Package: spatialcluster (via r-universe)

August 11, 2024

Title R port of redcap

Version 0.1.0.006

Description R port of redcap (Regionalization with dynamically constrained agglomerative clustering and partitioning).

License GPL-3

URL https://github.com/mpadge/spatialcluster

BugReports https://github.com/mpadge/spatialcluster/issues

Depends R (>= 3.3.0)

Imports alphahull, dplyr, ggplot2, ggthemes, magrittr, methods, Rcpp (>= 0.12.6), tibble, tripack

Suggests dbscan, knitr, rmarkdown, roxygen2, testthat

LinkingTo Rcpp, RcppArmadillo

VignetteBuilder knitr

Encoding UTF-8

LazyData true

NeedsCompilation yes

RoxygenNote 7.2.1

SystemRequirements C++11,

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

RemoteUrl https://github.com/mpadge/spatialcluster

RemoteRef HEAD

RemoteSha d8d36946558b44eedaab73af581175666fce375b

Contents

---

plot.scl *plot.scl*

Description

plot.scl

Usage

```
## S3 method for class 'scl'
plot(x, ..., hull_alpha = 1)
```

Arguments

x	object to be plotted
...	ignored here
hull_alpha	alpha value of (non-)convex hulls, with default generating a convex hull, and smaller values generating concave hulls. (See ?alphashape::ashape for details).

See Also

Other plot_fns: plot_merges()

Examples

```
set.seed (1)
n <- 100
xy <- matrix (runif (2 * n), ncol = 2)
dmat <- matrix (runif (n ^ 2), ncol = n)
scl <- scl_redcap (xy, dmat, ncl = 4)
plot (scl)
# Connect clusters according to highest (\code{shortest = FALSE}) values of
# \code{dmat}:
scl <- scl_redcap (xy, dmat, ncl = 4, shortest = FALSE, full_order = FALSE)
plot (scl)
```

plot_merges *plot_merges*

Description

Plot dendrogram of merges for `scl` object with `method = "full"`.

Usage

```
plot_merges(x, root_tree = FALSE)
```

Arguments

`x`	Object of class `scl` obtained with `method = "full"`.
`root_tree`	If `TRUE`, tree leaves are connected to bottom of plot, otherwise floating as determined by plot.hclust.

Value

Nothing (generates plot)

See Also

Other plot_fns: `plot.scl()`

scl_full *scl_full*

Description

Full spatially-constrained clustering.

Usage

```
scl_full(xy, dmat, ncl, linkage = "single", shortest = TRUE, nnbs = 6L)
```

Arguments

`xy`	Rectangular structure (matrix, data.frame, tibble), containing coordinates of points to be clustered.
`dmat`	Square structure (matrix, data.frame, tibble) containing distances or equivalent metrics between all points in `xy`. If `xy` has `n` rows, then `dat` must have `n` rows and `n` columns.
`ncl`	Desired number of clusters. See description of 'ncl_iterate' parameter for conditions under which actual number may be less than this value.

linkage	Either "single" or "average". For covariance clustering, use "single" with 'shortest = FALSE'.
shortest	If TRUE, the dmat is interpreted as distances such that lower values are preferentially selected; if FALSE, then higher values of dmat are interpreted to indicate stronger relationships, as is the case for example with covariances.
nnbs	Number of nearest neighbours to be used in calculating clustering trees. Triangulation will be used if nnbs <= 0.

See Also

Other clustering_fns: scl_recluster(), scl_redcap()

Examples

```
n <- 100
xy <- matrix (runif (2 * n), ncol = 2)
dmat <- matrix (runif (n ^ 2), ncol = n)
scl <- scl_full (xy, dmat, ncl = 4)
```

---

scl_recluster *scl_reccluster*

Description

Re-cut a spatial cluster tree (scl) at a different number of clusters.

Usage

```
scl_recluster(scl, ncl, shortest = TRUE, quiet = FALSE)
```

Arguments

scl	An scl object returned from scl_redcap.
ncl	Desired number of clusters. See description of 'ncl_iterate' parameter for conditions under which actual number may be less than this value.
shortest	If TRUE, the dmat is interpreted as distances such that lower values are preferentially selected; if FALSE, then higher values of dmat are interpreted to indicate stronger relationships, as is the case for example with covariances.
quiet	If 'FALSE' (default), display progress information on screen.

Value

Modified scl object in which tree is re-cut into ncl clusters.

See Also

Other clustering_fns: scl_full(), scl_redcap()

Examples

```
n <- 100
xy <- matrix (runif (2 * n), ncol = 2)
dmat <- matrix (runif (n ^ 2), ncol = n)
scl <- scl_redcap (xy, dmat, ncl = 4)
plot (scl)
scl <- scl_recluster (scl, ncl = 5)
plot (scl)
```

---

scl_redcap *scl_redcap*

Description

Cluster spatial data with REDCAP (REgionalization with Dynamically Constrained Agglomerative clustering and Partitioning) routines.

Usage

```
scl_redcap(
  xy,
  dmat,
  ncl,
  full_order = TRUE,
  linkage = "single",
  shortest = TRUE,
  nnbs = 6L,
  iterate_ncl = FALSE,
  quiet = FALSE
)
```

Arguments

xy	Rectangular structure (matrix, data.frame, tibble), containing coordinates of points to be clustered.
dmat	Square structure (matrix, data.frame, tibble) containing distances or equivalent metrics between all points in xy. If xy has n rows, then dat must have n rows and n columns.
ncl	Desired number of clusters. See description of 'ncl_iterate' parameter for conditions under which actual number may be less than this value.
full_order	If FALSE, build spanning trees from first-order relationships only, otherwise build from full-order relationships (see Note).
linkage	One of "single", "average", or "complete"; see Note.
shortest	If TRUE, the dmat is interpreted as distances such that lower values are preferentially selected; if FALSE, then higher values of dmat are interpreted to indicate stronger relationships, as is the case for example with covariances.

nnbs	Number of nearest neighbours to be used in calculating clustering trees. Triangulation will be used if nnbs <= 0.
iterate_ncl	Actual numbers of clusters found may be less than the specified value of 'ncl', because clusters formed from < 3 edges are removed. If 'iterate_ncl = FALSE' (the default), the value is returned with whatever number of actual clusters is found. Setting this parameter to 'TRUE' forces the algorithm to iterate until the exact number of clusters has been found. For large data sets, this may result in considerable longer calculation times.
quiet	If 'FALSE' (default), display progress information on screen.

Value

A object of class scl with tree containing the clustering scheme, and xy the original coordinate data of the clustered points. An additional component, tree_rest, enables the tree to be re-cut to a different number of clusters via scl_recluster, rather than calculating clusters anew.

Note

Please refer to the original REDCAP paper ('Regionalization with dynamically constrained agglomerative clustering and partitioning (REDCAP)', by D. Guo (2008), Int.J.Geo.Inf.Sci 22:801-823) for details of the full_order and linkage parameters. This paper clearly demonstrates the general inferiority of spanning trees constructed from first-order relationships. It is therefore strongly recommended that the default full_order = TRUE be used at all times.

See Also

Other clustering_fns: scl_full(), scl_recluster()

Examples

```
n <- 100
xy <- matrix (runif (2 * n), ncol = 2)
dmat <- matrix (runif (n ^ 2), ncol = n)
scl <- scl_redcap (xy, dmat, ncl = 4)
# Those clusters will by default be constructed by connecting edges with the
# lowest (\code{shortest}) values of \code{dmat}, and will differ from
scl <- scl_redcap (xy, dmat, ncl = 4, shortest = FALSE)
# using 'full_order = FALSE' constructs clusters from first-order
# relationships only; not recommended, but possible nevertheless:
scl <- scl_redcap (xy, dmat, ncl = 4, full_order = FALSE)
```

---

spatialcluster *spatialcluster.*

Description

R port of redcap (Regionalization with dynamically constrained agglomerative clustering and partitioning).

Index