Package 'spEDM'

May 16, 2025

Title Spatial Empirical Dynamic Modeling

Version 1.6

Description Inferring causation from spatial cross-sectional data through empirical dynamic modeling (EDM), with methodological extensions including geographical convergent cross mapping from Gao et al. (2023) <doi:10.1038/s41467-023-41619-6>, as well as the spatial causality test following the approach of Herrera et al. (2016) <doi:10.1111/pirs.12144>.

License GPL-3

Encoding UTF-8

RoxygenNote 7.3.2

URL https://stscl.github.io/spEDM/, https://github.com/stscl/spEDM

BugReports https://github.com/stscl/spEDM/issues

Depends R (>= 4.1.0)

LinkingTo Rcpp, RcppThread, RcppArmadillo

Imports dplyr, ggplot2, methods, sdsfun (>= 0.7.0), sf, terra

Suggests knitr, Rcpp, RcppThread, RcppArmadillo, rmarkdown, readr, plot3D

VignetteBuilder knitr

NeedsCompilation yes

Author Wenbo Lv [aut, cre, cph] (ORCID: <https://orcid.org/0009-0002-6003-3800>)

Maintainer Wenbo Lv <lyu.geosocial@gmail.com>

Repository CRAN

Date/Publication 2025-05-16 12:10:02 UTC

Contents

`detectThreads` *detect the number of available threads*

Description

detect the number of available threads

Usage

```
detectThreads()
```

Value

An integer

Examples

```
detectThreads()
```

`embedded` *embedding spatial cross sectional data*

Description

embedding spatial cross sectional data

Usage

```
## S4 method for signature 'sf'
embedded(data, target, E = 3, tau = 1, nb = NULL, trend.rm = FALSE)

## S4 method for signature 'SpatRaster'
embedded(data, target, E = 3, tau = 1, trend.rm = FALSE)
```

Arguments

data	The observation data.
target	Name of target variable.
E	(optional) Dimensions of the embedding.
tau	(optional) Step of spatial lags.
nb	(optional) The neighbours list.
trend.rm	(optional) Whether to remove the linear trend.

Value

A matrix

Examples

```
columbus = sf::read_sf(system.file("case/columbus.gpkg", package="spEDM"))
v = embedded(columbus,"crime")
v[1:5,]

cu = terra::rast(system.file("case/cu.tif", package="spEDM"))
r = embedded(cu,"cu")
r[1:5,]
```

---

fnn *false nearest neighbours*

Description

false nearest neighbours

Usage

```
## S4 method for signature 'sf'
fnn(
  data,
  target,
  lib = NULL,
  pred = NULL,
  E = 1:10,
  tau = 1,
  nb = NULL,
  rt = 10,
  eps = 2,
  threads = detectThreads(),
  trend.rm = TRUE
)
```

```
## S4 method for signature 'SpatRaster'
fnn(
  data,
  target,
  lib = NULL,
  pred = NULL,
  E = 1:10,
  tau = 1,
  rt = 10,
  eps = 2,
  threads = detectThreads(),
  trend.rm = TRUE
)
```

Arguments

data	The observation data.
target	Name of target variable.
lib	(optional) Libraries indices.
pred	(optional) Predictions indices.
E	(optional) Dimensions of the embedding.
tau	(optional) Step of spatial lags.
nb	(optional) The neighbours list.
rt	(optional) escape factor.
eps	(optional) neighborhood diameter.
threads	(optional) Number of threads.
trend.rm	(optional) Whether to remove the linear trend.

Value

A vector

References

Kennel M. B., Brown R. and Abarbanel H. D. I., Determining embedding dimension for phase-space reconstruction using a geometrical construction, Phys. Rev. A, Volume 45, 3403 (1992).

Examples

```
columbus = sf::read_sf(system.file("case/columbus.gpkg", package="spEDM"))

fnn(columbus,"crime")
```

gccm *geographical convergent cross mapping*

Description

geographical convergent cross mapping

Usage

```
## S4 method for signature 'sf'
gccm(
  data,
  cause,
  effect,
  libsizes,
  E = 3,
  tau = 1,
  k = E + 2,
  theta = 1,
  algorithm = "simplex",
  lib = NULL,
  pred = NULL,
  nb = NULL,
  threads = detectThreads(),
  parallel.level = "low",
  bidirectional = TRUE,
  trend.rm = TRUE,
  progressbar = TRUE
)

## S4 method for signature 'SpatRaster'
gccm(
  data,
  cause,
  effect,
  libsizes,
  E = 3,
  tau = 1,
  k = E + 2,
  theta = 1,
  algorithm = "simplex",
  lib = NULL,
  pred = NULL,
  threads = detectThreads(),
  parallel.level = "low",
  bidirectional = TRUE,
  trend.rm = TRUE,
```

```
  progressbar = TRUE
)
```

Arguments

data	The observation data.
cause	Name of causal variable.
effect	Name of effect variable.
libsizes	Number of spatial units used in prediction.
E	(optional) Dimensions of the embedding.
tau	(optional) Step of spatial lags.
k	(optional) Number of nearest neighbors used in prediction.
theta	(optional) Weighting parameter for distances, useful when algorithm is smap.
algorithm	(optional) Algorithm used in prediction.
lib	(optional) Libraries indices.
pred	(optional) Predictions indices.
nb	(optional) The neighbours list.
threads	(optional) Number of threads.
parallel.level	(optional) Level of parallelism, low or high.
bidirectional	(optional) whether to examine bidirectional causality.
trend.rm	(optional) Whether to remove the linear trend.
progressbar	(optional) whether to show the progress bar.

Value

A list

xmap cross mapping results

varname names of causal and effect variable

bidirectional whether to examine bidirectional causality

References

Gao, B., Yang, J., Chen, Z. et al. Causal inference from cross-sectional earth system data with geographical convergent cross mapping. Nat Commun 14, 5875 (2023).

Examples

```
columbus = sf::read_sf(system.file("case/columbus.gpkg", package="spEDM"))

g = gccm(columbus,"hoval","crime",libsizes = seq(5,45,5),E = 6)
g
plot(g, ylimits = c(0,0.85))
```

gcmc *geographical cross mapping cardinality*

Description

geographical cross mapping cardinality

Usage

```
## S4 method for signature 'sf'
gcmc(
  data,
  cause,
  effect,
  E = 3,
  tau = 1,
  k = NULL,
  r = 0,
  lib = NULL,
  pred = NULL,
  nb = NULL,
  threads = detectThreads(),
  bidirectional = TRUE,
  trend.rm = TRUE,
  progressbar = TRUE
)

## S4 method for signature 'SpatRaster'
gcmc(
  data,
  cause,
  effect,
  E = 3,
  tau = 1,
  k = NULL,
  r = 0,
  lib = NULL,
  pred = NULL,
  threads = detectThreads(),
  bidirectional = TRUE,
  trend.rm = TRUE,
  progressbar = TRUE
)
```

Arguments

data	The observation data.

cause	Name of causal variable.
effect	Name of effect variable.
E	(optional) Dimensions of the embedding.
tau	(optional) Step of spatial lags.
k	(optional) Number of nearest neighbors used in intersection.
r	(optional) Number of excluded neighbors in intersection.
lib	(optional) Libraries indices.
pred	(optional) Predictions indices.
nb	(optional) The neighbours list.
threads	(optional) Number of threads.
bidirectional	(optional) whether to examine bidirectional causality.
trend.rm	(optional) Whether to remove the linear trend.
progressbar	(optional) whether to show the progress bar.

Value

A list

`xmap` cross mapping results

`varname` names of causal and effect variable

`bidirectional` whether to examine bidirectional causality

Examples

```
columbus = sf::read_sf(system.file("case/columbus.gpkg", package="spEDM"))

g = gcmc(columbus,"hoval","crime",E = 5)
g
```

`multiview` *multiview embedding forecast*

Description

multiview embedding forecast

Usage

```
## S4 method for signature 'sf'
multiview(
  data,
  columns,
  target,
  nvar,
  lib = NULL,
  pred = NULL,
  E = 3,
  tau = 1,
  k = E + 2,
  nb = NULL,
  top = NULL,
  threads = detectThreads(),
  trend.rm = TRUE
)

## S4 method for signature 'SpatRaster'
multiview(
  data,
  columns,
  target,
  nvar,
  lib = NULL,
  pred = NULL,
  E = 3,
  tau = 1,
  k = E + 2,
  top = NULL,
  threads = detectThreads(),
  trend.rm = TRUE
)
```

Arguments

data	The observation data.
columns	Names of individual variables.
target	Name of target variable.
nvar	Number of variable combinations.
lib	(optional) Libraries indices.
pred	(optional) Predictions indices.
E	(optional) Dimensions of the embedding.
tau	(optional) Step of spatial lags.
k	(optional) Number of nearest neighbors used in prediction.
nb	(optional) The neighbours list.

top	(optional) Number of reconstructions used in MVE forecast.
threads	(optional) Number of threads.
trend.rm	(optional) Whether to remove the linear trend.

Value

A vector (when input is sf object) or matrix

References

Ye H., and G. Sugihara, 2016. Information leverage in interconnected ecosystems: Overcoming the curse of dimensionality. Science 353:922-925.

Examples

```
columbus = sf::read_sf(system.file("case/columbus.gpkg", package="spEDM"))

multiview(columbus,
          columns = c("inc","crime","open","plumb","discbd"),
          target = "hoval", nvar = 3)
```

---

sc.test *spatial causality test*

Description

spatial causality test

Usage

```
## S4 method for signature 'sf'
sc.test(
  data,
  cause,
  effect,
  k,
  block = 3,
  boot = 399,
  seed = 42,
  base = 2,
  lib = NULL,
  pred = NULL,
  nb = NULL,
  threads = detectThreads(),
  trend.rm = TRUE,
  normalize = FALSE,
```

```
  progressbar = FALSE
)

## S4 method for signature 'SpatRaster'
sc.test(
  data,
  cause,
  effect,
  k,
  block = 3,
  boot = 399,
  seed = 42,
  base = 2,
  lib = NULL,
  pred = NULL,
  threads = detectThreads(),
  trend.rm = TRUE,
  normalize = FALSE,
  progressbar = FALSE
)
```

Arguments

data	The observation data.
cause	Name of causal variable.
effect	Name of effect variable.
k	(optional) Number of nearest neighbors used in symbolization.
block	(optional) Number of blocks used in spatial block bootstrap.
boot	(optional) Number of bootstraps to perform.
seed	(optional) The random seed.
base	(optional) Base of the logarithm.
lib	(optional) Libraries indices.
pred	(optional) Predictions indices.
nb	(optional) The neighbours list.
threads	(optional) Number of threads.
trend.rm	(optional) Whether to remove the linear trend.
normalize	(optional) Whether to normalize the result.
progressbar	(optional) Whether to show the progress bar.

Value

A list

sc statistic for spatial causality

varname names of causal and effect variable

References

Herrera, M., Mur, J., & Ruiz, M. (2016). Detecting causal relationships between spatial processes. Papers in Regional Science, 95(3), 577–595.

Examples

```
columbus = sf::read_sf(system.file("case/columbus.gpkg", package="spEDM"))

sc.test(columbus,"hoval","crime", k = 15)
```

`simplex` *simplex forecast*

Description

simplex forecast

Usage

```
## S4 method for signature 'sf'
simplex(
  data,
  target,
  lib = NULL,
  pred = NULL,
  E = 1:10,
  tau = 1,
  k = E + 2,
  nb = NULL,
  threads = detectThreads(),
  trend.rm = TRUE
)

## S4 method for signature 'SpatRaster'
simplex(
  data,
  target,
  lib = NULL,
  pred = NULL,
  E = 1:10,
  tau = 1,
  k = E + 2,
  threads = detectThreads(),
  trend.rm = TRUE
)
```

Arguments

data	The observation data.
target	Name of target variable.
lib	(optional) Libraries indices.
pred	(optional) Predictions indices.
E	(optional) Dimensions of the embedding.
tau	(optional) Step of spatial lags.
k	(optional) Number of nearest neighbors used in prediction.
nb	(optional) The neighbours list.
threads	(optional) Number of threads.
trend.rm	(optional) Whether to remove the linear trend.

Value

A list

xmap self mapping prediction results

varname name of target variable

References

Sugihara G. and May R. 1990. Nonlinear forecasting as a way of distinguishing chaos from measurement error in time series. Nature, 344:734-741.

Examples

```
columbus = sf::read_sf(system.file("case/columbus.gpkg", package="spEDM"))

simplex(columbus,"crime")
```

---

smap *smap forecast*

Description

smap forecast

Usage

```
## S4 method for signature 'sf'
smap(
  data,
  target,
  lib = NULL,
  pred = NULL,
  E = 3,
  tau = 1,
  k = E + 2,
 theta = c(0, 1e-04, 3e-04, 0.001, 0.003, 0.01, 0.03, 0.1, 0.3, 0.5, 0.75, 1, 1.5, 2, 3,
    4, 6, 8),
  nb = NULL,
  threads = detectThreads(),
  trend.rm = TRUE
)

## S4 method for signature 'SpatRaster'
smap(
  data,
  target,
  lib = NULL,
  pred = NULL,
  E = 3,
  tau = 1,
  k = E + 2,
 theta = c(0, 1e-04, 3e-04, 0.001, 0.003, 0.01, 0.03, 0.1, 0.3, 0.5, 0.75, 1, 1.5, 2, 3,
    4, 6, 8),
  threads = detectThreads(),
  trend.rm = TRUE
)
```

Arguments

data	The observation data.
target	Name of target variable.
lib	(optional) Libraries indices.
pred	(optional) Predictions indices.
E	(optional) Dimensions of the embedding.
tau	(optional) Step of spatial lags.
k	(optional) Number of nearest neighbors used in prediction.
theta	(optional) Weighting parameter for distances.
nb	(optional) The neighbours list.
threads	(optional) Number of threads.
trend.rm	(optional) Whether to remove the linear trend.

Value

A list

`xmap` self mapping prediction results

`varname` name of target variable

References

Sugihara G. 1994. Nonlinear forecasting for the classification of natural time series. Philosophical Transactions: Physical Sciences and Engineering, 348 (1688):477-495.

Examples

```
columbus = sf::read_sf(system.file("case/columbus.gpkg", package="spEDM"))

smap(columbus,"inc")
```

Index