

Package ‘somspace’

April 28, 2023

Type Package

Title Spatial Analysis with Self-Organizing Maps

Version 1.2.4

Description Application of the Self-Organizing Maps technique for spatial classification of time series. The package uses spatial data, point or gridded, to create clusters with similar characteristics. The clusters can be further refined to a smaller number of regions by hierarchical clustering and their spatial dependencies can be presented as complex networks. Thus, meaningful maps can be created, representing the regional heterogeneity of a single variable. More information and an example of implementation can be found in Markonis and Strnad (2020, <[doi:10.1177/0959683620913924](https://doi.org/10.1177/0959683620913924)>).

License GPL-3

Depends R (>= 3.5.0), ggplot2, data.table, kohonen

Imports maps, reshape2

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

Suggests knitr, rmarkdown, testthat

VignetteBuilder knitr

NeedsCompilation no

Author Yannis Markonis [aut, cre],
Filip Strnad [aut],
Simon Michael Papalexiou [aut],
Mijael Rodrigo Vargas Godoy [ctb]

Maintainer Yannis Markonis <imarkonis@gmail.com>

Repository CRAN

Date/Publication 2023-04-28 17:40:02 UTC

R topics documented:

cnet 2

owda	3
plot_ts	3
regs	4
sominp	5
somregs	6
somsp	7
somspa	8

Index	10
--------------	-----------

cnet	<i>Complex network analysis</i>
------	---------------------------------

Description

cnet plots the canonical network map of a single classification scheme.

Usage

```
cnet(x, n, thres)
```

Arguments

x	regs object.
n	number of regions.
thres	the cross-correlation threshold of the network.

Details

The cnet function estimates the cross-correlation matrix of the average time series of each region and plots a map linking the regions with cross-correlations above the selected threshold.

Value

plot object

Examples

```
dummy <- owda[Time <= 1600]
inp_som <- sominp(dummy)
my_som <- somspa(inp_som, rlen = 100, grid = somgrid(3, 3, "hexagonal"))
my_regions <- somregs(my_som, nregions = 6)
cnet(my_regions, n = 5, thres = 0.2)
```

owda	<i>Old World Drought Atlas (1500-2012)</i>
------	--

Description

Reconstruction of European hydroclimate derived from tree-rings. The variable used is self-calibrated Palmer Drought Severity Index (scPDSI) at annual time step.

Usage

```
data(owda)
```

Format

An object of class `data.table` (inherits from `data.frame`) with 1355264 rows and 4 columns.

Source

[FZP archive](#)

References

Markonis et al. (2018) Nature Communications 9(1):1767 ([Nature Springer](#))

Examples

```
str(owda)
```

plot_ts	<i>Plot time series</i>
---------	-------------------------

Description

Plots the time series of SOM nodes or regions mean

Usage

```
plot_ts(x, n)
```

Arguments

x	is either a <code>somsp</code> or a <code>regs</code> object
n	is either the set of nodes for <code>somsp</code> or the number of regions for <code>regs</code>

Details

In case of `regs`, all the regions are plotted.

Value

plot object

See Also

[somspa](#)

regs

Regions class

Description

Regions class

Usage

regs

Format

An object of class regs of length 0.

Details

The regs class contains:

- A `summary.data.table` which updates the `somsp` object with the region ids of all classification schemes up to `nregions`. Each different classification scheme is stored as an individual region, e.g. `regions.2`, `regions.3`, etc.
- A `data.table` with the original data set, as in `somsp`.

It can be plotted by `plot` and `plot_ts`. If `plot` is used, three additional arguments are needed; a set with the classification schemes that will be plotted, number of rows and number of columns of the plotted panels. `plot_ts` plots all the time series of a given classification scheme.

See Also

[somsp](#)

[somregs](#)

sominp	<i>Create sominp object</i>
--------	-----------------------------

Description

sominp transforms the data set from `data.table` to `somsp` format, which can be used as argument in the `somspa` function.

Usage

```
sominp(x)
```

Arguments

`x` The `data.table` object which will be transformed to `somsp` object.

Details

`x` should be in tidy format with four columns: time, latitude, longitude and variable.

Value

A `sominp` object. It contains:

- a matrix that can be used as input for the `som` function of the `kohonen` package.
- a `data.table` with the with spatial point coordinates and a corresponding id.
- a `data.table` with the original dataset.

See Also

[somspa](#)

Examples

```
dummy <- owda[Time <= 1510]
inp_som <- sominp(dummy)
```

`somregs`*Classify SOM into regions*

Description

`somregs` applies hierarchical cluster analysis to the Self-Organizing Map to form regions with homogeneous characteristics (classification scheme).

Usage

```
somregs(x, nregions, ...)
```

Arguments

<code>x</code>	A <code>somsp</code> object.
<code>nregions</code>	The maximum number of classifications schemes to be determined starting from 2.
<code>...</code>	Other arguments passed to methods from <code>hclust</code> function which is used to determine the regions.

Details

`nregions` must be at least two, i.e., a classification scheme with two regions, and smaller than the number of SOM nodes. In the latter case, each SOM node corresponds to a region. The resulting `regs` object can be plotted by `plot` and `plot_ts`. If `plot` is used, three additional arguments are needed; a set with the classification schemes that will be plotted, number of rows and number of columns of the plotted panels. `plot_ts` plots all the time series of a given classification scheme.

Value

A `regs` object, which contains:

- A summary `data.table` which updates the `somsp` object with the region ids of all classification schemes up to `nregions`. Each different classification scheme is stored as an individual region, e.g., `regions.2`, `regions.3`, etc. to their corresponding winning unit, the number of points of each node, as well as the median latitude and longitude of each node coordinates and their standard deviation.
- The original time series which created the SOM as a `data.table`, as in `somsp`.

See Also

[somsp](#)

[somspa](#)

Examples

```
dummy <- owda[Time <= 1600]
inp_som <- sominp(dummy)
my_som <- somspa(inp_som, rlen = 100, grid = somgrid(4, 4, "hexagonal"))
my_regions <- somregs(my_som, nregions = 9)
plot(my_regions, regions = c(2, 4, 6, 8), nrow = 2, ncol = 2)
plot_ts(my_regions, n = 4)
```

somsp

Spatial SOM class

Description

Spatial SOM class

Usage

```
somsp
```

Format

An object of class somsp of length 0.

Details

The somsp objects are created by somspa function and contain:

- A summary data.table with the coordinates of each SOM node, the distances of objects to their corresponding winning unit, the number of points of each node, as well as the median latitude and longitude of each node coordinates and their standard deviation.
- A Self-Organizing Map object (see also [kohonen](#)).
- The sominp object used as input for the SOM, with an id number corresponding to location and a node number to the classification group of SOM.

They can be plotted by plot and plot_ts functions or summarized by summary.

See Also

[somspa](#)

[plot_ts](#)

`somspa`*Spatial SOM*

Description

`somspa` creates a Self-Organizing Map from spatial data.

Usage

```
somspa(x, ...)
```

Arguments

<code>x</code>	A <code>sominp</code> object.
<code>...</code>	Other arguments passed to methods from <code>kohonen::som</code> function which is used to create the SOM.

Details

`x` should be created by `sominp`. The output `somsp` objects can be plotted by `plot` and `plot_ts` functions or summarized by `summary`

Value

A `somsp` object, which contains:

- A summary `data.table` with the coordinates of each SOM node, the distances of objects to their corresponding winning unit, the number of points of each node, as well as the median latitude and longitude of each node coordinates and their standard deviation.
- A Self-Organizing Map object (see also [kohonen](#)).
- The `sominp` object used as input for the SOM, with an id number corresponding to location and a node number to the classification group of SOM.

See Also

[som](#)
[sominp](#)

Examples

```
dummy <- owda[Time <= 1600] #toy example
inp_som <- sominp(dummy)

my_som <- somspa(inp_som, rlen = 100, grid = somgrid(3, 3, "hexagonal"))
my_som$summary
my_som$som
```



```
plot(my_som)
plot_ts(my_som, n = 3)
plot_ts(my_som, n = c(1, 2, 4, 9))
plot_ts(my_som, n = 1:max(my_som$summary$node)) #plots all soms
```

Index

* datasets

- owda, 3
- regs, 4
- somsp, 7

cnet, 2

kohonen, 7, 8

owda, 3

plot_ts, 3, 7

regs, 4

som, 8

sominp, 5, 8

somregs, 4, 6

somsp, 4, 6, 7

somspa, 4–7, 8