Package ‘geohabnet’

May 21, 2025

Title Geographical Risk Analysis Based on Habitat Connectivity

Version 2.2

Date 2025-05-14

Description

The 'geohabnet' package is designed to perform a geographically or spatially explicit risk analysis of habitat connectivity. Xing et al (2021) <doi:10.1093/biosci/biaa067> proposed the concept of cropland connectivity as a risk factor for plant pathogen or pest invasions. As the functions in 'geohabnet' were initially developed thinking on cropland connectivity, users are recommended to first be familiar with the concept by looking at the Xing et al paper. In a nutshell, a habitat connectivity analysis combines information from maps of host density, estimates the relative likelihood of pathogen movement between habitat locations in the area of interest, and applies network analysis to calculate the connectivity of habitat locations.

The functions of 'geohabnet' are built to conduct a habitat connectivity analysis relying on geographic parameters (spatial resolution and spatial extent), dispersal parameters (in two commonly used dispersal kernels: inverse power law and negative exponential models), and network parameters (link weight thresholds and network metrics).

The functionality and main extensions provided by the functions in 'geohabnet' to habitat connectivity analysis are

a) Capability to easily calculate the connectivity of locations in a landscape using a single function, such as sensitivity_analysis() or msean().

b) As backbone datasets, the 'geohabnet' package supports the use of two publicly available global datasets to calculate cropland density. The backbone datasets in the 'geohabnet' package include crop distribution maps from Monfreda, C., N. Ramankutty, and J. A. Foley (2008) <doi:10.1029/2007gb002947> ``Farming the planet: 2. Geographic distribution of crop areas, yields, physiological types, and net primary production in the year 2000, Global Biogeochem. Cycles, 22, GB1022'' and International Food Policy Research Institute (2019) <doi:10.7910/DVN/PRFF8V> ``Global Spatially-Disaggregated Crop Production Statistics Data for 2010 Version 2.0, Harvard Dataverse, V4''. Users can also provide any other geographic dataset that represents host density.

c) Because the 'geohabnet' package allows R users to provide maps of host density (as originally in Xing et al (2021)), host landscape density (representing the geographic distribution of either crops or wild species), or habitat distribution (such as host landscape density adjusted by climate suitability) as inputs, we propose the term habitat connectivity.

d) The 'geohabnet' package allows R users to customize parameter values in the habitat connectivity analysis, facilitating context-specific (pathogen- or pest-specific) analyses.

e) The 'geohabnet' package allows users to automatically visualize maps of the habitat connectivity of locations resulting from a sensitivity analysis across all customized parameter combinations. The primary functions are msean() and sensitivity analysis().
Most functions in 'geohabnet' provide three main outcomes: i) A map of mean habitat connectivity across parameters selected by the user, ii) a map of variance of habitat connectivity across the selected parameters, and iii) a map of the difference between the ranks of habitat connectivity and habitat density.
Each function can be used to generate these maps as 'final' outcomes.
Each function can also provide intermediate outcomes, such as the adjacency matrices built to perform the analysis, which can be used in other network analysis.
Refer to article at <`https://garrettlab.github.io/HabitatConnectivity/articles/analysis.html`> to see examples of each function and how to access each of these outcome types.
To change parameter values, the file called 'parameters.yaml' stores the parameters and their values, can be accessed using 'get_parameters()' and set new parameter values with 'set_parameters()'.
Users can modify up to ten parameters.

Encoding UTF-8

RoxygenNote 7.3.2

Imports config (>= 0.3.1), geosphere (>= 1.5.18), igraph (>= 2.0.3), terra (>= 1.7.29), yaml (>= 2.3.7), stats, stringr (>= 1.5.0), memoise (>= 2.0.1), graphics, viridisLite (>= 0.4.2), beepr (>= 1.3), rnaturalearth (>= 0.3.3), tools, methods, future.apply (>= 1.11.0), future (>= 1.33.0), magrittr (>= 2.0.3), ggplot2 (>= 3.5.1), patchwork (>= 1.2.0)

Suggests devtools, knitr, lintr (>= 3.0.2), mockthat, pkgdown, rmarkdown, testthat (>= 3.1.7)

URL `https://garrettlab.github.io/HabitatConnectivity/`,

`https://CRAN.R-project.org/package=geohabnet/`,

`https://github.com/GarrettLab/HabitatConnectivity/tree/main/geohabnet/`,

`https://www.garrettlab.com/`

BugReports `https://github.com/GarrettLab/HabitatConnectivity/issues`

NeedsCompilation no

Author Krishna Keshav [aut, cre],
Aaron Plex [aut] (ORCID: <`https://orcid.org/0000-0001-7317-3090`>),
Garrett Lab [ctb] (https://garrettlab.com),
Karen Garrett [aut] (ORCID: <`https://orcid.org/0000-0002-6578-1616`>),
University of Florida [cph, fnd] (https://www.ufl.edu)

Maintainer Krishna Keshav <`krishnakeshav.pes@gmail.com`>

Repository CRAN

Date/Publication 2025-05-21 09:00:02 UTC

Contents

`.indices` *Internal function to extract risk indices from a list of crop rasters.*

Description

Internal function to extract risk indices from a list of crop rasters.

Usage

```
.indices(crop_rasters)
```

Arguments

`crop_rasters`	List of raster objects.

Value

A list of risk indices.

`connectivity` *Calculate and plot maps of habitat connectivity*

Description

Calculate the mean habitat connectivity across a set of selected parameters, variance in habitat connectivity, and the difference in ranks between mean habitat connectivity and habitat availability. The result is produced in form of maps plotted with predefined graphics settings. Currently, the settings for plot cannot be customized. Default value is `TRUE` for all logical arguments

Usage

```
connectivity(
  host,
  indices,
  global = FALSE,
  east = NULL,
  west = NULL,
  geoscale = NULL,
  res = reso(),
  pmean = TRUE,
  pvar = TRUE,
  pdiff = TRUE,
  outdir = tempdir()
)
```

Arguments

`host`	SpatRaster. A `SpatRaster` object for the spatial distribution of habitat availability (such as host availability or cropland density). Note that a valid input data for geohabnet is a raster layer of habitat availability (such as host availability), in which each grid cell has any values between zero and one. If you are using your own dataset, please also make sure that your raster layer is in the standard coordinate reference system (i.e., EPSG:4326), the only CRS supported by geohabnet for now.
`indices`	SpatRaster. Collection of risk indices.

`global`	Logical. Select `TRUE` if a global analysis is desired, `FALSE` otherwise. `east` and `west` are required when `TRUE`.
`east`	SpatRaster. Collection of risk indices on eastern extent.
`west`	SpatRaster. Collection of risk indices on western extent. When `TRUE`, `geoscale` is ignored. Default is `TRUE`.
`geoscale`	Vector. This refers to the geographical extent for the habitat connectivity analysis when `global` is set to `FALSE`. Default is `NULL`.
`res`	Numeric. This parameter refers to the spatial aggregation factor. This value is the number of cells that are grouped when aggregating a raster layer from fine to coarse spatial resolution to reduce computational costs. Setting this parameter to 1 would not aggregate the raster layers. Default is `reso()`.
`pmean`	Logical. `TRUE` if a map of mean habitat connectivity should be plotted, `FALSE` otherwise.
`pvar`	Logical. `TRUE` if a map of the variance in habitat connectivity should be plotted, `FALSE` otherwise.
`pdiff`	Logical. `TRUE` if a map of the difference in the ranks between the mean habitat connectivity and habitat availability should be plotted, `FALSE` otherwise.
`outdir`	Character. Output directory for saving raster in TIFF format. Default is `tempdir()`.

Details

`indexes` has a list of `spatRaster` objects resulting from the unique combinations of all parameters specified in either `parameters.yaml` or `sean()`. For each unique combination of parameters, an index of habitat connectivity is calculated for each location in a landscape. Then these indices are used to calculate the mean and the variance of habitat connectivity of a location across all specified parameters.

This function will save all the opted plots using - `pmean`, `pvar` and `pdiff`. File will be saved in provided value of `outdir` or `tempdir()`. If `interactive()` is `TRUE`, then plots can be seen in active plot window (e.g., plot panel in Rstudio). The maps are plotted using `SpatRaster` objects. These `SpatRaster` objects are also available as a return value of this function.

Value

Gmap. See details.

References

Yanru Xing, John F Hernandez Nopsa, Kelsey F Andersen, Jorge L Andrade-Piedra, Fenton D Beed, Guy Blomme, Mónica Carvajal-Yepes, Danny L Coyne, Wilmer J Cuellar, Gregory A Forbes, Jan F Kreuze, Jürgen Kroschel, P Lava Kumar, James P Legg, Monica Parker, Elmar Schulte-Geldermann, Kalpana Sharma, Karen A Garrett, *Global Cropland connectivity: A Risk Factor for Invasion and Saturation by Emerging Pathogens and Pests*, BioScience, Volume 70, Issue 9, September 2020, Pages 744–758, doi:10.1093/biosci/biaa067

Hijmans R (2023). *terra: Spatial Data Analysis*. R package version 1.7-46, `https://CRAN.R-project.org/package=terra`

See Also

`hci_mean()`, `hci_variance()`, `hci_diff()`

`dist_methods` *Distance methods supported*

Description

Contains supported strategies to calculate distance between two points. Use of one of two methods in `sean()` or `sensitivity_analysis()`.

Usage

```
dist_methods()
```

Value

vector

Examples

```
dist_methods()
```

`GeoModel-class` *GeoModel class*

Description

A reference class to represent results of dispersal models.

Slots

- `amatrix` A square adjacency matrix that represents the likely movement of a species between locations. In this adjacency matrix, rows and columns are the identification of the locations, and each entry indicates the relative likelihood of a species moving between a pair of locations. An adjacency matrix is produced for each unique value of the dispersal parameters chosen.
- `index` A raster object representing the habitat connectivity index of the locations in the selected region. Note that connectivity is calculated based on a weighted sum of the network metrics chosen by the user. A raster object is produced for each unique value of the dispersal parameters chosen.
- `hdthreshold` A numeric value representing the threshold for habitat availability (e.g., cropland density or host density) used in the sensitivity analysis.
- `aggregation` A character value representing the spatial aggregation method used for aggregating the habitat availability map before conducting the sensitivity analysis.

`linkthreshold` A numeric value representing the threshold for the link weights used to calculate habitat connectivity of each location. Note that link weights indicate the relative likelihood of a species moving between locations (nodes) and correspond to the entries in the adjacency matrix.

`beta` A numeric value representing the beta parameter. The beta parameter is the dispersal parameter in one of two dispersal kernel models (the inverse power law model) included in `geohabnet`.

`gamma` A numeric value representing the gamma parameter. The gamma parameter is the dispersal parameter in one of two dispersal kernel models (the negative exponential model) included in `geohabnet`.

`GeoNetwork-class` *GeoNetwork*

Description

An S4 class representing a network of geographical data. The GeoNetwork object will wrap all the results from the habitat connectivity analysis using `sean()` or `sensitivity_analysis()`. This class contains the field from `Gmap` class, which has results of the habitat connectivity analysis in the form of `SpatRaster` and TIFF file.

Usage

```
## S4 replacement method for signature 'GeoNetwork'
habitat_density(x) <- value
```

Arguments

`x`	GeoNetwork.
`value`	SpatRaster.

Value

GeoNetwork.

Slots

`rasters` A list of `GeoRasters` objects.

`habitat_density` A `SpatRaster` representing the habitat availability (or simply host density).

`me_rast` A raster representing mean habitat connectivity in a region.

`me_out` Character. A file path to where the mean habitat connectivity raster is saved.

`diff_rast` A raster representing the difference in ranks between habitat connectivity and habitat availability.

`diff_out` Character. A file path to where the difference raster is located.

`var_rast` A raster representing the variance in habitat connectivity in a region.

`var_out` Character. A file path to where the variance raster is located.

`GeoRasters-class` *GeoRaster class*

Description

A class to represent raster vis-a-vis risk indices. This class encapsulates the results of apply dispersal models and metrics.

Fields

`host_density` SpatRaster. A spatial raster representing host density.

`rasters` List. List of raster representing risk indices. These are of type `GeoModels`.

`global` Boolean. True if contains `GlobalRast` object, False otherwise.

`geoscale_param` *Get geographical scales from the parameters*

Description

This function returns a list of geographical scales set in global and custom extent in `parameters.yaml`. If `global` is `TRUE`, the `CustomExt` is ignored.

Usage

```
geoscale_param(gparams = load_parameters())
```

Arguments

`gparams`	Optional. `load_parameters()` or null

Value

Vector. A set of geographic coordinates that delimit the extent of a region of the world.

`get_parameters` *Get Parameters*

Description

This function retrieves the list of parameters and saves a copy of the parameter file (of type .yaml) to the specified output path.

Usage

```
get_parameters(out_path = tempdir(), iwindow = FALSE)
```

Arguments

`out_path`	character. The output path where the parameter file will be copied. The default is a temporary directory `tempdir()`
`iwindow`	logical. If window = `TRUE`, this will prompt the user to select the output directory using a file chooser window. The default value is `FALSE`.

Details

Using this configuration where the parameters are structurally listed in a yaml file is an alternative method used in the `sensitivity_analysis()` function. Once the parameter.yaml is saved in a local directory, the user can modify each parameter value, save this file with the changes, and get the new parameters back in R with `set_parameters()`.

Note that the `sean()` or `msean()` function will require to directly list the parameters within the function as it is typical in other R packages.

Value

character. The path to the copied parameter file.

See Also

`set_parameters()`

Examples

```
get_parameters()
get_parameters(out = tempdir())
```

`get_rasters` *Get rasters object from parameters See host object in `get_parameters()` or `load_parameters()`.*

Description

Get rasters object from parameters See host object in `get_parameters()` or `load_parameters()`.

Usage

```
get_rasters(host)
```

Arguments

`host`	SpatRaster. It is synonym to Hosts object in parameters. This is a wrapper to `terra::rast()` and generates a raster object if provided with a TIF file.

Value

List of SpatRaster.

See Also

`load_parameters()`, `get_parameters()`

Examples

```
# Get default raster
get_rasters(terra::rast())
```

`GlobalRast-class` *GlobalRast class*

Description

A class to represent raster objects for global scales. Global scales are accessible using `global_scales()`. However, this class encapsulates the results of apply dispersal models and metrics.

Fields

`east`	A list of raster objects for eastern hemisphere.
`west`	A list of raster objects for western hemisphere.

`global_scales` *Global geographical extent*

Description

This function provides the geographical extent used in a global analysis. This function returns the geographic extents of the eastern and western hemispheres used in the analysis. Each geographic extent is in the form of c(Xmin, Xmax, Ymin, Ymax). The geohabnet functions are designed to work with the coordinate reference system: lon/lat WGS 84 (EPSG:4326).

Usage

```
global_scales()
```

Details

When a habitat connectivity analysis is global, the functions in geohabnet will conduct two separate analyses, one on the geographical scale of the eastern hemisphere and another for the western hemisphere. The final outcomes (such as maps or adjacency matrices) are then combined for the global analysis.

Value

List. Named list with scales for eastern and western hemispheres

See Also

`set_global_scales()`

`Gmap-class` *Gmap class*

Description

An S4 class to organize various maps in the form of `SpatRaster` in a single object.

Usage

```
setmaps(x, me, vari, dif)
```

Arguments

`x`	Gmap object.
`me`	SpatRaster. A raster used as background when plotting the map of mean habitat connectivity.
`vari`	SpatRaster. A raster used as background when plotting the map of variance in habitat connectivity.
`dif`	SpatRaster. A raster used as background when plotting the map of difference in habitat connectivity and habitat availability.

Value

Gmap object.

Slots

`me_rast` A raster object representing habitat connectivity of a region averaged across all selected parameters.

`me_out` Character. A file path to where the mean habitat connectivity is saved.

`diff_rast` A raster object representing the difference in ranks between mean habitat connectivity and habitat availability in a region.

`diff_out` Character. A file path to where the difference raster is saved.

`var_rast` A raster object representing the variance in habitat connectivity of a region calculated across all specified parameters.

`var_out` Character. A file path to where the variance raster is saved.

`gplot` *Plot a Raster* object*

Description

This is a wrapper for `terra::plot()` with customized parameters for an enhanced visualization.

Usage

```
gplot(x, ...)
```

Arguments

`x`	a Raster* object
`...`	additional arguments passed to `terra::plot()`

Value

a plot

Examples

```
r <- terra::rast(nrows=108, ncols=21, xmin=0, xmax=10)
gplot(r)
gplot(r, col = "red")
gplot(r, col = "red", breaks = 10)
```

`habitat_density<-` *Set the habitat density.*

Description

This function helps to set the SpatRaster of the habitat availability or density in a GeoNetwork object. The function is an S4 replacement method in the geohabnet package. It allows you to assign a host availability SpatRaster to a geohabnet object.

Usage

```
habitat_density(x) <- value
```

Arguments

`x`	the GeoNetwork object.
`value`	SpatRaster.

Value

The same object type as x, that is, GeoNetwork. This function returns the updated S4 GeoNetwork object with the new habitat availability SpatRaster assigned.

`hci_diff` *Calculate difference map*

Description

This function produces a map of the difference in ranks between mean habitat connectivity and habitat availability.

Usage

```
hci_diff(x, y, global, geoscale, res = reso(), outdir = tempdir())
```

Arguments

`x`	SpatRaster.
`y`	SpatRaster.
`global`	Logical. Select `TRUE` if a global analysis is desired, `FALSE` otherwise. `east` and `west` are required when `TRUE`.
`geoscale`	Numeric vector. `x` will be cropped to this extent.
`res`	Numeric. This parameter refers to the spatial aggregation factor. This value is the number of cells that are grouped when aggregating a raster layer from fine to coarse spatial resolution to reduce computational costs. Setting this parameter to 1 would not aggregate the raster layers. Default is `reso()`.
`outdir`	Character. Output directory for saving raster in TIFF format. Default is `tempdir()`.

Details

Ideally, the function is tested to yield desired results when `length(which(y[] > 0)) > length(which(x[] > 0))`.

Value

RiskMap. Contains result in the form of `SpatRaster` objects and file path of the saved maps.

`hci_mean` *Calculate mean of habitat connectivity in a region*

Description

Wrapper for `terra::mean()`. Calculates mean of list of rasters for habitat connectivity.

Usage

```
hci_mean(
  indices,
  global = FALSE,
  east = NULL,
  west = NULL,
  geoscale = NULL,
  res = reso(),
  plt = TRUE,
  outdir = tempdir()
)
```

Arguments

`indices`	List of SpatRasters indicating the habitat connectivity of a region. This input represents the spatial raster collection for which mean is to be calculated.
`global`	Logical. Select `TRUE` if a global analysis is desired, `FALSE` otherwise. `east` and `west` are required when `TRUE`.
`east`	SpatRaster. Collection of risk indices on eastern extent.
`west`	SpatRaster. Collection of risk indices on western extent. When `TRUE`, `geoscale` is ignored. Default is `TRUE`.
`geoscale`	Vector. This refers to the geographical extent for the habitat connectivity analysis when `global` is set to `FALSE`. Default is `NULL`.
`res`	Numeric. This parameter refers to the spatial aggregation factor. This value is the number of cells that are grouped when aggregating a raster layer from fine to coarse spatial resolution to reduce computational costs. Setting this parameter to 1 would not aggregate the raster layers. Default is `reso()`.
`plt`	`TRUE` if need to plot mean map, `FALSE` otherwise.
`outdir`	Character. Output directory for saving raster in TIFF format. Default is `tempdir()`.

Value

RiskMap. Contains result in the form of `SpatRaster` objects and file path of the saved maps.

`hci_variance` *Calculate variance of habitat connectivity in a region*

Description

This function produces a map of variance of habitat connectivity across selected parameter values

Usage

```
hci_variance(
  indices,
  rast,
  global,
  east = NULL,
  west = NULL,
  geoscale,
  res = reso(),
  outdir = tempdir()
)
```

Arguments

`indices`	SpatRaster. Collection of risk indices.
`rast`	SpatRaster. Template for variance output
`global`	Logical. Select `TRUE` if a global analysis is desired, `FALSE` otherwise. `east` and `west` are required when `TRUE`.
`east`	SpatRaster. Collection of risk indices on eastern extent.
`west`	SpatRaster. Collection of risk indices on western extent. When `TRUE`, `geoscale` is ignored. Default is `TRUE`.
`geoscale`	Vector. This refers to the geographical extent for the habitat connectivity analysis when `global` is set to `FALSE`. Default is `NULL`.
`res`	Numeric. This parameter refers to the spatial aggregation factor. This value is the number of cells that are grouped when aggregating a raster layer from fine to coarse spatial resolution to reduce computational costs. Setting this parameter to 1 would not aggregate the raster layers. Default is `reso()`.
`outdir`	Character. Output directory for saving raster in TIFF format. Default is `tempdir()`.

Value

RiskMap. Contains result in the form of `SpatRaster` objects and file path of the saved maps.

`inv_powerlaw` *Dispersal kernels*

Description

-`[inv_powerlaw()]` Get parameters and values pertaining to the inverse power law model. -`[neg_exp()]` Get parameters and values pertaining to the negative exponential model.

Usage

```
inv_powerlaw(
  params = load_parameters(),
  betas = NULL,
  mets = NULL,
  we = NULL,
  linkcutoff = NULL
)

neg_expo(
  params = load_parameters(),
  gammas = NULL,
  mets = NULL,
  we = NULL,
  linkcutoff = NULL
)
```

Arguments

`params`	Object. This function uses the parameter values defined in `load_parameters()` by default. If `load_parameters()` is not provided, the function will require the user to specify all arguments as listed below. If both `load_parameters()` and values for the arguments listed below are provided, `load_parameters()` takes precedence over the specified arguments.
`betas`	Numeric. Beta is the dispersal parameter used in the inverse power law to estimate a species' dispersal gradient. Please refer to Mundt et al (2009) for details on how to calculate this parameter. Any beta values should be positive. Smaller beta values indicate a higher likelihood of dispersal between nodes.
`mets`	Character. There are seven network metrics supported by `geohabnet`: "node_strength", "sum_of_nearest_neighbors", "eigenvector_centrality", "closeness", "betweeness", "degree", and "page_rank". Each specified network metric is calculated for each location in the target region, based on the link weights between each pair of locations. Run, for example, `pagerank()` for details of each network metric.
`we`	Numeric. This parameter indicates the weight(s) of each specified network metric, representing the importance of the network metric in the analysis. Since these weights represent percentages, any weight(s) should be between 0 and 100, and the sum of all specified weights should be 100.
`linkcutoff`	Numeric. This parameter is only used to calculate `betweeness()` and `closeness()`, and is equivalent to `cutoff` in these functions in the `igraph` package.
`gammas`	Numeric. Gamma is the dispersal parameter used in the negative exponential to estimate a species' dispersal gradient. Any gamma values should be positive. Smaller gamma values indicate a higher likelihood of dispersal between nodes.

Details

Refer to Esker et al (2007) for a discussion on the characteristics of each dispersal gradient or kernel model (i.e., inverse power law and negative exponential). The resulting object produced by `load_parameters()` provides the following values used when running the analysis `-beta` is a dispersal parameter for calculating the inverse power law model. `-gamma` is a dispersal parameter for calculating the negative exponential model. `-metrics` Each network metric is applied to the adjacency matrix produced in the intermediate step. `-weights` The link weights that is used in the network analysis. `-cutoff` Currently used as a parameter to calculate two types of node centrality - `betweeness()` and `closeness()`. As defined in `igraph::betweenness()`, `cutoff` refers to the maximum length to consider when calculating centrality. If zero or negative, then there is no such limit.

Value

List with parameters and values. See details.

References

Esker PD, Sparks AH, Antony G, Bates M, Dall' Acqua W, Frank EE, Huebel L, Segovia V, Garrett KA (2007). "Ecology and Epidemiology in R: Modeling dispersal gradients." *The Plant Health Instructor*. doi:10.1094/PHIA2008012903

Mundt CC, Sackett KE, Wallace LD, Cowger C, Dudley JP (2009). “Aerial Dispersal and Multiple-Scale Spread of Epidemic Disease.” *Ecohealth*. doi:10.1007/s103930090251z

Csardi G, Nepusz T (2006). “The igraph software package for complex network research.” *InterJournal*, *Complex Systems*, 1695. `https://igraph.org`.

Csárdi G, Nepusz T, Traag V, Horvát Sz, Zanini F, Noom D, Müller K (2024). *igraph: Network Analysis and Visualization in R*. doi:10.5281/zenodo.7682609, R package version 1.5.1, `https://CRAN.R-project.org/package=igraph`.

See Also

`supported_metrics()`

`load_parameters` *Load Parameters from YAML File*

Description

This function loads parameters from a YAML file and stores them in an object.

Usage

```
load_parameters(filepath = .param_fp())
```

Arguments

`filepath`	Path to the YAML file containing the parameters. By default, it takes the value of `parameters.yaml` in R user’s directory.

Value

object with parameters and values

Examples

```
# Load parameters from default file
load_parameters()
```

`ndplot` *Network density plot*

Description

This function first calculates the network density for each dispersal parameter specified by the user. Network density compares the number of available links in a network versus the total number of possible links in the same network. Network density is a measure of how well an entire network is, ranging from 0 (not connected at all) to 1 (fully connected). This function then compares visually how network density changes with changes in dispersal parameter values. In other words, it calculates and plots the network density of a GeoNetwork object.

Usage

```
ndplot(x)
```

Arguments

`x`	A GeoNetwork object

Value

Vector. Up to two ggplot2 objects, one for the dispersal parameter values in the negative exponential model and one for the dispersal parameter values in the inverse power law model.

`ndplot,GeoNetwork-method` *Network density*

Description

This function first calculates the network density for each dispersal parameter specified by the user. Network density compares the number of available links in a network versus the total number of possible links in the same network. Network density is a measure of how well an entire network is, ranging from 0 (not connected at all) to 1 (fully connected). Calculates and plots the network density of a GeoNetwork object.

Usage

```
## S4 method for signature 'GeoNetwork'
ndplot(x)
```

Arguments

`x`	A GeoNetwork object

Value

Vector. Up to two ggplot2 objects

`nn_sum` *Calculation on network metrics a.k.a centralities.*

Description

These are functions under the `igraph` package adapted to calculate habitat connectivity. In the context of habitat connectivity, the functions can be interpreted as follows:

- `nn_sum()`: Calculates the sum of nearest neighbors `igraph::knn()`.
- `node_strength()`: Calculates the sum of edge weights of adjacent nodes `igraph::strength()`.
- `betweenness()`: Calculates the node betweenness based on the number of shortest paths. Because the `igraph::betweenness()` function in interprets link weights as distances to calculate the shortest paths, the `betweeness()` function in geohabnet transforms the link weights (or the relative likelihood of pathogen or pest movement) in the adjacency matrix so that higher link weight values will be the shortest (or more likely) paths for pathogen or pest movement.
- `ev()`: Calculates the eigenvector centrality of positions within the network `igraph::eigen_centrality()`.
- `closeness()`: measures how many steps is required to access every other vertex from a given vertex `igraph::closeness()`. Because the `igraph::closeness()` function interprets link weights as distances to calculate the shortest paths, this transforms the link weights (or the relative likelihood of pathogen or pest movement) in the adjacency matrix so that higher link weight values will be the shortest (or more likely) paths for pathogen or pest movement.
- `degree()`: number of adjacent edges `igraph::degree()`.
- `pagerank()`: page rank score for vertices `igraph::page_rank()`.

Usage

```
nn_sum(crop_dm, ...)

node_strength(crop_dm, ...)

betweeness(crop_dm, ...)

ev(crop_dm, ...)

degree(crop_dm, ...)

closeness(crop_dm, ...)

pagerank(crop_dm, ...)
```

Arguments

`crop_dm`	A square adjacency matrix, in which rows and columns names represent nodes (or locations) and each entry indicate the relative likelihood of pathogen, pest, or other species movement between a pair of nodes. In the internal workflow, the adjacency matrix comes as a result of operations within `sean()` function. This weight represents the importance of the network metric in the habitat connectivity analysis.
`...`	arguments to corresponding funtions in `igraph`

Value

SpatRaster. Representing connectivity of each node or location.

References

Csardi G, Nepusz T (2006). "The igraph software package for complex network research." *InterJournal*, *Complex Systems*, 1695. `https://igraph.org`.

See Also

Other metrics: `supported_metrics()`

`reset_params` *Reset parameters.yaml*

Description

Resets the values in the `parameters.yaml` file to the default initial values.

Usage

```
reset_params()
```

Value

Logical. `TRUE` if function was successfully executed

Examples

```
reset_params()
```

`reso` *Get resolution value*

Description

Resolution stored in `parameter.yaml`. Here, resolution values refer to the aggregation factor or granularity. Granularity is the number of small grid cells that are aggregated into larger grid cells in each direction (horizontally and vertically). For example, the finest spatial resolution of the Monfreda and MAPSPAM dataset in geohabnet is 5 minutes, a granularity value of 6 will result in maps with a spatial resolution of 0.5 degrees. If not provided, the resolution value used for the analysis is by default 12 (or two degrees when using the Monfreda and MAPSPAM dataset). Otherwise, a single integer value for granularity equal to or greater than one should be specified.

Usage

```
reso()
```

Value

Numeric. Resolution from `parameters.yaml`. The default is 12.

`RiskMap-class` *RiskMap class*

Description

An S4 class representing resulting maps from the specific operation type.

Fields

`map` Character. A file path to the map.

`riid` SpatRaster. This is one of the maps of habitat connectivity.

`spr` SpatRaster. A spatial raster representing the habitat connectivity index.

`fp` Character. A file path to the habitat connectivity raster.

`risk_indices` *Get risk indices*

Description

Get a habitat connectivity index for each unique combination of parameters from GeoRasters object.

Usage

```
risk_indices(ri)
```

Arguments

`ri`	GeoRasters object

Details

This function will unpack SpatRasters from GeoModel and thus is `future::future()` safe.

Value

List of habitat connectivity indices. If the `ri` is global, the list will contain two elements, one for each hemisphere. e.g. `list(east = list(), west = list())`. If the `ri` is not global, the list will contain a single element, e.g. `list()`.

`sa_onrasters` *Run sensitivity analysis*

Description

Same as `sensitivity_analysis()` but it takes raster object and other parameters as an input.

- `sa_onrasters()` is a wrapper around `sean()` function. Takes raster object and other parameters as an input.
- `msean_onrast()` same as `sa_onrasters()`. Use this for side effects + results. Produces and plots the maps for the outcomes and results are returned as an object. It produces and plots the maps for the outcomes and results are returned as an object.

Usage

```
sa_onrasters(rast, link_thresholds = c(0), hd_thresholds = c(0), ...)

msean_onrast(
  global = TRUE,
  geoscale = NULL,
  res = reso(),
  outdir = tempdir(),
  ...
)
```

Arguments

`rast`	Raster object which will be used in analysis.
`link_thresholds`	Numeric vector. link threshold values
`hd_thresholds`	Numeric vector. host density threshold values
`...`	Additional parameters to be passed to `sean()`.
`global`	Logical. `TRUE` if global analysis, `FALSE` otherwise. Default is `TRUE`
`geoscale`	Numeric vector. Geographical coordinates in the form of c(Xmin, Xmax, Ymin, Ymax) which EPSG:4326 in coordinate reference system. If `geoscale` is NuLL, the extent is extracted from `rast`(SpatRaster) using `terra::ext()`.
`res`	Numeric. The spatial aggregation factor that will be used to aggregate the raster layer of habitat availability, from fine to coars resolution. Default is `reso()`.
`outdir`	Character. Output directory for saving raster in TIFF format. Default is `tempdir()`.

Details

Error not handled for non-overlapping extents.

Value

A list of calculated CCRI indices after operations. An index is generated for each combination of paramters. One combination is equivalent to `sean()` function.

References

Yanru Xing, John F Hernandez Nopsa, Kelsey F Andersen, Jorge L Andrade-Piedra, Fenton D Beed, Guy Blomme, Mónica Carvajal-Yepes, Danny L Coyne, Wilmer J Cuellar, Gregory A Forbes, Jan F Kreuze, Jürgen Kroschel, P Lava Kumar, James P Legg, Monica Parker, Elmar Schulte-Geldermann, Kalpana Sharma, Karen A Garrett, *Global Cropland connectivity: A Risk Factor for Invasion and Saturation by Emerging Pathogens and Pests*, BioScience, Volume 70, Issue 9, September 2020, Pages 744–758, doi:10.1093/biosci/biaa067

Hijmans R (2023). *terra: Spatial Data Analysis*. R package version 1.7-46, `https://CRAN.R-project.org/package=terra`

See Also

Use `get_rasters()` to obtain raster object.

`msean_onrast()`

`sean` *Sensitivity analysis across maps of habitat connectivity*

Description

This function performs a sensitivity analysis across different values of habitat connectivity for each location in a map. For each combination of selected parameters, an index of habitat connectivity is calculated. `sensitivity_analysis()` is a wrapper around `sean()` function.

- `msean()` is a wrapper around `sean()` function. It has additional argument to specify maps which are calculated using `connectivity()` function.

Usage

```
sean(
  rast,
  global = TRUE,
  geoscale = NULL,
  agg_methods = c("sum", "mean"),
  dist_method = "geodesic",
  link_threshold = 0,
  hd_threshold = 0,
  res = reso(),
  inv_pl = inv_powerlaw(NULL, betas = c(0.5, 1, 1.5), mets = c("betweeness",
   "NODE_STRENGTH", "Sum_of_nearest_neighbors", "eigenVector_centrAlitY"), we = c(50,
    15, 15, 20), linkcutoff = -1),
  neg_exp = neg_expo(NULL, gammas = c(0.05, 1, 0.2, 0.3), mets = c("betweeness",
   "NODE_STRENGTH", "Sum_of_nearest_neighbors", "eigenVector_centrAlitY"), we = c(50,
    15, 15, 20), linkcutoff = -1)
)

msean(
  rast,
  global = TRUE,
  geoscale = NULL,
  res = reso(),
  ...,
  outdir = tempdir()
)
```

Arguments

`rast`	Raster object which will be used in analysis.
`global`	Logical. `TRUE` if global analysis, `FALSE` otherwise. Default is `TRUE`
`geoscale`	Numeric vector. Geographical coordinates in the form of c(Xmin, Xmax, Ymin, Ymax) which EPSG:4326 in coordinate reference system. If `geoscale` is NuLL, the extent is extracted from `rast`(SpatRaster) using `terra::ext()`.
`agg_methods`	Character. One or both methods of spatial aggregation - SUM, MEAN. Aggregation strategy for scaling the input raster to the desired resolution.
`dist_method`	Character. The method to calculate the distance matrix.
`link_threshold`	Numeric. A threshold value for link weight. All link weights that are below this threshold will be replaced with zero for the connectivity analysis. Link weights represent the relative likelihood of pathogen, pest, or invasive species movement between a pair of host locations, which is calculated using gravity models based on host density (or availability) and dispersal kernels.
`hd_threshold`	Numeric. A threshold value for habitat availability (e.g., cropland density or host density). All locations with a host density below the selected threshold will be excluded from the connectivity analysis, which focuses the analysis on the most important locations. The values for the habitat availability threshold can range between 0 and 1; if value is 1, all locations will be excluded from the analysis and 0 will include all locations in the analysis. Selecting a threshold for, for example, host density requires at least knowing what is the maximum value in the host density map to avoid excluding all locations in the analysis. if value is 1, all locations will be excluded from the analysis and 0 will include all locations in the analysis. Selecting a threshold for host density requires at least knowing what is the maximum value in the host density map to avoid excluding all locations in the analysis.
`res`	Numeric. The spatial aggregation factor that will be used to aggregate the raster layer of habitat availability, from fine to coars resolution. Default is `reso()`.
`inv_pl`	List. A named list of parameters for inverse power law. See details.
`neg_exp`	List. A named list of parameters for inverse negative exponential. See details. All locations with a host density below the selected threshold will be excluded from the connectivity analysis, which focuses the analysis on the most important locations. The values for the host density threshold can range between 0 and 1;
`...`	arguments passed to `sean()`
`outdir`	Character. Output directory for saving raster in TIFF format. Default is `tempdir()`.

Details

When `global = TRUE`, `geoscale` is ignored and `global_scales()` is used by default.

The functions `sean()` and `msean()` perform the same sensitivity analysis, but they differ in their return value. The return value of `msean()` is `GeoNetwork`, which contains the result from applying the `connectivity()` function on the habitat connectivity indexes. Essentially, the risk maps.

If neither the inverse power law nor the negative exponential dispersal kernel is specified, the function will return an error.

In `msean()`, three spatRasters are produced with the following values. For each location in the area of interest, the mean in habitat connectivity across selected parameters is calculated. For each location in the area of interest, the variance in habitat connectivity across selected parameters is calculated. For each location in the area of interest, the difference between the rank of habitat connectivity and the rank of host density is calculated. By default, each of these spatRasters is plotted for visualization.

Value

GeoRasters.

GeoNetwork.

References

Yanru Xing, John F Hernandez Nopsa, Kelsey F Andersen, Jorge L Andrade-Piedra, Fenton D Beed, Guy Blomme, Mónica Carvajal-Yepes, Danny L Coyne, Wilmer J Cuellar, Gregory A Forbes, Jan F Kreuze, Jürgen Kroschel, P Lava Kumar, James P Legg, Monica Parker, Elmar Schulte-Geldermann, Kalpana Sharma, Karen A Garrett, *Global Cropland connectivity: A Risk Factor for Invasion and Saturation by Emerging Pathogens and Pests*, BioScience, Volume 70, Issue 9, September 2020, Pages 744–758, doi:10.1093/biosci/biaa067

Hijmans R (2023). *terra: Spatial Data Analysis*. R package version 1.7-46, `https://CRAN.R-project.org/package=terra`

See Also

Uses `connectivity()`

Uses `msean()` `inv_powerlaw()` `neg_expo()`

`sensitivity_analysis` *Sensitivity analysis for habitat connectivity*

Description

This function runs a sensitivity analysis on habitat connectivity calculated based on every combination of selected parameters. Parameter values in `sensitivity_analysis()` should be provided using the function `set_parameters()`. If no parameters are provided, then the `sensitivity_analysis()` function will run the sensitivity analysis using a default set of parameter values, which is accessible through the function `get_parameters()`. To customize parameter values, open the parameters.yaml that was automatically downloaded when geohabnet was installed, change, remove, or add parameter values directly in the parameters.yaml and save it. Once the values have been changed manually, run `set_parameters()` to set the new parameter values, which will return TRUE if the parameters were set successfully.

Usage

```
sensitivity_analysis(maps = TRUE, alert = TRUE)
```

Arguments

`maps`	logical. `TRUE` if maps of outcomes are to be plotted, `FALSE` otherwise. If `TRUE`, three maps are possible: a map of mean habitat connectivity, a map of variance of habitat connectivity, and a map of the difference between the ranks in habitat connectivity and habitat density.
`alert`	logical. `TRUE` if a beep sound is to be played once the analysis is completed, `FALSE` otherwise

Details

For each location in a region, `sensitivity_analysis()` calculates the habitat connectivity risk index (CCRI) proposed by Xing et al. (2021). If you are providing a map of habitat availability (as opposed to simply cropland density or host availability), you could call the output of your sensitivity analysis as the habitat connectivity index, which is a broader term than CCRI. :) By default, sensitivity_analysis() runs a sensitivity analysis on a global extent, see `global_scales()` for details. This function also plots maps of the outcomes automatically, but it will suppress maps for outcomes if `maps` = `FALSE` or `interactive()` is `FALSE`. The returned object is of class `GeoNetwork`, which contains two types of outcomes. One outcome type corresponds to spatRasters representing the maps of habitat connectivity. The second type corresponds to adjacency matrices used to calculate the habitat connectivity, where columns and rows represent locations in the maps and entries are the relative likelihood of pathogen or pest movement between each pair of nodes.

Value

GeoNetwork. Check documentation of the `sean()` function for better explanation of the parameters used.

References

Yanru Xing, John F Hernandez Nopsa, Kelsey F Andersen, Jorge L Andrade-Piedra, Fenton D Beed, Guy Blomme, Mónica Carvajal-Yepes, Danny L Coyne, Wilmer J Cuellar, Gregory A Forbes, Jan F Kreuze, Jürgen Kroschel, P Lava Kumar, James P Legg, Monica Parker, Elmar Schulte-Geldermann, Kalpana Sharma, Karen A Garrett, *Global Cropland connectivity: A Risk Factor for Invasion and Saturation by Emerging Pathogens and Pests*, BioScience, Volume 70, Issue 9, September 2020, Pages 744–758, doi:10.1093/biosci/biaa067

Hijmans R (2023). *terra: Spatial Data Analysis*. R package version 1.7-46, `https://CRAN.R-project.org/package=terra`

See Also

`sa_onrasters()` `sean()` `global_scales()` `get_parameters()` `set_parameters()` `connectivity()`

`setmaps,Gmap-method` *Sets the map slots in the Gmap object.*

Description

This wraps the results(SpatRasters) from the risk analysis.

Usage

```
## S4 method for signature 'Gmap'
setmaps(x, me, vari, dif)
```

Arguments

`x`	A Gmap object.
`me`	A GeoRaster object representing mean habitat connectivity.
`vari`	A GeoRaster object representing variance in habitat connectivity.
`dif`	A GeoRaster object representing the difference in ranks between habitat connectivity and habitat availability.

Value

A Gmap object.

`setprops` *Set properties of the GeoModel object.*

Description

Set properties of the GeoModel object.

Usage

```
setprops(x, aggregation, hdthreshold, linkthreshold)
```

Arguments

`x`	The GeoModel object.
`aggregation`	Character. A value representing the spatial aggregation method used for aggregating the habitat availability map before conducting the sensitivity analysis. There are two aggregation methods available in geohabnet: sum and/or mean, either excludes NaNs during aggregation.
`hdthreshold`	Numeric. A numeric value representing the threshold for habitat availability (e.g., cropland density or host density) used in the sensitivity analysis.

`linkthreshold`	Numeric. A numeric value representing the threshold for the link weights used to calculate habitat connectivity of each location. Note that link weights indicate the relative likelihood of a species moving between locations (nodes) and correspond to the entries in the adjacency matrix.

Value

The GeoModel object with updated properties.

`setprops,GeoModel,character,numeric,numeric-method`

Set properties of the GeoModel object.

Description

Set properties of the GeoModel object.

Usage

```
## S4 method for signature 'GeoModel,character,numeric,numeric'
setprops(x, aggregation, hdthreshold, linkthreshold)
```

Arguments

`x`	The GeoModel object.
`aggregation`	Character. A value representing the spatial aggregation method used for aggregating the habitat availability map before conducting the sensitivity analysis. There are two aggregation methods available in geohabnet: sum and/or mean, either excludes NaNs during aggregation.
`hdthreshold`	Numeric. A numeric value representing the threshold for habitat availability (e.g., cropland density or host density) used in the sensitivity analysis.
`linkthreshold`	Numeric. A numeric value representing the threshold for the link weights used to calculate habitat connectivity of each location. Note that link weights indicate the relative likelihood of a species moving between locations (nodes) and correspond to the entries in the adjacency matrix.

Value

The GeoModel object with updated properties.

`set_global_scales` *Set global geographical extent*

Description

This function sets the geographical extents used in global analysis. See also `geoscale_param()` to set the geographic extent of an analysis that is not global. Each geographic extent should be in the form of c(Xmin, Xmax, Ymin, Ymax). Geographic extent must be specified by four values in degrees that represent the geographic limits of the area for analysis, following the order: minimum longitude, maximum longitude, minimum latitude, and maximum latitude. Degrees are in decimal notation and have a negative sign for the southern and western hemispheres.

Usage

```
set_global_scales(value)
```

Arguments

`value`	list. Named list of eastern and western hemisphere extents. See usage.

Value

List. Named list with scales for eastern and western hemispheres

See Also

`global_scales()` `terra::ext()`

Examples

```
set_global_scales(list(east = c(-24, 180, -58, 60), west = c(-140, -34, -58, 60)))
```

`set_parameters` *Set Parameters*

Description

This function allows the user to set the parameters by replacing the existing parameters file with a new one. Use `get_parameters()` to modify the parameter values. After running this function, users can use the modified parameters in `sensitivity_analysis()`.

Usage

```
set_parameters(new_params, iwindow = FALSE)
```

Arguments

`new_params`	The path to the new parameters file.
`iwindow`	Logical, indicating whether to prompt the user to select the new parameters file using a file selection window. The default value of this parameter is set to FALSE.

Value

None

Examples

```
param_fp <- get_parameters()
set_parameters(param_fp)
```

`supported_metrics` *Returns metrics currently supported in the analysis.*

Description

Returns metrics currently supported in the analysis.

Usage

```
supported_metrics()
```

Value

vector of supported metrics.

See Also

Other metrics: `nn_sum()`

Examples

```
supported_metrics()
```

Index