Package: HDSpatialScan (via r-universe)

October 8, 2026

Type Package

Title Multivariate and Functional Spatial Scan Statistics

Version 1.0.5

Description Allows to detect spatial clusters of abnormal values on multivariate or functional data (Frévent et al. (2022) <doi:10.32614/RJ-2022-045>). See also: Frévent et al. (2023) <doi:10.1093/jrsssc/qlad017>, Smida et al. (2022) <doi:10.1016/j.csda.2021.107378>, Frévent et al. (2021) <doi:10.1016/j.spasta.2021.100550>. Cucala et al. (2019) <doi:10.1016/j.spasta.2018.10.002>, Cucala et al. (2017) <doi:10.1016/j.spasta.2017.06.001>, Jung and Cho (2015) <doi:10.1186/s12942-015-0024-6>, Kulldorff et al. (2009) <doi:10.1186/1476-072X-8-58>.

Depends R (>= 3.5.0)

License GPL-3

Encoding UTF-8

LazyData true

Imports Rcpp, parallel, pbapply, purrr, matrixStats, SpatialNP, sp, sf, DT, TeachingDemos, plotrix, fmsb, swfscMisc, raster, graphics, stats, grDevices, methods

LinkingTo RcppArmadillo, Rcpp

Collate 'documentation_page.R' 'clusters.R' 'permutations.R' 'gaussian_multi.R' 'wmw_multi.R' 'wmw_fonc.R' 'scan_fonc.R' 'scan_multi.R' 'plot_map.R' 'RcppExports.R' 'dfree_index_multi.R' 'pointwise_dfree.R' 'pointwise_wmw_multi.R' 'pointwise_wmw_uni.R' 'transform_data.R' 'data_fmulti_data.R' 'data_funi_data.R' 'data_multi_data.R' 'data_map_sites.R' 'posteriori_filtering.R' 'non_overlap.R' 'scan_uni.R' 'wmw_uni.R' 'dfree.R' 'finalizing_scan_procedure.R' 'HDSpatialScan_plot.R' 'HDSpatialScan_Classes.R' 'HDSpatialScan_plotCurves.R' 'HDSpatialScan_plotSummary.R' 'HDSpatialScan_print.R' 'HDSpatialScan_summary.R' 'InitScan.R' 'SpatialScan.R'

RoxygenNote 7.3.1

Suggests knitr, rmarkdown

NeedsCompilation yes

Author Camille FREVENT [aut, cre, cph], Mohamed-Salem AHMED [aut], Julien SOULA [aut], Zaineb SMIDA [aut], Lionel CUCALA [aut], Sophie DABO-NIANG [aut], Michaël GENIN [aut]

Maintainer Camille FREVENT <camille.frevent@univ-lille.fr>

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

Date/Publication 2025-06-05 16:10:10 UTC

RemoteUrl https://github.com/cran/HDSpatialScan

RemoteRef HEAD

RemoteSha 7adf9363bc9e2fffcb36b9aa1c8e5f5a204a618c

Contents

HDSpatialScan-package . . . 3
clusters . . . 4
DFFSS . . . 5
dfree . . . 6
dfree_index_multi . . . 7
FinScan . . . 7
fmulti_data . . . 9
funi_data . . . 9
InitScan . . . 10
map_sites . . . 11
MDFFSS . . . 12
MG . . . 13
MNP . . . 14
MPFSS . . . 15
MRBFSS . . . 16
multi_data . . . 17
multi_fWMW . . . 17
multi_gaussian . . . 18
multi_signs_matrix . . . 18
multi_WMW . . . 19
non_overlap . . . 19
NPFSS . . . 20
permutate . . . 21
PFSS . . . 21
plot.ResScanOutput . . . 22
plot_map . . . 23
plot_map2 . . . 24
plot_schema . . . 25
plotCurves . . . 25
plotCurves.ResScanOutputMultiFunct . . . 26

plotCurves.ResScanOutputUniFunct . . . 27
plotSummary . . . 29
plotSummary.ResScanOutputMulti . . . 30
plotSummary.ResScanOutputMultiFunct . . . 31
plotSummary.ResScanOutputUniFunct . . . 32
pointwise_dfree . . . 33
pointwise_wmw_multi . . . 33
pointwise_wmw_uni . . . 34
post_filt_area . . . 34
post_filt_nb_sites . . . 35
post_filt_radius . . . 36
print.ResScanOutput . . . 36
ResScanOutput . . . 37
ResScanOutputMulti . . . 38
ResScanOutputMultiFunct . . . 39
ResScanOutputUni . . . 41
ResScanOutputUniFunct . . . 42
SpatialScan . . . 43
summary.ResScanOutputMulti . . . 47
summary.ResScanOutputMultiFunct . . . 48
summary.ResScanOutputUni . . . 49
summary.ResScanOutputUniFunct . . . 50
transform_data . . . 52
UG . . . 52
uni_fWMW . . . 53
uni_signs_matrix . . . 53
UNP . . . 54
URBFSS . . . 55
wmw_uni . . . 56

Index **57**

`HDSpatialScan-package` *Multivariate and Functional Spatial Scan Statistics*

Description

Allows to detect spatial clusters of abnormal values on multivariate or functional data.

Details

Package:	HDSpatialScan
Type:	Package
Version:	1.0.5
Date:	2025-06-05
License:	GPL-3
LazyLoad:	yes

Author(s)

FREVENT Camille, AHMED Mohamed-Salem, SOULA Julien, SMIDA Zaineb, CUCALA Lionel, DABO-NIANG Sophie and GENIN Michaël. Maintainer: FREVENT Camille <camille.frevent@univ-lille.fr>

References

Martin Kulldorff and Lan Huang and Kevin Konty (2009). A Scan Statistic for Continuous Data Based on the Normal Probability Model. International Journal of Health Geographics, 8 (58).

Inkyung Jung and Ho Jin Cho (2015). A Nonparametric Spatial Scan Statistic for Continuous Data. International Journal of Health Geographics, 14.

Lionel Cucala and Michaël Genin and Caroline Lanier and Florent Occelli (2017). A Multivariate Gaussian Scan Statistic for Spatial Data. Spatial Statistics, 21, 66-74.

Lionel Cucala and Michaël Genin and Florent Occelli and Julien Soula (2019). A Multivariate Nonparametric Scan Statistic for Spatial Data. Spatial statistics, 29, 1-14.

Camille Frévent and Mohamed-Salem Ahmed and Matthieu Marbac and Michaël Genin (2021). Detecting Spatial Clusters in Functional Data: New Scan Statistic Approaches. Spatial Statistics, 46.

Zaineb Smida and Lionel Cucala and Ali Gannoun and Ghislain Durif (2022). A Wilcoxon-Mann-Whitney spatial scan statistic for functional data. Computational Statistics & Data Analysis, 167.

Camille Frévent and Mohamed-Salem Ahmed and Sophie Dabo-Niang and Michaël Genin (2023). Investigating Spatial Scan Statistics for Multivariate Functional Data. Journal of the Royal Statistical Society Series C: Applied Statistics, 72(2), 450-475.

`clusters` *Creation of the matrix of potential clusters*

Description

This function creates the matrix in which each column corresponds to a potential clusters, taking the value 1 when a site (or an individual) is in the potential cluster and 0 otherwise.

Usage

```
clusters(sites_coord, system, mini, maxi, type_minimaxi, sites_areas)
```

Arguments

sites_coord	numeric matrix. Matrix of the coordinates of the sites (or the individuals, in that case there can be many individuals with the same coordinates). It has the same number of rows as the number of sites or individuals and 2 columns.
system	character. System in which the coordinates are expressed: "Euclidean" or "WGS84".
mini	numeric. Minimum for the clusters (see type_minimaxi).
maxi	numeric. Maximum for the clusters (see type_minimaxi).
type_minimaxi	character. Type of minimum and maximum: "area": the minimum and maximum area of the clusters, "radius": the minimum and maximum radius, or "sites/indiv": the minimum and maximum number of sites or individuals in the clusters.
sites_areas	numeric vector. Areas of the sites. It must contain the same number of elements than the rows of sites_coord. If the data is on individuals and not on sites, there can be duplicated values. By default: NULL

Value

The list of the following elements:

- matrix_clusters: numeric matrix of 0 and 1
- centres: the coordinates of the centres of each cluster (numeric matrix)
- radius: the radius of the clusters in km if system = "WGS84" or in the coordinates unit otherwise (numeric vector)
- areas: the areas of the clusters (in same units as in sites_areas). Provided only if sites_areas is not NULL. Numeric vector
- system: the system of coordinates (character)

DFFSS *DFFSS scan procedure*

Description

This function computes the DFFSS (Distribution-Free Functional scan statistic).

Usage

```
DFFSS(
  data,
  MC = 999,
  typeI = 0.05,
  nbCPU = 1,
  times = NULL,
  initialization,
  permutations
)
```

Arguments

data	matrix. Matrix of the data, the rows correspond to the sites (or to the individuals if the observations are by individuals and not by sites) and each column represents an observation time. The times must be the same for each site/individual.
MC	numeric. Number of Monte-Carlo permutations to evaluate the statistical significance of the clusters. By default: 999.
typeI	numeric. The desired type I error. A cluster will be evaluated as significant if its associated p-value is less than typeI. By default 0.05.
nbCPU	numeric. Number of CPU. If nbCPU > 1 parallelization is done. By default: 1.
times	numeric. Times of observation of the data. By default NULL.
initialization	list. Initialization for the scan procedure (see InitScan for more details).
permutations	matrix. Indices of permutations of the data.

Value

An object of class ResScanOutputUniFunct.

References

Camille Frévent and Mohamed-Salem Ahmed and Matthieu Marbac and Michaël Genin (2021). Detecting Spatial Clusters in Functional Data: New Scan Statistic Approaches. Spatial Statistics, 46.

---

dfree *Index for the UG scan procedure*

Description

This function returns the index we want to maximize on the set of potential clusters, for each potential cluster and each permutation

Usage

```
dfree(data, matrix_clusters)
```

Arguments

data	numeric matrix. Matrix of the data. The rows correspond to the sites (or the individuals) and each column represents a permutation.
matrix_clusters	numeric matrix. Matrix in which each column represents a potential cluster. It is the result of the "clusters" function.

Value

numeric matrix.

dfree_index_multi *Index for the MDFFSS scan procedure*

Description

This function returns the index we want to maximize on the set of potential clusters, for each potential cluster

Usage

```
dfree_index_multi(data, matrix_clusters)
```

Arguments

data	List. List of the data, each element of the list corresponds to a site (or an individual), for each element each row corresponds to a variable and each column represents an observation time.
matrix_clusters	numeric matrix. Matrix in which each column represents a potential cluster. It is the result of the "clusters" function.

Value

numeric vector.

FinScan *Finalization of the scan procedures*

Description

This function finalizes the scan procedures.

Usage

```
FinScan(
  index_clusters_temp,
  index,
  filtering_post,
  type_minimaxi_post,
  mini_post,
  maxi_post,
  nb_sites,
  matrix_clusters,
  radius,
  areas,
```

```
  centres,
  pvals,
  maximize = TRUE
)
```

Arguments

index_clusters_temp	numeric vector. Indices of the significant clusters.
index	numeric vector. Index of concentration for each potential cluster.
filtering_post	logical. Is there an a posteriori filtering?
type_minimaxi_post	character. Type of minimum and maximum a posteriori: by default "sites/indiv": the mini_post and maxi_post are on the number of sites or individuals in the significant clusters. Other possible values are "area": the minimum and maximum area of the clusters, or "radius": the minimum and maximum radius.
mini_post	numeric. A minimum to filter the significant clusters a posteriori (see type_minimaxi_post). The default NULL is for no filtering with a a posteriori minimum.
maxi_post	numeric. A maximum to filter the significant clusters a posteriori (see type_minimaxi_post). The default NULL is for no filtering with a a posteriori maximum.
nb_sites	numeric. The number of considered sites or individuals.
matrix_clusters	matrix. The matrix of potential clusters taking the value 1 at lign i and column j if the cluster j contains the site i, 0 otherwise.
radius	numeric vector. The radius of the potential clusters.
areas	numeric vector. The areas of the potential clusters.
centres	numeric matrix. The coordinates of the centres of each potential cluster.
pvals	numeric vector. The pvalue of each potential cluster.
maximize	logical. Should the index be maximized? By default TRUE. If FALSE it will be minimized.

Value

The list of the following elements:

- pval_clusters: pvalues of the selected clusters.
- sites_clusters: the indices of the sites of the selected clusters.
- centres_clusters: the coordinates of the centres of each selected cluster.
- radius_clusters: the radius of the selected clusters.
- areas_clusters: the areas of the selected clusters.

fmulti_data *Multivariate functional data*

Description

Concentrations over the time of NO2, O3, PM10 and PM2.5 from 2020/05/01 to 2020/06/25 in each canton (administrative subdivision) of Nord-Pas-de-Calais (a region from France).

Usage

```
data("fmulti_data")
```

Format

A list of 169 elements. Each element corresponds to a canton and is a matrix of 56 columns (for the 56 days of observation) and 4 rows (4 variables, in the order NO2, O3, PM10 and PM2.5).

References

Data from the National Air Quality Forecasting Platform www.prevair.org

funi_data *Univariate functional data*

Description

Concentration over the time of the pollutant NO2 from 2020/05/01 to 2020/06/25 in each canton (administrative subdivision) of Nord-Pas-de-Calais (a region from France).

Usage

```
data("funi_data")
```

Format

A matrix of 169 rows and 56 columns. Each row corresponds to a canton, and each column is an observation time (a day). The 56 observation times are thus equally spaced times.

References

Data from the National Air Quality Forecasting Platform www.prevair.org

`InitScan` *Initalization of the scan procedures by creating the matrix of potential clusters*

Description

This function initializes the scan procedures by creating the matrix of potential clusters.

Usage

```
InitScan(
  mini_post,
  maxi_post,
  type_minimaxi_post,
  sites_areas,
  sites_coord,
  system,
  mini,
  maxi,
  type_minimaxi
)
```

Arguments

`mini_post`	numeric. A minimum to filter the significant clusters a posteriori (see type_minimaxi_post). The default NULL is for no filtering with a a posteriori minimum.
`maxi_post`	numeric. A maximum to filter the significant clusters a posteriori (see type_minimaxi_post). The default NULL is for no filtering with a a posteriori maximum.
`type_minimaxi_post`	character. Type of minimum and maximum a posteriori: by default "sites/indiv": the mini_post and maxi_post are on the number of sites or individuals in the significant clusters. Other possible values are "area": the minimum and maximum area of the clusters, or "radius": the minimum and maximum radius.
`sites_areas`	numeric vector. Areas of the sites. It must contain the same number of elements than the rows of sites_coord. If the data is on individuals and not on sites, there can be duplicated values. By default: NULL
`sites_coord`	numeric matrix. Coordinates of the sites (or the individuals, in that case there can be many individuals with the same coordinates).
`system`	character. System in which the coordinates are expressed: "Euclidean" or "WGS84".
`mini`	integer. A minimum for the clusters (see type_minimaxi). Changing the default value may bias the inference.
`maxi`	integer. A Maximum for the clusters (see type_minimaxi). Changing the default value may bias the inference.

`type_minimaxi` character. Type of minimum and maximum: by default "sites/indiv": the mini and maxi are on the number of sites or individuals in the potential clusters. Other possible values are "area": the minimum and maximum area of the clusters, or "radius": the minimum and maximum radius.

Value

The list of the following elements:

- filtering_post: logical, is there an a posteriori filtering?
- matrix_clusters: the matrix of potential clusters
- centres: the coordinates of the centres of each potential cluster
- radius: the radius of the potential clusters in km if system = WGS84 or in the user units
- areas: the areas of the potential clusters (in the same units as sites_areas).
- sites_coord: coordinates of the sites
- system: system in which the coordinates are expressed
- mini_post: a minimum to filter the significant clusters a posteriori
- maxi_post: a maximum to filter the significant clusters a posteriori
- type_minimaxi_post: type of minimum and maximum a posteriori

---

`map_sites` *Spatial object corresponding to the sites of the data of the package HDSpatialScan*

Description

Spatial object corresponding to the sites (169 cantons) of the data of the package HDSpatialScan.

Usage

```
data("map_sites")
```

Format

A SpatialPolygonsDataFrame.

MDFFSS *MDFFSS scan procedure*

Description

This function computes the MDFFSS (Multivariate Distribution-Free Functional scan statistic).

Usage

```
MDFFSS(
  data,
  MC = 999,
  typeI = 0.05,
  nbCPU = 1,
  variable_names = NULL,
  times = NULL,
  initialization,
  permutations
)
```

Arguments

data	list of numeric matrices. List of nb_sites (or nb_individuals if the observations are by individuals and not by sites) matrices of the data, the rows correspond to the variables and each column represents an observation time. The times must be the same for each site/individual.
MC	numeric. Number of Monte-Carlo permutations to evaluate the statistical significance of the clusters. By default: 999.
typeI	numeric. The desired type I error. A cluster will be evaluated as significant if its associated p-value is less than typeI. By default 0.05.
nbCPU	numeric. Number of CPU. If nbCPU > 1 parallelization is done. By default: 1.
variable_names	character. Names of the variables. By default NULL.
times	numeric. Times of observation of the data. By default NULL.
initialization	list. Initialization for the scan procedure (see InitScan for more details).
permutations	matrix. Indices of permutations of the data.

Value

An object of class ResScanOutputMultiFunct.

References

Camille Frévent and Mohamed-Salem Ahmed and Sophie Dabo-Niang and Michaël Genin (2023). Investigating Spatial Scan Statistics for Multivariate Functional Data. Journal of the Royal Statistical Society Series C: Applied Statistics, 72(2), 450-475.

MG *MG scan procedure*

Description

This function computes the MG (Multivariate Gaussian scan statistic).

Usage

```
MG(
  data,
  MC = 999,
  typeI = 0.05,
  nbCPU = 1,
  variable_names = NULL,
  initialization,
  permutations
)
```

Arguments

data	matrix. Matrix of the data, the rows correspond to the sites (or the individuals if the observations are by individuals and not by sites) and each column represents a variable.
MC	numeric. Number of Monte-Carlo permutations to evaluate the statistical significance of the clusters. By default: 999.
typeI	numeric. The desired type I error. A cluster will be evaluated as significant if its associated p-value is less than typeI. By default 0.05.
nbCPU	numeric. Number of CPU. If nbCPU > 1 parallelization is done. By default: 1.
variable_names	character. Names of the variables. By default NULL.
initialization	list. Initialization for the scan procedure (see InitScan for more details).
permutations	matrix. Indices of permutations of the data.

Value

An object of class ResScanOutputMulti.

References

Lionel Cucala and Michaël Genin and Caroline Lanier and Florent Occelli (2017). A Multivariate Gaussian Scan Statistic for Spatial Data. Spatial Statistics, 21, 66-74.

MNP *MNP scan procedure*

Description

This function computes the MNP (Multivariate Nonparametric scan statistic).

Usage

```
MNP(
  data,
  MC = 999,
  typeI = 0.05,
  nbCPU = 1,
  variable_names = NULL,
  initialization,
  permutations
)
```

Arguments

data	matrix. Matrix of the data, the rows correspond to the sites (or the individuals if the observations are by individuals and not by sites) and each column represents a variable.
MC	numeric. Number of Monte-Carlo permutations to evaluate the statistical significance of the clusters. By default: 999.
typeI	numeric. The desired type I error. A cluster will be evaluated as significant if its associated p-value is less than typeI. By default 0.05.
nbCPU	numeric. Number of CPU. If nbCPU > 1 parallelization is done. By default: 1.
variable_names	character. Names of the variables. By default NULL
initialization	list. Initialization for the scan procedure (see InitScan for more details).
permutations	matrix. Indices of permutations of the data.

Value

An object of class ResScanOutputMulti.

References

Lionel Cucala and Michaël Genin and Florent Occelli and Julien Soula (2019). A Multivariate Nonparametric Scan Statistic for Spatial Data. Spatial statistics, 29, 1-14.

MPFSS *MPFSS scan procedure*

Description

This function computes the MPFSS (Parametric Multivariate Functional scan statistic).

Usage

```
MPFSS(
  data,
  MC = 999,
  typeI = 0.05,
  method = c("LH", "W", "P", "R"),
  nbCPU = 1,
  variable_names = NULL,
  times = NULL,
  initialization,
  permutations
)
```

Arguments

data	list of numeric matrices. List of nb_sites (or nb_individuals if the observations are by individuals and not by sites) matrices of the data, the rows correspond to the variables and each column represents an observation time. The times must be equally spaced and the same for each site/individual.
MC	numeric. Number of Monte-Carlo permutations to evaluate the statistical significance of the clusters. By default: 999.
typeI	numeric. The desired type I error. A cluster will be evaluated as significant if its associated p-value is less than typeI. By default 0.05.
method	character vector. The methods to compute the significant clusters. Options: "LH", "W", "P", "R" for respectively the Lawley-Hotelling trace test statistic, The Wilks lambda test statistic, the Pillai trace test statistic and the Roy's maximum root test statistic. By default all are computed.
nbCPU	numeric. Number of CPU. If nbCPU > 1 parallelization is done. By default: 1.
variable_names	character. Names of the variables. By default NULL.
times	numeric. Times of observation of the data. By default NULL.
initialization	list. Initialization for the scan procedure (see InitScan for more details).
permutations	matrix. Indices of permutations of the data.

Value

List of objects of class ResScanOutputMultiFunct (one element by method)

References

Camille Frévent and Mohamed-Salem Ahmed and Sophie Dabo-Niang and Michaël Genin (2023). Investigating Spatial Scan Statistics for Multivariate Functional Data. Journal of the Royal Statistical Society Series C: Applied Statistics, 72(2), 450-475.

`MRBFSS` *MRBFSS scan procedure*

Description

This function computes the MRBFSS (Multivariate Rank-Based Functional scan statistic).

Usage

```
MRBFSS(
  data,
  MC = 999,
  typeI = 0.05,
  nbCPU = 1,
  variable_names = NULL,
  times = NULL,
  initialization,
  permutations
)
```

Arguments

`data`	list of numeric matrices. List of nb_sites (or nb_individuals if the observations are by individuals and not by sites) matrices of the data, the rows correspond to the variables and each column represents an observation time. The times must be the same for each site/individual.
`MC`	numeric. Number of Monte-Carlo permutations to evaluate the statistical significance of the clusters. By default: 999.
`typeI`	numeric. The desired type I error. A cluster will be evaluated as significant if its associated p-value is less than typeI. By default 0.05.
`nbCPU`	numeric. Number of CPU. If nbCPU > 1 parallelization is done. By default: 1.
`variable_names`	character. Names of the variables. By default NULL.
`times`	numeric. Times of observation of the data. By default NULL.
`initialization`	list. Initialization for the scan procedure (see `InitScan` for more details).
`permutations`	matrix. Indices of permutations of the data.

Value

An object of class ResScanOutputMultiFunct

References

Camille Frévent and Mohamed-Salem Ahmed and Sophie Dabo-Niang and Michaël Genin (2023). Investigating Spatial Scan Statistics for Multivariate Functional Data. Journal of the Royal Statistical Society Series C: Applied Statistics, 72(2), 450-475.

`multi_data` *Multivariate non-functional data*

Description

Average concentrations over the time of NO2, O3, PM10 and PM2.5 from 2020/05/01 to 2020/06/25 in each canton (administrative subdivision) of Nord-Pas-de-Calais (a region from France).

Usage

```
data("multi_data")
```

Format

A matrix of 169 rows and 4 columns. Each row corresponds to a canton, and each column is a concentration mean in the order NO2, O3, PM10 and PM2.5.

References

Data from the National Air Quality Forecasting Platform www.prevair.org

---

`multi_fWMW` *Index for the NPFSS scan procedure (multivariate functional case)*

Description

This function returns the index we want to maximize on the set of potential clusters, for each potential cluster

Usage

```
multi_fWMW(signs, matrix_clusters)
```

Arguments

`signs`	list of numeric matrices. List of nb_sites (or nb_individuals) sign matrices, the rows correspond to the variables and each column represents an observation time.
`matrix_clusters`	numeric matrix. Matrix in which each column represents a potential cluster. It is the result of the "clusters" function.

Value

numeric vector.

`multi_gaussian` *Index for the MG scan procedure*

Description

This function returns the index we want to minimize on the set of potential clusters, for each potential cluster

Usage

```
multi_gaussian(data, matrix_clusters)
```

Arguments

`data`	numeric matrix. Matrix of the data, the rows correspond to the sites (or individuals) and each column represents a variable.
`matrix_clusters`	numeric matrix. Matrix in which each column represents a potential cluster. It is the result of the "clusters" function.

Value

numeric vector.

`multi_signs_matrix` *List of matrix of signs (multivariate functional data)*

Description

This function returns the list of matrix of signs for the multivariate functional data

Usage

```
multi_signs_matrix(data)
```

Arguments

`data`	list of numeric matrices. List of nb_sites (or nb_individuals) matrices of the data, the rows correspond to the variables and each column represents an observation time.

Value

list of numeric matrices.

multi_WMW *Index for the MNP scan procedure*

Description

This function returns the index we want to maximize on the set of potential clusters, for each potential cluster

Usage

```
multi_WMW(rank_data, matrix_clusters)
```

Arguments

rank_data	numeric matrix. Matrix of the ranks of the initial data, the rows correspond to the sites (or the individuals) and each column represents a variable.
matrix_clusters	numeric matrix. Matrix in which each column represents a potential cluster. It is the result of the "clusters" function.

Value

numeric vector.

non_overlap *Return only the detected clusters with no overlapping in their order of detection*

Description

This function allows to return only the detected clusters with no overlapping in their order of detection.

Usage

```
non_overlap(index_clusters, matrix_clusters)
```

Arguments

index_clusters	numeric vector. The indices of the detected clusters.
matrix_clusters	numeric matrix. Matrix in which each column represents a potential cluster. A value of 1 indicate that the site (or the individual) is in the cluster, 0 otherwise.

Value

The detecting clusters with no overlapping, in their order of detection.

NPFSS *NPFSS scan procedure (univariate functional or multivariate functional)*

Description

This function computes the NPFSS (Nonparametric Functional scan statistic for multivariate or univariate functional data).

Usage

```
NPFSS(
  data,
  MC = 999,
  typeI = 0.05,
  nbCPU = 1,
  variable_names = NULL,
  times = NULL,
  initialization,
  permutations
)
```

Arguments

data	list of numeric matrices or a matrix. List of nb_sites (or nb_individuals if the observations are by individuals and not by site) matrices of the data, the rows correspond to the variables and each column represents an observation time (multivariate case) ; or Matrix of the data, the rows correspond to the sites (or to the individuals) and each column represents an observation time (univariate case). The times must be equally spaced and the same for each site/individual.
MC	numeric. Number of Monte-Carlo permutations to evaluate the statistical significance of the clusters. By default: 999.
typeI	numeric. The desired type I error. A cluster will be evaluated as significant if its associated p-value is less than typeI. By default 0.05.
nbCPU	numeric. Number of CPU. If nbCPU > 1 parallelization is done. By default: 1.
variable_names	character. Names of the variables. By default NULL. Ignored if the data is a matrix (univariate functional case).
times	numeric. Times of observation of the data. By default NULL.
initialization	list. Initialization for the scan procedure (see InitScan for more details).
permutations	matrix. Indices of permutations of the data.

Value

An object of class ResScanOutputUniFunct or ResScanOutputMultiFunct depending on the data

References

Zaineb Smida and Lionel Cucala and Ali Gannoun and Ghislain Durif (2022). A Wilcoxon-Mann-Whitney spatial scan statistic for functional data. Computational Statistics & Data Analysis, 167.

`permutate` *Permutates the data*

Description

This function will permit to permute the data for the MC simulations

Usage

```
permutate(to_permute, nb_permu)
```

Arguments

`to_permute`	vector. Vector of indices we want to permute.
`nb_permu`	numeric. Number of permutations.

Value

matrix. Matrix of nb_permu rows and length(to_permute) columns.

`PFSS` *PFSS scan procedure*

Description

This function computes the PFSS (Parametric Functional scan statistic).

Usage

```
PFSS(
  data,
  MC = 999,
  typeI = 0.05,
  nbCPU = 1,
  times = NULL,
  initialization,
  permutations
)
```

Arguments

data	matrix. Matrix of the data, the rows correspond to the sites (or to the individuals if the observations are by individuals and not by sites) and each column represents an observation time. The times must be equally spaced and the same for each site/individual.
MC	numeric. Number of Monte-Carlo permutations to evaluate the statistical significance of the clusters. By default: 999.
typeI	numeric. The desired type I error. A cluster will be evaluated as significant if its associated p-value is less than typeI. By default 0.05.
nbCPU	numeric. Number of CPU. If nbCPU > 1 parallelization is done. By default: 1.
times	numeric. Times of observation of the data. By default NULL.
initialization	list. Initialization for the scan procedure (see InitScan for more details).
permutations	matrix. Indices of permutations of the data.

Value

An object of class ResScanOutputUniFunct.

References

Camille Frévent and Mohamed-Salem Ahmed and Matthieu Marbac and Michaël Genin (2021). Detecting Spatial Clusters in Functional Data: New Scan Statistic Approaches. Spatial Statistics, 46.

---

plot.ResScanOutput *Schema or map of the clusters*

Description

This function plots a schema or a map of the sites and the clusters

Usage

```
## S3 method for class 'ResScanOutput'
plot(
  x,
  type,
  spobject = NULL,
  system_conv = NULL,
  colors = "red",
  only.MLC = FALSE,
  ...
)
```

Arguments

x	ResScanOutput. Output of a scan function (UG, UNP, MG, MNP, PFSS, DFFSS, URBFSS, NPFSS, MPFSS, MDFFSS or MRBFSS)
type	character. Type of plot: "schema", "map" (the clusters are represented by circles) or "map2" (the clusters are colored on the map)
spobject	SpObject. SpatialObject with the same coordinates system the one used for the scan. Only considered if type is "map" or "map2"
system_conv	character. System to convert the coordinates for the plot. Only considered if the system used in the scan was "WGS84" and if type is "schema". Else it will be ignored. Must be entered as in the PROJ.4 documentation
colors	character. Colors of the clusters. If length(colors)=1 all the clusters will be in this color. Else it should be a vector of length the number of clusters to plot.
only.MLC	logical. Should we plot only the MLC or all the significant clusters?
...	Further arguments to be passed to or from methods.

Value

No value returned, plots a schema or a map of the sites and the clusters.

Examples

```
library(sp)
data("map_sites")
data("funi_data")
coords <- coordinates(map_sites)

res_npfss <- SpatialScan(method = "NPFSS", data = funi_data, sites_coord = coords,
system = "WGS84", mini = 1, maxi = nrow(coords)/2)$NPFSS

plot(x = res_npfss, type = "schema", system_conv = "+init=epsg:2154")
plot(x = res_npfss, type = "map", spobject = map_sites)
plot(x = res_npfss, type = "map2", spobject = map_sites)
```

---

plot_map *Map of circular clusters*

Description

This function plots a map of the sites and the circular clusters.

Usage

```
plot_map(spobject, centres, radius, system, colors = "red")
```

Arguments

spobject	SpObject. SpatialObject with the same coordinates system that centres (the same that sites_coord in the scan functions)
centres	numeric matrix or vector if only one cluster was detected. Coordinates of the centres of each cluster.
radius	numeric vector. Radius of each cluster in the user units if system = "Euclidean", or in km if system = "WGS84" (in the output of the scan functions)
system	character. System in which the coordinates are expressed: "Euclidean" or "WGS84".
colors	character. Colors of the clusters. If length(colors)=1 all the clusters will be in this color. Else it should be a vector of length the number of clusters to plot.

Value

No value returned, plots a map of the sites and the circular clusters.

---

plot_map2 *Map of the clusters*

Description

This function plots a map of the sites and the clusters

Usage

```
plot_map2(spobject, sites_coord, output_clusters, system, colors = "red")
```

Arguments

spobject	SpObject. SpatialObject corresponding the sites.
sites_coord	numeric matrix. Coordinates of the sites or the individuals, in the same order that the data for the cluster detection.
output_clusters	list. List of the sites in the clusters: it is the sites_clusters of the output of NPFSS, PFSS, DFFSS, URBFSS, MDFFSS, MRBFSS, MG, MNP, UG or UNP, or the sites_clusters_LH/sites_clusters_W/sites_clusters_P/sites_clusters_R of the MPFSS.
system	character. System in which the coordinates are expressed: "Euclidean" or "WGS84".
colors	character. Colors of the clusters. If length(colors)=1 all the clusters will be in this color. Else it should be a vector of length the number of clusters to plot.

Value

No value returned, plots a map of the sites and the clusters.

plot_schema *Schema of the clusters*

Description

This function plots a schema of the sites and the clusters

Usage

```
plot_schema(
  output_clusters,
  sites_coord,
  system,
  system_conv = NULL,
  colors = "red"
)
```

Arguments

output_clusters	list. List of the sites in the clusters: it is the sites_clusters of the output of NPFSS, PFSS, DFFSS, URBFSS, MDFFSS, MRBFSS, MG, MNP, UG or UNP, or the sites_clusters_LH/sites_clusters_W/sites_clusters_P/sites_clusters_R of the MPFSS.
sites_coord	numeric matrix. Coordinates of the sites, in the same order that the data for the cluster detection.
system	character. System in which the coordinates are expressed: "Euclidean" or "WGS84".
system_conv	character. System to convert the coordinates for the plot. Only considered if system is "WGS84". Must be entered as in the PROJ.4 documentation
colors	character. Colors of the clusters. If length(colors)=1 all the clusters will be in this color. Else it should be a vector of length the number of clusters to plot.

Value

No value returned, plots a schema of the sites and the clusters.

plotCurves *Generic function to plot curves*

Description

This function is a generic function to plot curves.

Usage

```
plotCurves(x, ...)
```

Arguments

x	An object for which the curves are to be plotted.
...	Additional arguments affecting the output.

Value

No value returned, plots the curves.

See Also

plotCurves.ResScanOutputUniFunct and plotCurves.ResScanOutputMultiFunct

Examples

```
library(sp)
data("map_sites")
data("funi_data")
coords <- coordinates(map_sites)

res_npfss <- SpatialScan(method = "NPFSS", data = funi_data, sites_coord = coords, system = "WGS84",
mini = 1, maxi = nrow(coords)/2)$NPFSS

plotCurves(x = res_npfss, add_mean = TRUE, add_median = TRUE)
```

---

plotCurves.ResScanOutputMultiFunct

Plots the curves in the clusters detected by the multivariate functional scan functions (MPFSS, NPFSS, MDFFSS or MRBFSS)

Description

This function plot the curves in the clusters detected by the multivariate functional scan functions (MPFSS, NPFSS, MDFFSS or MRBFSS).

Usage

```
## S3 method for class 'ResScanOutputMultiFunct'
plotCurves(
  x,
  add_mean = FALSE,
  add_median = FALSE,
  colors = "red",
```

```
  only.MLC = FALSE,
  ...
)
```

Arguments

x	ResScanOutputMultiFunct. Output of a multivariate functional scan function (MPFSS, NPFSS, MDFFSS or MRBFSS).
add_mean	boolean. If TRUE it adds the global mean curve in black.
add_median	boolean. If TRUE it adds the global median curve in blue.
colors	character. The colors to plot the clusters' curves. If length(colors)==1 then all the clusters will be plotted in this color. Else there must be the same number of elements in colors than the number of clusters
only.MLC	logical. Should we plot only the MLC or all the significant clusters?
...	Further arguments to be passed to or from methods.

Value

No value returned, plots the curves.

Examples

```
library(sp)
data("map_sites")
data("fmulti_data")
coords <- coordinates(map_sites)

res_npfss <- SpatialScan(method = "NPFSS", data = fmulti_data, sites_coord = coords,
system = "WGS84", mini = 1, maxi = nrow(coords)/2)$NPFSS

plotCurves(x = res_npfss, add_mean = TRUE, add_median = TRUE)
```

---

plotCurves.ResScanOutputUniFunct

Plots the curves in the clusters detected by the univariate functional scan functions (PFSS, NPFSS, DFFSS or URBFSS)

Description

This function plot the curves in the clusters detected by the univariate functional scan functions (PFSS, NPFSS, DFFSS or URBFSS).

Usage

```
## S3 method for class 'ResScanOutputUniFunct'
plotCurves(
  x,
  add_mean = FALSE,
  add_median = FALSE,
  colors = "red",
  only.MLC = FALSE,
  ...
)
```

Arguments

x	ResScanOutputUniFunct. Output of a univariate functional scan function (PFSS, NPFSS, DFFSS or URBFSS).
add_mean	boolean. If TRUE it adds the global mean curve in black.
add_median	boolean. If TRUE it adds the global median curve in blue.
colors	character. The colors to plot the clusters' curves. If length(colors)==1 then all the clusters will be plotted in this color. Else there must be the same number of elements in colors than the number of clusters
only.MLC	logical. Should we plot only the MLC or all the significant clusters?
...	Further arguments to be passed to or from methods.

Value

No value returned, plots the curves.

Examples

```
library(sp)
data("map_sites")
data("funi_data")
coords <- coordinates(map_sites)

res_npfss <- SpatialScan(method = "NPFSS", data = funi_data, sites_coord = coords,
system = "WGS84", mini = 1, maxi = nrow(coords)/2)$NPFSS

plotCurves(x = res_npfss, add_mean = TRUE, add_median = TRUE)
```

plotSummary *Generic function to plot a summary*

Description

This function is a generic function to plot a summary.

Usage

```
plotSummary(x, ...)
```

Arguments

x	An object for which the summary is to be plotted.
...	Additional arguments affecting the summary produced.

Value

No value returned, plots the summary.

See Also

plotSummary.ResScanOutputMulti, plotSummary.ResScanOutputUniFunct and plotSummary.ResScanOutputMultiFu

Examples

```
library(sp)
data("map_sites")
data("multi_data")
coords <- coordinates(map_sites)

res_mnp <- SpatialScan(method = "MNP", data = multi_data, sites_coord = coords,
system = "WGS84", mini = 1, maxi = nrow(coords)/2,
variable_names = c("NO2", "O3", "PM10", "PM2.5"))$MNP

plotSummary(x = res_mnp, type = "mean")
```

`plotSummary.ResScanOutputMulti` *Plots the mean or median spider chart of the clusters detected by a multivariate scan function (MG or MNP)*

Description

This function plots the mean or median spider chart of the clusters detected by a multivariate scan function (MG or MNP).

Usage

```
## S3 method for class 'ResScanOutputMulti'
plotSummary(x, type = "mean", colors = "red", only.MLC = FALSE, ...)
```

Arguments

`x`	ResScanOutputMulti. Output of a multivariate scan function (MG or MNP).
`type`	character. "mean" or "median". If "mean": the means in the clusters are plotted in solid lines, outside the cluster in dots, the global mean is in black. If "median": the medians in the clusters are plotted in solid lines, outside the cluster in dots, the global median is in black.
`colors`	character. The colors to plot the clusters' summaries. If length(colors)==1 then all the clusters will be plotted in this color. Else there must be the same number of elements in colors than the number of clusters
`only.MLC`	logical. Should we plot only the MLC or all the significant clusters?
`...`	Further arguments to be passed to or from methods.

Value

No value returned, plots the spider chart.

Examples

```
library(sp)
data("map_sites")
data("multi_data")
coords <- coordinates(map_sites)

res_mnp <- SpatialScan(method = "MNP", data=multi_data, sites_coord = coords,
system = "WGS84", mini = 1, maxi = nrow(coords)/2,
variable_names = c("NO2", "O3", "PM10", "PM2.5"))$MNP

plotSummary(x = res_mnp, type = "mean")
```

plotSummary.ResScanOutputMultiFunct

Plots the mean or median curves in the clusters detected by a multivariate functional scan procedure (MPFSS, NPFSS, MDFFSS or MRBFSS)

Description

This function plots the mean or median curves in the clusters detected by a multivariate functional scan procedure (MPFSS, NPFSS, MDFFSS or MRBFSS).

Usage

```
## S3 method for class 'ResScanOutputMultiFunct'
plotSummary(x, type = "mean", colors = "red", only.MLC = FALSE, ...)
```

Arguments

x	ResScanOutputMultiFunct. Output of a multivariate functional scan function (MPFSS, NPFSS, MDFFSS or MRBFSS).
type	character. "mean" or "median". If "mean": the mean curves in the clusters are plotted in solid lines, outside the cluster in dots, the global mean curve is in black. If "median": the median curves in the clusters are plotted in solid lines, outside the cluster in dots, the global median curve is in black.
colors	character. The colors to plot the clusters' summary curves. If length(colors)==1 then all the clusters will be plotted in this color. Else there must be the same number of elements in colors than the number of clusters
only.MLC	logical. Should we plot only the MLC or all the significant clusters?
...	Further arguments to be passed to or from methods.

Value

No value returned, plots the curves.

Examples

```
library(sp)
data("map_sites")
data("fmulti_data")
coords <- coordinates(map_sites)

res_npfss <- SpatialScan(method = "NPFSS", data = fmulti_data, sites_coord = coords,
system = "WGS84", mini = 1, maxi = nrow(coords)/2)$NPFSS

plotSummary(x = res_npfss, type = "median")
```

`plotSummary.ResScanOutputUniFunct` *Plots the mean or median curves in the clusters detected by a univariate functional scan procedure (PFSS, NPFSS, DFFSS or URBFSS)*

Description

This function plots the mean or median curves in the clusters detected by a univariate functional scan procedure (PFSS, NPFSS, DFFSS or URBFSS).

Usage

```
## S3 method for class 'ResScanOutputUniFunct'
plotSummary(x, type = "mean", colors = "red", only.MLC = FALSE, ...)
```

Arguments

`x`	ResScanOutputUniFunct. Output of a univariate functional scan function (PFSS, NPFSS, DFFSS or URBFSS).
`type`	character. "mean" or "median". If "mean": the mean curves in the clusters are plotted in solid lines, outside the cluster in dots, the global mean curve is in black. If "median": the median curves in the clusters are plotted in solid lines, outside the cluster in dots, the global median curve is in black.
`colors`	character. The colors to plot the clusters' summary curves. If length(colors)==1 then all the clusters will be plotted in this color. Else there must be the same number of elements in colors than the number of clusters
`only.MLC`	logical. Should we plot only the MLC or all the significant clusters?
`...`	Further arguments to be passed to or from methods.

Value

No value returned, plots the curves.

Examples

```
library(sp)
data("map_sites")
data("funi_data")
coords <- coordinates(map_sites)

res_npfss <- SpatialScan(method = "NPFSS", data = funi_data, sites_coord = coords,
system = "WGS84", mini = 1, maxi = nrow(coords)/2)$NPFSS

plotSummary(x = res_npfss, type = "median")
```

pointwise_dfree *Index for the DFFSS scan procedure*

Description

This function returns the index we want to maximize on the set of potential clusters, for each potential cluster

Usage

```
pointwise_dfree(data, matrix_clusters)
```

Arguments

data	numeric matrix. Matrix of the data. The rows correspond to the sites (or the individuals) and each column represents an observation time.
matrix_clusters	numeric matrix. Matrix in which each column represents a potential cluster. It is the result of the "clusters" function.

Value

numeric vector.

pointwise_wmw_multi *Index for the MRBFSS scan procedure*

Description

This function returns the index we want to maximize on the set of potential clusters, for each potential cluster

Usage

```
pointwise_wmw_multi(transform_data, matrix_clusters)
```

Arguments

transform_data	List. List of the data transformed with the function transform_data, each element of the list corresponds to an observation time. Each row of each element is a site (or an individual), and each column represents a variable.
matrix_clusters	numeric matrix. Matrix in which each column represents a potential cluster. It is the result of the "clusters" function.

Value

numeric vector.

pointwise_wmw_uni *Index for the URBFSS scan procedure*

Description

This function returns the index we want to maximize on the set of potential clusters, for each potential cluster

Usage

```
pointwise_wmw_uni(rank_data, matrix_clusters)
```

Arguments

rank_data	matrix. Matrix of the ranks of the data for each time. Each column corresponds to an observation time and each row corresponds to a site or an individual.
matrix_clusters	numeric matrix. Matrix in which each column represents a potential cluster. It is the result of the "clusters" function.

Value

numeric vector.

post_filt_area *A posteriori filtering on the area*

Description

This function allows the a posteriori filtering on the area.

Usage

```
post_filt_area(mini_post, maxi_post, areas_clusters, index_clusters_temp)
```

Arguments

mini_post	numeric. A minimum to filter the significant clusters a posteriori. The default NULL is for no filtering with a a posteriori minimum.
maxi_post	numeric. A maximum to filter the significant clusters a posteriori. The default NULL is for no filtering with a a posteriori maximum.
areas_clusters	numeric vector. The areas of the clusters.
index_clusters_temp	numeric vector. The indices of the detected clusters.

Value

The detecting clusters with the a posteriori filtering.

`post_filt_nb_sites` *A posteriori filtering on the number of sites/individuals*

Description

This function allows the a posteriori filtering on the number of sites/individuals.

Usage

```
post_filt_nb_sites(
  mini_post,
  maxi_post,
  nb_sites,
  index_clusters_temp,
  matrix_clusters
)
```

Arguments

`mini_post`	numeric. A minimum to filter the significant clusters a posteriori. The default NULL is for no filtering with a a posteriori minimum.
`maxi_post`	numeric. A maximum to filter the significant clusters a posteriori. The default NULL is for no filtering with a a posteriori maximum.
`nb_sites`	numeric. The number of sites/individuals.
`index_clusters_temp`	numeric vector. The indices of the detected clusters.
`matrix_clusters`	numeric matrix. Matrix in which each column represents a potential cluster. A value of 1 indicate that the site (or the individual) is in the cluster, 0 otherwise.

Value

The detecting clusters with the a posteriori filtering.

post_filt_radius *A posteriori filtering on the radius*

Description

This function allows the a posteriori filtering on the radius.

Usage

```
post_filt_radius(mini_post, maxi_post, radius, index_clusters_temp)
```

Arguments

mini_post	numeric. A minimum to filter the significant clusters a posteriori. The default NULL is for no filtering with a a posteriori minimum.
maxi_post	numeric. A maximum to filter the significant clusters a posteriori. The default NULL is for no filtering with a a posteriori maximum.
radius	numeric vector. The radius of each cluster.
index_clusters_temp	numeric vector. The indices of the detected clusters.

Value

The detecting clusters with the a posteriori filtering.

print.ResScanOutput *Prints a result of a scan procedure*

Description

This function prints a result of a scan procedure.

Usage

```
## S3 method for class 'ResScanOutput'
print(x, ...)
```

Arguments

x	ResScanOutput. Output of a scan function (UG, UNP, MG, MNP, PFSS, DFFSS, URBFSS, NPFSS, MPFSS, MDFFSS or MRBFSS)
...	Further arguments to be passed to or from methods.

Value

No value returned, print the ResScanOutput object.

Examples

```
library(sp)
data("map_sites")
data("funi_data")
coords <- coordinates(map_sites)

res_npfss <- SpatialScan(method = "NPFSS", data = funi_data, sites_coord = coords,
system = "WGS84", mini = 1, maxi = nrow(coords)/2)$NPFSS

print(x = res_npfss)
```

ResScanOutput *Constructor function for objects of the ResScanOutput class*

Description

This is the constructor function for objects of the ResScanOutput class.

Usage

```
ResScanOutput(
  sites_clusters,
  pval_clusters,
  centres_clusters,
  radius_clusters,
  areas_clusters,
  system,
  sites_coord,
  data,
  method
)
```

Arguments

sites_clusters	list. List of the indices of the sites of the selected clusters.
pval_clusters	numeric vector. The pvalues of the selected clusters.
centres_clusters	numeric matrix. Coordinates of the centres of the selected clusters.
radius_clusters	numeric vector. Radius of the selected clusters.
areas_clusters	numeric vector. Areas of the selected clusters.
system	character. System in which the coordinates are expressed: "Euclidean" or "WGS84".
sites_coord	numeric matrix. Coordinates of the sites (or the individuals, in that case there can be many individuals with the same coordinates).

data	list of numeric matrices or a matrix or a vector. List of nb_sites (or nb_individuals if the observations are by individuals and not by site) matrices of the data, the rows correspond to the variables and each column represents an observation time (multivariate functional case) ; or Matrix of the data, the rows correspond to the sites (or to the individuals) and each column represents an observation time (univariate functional case) or a variable (multivariate case) ; or Vector of the data, the elements correspond to the sites (or to the individuals) (univariate case).
method	character. The scan procedure used.

Value

An object of class ResScanOutput which is a list of the following elements:

- sites_clusters: List of the indices of the sites of the selected clusters.
- pval_clusters: The pvalues of the selected clusters.
- centres_clusters: Coordinates of the centres of the selected clusters.
- radius_clusters: Radius of the selected clusters.
- areas_clusters: Areas of the selected clusters.
- system: System in which the coordinates are expressed: "Euclidean" or "WGS84".
- sites_coord: Coordinates of the sites (or the individuals, in that case there can be many individuals with the same coordinates).
- data: List of numeric matrices or a matrix or a vector.
- method: The scan procedure used.

ResScanOutputMulti *Constructor function for objects of the ResScanOutputMulti class*

Description

This is the constructor function for objects of the ResScanOutputMulti class which inherits from class ResScanOutput.

Usage

```
ResScanOutputMulti(
  sites_clusters,
  pval_clusters,
  centres_clusters,
  radius_clusters,
  areas_clusters,
  system,
  variable_names = NULL,
  sites_coord,
  data,
  method
)
```

Arguments

sites_clusters	list. List of the indices of the sites of the selected clusters.
pval_clusters	numeric vector. The pvalues of the selected clusters.
centres_clusters	numeric matrix. Coordinates of the centres of the selected clusters.
radius_clusters	numeric vector. Radius of the selected clusters.
areas_clusters	numeric vector. Areas of the selected clusters.
system	character. System in which the coordinates are expressed: "Euclidean" or "WGS84".
variable_names	character. Names of the variables. By default NULL.
sites_coord	numeric matrix. Coordinates of the sites (or the individuals, in that case there can be many individuals with the same coordinates).
data	matrix. Matrix of the data, the rows correspond to the sites (or to the individuals) and each column represents a variable.
method	character. The scan procedure used.

Value

An object of class ResScanOutputMulti which is a list of the following elements:

- sites_clusters: List of the indices of the sites of the selected clusters.
- pval_clusters: The pvalues of the selected clusters.
- centres_clusters: Coordinates of the centres of the selected clusters.
- radius_clusters: Radius of the selected clusters.
- areas_clusters: Areas of the selected clusters.
- system: System in which the coordinates are expressed: "Euclidean" or "WGS84".
- sites_coord: Coordinates of the sites (or the individuals, in that case there can be many individuals with the same coordinates).
- data: Matrix.
- variable_names: names of the variables.
- method: The scan procedure used.

ResScanOutputMultiFunct

Constructor function for objects of the ResScanOutputMultiFunct class

---

Description

This is the constructor function for objects of the ResScanOutputMultiFunct class which inherits from class ResScanOutput.

Usage

```
ResScanOutputMultiFunct(
  sites_clusters,
  pval_clusters,
  centres_clusters,
  radius_clusters,
  areas_clusters,
  system,
  times = NULL,
  variable_names = NULL,
  sites_coord,
  data,
  method
)
```

Arguments

sites_clusters	list. List of the indices of the sites of the selected clusters.
pval_clusters	numeric vector. The pvalues of the selected clusters.
centres_clusters	numeric matrix. Coordinates of the centres of the selected clusters.
radius_clusters	numeric vector. Radius of the selected clusters.
areas_clusters	numeric vector. Areas of the selected clusters.
system	character. System in which the coordinates are expressed: "Euclidean" or "WGS84".
times	numeric. Times of observation of the data. By default NULL.
variable_names	character. Names of the variables. By default NULL.
sites_coord	numeric matrix. Coordinates of the sites (or the individuals, in that case there can be many individuals with the same coordinates).
data	list of numeric matrices. List of nb_sites (or nb_individuals if the observations are by individuals and not by site) matrices of the data, the rows correspond to the variables and each column represents an observation time.
method	character. The scan procedure used.

Value

An object of class ResScanOutputMultiFunct which is a list of the following elements:

- sites_clusters: List of the indices of the sites of the selected clusters.
- pval_clusters: The pvalues of the selected clusters.
- centres_clusters: Coordinates of the centres of the selected clusters.
- radius_clusters: Radius of the selected clusters.
- areas_clusters: Areas of the selected clusters.
- system: System in which the coordinates are expressed: "Euclidean" or "WGS84".

- sites_coord: Coordinates of the sites (or the individuals, in that case there can be many individuals with the same coordinates).
- data: list of numeric matrices.
- times: times of observation of the data.
- variable_names: names of the variables.
- method: the scan procedure used.

`ResScanOutputUni` *Constructor function for objects of the ResScanOutputUni class*

Description

This is the constructor function for objects of the ResScanOutputUni class which inherits from class ResScanOutput.

Usage

```
ResScanOutputUni(
  sites_clusters,
  pval_clusters,
  centres_clusters,
  radius_clusters,
  areas_clusters,
  system,
  sites_coord,
  data,
  method
)
```

Arguments

`sites_clusters`	list. List of the indices of the sites of the selected clusters.
`pval_clusters`	numeric vector. The pvalues of the selected clusters.
`centres_clusters`	numeric matrix. Coordinates of the centres of the selected clusters.
`radius_clusters`	numeric vector. Radius of the selected clusters.
`areas_clusters`	numeric vector. Areas of the selected clusters.
`system`	character. System in which the coordinates are expressed: "Euclidean" or "WGS84".
`sites_coord`	numeric matrix. Coordinates of the sites (or the individuals, in that case there can be many individuals with the same coordinates).
`data`	vector. Vector of the data, the elements correspond to the sites (or to the individuals).
`method`	character. The scan procedure used.

Value

An object of class ResScanOutputUni which is a list of the following elements:

- sites_clusters: List of the indices of the sites of the selected clusters.
- pval_clusters: The pvalues of the selected clusters.
- centres_clusters: Coordinates of the centres of the selected clusters.
- radius_clusters: Radius of the selected clusters.
- areas_clusters: Areas of the selected clusters.
- system: System in which the coordinates are expressed: "Euclidean" or "WGS84".
- sites_coord: Coordinates of the sites (or the individuals, in that case there can be many individuals with the same coordinates).
- data: Vector.
- method: The scan procedure used.

ResScanOutputUniFunct *Constructor function for objects of the ResScanOutputUniFunct class*

Description

This is the constructor function for objects of the ResScanOutputUniFunct class which inherits from class ResScanOutput.

Usage

```
ResScanOutputUniFunct(
  sites_clusters,
  pval_clusters,
  centres_clusters,
  radius_clusters,
  areas_clusters,
  system,
  times = NULL,
  sites_coord,
  data,
  method
)
```

Arguments

sites_clusters	list. List of the indices of the sites of the selected clusters.
pval_clusters	numeric vector. The pvalues of the selected clusters.
centres_clusters	numeric matrix. Coordinates of the centres of the selected clusters.

radius_clusters	numeric vector. Radius of the selected clusters.
areas_clusters	numeric vector. Areas of the selected clusters.
system	character. System in which the coordinates are expressed: "Euclidean" or "WGS84".
times	numeric. Times of observation of the data. By default NULL.
sites_coord	numeric matrix. Coordinates of the sites (or the individuals, in that case there can be many individuals with the same coordinates).
data	matrix. Matrix of the data, the rows correspond to the sites (or to the individuals) and each column represents an observation time.
method	character. The scan procedure used.

Value

An object of class ResScanOutputUniFunct which is a list of the following elements:

- sites_clusters: List of the indices of the sites of the selected clusters.
- pval_clusters: The pvalues of the selected clusters.
- centres_clusters: Coordinates of the centres of the selected clusters.
- radius_clusters: Radius of the selected clusters.
- areas_clusters: Areas of the selected clusters.
- system: System in which the coordinates are expressed: "Euclidean" or "WGS84".
- sites_coord: Coordinates of the sites (or the individuals, in that case there can be many individuals with the same coordinates).
- data: Matrix.
- times: times of observation of the data.
- method : the scan procedure used

SpatialScan *Spatial scan procedure*

Description

This function computes the different scan procedures available in the package.

Usage

```
SpatialScan(
  method,
  data,
  sites_coord = NULL,
  system = NULL,
  mini = 1,
  maxi = nrow(sites_coord)/2,
```

```
  type_minimaxi = "sites/indiv",
  mini_post = NULL,
  maxi_post = NULL,
  type_minimaxi_post = "sites/indiv",
  sites_areas = NULL,
  MC = 999,
  typeI = 0.05,
  nbCPU = 1,
  variable_names = NULL,
  times = NULL
)
```

Arguments

`method` character vector. The scan procedures to apply on the data. Possible values are:

- Univariate scan procedures: "UG" (univariate gaussian, see UG), "UNP" (univariate nonparametric, see UNP)
- Multivariate scan procedures: "MG" (multivariate gaussian, see MG), "MNP" (multivariate nonparametric, see MNP)
- Univariate functional scan procedures: "NPFSS" (nonparametric functional scan statistic, see NPFSS), "PFSS" (parametric functional scan statistic, see PFSS), "DFFSS" (distribution-free functional scan statistic, see DFFSS), "URBFSS" (univariate rank-based functional scan statistic, see URBFSS)
- Multivariate functional scan procedures: "NPFSS" (nonparametric functional scan statistic, see NPFSS), "MDFFSS" (multivariate distribution-free functional scan statistic, see MDFFSS), "MRBFSS" (multivariate rank-based functional scan statistic, see MRBFSS), "MPFSS", "MPFSS-LH", "MPFSS-W", "MPFSS-P" and "MPFSS-R" (parametric multivariate functional scan statistic ; "LH", "W", "P", "R" correspond respectively to the Lawley-Hotelling trace test statistic, The Wilks lambda test statistic, the Pillai trace test statistic and the Roy's maximum root test statistic, see MPFSS). Note that "MPFSS" computes "MPFSS-LH", "MPFSS-W", "MPFSS-P" and "MPFSS-R".

`data` list of numeric matrices or a matrix or a vector:

- Univariate case: Vector of the data, each element corresponds to a site (or an individual if the observations are by individuals and not by sites).
- Multivariate case: Matrix of the data, the rows correspond to the sites (or the individuals if the observations are by individuals and not by sites) and each column represents a variable.
- Univariate functional case: Matrix of the data, the rows correspond to the sites (or to the individuals if the observations are by individuals and not by sites) and each column represents an observation time. The times must be the same for each site/individual. Depending on the scan procedure they also need to be equally-spaced.
- Multivariate functional case: List of nb_sites (or nb_individuals if the observations are by individuals and not by sites) matrices of the data, the rows correspond to the variables and each column represents an observation time.

The times must be the same for each site/individual. Depending on the scan procedure they also need to be equally-spaced.

`sites_coord` numeric matrix. Coordinates of the sites (or the individuals, in that case there can be many individuals with the same coordinates).

`system` character. System in which the coordinates are expressed: "Euclidean" or "WGS84".

`mini` numeric. A minimum for the clusters (see type_minimaxi). Changing the default value may bias the inference.

`maxi` numeric. A Maximum for the clusters (see type_minimaxi). Changing the default value may bias the inference.

`type_minimaxi` character. Type of minimum and maximum: by default "sites/indiv": the mini and maxi are on the number of sites or individuals in the potential clusters. Other possible values are "area": the minimum and maximum area of the clusters, or "radius": the minimum and maximum radius.

`mini_post` numeric. A minimum to filter the significant clusters a posteriori (see type_minimaxi_post). The default NULL is for no filtering with a a posteriori minimum.

`maxi_post` numeric. A maximum to filter the significant clusters a posteriori (see type_minimaxi_post). The default NULL is for no filtering with a a posteriori maximum.

`type_minimaxi_post` character. Type of minimum and maximum a posteriori: by default "sites/indiv": the mini_post and maxi_post are on the number of sites or individuals in the significant clusters. Other possible values are "area": the minimum and maximum area of the clusters, or "radius": the minimum and maximum radius.

`sites_areas` numeric vector. Areas of the sites. It must contain the same number of elements than the rows of sites_coord. If the data is on individuals and not on sites, there can be duplicated values. By default: NULL

`MC` numeric. Number of Monte-Carlo permutations to evaluate the statistical significance of the clusters. By default: 999.

`typeI` numeric. The desired type I error. A cluster will be evaluated as significant if its associated p-value is less than typeI. By default 0.05.

`nbCPU` numeric. Number of CPU. If nbCPU > 1 parallelization is done. By default: 1. Ignored for "UG" and "UNP"

`variable_names` character. Names of the variables. By default NULL. Ignored for the univariate and univariate functional scan procedures.

`times` numeric. Times of observation of the data. By default NULL. Ignored for the univariate and multivariate scan procedures.

Value

A list of objects of class ResScanOutput:

- Univariate case (UG, UNP): A list of objects of class ResScanOutputUni
- Multivariate case (MG, MNP): A list of objects of class ResScanOutputMulti
- Univariate functional case (NPFSS, PFSS, DFFSS, URBFSS): A list of objects of class ResScanOutputUniFunct
- Multivariate functional case (NPFSS, MPFSS, MDFFSS, MRBFSS): A list of objects of class ResScanOutputMultiFunct

References

For univariate scan statistics:

- Inkyung Jung and Ho Jin Cho (2015). A Nonparametric Spatial Scan Statistic for Continuous Data. International Journal of Health Geographics, 14.
- Martin Kulldorff and Lan Huang and Kevin Konty (2009). A Scan Statistic for Continuous Data Based on the Normal Probability Model. International Journal of Health Geographics, 8 (58).

For multivariate scan statistics:

- Lionel Cucala and Michaël Genin and Florent Occelli and Julien Soula (2019). A Multivariate Nonparametric Scan Statistic for Spatial Data. Spatial statistics, 29, 1-14.
- Lionel Cucala and Michaël Genin and Caroline Lanier and Florent Occelli (2017). A Multivariate Gaussian Scan Statistic for Spatial Data. Spatial Statistics, 21, 66-74.

For functional scan statistics:

- Zaineb Smida and Lionel Cucala and Ali Gannoun. A Nonparametric Spatial Scan Statistic for Functional Data. Pre-print <https://hal.archives-ouvertes.fr/hal-02908496>.
- Camille Frévent and Mohamed-Salem Ahmed and Matthieu Marbac and Michaël Genin. Detecting Spatial Clusters in Functional Data: New Scan Statistic Approaches. Pre-print <arXiv:2011.03482>.
- Camille Frévent and Mohamed-Salem Ahmed and Sophie Dabo-Niang and Michaël Genin. Investigating Spatial Scan Statistics for Multivariate Functional Data. Pre-print <arXiv:2103.14401>.

See Also

ResScanOutput, ResScanOutputUni, ResScanOutputMulti, ResScanOutputUniFunct and ResScanOutputMultiFunct

Examples

```
# Univariate scan statistics

library(sp)
data("map_sites")
data("multi_data")
uni_data <- multi_data[,1]
coords <- coordinates(map_sites)
res <- SpatialScan(method = c("UG", "UNP"), data = uni_data, sites_coord = coords,
system = "WGS84", mini = 1, maxi = nrow(coords)/2)

# Multivariate scan statistics

library(sp)
data("map_sites")
data("multi_data")
coords <- coordinates(map_sites)
res <- SpatialScan(method = c("MG", "MNP"), data = multi_data, sites_coord = coords,
system = "WGS84", mini = 1, maxi = nrow(coords)/2)
```

```
# Univariate functional scan statistics

library(sp)
data("map_sites")
data("funi_data")
coords <- coordinates(map_sites)
res <- SpatialScan(method = c("NPFSS", "PFSS", "DFFSS", "URBFSS"), data = funi_data,
sites_coord = coords, system = "WGS84", mini = 1, maxi = nrow(coords)/2)

# Multivariate functional

library(sp)
data("map_sites")
data("fmulti_data")
coords <- coordinates(map_sites)
res <- SpatialScan(method = c("NPFSS", "MPFSS", "MDFFSS", "MRBFSS"), data = fmulti_data,
sites_coord = coords, system = "WGS84", mini = 1, maxi = nrow(coords)/2)
```

---

summary.ResScanOutputMulti *Summary of the clusters obtained with a multivariate scan function (MG or MNP).*

Description

This function gives a summary of the clusters in a table

Usage

```
## S3 method for class 'ResScanOutputMulti'
summary(
  object,
  type_summ = "param",
  digits = 3,
  quantile.type = 7,
  only.MLC = FALSE,
  ...
)
```

Arguments

object	ResScanOutputMulti. Output of a multivariate scan function (MG or MNP).
type_summ	character. "param" or "nparam". "param" gives the mean and the sd for each variable in the clusters, outside, and globally and "nparam" gives the Q25, Q50 and Q75 quantiles for each variables in the clusters, outside, and globally.

digits	integer. Number of decimals in output.
quantile.type	An integer between 1 and 9 (see function quantile). Ignored if type_summ is "param"
only.MLC	logical. Should we summarize only the MLC or all the significant clusters?
...	Further arguments to be passed to or from methods.

Value

No value returned, displays the results in the console

Examples

```
library(sp)
data("map_sites")
data("multi_data")
coords <- coordinates(map_sites)
res_mg <- SpatialScan(method = "MG", data=multi_data, sites_coord = coords,
system = "WGS84", mini = 1, maxi = nrow(coords)/2)$MG
summary(object = res_mg)
```

---

summary.ResScanOutputMultiFunct

Summary of the clusters obtained with a multivariate functional scan function (MPFSS, NPFSS, MDFFSS or MRBFSS).

Description

This function gives a summary of the clusters in a table

Usage

```
## S3 method for class 'ResScanOutputMultiFunct'
summary(
  object,
  type_summ = "param",
  digits = 3,
  quantile.type = 7,
  only.MLC = FALSE,
  ...
)
```

Arguments

object	ResScanOutputMultiFunct. Output of an multivariate functional scan function (MPFSS, NPFSS, MDFFSS or MRBFSS).
type_summ	character. "param" or "nparam". "param" gives the mean and the sd for each variable in the clusters, outside, and globally and "nparam" gives the Q25, Q50 and Q75 quantiles for each variables in the clusters, outside, and globally.
digits	integer. Number of decimals in the output.
quantile.type	An integer between 1 and 9 (see function quantile). Ignored if type_summ is "param"
only.MLC	logical. Should we summarize only the MLC or all the significant clusters?
...	Further arguments to be passed to or from methods.

Value

No value returned, displays the results in the console

Examples

```
library(sp)
data("map_sites")
data("fmulti_data")
coords <- coordinates(map_sites)

res_npfss <- SpatialScan(method = "NPFSS", data = fmulti_data, sites_coord = coords,
system = "WGS84", mini = 1, maxi = nrow(coords)/2)$NPFSS

summary(object = res_npfss, type_summ = "nparam")
```

---

summary.ResScanOutputUni

Summary of the clusters obtained with a univariate scan function (UG or UNP).

Description

This function gives a summary of the clusters in a table

Usage

```
## S3 method for class 'ResScanOutputUni'
summary(
  object,
  type_summ = "param",
  digits = 3,
```

```
  quantile.type = 7,
  only.MLC = FALSE,
  ...
)
```

Arguments

object	ResScanOutputUni. Output of a univariate scan function (UG or UNP).
type_summ	character. "param" or "nparam". "param" gives the mean and the sd for each variable in the clusters, outside, and globally and "nparam" gives the Q25, Q50 and Q75 quantiles for each variables in the clusters, outside, and globally.
digits	integer. Number of decimals in the output.
quantile.type	An integer between 1 and 9 (see function quantile). Ignored if type_summ is "param"
only.MLC	logical. Should we summarize only the MLC or all the significant clusters?
...	Further arguments to be passed to or from methods.

Value

No value returned, displays the results in the console

Examples

```
library(sp)
data("map_sites")
data("multi_data")
uni_data <- multi_data[,1]
coords <- coordinates(map_sites)
res_unp <- SpatialScan(method = "UNP", data=uni_data, sites_coord = coords,
system = "WGS84", mini = 1, maxi = nrow(coords)/2)$UNP

summary(object = res_unp, type_summ = "nparam")
```

---

summary.ResScanOutputUniFunct

Summary of the clusters obtained with a univariate functional scan function (PFSS, NPFSS, DFFSS or URBFSS).

Description

This function gives a summary of the clusters in a table

Usage

```
## S3 method for class 'ResScanOutputUniFunct'
summary(
  object,
  type_summ = "param",
  digits = 3,
  quantile.type = 7,
  only.MLC = FALSE,
  ...
)
```

Arguments

object	ResScanOutputUniFunct. Output of a univariate functional scan function (PFSS, NPFSS, DFFSS or URBFSS).
type_summ	character. "param" or "nparam". "param" gives the mean and the sd for each variable in the clusters, outside, and globally and "nparam" gives the Q25, Q50 and Q75 quantiles for each variables in the clusters, outside, and globally.
digits	integer. Number of decimals in the output.
quantile.type	An integer between 1 and 9 (see function quantile). Ignored if type_summ is "param"
only.MLC	logical. Should we summarize only the MLC or all the significant clusters?
...	Further arguments to be passed to or from methods.

Value

No value returned, displays the results in the console

Examples

```
library(sp)
data("map_sites")
data("funi_data")
coords <- coordinates(map_sites)

res_npfss <- SpatialScan(method = "NPFSS", data = funi_data, sites_coord = coords,
system = "WGS84", mini = 1, maxi = nrow(coords)/2)$NPFSS

summary(object = res_npfss, type_summ = "nparam")
```

transform_data *Computation of the multivariate functional ranks*

Description

This function computes the multivariate ranks of the data for each observation time

Usage

```
transform_data(data)
```

Arguments

data	List. List of the data, each element of the list corresponds to a site (or an individual), each row corresponds to a variable and each column represents an observation time.

Value

List

UG *UG scan procedure*

Description

This function computes the UG (Univariate Gaussian scan statistic).

Usage

```
UG(data, MC = 999, typeI = 0.05, initialization, permutations)
```

Arguments

data	vector. Vector of the data, each element corresponds to a site (or an individual if the observations are by individuals and not by sites).
MC	numeric. Number of Monte-Carlo permutations to evaluate the statistical significance of the clusters. By default: 999.
typeI	numeric. The desired type I error. A cluster will be evaluated as significant if its associated p-value is less than typeI. By default 0.05.
initialization	list. Initialization for the scan procedure (see InitScan for more details).
permutations	matrix. Indices of permutations of the data.

Value

An object of class ResScanOutputUni.

References

Martin Kulldorff and Lan Huang and Kevin Konty (2009). A Scan Statistic for Continuous Data Based on the Normal Probability Model. International Journal of Health Geographics, 8 (58).

`uni_fWMW` *Index for the NPFSS scan procedure (univariate functional case)*

Description

This function returns the index we want to maximize on the set of potential clusters, for each potential cluster

Usage

```
uni_fWMW(signs, matrix_clusters)
```

Arguments

`signs`	numeric matrix. Matrix of signs of the data, the rows correspond to the sites (or the individuals) and each column represents an observation time.
`matrix_clusters`	numeric matrix. Matrix in which each column represents a potential cluster. It is the result of the "clusters" function.

Value

numeric vector.

`uni_signs_matrix` *Computation of the matrix of signs*

Description

This function returns the matrix of signs of the data.

Usage

```
uni_signs_matrix(data)
```

Arguments

data	numeric matrix. Matrix of the data, the rows correspond to the sites (or the individuals) and each column represents an observation time.

Value

numeric matrix.

UNP *UNP scan procedure*

Description

This function computes the UNP (Univariate Nonparametric scan statistic).

Usage

```
UNP(data, MC = 999, typeI = 0.05, initialization, permutations)
```

Arguments

data	vector. Vector of the data, each element corresponds to a site (or an individual if the observations are by individuals and not by sites).
MC	numeric. Number of Monte-Carlo permutations to evaluate the statistical significance of the clusters. By default: 999.
typeI	numeric. The desired type I error. A cluster will be evaluated as significant if its associated p-value is less than typeI. By default 0.05.
initialization	list. Initialization for the scan procedure (see InitScan for more details).
permutations	matrix. Indices of permutations of the data.

Value

An object of class ResScanOutputUni.

References

Inkyung Jung and Ho Jin Cho (2015). A Nonparametric Spatial Scan Statistic for Continuous Data. International Journal of Health Geographics, 14.

URBFSS *URBFSS scan procedure*

Description

This function computes the URBFSS (Univariate Rank-Based Functional scan statistic).

Usage

```
URBFSS(
  data,
  MC = 999,
  typeI = 0.05,
  nbCPU = 1,
  times = NULL,
  initialization,
  permutations
)
```

Arguments

data	matrix. Matrix of the data, the rows correspond to the sites (or to the individuals if the observations are by individuals and not by sites) and each column represents an observation time. The times must be the same for each site/individual.
MC	numeric. Number of Monte-Carlo permutations to evaluate the statistical significance of the clusters. By default: 999.
typeI	numeric. The desired type I error. A cluster will be evaluated as significant if its associated p-value is less than typeI. By default 0.05.
nbCPU	numeric. Number of CPU. If nbCPU > 1 parallelization is done. By default: 1.
times	numeric. Times of observation of the data. By default NULL.
initialization	list. Initialization for the scan procedure (see InitScan for more details).
permutations	matrix. Indices of permutations of the data.

Value

An object of class ResScanOutputUniFunct.

See Also

MRBFSS which is the multivariate version of the URBFSS

wmw_uni *Index for the UNP scan procedure*

Description

This function returns the index we want to maximize on the set of potential clusters, for each potential cluster, and each permutation

Usage

```
wmw_uni(rank_data, matrix_clusters)
```

Arguments

rank_data	matrix. Matrix of the ranks of the data for all permutations. Each column corresponds to a permutation and each row corresponds to a site or an individual.
matrix_clusters	numeric matrix. Matrix in which each column represents a potential cluster. It is the result of the "clusters" function.

Value

numeric matrix.

Index

∗ **datasets**
fmulti_data, 9
funi_data, 9
map_sites, 11
multi_data, 17
∗ **package**
HDSpatialScan-package, 3

clusters, 4

DFFSS, 5, *44*
dfree, 6
dfree_index_multi, 7

FinScan, 7
fmulti_data, 9
funi_data, 9

HDSpatialScan *(*HDSpatialScan-package*)*, 3
HDSpatialScan-package, 3

InitScan, *6*, 10, *12–16*, *20*, *22*, *52*, *54, 55*

map_sites, 11
MDFFSS, 12, *44*
MG, 13, *44*
MNP, 14, *44*
MPFSS, 15, *44*
MRBFSS, 16, *44*, *55*
multi_data, 17
multi_fWMW, 17
multi_gaussian, 18
multi_signs_matrix, 18
multi_WMW, 19

non_overlap, 19
NPFSS, 20, *44*

permutate, 21
PFSS, 21, *44*
plot.ResScanOutput, 22
plot_map, 23
plot_map2, 24
plot_schema, 25
plotCurves, 25
plotCurves.ResScanOutputMultiFunct, 26, *26*
plotCurves.ResScanOutputUniFunct, *26*, 27
plotSummary, 29
plotSummary.ResScanOutputMulti, *29*, 30
plotSummary.ResScanOutputMultiFunct, *29*, 31
plotSummary.ResScanOutputUniFunct, *29*, 32
pointwise_dfree, 33
pointwise_wmw_multi, 33
pointwise_wmw_uni, 34
post_filt_area, 34
post_filt_nb_sites, 35
post_filt_radius, 36
print.ResScanOutput, 36

ResScanOutput, 37, *46*
ResScanOutputMulti, 38, *46*
ResScanOutputMultiFunct, 39, *46*
ResScanOutputUni, 41, *46*
ResScanOutputUniFunct, 42, *46*

SpatialScan, 43
summary.ResScanOutputMulti, 47
summary.ResScanOutputMultiFunct, 48
summary.ResScanOutputUni, 49
summary.ResScanOutputUniFunct, 50

transform_data, 52

UG, *44*, 52
uni_fWMW, 53
uni_signs_matrix, 53
UNP, *44*, 54

URBFSS, *44*, 55

wmw_uni, 56