Package 'autoSTK'

July 16, 2021

Type Package

Title Automatic spatio-temporal kriging

Version 1.1.0

Author Insang Song

Maintainer Insang Song <sigmafelix@hotmail.com>

Description Automatic spatio-temporal Kriging
This package offers several functions to fit spatio-temporal variograms in terms of minimal SSE by expanding base functions in automap package (Hiemstra et al. 2010). Some candidate models are applied in default, and users can choose what theoretical models will be applied. Four cross-validation strategies for spatiotemporal data are implemented.

License MIT

Encoding UTF-8

BugReports https://github.com/sigmafelix/autoSTK/issues

URL https://github.com/sigmafelix/autoSTK

Depends roxygen2

Imports dplyr, automap, gstat, spacetime, sp, sf, lattice, rgl

Suggests

RoxygenNote 7.1.1

R topics documented:

autofitVariogramST *Automatically fit a spatiotemporal variogram from ST*DF*

Usage

```
autofitVariogramST(
                   stf,
                   formula,
                   typestv ='sumMetric',
                   candidate_model = c('Ste','Exc','Exp','Wav'),
                   guess_nugget = NULL,
                   guess_psill = NULL,
                   tlags = 0:6,
                   cutoff = 2e4,
                   width = 5e2,
                   aniso_method = 'vgm',
                   type_joint = 'Exp',
                   prodsum_k = NULL,
                   surface = FALSE,
                   cores = 1
                       )
```

Arguments

stf	An object of the STI-class or STF-class containing the data from which the spatiotemporal variogram to be estimated.
formula	formula that defines the dependent variable as a linear model of independent variables; suppose the dependent variable has name 'z', for ordinary and simple kriging use the formula 'z~1'; for simple kriging also define 'beta' (see below); for universal kriging, suppose 'z' is linearly dependent on 'x' and 'y', use the formula 'z~x+y'.
typestv	Model for the spatiotemporal variogram
candidate_model	List of models that will be tested during automatic variogram fitting. Default values are "Sph", "Exp", "Gau", and "Ste"
guess_nugget	A user-defined value for the nugget of the spatiotemporal variogram.
guess_psill	A user-defined value for the partial sill of the spatiotemporal variogram
tlags	The range of time lags for fitting STVariogram
cutoff	The maximum range of spatial lags for fitting STVariogram
width	The interval for fitting spatial part of the STVariogram
aniso_method	The method to estimate the spatiotemporal anisotropy (one of `linear`, `range`, `vgm`, or `metric`)
type_joint	The type of theoretical model of the joint spatiotemporal variogram. Only applied when `joint` type of spatiotemporal variogram is chosen.

prodsum_k	k value for stModel in vgmST is productSum. See vgmST for more. #
surface	logical. Set TRUE if you want to get the wireframe plot of the fitted spatiotemporal variogram
cores	The number of cores to be used for estimating variogramST. See variogramST for detail.

Value

jointSTV	The fitted theoretical spatiotemporal variogram
empSTV	The empirical spatiotemporal variogram estimated from the input data
SpV	The spatial component of jointSTV
TV	The temporal component of jointSTV
STVsurface	data.frame. The variogram surface of jointSTV

Author(s)

Insang Song, <sigmafelix@hotmail.com>

---

autoKrigeST *Performs an automatic spatiotemporal interpolation*

Description

This function performs automatic spatiotemporal kriging on the given dataset. The variogram is generated automatically using autofitVariogramST.

Usage

```
autoKrigeST(formula,
                    input_data,
                    new_data,
                    type_stv = 'sumMetric',
                    data_variogram = input_data,
                    block = 0,
                    model = c("Sph", "Exp", "Gau", "Ste"),
                    kappa = c(0.05, seq(0.2, 2, 0.1), 5, 10),
   fix.values = c(NA, NA, NA),
                    newdata_mode = 'rect',
   newdata_npoints = 3e3,
                    GLS.model = NA,
               tlags = 0:6,
               cutoff = 2e4,
               width = 5e2,
                    predict_chunk = NULL,
                    nmax = Inf,
```

```
                aniso_method = 'vgm',
                type_joint = 'Exp',
                prodsum_k = 0.25,
                start_vals = c(NA, NA, NA),
                     miscFitOptions = list(),
                     measurement_error = c(0,0,0),
                     cores = 1,
                     verbose = TRUE)
```

Arguments

formula	formula that defines the dependent variable as a linear model of independent variables; suppose the dependent variable has name 'z', for ordinary and simple kriging use the formula 'z~1'; for simple kriging also define 'beta' (see below); for universal kriging, suppose 'z' is linearly dependent on 'x' and 'y', use the formula 'z~x+y'.
input_data	An object of the STI, STS or STF containing the data to be interpolated.
new_data	A STI or STF object containing the prediction locations. Must not contain NA's. If this object is not provided a default is calculated. This is done by taking the convex hull of input_data and placing around 3000 gridcells in that convex hull.
data_variogram	An optional way to provide a different dataset for the building of the variogram then for the spatial interpolation.
block	Use this parameter to pass on a specification for the block size. e.g. c(1000,1000)
model	List of models that will be tested during automatic variogram fitting. Default values are "Sph", "Exp", "Gau", and "Ste"
kappa	List of values for the smoothing parameter of the Matern model that will be tested during automatic variogram fitting.
fix.values	Can be used to fix a variogram parameter to a certain value. It consists of a list with a length of three. The items describe the fixed value for the nugget, range and sill respectively. Setting the value to NA means that the value is not fixed. Is passed on to autofitVariogram.
newdata_mode	How the new data will be generated in shape. One of "rect" (rectangle) and "hull" (convex hull)
newdata_npoints	The number of points will be generated in the extent of the new data
GLS.model	If a variogram model is passed on through this parameter a Generalized Least Squares sample variogram is calculated.
tlags	The range of time lags for fitting STVariogram
cutoff	The maximum range of spatial lags for fitting STVariogram
width	The interval for fitting spatial part of the STVariogram
predict_chunk	The number of chunks to predict values in 'new_data'. If this value is not 'NULL', the new data will be split into chunks in size of 'predict_chunk' and the prediction will be done per chunk.

nmax	The maximum number of spatiotemporal neighborhood to make predictions (not stable)
aniso_method	The method to estimate the spatiotemporal anisotropy (one of linear, range, vgm, or metric)
type_joint	The type of theoretical model of the joint spatiotemporal variogram. Only applied when joint type of spatiotemporal variogram is chosen.
prodsum_k	k value for stModel in vgmST is productSum. See vgmST for more. #
start_vals	Can be used to give the starting values for the variogram fitting. The items describe the # fixed value for the nugget, range and sill respectively. They need to be given in that order. # Setting the value to NA means that the value will be automatically chosen.
miscFitOptions	Additional options to set the behavior of autofitVariogram. For details see the documentation of autofitVariogram.
measurement_error	integer vector (3). Adds measurement error components for spatial, temporal, and joint spatiotemporal variogram models, respectively. IT IS HIGHLY EXPERIMENTAL. MAY RESULT IN ERRORS.
cores	The number of cores to be used for estimating variogramST. See variogramST for more detail.
verbose	logical, if TRUE autoKrige will give extra information on the fitting process. Default is TRUE.

Details

autoKrigeST calls the function autofitVariogramST that fits a spatiotemporal variogram model to the given dataset. This variogram model and the data are used to make predictions on the spatiotemporal locations in new_data. If new_data is not specified, an internal function will automatically generate a new ST*DF data to perform the spatiotemporal interpolation.

Value

This function returns an autoKrige object containing the results of the interpolation (prediction, variance and standard deviation), the sample variogram, the variogram model that was fitted by autofitVariogram and the sums of squares between the sample variogram and the fitted variogram model. The attribute names are krige_output, exp_var, var_model and sserr respectively.

Author(s)

Insang Song, <sigmafelix@hotmail.com>

See Also

autofitVariogramST, krigeST

Examples

```
# The first part of the example is from the example of krigeST
library(spacetime)
library(sp)
library(stars)
data(air)
stations = spTransform(stations, CRS('+init=epsg:3857'))
rural = STFDF(stations, dates, data.frame(PM10 = as.vector(air)))

rr <- rural[,2701:2731]
rr <- as(rr,"STSDF")
air.stk <- autoKrigeST(formula = PM10~1,
                       input_data = rr,
                       type_stv = 'sumMetric',
                       tlags = 0:7,
                       cutoff = 3e5,
                       width = 2e4,
                       cores = 4)
stplot(air.stk[[1]])
```

`autoKrigeST.cv` *Cross-validation of spatiotemporal Kriging*

Description

Cross-validation of spatiotemporal Kriging

Usage

```
autoKrigeST.cv(
  data,
  fold_dim,
  nfold = 10L,
  formula,
  type_stv = "sumMetric",
  block = 0,
  model = c("Sph", "Exp", "Gau", "Ste"),
  kappa = c(0.05, seq(0.2, 2, 0.1), 5, 10),
  fix.values = c(NA, NA, NA),
  tlags = 0:6,
  cutoff = 20000,
  width = 500,
  nmax = Inf,
  aniso_method = "vgm",
  type_joint = "Exp",
  prodsum_k = 0.25,
  surface = FALSE,
```

```
  start_vals = c(NA, NA, NA),
  miscFitOptions = list(),
  measurement_error = c(0, 0, 0),
  cores = 1,
  seed = 130425L
)
```

Arguments

data	a 'STFDF'-class object
fold_dim	character. the dimension at which you want to cross-validate (spatial, temporal, and random)
nfold	integer. the number of folds. 10 as the default
formula	formula. e.g., y~1
type_stv	character. One of 'sumMetric', 'metric', 'productSum', and 'separable'
block	numeric. passed to conduct block spatiotemporal Kriging.
model	character vector. Default is c("Sph", "Exp", "Gau", "Ste"), but users can specify the list of theoretical variograms by referring gstat::vgm.
kappa	numeric vector. Kappa values tested for Matern-family variogram models.
fix.values	numeric vector. Initial values in order of nugget, range, and sill, respectively.
tlags	integer vector (increasing, preferably to be consecutive). temporal lags.
cutoff	numeric. The maximum distance at which the sample variogram will be computed.
width	numeric. The interval at which the variogram cloud will be summarized.
nmax	integer or positive infinite. The maximum number of spatiotemporal neighbors to conduct the local spatiotemporal Kriging.
aniso_method	character. One of 'vgm', 'linear', 'range', and 'metric'. Please refer to ?gstat::estiStAni.
type_joint	character. The model form of joint spatiotemporal variogram.
prodsum_k	numeric. The parameter for the case when 'productSum' is chosen for type_stv.
start_vals	numeric vector (3). The initial values to optimize the spatiotemporal variogram model.
miscFitOptions	list. See ?automap::autofitVariogram.
cores	integer. The number of threads that will be used to compute the sample spatiotemporal variogram.
newdata_mode	character. One of 'rect' (rectangular grid) and 'chull' (convex hull)
newdata_npoints	integer. The number of points that will be generated in the range of geometry the user specified (one of rectangle or convex hull)
GLS.model	a variogram model. The default value is NA. If a variogram model is passed, a Generalized Lease Squares sample variogram will be calculated.
predict_chunk	integer. The number of data points per chunk in the new data for the large data. It should be meticulously chosen according to the user's machine specification.

Value

The cross-validated spatiotemporal Kriging results.

Examples

```
library(sp)
library(gstat)
library(spacetime)
library(stars)
library(dplyr)
data(air)
deair = STFDF(stations, dates, data.frame(PM10 = as.vector(air)))
deair_sf = st_as_stars(deair, crs = '+proj=longlat +ellps=sphere')
deair_sf = st_transform(deair_sf, 3857)
deair_r = as(deair_sf, 'STFDF')
deair_r@sp@proj4string = CRS('+init=epsg:3857')
deair_rs = deair_r[,3751:3800]
## autoKrigeST.cv test
akst_cv_t = autoKrigeST.cv(formula = PM10~1, data = deair_rs,  nfold = 3, fold_dim = 'temporal',
                           cutoff = 300000, width = 30000, tlags = 0:7, cores = 8)
akst_cv_s = autoKrigeST.cv(formula = PM10~1, data = deair_rs,  nfold = 3, fold_dim = 'spatial',
                           cutoff = 300000, width = 30000, tlags = 0:7, cores = 8)
#akst_cv_r = autoKrigeST.cv(formula = PM10~1, data = deair_rs,  nfold = 3, fold_dim = 'random',
#                             cutoff = 300000, width = 30000, tlags = 0:7, cores = 8)
akst_cv_spt = autoKrigeST.cv(formula = PM10~1, data = deair_rs,  nfold = 4, fold_dim = 'spacetime',
                           cutoff = 300000, width = 30000, tlags = 0:7, cores = 8)
```

---

create_new_data *Create new spatial data for interpolation*

Description

Create new spatial data for interpolation

Usage

```
create_new_data(obj, gen_mode = "chull", npoints = 10000)
```

Arguments

obj	A Spatial*DataFrame.
gen_mode	character. One of 'rect' (rectangular) and 'chull' (convex hull).
npoints	integer. the number of points that will be generated

Value

A SpatialPointsDataFrame.

create_new_data.ST *Generate a new spatiotemporal points for the spatiotemporal prediction and interpolation*

Description

Generate a new spatiotemporal points for the spatiotemporal prediction and interpolation

Usage

```
create_new_data.ST(obj, form, gen_mode = "chull", npoints = 10000, forward = 6)
```

Arguments

obj	a ST*DF object.
form	formula.
gen_mode	character. One of 'rect' (rectangular) and 'chull' (convex hull).
npoints	integer. the number of points that will be generated
forward	integer. the time length of the data will generate ahead of the last time point of the input data. If NULL is passed, the spatiotemporal interpolation mode in obj will be conducted.

Value

A STFDF object.

detect_temporal_unit *Autodetect the temporal unit in a xts object*

Description

Autodetect the temporal unit in a xts object

Usage

```
detect_temporal_unit(temporal)
```

Arguments

temporal	a xts object.

Value

A character that indicates the temporal unit of the input xts object.

marginal.variogramST *Compute the marginal spatial or temporal sample variogram*

Description

Compute the marginal spatial or temporal sample variogram

Usage

```
marginal.variogramST(stv, bound, spatial = TRUE)
```

Arguments

stv	A STFDF.
bound	numeric. The maximum distance that will be used to compute a spatial variogram.
spatial	boolean. if TRUE, the spatial marginal variogram will be obtained, temporal otherwise.

Value

A gstatVariogram object.

plot.autofitVariogram *Plot the automatically fitted variogram*

Description

Plot the automatically fitted variogram

Usage

```
plot.autofitVariogram(
  x,
  plotit = TRUE,
  title = "Experimental variogram and fitted variogram model",
  ...
)
```

Arguments

x	A result object of autofitVariogram.
plotit	boolean. Print graph or not.
title	character. the title of the plot.
'...'	passed to xyplot

Value

A lattice::xyplot object.

`setSTI` *A convenience function for the sample spatiotemporal variogram*

Description

A convenience function for the sample spatiotemporal variogram

Usage

```
setSTI(
  stf,
  formula,
  tlags = 0:6,
  cutoff = 30000,
  width = 1000,
  assumeRegular = TRUE,
  pseudo = TRUE,
  logarithm = FALSE,
  na.omit = TRUE,
  wireframe = TRUE,
  plot3d = FALSE,
  cores = 1
)
```

Arguments

`stf`	A ST*DF object.
`formula`	formula (inherits the same parameter in variogramST)
`tlags`	temporal lags to compute semivariance (inherits the same parameter in variogramST)
`cutoff`	the maximum bound of the set of spatial lags (inherits the same parameter in variogramST)
`width`	integer (1). spatial lag (inherits the same parameter in variogramST)
`assumeRegular`	Boolean. Assuming regular grid?
`pseudo`	Boolean. See ?gstat::variogramST
`logarithm`	Boolean. log-transformation
`na.omit`	Boolean. Omit NA values.
`wireframe`	Boolean. Whether you plot a StVariogram in wireframe or not. If not, the return will be in class of data.frame, not a list
`plot3d`	Boolean. Wheter you make a three-dimensional graph with rgl package

Value

Depends on the arguments wireframe (if TRUE, list of length 2) and plot3d (if TRUE, list of length 3), a StVariogram object otherwise.