Package ‘spectre’

July 23, 2025

Type Package

Title Predict Regional Community Composition

Version 1.0.2

Maintainer Craig Simpkins <`simpkinscraig063@gmail.com`>

Description Predict regional community composition at a fine spatial resolution using only sparse biological and environmental data. The package is based on the DynamicFOAM algorithm described in Mokany et al. (2011) <doi:10.1111/j.1461-0248.2011.01675.x>.

License GPL-3

Depends R (>= 3.5)

Imports ggplot2, Rcpp (>= 1.0.1)

Suggests dplyr, testthat (>= 3.0.2), knitr, rmarkdown, covr, RcppProgress

LinkingTo Rcpp, RcppProgress, testthat

ByteCompile true

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

SystemRequirements C++11

VignetteBuilder knitr

NeedsCompilation yes

Author Craig Simpkins [aut, cre],
Sebastian Hanss [aut],
Maximilian Hesselbarth [aut],
Matthias Spangenberg [aut],
Jan Salecker [aut]

Repository CRAN

Date/Publication 2021-07-17 06:30:02 UTC

Contents

alpha_list 2
calc_commonness_error 2
estimated_gamma 3
generate_commonness_matrix_from_gdm 3
minimal_example_data 4
plot_commonness 5
plot_error 5
run_optimization_min_conf 6
spectre 7
target_matrix 8

Index **9**

`alpha_list` *Matrix of predicted alpha diversity in each cell.*

Description

Matrix of predicted alpha diversity in each cell.

Usage

```
alpha_list
```

Format

vector.

`calc_commonness_error` *calc_commonness_error*

Description

Calculate commonness error

Usage

```
calc_commonness_error(x, objective_matrix)
```

Arguments

`x`	Results object from run_optimization_min_conf.
`objective_matrix`	Matrix from (modeled) alpha-diversity and Bray-Curtis dissimilarity

Details

Calculate mean absolute commonness error (MAE_c) and relative commonness error in percentage (RCE).

Value

vector

`estimated_gamma` *Total (estimated) species in the system.*

Description

Total (estimated) species in the system.

Usage

```
estimated_gamma
```

Format

numeric

`generate_commonness_matrix_from_gdm`
generate_commonness_matrix_from_gdm

Description

Creates a pairwise site by site commonness matrix from estimates of species richness and Bray-Curtis dissimilarity.

Usage

```
generate_commonness_matrix_from_gdm(gdm_predictions, alpha_list)
```

Arguments

`gdm_predictions`	a square pairwise `matrix` of Bray-Curtis dissimilarity estimates between site pairs. We recommend using the gdm-package (Fitzpatrick et al. 2020) to generate this matrix
`alpha_list`	a `vector` of species richness for every site in the study area. The length of this vector must be equivalent to one of the dimensions of the `gdm_predictions`

Details

`generate_commonness_matrix_from_gdm` uses a vector of estimated species richness per site and a pairwise matrix of site by site Bray-Curtis dissimilarity (we recommend using the gdm-package (Fitzpatrick et al. 2020) to generate this matrix) to produce a matrix of the estimated species in common between site pairs (referred to as a commonness matrix). The commonness between sites is calculated using

$$C_{ij} = (1 - \beta_{ij})(S_i + S_j)/2$$

Where β_{ij} is the dissimilarity between sites, C_{ij} is the species in common between sites, and S is the number of species in each site. For more details see Mokany et al 2011.

Value

A pairwise site by site `matrix` of the number of species in common between each site pair, with dimensions equal to that of the provided dissimilarity matrix.

References

Mokany, K., Harwood, T.D., Overton, J.M., Barker, G.M., & Ferrier, S. (2011). Combining α and β diversity models to fill gaps in our knowledge of biodiversity. Ecology Letters, 14(10), 1043-1051.

`minimal_example_data` *List with example data created using the* `gdm` *package*

Description

List with example data created using the `gdm` package

Usage

```
minimal_example_data
```

Format

list

`plot_commonness` *plot_commonness*

Description

Plot commonness between observed and optimized data

Usage

```
plot_commonness(x, target)
```

Arguments

`x`	Results object of run_optimization_min_conf()
`target`	Pairwise matrix of species in common.

Details

Plot a heatmap of commonness between observed data and optimized data. This visual style allows for easier spatial understanding of commonness differences to be ascertained.

Value

ggplot

`plot_error` *plot_error*

Description

Plot the absolute error

Usage

```
plot_error(x)
```

Arguments

`x`	Results object from run_optimization_min_conf

Details

Plot error over time

Value

ggplot

References

xxx

`run_optimization_min_conf`

run_optimization_min_conf

Description

Generate an optimized estimate of community composition (species presences and absences) for every site in the study area.

Usage

```
run_optimization_min_conf(
  alpha_list,
  total_gamma,
  target,
  max_iterations,
  partial_solution = NULL,
  fixed_species = NULL,
  seed = NA,
  verbose = TRUE,
  interruptible = TRUE
)
```

Arguments

`alpha_list`	`Matrix` of predicted alpha diversity (species richness) in each cell.
`total_gamma`	Total number of species present throughout the entire landscape.
`target`	Pairwise matrix of species in common between each site by site pair. Only the upper triangle of the matrix is actually needed.
`max_iterations`	The maximum number of iterations that the optimization algorithm may run through before stopping.
`partial_solution`	An initial `matrix` of species presences and absences for each site in the landscape. The total number of presences must match the estimated species richness of each site.
`fixed_species`	Fixed partial solution with species that are considered as given. Those species are not going to be changed during optimization.
`seed`	Seed for random number generator. Seed must be a positive integer value. `seed = NA` means that a random integer is used as seed.
`verbose`	If `TRUE` (default), a progress report is printed during the optimization run.
`interruptible`	Allow a run to be interrupted before completion.

Details

`run_optimization_min_conf` is the core function of the `spectre` package. The underlying algorithm of this function is adapted from Mokany et al. (2011). A pairwise commonness matrix (having the same structure as the `target` matrix) is calculated from the `partial_solution` matrix and the value difference with the `target` determined. If a difference is present and depending on the set stopping criteria the algorithm continues. A random site in the presence/absence matrix is selected, and a random presence record at this site replaced with an absence. Every absence in the selected site is then individually flipped to a presence and the value difference with the objective recorded. The presence record which resulted in the lowest value difference (minimum conflict) is retained. This cycle continues, with a random site selected every iteration, until the pairwise commonness and objective matrices match or the algorithm runs beyond the `max_iterations`.

Value

A species presence/absence `matrix` of the study landscape.

References

Mokany, K., Harwood, T.D., Overton, J.M., Barker, G.M., & Ferrier, S. (2011). Combining α and β diversity models to fill gaps in our knowledge of biodiversity. Ecology Letters, 14(10), 1043-1051.

`spectre` *spectre*

Description

The goal of `spectre` is to provide an open source tool capable of predicting regional community composition at fine spatial resolutions using only sparse biological and environmental data.

Author(s)

Maintainer: Craig Simpkins `<simpkinscraig063@gmail.com>`

Authors:

- Sebastian Hanss
- Maximilian Hesselbarth
- Matthias Spangenberg
- Jan Salecker

`target_matrix` *Pairwise matrix of species in common.*

Description

Pairwise matrix of species in common.

Usage

```
target_matrix
```

Format

matrix

Index

∗ **datasets**
    alpha_list, 2
    estimated_gamma, 3
    minimal_example_data, 4
    target_matrix, 8

alpha_list, 2

calc_commonness_error, 2

estimated_gamma, 3

generate_commonness_matrix_from_gdm, 3

minimal_example_data, 4

plot_commonness, 5
plot_error, 5

run_optimization_min_conf, 6

spectre, 7
spectre-package *(spectre)*, 7

target_matrix, 8