

Package ‘heterogen’

July 22, 2025

Type Package

Title Spatial Functions for Heterogeneity and Climate Variability

Version 1.2.33

Date 2023-08-13

Description A comprehensive suite of spatial functions created to analyze and assess data heterogeneity and climate variability in spatial datasets. This package is specifically designed to address the challenges associated with characterizing and understanding complex spatial patterns in environmental and climate-related data.

Maintainer P.Joser Atauchí <patauchi@gmail.com>

URL <https://github.com/patauchi/heterogen>

BugReports <https://github.com/patauchi/heterogen/issues>

NeedsCompilation yes

Depends terra

Imports methods, rio, scales, future, parallel

LinkingTo Rcpp, RcppArmadillo, RcppEigen

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.2.3

Author P.Joser Atauchí [aut, cre],
A. Townsend Peterson [ctb]

Repository CRAN

Date/Publication 2023-08-17 07:42:33 UTC

Contents

bg_transpose	2
distance_weighted_gauss	2
float_relative	3

float_round	3
gw pca_core	4
gw pca_df_mc	4
heterog	6
matrixcec_square	8
matrixmult	8
matrixvec_plus	9
matrixvec_subs	9
plot,SpatHetero,ANY-method	10
SpatHetero-class	10

Index	11
--------------	-----------

bg_transpose	<i>bg_transpose</i>
--------------	---------------------

Description

Transpose of a matrix based on row or column index.

Usage

```
bg_transpose(mat, byrow = FALSE)
```

Arguments

mat	A Matrix.
byrow	FALSE computes based on row index. TRUE computes based on column index.

Value

A matrix transposed.

distance_weighted_gauss	<i>distance_weighted_gauss</i>
-------------------------	--------------------------------

Description

Weighted Distance based on Gaussian function

Usage

```
distance_weighted_gauss(coord_xy, point_xy, tau)
```

Arguments

coord_xy	A Matrix with lon/lat coordinates.
point_xy	lon/lat coordinate.
tau	bandwidth.

Value

A vector.

float_relative	<i>float_relative</i>
----------------	-----------------------

Description

Relative sum formula

Usage

`float_relative(xx)`

Arguments

xx	A Matrix with lon/lat coordinates.
----	------------------------------------

Value

A vector.

float_round	<i>float_round</i>
-------------	--------------------

Description

Rouding of Numbers

Usage

`float_round(float_n, digits = 0L)`

Arguments

float_n	A numeric vector.
digits	integer indicating the number of decimal places .

Value

A vector.

 gw pca_core

Core Function of GWPCA

Description

The gw pca_core function is a core implementation of Generalized Weighted Principal Component Analysis for each iteration.

Usage

```
gw pca_core(xy, p_xy, env, env_trans, tau)
```

Arguments

xy	A matrix containing the coordinates of the points where environmental measurements were taken. The matrix should have two columns, representing the X and Y coordinates.
p_xy	A matrix containing the coordinates of the point GWPCA will be estimated. It should have two columns for X and Y coordinates.
env	A data matrix representing the environmental variables. Rows represent observations (points or grid cells), and columns represent environmental variables.
env_trans	Transpose of env matrix.
tau	The bandwidth parameter for spatial weighting in GWPCA. It determines the extent of spatial influence on the estimation of principal components.

Value

A vector of eigenvalues from local PCA

 gw pca_df_mc

Perform GWPCA from data.frame with spatial structure.

Description

gw pca_df is an R function that performs Generalized Weighted Principal Component Analysis (GWPCA) on a given dataset. This function allow to calculate the environmental heterogeneity from data.frame with spatial structure.

Usage

```
gwPCA_df_mc(
  data_df,
  bandwidth = 0.2,
  tolerance = 5,
  nprocess = 10000,
  parallel = FALSE,
  ncores = 2,
  normalized = FALSE,
  method = "iter",
  dirDS = "rds"
)
```

Arguments

<code>data_df</code>	The input data matrix for which GWPCA needs to be performed. It should contain numerical values only. Rows represent cells, and columns represent bioclimatic variables.
<code>bandwidth</code>	The bandwidth for the spatial weighting function.
<code>tolerance</code>	The tolerance for spatial weight computation.
<code>nprocess</code>	(Optional) The number of iterations for calculating the principal components. Default is set to 1000.
<code>parallel</code>	(Optional) A logical value indicating whether to run the computation in parallel. If TRUE, multiple cores will be used for processing. Default is FALSE.
<code>ncores</code>	(Optional) The number of cores to be used for parallel computation. Only applicable if <code>parallel</code> is set to TRUE. Default is 4.
<code>normalized</code>	(Optional) A logical value indicating whether the input data should be normalized before performing GWPCA. Default is FALSE, meaning the data will not be normalized. Take in account that core function performs correlation analysis in order to normalize the input variables.
<code>method</code>	The method used for GWPCA computation. It can take one of the following values. <code>local</code> Performs GWPCA locally and will save each iteration on <code>.rds</code> files. Recommended for large-scale data sets. <code>inter</code> Uses RAM memory to . Default is <code>inter</code> .
<code>dirDS</code>	(Optional) The directory where the results will be saved in RDS format. Default is <code>rds</code> .

Value

A matrix of eigenvalues

Examples

```
path_csv <- system.file("extdata", "south.csv", package="heterogen")
south_csv <- rio::import(path_csv)
```

```
# notice: south_csv object contains x,y (lot/lat coordinates)
# and environmental variables
north_het <- gwPCA_df_mc(as.matrix(south_csv), parallel = TRUE,
ncores = 2, bandwidth = 0.1, tolerance = 10)
```

heterog

Heterogeneity (rasters)

Description

The heterog function is designed to calculate environmental heterogeneity metric from a raster stack dataset. This function aids in assessing the spatial variation and diversity of environmental variables within the raster data, providing valuable insights into the heterogeneity of the study area.

Usage

```
heterog(
  datastack,
  bandwidth = 0.3,
  tolerance = 5,
  nprocess = 1000,
  parallel = FALSE,
  ncores = 2,
  normalized = FALSE,
  method = "iter",
  dirds = "rds"
)
```

Arguments

datastack	SpatRaster class. The input raster stack representing environmental variables. Each layer in the stack corresponds to a different environmental variable, and the function calculates heterogeneity based on the variability across these layers.
bandwidth	The bandwidth for the spatial weighting function.
tolerance	The tolerance for spatial weight computation.
nprocess	(Optional) The number of iterations for calculating the principal components. Default is set to 1000.
parallel	(Optional) A logical value indicating whether to run the computation in parallel. If TRUE, multiple cores will be used for processing. Default is FALSE.
ncores	(Optional) The number of cores to be used for parallel computation. Only applicable if parallel is set to TRUE. Default is 4.

normalized	(Optional) A logical value indicating whether the input data should be normalized before performing GWPCA. Default is FALSE, meaning the data will not be normalized. Take in account that core function performs correlation analysis in order to normalize the input variables.
method	The method used for GWPCA computation. It can take one of the following values. <code>local</code> Performs GWPCA locally and will save each iteration on <code>.rds</code> files. Recommended for large-scale data sets. <code>inter</code> Uses RAM memory to . Default is <code>inter</code> .
dir rds	(Optional) The directory where the results will be saved in RDS format. Default is <code>rds</code> .

Value

A `SpatHetero` object

- `hetero` A heterogeneity layer
- `matrix` A Matrix of eigenvalues
- `rasters` A complete set of heterogeneity layers for each component

Examples

```
# Case 01: South
path <- system.file("extdata","south", package="heterogen")
south_rast <- terra::rast(list.files(path, full.names = TRUE,
pattern = '.tif'))

south_het <- heterog(south_rast, parallel = TRUE,
bandwidth = 0.1, tolerance = 10)
plot(south_het)

# Case 02: North
path <- system.file("extdata","north", package="heterogen")
north_rast <- terra::rast(list.files(path, full.names = TRUE,
pattern = '.tif'))

north_het <- heterog(north_rast, parallel = TRUE,
bandwidth = 0.1, tolerance = 10)
plot(north_het)
```

matrixcec_square *matrixcec_square*

Description

Matrix Square

Usage

```
matrixcec_square(X, y)
```

Arguments

X	A Matrix.
y	A Vector

Value

A matrix.

matrixmult *matrixmult*

Description

Matrix Multiplication

Usage

```
matrixmult(A, B)
```

Arguments

A	A Matrix.
B	A Matrix

Value

A matrix.

matrixvec_plus *matrixvec_plus*

Description

Matrix Multiplication

Usage

`matrixvec_plus(X, y)`

Arguments

X A Matrix.
y A Vector

Value

A matrix.

matrixvec_subs *matrixvec_subs*

Description

Matrix Substraction

Usage

`matrixvec_subs(X, y)`

Arguments

X A Matrix.
y A Vector

Value

A matrix.

plot, SpatHetero, ANY-method

Plot Heterogeneity Layer

Description

Plot

Usage

```
## S4 method for signature 'SpatHetero,ANY'
plot(x, comp = NULL, ...)
```

Arguments

x	SpatHetero Class
comp	integer. Plot specific component of the heterogeneity.
...	Plot parameters forwarded.

Value

No return value, called for side effects.

SpatHetero-class

SpatHetero

Description

SpatHetero

Slots

hetero A Heterogeneity Layer
matrix SpatHetero_in data
rasters A SpatRaster for Each Component

Index

[bg_transpose](#), 2

[distance_weighted_gauss](#), 2

[float_relative](#), 3

[float_round](#), 3

[gwpca_core](#), 4

[gwpca_df_mc](#), 4

[heterog](#), 6

[matrixcec_square](#), 8

[matrixmult](#), 8

[matrixvec_plus](#), 9

[matrixvec_subs](#), 9

[plot](#), [SpatHetero](#), ANY-method, 10

[SpatHetero](#)-class, 10