Package: easynem (via r-universe)

October 27, 2026

Title Nematode Community Analysis

Version 1.0.3

Description Provides a built-in Nemaplex database for nematodes, which can be used to search for various nematodes. Also supports various nematode community and functional analyses such as nematode diversity, maturity index, metabolic footprint, and functional guild. The methods are based on <https://shiny.wur.nl/ninja/>, Bongers, T. (1990) <doi:10.1007/BF00324627>, Ferris, H. (2010) <doi:10.1016/j.ejsobi.2010.01.003>, Wan, B. et al. (2022) <doi:10.1016/j.soilbio.2022.108695>, and Van Den Hoogen, J. et al. (2019) <doi:10.1038/s41586-019-1418-6>.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Imports agricolae, broom, dplyr, DT, ggplot2, ggpubr, ggraph, igraph, methods, multcompView, readr, reshape2, rlang, rstatix, shiny, shinydashboard, stats, Ternary, tibble, tidyr, utils, vegan

Depends R (>= 2.10)

LazyData true

Suggests spelling, testthat (>= 3.0.0)

Config/testthat/edition 3

Language en-US

Config/pak/sysreqs cmake libfontconfig1-dev libfreetype6-dev libglpk-dev make libicu-dev libuv1-dev libxml2-dev libx11-dev zlib1g-dev

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

Date/Publication 2025-11-24 14:07:36 UTC

RemoteUrl https://github.com/whkygl/easynem

RemoteRef HEAD

RemoteSha e0c5215144885488f3253779d0672c173bd68fc5

Contents

alpha-class . . . 3
beta-class . . . 4
beta2-class . . . 4
calc_alpha . . . 5
calc_beta . . . 6
calc_beta2 . . . 7
calc_compare . . . 8
calc_compare2 . . . 9
calc_ef . . . 10
calc_ef2 . . . 12
calc_funguild . . . 14
calc_funguild2 . . . 15
calc_lm . . . 16
calc_lm2 . . . 17
calc_mf . . . 18
calc_mf2 . . . 19
calc_nemindex . . . 20
calc_ter . . . 23
calc_ter2 . . . 24
compare-class . . . 25
compare2-class . . . 26
easynem-class . . . 26
easynem_example . . . 27
ef-class . . . 27
ef2-class . . . 28
filter_name . . . 28
filter_num . . . 29
funguild-class . . . 30
funguild2-class . . . 30
geom_encircle . . . 31
HSD . . . 32
HSD2 . . . 33
KruskalTest . . . 34
KruskalTest2 . . . 35
lme-class . . . 37
lme2-class . . . 37
LSD . . . 38
LSD2 . . . 39
mf-class . . . 40
mf2-class . . . 40
nem_calc . . . 40
nem_database . . . 41
nem_plot . . . 41
nem_plot,beta-method . . . 42
nem_plot,beta2-method . . . 43
nem_plot,compare-method . . . 44

nem_plot,compare2-method . . . 45
nem_plot,ef-method . . . 47
nem_plot,ef2-method . . . 48
nem_plot,funguild-method . . . 49
nem_plot,funguild2-method . . . 50
nem_plot,lme-method . . . 51
nem_plot,lme2-method . . . 52
nem_plot,mf-method . . . 53
nem_plot,mf2-method . . . 54
nem_plot,ter-method . . . 56
nem_plot,ter2-method . . . 57
nem_trans . . . 58
nemindex-class . . . 59
nemmeta . . . 59
nemtab . . . 60
nemtax . . . 60
order_factor . . . 61
read_nem . . . 62
read_nem2 . . . 63
ter-class . . . 64
ter2-class . . . 64
trans_combine . . . 64
trans_formula . . . 65
trans_formula_v . . . 66
trans_name . . . 67
trans_norm . . . 68
trans_rare . . . 69
TTest . . . 70
TTest2 . . . 71
WilcoxTest . . . 72
WilcoxTest2 . . . 73

Index **75**

`alpha-class` *Class for storing alpha diversity calculation results*

Description

The `alpha-class` is an extension of the `easynem-class` to store the results of alpha diversity calculations.

Slots

`result` The calculation results of storage alpha diversity.

See Also

The constructor, `calc_alpha`; Visualization function, `nem_plot`.

beta-class *A class to store beta diversity results (single factor)*

Description

beta-class is used to store the results of beta diversity analysis, including results for drawing and comparing differences between groups.

Details

Users can construct a beta-class through calc_beta, which can then be connected to nem_plot to visualize the results.

Slots

meta A data frame storing basic elements for visualization.

result A character of pairwise comparison results.

temp A character vector of the difference comparison.

See Also

The constructor, calc_beta; Class for storing two-factor beta diversity analysis, beta2-class; Visualization function, nem_plot.

beta2-class *A class to store beta diversity results (two-factor)*

Description

beta2-class is used to store the results of beta diversity analysis, including results for drawing and comparing differences between groups.

Details

Users can construct a beta2-class through calc_beta2, which can then be connected to nem_plot to visualize the results.

Slots

meta A data frame storing basic elements for visualization.

result A character of pairwise comparison results.

temp A character vector of the difference comparison.

See Also

The constructor, calc_beta2; Class for storing single factor beta diversity analysis, beta-class; Visualization function, nem_plot.

calc_alpha *Alpha diversity analysis, generating alpha-class*

Description

The calc_alpha() is used to perform alpha diversity analysis and create alpha-class. This function can be used to calculate various alpha diversity indices such as Chao1, ACE, Shannon, Simpson, etc.

Usage

```
calc_alpha(data, ...)
```

Arguments

data	An easynem-class data.
...	Other parameters for diversity, simpson.unb, fisher.alpha, specnumber, and estimateR.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_alpha <- nem |> calc_alpha()
```

Value

A alpha-class for storing alpha diversity analysis results.

See Also

Other functions in this R package for data calculations: calc_beta2, calc_compare, calc_compare2, calc_beta, calc_nemindex, calc_funguild, calc_funguild2, calc_mf, calc_mf2, calc_ter, calc_ter2, calc_ef, calc_ef2.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_alpha <- nem |> calc_alpha()
show(nem_alpha)
```

calc_beta *Beta diversity analysis, generating beta-class (single factor)*

Description

The calc_beta() is used to perform beta diversity analysis and create beta-class. This function is only applicable to single factor analysis, see calc_beta2 for a two-factor version of the function.

Usage

```
calc_beta(data, type, .group, method, ...)
```

Arguments

data	An easynem-class data.
type	Types of beta diversity analysis (pca, pcoa or nmds).
.group	Treatment factors that need to be compared.
method	Dissimilarity index, partial match to "manhattan", "euclidean", "canberra", "clark", "bray", "kulczynski", "jaccard", "gower", "altGower", "morisita", "horn", "mountford", "raup", "binomial", "chao", "cao", "mahalanobis", "chisq", "chord", "hellinger", "aitchison", or "robust.aitchison". See vegdist.
...	Other parameters for cmdscale, vegdist and adonis2.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_pca <- nem |> calc_beta(pca, Treatments, method = "bray")
```

Value

A beta-class for storing beta diversity analysis results.

See Also

Other functions in this R package for data calculations: calc_beta2, calc_compare, calc_compare2, calc_alpha, calc_nemindex, calc_funguild, calc_funguild2, calc_mf, calc_mf2, calc_ter, calc_ter2, calc_ef, calc_ef2.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_pcoa <- nem |> calc_beta(pcoa, Treatments, method = "bray")
show(nem_pcoa)
nem_nmds <- nem |> calc_beta(nmds, Treatments, method = "bray")
show(nem_nmds)
```

calc_beta2 *Beta diversity analysis, generating beta2-class (two-factor)*

Description

The calc_beta2() is used to perform beta diversity analysis and create beta2-class. This function is only applicable to two-factor factor analysis, see calc_beta for a single factor version of the function.

Usage

```
calc_beta2(data, type, .group1, .group2, method, ...)
```

Arguments

data	An easynem-class data.
type	Types of beta diversity analysis (pca, pcoa or nmds).
.group1	Treatment factors 1 that need to be compared.
.group2	Treatment factors 2 that need to be compared.
method	Dissimilarity index, partial match to "manhattan", "euclidean", "canberra", "clark", "bray", "kulczynski", "jaccard", "gower", "altGower", "morisita", "horn", "mountford", "raup", "binomial", "chao", "cao", "mahalanobis", "chisq", "chord", "hellinger", "aitchison", or "robust.aitchison". See vegdist.
...	Other parameters for cmdscale, vegdist and adonis2.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_pca <- nem |> calc_beta2(pca, con_crop, season, method = "bray")
```

Value

A beta2-class for storing beta diversity analysis results.

See Also

Other functions in this R package for data calculations: calc_beta, calc_compare, calc_compare2, calc_alpha, calc_nemindex, calc_funguild, calc_funguild2, calc_mf, calc_mf2, calc_ter, calc_ter2, calc_ef, calc_ef2.

Examples

```
nem <- read_nem(tab = easynem_example("nemtab1.csv"),
                tax = easynem_example("nemtax1.csv"),
                meta = easynem_example("nemmeta1.csv"))
nem_pcoa <- nem |> calc_beta2(pcoa, con_crop, season, method = "bray")
show(nem_pcoa)
nem_nmds <- nem |> calc_beta2(nmds, con_crop, season, method = "bray")
show(nem_nmds)
```

---

calc_compare *Multiple comparisons between treatments (single factor)*

Description

The calc_compare is used for multiple comparisons between different treatments and create compare-class. This function is only applicable to single factor analysis, see calc_compare2 for a two factor version of the function.

Usage

```
calc_compare(data, .group, y, method, ...)
```

Arguments

data	An easynem-class data.
.group	Grouping variables (supports only two groups).
y	Dependent variable (numeric data).
method	The method of difference comparison. Such as TTest, TTest2, WilcoxTest, WilcoxTest2, KruskalTest, KruskalTest2, LSD, LSD2, HSD, HSD2, etc.
...	Other parameters for t.test.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_compare <- nem |> calc_compare(.group = con_crop, y = pH, method = TTest)
```

Value

An compare-class object.

See Also

Other functions in this R package for data calculations: calc_beta, calc_beta2, calc_compare2, calc_alpha, calc_nemindex, calc_funguild, calc_funguild2, calc_mf, calc_mf2, calc_ter, calc_ter2, calc_ef, calc_ef2.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_ttest <- nem |>
              filter_name(meta, Treatments %in% c("CK", "C8")) |>
              calc_compare(.group = Treatments, y = Mesorhabditis, method = TTest)
nem_ttest
```

`calc_compare2` *Multiple comparisons between treatments (two-factor)*

Description

The `calc_compare2` is used for multiple comparisons between different treatments and create `compare2-class`. This function is only applicable to two-factor analysis, see `calc_compare` for a single factor version of the function.

Usage

```
calc_compare2(data, .group1, .group2, y, method, ...)
```

Arguments

`data`	An `easynem-class` data.
`.group1`	Grouping variables factor 1 (supports only two groups).
`.group2`	Grouping variables factor 1 (supports only two groups).
`y`	Dependent variable (numeric data).
`method`	The method of difference comparison. Such as `TTest`, `TTest2`, `WilcoxTest`, `WilcoxTest2`, `KruskalTest`, `KruskalTest2`, `LSD`, `LSD2`, `HSD`, `HSD2`, etc.
`...`	Other parameters for `t.test`.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol `|>` to connect functions:

```
nem_compare <- nem |> calc_compare2(.group1 = con_crop, .group2 = season, y = pH, method = TTest2)
```

Value

An `compare2-class` object.

See Also

Other functions in this R package for data calculations: `calc_beta`, `calc_beta2`, `calc_compare`, `calc_alpha`, `calc_nemindex`, `calc_funguild`, `calc_funguild2`, `calc_mf`, `calc_mf2`, `calc_ter`, `calc_ter2`, `calc_ef`, `calc_ef2`.

Examples

```
nem <- read_nem(tab = easynem_example("nemtab1.csv"),
                tax = easynem_example("nemtax1.csv"),
                meta = easynem_example("nemmeta1.csv"))
nem_ttest <- nem |>
              filter_name(meta, con_crop %in% c("Y2", "Y11")) |>
             calc_compare2(.group1 = con_crop, .group2 = season, y = pH, method = TTest2)
nem_ttest
```

---

calc_ef *Calculation of energy flow in nematode communities (single factor)*

Description

The `calc_ef()` function is used to calculate the energy flow of a nematode community. For detailed calculation method, see `Wan et al. (2022)`: Step 1, the fresh biomass of each nematode individuals was calculated based on the measurement of body size or using publicly available data. Step 2, nematode metabolism (F) was then calculated according to `Ferris (2010)` and `van den Hoogen et al. (2019)`, where Nt, Wt and mt are the number of individuals, the fresh weight and the cp class of taxon t, respectively. Step 3, a five-node food web topology was constructed and the feeding preferences of omnivores-carnivores on other trophic groups was assumed according to community density. Step 4, the metabolism of each node was summed by all individual metabolism of the respective trophic group. Step 5, we used assimilation efficiencies (ea) of 0.25 for herbivores, 0.60 for bacterivores, 0.38 for fungivores and 0.5 for omnivores-carnivores according to `Barnes et al. (2014)` and `De Ruiter et al. (1993)`. Step 6, energy fluxes between nodes was calculated as follows: `Fi = (F + L)/ea`, where L is the energy loss to higher trophic levels.

Usage

```
calc_ef(data, .group)
```

Arguments

`data`	An `nemindex-class` object.
`.group`	The group variable.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol `|>` to connect functions:

```
nem_ter <- nem |> nem_index() |> calc_ef(con_crop)
```

Value

An ef-class object that stores the desired visualization results.

- OF, Energy flow metabolism of omnivorous predatory nematodes.
- OM, Fresh biomass (ug / 100g dry soil) of omnivorous predatory nematodes.
- BF, Energy flow metabolism of bacteria-feeding nematodes.
- BM, Fresh biomass (ug / 100g dry soil) of omnivorous predatory nematodes.
- HF, Energy flow metabolism of herbivorous nematodes.
- HM, Fresh biomass (ug / 100g dry soil) of herbivorous nematodes.
- FF, Energy flow metabolism of fungus-feeding nematodes.
- FM, Fresh biomass (ug / 100g dry soil) of fungus-feeding nematodes.
- bp, Feeding preference of predatory nematodes over bacteria-feeding nematodes.
- hp, Feeding preference of predatory nematodes over herbivorous nematodes.
- fp, Feeding preferences of predatory nematodes over fungivorous nematodes.
- fbo, Energy flow (ug C / 100g dry soil / day) between bacteria-feeding nematodes and omnivorous predatory nematodes.
- fho, Energy flow (ug C / 100g dry soil / day) between herbivorous nematodes and omnivorous predatory nematodes.
- ffo, Energy flow (ug C / 100g dry soil / day) between fungus-feeding nematodes and omnivorous predatory nematodes.
- frb, Energy flow (ug C / 100g dry soil / day) between basal resources and bacteria-feeding nematodes.
- frh, Energy flow (ug C / 100g dry soil / day) between basal resources and herbivorous nematodes.
- frf, Energy flow (ug C / 100g dry soil / day) between basal resources and fungivorous nematodes.
- U, Uniformity (U) of soil nematode energetic structure (unitless, mean ± standard error) was calculated as the ratio of the mean of summed energy flux through each energy channel to the standard deviation of these mean values.

References

- Wan, Bingbing, et al. "Organic amendments increase the flow uniformity of energy across nematode food webs." Soil Biology and Biochemistry 170 (2022): 108695.
- Ferris, H., 2010. Form and function: metabolic footprints of nematodes in the soil food web. European Journal of Soil Biology 46, 97–104.
- Van Den Hoogen, Johan, et al. "Soil nematode abundance and functional group composition at a global scale." Nature 572.7768 (2019): 194-198.
- Barnes, A.D., Jochum, M., Mumme, S., Haneda, N.F., Farajallah, A., Widarto, T.H., Brose, U., 2014. Consequences of tropical land use for multitrophic biodiversity and ecosystem functioning. Nature Communications 5, 1–7.
- De Ruiter, P.C., Van Veen, J.A., Moore, J.C., Brussaard, L., Hunt, H.W., 1993. Calculation of nitrogen mineralization in soil food webs. Plant and Soil 157, 263–273.

See Also

Other functions in this R package for data calculations: calc_beta2, calc_compare, calc_compare2, calc_beta, calc_alpha, calc_nemindex, calc_funguild, calc_funguild2, calc_mf2, calc_mf, calc_ter2, calc_ter, calc_ef2.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_index <- nem |> calc_nemindex() |> calc_ef(Treatments)
nem_index
```

---

calc_ef2 *Calculation of energy flow in nematode communities (two-factor)*

Description

The calc_ef2() function is used to calculate the energy flow of a nematode community. For detailed calculation method, see Wan et al. (2022): Step 1, the fresh biomass of each nematode individuals was calculated based on the measurement of body size or using publicly available data. Step 2, nematode metabolism (F) was then calculated according to Ferris (2010) and van den Hoogen et al. (2019), where Nt, Wt and mt are the number of individuals, the fresh weight and the cp class of taxon t, respectively. Step 3, a five-node food web topology was constructed and the feeding preferences of omnivores-carnivores on other trophic groups was assumed according to community density. Step 4, the metabolism of each node was summed by all individual metabolism of the respective trophic group. Step 5, we used assimilation efficiencies (ea) of 0.25 for herbivores, 0.60 for bacterivores, 0.38 for fungivores and 0.5 for omnivores-carnivores according to Barnes et al. (2014) and De Ruiter et al. (1993). Step 6, energy fluxes between nodes was calculated as follows: Fi = (F + L)/ea, where L is the energy loss to higher trophic levels.

Usage

```
calc_ef2(data, .group1, .group2)
```

Arguments

data	An nemindex-class object.
.group1	The group variable factor 1.
.group2	The group variable factor 2.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_ter <- nem |> nem_index() |> calc_ef2(con_crop, season)
```

Value

An ef2-class object that stores the desired visualization results.

- OF, Energy flow metabolism of omnivorous predatory nematodes.
- OM, Fresh biomass (ug / 100g dry soil) of omnivorous predatory nematodes.
- BF, Energy flow metabolism of bacteria-feeding nematodes.
- BM, Fresh biomass (ug / 100g dry soil) of omnivorous predatory nematodes.
- HF, Energy flow metabolism of herbivorous nematodes.
- HM, Fresh biomass (ug / 100g dry soil) of herbivorous nematodes.
- FF, Energy flow metabolism of fungus-feeding nematodes.
- FM, Fresh biomass (ug / 100g dry soil) of fungus-feeding nematodes.
- bp, Feeding preference of predatory nematodes over bacteria-feeding nematodes.
- hp, Feeding preference of predatory nematodes over herbivorous nematodes.
- fp, Feeding preferences of predatory nematodes over fungivorous nematodes.
- fbo, Energy flow (ug C / 100g dry soil / day) between bacteria-feeding nematodes and omnivorous predatory nematodes.
- fho, Energy flow (ug C / 100g dry soil / day) between herbivorous nematodes and omnivorous predatory nematodes.
- ffo, Energy flow (ug C / 100g dry soil / day) between fungus-feeding nematodes and omnivorous predatory nematodes.
- frb, Energy flow (ug C / 100g dry soil / day) between basal resources and bacteria-feeding nematodes.
- frh, Energy flow (ug C / 100g dry soil / day) between basal resources and herbivorous nematodes.
- frf, Energy flow (ug C / 100g dry soil / day) between basal resources and fungivorous nematodes.
- U, Uniformity (U) of soil nematode energetic structure (unitless, mean ± standard error) was calculated as the ratio of the mean of summed energy flux through each energy channel to the standard deviation of these mean values.

References

- Wan, Bingbing, et al. "Organic amendments increase the flow uniformity of energy across nematode food webs." Soil Biology and Biochemistry 170 (2022): 108695.
- Ferris, H., 2010. Form and function: metabolic footprints of nematodes in the soil food web. European Journal of Soil Biology 46, 97–104.
- Van Den Hoogen, Johan, et al. "Soil nematode abundance and functional group composition at a global scale." Nature 572.7768 (2019): 194-198.
- Barnes, A.D., Jochum, M., Mumme, S., Haneda, N.F., Farajallah, A., Widarto, T.H., Brose, U., 2014. Consequences of tropical land use for multitrophic biodiversity and ecosystem functioning. Nature Communications 5, 1–7.
- De Ruiter, P.C., Van Veen, J.A., Moore, J.C., Brussaard, L., Hunt, H.W., 1993. Calculation of nitrogen mineralization in soil food webs. Plant and Soil 157, 263–273.

See Also

Other functions in this R package for data calculations: calc_beta2, calc_compare, calc_compare2, calc_beta, calc_alpha, calc_nemindex, calc_funguild, calc_funguild2, calc_mf2, calc_mf, calc_ter2, calc_ter, calc_ef.

Examples

```
nem <- read_nem(tab = easynem_example("nemtab1.csv"),
                tax = easynem_example("nemtax1.csv"),
                meta = easynem_example("nemmeta1.csv"))
nem_index <- nem |> calc_nemindex() |> calc_ef2(con_crop, season)
nem_index
```

---

calc_funguild *Nematode food web analysis (single factor)*

Description

The calc_funguild() is used for nematode food web analysis and generate funguild-class.

Usage

```
calc_funguild(data, .group)
```

Arguments

data	A nemindex-class object.
.group	The group variable.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_fun <- nem |> calc_funguild(con_crop)
```

Value

A funguild-class object that stores the desired visualization results.

References

- https://shiny.wur.nl/ninja/
- Ferris, Howard, Tom Bongers, and Ron GM de Goede. "A framework for soil food web diagnostics: extension of the nematode faunal analysis concept." Applied soil ecology 18.1 (2001): 13-29.

See Also

Other functions in this R package for data calculations: calc_beta2, calc_compare, calc_compare2, calc_beta, calc_alpha, calc_nemindex, calc_funguild2, calc_mf, calc_mf2, calc_ter, calc_ter2, calc_ef, calc_ef2.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_index <- nem |> calc_nemindex() |> calc_funguild(Treatments)
nem_index
```

---

calc_funguild2 *Nematode food web analysis (two-factor)*

Description

The calc_funguild2() is used for nematode food web analysis and generate funguild2-class.

Usage

```
calc_funguild2(data, .group1, .group2)
```

Arguments

data	A nemindex-class object.
.group1	The group variable factor 1.
.group2	The group variable factor 2.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_fun <- nem |> calc_funguild2(con_crop, season)
```

Value

A funguild2-class object that stores the desired visualization results.

References

- https://shiny.wur.nl/ninja/
- Ferris, Howard, Tom Bongers, and Ron GM de Goede. "A framework for soil food web diagnostics: extension of the nematode faunal analysis concept." Applied soil ecology 18.1 (2001): 13-29.

See Also

Other functions in this R package for data calculations: calc_beta2, calc_compare, calc_compare2, calc_beta, calc_alpha, calc_nemindex, calc_funguild, calc_mf, calc_mf2, calc_ter, calc_ter2, calc_ef, calc_ef2.

Examples

```
nem <- read_nem(tab = easynem_example("nemtab1.csv"),
                tax = easynem_example("nemtax1.csv"),
                meta = easynem_example("nemmeta1.csv"))
nem_index <- nem |> calc_nemindex() |> calc_funguild2(con_crop, season)
nem_index
```

---

calc_lm *Linear regression analysis of easynem-class (single factor)*

Description

The calc_lm() function is used for linear regression analysis of easynem-class. Note: Both the horizontal and vertical coordinates of this function must be continuous variables.

Usage

```
calc_lm(data, group, x, y, ...)
```

Arguments

data	An easynem-class object.
group	The group variable.
x	X-axis.
y	Y-axis.
...	Other parameters of the lm function.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_lm <- nem |> calc_lm(con_crop, x = SOC, y = pH)
```

Value

Returns an lme-class object storing the results of a linear regression analysis.

See Also

Other functions in this R package for data calculations: calc_beta2, calc_compare, calc_compare2, calc_beta, calc_alpha, calc_nemindex, calc_funguild, calc_funguild2, calc_mf2, calc_mf, calc_ter2, calc_ef, calc_ef2, calc_lm2

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_index <- nem |>
             calc_alpha() |>
             calc_nemindex() |>
             calc_lm(group = Treatments,
                     x = Chao1,
                     y = TotalBiomass)
nem_index
```

`calc_lm2` *Linear regression analysis of easynem-class (two-factor)*

Description

The `calc_lm2()` function is used for linear regression analysis of `easynem-class`. Note: Both the horizontal and vertical coordinates of this function must be continuous variables.

Usage

```
calc_lm2(data, group1, group2, x, y, ...)
```

Arguments

`data`	An `easynem-class` object.
`group1`	The group variable factor 1.
`group2`	The group variable factor 2.
`x`	X-axis.
`y`	Y-axis.
`...`	Other parameters of the `lm` function.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol `|>` to connect functions:

```
nem_lm <- nem |> calc_lm2(con_crop, season, x = SOC, y = pH)
```

Value

Returns an `lme2-class` object storing the results of a linear regression analysis.

See Also

Other functions in this R package for data calculations: `calc_beta2`, `calc_compare`, `calc_compare2`, `calc_beta`, `calc_alpha`, `calc_nemindex`, `calc_funguild`, `calc_funguild2`, `calc_mf2`, `calc_mf`, `calc_ter2`, `calc_ef`, `calc_ef2`, `calc_lm`.

Examples

```
nem <- read_nem(tab = easynem_example("nemtab1.csv"),
                tax = easynem_example("nemtax1.csv"),
                meta = easynem_example("nemmeta1.csv"))
nem_lm <- nem |> calc_lm2(con_crop, season, x = pH, y = Fe)
```

`calc_mf` *Calculating the metabolic footprint of nematodes (single factor)*

Description

Metabolic footprints quantify the amplitude of Carbon utilisation by different food web components. The point in the middle of a rhombus represents the intersection of EI and SI and length of vertical and horizontal axes of the rhombus corresponds to the footprints of enrichment and structure components respectively.

Usage

```
calc_mf(data, .group)
```

Arguments

`data`	A `nemindex-class` object.
`.group`	The group variable.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol `|>` to connect functions:

```
nem_fun <- nem |> calc_nemindex() |> calc_mf(con_crop)
```

Value

A `mf-class` object that stores the desired visualization results.

References

- https://shiny.wur.nl/ninja/
- Ferris, Howard. "Form and function: metabolic footprints of nematodes in the soil food web." European Journal of Soil Biology 46.2 (2010): 97-104.

See Also

Other functions in this R package for data calculations: `calc_beta2`, `calc_compare`, `calc_compare2`, `calc_beta`, `calc_alpha`, `calc_nemindex`, `calc_funguild`, `calc_funguild2`, `calc_mf2`, `calc_ter`, `calc_ter2`, `calc_ef`, `calc_ef2`.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_index <- nem |> calc_nemindex() |> calc_mf(Treatments)
nem_index
```

`calc_mf2` *Calculating the metabolic footprint of nematodes (two-factor)*

Description

Metabolic footprints quantify the amplitude of Carbon utilisation by different food web components. The point in the middle of a rhombus represents the intersection of EI and SI and length of vertical and horizontal axes of the rhombus corresponds to the footprints of enrichment and structure components respectively.

Usage

```
calc_mf2(data, .group1, .group2)
```

Arguments

`data`	A `nemindex-class` object.
`.group1`	The group variable factor 1.
`.group2`	The group variable factor 2.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol `|>` to connect functions:

```
nem_fun <- nem |> calc_nemindex() |> calc_mf2(con_crop, season)
```

Value

A `mf2-class` object that stores the desired visualization results.

References

- https://shiny.wur.nl/ninja/
- Ferris, Howard. "Form and function: metabolic footprints of nematodes in the soil food web." European Journal of Soil Biology 46.2 (2010): 97-104.

See Also

Other functions in this R package for data calculations: `calc_beta2`, `calc_compare`, `calc_compare2`, `calc_beta`, `calc_alpha`, `calc_nemindex`, `calc_funguild`, `calc_funguild2`, `calc_mf`, `calc_ter`, `calc_ter2`, `calc_ef`, `calc_ef2`.

Examples

```
nem <- read_nem(tab = easynem_example("nemtab1.csv"),
                tax = easynem_example("nemtax1.csv"),
                meta = easynem_example("nemmeta1.csv"))
nem_index <- nem |> calc_nemindex() |> calc_mf2(con_crop, season)
nem_index
```

`calc_nemindex` *Calculate multiple nematode ecological indices and generate nemindex class*

Description

The `calc_nemindex()` is used to Calculate multiple nematode ecological indices and generate `nemindex-class`. The ecological indexes that can be calculated by this function include `MI`, `sigMI`, `sigMI25`, `MI25`, `PPI`, `WI`, `NCR`, `CI`, `BI`, `SI`, `EI`, etc.

Usage

```
calc_nemindex(data)
```

Arguments

`data`	An `easynem-class` data.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol `|>` to connect functions:

```
nem_index <- nem |> calc_nemindex()
```

Value

A `nemindex-class` for storing nematode ecological indices analysis results.

- `MI`, Maturity Index. Indicates environmental disturbance resulting from perturbations (range, 1-5). Low values (<2) indicate an early (primary or secondary) successional stage or a temporary level of increased nutrient availability. Values close to 2 indicate a high level of disturbance with low soil food web structure, while intermediate values (2.5–3) indicate some soil food web maturity. High values (>3) indicate a well-structured and complex soil food web likely with connectivity and energy flow between trophic levels.
- `sigMI`, Sigma Maturity Index (SigmaMI). Indicates environmental disturbance resulting from perturbations in non-agricultural soils (range, 1-5). Low values (<2) indicate a high level of nutrient availability and minimal plant-parasitic pressure, while values close to 2 indicate a high level of disturbance with low soil food web structure. Intermediate values (2.5–3) indicate some soil food web maturity. High values (>3), in turn, indicate a well-structured and complex soil food web likely with connectivity and energy flow between trophic levels, which might include larger plant-parasitic nematodes. This index is less sensitive to enrichment in agricultural soils.

- `sigMI25`, Sigma Maturity Index 2-5 (SigmaMI25). computes the MI for all nematodes in the c-p2-5 range (Neher & Campbell, 1996). The index recognizes that the higher c-p value plant-feeding species also provide information of environmental stress but bears some of the burden of the SigmaMI in situations of nutrient enrichment.
- `MI25`, Maturity Index 2–5. Indicates Environmental disturbance resulting from perturbations unrelated to nutrient enrichment in agricultural fields (range, 2-5). Low values (close to 2) indicate substantial disturbance resulting from perturbations unrelated to nutrient enrichment. High values (>3) indicate greater maturity with minimal or no effect resulting from perturbations.
- `PPI`, Plant-Parasitic Index. Indicates Assemblage composition of plant-parasitic nematodes (range, 2-5). Low values (close to 2) indicate plant-parasitic nematode assemblages dominated by small and medium-sized ectoparasites that feed on single plant cells. Higher values indicate assemblages dominated by medium and large (semi-) endoparasitic (e.g., Meloidogyne and Heterodera spp.) or ectoparasitic virus transmitting nematodes (e.g., Xiphinema and Longidorus spp.).
- `PPI_MI`, PPI/MI. The PPI/MI ratio is lower under nutrient poor conditions than under nutrient rich conditions. It is a sensitive indicator of enrichment in agroecosystems (Bongers & Korthals, 1995; Bongers et al., 1997).
- `WI`, Wasilewska Index. Wasilewska Index is calculated by dividing the sum of bacteria-feeding nematodes and fungi-feeding nematodes by the number of herbivorous nematodes. This index is used to indicate the impact of nematode communities on crop production. The smaller the index, the greater the negative impact of nematode communities on crop production.
- `NCR`, Nematode Channel Ratio. The Nematode Channel Ratio (NCR) is a parameter used in soil ecology to assess the balance between bacterial and fungal energy channels in the soil food web. This ratio is calculated by comparing the abundance of bacterial-feeding nematodes to fungal-feeding nematodes. High NCR: Indicates a bacterial-dominated energy channel. This is often found in soils with frequent disturbance or high inputs of easily decomposable organic matter. Low NCR: Indicates a fungal-dominated energy channel. This is commonly found in more stable, less disturbed soils, such as forests or natural grasslands, where organic matter decomposition is slower and more complex.
- `CI`, Channel Index. Indicates predominant decomposition pathway of organic matter (range, 0-100). Lower values (<50) indicate increasing decomposition dominance by bacteria, while higher values (>50) indicate increasing decomposition dominance by fungi. Bacterial dominance indicates the presence of rapidly decomposed organic matter, while fungal dominated decomposition indicates the slow breakdown of more complex organic matter. The focus on opportunistic bacterial and fungal feeders makes this a highly responsive index, which can be used to detect alternating decomposition pathways over time.
- `EI`, Enrichment Index. Indicates food availability and nutrient enrichment (range, 0-100). Low (0–30), intermediate (30–60), and high (60–100) values indicate equivalent levels of food availability (e.g., labile organic carbon) and nutrient enrichment.
- `SI`, Structure Index. Indicates Soil food web structure and complexity, as well as disturbance due to environmental (e.g., salinity and drought) or anthropogenic (e.g. tillage, mining, and chemical pollution) causalities (range, 0-100). Low (0–30), intermediate (30–60), and high (60–100) values indicate equivalent levels of soil food web complexity. Lower values are indicative of perturbed soil food webs, while higher values indicate a structured soil food web.

- `BI`, Basal Index. Indicates food web structure and complexity (range, 0-100). Low (0–30), intermediate (30–60), and high (60–100) values indicate equivalent levels of soil perturbation. Therefore, higher values (>50) are indicative of a depleted and damaged soil food web.
- `TotalBiomass`, Total biomass of nematode community.
- `MetabolicFootprint`, Metabolic Footprints. Indicates magnitude of ecosystem functions and services fulfilled by nematode community (range, 0-infinite). Higher metabolic footprint values are indicative of greater carbon channelling and therefore an increased contribution to the fulfilment of soil ecosystem functions and services. This can be considered per trophic group (e.g. bacterivore footprint), or per component of the nematode community that indicate enrichment (enrichment footprint) and structure (structure footprint).
- `EnrichmentFootprint`, Enrichment Footprint.
- `StructureFootprint`, Structure Footprint.
- `HerbivoreFootprint`, Herbivore Footprint.
- `FungivoreFootprint`, Fungivore Footprint
- `BacterivoreFootprint`, Bacterivore Footprint.
- `PrOmFootprint`, Metabolic footprint of an omnivorous predatory nematode.
- `Numbers`, Number of nematodes.
- `CAssimilated`, Carbon assimilated by nematodes.
- `CRespired`, Carbon consumed by nematode respiration.

References

- https://shiny.wur.nl/ninja/
- http://nemaplex.ucdavis.edu/Ecology/Indices_of_ecosystem_condition.html
- Du Preez G, Daneel M, De Goede R, et al. Nematode-based indices in soil ecology: Application, utility, and future directions. Soil Biology and Biochemistry, 2022, 169: 108640.
- Bongers T. The maturity index: an ecological measure of environmental disturbance based on nematode species composition. Oecologia, 1990, 83: 14-19.
- Bongers T, Goede R G N, Korthals G W, et al. Proposed changes of cp classification for nematodes. 1995.
- Ferris, H. O. W. A. R. D., and Tom Bongers. "Indices developed specifically for analysis of nematode assemblages." Nematodes as environmental indicators. Wallingford UK: CABI, 2009. 124-145.
- Goede, RGM de, T. Bongers, and C. H. Ettema. "Graphical presentation and interpretation of nematode community structure: cp triangles." (1993): 743-750.
- Ferris, Howard, Tom Bongers, and Ron GM de Goede. "A framework for soil food web diagnostics: extension of the nematode faunal analysis concept." Applied soil ecology 18.1 (2001): 13-29.
- Ferris, Howard. "Form and function: metabolic footprints of nematodes in the soil food web." European Journal of Soil Biology 46.2 (2010): 97-104.

See Also

Other functions in this R package for data calculations: calc_beta2, calc_compare, calc_compare2, calc_beta, calc_alpha, calc_funguild, calc_funguild2, calc_mf, calc_mf2, calc_ter, calc_ter2, calc_ef, calc_ef2.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_index <- nem |> calc_nemindex()
show(nem_index)
```

---

calc_ter *Trivariate analysis of nematode feeding or cp value (single factor)*

Description

The calc_ter() function is used to perform ternary analysis on nematode feeding (Relative biomass of bacteria-feeding nematodes, fungi-feeding nematodes, and herbivorous nematodes) or cp values (Relative abundance of cp1 nematodes, cp2 nematodes, and cp3-5 nematodes).

Usage

```
calc_ter(data, .group)
```

Arguments

data	An easynem-class object.
.group	The group variable.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_ter <- nem |> calc_ter(con_crop)
```

Value

A ter-class object that stores the desired visualization results.

References

- https://shiny.wur.nl/ninja/
- Goede, RGM de, T. Bongers, and C. H. Ettema. "Graphical presentation and interpretation of nematode community structure: cp triangles." (1993): 743-750.

See Also

Other functions in this R package for data calculations: calc_beta2, calc_compare, calc_compare2, calc_beta, calc_alpha, calc_nemindex, calc_funguild, calc_funguild2, calc_mf2, calc_mf, calc_ter2, calc_ef, calc_ef2.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_index <- nem |> calc_ter(Treatments)
nem_index
```

---

calc_ter2 *Trivariate analysis of nematode feeding or cp value (two-factor)*

Description

The calc_ter2() function is used to perform ternary analysis on nematode feeding (Relative biomass of bacteria-feeding nematodes, fungi-feeding nematodes, and herbivorous nematodes) or cp values (Relative abundance of cp1 nematodes, cp2 nematodes, and cp3-5 nematodes).

Usage

```
calc_ter2(data, .group1, .group2)
```

Arguments

data	An easynem-class object.
.group1	The group variable factor 1.
.group2	The group variable factor 2.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_ter <- nem |> calc_ter2(con_crop, season)
```

Value

A ter2-class object that stores the desired visualization results.

References

- https://shiny.wur.nl/ninja/
- Goede, RGM de, T. Bongers, and C. H. Ettema. "Graphical presentation and interpretation of nematode community structure: cp triangles." (1993): 743-750.

See Also

Other functions in this R package for data calculations: calc_beta2, calc_compare, calc_compare2, calc_beta, calc_alpha, calc_nemindex, calc_funguild, calc_funguild2, calc_mf2, calc_mf, calc_ter, calc_ef, calc_ef2.

Examples

```
nem <- read_nem(tab = easynem_example("nemtab1.csv"),
                tax = easynem_example("nemtax1.csv"),
                meta = easynem_example("nemmeta1.csv"))
nem_index <- nem |> calc_ter2(con_crop, season)
nem_index
```

---

compare-class *A S4 class to store multiple comparisons results (single factor).*

Description

compare-class is used to store the results of multiple comparisons results, including results for drawing and comparing differences between groups.

Details

Users can construct a compare-class through calc_compare, which can then be connected to nem_plot to visualize the results.

Slots

meta A data frame storing basic elements for visualization.

result A data frame of multiple comparisons results.

temp A character vector of the difference comparison.

See Also

The constructor, calc_compare; Class for storing two-factor multiple comparisons analysis, compare2-class; Visualization function, nem_plot.

compare2-class *A S4 class to store multiple comparisons results (two-factor).*

Description

compare2-class is used to store the results of multiple comparisons results, including results for drawing and comparing differences between groups.

Details

Users can construct a compare2-class through calc_compare2, which can then be connected to nem_plot to visualize the results.

Slots

meta A data frame storing basic elements for visualization.

result A data frame of multiple comparisons results.

temp A character vector of the difference comparison.

See Also

The constructor, calc_compare2; Class for storing single factor multiple comparisons analysis, compare-class; Visualization function, nem_plot.

easynem-class *The main experiment-level class for easynem data*

Description

Integrate the nematode abundance table, nematode classification table, and experimental design table into an easynem-class, which makes it easier to filter and manage nematode data, and easier to link to the nematode database and conduct subsequent analysis.

Details

Users can read data via read_nem or read_nem2. When there are missing slots in easynem, the system will issue a warning, but this will not affect subsequent analysis.

Slots

tab A single object of nematode abundance table.

tax A single object of nematode classification table.

meta A single object of experimental design table.

See Also

The constructor, read_nem for reading csv files and read_nem2 for reading tibble type data.

easynem_example *Path to example files*

Description

This function returns the path to the example files.

Usage

```
easynem_example(path = NULL)
```

Arguments

path	The path to the example files.

Value

The path to the example files.

ef-class *A S4 class to store energy flow results (single factor)*

Description

The ef-class is an extension of the easynem-class to store the results of nematode energy flow analysis.

Slots

result A data frame for storing the results of energy flow analysis.

See Also

The constructor, calc_ef; Visualization function, nem_plot.

ef2-class *A S4 class to store energy flow results (two-factor)*

Description

The ef2-class is an extension of the easynem-class to store the results of nematode energy flow analysis.

Slots

result A data frame for storing the results of energy flow analysis.

See Also

The constructor, calc_ef2; Visualization function, nem_plot.

filter_name *Filter easynem by column name and keep rows that match a condition*

Description

The filter_name() is the extension of the filter function for easynem type data, used to subset an easynem object, retaining all rows that satisfy your conditions. This function selects one of tab, tax or meta in easynem for filtering. When any of the three components changes, the related components will also change accordingly. To be retained, the row must produce a value of TRUE for all conditions.

Usage

```
filter_name(data, target, ...)
```

Arguments

data	An easynem-class data.
target	tab, tax or meta, where tab represents the species abundance table, tax represents the species classification table, and meta represents the experimental design table.
...	Other parameters of the filter function of the dplyr package.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_filter <- nem |> filter_name(target = meta, season == "Summer")
```

Value

An easynem-class data. The rows of each component are a subset of the input, but appear in the same order and the columns of each component are not modified.

See Also

Other functions in this package for filtering and transforming data sets: filter_num, trans_formula, trans_formula_v, trans_name, trans_norm, trans_rare, trans_combine

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_filter <- nem |> filter_name(target = meta, Treatments == "C4")
show(nem_filter)
```

---

filter_num *Filter easynem's tab by discovery rate or abundance*

Description

The filter_num() is used to filter the rows of the easynem tab by abundance or discovery rate. If num>1, filter by abundance, num is the lowest abundance of the tab; if num<1, filter by discovery rate, num is the lowest discovery rate of the tab.

Usage

```
filter_num(data, num)
```

Arguments

data	An easynem-class data.
num	Filter threshold value. If num>1, filter by abundance, num is the lowest abundance of the tab; if num<1, filter by discovery rate, num is the lowest discovery rate of the tab.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_filter <- nem |> filter_num(target = meta, num = 0.85)

nem_filter <- nem |> filter_num(target = meta, num = 500)
```

Value

An easynem-class data.The results of tab, tax, and meta are the retention values after filtering the tab by abundance or discovery rate.

See Also

Other functions in this package for filtering and transforming data sets: filter_name, trans_formula, trans_formula_v, trans_name, trans_norm, trans_rare, trans_combine

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_filter <- nem |> filter_num(num = 0.9)
show(nem_filter)
nem_filter <- nem |> filter_num(num = 1000)
show(nem_filter)
```

---

funguild-class *Class for storing computational results of nematode functional guild analysis (single factor)*

Description

The funguild-class is used to store the results of nematode functional guild analysis.

Slots

result A data frame of storing computational results of nematode functional guild analysis.

See Also

The constructor, calc_funguild; Visualization function, nem_plot.

---

funguild2-class *Class for storing computational results of nematode functional guild analysis (two-factor)*

Description

The funguild2-class is used to store the results of nematode functional guild analysis.

Slots

result A data frame of storing computational results of nematode functional guild analysis.

See Also

The constructor, calc_funguild2; Visualization function, nem_plot.

`geom_encircle` *Automatically enclose points in a polygon*

Description

Automatically enclose points in a polygon

Usage

```
geom_encircle(
  mapping = NULL,
  data = NULL,
  stat = "identity",
  position = "identity",
  na.rm = FALSE,
  show.legend = NA,
  inherit.aes = TRUE,
  ...
)
```

Arguments

`mapping`	mapping
`data`	data
`stat`	stat
`position`	position
`na.rm`	na.rm
`show.legend`	show.legend
`inherit.aes`	inherit.aes
`...`	dots

Details

A sample of the output from `geom_encircle()`

Value

adds a circle around the specified points

Author(s)

Ben Bolker

Examples

```
d <- data.frame(x=c(1,1,2),y=c(1,2,2)*100)

gg <- ggplot2::ggplot(d,ggplot2::aes(x,y))
gg <- gg + ggplot2::scale_x_continuous(expand=c(0.5,1))
gg <- gg + ggplot2::scale_y_continuous(expand=c(0.5,1))

gg + geom_encircle(s_shape=1, expand=0) + ggplot2::geom_point()

gg + geom_encircle(s_shape=1, expand=0.1, colour="red") + ggplot2::geom_point()

gg + geom_encircle(s_shape=0.5, expand=0.1, colour="purple") + ggplot2::geom_point()

gg + geom_encircle(data=subset(d, x==1), colour="blue", spread=0.02) +
  ggplot2::geom_point()

gg +geom_encircle(data=subset(d, x==2), colour="cyan", spread=0.04) +
  ggplot2::geom_point()

gg <- ggplot2::ggplot(ggplot2::mpg, ggplot2::aes(displ, hwy))
gg + geom_encircle(data=subset(ggplot2::mpg, hwy>40)) + ggplot2::geom_point()
gg + geom_encircle(ggplot2::aes(group=manufacturer)) + ggplot2::geom_point()
gg + geom_encircle(ggplot2::aes(group=manufacturer,fill=manufacturer),alpha=0.4)+
       ggplot2::geom_point()
gg + geom_encircle(ggplot2::aes(group=manufacturer,colour=manufacturer))+
       ggplot2::geom_point()

ss <- subset(ggplot2::mpg,hwy>31 & displ<2)

gg + geom_encircle(data=ss, colour="blue", s_shape=0.9, expand=0.07) +
  ggplot2::geom_point() + ggplot2::geom_point(data=ss, colour="blue")
```

HSD *Compute Tukey Honest Significant Differences (single factor)*

Description

The HSD() is used to Compute Tukey Honest Significant Differences for grouped data and create compare-class. This function is only applicable to single factor analysis, see HSD2 for a two factor version of the function.

Usage

```
HSD(data, .group, y, ...)
```

Arguments

data	An easynem-class data.
.group	Grouping variables.

y	Dependent variable (numeric data).
...	Other parameters for TukeyHSD.

Details

To facilitate code interpretation, It is recommended to use this function in conjunction with the calc_compare function:

```
nem_compare <- nem |> calc_compare(.group = con_crop, y = pH, method = HSD)
```

Value

An compare-class object.

See Also

Other functions related to differential analysis methods: TTest2, TTest, WilcoxTest2, WilcoxTest, KruskalTest2, KruskalTest, LSD2, LSD, HSD2.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_test <- nem |>
            calc_compare(.group = Treatments,
              y = Mesorhabditis,
              method = HSD)
nem_test
```

HSD2 *Compute Tukey Honest Significant Differences (two-factor)*

Description

The HSD2() is used to Compute Tukey Honest Significant Differences for grouped data and create compare2-class. This function is only applicable to two-factor analysis, see HSD for a single factor version of the function.

Usage

```
HSD2(data, .group1, .group2, y, ...)
```

Arguments

data	An easynem-class data.
.group1	Grouping variables factor 1.
.group2	Grouping variables factor 2.
y	Dependent variable (numeric data).
...	Other parameters for TukeyHSD.

Details

To facilitate code interpretation, It is recommended to use this function in conjunction with the calc_compare2 function:

```
nem_compare <- nem |> calc_compare2(.group1 = con_crop, .group2 = season, y = pH, method = HSD2)
```

Value

An compare2-class object.

See Also

Other functions related to differential analysis methods: TTest2, TTest, WilcoxTest2, WilcoxTest, KruskalTest2, KruskalTest, LSD2, LSD, HSD.

Examples

```
nem <- read_nem(tab = easynem_example("nemtab1.csv"),
                tax = easynem_example("nemtax1.csv"),
                meta = easynem_example("nemmeta1.csv"))
nem_test <- nem |>
              calc_compare2(.group1 = con_crop, .group2 = season, y = pH, method = HSD2)
nem_test
```

KruskalTest *Perform Kruskal-Wallis test on easynem meta-table by treatment (single factor)*

Description

The KruskalTest() is used to perform Kruskal-Wallis test for grouped data and create compare-class. This function is only applicable to single factor analysis, see KruskalTest2 for a two factor version of the function.

Usage

```
KruskalTest(data, .group, y, exact=FALSE, sort=TRUE, .method=c("holm",
"hochberg", "hommel", "bonferroni", "BH", "BY", "fdr", "none"), ...)
```

Arguments

data	An easynem-class data.
.group	Grouping variables.
y	Dependent variable (numeric data).
exact	logical. If TRUE, calculate exact Wilcoxon tests. Default exact = FALSE.
sort	logical. If TRUE, sort groups by median dependent variable values. Default sort = TRUE.
.method	method for correcting p-values for multiple comparisons.
...	Other parameters for kruskal.test.

Details

To facilitate code interpretation, It is recommended to use this function in conjunction with the calc_compare function:

```
nem_compare <- nem |> calc_compare(.group = con_crop, y = pH, method = KruskalTest)
```

Value

An compare-class object.

References

R in Action: Data Analysis and Graphics with R, Second Edition by Robert I. Kabacoff, published by Manning Publications. 178 South Hill Drive, Westampton, NJ 08060 USA. Copyright 2015 by Manning Publications.

See Also

Other functions related to differential analysis methods: TTest2, TTest, WilcoxTest2, WilcoxTest, KruskalTest2, LSD, LSD2, HSD, HSD2.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_test <- nem |>
            calc_compare(.group = Treatments,
              y = Mesorhabditis,
              method = KruskalTest)
nem_test
```

KruskalTest2 *Perform Kruskal-Wallis test on easynem meta-table by treatment (two-factor)*

Description

The KruskalTest2() is used to perform Kruskal-Wallis test for grouped data and create compare2-class. This function is only applicable to two-factor analysis, see KruskalTest for a single factor version of the function.

Usage

```
KruskalTest2(data, .group1, .group2, y, p.adj = "none", ...)
```

Arguments

data	An easynem-class data.
.group1	Grouping variables factor 1.
.group2	Grouping variables factor 2.
y	Dependent variable (numeric data).
p.adj	method for correcting p-values for multiple comparisons. Default p.adj = "none".
...	Other parameters for kruskal.test.

Details

To facilitate code interpretation, It is recommended to use this function in conjunction with the calc_compare2 function:

```
nem_compare <- nem |> calc_compare2(.group1 = con_crop, .group2 = season, y = pH, method = KruskalTest)
```

Value

An compare2-class object.

References

R in Action: Data Analysis and Graphics with R, Second Edition by Robert I. Kabacoff, published by Manning Publications. 178 South Hill Drive, Westampton, NJ 08060 USA. Copyright 2015 by Manning Publications.

See Also

Other functions related to differential analysis methods: TTest2, TTest, WilcoxTest2, WilcoxTest, KruskalTest, LSD, LSD2, HSD, HSD2.

Examples

```
nem <- read_nem(tab = easynem_example("nemtab1.csv"),
                tax = easynem_example("nemtax1.csv"),
                meta = easynem_example("nemmeta1.csv"))
nem_test <- nem |>
          calc_compare2(.group1 = con_crop, .group2 = season, y = pH, method = KruskalTest2)
nem_test
```

lme-class *A S4 class to store the linear regression analysis results (single factor)*

Description

The lme-class is used to store the results of linear regression analysis.

Slots

meta Stores the data frame for plotting.

result A data frame for storing the results of linear regression analysis.

See Also

The constructor, calc_lm; Visualization function, nem_plot.

lme2-class *A S4 class to store the linear regression analysis results (two-factor)*

Description

The lme2-class is used to store the results of linear regression analysis.

Slots

meta Stores the data frame for plotting

result A data frame for storing the results of linear regression analysis.

See Also

The constructor, calc_lm2; Visualization function, nem_plot.

LSD *Multiple comparisons, "Least significant difference" and Adjust P-values (single factor)*

Description

The LSD() is used to perform "Least significant difference" for grouped data and create compare-class. This function is only applicable to single factor analysis, see LSD2 for a two factor version of the function.

Usage

```
LSD(data, .group, y, ...)
```

Arguments

data	An easynem-class data.
.group	Grouping variables.
y	Dependent variable (numeric data).
...	Other parameters for LSD.test.

Details

To facilitate code interpretation, It is recommended to use this function in conjunction with the calc_compare function:

```
nem_compare <- nem |> calc_compare(.group = con_crop, y = pH, method = LSD)
```

Value

An compare-class object.

See Also

Other functions related to differential analysis methods: TTest2, TTest, WilcoxTest2, WilcoxTest, KruskalTest2, KruskalTest, LSD2, HSD, HSD2.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_test <- nem |>
            calc_compare(.group = Treatments,
              y = Mesorhabditis,
              method = LSD)
nem_test
```

LSD2 *Multiple comparisons, "Least significant difference" and Adjust P-values (two-factor)*

Description

The LSD2() is used to perform "Least significant difference" for grouped data and create compare2-class. This function is only applicable to two-factor analysis, see LSD for a single factor version of the function.

Usage

```
LSD2(data, .group1, .group2, y, ...)
```

Arguments

data	An easynem-class data.
.group1	Grouping variables factor 1.
.group2	Grouping variables factor 2.
y	Dependent variable (numeric data).
...	Other parameters for LSD.test.

Details

To facilitate code interpretation, It is recommended to use this function in conjunction with the calc_compare2 function:

```
nem_compare <- nem |> calc_compare2(.group1 = con_crop, .group2 = season, y = pH, method = LSD2)
```

Value

An compare2-class object.

See Also

Other functions related to differential analysis methods: TTest2, TTest, WilcoxTest2, WilcoxTest, KruskalTest2, KruskalTest, LSD, HSD, HSD2.

Examples

```
nem <- read_nem(tab = easynem_example("nemtab1.csv"),
                tax = easynem_example("nemtax1.csv"),
                meta = easynem_example("nemmeta1.csv"))
nem_test <- nem |>
             calc_compare2(.group1 = con_crop, .group2 = season, y = pH, method = LSD2)
nem_test
```

mf-class *A S4 class to store Metabolic footprints results (single factor)*

Description

The `mf-class` is used to store the results of nematode metabolic footprints analysis.

Slots

`result` A data frame for storing the results of metabolic footprinting analysis.

See Also

The constructor, `calc_mf`; Visualization function, `nem_plot`.

mf2-class *A S4 class to store Metabolic footprints results (two-factor)*

Description

The `mf2-class` is used to store the results of nematode metabolic footprints analysis.

Slots

`result` A data frame for storing the results of metabolic footprinting analysis.

See Also

The constructor, `calc_mf2`; Visualization function, `nem_plot`.

nem_calc *calc*

Description

For microbial or nematode community calculations.

Usage

```
nem_calc(data, f, ...)
```

Arguments

data	easynem type data.
f	Function parameters for microbial or nematode community calculations.
...	Other parameters.

Value

easynem or other data types.

---

nem_database *Nematode database search*

Description

This function provides a visual interface for retrieving basic data of nematodes. The database used is from http://nemaplex.ucdavis.edu/Ecology/EcophysiologyParms/EcoParameterMenu.html

Usage

```
nem_database()
```

Value

A web interface

References

http://nemaplex.ucdavis.edu/Ecology/EcophysiologyParms/EcoParameterMenu.html

---

nem_plot *Visualize the results of the calculation*

Description

The nem_plot() is used to visualize the calculation results and is a generalized function for multiple classes including beta-class, beta2-class, compare-class, compare2-class, ef-class, ef2-class, funguild-class, funguild2-class, mf-class, mf2-class, ter-class, ter2-class, etc.

Usage

```
nem_plot(object, ...)
```

Arguments

object	beta-class or other types data (beta2-class, compare-class, compare2-class, ef-class, ef2-class, funguild-class, funguild2-class, mf-class, mf2-class, ter-class, ter2-class, etc.).
...	Other parameters to be expanded.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_plot <- nem |> calc_beta(pca, Treatments, method = "bray") |> nem_plot()
```

Value

A plot object. Typically a ggplot object for most classes, or a recordedplot object for ter-class and ter2-class ternary plots.

---

nem_plot,beta-method *Visualization of beta diversity results (single factor)*

Description

The nem_plot function is generalized to the beta-class and is used to visualize the single-factor beta diversity results.

Usage

```
## S4 method for signature 'beta'
nem_plot(object, level = 0.6, type = 1, ...)
```

Arguments

object	A beta-class object.
level	Used to adjust the size of the confidence ellipse. Default level = 0.6. See stat_ellipse.
type	Method used to adjust the display of scatter area. type = 1, displays as a confidence ellipse; type = 2, displays as a polygon. Default type = 1.
...	Other parameters to be expanded.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_plot <- nem |> calc_beta(pca, Treatments, method = "bray") |> nem_plot()
```

Value

An gg or ggplot object.

See Also

The nem_plot() is used to visualize the calculation results and is a generalized function for multiple classes including beta-class, beta2-class, compare-class, compare2-class, ef-class, ef2-class, funguild-class, funguild2-class, mf-class, mf2-class, ter-class, ter2-class, etc.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_plot <- nem |>
            calc_beta(pcoa, Treatments, method = "bray") |>
            nem_plot(level = 0)
nem_plot
nem_plot <- nem |>
            calc_beta(nmds, Treatments, method = "bray") |>
            nem_plot(type = 2)
nem_plot
```

---

nem_plot,beta2-method *Visualization of beta diversity results (two-factor)*

Description

The nem_plot function is generalized to the beta2-class and is used to visualize the two-factor beta diversity results.

Usage

```
## S4 method for signature 'beta2'
nem_plot(object, level = 0.6, type = 1, ...)
```

Arguments

object	A beta2-class object.
level	Used to adjust the size of the confidence ellipse. Default level = 0.6. See stat_ellipse.
type	Method used to adjust the display of scatter area. type = 1, displays as a confidence ellipse; type = 2, displays as a polygon. Default type = 1.
...	Other parameters to be expanded.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_plot <- nem |> calc_beta2(pca, con_crop, season, method = "bray") |> nem_plot()
```

Value

An gg or ggplot object.

See Also

The nem_plot() is used to visualize the calculation results and is a generalized function for multiple classes including beta-class, beta2-class, compare-class, compare2-class, ef-class, ef2-class, funguild-class, funguild2-class, mf-class, mf2-class, ter-class, ter2-class, etc.

Examples

```
nem <- read_nem(tab = easynem_example("nemtab1.csv"),
                tax = easynem_example("nemtax1.csv"),
                meta = easynem_example("nemmeta1.csv"))
nem_plot <- nem |>
            calc_beta2(pcoa, con_crop, season, method = "bray") |>
            nem_plot(level = 0)
nem_plot
nem_plot <- nem |>
            calc_beta2(nmds, con_crop, season, method = "bray") |>
            nem_plot(type = 2)
nem_plot
```

---

nem_plot,compare-method

Visualizing the results of multiple comparisons (single factor)

Description

The nem_plot function is generalized to the compare-class and is used to visualize the results of single-factor multiple comparisons.

Usage

```
## S4 method for signature 'compare'
nem_plot(object, type = 1, add, ...)
```

Arguments

object	A compare-class object.
type	type = 1, draws a box plot; type = 2 draws a bar plot. Default type = 1.
add	Add standard deviation or standard error (only used when drawing a bar plot).
...	Other parameters to be expanded.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_plot <- nem |> calc_compare(.group = con_crop, y = pH, method = LSD) |> nem_plot()
```

Value

An gg or ggplot object.

See Also

The nem_plot() is used to visualize the calculation results and is a generalized function for multiple classes including beta-class, beta2-class, compare-class, compare2-class, ef-class, ef2-class, funguild-class, funguild2-class, mf-class, mf2-class, ter-class, ter2-class, etc.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_plot <- nem |> calc_compare(.group = Treatments,
              y = Mesorhabditis,
              method = LSD) |>
            nem_plot()
nem_plot
nem_plot <- nem |> calc_compare(.group = Treatments,
              y = Mesorhabditis,
              method = HSD) |>
            nem_plot(type = 2, add = "mean_se")
nem_plot
```

---

nem_plot,compare2-method

Visualizing the results of multiple comparisons (two-factor)

Description

The nem_plot function is generalized to the compare2-class and is used to visualize the results of two-factor multiple comparisons.

Usage

```
## S4 method for signature 'compare2'
nem_plot(object, type1 = 1, type2 = 1, add, ...)
```

Arguments

object	A compare2-class object.
type1	type1 = 1, draws a box plot; type1 = 2, draws a bar plot. Default type1 = 1.
type2	type2 = 1, draw a cluster plot; type2 = 2, draws faceted plot. Default type2 = 1.
add	Add standard deviation or standard error (only used when drawing a bar plot).
...	Other parameters to be expanded.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_plot <- nem |> calc_compare2(.group1 = con_crop, .group2 = season, y = pH, method = LSD) |> nem_plot(
```

Value

An gg or ggplot object.

See Also

The nem_plot() is used to visualize the calculation results and is a generalized function for multiple classes including beta-class, beta2-class, compare-class, compare2-class, ef-class, ef2-class, funguild-class, funguild2-class, mf-class, mf2-class, ter-class, ter2-class, etc.

Examples

```
nem <- read_nem(tab = easynem_example("nemtab1.csv"),
                tax = easynem_example("nemtax1.csv"),
                meta = easynem_example("nemmeta1.csv"))
nem_plot <- nem |> calc_compare2(.group1 = con_crop,
                    .group2 = season, y = pH, method = LSD2) |>
                    nem_plot(type2 = 2)
nem_plot
nem_plot <- nem |> calc_compare2(.group1 = con_crop,
                    .group2 = season, y = pH, method = HSD2) |>
                    nem_plot(type1 = 2, type2 = 2, add = "mean_sd")
nem_plot
```

`nem_plot,ef-method` *Visualizing the energy structure of nematode communities (single factor)*

Description

The `nem_plot` function is generalized to the `ef-class` and is used to visualize the energy structure of nematode communities. a five-node food web was constructed with bacterivores, fungivores and herbivores receiving energy from basal resources (R), omnivores-carnivores receiving energy from other nodes. Numbers along the lines represented energy flux (ug C / 100 g dry soil / day). The size of nodes corresponds to the fresh biomass (ug / 100 g dry soil). Uniformity (U) of soil nematode energetic structure (unitless, mean ± standard error) was calculated as the ratio of the mean of summed energy flux through each energy channel to the standard deviation of these mean values.

Usage

```
## S4 method for signature 'ef'
nem_plot(object)
```

Arguments

`object`	A `ef-class` object.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol `|>` to connect functions:

```
nem_plot <- nem |> calc_nemindex() |> calc_ef(Treatments) |> nem_plot()
```

Value

An gg or `ggplot` object.

References

- Wan, Bingbing, et al. "Organic amendments increase the flow uniformity of energy across nematode food webs." Soil Biology and Biochemistry 170 (2022): 108695.
- Ferris, H., 2010. Form and function: metabolic footprints of nematodes in the soil food web. European Journal of Soil Biology 46, 97–104.
- Van Den Hoogen, Johan, et al. "Soil nematode abundance and functional group composition at a global scale." Nature 572.7768 (2019): 194-198.
- Barnes, A.D., Jochum, M., Mumme, S., Haneda, N.F., Farajallah, A., Widarto, T.H., Brose, U., 2014. Consequences of tropical land use for multitrophic biodiversity and ecosystem functioning. Nature Communications 5, 1–7.
- De Ruiter, P.C., Van Veen, J.A., Moore, J.C., Brussaard, L., Hunt, H.W., 1993. Calculation of nitrogen mineralization in soil food webs. Plant and Soil 157, 263–273.

See Also

The nem_plot() is used to visualize the calculation results and is a generalized function for multiple classes including beta-class, beta2-class, compare-class, compare2-class, ef-class, ef2-class, funguild-class, funguild2-class, mf-class, mf2-class, ter-class, ter2-class, etc.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_plot <- nem |>
            calc_nemindex() |>
            calc_ef(Treatments) |>
            nem_plot()
nem_plot
```

---

nem_plot,ef2-method *Visualizing the energy structure of nematode communities (two-factor)*

Description

The nem_plot function is generalized to the ef2-class and is used to visualize the energy structure of nematode communities. a five-node food web was constructed with bacterivores, fungivores and herbivores receiving energy from basal resources (R), omnivores-carnivores receiving energy from other nodes. Numbers along the lines represented energy flux (ug C / 100 g dry soil / day). The size of nodes corresponds to the fresh biomass (ug / 100 g dry soil). Uniformity (U) of soil nematode energetic structure (unitless, mean ± standard error) was calculated as the ratio of the mean of summed energy flux through each energy channel to the standard deviation of these mean values.

Usage

```
## S4 method for signature 'ef2'
nem_plot(object)
```

Arguments

object	A ef2-class object.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_plot <- nem |> calc_nemindex() |> calc_ef2(con_crop, season) |> nem_plot()
```

Value

An gg or ggplot object.

References

- Wan, Bingbing, et al. "Organic amendments increase the flow uniformity of energy across nematode food webs." Soil Biology and Biochemistry 170 (2022): 108695.
- Ferris, H., 2010. Form and function: metabolic footprints of nematodes in the soil food web. European Journal of Soil Biology 46, 97–104.
- Van Den Hoogen, Johan, et al. "Soil nematode abundance and functional group composition at a global scale." Nature 572.7768 (2019): 194-198.
- Barnes, A.D., Jochum, M., Mumme, S., Haneda, N.F., Farajallah, A., Widarto, T.H., Brose, U., 2014. Consequences of tropical land use for multitrophic biodiversity and ecosystem functioning. Nature Communications 5, 1–7.
- De Ruiter, P.C., Van Veen, J.A., Moore, J.C., Brussaard, L., Hunt, H.W., 1993. Calculation of nitrogen mineralization in soil food webs. Plant and Soil 157, 263–273.

See Also

The `nem_plot()` is used to visualize the calculation results and is a generalized function for multiple classes including `beta-class`, `beta2-class`, `compare-class`, `compare2-class`, `ef-class`, `ef2-class`, `funguild-class`, `funguild2-class`, `mf-class`, `mf2-class`, `ter-class`, `ter2-class`, etc.

Examples

```
nem <- read_nem(tab = easynem_example("nemtab1.csv"),
                tax = easynem_example("nemtax1.csv"),
                meta = easynem_example("nemmeta1.csv"))
nem_plot <- nem |>
            calc_nemindex() |>
            calc_ef2(con_crop, season) |>
            nem_plot()
nem_plot
```

`nem_plot,funguild-method`

Visualizing nematode functional guild data (single factor)

Description

The `nem_plot` function is generalized to the `funguild-class` and is used to visualize the nematode functional guild data.

Usage

```
## S4 method for signature 'funguild'
nem_plot(object)
```

Arguments

object	A funguild-class object.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_plot <- nem |> calc_nemindex() |> calc_funguild(Treatments) |> nem_plot()
```

Value

An gg or ggplot object.

See Also

The nem_plot() is used to visualize the calculation results and is a generalized function for multiple classes including beta-class, beta2-class, compare-class, compare2-class, ef-class, ef2-class, funguild-class, funguild2-class, mf-class, mf2-class, ter-class, ter2-class, etc.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_plot <- nem |>
            calc_nemindex() |>
            calc_funguild(Treatments) |>
            nem_plot()
nem_plot
```

---

nem_plot,funguild2-method

Visualizing nematode functional guild data (two-factor)

Description

The nem_plot function is generalized to the funguild2-class and is used to visualize the nematode functional guild data.

Usage

```
## S4 method for signature 'funguild2'
nem_plot(object)
```

Arguments

object	A funguild2-class object.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_plot <- nem |> calc_nemindex() |> calc_funguild2(con_crop, season) |> nem_plot()
```

Value

An gg or ggplot object.

See Also

The nem_plot() is used to visualize the calculation results and is a generalized function for multiple classes including beta-class, beta2-class, compare-class, compare2-class, ef-class, ef2-class, funguild-class, funguild2-class, mf-class, mf2-class, ter-class, ter2-class, etc.

Examples

```
nem <- read_nem(tab = easynem_example("nemtab1.csv"),
                tax = easynem_example("nemtax1.csv"),
                meta = easynem_example("nemmeta1.csv"))
nem_plot <- nem |>
            calc_nemindex() |>
            calc_funguild2(con_crop, season) |>
            nem_plot()
nem_plot
```

---

nem_plot,lme-method *Visualizing the results of linear regression (single factor)*

Description

The nem_plot function is generalized to the lme-class and is used to visualize the results of linear regression.

Usage

```
## S4 method for signature 'lme'
nem_plot(object, ...)
```

Arguments

object	A lme-class object.
...	Other parameters of stat_cor function.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_plot <- nem |> calc_lm(Treatments, Chao1, TotalBiomass) |> nem_plot()
```

Value

An gg or ggplot object.

See Also

The nem_plot() is used to visualize the calculation results and is a generalized function for multiple classes including beta-class, beta2-class, compare-class, compare2-class, ef-class, ef2-class, funguild-class, funguild2-class, mf-class, mf2-class, ter-class, ter2-class, etc.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_plot <- nem |>
            calc_alpha() |>
            calc_nemindex() |>
            calc_lm(group = Treatments,
                    x = Chao1,
                    y = TotalBiomass) |>
            nem_plot()
nem_plot
```

---

nem_plot,lme2-method *Visualizing the results of linear regression (two-factor)*

Description

The nem_plot function is generalized to the lme2-class and is used to visualize the results of linear regression.

Usage

```
## S4 method for signature 'lme2'
nem_plot(object, ...)
```

Arguments

object	A lme2-class object.
...	Other parameters of stat_cor function.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_plot <- nem |> calc_lm2(con_crop, season, x = SOC, y = pH) |> nem_plot()
```

Value

An gg or ggplot object.

See Also

The nem_plot() is used to visualize the calculation results and is a generalized function for multiple classes including beta-class, beta2-class, compare-class, compare2-class, ef-class, ef2-class, funguild-class, funguild2-class, mf-class, mf2-class, ter-class, ter2-class, etc.

Examples

```
nem <- read_nem(tab = easynem_example("nemtab1.csv"),
                tax = easynem_example("nemtax1.csv"),
                meta = easynem_example("nemmeta1.csv"))
nem_lm <- nem |> calc_lm2(con_crop, season, x = pH, y = Fe) |> nem_plot()
nem_lm
```

---

nem_plot,mf-method *Visualizing the metabolic footprint of nematode communities (single factor)*

Description

The nem_plot function is generalized to the mf-class and is used to visualize the metabolic footprint of nematode communities. Metabolic footprints quantify the amplitude of Carbon utilisation by different food web components. The point in the middle of a rhombus represents the intersection of EI and SI and length of vertical and horizontal axes of the rhombus corresponds to the footprints of enrichment and structure components respectively.

Usage

```
## S4 method for signature 'mf'
nem_plot(object, kei = 1, ksi = 1)
```

Arguments

object	A mf-class object.
kei	Adjust the width of the diamond, default kei = 1.
ksi	Adjust the length of the diamond, default ksi = 1.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_plot <- nem |> calc_nemindex() |> calc_mf(Treatments) |> nem_plot()
```

Value

An gg or ggplot object.

References

- https://shiny.wur.nl/ninja/
- Ferris, Howard. "Form and function: metabolic footprints of nematodes in the soil food web." European Journal of Soil Biology 46.2 (2010): 97-104.

See Also

The nem_plot() is used to visualize the calculation results and is a generalized function for multiple classes including beta-class, beta2-class, compare-class, compare2-class, ef-class, ef2-class, funguild-class, funguild2-class, mf-class, mf2-class, ter-class, ter2-class, etc.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_plot <- nem |>
            calc_nemindex() |>
            calc_mf(Treatments) |>
            nem_plot(kei = 30, ksi = 20)
nem_plot
```

---

nem_plot,mf2-method *Visualizing the metabolic footprint of nematode communities (two-factor)*

Description

The nem_plot function is generalized to the mf2-class and is used to visualize the metabolic footprint of nematode communities. Metabolic footprints quantify the amplitude of Carbon utilisation by different food web components. The point in the middle of a rhombus represents the intersection of EI and SI and length of vertical and horizontal axes of the rhombus corresponds to the footprints of enrichment and structure components respectively.

Usage

```
## S4 method for signature 'mf2'
nem_plot(object, kei = 1, ksi = 1)
```

Arguments

object	A mf2-class object.
kei	Adjust the width of the diamond, default kei = 1.
ksi	Adjust the length of the diamond, default ksi = 1.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_plot <- nem |> calc_nemindex() |> calc_mf2(con_crop, season) |> nem_plot()
```

Value

An gg or ggplot object.

References

- https://shiny.wur.nl/ninja/
- Ferris, Howard. "Form and function: metabolic footprints of nematodes in the soil food web." European Journal of Soil Biology 46.2 (2010): 97-104.

See Also

The nem_plot() is used to visualize the calculation results and is a generalized function for multiple classes including beta-class, beta2-class, compare-class, compare2-class, ef-class, ef2-class, funguild-class, funguild2-class, mf-class, mf2-class, ter-class, ter2-class, etc.

Examples

```
nem <- read_nem(tab = easynem_example("nemtab1.csv"),
                tax = easynem_example("nemtax1.csv"),
                meta = easynem_example("nemmeta1.csv"))
nem_plot <- nem |>
            calc_nemindex() |>
            calc_mf2(con_crop, season) |>
            nem_plot(kei = 35, ksi = 35)
nem_plot
```

nem_plot,ter-method *Visualizing the results of the ternary analysis (single factor)*

Description

The nem_plot function is generalized to the ter-class and is used to visualize the results of the ternary analysis. This function visualizes the distribution of nematode communities using the relative abundance of nematodes of cp1, cp2, and cp3-5 or the relative biomass of herbivorous nematodes, bacterivorous nematodes, and fungivorous nematodes as the three axes of a ternary plot.

Usage

```
## S4 method for signature 'ter'
nem_plot(object, type, point_size = 1, legend_cex = 0.9, ...)
```

Arguments

object	A ter-class object.
type	Visualize the nematodes by their feeding habits or by their cp values.
point_size	Size of the points. Default is 1.
legend_cex	Size of the legend text. Default is 0.9
...	Additional parameters passed to Ternary::TernaryPlot().

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_plot <- nem |> calc_ter(Treatments) |> nem_plot()
```

Value

A recordedplot object from Ternary::TernaryPlot.

References

- https://shiny.wur.nl/ninja/
- Goede, RGM de, T. Bongers, and C. H. Ettema. "Graphical presentation and interpretation of nematode community structure: cp triangles." (1993): 743-750.

See Also

The nem_plot() is used to visualize the calculation results and is a generalized function for multiple classes including beta-class, beta2-class, compare-class, compare2-class, ef-class, ef2-class, funguild-class, funguild2-class, mf-class, mf2-class, ter-class, ter2-class, etc.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_plot <- nem |>
            calc_ter(Treatments) |>
            nem_plot(type = feeding)
nem_plot
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_plot <- nem |>
            calc_ter(Treatments) |>
            nem_plot(type = cp)
nem_plot
```

---

nem_plot,ter2-method *Visualizing the results of the ternary analysis (two-factor)*

Description

The nem_plot function is generalized to the ter2-class and is used to visualize the results of the ternary analysis. This function visualizes the distribution of nematode communities using the relative abundance of nematodes of cp1, cp2, and cp3-5 or the relative biomass of herbivorous nematodes, bacterivorous nematodes, and fungivorous nematodes as the three axes of a ternary plot.

Usage

```
## S4 method for signature 'ter2'
nem_plot(object, type, point_size = 1, legend_cex = 0.9, ...)
```

Arguments

object	A ter2-class object.
type	Visualize the nematodes by their feeding habits or by their cp values.
point_size	Size of the points. Default is 1.
legend_cex	Size of the legend text. Default is 0.9
...	Additional parameters passed to Ternary::TernaryPlot().

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_plot <- nem |> calc_ter2(con_crop, season) |> nem_plot()
```

Value

A recordedplot object from Ternary::TernaryPlot.

References

- https://shiny.wur.nl/ninja/
- Goede, RGM de, T. Bongers, and C. H. Ettema. "Graphical presentation and interpretation of nematode community structure: cp triangles." (1993): 743-750.

See Also

The nem_plot() is used to visualize the calculation results and is a generalized function for multiple classes including beta-class, beta2-class, compare-class, compare2-class, ef-class, ef2-class, funguild-class, funguild2-class, mf-class, mf2-class, ter-class, ter2-class, etc.

Examples

```
nem <- read_nem(tab = easynem_example("nemtab1.csv"),
                tax = easynem_example("nemtax1.csv"),
                meta = easynem_example("nemmeta1.csv"))
nem_plot <- nem |>
            calc_ter2(con_crop, season) |>
            nem_plot(type = feeding)
nem_plot
nem_plot <- nem |>
            calc_ter2(con_crop, season) |>
            nem_plot(type = cp)
nem_plot
```

---

nem_trans *nem_trans*

Description

Used to convert and filter easynem type data.

Usage

```
nem_trans(data, f, ...)
```

Arguments

data	easynem type data.
f	Function parameters for data filtering and transformation.
...	Other parameters.

Value

An easynem object.

nemindex-class *Class for storing nematode ecological index calculation results*

Description

The nemindex-class is an extension of the easynem-class to store the results of nematode ecological index calculations.

Slots

result The calculation results of storage nematode ecological index.

See Also

The constructor, calc_nemindex; Visualization function, nem_plot.

nemmeta *Experimental Design of Kiwifruit Cover Crops*

Description

Experimental design table of "Responses of soil nematode abundance and food web to cover crops in a kiwifruit orchard". The variables are as follows:

Usage

```
nemmeta
```

Format

A tibble with 12 rows and 2 variables:

SampleID IDs of different observations, corresponding to the column names of nemtab

Treatments Diversity of cover crops in different observations: CK has no cover crops, C2 has two cover crops, C4 has four cover crops, and C8 has eight cover crops

References

This dataset referenced from "Li Q-m, Qi X-X, Zhang H-f, Zhang Y-j, Liu H-m, Zhao J-n, Yang D and Wang H (2023) Responses of soil nematode abundance and food web to cover crops in a kiwifruit orchard. Front. Plant Sci. 14:1173157. doi: 10.3389/fpls.2023.1173157"

Examples

```
data(nemmeta)
head(nemmeta)
```

nemtab *Table of soil nematode abundance in kiwifruit orchards*

Description

Abundance (individuals / 100 g dry soil) of nematodes functional guilds under different cover crop diversity treatments. The variables are as follows:

Usage

```
nemtab
```

Format

A tibble with 46 rows and 13 variables (The numbers after _ in the columns represent the replicates of each treatment):

OTUID Taxonomic ID of nematodes

CK No cover crop

C2 Two cover crop species

C4 Four cover crop species

C8 Eight cover crop species

References

This dataset referenced from "Li Q-m, Qi X-X, Zhang H-f, Zhang Y-j, Liu H-m, Zhao J-n, Yang D and Wang H (2023) Responses of soil nematode abundance and food web to cover crops in a kiwifruit orchard. Front. Plant Sci. 14:1173157. doi: 10.3389/fpls.2023.1173157"

Examples

```
data(nemtab)
head(nemtab)
```

nemtax *Nematode taxonomy table*

Description

Nematode taxonomy table corresponding to the nematode taxonomy ID in the nematode abundance table. The first column of this table corresponds to the first column in nemtab. If calculations related to nematode communities are to be performed, the taxonomy table should be accurate to at least the family and genus level. The variables are as follows:

Usage

```
nemtax
```

Format

A tibble with 46 rows and 5 variables:

OTUID Taxonomic ID of nematodes.This column corresponds to the first column of nemtab and cannot have duplicate values.

Kingdom Classification of nematodes at the kingdom level.

Phylum Classification of nematodes at the Phylum level.When reading in data, this R package will determine whether the table is a nematode classification table based on whether the Phylum column in the classification table contains Nematoda. Therefore, if you want to use this package to analyze the nematode community structure, the Phylum in the classification table must be Nematoda, otherwise the read-in data will not be automatically associated with the nematode database.

Family Classification of nematodes at the Family level.

Genus Classification of nematodes at the Genus level.

References

This dataset referenced from "Li Q-m, Qi X-X, Zhang H-f, Zhang Y-j, Liu H-m, Zhao J-n, Yang D and Wang H (2023) Responses of soil nematode abundance and food web to cover crops in a kiwifruit orchard. Front. Plant Sci. 14:1173157. doi: 10.3389/fpls.2023.1173157"

Examples

```
data(nemtax)
head(nemtax)
```

---

order_factor *order_factor*

Description

Meta attributes of easynem grouping factors in order to rearrangement.

Usage

```
order_factor(data, group, order)
```

Arguments

data	easynem type data.
group	Selection of meta columns.
order	Order of factors.

Value

An easynem object.

`read_nem` *Build easynem-class objects from their csv file*

Description

`read_nem()` is a constructor method. This is the main method suggested for constructing an experiment-level (`easynem-class`) object from its component data (component data: `tab`, `tax`, `meta`).

Usage

```
read_nem(tab = 0, tax = 0, meta = 0, ...)
```

Arguments

`tab`	Nematode abundance table.
`tax`	Nematode abundance table.
`meta`	Experimental design table.
`...`	Other default parameters for `read_csv` function.

Value

An easynem object. The components in the class are interconnected to facilitate the subsequent screening and management of nematode data. When this class is generated, it will automatically check whether there is nematode information in the species classification table. If not, it will not be associated with the nematode database.

See Also

`read_nem2`

Examples

```
easynem <- read_nem(tab = easynem_example("nemtab.csv"),
                    tax = easynem_example("nemtax.csv"),
                    meta = easynem_example("nemmeta.csv"))
show(easynem)
```

read_nem2 *Build easynem-class objects from their tibble type object*

Description

read_nem2() is a constructor method. This is the main method suggested for constructing an experiment-level (easynem-class) object from its tibble type object (component data: tab, tax, meta).

Usage

```
read_nem2(tab = 0, tax = 0, meta = 0, ...)
```

Arguments

tab	Nematode abundance table.
tax	Nematode abundance table.
meta	Experimental design table.
...	Other default parameters for read_csv function.

Value

An easynem object. The components in the class are interconnected to facilitate the subsequent screening and management of nematode data. When this class is generated, it will automatically check whether there is nematode information in the species classification table. If not, it will not be associated with the nematode database.

See Also

read_nem

Examples

```
easynem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
show(easynem)
```

ter-class *A S4 class to store the ternary analysis results (single factor)*

Description

The `ter-class` is used to store the results of nematode ternary analysis.

Slots

`result` A data frame for storing the results of ternary analysis.

See Also

The constructor, `calc_ter`; Visualization function, `nem_plot`.

ter2-class *A S4 class to store the ternary analysis results (two-factor)*

Description

The `ter2-class` is used to store the results of nematode ternary analysis.

Slots

`result` A data frame for storing the results of ternary analysis.

See Also

The constructor, `calc_ter2`; Visualization function, `nem_plot`.

trans_combine *Merge multiple columns of easynem's meta*

Description

The `trans_combine()` is used for the special case of merging columns in easynem's meta. For example, `Cp35%` (the sum of percentages from `Cp3` to `Cp5`) is often used in nematode community analysis. This function can quickly merge `Cp3` to `Cp5`.

Usage

```
trans_combine(data, col)
```

Arguments

data	An easynem-class data.
col	The name of the column to be summed.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_trans <- nem |> trans_combine(c("3", "4", "5"))
```

Value

An easynem-class data.

See Also

Other functions in this package for filtering and transforming data sets: filter_name, trans_formula, trans_formula_v, trans_name, filter_num, trans_norm, trans_rare

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_trans <- nem |>
             trans_name(cp_value) |>
             trans_norm(method = percent) |>
             trans_combine(c("3", "4", "5"))
show(nem_trans)
nem_trans@meta$`3_4_5`
```

---

trans_formula *Formula conversion for easynem's meta*

Description

The trans_formula() is used to convert the formula of easynem meta. Formula transformation is sometimes necessary in nematode community analysis. For example, to ensure that the data is normally distributed, it is often necessary to perform ln(x+1) transformation or other forms of formula transformation on nematode abundance. This function only works on a single variable. For a vectorized variant of this function, see trans_formula_v.

Usage

```
trans_formula(data, var, formu)
```

Arguments

data	An easynem-class data.
var	Variable name to be converted.
formu	Formula parameters for data conversion. Such as ~log(x+1).

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_trans <- nem |> trans_formula(Mesorhabditis, ~log(x+1))
```

Value

An easynem-class data that stores the result of formula conversion.

See Also

Other functions in this package for filtering and transforming data sets: filter_name, filter_num, trans_formula_v, trans_name, trans_norm, trans_rare, trans_combine

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_trans <- nem |> trans_formula(Mesorhabditis, ~log(x+1))
show(nem_trans)
```

---

trans_formula_v *Formula conversion for easynem's meta (Vectorization)*

Description

The trans_formula_v() is used to convert the formula of easynem meta. Formula transformation is sometimes necessary in nematode community analysis. For example, to ensure that the data is normally distributed, it is often necessary to perform ln(x+1) transformation or other forms of formula transformation on nematode abundance. This function can transfer vectors to achieve multi-variable formula conversion. For a univariate simplified version of this function, see trans_formula.

Usage

```
trans_formula_v(data, var, formu)
```

Arguments

data	An easynem-class data.
var	Vectorized variable names for formula conversion.
formu	Formula parameters for data conversion. Such as ~log(x+1).

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_trans <- nem |> trans_formula_v(colnames(resultmeta)[5:10], ~log(x+1))
```

Value

An easynem-class data that stores the result of formula conversion.

See Also

Other functions in this package for filtering and transforming data sets: filter_name, filter_num, trans_formula, trans_name, trans_norm, trans_rare, trans_combine

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_trans <- nem |> trans_formula_v(nem@tab$OTUID, ~log(x+1))
show(nem_trans)
```

---

trans_name *Reorganize the easynem's tab by taxonomic name*

Description

The trans_name() is used to re-summarize the nematode abundance table by nematode taxonomy table.

Usage

```
trans_name(data, taxonomy)
```

Arguments

data	An easynem-class data.
taxonomy	Nematode taxonomic name or other nematode attributes.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_trans <- nem |> trans_name(Family)
```

Value

A reclassified and aggregated easynem-class.

Feedings

Since the nematode taxonomy table is automatically associated with the nematode database (nem_database) including feeding and cp_value when reading data through read_nem or read_nem2, feeding can also be passed as a parameter to trans_name(). The corresponding relationship between the feeding value and the actual nematode feeding habits is as follows:

- feeding = 1, plant feeding
- feeding = 2, fungal hyphal feeding
- feeding = 3, bacterial feeding
- feeding = 4, substrate ingestion
- feeding = 5, predation (including specialist predators of nematodes)
- feeding = 6, eucaryote feeding
- feeding = 7, dispersal stages or animal parasites
- feeding = 8, omnivory (including general predators of nematodes)

See Also

Other functions in this package for filtering and transforming data sets: filter_name, trans_formula, trans_formula_v, filter_num, trans_norm, trans_rare, trans_combine

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_trans <- nem |> trans_name(Family)
show(nem_trans)
nem_trans <- nem |> trans_name(feeding)
show(nem_trans)
```

---

trans_norm *Normalizing the nematode abundance table*

Description

The trans_norm() is an extension of the decostand function of the vegan package for easynem-class data, which is used to standardize the nematode abundance table to reduce the order of magnitude differences of nematodes in each treatment.

Usage

```
trans_norm(data, method, MARGIN = 2, ...)
```

Arguments

data	An easynem-class data.
method	Standardization method. For details, refer to the decostand function of the vegan package.
MARGIN	Margin, 1 = rows, and 2 = columns of easynem's tab. Default MARGIN = 2.
...	Other parameters of the decostand function of the vegan package.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_trans <- nem |> trans_norm(method = total)
```

Value

A normalized easynem-class data.

See Also

Other functions in this package for filtering and transforming data sets: filter_name, trans_formula, trans_formula_v, trans_name, filter_num, trans_rare, trans_combine

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_trans <- nem |> trans_norm(method = total)
colSums(nem_trans@tab[,-1])
nem_trans <- nem |> trans_norm(method = percent)
colSums(nem_trans@tab[,-1])
```

---

trans_rare — *Randomly rarefied OTU or ASV tables of nematodes for amplicon sequencing data*

Description

The trans_rare() is an extension of the rrarefy function of the vegan package for easynem-class data, which is used to randomly rarefied OTU or ASV tables of nematodes for amplicon sequencing data. The default is to rare according to the minimum abundance of nematode in each treatment.

Usage

```
trans_rare(data, sample = 0, ...)
```

Arguments

data	An easynem-class data.
sample	Subsample size for rarefying community. The default sample = 0, the minimum abundance is used for rarefied OTU or ASV tables of nematodes.
...	Other parameters of the rrarefy function of the vegan package.

Details

To facilitate code interpretation, it is recommended to use the pipe symbol |> to connect functions:

```
nem_trans <- nem |> trans_rare(1500)
```

Value

A rarefied easynem-class data.

See Also

Other functions in this package for filtering and transforming data sets: filter_name, trans_formula, trans_formula_v, trans_name, filter_num, trans_norm, trans_combine

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_trans <- nem |> trans_rare()
colSums(nem_trans@tab[,-1])
nem_trans <- nem |> trans_rare(1500)
colSums(nem_trans@tab[,-1])
```

TTest *Perform t-test on easynem meta-table by treatment (single factor)*

Description

The TTest() is used to perform t-test for grouped data and create compare-class. This function is only applicable to single factor analysis, see TTest2 for a two factor version of the function.

Usage

```
TTest(data, .group, y, ...)
```

Arguments

data	An easynem-class data.
.group	Grouping variables (supports only two groups).
y	Dependent variable (numeric data).
...	Other parameters for t.test.

Details

Note: The t-test is only applicable to comparisons between two groups of data. To facilitate code interpretation, It is recommended to use this function in conjunction with the calc_compare function:

```
nem_compare <- nem |> calc_compare(.group = con_crop, y = pH, method = TTest)
```

Value

An compare-class object.

See Also

Other functions related to differential analysis methods: TTest2, WilcoxTest, WilcoxTest2, KruskalTest, KruskalTest2, LSD, LSD2, HSD, HSD2.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_ttest <- nem |>
              filter_name(meta, Treatments %in% c("CK", "C8")) |>
              calc_compare(.group = Treatments, y = Mesorhabditis, method = TTest)
nem_ttest
```

TTest2 *Perform t-test on easynem meta-table by treatment (two-factor)*

Description

The TTest2() is used to perform t-test for grouped data and create compare2-class. This function is only applicable to two-factor analysis, see TTest for a single factor version of the function.

Usage

```
TTest2(data, .group1, .group2, y, ...)
```

Arguments

data	An easynem-class data.
.group1	Grouping variables factor 1 (supports only two groups).
.group2	Grouping variables factor 2 (supports only two groups).
y	Dependent variable (numeric data).
...	Other parameters for t.test.

Details

Note: The t-test is only applicable to comparisons between two groups of data. To facilitate code interpretation, It is recommended to use this function in conjunction with the calc_compare2 function:

```
nem_compare <- nem |> calc_compare2(.group1 = con_crop, .group2 = season, y = pH, method = TTest2)
```

Value

An compare2-class object.

See Also

Other functions related to differential analysis methods: TTest, WilcoxTest, WilcoxTest2, KruskalTest, KruskalTest2, LSD, LSD2, HSD, HSD2.

Examples

```
nem <- read_nem(tab = easynem_example("nemtab1.csv"),
                tax = easynem_example("nemtax1.csv"),
                meta = easynem_example("nemmeta1.csv"))
nem_ttest <- nem |>
              filter_name(meta, con_crop %in% c("Y2", "Y11")) |>
             calc_compare2(.group1 = con_crop, .group2 = season, y = pH, method = TTest2)
nem_ttest
```

WilcoxTest *Perform wilcoxon-test on easynem meta-table by treatment (single factor)*

Description

The WilcoxTest() is used to perform wilcoxon-test for grouped data and create compare-class. This function is only applicable to single factor analysis, see WilcoxTest2 for a two factor version of the function.

Usage

```
WilcoxTest(data, .group, y, ...)
```

Arguments

data	An easynem-class data.
.group	Grouping variables (supports only two groups).
y	Dependent variable (numeric data).
...	Other parameters for wilcox.test.

Details

Note: The wilcoxon-test is only applicable to comparisons between two groups of data. To facilitate code interpretation, It is recommended to use this function in conjunction with the calc_compare function:

```
nem_compare <- nem |> calc_compare(.group = con_crop, y = pH, method = WilcoxTest)
```

Value

An compare-class object.

See Also

Other functions related to differential analysis methods: TTest2, TTest, WilcoxTest2, KruskalTest, KruskalTest2, LSD, LSD2, HSD, HSD2.

Examples

```
nem <- read_nem2(tab = nemtab, tax = nemtax, meta = nemmeta)
nem_test <- nem |>
              filter_name(meta, Treatments %in% c("CK", "C8")) |>
              calc_compare(.group = Treatments, y = Mesorhabditis, method = WilcoxTest)
nem_test
```

WilcoxTest2 *Perform wilcoxon-test on easynem meta-table by treatment (two-factor)*

Description

The WilcoxTest2() is used to perform wilcoxon-test for grouped data and create compare2-class. This function is only applicable to two-factor analysis, see WilcoxTest for a single factor version of the function.

Usage

```
WilcoxTest2(data, .group1, .group2, y, ...)
```

Arguments

data	An easynem-class data.
.group1	Grouping variables factor 1 (supports only two groups).
.group2	Grouping variables factor 2 (supports only two groups).
y	Dependent variable (numeric data).
...	Other parameters for wilcox.test.

Details

Note: The `wilcoxon-test` is only applicable to comparisons between two groups of data. To facilitate code interpretation, It is recommended to use this function in conjunction with the `calc_compare` function:

```
nem_compare <- nem |> calc_compare2(.group1 = con_crop, .group2 = season, y = pH, method = WilcoxTest2)
```

Value

An `compare2-class` object.

See Also

Other functions related to differential analysis methods: `TTest2`, `TTest`, `WilcoxTest`, `KruskalTest`, `KruskalTest2`, `LSD`, `LSD2`, `HSD`, `HSD2`.

Examples

```
nem <- read_nem(tab = easynem_example("nemtab1.csv"),
                tax = easynem_example("nemtax1.csv"),
                meta = easynem_example("nemmeta1.csv"))
nem_test <- nem |>
              filter_name(meta, con_crop %in% c("Y2", "Y11")) |>
          calc_compare2(.group1 = con_crop, .group2 = season, y = pH, method = WilcoxTest2)
nem_test
```

Index

∗ **datasets**
    nemmeta, 59
    nemtab, 60
    nemtax, 60

adonis2, *6, 7*
alpha-class, 3

beta-class, 4
beta2-class, 4

calc_alpha, *3*, 5, *6–9*, *12*, *14–19*, *23–25*
calc_beta, *4, 5*, 6, *7–9*, *12*, *14–19*, *23–25*
calc_beta2, *4–6*, 7, *8, 9*, *12*, *14–19*, *23–25*
calc_compare, *5–7*, 8, *9*, *12*, *14–19*, *23–25*, *33*, *35*, *38*, *71*, *73, 74*
calc_compare2, *5–8*, 9, *12*, *14–19*, *23–26*, *34*, *36*, *39*, *72*
calc_ef, *5–9*, 10, *14–19*, *23–25*, *27*
calc_ef2, *5–9*, 12, *12*, *15–19*, *23–25*, *28*
calc_funguild, *5–9*, *12*, 14, *14*, *16–19*, *23–25*, *30*
calc_funguild2, *5–9*, *12*, *14*, 15, *15–19*, *23–25*, *30*
calc_lm, 16, *17*, *37*
calc_lm2, *16*, 17, *37*
calc_mf, *5–9*, *12*, *14–17*, 18, *19*, *23–25*, *40*
calc_mf2, *5–9*, *12*, *14–18*, 19, *23–25*, *40*
calc_nemindex, *5–9*, *12*, *14–19*, 20, *24, 25*, *59*
calc_ter, *5–9*, *12*, *14–16*, *18, 19*, 23, *23*, *25*, *64*
calc_ter2, *5–9*, *12*, *14–19*, *23*, 24, *24*, *64*
cmdscale, *6, 7*
compare-class, 25
compare2-class, 26

decostand, *68, 69*
diversity, *5*

easynem-class, 26
easynem_example, 27

ef-class, 27
ef2-class, 28
estimateR, *5*

filter, *28*
filter_name, 28, *30*, *65–70*
filter_num, 29, *29*, *65–70*
fisher.alpha, *5*
funguild-class, 30
funguild2-class, 30

geom_encircle, 31

HSD, *8, 9*, 32, *33–36*, *38, 39*, *71–74*
HSD2, *8, 9*, *32*, 33, *33*, *35, 36*, *38, 39*, *71–74*

kruskal.test, *34*, *36*
KruskalTest, *8, 9*, *33*, 34, *34–36*, *38, 39*, *71–74*
KruskalTest2, *8, 9*, *33, 34*, 35, *35*, *38, 39*, *71–74*

lm, *16, 17*
lme-class, 37
lme2-class, 37
LSD, *8, 9*, *33–36*, 38, *39*, *71–74*
LSD.test, *38, 39*
LSD2, *8, 9*, *33–36*, *38*, 39, *71–74*

mf-class, 40
mf2-class, 40

nem_calc, 40
nem_database, 41, *68*
nem_plot, *3, 4*, *25–28*, *30*, *37*, *40*, 41, *42–45*, *47–54*, *56, 57*, *59*, *64*
nem_plot,beta-method, 42
nem_plot,beta2-method, 43
nem_plot,compare-method, 44
nem_plot,compare2-method, 45
nem_plot,ef-method, 47

nem_plot,ef2-method, 48
nem_plot,funguild-method, 49
nem_plot,funguild2-method, 50
nem_plot,lme-method, 51
nem_plot,lme2-method, 52
nem_plot,mf-method, 53
nem_plot,mf2-method, 54
nem_plot,ter-method, 56
nem_plot,ter2-method, 57
nem_trans, 58
nemindex-class, 59
nemmeta, 59
nemtab, 60
nemtax, 60

order_factor, 61

read_csv, *62, 63*
read_nem, *26*, 62, *63*, *68*
read_nem2, *26*, *62*, 63, *68*
rrarefy, *69, 70*

simpson.unb, *5*
specnumber, *5*
stat_cor, *51, 52*
stat_ellipse, *42, 43*

t.test, *8, 9*, *70, 71*
ter-class, 64
ter2-class, 64
trans_combine, *29, 30*, 64, *66–70*
trans_formula, *29, 30*, 65, *65–70*
trans_formula_v, *29, 30*, *65*, 66, *66*, *68–70*
trans_name, *29, 30*, *65, 66*, 67, *67*, *69, 70*
trans_norm, *29, 30*, *65–67*, 68, *68*, *70*
trans_rare, *29, 30*, *65–68*, 69, *69*
TTest, *8, 9*, *33–36*, *38, 39*, 70, *71–74*
TTest2, *8, 9*, *33–36*, *38, 39*, *70*, 71, *71*, *73, 74*
TukeyHSD, *33*

vegdist, *6, 7*

wilcox.test, *72, 73*
WilcoxTest, *8, 9*, *33–36*, *38, 39*, *71*, 72, *72–74*
WilcoxTest2, *8, 9*, *33–36*, *38, 39*, *71, 72*, 73, *73*