Package ‘StreamNetgstat’

January 21, 2019

Type Package

Title Spatial stream networks analysis

Version 1.0.0

Description This package provides a C++ implementation of functions that allow the user to fit a spatial linear model on a stream network, with the final goal to perform kriging. The data it can handle are the .ssn directories provided by the SSN package.

Depends R (>= 3.2.0), Rcpp (>= 0.12.16), RcppEigen (>= 0.3.3.4.0), SSN, rlist

LinkingTo Rcpp, RcppEigen

NeedsCompilation yes

SystemRequirements C++11

License What license is it under?

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

R topics documented:

do_SSN_kriging *Make prediction on a Spatial Stream Network object*

Description

Given a SpatialStreamNetwork-class object and the covariance parameters values, makes predictions using universal kriging.

Usage

```
do_SSN_kriging(ssn, varNames, weightVar, predpts, CorModels,
  useNugget = TRUE, theta, covMat, singleNet = NULL, matrices = NULL)
```

Arguments

ssn	a SpatialStreamNetwork-class object.
varNames	a vector of strings, the names of the variables used in the model.
weightVar	a string indicating the name of the variable to compute the spatial weights.
predpts	a string indicating the name of the group of points in which make predictions.
CorModels	a vector of strings, the names of the covariance models.
useNugget	If `FALSE` the nugget effect is not included in the model. Default to `TRUE`.
theta	vector of the parameters values used for computing the covariance matrix.
covMat	covariance matrix of the observed points.
singleNet	an interger, indicating the network ID that is to be analysed. Default to `NULL`, so that the analysis is carried on the entire dataset.
matrices	a vector of matrices, containing the flow-connection binary matrix, the hydrologic distance matrix and, not necessarily, the Euclidean distance matrix, returned by the function get_plots. These matrices consider the relationships between observed points.

Details

This function calculates prediction values and kriging variance for prediction sites based on the results of a linear model for the SpatialStreamNetwork-class object.

Value

A `ssn.object` (updated SpatialStreamNetwork-class object), with predicted values included

References

Peterson, E.E. and Ver Hoef, J.M. (2010) A mixed-model moving-average approach to geostatistical modeling in stream networks. Ecology 91(3), 644–651.

Ver Hoef, J.M. and Peterson, E.E. (2010) A moving average approach for spatial statistical models of stream networks (with discussion). Journal of the American Statistical Association 105, 6–18.

emp_semivario_and_torg *Compute the Empirical Semivariogram for a Spatial Stream Network object*

Description

Given a SpatialStreamNetwork-class object, computes the Torgegram (empirical semivariogram from the data based on hydrologic distance) and the empirical semivariogram from the data based on the Euclidean distance.

Usage

```
emp_semivario_and_torg(ssn, ResponseName, maxlag_Torg = NULL,
  nlag_Torg = 6, inc_Torg = 0, nlagcutoff_Torg = 15,
  maxlag_EmpVar = 1e+32, nlag_EmpVar = 20, inc_EmpVar = 0,
  nlagcutoff_EmpVar = 1, directions = c(0, 45, 90, 135),
  tolerance = 22.5, EmpVarMeth = "MethMoment")
```

Arguments

`ssn`	a SpatialStreamNetwork-class object.
`ResponseName`	the name of the response variable.
`maxlag_Torg`	the maximum lag distance to consider when binning pairs of locations by the hydrologic distance that separates them. The default is the median distance between all pairs of locations.
`nlag_Torg`	the number of lag bins to create for computing the Torgegram. The hydrologic distance between endpoints that define a bin will have equal lengths for all bins. The bin sizes are then determined from the minimum lag in the data, and the specification of maxlag_Torg.
`inc_Torg`	the bin hydrologic distance between endpoints. It is possible to specify the bin distance rather than nlag_Torg. In this case, the number of bins is determined by the bin distance and the hydrologic distance between the mininum and maximum (maxlag_Torg) lag in the data.
`nlagcutoff_Torg`	the minimum number of pairs needed to estimate the semivariance for a bin in the Torgegram computation. If the sample sizes is less than this value, the semivariance for the bin is not calculated.
`maxlag_EmpVar`	the maximum lag distance to consider when binning pairs of locations by the Euclidean distance that separates them. If the specified maxlag is larger than the maximum Euclidean distance among pairs of points, then maxlag_EmpVar is set to the maximum distance among pairs. If inc_EmpVar is greater than 0, then maxlag_EmpVar is disregarded.
`nlag_EmpVar`	the number of lag bins to create for computing the Empirical Semivariogram, by direction if directions are specified. The distance between endpoints that define a bin will have equal lengths for all bins. The bin sizes are then determined from the minimum lag in the data, and the specification of maxlag_EmpVar.
`inc_EmpVar`	the Euclidean distance increment for each bin class. Default is 0, in which case maxlag and nclasses determine the distance increments.

nlagcutoff_EmpVar
: the minimum number of pairs needed to estimate the semivariance for a bin in the Empirical Semivariogram. If the sample size is less than this value, the semivariance for the bin is not calculated.

directions
: directions in degrees clockwise from north that allow lag binning to be directional. Default is `c(0, 45, 90, 135)`. Values should be between 0 and 180, as there is radial symmetry in orientation between two points.

tolerance
: the angle on either side of the directions to determine if a pair of points falls in that direction class. Note, a pair of points may be in more than one lag bin if tolerances for different directions overlap.

EmpVarMeth
: method for computing semivariances. The default is "MethMoment", the classical method of moments, which is just the average difference-squared within bin classes. "Covariance" computes covariance rather than semivariance, but may be more biased because it subtracts off the simple mean of the response variable. "RobustMedian" and "RobustMean" are robust estimators proposed by Cressie and Hawkins (1980). If v is a vector of all pairwise square-roots of absolute differences within a bin class, then RobustMedian computes median(v)^4/.457. "RobustMean" computes mean(v)^4/(.457 + .494/length(v)).

Details

Given a SpatialStreamNetwork-class object, the function creates a list of hydrologic distances and Euclidean distances and empirical semivariogram values for both, along with number of pairs of points in each bin, for flow-connected and flow-unconnected sites when considering the hydrologic distance. Flow-connected locations lie on the same stream network (share a common downstream junction) and water flows from one location to the other. Flow-unconnected locations also lie on the same stream network, but do not share flow. The output is a list, containing an object of class Torgegram, an object of class EmpiricalSemivariogram and a list containing the distance-matrices.

Value

A list with the following fields:

EmpSemiVar
: object of class EmpiricalSemivariogram.

Torg
: object of class Torgegram.

distMatrices
: a vector of matrices, containing the flow-connection binary matrix, the hydrologic distance matrix and, not necessarily, the Euclidean distance matrix. These matrices consider the relationships between observed points.

References

Peterson, E.E. and Ver Hoef, J.M. (2010) A mixed-model moving-average approach to geostatistical modeling in stream networks. Ecology 91(3), 644–651.

Ver Hoef, J.M. and Peterson, E.E. (2010) A moving average approach for spatial statistical models of stream networks (with discussion). Journal of the American Statistical Association 105, 6–18.

`get_plots` *Plot the Torgegram and the Empirical Semivariogram for a Spatial Stream Network object*

Description

Given a SpatialStreamNetwork-class object, computes the Torgegram (empirical semivariogram from the data based on hydrologic distance) and the empirical semivariogram from the data based on the Euclidean distance. It provides the plots, useful for a complete spatial analysis of a stream network.

Usage

```
get_plots(ssn, ResponseName, Euclidean = FALSE, maxlag_Torg = NULL,
  nlag_Torg = 6, inc_Torg = 0, nlagcutoff_Torg = 15,
  maxlag_EmpVar = 1e+32, nlag_EmpVar = 20, inc_EmpVar = 0,
  nlagcutoff_EmpVar = 1, directions = c(0, 45, 90, 135),
  tolerance = 22.5, EmpVarMeth = "MethMoment")
```

Arguments

`ssn`	a SpatialStreamNetwork-class object.
`ResponseName`	the name of the response variable.
`Euclidean`	If `TRUE` the Empirical Semivariogram based on Euclidean distances is also plotted.
`maxlag_Torg`	the maximum lag distance to consider when binning pairs of locations by the hydrologic distance that separates them. The default is the median distance between all pairs of locations.
`nlag_Torg`	the number of lag bins to create for computing the Torgegram. The hydrologic distance between endpoints that define a bin will have equal lengths for all bins. The bin sizes are then determined from the minimum lag in the data, and the specification of maxlag_Torg.
`inc_Torg`	the bin hydrologic distance between endpoints. It is possible to specify the bin distance rather than nlag_Torg. In this case, the number of bins is determined by the bin distance and the hydrologic distance between the mininum and maximum (maxlag_Torg) lag in the data.
`nlagcutoff_Torg`	the minimum number of pairs needed to estimate the semivariance for a bin in the Torgegram computation. If the sample sizes is less than this value, the semivariance for the bin is not calculated.
`maxlag_EmpVar`	the maximum lag distance to consider when binning pairs of locations by the Euclidean distance that separates them. If the specified maxlag is larger than the maximum Euclidean distance among pairs of points, then maxlag_EmpVar is set to the maximum distance among pairs. If inc_EmpVar is greater than 0, then maxlag_EmpVar is disregarded.
`nlag_EmpVar`	the number of lag bins to create for computing the Empirical Semivariogram, by direction if directions are specified. The distance between endpoints that define a bin will have equal lengths for all bins. The bin sizes are then determined from the minimum lag in the data, and the specification of maxlag_EmpVar.

`inc_EmpVar` the Euclidean distance increment for each bin class. Default is 0, in which case maxlag and nclasses determine the distance increments.

`nlagcutoff_EmpVar`
the minimum number of pairs needed to estimate the semivariance for a bin in the Empirical Semivariogram. If the sample size is less than this value, the semivariance for the bin is not calculated.

`directions` directions in degrees clockwise from north that allow lag binning to be directional. Default is `c(0, 45, 90, 135)`. Values should be between 0 and 180, as there is radial symmetry in orientation between two points.

`tolerance` the angle on either side of the directions to determine if a pair of points falls in that direction class. Note, a pair of points may be in more than one lag bin if tolerances for different directions overlap.

`EmpVarMeth` method for computing semivariances. The default is "MethMoment", the classical method of moments, which is just the average difference-squared within bin classes. "Covariance" computes covariance rather than semivariance, but may be more biased because it subtracts off the simple mean of the response variable. "RobustMedian" and "RobustMean" are robust estimators proposed by Cressie and Hawkins (1980). If v is a vector of all pairwise square-roots of absolute differences within a bin class, then RobustMedian computes median(v)^4/.457. "RobustMean" computes mean(v)^4/(.457 + .494/length(v)).

Value

A list of distance matrices, containing the flow-connection binary matrix, the hydrologic distance matrix and, not necessarily, the Euclidean distance matrix. These matrices consider the relationships between observed points..

References

Peterson, E.E. and Ver Hoef, J.M. (2010) A mixed-model moving-average approach to geostatistical modeling in stream networks. Ecology 91(3), 644–651.

Ver Hoef, J.M. and Peterson, E.E. (2010) A moving average approach for spatial statistical models of stream networks (with discussion). Journal of the American Statistical Association 105, 6–18.

`get_SSN_model` *Fit a variance component model for a Spatial Stream Network object*

Description

Given a SpatialStreamNetwork-class object, fits a linear model using a variance component approach.

Usage

```
get_SSN_model(ssn, varNames, weightVar, CorModels, useNugget = TRUE,
  singleNet = NULL, matrices = NULL, bounds = NULL,
  useCholeskyDec = FALSE)
```

Arguments

`ssn`	a SpatialStreamNetwork-class object.
`varNames`	a vector of strings, the names of the variables used in the model.
`weightVar`	a string indicating the name of the variable to compute the spatial weights.
`CorModels`	a vector of strings, the names of the covariance models.
`useNugget`	If `FALSE` the nugget effect is not included in the model. Default to `TRUE`.
`singleNet`	an interger, indicating the network ID that is to be analysed. Default to `NULL`, so that the analysis is carried on the entire dataset.
`matrices`	a vector of matrices, containing the flow-connection binary matrix, the hydrologic distance matrix and, not necessarily, the Euclidean distance matrix, returned by the function get_plots. These matrices consider the relationships between observed points.
`bounds`	a vector of doubles, representing the bounds for the parsills of the models considered. If a bound is required, all the models should have one. The highest can be set at 1e+04.
`useCholeskyDec`	If `TRUE` the Cholesky decomposition for inverting positive definite matrices is always used in the optimization algorithm (quickest version, at the expense of accuracy). Default to `FALSE`.

Details

This function works on objects of SpatialStreamNetwork-class to fit generalized linear models with spatially autocorrelated errors using normal likelihood methods.

Value

A list with the following fields:

`modelParam`	vector of the parameters values of the fitted model.
`modelBeta`	vector of the beta values of the fitted model.
`modelCovariance`	covariance matrix of the fitted model.

References

Peterson, E.E. and Ver Hoef, J.M. (2010) A mixed-model moving-average approach to geostatistical modeling in stream networks. Ecology 91(3), 644–651.

Ver Hoef, J.M. and Peterson, E.E. (2010) A moving average approach for spatial statistical models of stream networks (with discussion). Journal of the American Statistical Association 105, 6–18.

`get_SSN_model_kriging` *Fit a variance component model for a Spatial Stream Network object and make prediction*

Description

Given a SpatialStreamNetwork-class object, fits a linear model using a variance component approach. Then it makes predictions using universal kriging.

Usage

```
get_SSN_model_kriging(ssn, varNames, weightVar, predpts, CorModels,
  useNugget = TRUE, singleNet = NULL, matrices = NULL,
  bounds = NULL, useCholeskyDec = FALSE)
```

Arguments

`ssn`	a SpatialStreamNetwork-class object.
`varNames`	a vector of strings, the names of the variables used in the model.
`weightVar`	a string indicating the name of the variable to compute the spatial weights.
`predpts`	a string indicating the name of the group of points in which make predictions.
`CorModels`	a vector of strings, the names of the covariance models.
`useNugget`	If `FALSE` the nugget effect is not included in the model. Default to `TRUE`.
`singleNet`	an interger, indicating the network ID that is to be analysed. Default to `NULL`, so that the analysis is carried on the entire dataset.
`matrices`	a vector of matrices, containing the flow-connection binary matrix, the hydrologic distance matrix and, not necessarily, the Euclidean distance matrix, returned by the function get_plots. These matrices consider the relationships between observed points.
`bounds`	a vector of doubles, representing the bounds for the parsills of the models considered. If a bound is required, all the models should have one. The highest can be set at 1e+04.
`useCholeskyDec`	If `TRUE` the Cholesky decomposition for inverting positive definite matrices is always used in the optimization algorithm (quickest version, at the expense of accuracy). Default to `FALSE`.

Details

This function works on objects of SpatialStreamNetwork-class to fit generalized linear models with spatially autocorrelated errors using normal likelihood methods. Then it calculates prediction values and kriging variance for prediction sites.

Value

A list with the following fields:

`ssn.object`	updated SpatialStreamNetwork-class object, with predicted values included.
`modelParam`	vector of the parameters values of the fitted model.
`modelBeta`	vector of the beta values of the fitted model.

References

Peterson, E.E. and Ver Hoef, J.M. (2010) A mixed-model moving-average approach to geostatistical modeling in stream networks. Ecology 91(3), 644–651.

Ver Hoef, J.M. and Peterson, E.E. (2010) A moving average approach for spatial statistical models of stream networks (with discussion). Journal of the American Statistical Association 105, 6–18.

`plot.predictions` *Plot the predicted values along a Spatial Stream Network object*

Description

Plot the predicted values along a Spatial Stream Network object

Usage

```
plot.predictions(ssn.object, VariableName = NULL, predpts = NULL,
  VarPlot = "Both", color.palette = rainbow(nclasses, start = 0.66, end
  = 0.99), nclasses = 10, breaktype = "quantile", dec.dig = 2,
  SEcex.min = 0.5, SEcex.max = 2, brks = NULL, add = FALSE, ...)
```

Arguments

`ssn.object`	a SpatialStreamNetwork-class object.
`VariableName`	name of variable to be plotted
`VarPlot`	a character argument that must be one of "Both", "Predictions", or "Standard Errors". Default is "Both", which colors predictions by their values and makes their size inversely proportional to the prediction standard errors.
`color.palette`	a color palette for plotting points. The default is rainbow(nclasses, start = .66, end = .99). The number of colors should equal to the number of classes. See `palette` for many ways to create palettes.
`nclasses`	the number of classes for coloring the predictions (or standard errors) according to their value. The default is 10. If brks = c(...) is specified, then nclasses is automatically set to the number of breaks + 1.
`breaktype`	the method for breaking the predictions (or standard errors) into classes for coloring while plotting. A character argument that must be one of "quantile" (default), "even", or "user".
`dec.dig`	the number of decimal places to print in the legend. Default is 2.
`SEcex.min`	if VarPlot = "both", the minimum cex value when making point sizes is inversely proportional to the prediction standard errors. See `par` for more on cex. Also see details below. Default is 1.
`SEcex.max`	if VarPlot = "both", the maximum cex value when making point sizes inversely proportional to the prediction standard errors. See `par` for more on cex. Also see details below. Default is 3.
`brks`	if breaktype = "user", the break values must be specified here as a vector or matrix using c(...) or cbind(...). The sorted unique values are used as break points (together with the min and max of the variable being plotted if required)
`add`	Logical value indicating whether the predictions should be added to an existing plot, such as a plot of colored values for observed data. Default is FALSE.
`...`	Arguments to be passed to methods, such as graphical parameters (see `par`).

Details

The `plot.predictions` function creates a map showing color-coded predictions or prediction standard error values. When VarPlot = "Both", predictions values are colored according to breaks. The size of the points is inversely proportional to the prediction standard errors.

Value

Maps of stream networks with prediction and prediction standard error values.

References

Peterson, E.E. and Ver Hoef, J.M. (2010) A mixed-model moving-average approach to geostatistical modeling in stream networks. Ecology 91(3), 644–651.

Ver Hoef, J.M. and Peterson, E.E. (2010) A moving average approach for spatial statistical models of stream networks (with discussion). Journal of the American Statistical Association 105, 6–18.

`plot.torg` *Plotting Method for Torgegram Objects*

Description

`plot`.Torg is a generic plot function that has been adapted for Torgegram objects, which are created using the torgegram function.

Usage

```
plot.torg(x, sp.relationship = c("fc", "fu"), min.cex = 1.5,
  max.cex = 6, leg.auto = TRUE, main = "", ylab = "",
  xlab = "Stream Distance")
```

Arguments

`x`	an object of class Torgegram.
`sp.relationship`	a string or character vector representing the in-stream spatial relationship to be plotted. "fc" specifies plotting of only flow-connected, and "fu" specifies plotting of only flow-unconnected. Default is both.
`min.cex`	Minimum character expansion size of the plotting symbols. Symbols are scaled according to how many pairs of points went into computing each bin of the semivariogram. The bin with the smallest sample size will be be plotted with this symbol size. The default is 1.5.
`max.cex`	Maximum character expansion size of the plotting symbols. Symbols are scaled according to how many pairs of points went into computing each bin of the semivariogram. The bin with the largest sample size will be be plotted with this symbol size. The default is 6.
`leg.auto`	Logical, default to TRUE. Include a legend.
`main`	Title for plot.
`ylab`	Label for y-axis.
`xlab`	Label for x-axis.

Details

This function provides the plot of the empirical semivariogram based on stream distance. The results are presented separately for flow-connected and flow-unconnected sites.

Value

Plot of empirical semivariogram values.

References

Peterson, E.E. and Ver Hoef, J.M. (2010) A mixed-model moving-average approach to geostatistical modeling in stream networks. Ecology 91(3), 644–651.

Ver Hoef, J.M. and Peterson, E.E. (2010) A moving average approach for spatial statistical models of stream networks (with discussion). Journal of the American Statistical Association 105, 6–18.

`torgegram` *Compute the Torgegram for a Spatial Stream Network object*

Description

`torgegram` computes the empirical semivariogram from the data based on hydrologic distance. The results are presented separately for flow-connected and flow-unconnected sites.

Usage

```
torgegram(ssn, ResponseName, maxlag = NULL, nlag = 6, inc = 0,
  nlagcutoff = 15, EmpVarMeth = "MethMoment")
```

Arguments

`ssn`	a SpatialStreamNetwork-class object.
`ResponseName`	the name of the response variable.
`maxlag`	the maximum lag distance to consider when binning pairs of locations by the hydrologic distance that separates them. The default is the median distance between all pairs of locations.
`nlag`	the number of lag bins to create for computing the Torgegram. The hydrologic distance between endpoints that define a bin will have equal lengths for all bins. The bin sizes are then determined from the minimum lag in the data, and the specification of maxlag_Torg.
`inc`	the bin hydrologic distance between endpoints. It is possible to specify the bin distance rather than nlag_Torg. In this case, the number of bins is determined by the bin distance and the hydrologic distance between the mininum and maximum (maxlag_Torg) lag in the data.
`nlagcutoff`	the minimum number of pairs needed to estimate the semivariance for a bin in the Torgegram computation. If the sample sizes is less than this value, the semivariance for the bin is not calculated.

`EmpVarMeth` method for computing semivariances. The default is "MethMoment", the classical method of moments, which is just the average difference-squared within bin classes. "Covariance" computes covariance rather than semivariance, but may be more biased because it subtracts off the simple mean of the response variable. "RobustMedian" and "RobustMean" are robust estimators proposed by Cressie and Hawkins (1980). If v is a vector of all pairwise square-roots of absolute differences within a bin class, then RobustMedian computes median(v)^4/.457. "RobustMean" computes mean(v)^4/(.457 + .494/length(v)).

Details

The `torgegram` function creates a list of hydrologic distances and empirical semivariogram values, along with number of pairs of points in each bin, for both flow-connected and flow-unconnected sites. Flow-connected locations lie on the same stream network (share a common downstream junction) and water flows from one location to the other. Flow-unconnected locations also lie on the same stream network, but do not share flow. The output is of class Torgegram.

Value

A list.

References

Peterson, E.E. and Ver Hoef, J.M. (2010) A mixed-model moving-average approach to geostatistical modeling in stream networks. Ecology 91(3), 644–651.

Ver Hoef, J.M. and Peterson, E.E. (2010) A moving average approach for spatial statistical models of stream networks (with discussion). Journal of the American Statistical Association 105, 6–18.

Index