Package 'TaxaNorm'

December 12, 2023

Title Feature-Wise Normalization for Microbiome Sequencing Data

Version 2.4

Maintainer Dillon Lloyd <dtlloyd@ncsu.edu>

Description A novel feature-wise normalization method based on a zero-inflated negative binomial model. This method assumes that the effects of sequencing depth vary for each taxon on their mean and also incorporates a rational link of zero probability and taxon dispersion as a function of sequencing depth. Ziyue Wang, Dillon Lloyd, Shanshan Zhao, Alison Motsinger-Reif (2023) <doi:10.1101/2023.10.31.563648>.

License GPL-3

Depends R (>= 4.0.0), microbiome,

Imports phyloseq, stats, S4Vectors, BiocGenerics, vegan, methods, MASS, future, future.apply, matrixStats, pscl, parallelly, ggplot2, utils

URL https://github.com/wangziyue57/TaxaNorm

biocViews Sequencing, Microbiome, Metagenomics, Normalization, Visualization

Suggests rmarkdown, knitr

VignetteBuilder knitr

Encoding UTF-8

RoxygenNote 7.2.3

BugReports https://github.com/wangziyue57/TaxaNorm/issues

NeedsCompilation no

Author Ziyue Wang [aut],
Dillon Lloyd [aut, cre, cph],
Shanshan Zhao [aut, ctb],
Alison Motsinger-Reif [aut, ctb]

Repository CRAN

Date/Publication 2023-12-12 18:30:05 UTC

R topics documented:

TaxaNorm-datasets 2
TaxaNormGenerics 3
TaxaNorm_Model_Parameters-class 4
TaxaNorm_Model_QC 6
TaxaNorm_NMDS 6
TaxaNorm_Normalization 7
TaxaNorm_QC_Input 8
TaxaNorm_Results-class 9
TaxaNorm_Run_Diagnose 11

Index **13**

`TaxaNorm-datasets` *TaxaNorm data objects*

Description

Objects included in the TaxaNorm package, loaded with utils::data

Usage

```
data(TaxaNorm_Example_Input, package = "TaxaNorm")

data(TaxaNorm_Example_Output, package = "TaxaNorm")
```

TaxaNorm_Example_Input

Example data #'

TaxaNorm_Example_Output

Example output

Examples

```
data(TaxaNorm_Example_Input, package = "TaxaNorm")
          data(TaxaNorm_Example_Output, package = "TaxaNorm")
```

TaxaNormGenerics *TaxaNorm package generics*

Description

TaxaNorm package generics; see class man pages for associated methods

Usage

```
input_data(x, ...)

input_data(x, ...) <- value

rawdata(x, ...)

rawdata(x, ...) <- value

normdata(x, ...)

normdata(x, ...) <- value

ecdf(x, ...)

ecdf(x, ...) <- value

model_pars(x, ...)

model_pars(x, ...) <- value

converge(x, ...)

converge(x, ...) <- value

llk(x, ...)

llk(x, ...) <- value

final_df(x, ...)

final_df(x, ...) <- value

coefficients(x, ...)

coefficients(x, ...) <- value

mu(x, ...)
```

```
mu(x, ...) <- value

theta(x, ...)

theta(x, ...) <- value

pi(x, ...)

pi(x, ...) <- value
```

Arguments

x	TaxaNorm S4 object
...	Included for extendability; not currently used
value	Replacement value

Value

TaxaNorm generic functions return the specified slot of the TaxaNorm object given to the function

---

TaxaNorm_Model_Parameters-class *TaxaNorm_Model_Parameters*

Description

S4 class to store TaxaNorm Parameters

Usage

```
TaxaNorm_Model_Parameters(coefficients, mu, theta, pi)

## S4 method for signature 'TaxaNorm_Model_Parameters'
coefficients(x)

## S4 replacement method for signature 'TaxaNorm_Model_Parameters'
coefficients(x) <- value

## S4 method for signature 'TaxaNorm_Model_Parameters'
mu(x)

## S4 replacement method for signature 'TaxaNorm_Model_Parameters'
mu(x) <- value

## S4 method for signature 'TaxaNorm_Model_Parameters'
theta(x)
```

```
## S4 replacement method for signature 'TaxaNorm_Model_Parameters'
theta(x) <- value

## S4 method for signature 'TaxaNorm_Model_Parameters'
pi(x)

## S4 replacement method for signature 'TaxaNorm_Model_Parameters'
pi(x) <- value
```

Arguments

coefficients	Passed to coefficients slot
mu	Passed to mu slot
theta	Passed to theta slot
pi	Passed to pi slot
x	TaxaNorm_Model_Parameters object
value	Replacement value

Details

Parameters for TaxaNorm Method

Functions

- coefficients(TaxaNorm_Model_Parameters): Return coefficients slot
- mu(TaxaNorm_Model_Parameters): Return mu slot
- theta(TaxaNorm_Model_Parameters): Return theta slot
- pi(TaxaNorm_Model_Parameters): Return pi slot

Slots

coefficients matrix coefficients

mu matrix mu

theta matrix theta

pi matrix pi

Examples

```
coefficients <-  matrix(c(1,2,3,4,5,6,7,8,9),nrow=3,ncol=3,byrow=TRUE)
mu <- matrix(c(1,2,3,4,5,6,7,8,9),nrow=3,ncol=3,byrow=TRUE)
theta <-  matrix(c(1,2,3,4,5,6,7,8,9),nrow=3,ncol=3,byrow=TRUE)
pi <- matrix(c(1,2,3,4,5,6,7,8,9),nrow=3,ncol=3,byrow=TRUE)
TaxaNorm_Model_Parameters(coefficients = coefficients,mu = mu,theta = theta,pi = pi)
```

TaxaNorm_Model_QC *Function to QC TaxNorm algorithm*

Description

Function to QC TaxNorm algorithm

Usage

```
TaxaNorm_Model_QC(TaxaNormResults)
```

Arguments

TaxaNormResults	Input data; Results from TaxaNorm normalization

Value

a list containing qc taxnorm object

Examples

```
data("TaxaNorm_Example_Output", package = "TaxaNorm")
TaxaNorm_Model_QC(TaxaNormResults = TaxaNorm_Example_Output)
```

TaxaNorm_NMDS *Function for TaxNorm NMDS*

Description

Function for TaxNorm NMDS

Usage

```
TaxaNorm_NMDS(TaxaNormResults, group_column)
```

Arguments

TaxaNormResults	(Required) Input data; should be either a phyloseq object or a count matrix
group_column	column to cluster on

Value

NMDS Plot

Examples

```
data("TaxaNorm_Example_Output", package = "TaxaNorm")
TaxaNorm_NMDS(TaxaNorm_Example_Output, group_column = "body_site")
```

`TaxaNorm_Normalization` *Function to run TaxaNorm algorithm*

Description

Function to run TaxaNorm algorithm

Usage

```
TaxaNorm_Normalization(
  data,
  depth = NULL,
  group = NULL,
  meta.data = NULL,
  filter.cell.num = 10,
  filter.taxa.count = 0,
  random = FALSE,
  ncores = NULL
)
```

Arguments

`data`	(Required) Input data; should be either a phyloseq object or a count matrix
`depth`	sequencing depth if pre-calculated. It should be a vector with the same length and order as the column of the count data
`group`	condition variables if samples are from multiple groups; should be correpsond to the column of the count data. default is NULL, where no grouping is considered
`meta.data`	meta data for Taxa
`filter.cell.num`	taxa with "filter.cell.num" in more than the value provided will be filtered
`filter.taxa.count`	"filter.taxa.count" samples will be removed before testing. default is keep taxa appear in at least 10 samples within each group
`random`	calculate randomized normal quantile residual
`ncores`	whether multiple cores is used for parallel computing; default is max(1, detectCores() - 1)

Value

a TaxaNorm Object containing the normalized count values and accessory information

Examples

```
data("TaxaNorm_Example_Input", package = "TaxaNorm")
Normalized_Data <- TaxaNorm_Normalization(data= TaxaNorm_Example_Input,
                                          depth = NULL,
                                    group = sample_data(TaxaNorm_Example_Input)$body_site,
                                          meta.data = NULL,
                                          filter.cell.num = 10,
                                          filter.taxa.count = 0,
                                          random = FALSE,
                                          ncores = 1)
```

`TaxaNorm_QC_Input` *Function for TaxNorm input data*

Description

Function for TaxNorm input data

Usage

```
TaxaNorm_QC_Input(data)
```

Arguments

`data`	(Required) Input data; should be either a phyloseq object or a count matrix

Value

QC PLots

Examples

```
data("TaxaNorm_Example_Input", package = "TaxaNorm")
qc_data <- TaxaNorm_QC_Input(TaxaNorm_Example_Input)
```

TaxaNorm_Results-class *TaxaNorm Results*

Description

S4 class to store TaxaNorm Results

Usage

```
TaxaNorm_Results(
  input_data,
  rawdata,
  normdata,
  ecdf,
  model_pars,
  converge,
  llk,
  final_df
)

## S4 method for signature 'TaxaNorm_Results'
input_data(x)

## S4 replacement method for signature 'TaxaNorm_Results'
input_data(x) <- value

## S4 method for signature 'TaxaNorm_Results'
rawdata(x)

## S4 replacement method for signature 'TaxaNorm_Results'
rawdata(x) <- value

## S4 method for signature 'TaxaNorm_Results'
normdata(x)

## S4 replacement method for signature 'TaxaNorm_Results'
normdata(x) <- value

## S4 method for signature 'TaxaNorm_Results'
ecdf(x)

## S4 replacement method for signature 'TaxaNorm_Results'
ecdf(x) <- value

## S4 method for signature 'TaxaNorm_Results'
model_pars(x)
```

```
## S4 replacement method for signature 'TaxaNorm_Results'
model_pars(x) <- value

## S4 method for signature 'TaxaNorm_Results'
converge(x)

## S4 replacement method for signature 'TaxaNorm_Results'
converge(x) <- value

## S4 method for signature 'TaxaNorm_Results'
llk(x)

## S4 replacement method for signature 'TaxaNorm_Results'
llk(x) <- value

## S4 method for signature 'TaxaNorm_Results'
final_df(x)

## S4 replacement method for signature 'TaxaNorm_Results'
final_df(x) <- value
```

Arguments

input_data	passed to input_data slot
rawdata	Passed to rawdata slot
normdata	Passed to normdata slot
ecdf	Passed to ecdf slot
model_pars	Passed to model_pars slot
converge	Passed to converge slot
llk	Passed to llk slot
final_df	Passed to final_df slot
x	TaxaNorm_Results object
value	Replacement value

Details

All results from the TaxaNorm method and what was used to get those results

Functions

- input_data(TaxaNorm_Results): Return input_data slot
- rawdata(TaxaNorm_Results): Return rawdata slot
- normdata(TaxaNorm_Results): Return normdata slot
- ecdf(TaxaNorm_Results): Return ecdf slot

- model_pars(TaxaNorm_Results): Return model_pars slot
- converge(TaxaNorm_Results): Return converge slot
- llk(TaxaNorm_Results): Return llk slot
- final_df(TaxaNorm_Results): Return final_df slot

Slots

input_data ANY phyloseq input data

rawdata data.frame Data frame of counts to use

normdata data.frame Normalized Data

ecdf data.frame ecdf

model_pars TaxaNorm_Model_Parameters list of model parameters

converge vector(<logical>) converge

llk ANY llk

final_df ANY final_df

Examples

```
coefficients <-  matrix(c(1,2,3,4,5,6,7,8,9),nrow=3,ncol=3,byrow=TRUE)
mu <- matrix(c(1,2,3,4,5,6,7,8,9),nrow=3,ncol=3,byrow=TRUE)
theta <-  matrix(c(1,2,3,4,5,6,7,8,9),nrow=3,ncol=3,byrow=TRUE)
pi <- matrix(c(1,2,3,4,5,6,7,8,9),nrow=3,ncol=3,byrow=TRUE)
model_pars <- TaxaNorm_Model_Parameters(coefficients = coefficients,mu = mu,theta = theta,pi = pi)
data("TaxaNorm_Example_Input", package = "TaxaNorm")
rawdata <- data.frame(Taxa1 = c(1,2,3),Taxa2 = c(3,4,5),Taxa3 = c(6,7,8))
normdata <- data.frame(Taxa1 = c(-1.4,-1.09,-0.73),
Taxa2 = c( -0.36,0,0.36), Taxa3 = c(0.73,1.09,1.46))
ecdf <- data.frame(0.05,0.23,0.89)
converge <- c(TRUE,TRUE,FALSE)
llk <- c(1,1.5,0.5)
final_df <- data.frame(Taxa1 = c(1,2,3),Taxa2 = c(3,4,5),Taxa3 = c(6,7,8))
TaxaNorm_Results(input_data = TaxaNorm_Example_Input,
                                  rawdata = rawdata,
                                   normdata = normdata,
                                   ecdf = ecdf,
                                   model_pars = model_pars,
                                   converge = converge,
                                   llk = llk,
                                   final_df = final_df)
```

TaxaNorm_Run_Diagnose *Function to run TaxNorm algorithm*

Description

Function to run TaxNorm algorithm

Usage

```
TaxaNorm_Run_Diagnose(Normalized_Results, prev = TRUE, equiv = TRUE, group)
```

Arguments

Normalized_Results	(Required) Input results from from run_norm()
prev	run prev test
equiv	run equiv test
group	group used for taxanorm normalization

Value

a list containing the normalized count values

Examples

```
data("TaxaNorm_Example_Input", package = "TaxaNorm")
data("TaxaNorm_Example_Output", package = "TaxaNorm")
Diagnose_Data <- TaxaNorm_Run_Diagnose(Normalized_Results = TaxaNorm_Example_Output,
                                       prev = TRUE,
                                       equiv = TRUE,
                                   group = sample_data(TaxaNorm_Example_Input)$body_site)
```

Index

coefficients *(TaxaNormGenerics)*, 3
coefficients,TaxaNorm_Model_Parameters-method *(TaxaNorm_Model_Parameters-class)*, 4
coefficients<- *(TaxaNormGenerics)*, 3
coefficients<-,TaxaNorm_Model_Parameters-method *(TaxaNorm_Model_Parameters-class)*, 4
converge *(TaxaNormGenerics)*, 3
converge,TaxaNorm_Results-method *(TaxaNorm_Results-class)*, 9
converge<- *(TaxaNormGenerics)*, 3
converge<-,TaxaNorm_Results-method *(TaxaNorm_Results-class)*, 9

ecdf *(TaxaNormGenerics)*, 3
ecdf,TaxaNorm_Results-method *(TaxaNorm_Results-class)*, 9
ecdf<- *(TaxaNormGenerics)*, 3
ecdf<-,TaxaNorm_Results-method *(TaxaNorm_Results-class)*, 9

final_df *(TaxaNormGenerics)*, 3
final_df,TaxaNorm_Results-method *(TaxaNorm_Results-class)*, 9
final_df<- *(TaxaNormGenerics)*, 3
final_df<-,TaxaNorm_Results-method *(TaxaNorm_Results-class)*, 9

input_data *(TaxaNormGenerics)*, 3
input_data,TaxaNorm_Results-method *(TaxaNorm_Results-class)*, 9
input_data<- *(TaxaNormGenerics)*, 3
input_data<-,TaxaNorm_Results-method *(TaxaNorm_Results-class)*, 9

llk *(TaxaNormGenerics)*, 3
llk,TaxaNorm_Results-method *(TaxaNorm_Results-class)*, 9
llk<- *(TaxaNormGenerics)*, 3
llk<-,TaxaNorm_Results-method *(TaxaNorm_Results-class)*, 9

model_pars *(TaxaNormGenerics)*, 3
model_pars,TaxaNorm_Results-method *(TaxaNorm_Results-class)*, 9
model_pars<- *(TaxaNormGenerics)*, 3
model_pars<-,TaxaNorm_Results-method *(TaxaNorm_Results-class)*, 9
mu *(TaxaNormGenerics)*, 3
mu,TaxaNorm_Model_Parameters-method *(TaxaNorm_Model_Parameters-class)*, 4
mu<- *(TaxaNormGenerics)*, 3
mu<-,TaxaNorm_Model_Parameters-method *(TaxaNorm_Model_Parameters-class)*, 4

normdata *(TaxaNormGenerics)*, 3
normdata,TaxaNorm_Results-method *(TaxaNorm_Results-class)*, 9
normdata<- *(TaxaNormGenerics)*, 3
normdata<-,TaxaNorm_Results-method *(TaxaNorm_Results-class)*, 9

pi *(TaxaNormGenerics)*, 3
pi,TaxaNorm_Model_Parameters-method *(TaxaNorm_Model_Parameters-class)*, 4
pi<- *(TaxaNormGenerics)*, 3
pi<-,TaxaNorm_Model_Parameters-method *(TaxaNorm_Model_Parameters-class)*, 4

rawdata *(TaxaNormGenerics)*, 3
rawdata,TaxaNorm_Results-method *(TaxaNorm_Results-class)*, 9
rawdata<- *(TaxaNormGenerics)*, 3
rawdata<-,TaxaNorm_Results-method *(TaxaNorm_Results-class)*, 9

TaxaNorm-datasets, 2
TaxaNorm_Example_Input (TaxaNorm-datasets), 2
TaxaNorm_Example_Output (TaxaNorm-datasets), 2
TaxaNorm_Model_Parameters, *11*
TaxaNorm_Model_Parameters (TaxaNorm_Model_Parameters-class), 4
TaxaNorm_Model_Parameters-class, 4
TaxaNorm_Model_Parameters-coefficients (TaxaNorm_Model_Parameters-class), 4
TaxaNorm_Model_Parameters-mu (TaxaNorm_Model_Parameters-class), 4
TaxaNorm_Model_Parameters-pi (TaxaNorm_Model_Parameters-class), 4
TaxaNorm_Model_Parameters-theta (TaxaNorm_Model_Parameters-class), 4
TaxaNorm_Model_QC, 6
TaxaNorm_NMDS, 6
TaxaNorm_Normalization, 7
TaxaNorm_QC_Input, 8
TaxaNorm_Results (TaxaNorm_Results-class), 9
TaxaNorm_Results-class, 9
TaxaNorm_Results-converge (TaxaNorm_Results-class), 9
TaxaNorm_Results-ecdf (TaxaNorm_Results-class), 9
TaxaNorm_Results-final_df (TaxaNorm_Results-class), 9
TaxaNorm_Results-input_data (TaxaNorm_Results-class), 9
TaxaNorm_Results-llk (TaxaNorm_Results-class), 9
TaxaNorm_Results-model_pars (TaxaNorm_Results-class), 9
TaxaNorm_Results-normdata (TaxaNorm_Results-class), 9
TaxaNorm_Results-rawdata (TaxaNorm_Results-class), 9
TaxaNorm_Run_Diagnose, 11
TaxaNormGenerics, 3
theta (TaxaNormGenerics), 3
theta,TaxaNorm_Model_Parameters-method (TaxaNorm_Model_Parameters-class), 4
theta<- (TaxaNormGenerics), 3
theta<-,TaxaNorm_Model_Parameters-method (TaxaNorm_Model_Parameters-class), 4

utils::data, *2*