". Default= compositional

Value

A data frames/ grouped tibble

Examples

```
## Not run:
library(microbiomeutilities)
data("zackular2014")
p0 <- zackular2014
get_group_abundances(p0, level = "Phylum", group = "DiseaseState")

## End(Not run)
```

get_microbiome_data *Download test microbiome data*

Description

Test microbiome data in phyloseq format.

Usage

```
get_microbiome_data(disease, study)
```

Arguments

disease	Disease of interest as shown in list_microbiome_data()
study	Name of the study as shown in list_microbiome_data()

Details

You can download few example datasets in phyloseq format from Duvallet et al 2017 https://www.nature.com/articles/s41467-017-01973-8.pdf?origin=ppub. The source file for these data is the microbiomedatarepo https://github.com/microsud/microbi

Value

phyloseq-class object.

Examples

```
## Not run:
# Example data
library(microbiome)
library(microbiomeUtilities)
list_microbiome_data()
ps1 <- get_microbiome_data(disease = "CDI", "Schubert2014_CDI")
print_ps(ps1)

## End(Not run)
```

---

get_tibble *Convert Phyloseq Slots to Tibbles*

Description

Utility to convert phyloseq slots to tibbles.

Usage

```
get_tibble(x, slot = "otu_table", column_id = "column_id")
```

Arguments

x	phyloseq-class object
slot	Must be one of c("otu_table", "sam_data", "tax_table"). Default= "otu_table"
column_id	Provide name for the column which will hold the rownames of slot.

Value

A tibble

Examples

```
library(microbiomeutilities)
data("zackular2014")
p0 <- zackular2014
otu_tibble <- get_tibble(p0,slot="otu_table",column_id="OTUID")
head(otu_tibble)
```

---

hmp2 *Test data 2*

Description

Data from a Stansfield J, Dozmorov M. "16s rRNA sequencing data from the Human Microbiome Project 2". Randomly choosen 13 particpants with multiple timepoints for rectum samples.

Usage

```
data("hmp2")
```

Format

An object of class "phyloseq".

References

- Stansfield J, Dozmorov M (2019). HMP2Data: 16s rRNA sequencing data from the Human Microbiome Project 2. R package version 1.1.0, https://bioconductor.org/packages/HMP2Data/

Examples

```
## Not run:
library(microbiomeutilities)
data("hmp2")
pseq <- hmp2
print(hmp2)

## End(Not run)
```

join_otu_tax *Join otu_table and tax_table to Tibble*

Description

Utility to join otu_table and tax_table to tibble.

Usage

```
join_otu_tax(x, column_id = "OTUID")
```

Arguments

x	phyloseq-class object
column_id	Provide name for the column which will hold the rownames of slot.

Value

A tibble

Examples

```
library(microbiomeutilities)
data("zackular2014")
p0 <- zackular2014
otu_tax <- join_otu_tax(p0,column_id = "OTUID")
head(otu_tax)
```

list_microbiome_data *List of available datasets*

Description

Data are used from Duvallet et al 2017 https://www.nature.com/articles/s41467-017-01973-8.pdf?origin=ppub.

Usage

```
list_microbiome_data(printtab = TRUE)
```

Arguments

printtab	Print in console or not, defaut is TRUE and will print output.

Details

Data for practice, also an example for importing mothur files from Baxtrer et al 2016. The source file for these data is the microbiomedatarepo https://github.com/microsud/microbiomedatarepo.

Examples

```
## Not run:
library(microbiomeutilities)

df <- list_microbiome_data(printtab = FALSE)

## End(Not run)
```

`make_pairs` *Make pairs*

Description

Creates a combination of variables for use with ggpubr::stat_compare_means.

Usage

```
make_pairs(x)
```

Arguments

`x`	list of vector to compare

Examples

```
# library(microbiomeutilities)
# data("zackular2014")
# pseq <- zackular2014
# comps <- make_pairs(meta(pseq)$DiseaseState)
```

`peak-methods` *Peak into* `phyloseq` *objects*

Description

These functions work on `otu_table`, `tax_table`, `sample_data` or on `data.frame` and `matrix`.

`peak_abundance` returns, user specified number of rows and columns for `otu_table`.

#' `peak_taxonomy` returns, user specified number of rows and columns for `tax_table`.

`peak_sample` returns, user specified number of rows and columns for `sample_data`.

`peak_base` returns, user specified number of rows and columns for `data.frame` and `matrix`.

Usage

```
peak_abundance(x, nrows = 1:5, ncols = 1:5)

peak_taxonomy(x, nrows = 1:5, ncols = 1:5)

peak_sample(x, nrows = 1:5, ncols = 1:5)

peak_base(x, nrows = 1:5, ncols = 1:5)

## S4 method for signature 'phyloseq'
peak_abundance(x, nrows = 1:5, ncols = 1:5)

## S4 method for signature 'phyloseq'
peak_taxonomy(x, nrows = 1:5, ncols = 1:5)

## S4 method for signature 'phyloseq'
peak_sample(x, nrows = 1:5, ncols = 1:5)

## S4 method for signature 'ANY'
peak_base(x, nrows = 1:5, ncols = 1:5)
```

Arguments

x	a phyloseq or data.frame or matrix object
nrows	number of rows, to be specified as numeric e.g. 1, or sequence of numeric specified as 1:5. to return first to fifth row.
ncols	number of cols, to be specified as numeric e.g. 1, or sequence of numeric specified as 1:5 to return first to fifth col.

Value

Print user specified rows and columns

Examples

```
data("zackular2014")

peak_abundance(zackular2014, nrows=1:3, ncols = 1:3)

peak_taxonomy(zackular2014, nrows=1:3, ncols = 1:3)

peak_sample(zackular2014, nrows=1:3, ncols = 1:3)

dat.frm <- meta(zackular2014)
# specify specific columns
peak_base(dat.frm, nrows=1:3, ncols = c(1, 3, 4))

matrix_ab <- abundances(zackular2014)
peak_base(matrix_ab, nrows=1:3, ncols = 1:3)
```

percent_classified *Summarize the percent taxa classification for* phyloseq-class

Description

Summarize the percent taxa classification for phyloseq-class.

Usage

```
percent_classified(x)
```

Arguments

x	phyloseq-class object

Value

Table with information on percent OTUs classified.

Author(s)

Contact: Sudarshan A. Shetty <sudarshanshetty9@gmail.com>

Examples

```
## Not run:
library(microbiomeutilities)
data("zackular2014")
pseq <- zackular2014
percent_classified(pseq)

## End(Not run)
```

phy_to_ldf *Convert* phyloseq-class *object to long data format*

Description

An alternative to psmelt function from phyloseq-class object.

Usage

```
phy_to_ldf(x, transform.counts)
```

Arguments

x	phyloseq-class object
transform.counts	Data transform to be used in plotting (but not in sample/taxon ordering). The options are 'Z-OTU', 'Z-Sample', 'log10' and 'compositional'. See the transform function

Value

A data frame in long format with appropriate transfomation if requested

Examples

```
## Not run:
# Example data
library(microbiomeutilities)
data("zackular2014")
pseq <- zackular2014
pseq_df <- phy_to_ldf(pseq, transform.counts = NULL)

## End(Not run)
```

---

plasticity *Plasticity*

Description

Calculated difference in microbiota composition for each individual between two timepoints.

Usage

```
plasticity(x, dist.method = "bray", participant.col)
```

Arguments

x	phyloseq-class object
dist.method	Any of the methods supported by phyloseq::distance or correlation method cor()
participant.col	Column name with participant IDs

Details

Using a beta diversity metrics or correlation matrix to identify variability in microbiota of an individual. The code is slight modification from Grembi et. al. see ref below. This is useful for instance if one wants to quantiify changes in microbiota before and after a treatment, dietary modulation, antibiotic treatment, etc. The choice of index is important. For example, Bray-Curtis dissimilarity, the higher values mean higher plasticity/variability. On the contrary, higer spearman correlation values mean lower plasticity.

Value

plot

References

- Grembi, J.A., Nguyen, L.H., Haggerty, T.D. et al. Gut microbiota plasticity is correlated with sustained weight loss on a low-carb or low-fat dietary intervention. Sci Rep 10, 1405 (2020).https://www.nature.com/articles/s41598-020-58000-y

Examples

```
## Not run:
library(microbiome)
library(microbiomeutilities)
library(dplyr)
library(ggpubr)
data(peerj32)
pseq <- peerj32$phyloseq
pseq.rel <- microbiome::transform(pseq, "compositional")
pl <- plasticity(pseq.rel, participant.col = "subject")

## End(Not run)
```

---

plot_abund_prev *Mean Abundance-Prevalence relation*

Description

Plots Mean Abundance-Prevalence for taxa. Mean abundance, mean prevalence, and upper and lower confidence interval for each taxa is calculated by random subsampling.

Usage

```
plot_abund_prev(
  x,
  lower.conf = 0.025,
  upper.conf = 0.975,
  bs.iter = 99,
  color = "steelblue",
  dot.opacity = 0.5,
  dot.size = 2,
  label.core = FALSE,
  label.size = NULL,
  label.opacity = NULL,
  label.color = "grey70",
  mean.abund.thres = NULL,
  mean.prev.thres = NULL,
  log.scale = TRUE,
```

```
  nudge.label = NULL,
  ...
)
```

Arguments

x	phyloseq-class object
lower.conf	Lower confidence interval =0.025
upper.conf	Upper confidence interval =0.975
bs.iter	Number of bootstrap iterations =99
color	taxa level to color. Preferably at phylum or just a single color
dot.opacity	Numeric for ggplot alpha. Default is 0.5
dot.size	Numeric size of point
label.core	Logical default is FALSE
label.size	If label_core is TRUE specify text size. Default is NULL
label.opacity	Numeric for ggplot alpha. Defualt is NULL
label.color	Color for labels. Default="grey70"
mean.abund.thres	If label_core is TRUE specify mean abundance threshold. Default is NULL
mean.prev.thres	If label_core is TRUE specify mean prevalence threshold. Default is NULL
log.scale	Plot log10 scale. Default is TRUE abundance criteria. Default is NULL
nudge.label	Argument to pass to ggrepel::geom_text_repel Default is NULL
...	Arguments to pass to sample() function.

Details

Check if there are spurious OTUs/ASVs.

Value

A ggplot plot object.

Examples

```
## Not run:
# Example data
library(microbiomeutilities)
asv_ps <- zackular2014
asv_ps <- microbiome::transform(asv_ps, "compositional")
asv_ps <- core(asv_ps, detection = 0.0001, prevalence = 0.5)
asv_ps <- format_to_besthit(asv_ps)
set.seed(2349)
p_v <- plot_abund_prev(asv_ps, size = 20, replace = TRUE) +
  geom_vline(xintercept = 0.75, lty = "dashed", alpha = 0.7) +
  geom_hline(yintercept = 0.01, lty = "dashed", alpha = 0.7) +
```

```
  scale_color_brewer(palette = "Paired")
p_v

## End(Not run)
```

`plot_alpha_diversities` *Creat a plot for alpha diversities calculated using the* `microbiome` *package*

Description

Utility plot function for diversity measures calcualted by `microbiome` package.

Usage

```
plot_alpha_diversities(
  x,
  type,
  index.val = "all",
  plot.type,
  variableA,
  palette
)
```

Arguments

`x`	`phyloseq-class` object.
`type`	Either alpha (Diversity Index) or dominance (Dominance Index) or evenness (Evenness Index)
`index.val`	see global function in `microbiome` package
`plot.type`	Three optons c("stripchart", "boxplot", "violin")
`variableA`	Variable of interested to be checked. This will also be used to color the plot
`palette`	Any of the `RColorBrewer` plettes

Details

Uses the `microbiome` package global function to calculate diversities and then returns a plot.

Value

`ggplot` object. This can be further modified using ggpubr

Examples

```
library(microbiome)
library(microbiomeutilities)
data("zackular2014")
p0 <- zackular2014
p <- plot_alpha_diversities(p0,
  type = "dominance",
  index.val = "all",
  plot.type = "stripchart",
  variableA = "DiseaseState",
  palette = "jco"
)

print(p)
```

`plot_alpha_rcurve` *Rarefaction curves for alpha diversity indices*

Description

Calculates alpha diversity idenx at varying sampling units (sequencing depth).

Usage

```
plot_alpha_rcurve(
  x,
  index = "observed",
  subsamples = c(100, 1000, 2000, 3000, 4000, 5000),
  lower.conf = 0.025,
  upper.conf = 0.975,
  group = NULL,
  linetype.main = 1,
  line.opacity.main = 0.5,
  linetype.type = 2,
  line.opacity.type = 0.25,
  type = "CI",
  label.min = TRUE,
  label.size = 3,
  label.color = "grey70"
)
```

Arguments

x	phyloseq-class object
index	Default:: "observed",
subsamples	Default: c(100,1000, 2000, 3000, 4000, 5000)
lower.conf	Default: 0.025. If type=CI

upper.conf	Default: 0.975.
group	Default: NULL
linetype.main	For ggplot line type for line by group. Default: 1
line.opacity.main	For ggplot alpha to determine opacity for line by group. Default: 0.5
linetype.type	For ggplot line type for line CI or SD. Default: 2
line.opacity.type	For ggplot line type to determine opacity for line CI or SD. Default: 0.25
type	Either CI (confidence interval) or SD (Standard deviation) Default: CI
label.min	TRUE or FALSE. Default: TRUE
label.size	Label min size
label.color	Label min color

Examples

```
## Not run:
library(microbiomeutilities)
data("zackular2014")
p0 <- zackular2014
# e.g. to make range of
# subsamples <- seq(0, 5000, by=100)[-1]
p <- plot_alpha_rcurve(p0, index="observed",
lower.conf = 0.025, upper.conf = 0.975,
group="DiseaseState") +
scale_color_brewer(palette = "Paired") +
scale_fill_brewer(palette = "Paired")
print(p )

## End(Not run)
```

---

plot_area *Area plot*

Description

Create an area plot for longitudinal samples with ggplot2 package.

Usage

```
plot_area(
  x,
  xvar = NULL,
  level = NULL,
  facet.by = NULL,
  fill.colors = brewer.pal(6, "Paired"),
  abund.thres = 0.001,
```

```
  prev.thres = 0.5,
  ncol = 5,
  nrow = 5
)
```

Arguments

x	phyloseq-class object.
xvar	Column name to plot on x-axis.
level	Taxonomic level. OTU/ASV level not supported.
facet.by	Column with variable that has multiple measurements.
fill.colors	brewer.pal(6,"Paired"). Specify colors.
abund.thres	= 0.001 check microbiome package aggregate_rare function.
prev.thres	= 0.1 check microbiome package aggregate_rare function.
ncol	wrap, specify number of columns.
nrow	wrap, specify number of rows.

Value

ggplot object.

Examples

```
## Not run:
library(microbiomeutilities)
data("hmp2")
ps <- hmp2
ps.rel <- microbiome::transform(ps, "compositional")
p <- plot_area(ps.rel, xvar="visit_number",
              level = "Phylum",
              facet.by = "subject_id",
              fill.colors=brewer.pal(6,"Paired"))

## End(Not run)
```

---

plot_diversity_stats *Diversity plot with stats*

Description

Diversity plot with stats

Usage

```
plot_diversity_stats(
  x,
  index,
  group = NULL,
  group.colors = c("brown3", "steelblue"),
  dot.opacity = 0.25,
  box.opacity = 0.25,
  violin.opacity = 0.5,
  group.order = NULL,
  stats = TRUE,
  label.format = "p.format",
  ...
)
```

Arguments

x	phyloseq-class object
index	diversity index. Calculated using microbiome::alpha
group	Grouping variable to compare
group.colors	Colors for plotting groups
dot.opacity	for ggplot alpha to determine opacity for points
box.opacity	for ggplot alpha to determine opacity for box
violin.opacity	for ggplot alpha to determine opacity for violin
group.order	Default is NULL. a list specifing order of x-axis.
stats	Logical TRUE or FALSE. Calls ggpubr::stat_compare_means.
label.format	For ggpubr::stat_compare_means "p.signif" E.g. c("H","CRC","nonCRC")
...	params for ggpubr::stat_compare_means

Examples

```
## Not run:
library(microbiomeutilities)
library(ggpubr)
data("zackular2014")
p0 <- zackular2014
mycols <- c("brown3", "steelblue", "grey50")
p.m <- plot_diversity_stats(p0,
  group = "DiseaseState",
  index = "diversity_shannon",
  group.order = c("H", "CRC", "nonCRC"),
  group.colors = mycols
)
print(p.m)

## End(Not run)
```

`plot_listed_taxa` *A boxplot for user specified list of taxa*

Description

User specifed OTUs are plotted.

Usage

```
plot_listed_taxa(
  x,
  select.taxa,
  group,
  group.colors,
  dot.opacity = 0.25,
  box.opacity = 0.25,
  group.order = NULL,
  add.violin = TRUE,
  violin.opacity = 0.25,
  panel.arrange = "grid",
  ncol = NULL,
  nrow = NULL
)
```

Arguments

`x`	phyloseq-class object.
`select.taxa`	a character list of taxa to be plotted. eg. select.taxa <- c("OTU-370251", "OTU-311173", "OTU-341024").
`group`	Grouping variable to compare
`group.colors`	Colors for plotting groups
`dot.opacity`	For ggplot alpha to determine opacity for points
`box.opacity`	For ggplot alpha to determine opacity for box
`group.order`	Default is NULL. a list specifing order of x-axis.
`add.violin`	Loical. If half violoin to the added. Default=TRUE
`violin.opacity`	If add.violin=TRUE, opacity for violin.
`panel.arrange`	panels "grid" or "wrap" ggplot's facet_XXX
`ncol`	if wrap, specify number of columns.
`nrow`	if wrap, specify number of rows.

Details

Useful for instances where user is interested only in some OTUs. For example OTUs reported to be significantly diferent. This can also be used at higher taxonomic levels, output from phyloseq::tax_glom or microbiome::aggregate_taxa.

Value

ggplot object. This can be further modified using ggpubr.

Examples

```
## Not run:
# Example data
library(microbiome)
library(microbiomeutilities)
data("zackular2014")
p0 <- zackular2014
p0.f <- format_to_besthit(p0)
select.taxa <- c("OTU-d__denovo31:Dorea", "OTU-d__denovo24:Blautia")
mycols <- c("brown3", "steelblue", "grey50")
p <- plot_listed_taxa(p0.f, select.taxa,
  group = "DiseaseState",
  add.violin = TRUE,
  group.colors = mycols
)
print(p)

## End(Not run)
```

---

plot_ordination_utils *Plot species loading with ordinations*

Description

This function extends the plot_ordination function of phyloseqto highlight the top taxa loadings on the species ordination.

Usage

```
plot_ordination_utils(
  x,
  ordiObject,
  color.opt = NULL,
  plot.arrow = TRUE,
  scale.arrow = NULL,
  top.taxa = 5
)
```

Arguments

x	phyloseq-class object
ordiObject	Output of ordinate from package phyloseq. Only NMDS/CCA and Bray supported.
color.opt	Variable of interest from metadata.

plot.arrow	If arrow should be plotted for species either TRUE or FALSE.
scale.arrow	If arrow is plotted a constant to multiply axis values for clearing visualisations.
top.taxa	Top varying taxa to plot, default is 5.

Details

This function is useful for visualizing specifc taxa that could be important in explaining variations in ordinations.

Value

plot

Examples

```
## Not run:

library(microbiomeutilities)
library(RColorBrewer)
data("zackular2014")
ps1 <- zackular2014
ps2 <- tax_glom(ps1, "Genus")
ps2f <- format_to_besthit(ps2)
orddi <- ordinate(ps2f, method = "CCA", distance = "bray")
p <- plot_ordination_utils(ps2f, orddi,
  color = "DiseaseState", plot.arrow = TRUE,
  scale.arrow = 1.3, top.taxa = 5
)
print(p)

## End(Not run)
```

---

plot_ordiplot_core *Plotting core microbiota on ordinations*

Description

This function will plot the ordaination along with highligthing the core microbes on the species ordination.

Usage

```
plot_ordiplot_core(
  x,
  ordiObject,
  prevalences,
  detections,
  min.prevalence,
```

```
  color.opt,
  shape,
  Samples = c(TRUE, FALSE)
)
```

Arguments

x	phyloseq-class object
ordiObject	Output of ordinate from package phyloseq. Only NMDS and Bray supported.
prevalences	Prevalences as supported by microbiome package.
detections	Detections as supported by microbiome package.
min.prevalence	Minimum prevalence value to plot.
color.opt	Variable of interest from metadata.
shape	Variable of interest from metadata.
Samples	c("TRUE" or "FALSE")

Details

This function is useful for visualizing core taxa in a 2D ordination plot.

Value

plot

Examples

```
## Not run:

library(microbiomeutilities)
library(RColorBrewer)
data("zackular2014")
p0 <- zackular2014
ps1 <- format_to_besthit(p0)
ps1 <- subset_samples(ps1, DiseaseState == "H")
ps1 <- prune_taxa(taxa_sums(ps1) > 0, ps1)
prev.thres <- seq(.05, 1, .05)
det.thres <- 10^seq(log10(1e-4), log10(.2), length = 10)
pseq.rel <- microbiome::transform(ps1, "compositional")
ord.bray <- ordinate(pseq.rel, "NMDS", "bray")

p <- plot_ordiplot_core(pseq.rel, ord.bray,
  prev.thres, det.thres,
  min.prevalence = 0.9,
  color.opt = "DiseaseState", shape = NULL, Sample = TRUE
)
p

## End(Not run)
```

`plot_paired_abundances` *A paired-boxplot for user specified list of taxa*

Description

User specified taxa are plotted.

Usage

```
plot_paired_abundances(
  x,
  select.taxa = NULL,
  group = NULL,
  group.colors = NULL,
  dot.opacity = 0.25,
  dot.size = 2,
  add.box = FALSE,
  box.opacity = 0.25,
  group.order = NULL,
  add.violin = TRUE,
  violin.opacity = 0.25,
  ncol = NULL,
  nrow = NULL,
  line = NULL,
  line.down = "#7209b7",
  line.stable = "#8d99ae",
  line.up = "#14213d",
  line.na.value = "grey50",
  line.guide = "legend",
  line.opacity = 0.25,
  line.size = 1,
  jitter.width = 0
)
```

Arguments

x	phyloseq-class object.
select.taxa	a character list of taxa to be plotted. eg. select.taxa <- c("OTU-370251", "OTU-311173", "OTU-341024").
group	Grouping variable to compare. x axis, eg. before-after, t1-t2.
group.colors	Colors for plotting groups.
dot.opacity	For ggplot alpha to determine opacity for points.
dot.size	For ggplot point size.
add.box	Logical. If boxplot to the added. Default=TRUE

box.opacity	For ggplot alpha to determine opacity for box.
group.order	Default is NULL. a list specifying order of x-axis.
add.violin	Logical. If half violin to the added. Default=TRUE
violin.opacity	If add.violin=TRUE, opacity for violin.
ncol	If 2 or more taxa to plot, specify number of columns.
nrow	If 2 or more taxa to plot, specify number of rows.
line	Variable to use for lines. E.g. "subject" before-after
line.down	Line Color for change when negative. Decreased abundance.
line.stable	Line Color for no change.
line.up	Line Color for change when positive. Increased abundance.
line.na.value	"grey50" for no/missing observations.
line.guide	"none" to plot guide for line.
line.opacity	Line opacity.
line.size	Size of line to plot.
jitter.width	Value to avoid over plotting by moving points.

Details

Useful for instances where user is interested only in some taxa and thier change after an intervention. This can also be used at higher taxonomic levels, output from phyloseq::tax_glom or microbiome::aggregate_taxa.

Value

ggplot object. This can be further modified using ggpubr.

Examples

```
## Not run:
library(microbiome)
library(microbiomeutilities)
library(gghalves)
library(tidyr)
data(peerj32) # Source: https://peerj.com/articles/32/
pseq <- peerj32$phyloseq # Ren
pseq.rel <- microbiome::transform(pseq, "compositional")
select.taxa <- c("Akkermansia", "Dialister")
group.colors <- c("brown3", "steelblue", "grey70")
p <- plot_paired_abundances(pseq.rel,
  select.taxa = select.taxa,
  group = "time",
  group.colors = group.colors,
  dot.opacity = 0.25,
  dot.size = 2,
  group.order = NULL,
  line = "subject"
```

```
)
p

## End(Not run)
```

`plot_read_distribution` *Distribution of reads*

Description

Plots distribution of reads.

Usage

```
plot_read_distribution(x, groups, plot.type = c("density", "histogram"))
```

Arguments

`x`	`phyloseq-class` object
`groups`	Metadata variable to check for groups based sequencing effort.
`plot.type`	Either density or histogram plot

Value

A `ggplot` plot object.

Author(s)

Contact: Sudarshan Shetty <sudarshanshetty9@gmail.com>

Examples

```
library(microbiome)
data(zackular2014)
ps0 <- zackular2014
p <- plot_read_distribution(ps0, groups = "DiseaseState", plot.type = "density")
print(p)
```

plot_select_taxa *A boxplot for user specified list of taxa*

Description

User specifed OTUs are plotted.

Usage

```
plot_select_taxa(
  x,
  select.taxa,
  variableA,
  palette,
  plot.type,
  group.order = NULL
)
```

Arguments

x	phyloseq-class object.
select.taxa	a character list of taxa to be plotted. eg. select.taxa <- c("OTU-370251", "OTU-311173", "OTU-341024").
variableA	Variable of interested to be checked. This will also be used to color the plot.
palette	Any of the RColorBrewer plettes.
plot.type	Three optons c("stripchart", "boxplot", "violin")
group.order	Default is NULL. a list specifing order of x-axis. E.g. c("H","CRC","nonCRC")

Details

Useful for instances where user is interested only in some OTUs. For example OTUs reported to be significantly diferent.

Value

ggplot object. This can be further modified using ggpubr.

Examples

```
## Not run:
# Example data
library(microbiome)
library(microbiomeutilities)
data("zackular2014")
p0 <- zackular2014
p0.f <- format_to_besthit(p0)
select.taxa <- c("OTU-d__denovo31:Dorea", "OTU-d__denovo24:Blautia")
```

```
p <- plot_select_taxa(p0.f, select.taxa, "DiseaseState", "Paired", plot.type = "stripchart")
print(p)

## End(Not run)
```

`plot_spaghetti` *Spaghetti Plots*

Description

Spaghetti Plots

Usage

```
plot_spaghetti(
  x,
  plot.var = "by_taxa",
  select.taxa = NULL,
  xvar = NULL,
  group = NULL,
  line.bg.color = "#8d99ae",
  bg.opacity = 0.5,
  focus.color = "brown3",
  ncol = NULL,
  nrow = NULL,
  focus.line.size = 0.5,
  line.size = 1
)
```

Arguments

`x`	phyloseq-class object
`plot.var`	One of "by_sample":Many sample one taxa or "by_taxa":Many taxa one sample
`select.taxa`	a character list of taxa to be plotted. eg. select.taxa <- c("OTU-370251", "OTU-311173", "OTU-341024").
`xvar`	X-axis variable. This can be visit_number for participants in subject_id.
`group`	Column describing sample that have multiple measurements. For e.g. Subject_Ids which has all subject_ids listed.
`line.bg.color`	Line color for background. Default ="#8d99ae",
`bg.opacity`	Line opacity for background.
`focus.color`	Line color for focus being plotted.
`ncol`	if wrap, specify number of columns.
`nrow`	if wrap, specify number of rows.
`focus.line.size`	Line size for focus being plotted.
`line.size`	Line size for background.

Value

ggplot object. This can be further modified using ggpubr.

Examples

```
## Not run:
# Example data
library(microbiomeutilities)
data("hmp2")
pseq <- hmp2 # Ren
pseq.rel <- microbiome::transform(pseq, "compositional")
pseq.relF <- format_to_besthit(pseq.rel)
# Choose one participant
phdf.s <- subset_samples(pseq.relF, subject_id ==
  "Participant_1")
# Choose top 12 taxa to visualize
ntax <- top_taxa(phdf.s, 12)
phdf.s <- prune_taxa(ntax, phdf.s)

plot_spaghetti(phdf.s,
  plot.var = "by_taxa",
  select.taxa = ntax,
  xvar = "visit_number",
  line.bg.color = "#8d99ae",
  focus.color = "#555b6e",
  ncol = 3,
  nrow = 4,
  line.size = 0.2
)

print(p)

## End(Not run)
```

---

plot_taxa_boxplot *Taxonomic Composition Plot boxplot*

Description

Plot taxon abundance for samples.

Usage

```
plot_taxa_boxplot(
  x,
  taxonomic.level,
  top.otu,
  keep.other = FALSE,
  group,
```

```
  title,
  group.colors = NULL,
  group.order = NULL,
  add.violin = TRUE,
  violin.opacity = 0.25,
  box.opacity = 0.25,
  dot.opacity = 0.25,
  dot.size = 2
)
```

Arguments

x	phyloseq-class object
taxonomic.level	Merge the OTUs (for phyloseq object) into a higher taxonomic level. This has to be one from colnames(tax_table(x)).
top.otu	Top number of taxa to plot.
keep.other	TRUE or FALSE. Default is FALSE. This will not plot taxa group as Other
group	Specify main variable of interest. This should be one of the variables in sample_variables(x).
title	title for the plot
group.colors	Colors for plotting groups
group.order	Default is NULL. a list specifing order of x-axis. E.g. c("H","CRC","nonCRC")
add.violin	Loical. If half violoin to the added. Default=TRUE
violin.opacity	If add.violin=TRUE, opacity for violin.
box.opacity	For ggplot alpha to determine opacity for box
dot.opacity	For ggplot alpha to determine opacity for points
dot.size	For ggplot alpha to determine size for points

Value

A ggplot plot object.

Examples

```
## Not run:
# Example data
library(microbiomeutilities)
library(RColorBrewer)
data("zackular2014")
ps0 <- zackular2014
mycols <- c("brown3", "steelblue", "grey50")
pn <- plot_taxa_boxplot(ps0,
  taxonomic.level = "Phylum",
  top.otu = 6,
  group = "DiseaseState",
```

```
  title = "Relative abudance plot",
  keep.other = FALSE,
  group.order = c("H", "CRC", "nonCRC"),
  group.colors = mycols
)
print(pn + theme_biome_utils())

## End(Not run)
```

`plot_taxa_composition` *Taxonomic Composition Plot*

Description

Plot taxon abundance for samples. It is a legacy function from microbiome.

Usage

```
plot_taxa_composition(
  x,
  sample.sort = NULL,
  taxonomic.level = "Phylum",
  transform = "compositional",
  otu.sort = NULL,
  palette = brewer.pal(12, "Paired"),
  x.label = "sample",
  plot.type = "barplot",
  average_by = NULL,
  verbose = FALSE,
  mar = c(5, 12, 1, 1),
  ...
)
```

Arguments

x	phyloseq-class object
sample.sort	Order samples. Various criteria are available: • NULL or 'none': No sorting • A single character string: indicate the metadata field to be used for ordering • A character vector: sample IDs indicating the sample ordering. • 'neatmap' Order samples based on the neatmap approach. See neatsort. By default, 'NMDS' method with 'bray' distance is used. For other options, arrange the samples manually with the function.
taxonomic.level	Merge the OTUs (for phyloseq object) into a higher taxonomic level. This has to be one from colnames(tax_table(x)).

transform	Data transform to be used in plotting (but not in sample/taxon ordering). The options are 'Z-OTU', 'Z-Sample', 'log10' and 'compositional'. See the transform function.
otu.sort	Order taxa. Same options as for the sample.sort argument but instead of metadata, taxonomic table is used. Also possible to sort by 'abundance'.
palette	The number and palette RColorBrewer has to be specified e.g brewer.pal(12, "Paired").
x.label	Specify how to label the x axis. This should be one of the variables in sample_variables(x).
plot.type	Plot type: 'barplot' or 'lineplot'.
average_by	Variable to group.
verbose	verbose.
mar	Figure margins.
...	Arguments to be passed (for neatsort function)

Value

A ggplot plot object.

Examples

```
## Not run:
# Example data
library(microbiome)
library(microbiomeutilities)
data("biogeogut")
pseq <- biogeogut
plot_taxa_composition(pseq, taxonomic.level = "Phylum")

## End(Not run)
```

---

plot_taxa_cv *Coefficient of variations*

Description

Plots CV for OTUs/ASVs.

Usage

```
plot_taxa_cv(x, plot.type)
```

Arguments

x	phyloseq-class object.
plot.type	scatter or hist (histogram)

Details

Check if there are spurious OTUs/ASVs.

Value

A ggplot plot object.

Examples

```
## Not run:
# Example data
library(microbiome)
library(microbiomeutilities)
data("zackular2014")
p0 <- zackular2014
p <- plot_taxa_cv(p0, plot.type = "hist")
print(p)

## End(Not run)
```

---

plot_taxa_heatmap *Heatmap using* phyloseq-class *and* pheatmap

---

Description

Plot heatmap using phyloseq-class object as input.

Usage

```
plot_taxa_heatmap(
  x,
  subset.top,
  transformation,
  VariableA,
  heatcolors = NULL,
  ...
)
```

Arguments

x	phyloseq-class object.
subset.top	either NA or number of Top OTUs to use for plotting.
transformation	either 'log10', 'clr','Z', 'compositional', or NA
VariableA	main variable of Interest.
heatcolors	is the option for colors in pheatmap. Default is to use Spectral
...	Arguments to be passed pheatmap.

Value

A pheatmap plot object.

Author(s)

Sudarshan A. Shetty (sudarshanshetty9@gmail.com)

Examples

```
library(microbiomeutilities)
library(viridis)
library(RColorBrewer)
data("zackular2014")
ps0 <- zackular2014

heat.sample <- plot_taxa_heatmap(ps0,
  subset.top = 20,
  VariableA = "DiseaseState",
  heatcolors = colorRampPalette(rev(brewer.pal(n = 7, name = "RdYlBu")))(100),
  transformation = "log10"
)
```

---

prep_ternary *Create table for Ternary plot*

Description

Create a table for ternary plot ggtern R package.

Usage

```
prep_ternary(
  x,
  abund.thres = 1e-04,
  prev.thres = 0.1,
  group = NULL,
  level = "lowest"
)
```

Arguments

x	phyloseq-class object
abund.thres	= 0.0001 check microbiome package core function remove taxa that are dectected at 0.0001 in less than prev.thres of samples
prev.thres	= 0.1 check microbiome package core function
group	Grouping variable to compare, for this plot there has to be three groups in the data
level	= "Genus" Taxonomic level. If OTU/ASV level specify="lowest" Does not support phylum level aggregation

Details

Plots the mean relative abundance of taxa in 3 groups being compared.

Value

Tibble object.

Examples

```
library(microbiome)
library(microbiomeutilities)
library(dplyr)
data("zackular2014")
p0 <- zackular2014
prep_ternary(p0, group = "DiseaseState", abund.thres = 0.0001, level = "Genus", prev.thres = 0.25)
```

---

prep_tern_otu *Create table for ternary plot OTU*

Description

Create a table for ternary plot ggtern package.

Usage

```
prep_tern_otu(x, abund.thres = 1e-04, prev.thres = 0.1, group = NULL)
```

Arguments

x	phyloseq-class object.
abund.thres	= 0.0001 check microbiome package core function. remove taxa that are detected at 0.0001 in less than prev.thres of samples.
prev.thres	= 0.1 check microbiome package core function.
group	Grouping variable to compare, for this plot there has to be three groups in the data

Details

Plots the mean relative abundance of taxa in 3 groups being compared.

Value

tibble object.

Examples

```
# library(microbiome)
# library(microbiomeutilities)
# library(dplyr)
# data("zackular2014")
# p0 <- zackular2014
# prep_tern_otu(p0, group="DiseaseState",
# abund.thres=0.0001, prev.thres=0.25)
```

---

print_ps *Overview of* phyloseq-class

Description

Prints an overview phyloseq-class.

Usage

```
print_ps(x)
```

Arguments

x	phyloseq-class object

Value

Prints information about the phyloseq-class object.

Author(s)

Contact: Sudarshan A. Shetty <sudarshanshetty9@gmail.com>

Examples

```
library(microbiomeutilities)
data("zackular2014")
pseq <- zackular2014
print_ps(pseq)
```

simple_heatmap *Simple Heatmap*

Description

Create a simple heatmap with ggplot2 package.

Usage

```
simple_heatmap(
  x,
  group.facet = "DiseaseState",
  group.order = c("H", "CRC", "nonCRC"),
  abund.thres = 0.01,
  prev.thres = 0.1,
  level = "Genus",
  scale.color = "log10",
  na.fill = "white",
  color.fill = NULL,
  taxa.arrange = TRUE,
  panel.arrange = NULL,
  remove.other = TRUE,
  ncol = NULL,
  nrow = NULL
)
```

Arguments

x	phyloseq-class object.
group.facet	Variable to make facet/panel the plot.
group.order	Default is NULL. a list specifing order of x-axis. E.g. c("H","CRC","nonCRC")
abund.thres	= 0.01 check microbiome package aggregate_rare function.
prev.thres	= 0.1 check microbiome package aggregate_rare function.
level	= "Genus" Taxonomic level. OTU/ASV level not supported. Check plot_taxa_heatmap
scale.color	Scale the tiles colors "log10" or "sqrt"
na.fill	Color to fill NAs. e.g. "white"
color.fill	User specified color vectors.
taxa.arrange	Arrange the order of taxa. User can supply a list of vectors.
panel.arrange	panels "grid" or "wrap" ggplot's facet_XXX
remove.other	Rare clubbed as Other to be removed. Logical TRUE/FALSE.
ncol	if wrap, specify number of columns.
nrow	if wrap, specify number of rows.

Details

Wrapper converts phyloseq-class object to long data frame and generates a heatmap.

Value

ggplot object.

Examples

```
library(microbiome)
library(microbiomeutilities)
library(dplyr)
data("zackular2014")
p0 <- zackular2014
p0.rel <- transform(p0, "compositional")
p <- simple_heatmap(p0.rel,
  group.facet = "DiseaseState",
  group.order = c("H", "CRC", "nonCRC"),
  abund.thres = 0.01,
  prev.thres = 0.1,
  level = "Genus",
  scale.color = "log10",
  na.fill = "white",
  color.fill = NULL,
  taxa.arrange = TRUE,
  remove.other = TRUE,
  panel.arrange = "wrap",
  ncol = 2,
  nrow = 2
)

print(p)
```

---

taxa_distribution *Distribution of taxa*

Description

Plots distribution of taxa.

Usage

```
taxa_distribution(
  x,
  color.level = "Phylum",
  color.taxa = brewer.pal(12, "Paired")
)
```

Arguments

x	phyloseq-class object
color.level	Taxonomic level to color
color.taxa	vector of colors specified by user Default is brewer.pal(12,"Paired")

Value

ggplot2 object

Examples

```
library(microbiomeutilities)
data("zackular2014")
pseq <- zackular2014
p <- taxa_distribution(pseq)
p
```

---

taxa_pooler_mcola *Pool Taxa*

Description

Creates a list of dataframes at different taxonomic levels.

Usage

```
taxa_pooler_mcola(x)
```

Arguments

x	phyloseq-class object

Examples

```
# library(phyloseq)
# library(microbiome)
# data("zackular2014")
# sub.sm <- sample(sample_names(zackular2014), 20)
# pseq <- prune_samples(sub.sm,zackular2014)
# taxa_pooler_mcola(pseq)
```

taxa_summary *Give taxa summary at specified taxonomic level*

Description

Data frame with mean, max, median standard deviation of relative abundance.

Usage

```
taxa_summary(x, level)
```

Arguments

x	phyloseq-class object
level	Taxonomic level for which summary is required

Value

returns a data frame with relative abundance summary.

Author(s)

Contact: Sudarshan A. Shetty <sudarshanshetty9@gmail.com>

Examples

```
## Not run:
# Example data
library(microbiomeutilities)
data("zackular2014")
p0 <- zackular2014
p0.rel <- microbiome::transform(p0, "compositional")
tx.sum1 <- taxa_summary(p0, "Phylum")

tx.sum2 <- taxa_summary(p0.rel, "Phylum")

## End(Not run)
```

theme_biome_utils *Custom theme for microbiomeutilities pkg*

Description

Opiniated elegant theme.

Usage

```
theme_biome_utils()
```

zackular2014 *Test data*

Description

Data from a Zackular, Joseph P., et al. "The gut microbiome modulates colon tumorigenesis." MBio 4.6 (2013): e00692 -13.

Usage

```
data("zackular2014")
```

Format

An object of class "phyloseq".

References

- Zackular, Joseph P., et al. "The gut microbiome modulates colon tumorigenesis." MBio 4.6 (2013): e00692-13., https://mbio.asm.org/content/4/6/e00692-13.short

Examples

```
## Not run:
library(microbiomeutilities)
data("zackular2014")
pseq <- zackular2014
print(zackular2014)

## End(Not run)
```

Index