

Package: geosptdb (via r-universe)

October 29, 2024

Type Package

Title Spatio-Temporal Radial Basis Functions with Distance-Based Methods (Optimization, Prediction and Cross Validation)

Version 1.0-1

Date 2022-05-12

Maintainer Carlos Melo <cmelo@udistrital.edu.co>

Depends R (>= 3.5.0), FD, StatMatch, fields, sp

Imports minqa, limSolve, gsl, geospt, grDevices, methods, stats, utils

Description Spatio-temporal radial basis functions (optimization, prediction and cross-validation), summary statistics from cross-validation, Adjusting distance-based linear regression model and generation of the principal coordinates of a new individual from Gower's distance.

License GPL (>= 2)

Encoding UTF-8

Author Carlos Melo [aut, cre]

(<<https://orcid.org/0000-0002-5598-1913>>), Oscar Melo [aut]

(<<https://orcid.org/0000-0002-0296-4511>>), Sandra Melo [aut]

NeedsCompilation no

Date/Publication 2022-05-12 21:30:04 UTC

Repository <https://carlosm77.r-universe.dev>

RemoteUrl <https://github.com/cran/geosptdb>

RemoteRef HEAD

RemoteSha 59f29b5165abd67687dcddfb1f3df799220e1204

Contents

geosptdb-package	2
cp.xnews	3
criteriaST.cv	4

croatia	5
croatia.grid7cp	6
croatia.temp	7
croatia2008	8
croatiadb	9
dblm	10
extractFormula	11
graph.rbfST	12
idwST	14
idwST.cv	16
idwST.cv1	17
idwST.tcv	18
rbfST	20
rbfST.cv	22
rbfST.cv1	23
rbfST.tcv	24
standardize	26

Index	27
--------------	-----------

geosptdb-package	<i>Spatio-Temporal Radial Basis Functions with Distance-Based Methods (Optimization, Prediction and Cross Validation)</i>
------------------	---

Description

Spatio-temporal: Inverse Distance Weighting (IDW) and radial basis functions; optimization, prediction, summary statistics from leave-one-out cross-validation, adjusting distance-based linear regression model and generation of the principal coordinates of a new individual from Gower's distance.

Details

Package:	geosptdb
Type:	Package
Version:	1.0-1
Date:	2022-05-12
License:	GPL (>= 2)
LazyLoad:	yes

Author(s)

Carlos Melo <cmelo@udistrital.edu.co>, Oscar Melo <oomelom@unal.edu.co>, Sandra Melo <semelom@unal.edu.co>
 Maintainer: Carlos Melo <cmelo@udistrital.edu.co>

References

- Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.
- Cuadras, CM. and Arenas, C. (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279
- Gower, J. C. (1971). *A general coefficient of similarity and some of its properties*. Biometrics 27:857-871.
- Hengl, T. (2009). *A Practical Guide to Geostatistical Mapping*, 2nd edn, University of Amsterdam, Amsterdam.
- Hengl, T., Heuvelink Gerard, B. M., Percec Tadic, M. & Pebesma, E. J. (2012). *Spatio-temporal prediction of daily temperatures using time-series of MODIS LST images*, Theoretical and Applied Climatology 107, 1-2, 265-277.
- Johnston, K., Ver, J., Krivoruchko, K., Lucas, N. 2001. *Using ArcGIS Geostatistical Analysis*. ESRI.
- Melo, C. E. (2012). *Analisis geoestadístico espacio tiempo basado en distancias y splines con aplicaciones*. PhD. Thesis. Universitat de Barcelona. 276 p. [\[link\]](#)

See Also

[rbfST](#), [graph.rbfST](#), [cp.xnews](#), [croatiadb](#)

cp.xnews	<i>Generate the principal coordinates of a new individual from Gower's distance.</i>
----------	--

Description

Function for generates a numeric matrix with principal coordinates of a new individual then you could obtain distances from this matrix and you can do a prediction using a Gower's result (1971) and Cuadras & Arenas (1990) which relates the squared distances vector with the principal coordinates vector associated to the new individual.

Usage

```
cp.xnews(newdata, eigenvalues, data, trend, ...)
```

Arguments

newdata	data frame values of new individual.
eigenvalues	the n eigenvalues computed during the scaling process (see cmdscale)
data	matrix or data frame containing the explanatory variables. These variables can be numeric, ordered, or factor, the symmetric or asymmetric binary variables should be numeric and only contain 0 and 1 character variables will be converted to factor. NAs are tolerated. With these variables the principal coordinates are built which become the regressors in the linear model.

trend matrix $n \times k$ of the k most statistically significant principal coordinates (5%) with the response variable, obtained from the matrix or data frame containing explanatory variables.

... further parameters to be passed to the `gower.dist` function (see `gower.dist`).

Value

Returns a numeric matrix with principal coordinates of the new individual.

References

Cuadras, CM. and Arenas, C. (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279

Gower, J. C. (1971). *A general coefficient of similarity and some of its properties*. Biometrics 27:857-871.

Melo, C. E. (2012). *Análisis geoestadístico espacio tiempo basado en distancias y splines con aplicaciones*. PhD. Thesis. Universitat de Barcelona. 276 p. [\[link\]](#)

See Also

[dblm](#), [rbfST](#)

Examples

```
## Not run:
data(croatia.temp)
data(croatiadb)
# prediction case: one point
point <- data.frame(670863,5043464,5,170,200,15.7,3)
names(point) <- c("x","y","t","dem","dsea","twi","est")

croatia.temp[,7] <- as.factor(croatia.temp[,7])
dblm1 <- dblm(data=croatia.temp,y=croatiadb$MTEMP)
newdata1 <- t(cp.xnews(newdata=point,eigenvalues=dblm1$ev, data=croatia.temp,
                      trend=dblm1$cp))
colnames(newdata1) <- c("X1","X2","X3","X4","X5","X6","X7","X8","X9","X10")

## End(Not run)
```

criteriaST.cv

Cross-validation summaries

Description

Generate a data frame of statistical values associated with cross-validation

Usage

```
criteriaST.cv(m.cv)
```

Arguments

`m.cv` data frame containing: prediction columns, prediction variance of cross-validation data points, observed values, residuals, zscore (residual divided by kriging standard error), and fold. If the `rbfST.tcv` function is used, the prediction variance, zscore (residual divided by standard error) will have NA's, coordinates data and time.

Value

data frame containing: mean prediction errors (MPE), average kriging standard error (AKSE), root-mean-square prediction errors (RMSPE), mean standardized prediction errors (MSPE), root-mean-square standardized prediction errors (RMSSPE), mean absolute percentage prediction errors (MAPPE), coefficient of correlation of the prediction errors (CCPE), coefficient of determination (R2) and squared coefficient of correlation of the prediction errors (pseudoR2)

Examples

```
# leave-one-out cross validation:
data(croatiadb)
coordinates(croatiadb) <- ~x+y

# inverse multiquadratic function, predefined eta and rho
tempm <- rbfST.tcv(MTEMP~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10, croatiadb, eta=0.0108,
                  rho=0.00004, n.neigh=25, func="IM")
criteriaST.cv(tempm)
```

croatia

Map Croatia.

Description

Map Croatia. Spatial reference system: UTM 33N

Usage

```
data(croatia)
```

Format

The format is: Formal class 'SpatialPolygonsDataFrame' [package "sp"]

References

Hengl, T. (2009). *A Practical Guide to Geostatistical Mapping*, 2nd edn, University of Amsterdam, Amsterdam.

Examples

```
data(croatia)
pts <- spsample(croatia, n=25000, type="regular")
plot(pts)
```

croatia.grid7cp

Principal coordinates of a pixelated size 4994 in Croatia.

Description

data frame 4994×13 of spatio-temporal coordinates and principal coordinates associated with a pixelated size 4994 in Croatia. Spatial reference system: UTM 33N.

Usage

```
data(croatia.grid7cp)
```

References

Hengl, T. (2009). *A Practical Guide to Geostatistical Mapping*, 2nd edn, University of Amsterdam, Amsterdam.

Hengl, T., Heuvelink Gerard, B. M., Percec Tadic, M. & Pebesma, E. J. (2012). *Spatio-temporal prediction of daily temperatures using time-series of MODIS LST images*, Theoretical and Applied Climatology 107, 1-2, 265-277.

Melo, C. E. (2012). *Analisis geoestadístico espacio tiempo basado en distancias y splines con aplicaciones*. PhD. Thesis. Universitat de Barcelona. 276 p. [\[link\]](#)

See Also

[croatia.temp](#)

Examples

```
data(croatia.grid7cp)
plot(croatia.grid7cp[,1:2])
```

`croatia.temp`*Data climatic stations in Croatia.*

Description

Information of 142 climatic stations in Croatia in 2008, with topographical static predictors (Digital Elevation Model, (DEM, in meters), topographically weighted distances from the coast line (DSEA, in km), topographic wetness index (TWI))

Usage

```
data(croatia.temp)
```

Format

A data frame with 1752 observations on the following 7 variables:

`x` a numeric vector; x-coordinate; Spatial reference system: UTM 33N

`y` a numeric vector; y-coordinate; Spatial reference system: UTM 33N

`t` a numeric vector; t-coordinate (1-12 for the months from January to December)

`dem` a numeric vector, Digital Elevation Model (DEM, in meters)

`dsea` a numeric vector with topographically weighted distances from the coast line (DSEA, in km)

`twi` a numeric vector with topographic wetness index

`est` a numeric vector with seasons (1 for January, February and March, 2 for April, May and June, 3 for July, August and September and 4 for October, November and December)

References

Hengl, T. (2009). *A Practical Guide to Geostatistical Mapping*, 2nd edn, University of Amsterdam, Amsterdam.

Hengl, T., Heuvelink Gerard, B. M., Percec Tadic, M. & Pebesma, E. J. (2012). *Spatio-temporal prediction of daily temperatures using time-series of MODIS LST images*, *Theoretical and Applied Climatology* 107, 1-2, 265-277.

Examples

```
data(croatia.temp)
summary(croatia.temp)
```

`croatia2008`*Data climatic stations in Croatia.*

Description

Information of 154 climatic stations in Croatia in 2008, with topographical static predictors (Digital Elevation Model, (DEM, in meters), topographically weighted distances from the coast line (DSEA, in km), topographic wetness index (TWI), Geographical coordinates: latitude (lat) and longitude (lon), and earth's monthly average temperature (MTEMP))

Usage

```
data(croatia.temp)
```

Format

A data frame with 1845 observations on the following 9 variables:

x a numeric vector; x-coordinate; Spatial reference system: UTM 33N

y a numeric vector; y-coordinate; Spatial reference system: UTM 33N

t a numeric vector; t-coordinate (1-12 for the months from January to December)

dem a numeric vector, Digital Elevation Model (DEM, in meters)

dsea a numeric vector with topographically weighted distances from the coast line (DSEA, in km)

twi a numeric vector with topographic wetness index

Lat a numeric vector; latitude-coordinate; Spatial reference system: UTM 33N

Lon a numeric vector; longitude-coordinate; Spatial reference system: UTM 33N

MTEMP a numeric vector with earth's monthly average temperature

References

Hengl, T. (2009). *A Practical Guide to Geostatistical Mapping*, 2nd edn, University of Amsterdam, Amsterdam.

Hengl, T., Heuvelink Gerard, B. M., Percec Tadic, M. & Pebesma, E. J. (2012). *Spatio-temporal prediction of daily temperatures using time-series of MODIS LST images*, *Theoretical and Applied Climatology* 107, 1-2, 265-277.

Examples

```
data(croatia2008)
summary(croatia2008)
```

croatiadb	<i>principal coordinates associated with data climatic stations in Croatia 2008.</i>
-----------	--

Description

data frame 1752×14 of spatio-temporal coordinates, earth's average temperature monthly and 10 principal coordinates associated with data climatic stations in Croatia 2008.

Usage

```
data(croatiadb)
```

Format

The format is: Formal class 'data.frame' [package "base"]

References

Hengl, T. (2009). *A Practical Guide to Geostatistical Mapping*, 2nd edn, University of Amsterdam, Amsterdam.

Hengl, T., Heuvelink Gerard, B. M., Percec Tadic, M. & Pebesma, E. J. (2012). *Spatio-temporal prediction of daily temperatures using time-series of MODIS LST images*, Theoretical and Applied Climatology 107, 1-2, 265-277.

Melo, C. E. (2012). *Analisis geoestadístico espacio tiempo basado en distancias y splines con aplicaciones*. PhD. Thesis. Universitat de Barcelona. 276 p. [\[link\]](#)

See Also

[croatia.grid7cp](#), [croatia.temp](#)

Examples

```
data(croatiadb)
str(croatiadb)
names(croatiadb)
```

 dblm

Adjusting distance-based linear regression model

Description

`dblm` is a linear model variety where explanatory information is coded as distances among individuals so these distances can also be computed from observed explanatory variables (a mix of continuous, qualitative explanatory variables or from more general quantities). The response is a continuous variable as in the classic linear model.

`lm` is used internally to adjust a distance-based linear regression model. The method considers the Gower's distance for mixed covariates (numeric, ordered, or factor), for explanation on the meaning of distance-based linear regression model and distance of Gower see the bibliography references below.

Usage

```
dblm(data, y, sc, ev.min, ...)
```

Arguments

<code>data</code>	matrix or data frame containing the explanatory variables. These variables can be numeric, ordered, or factor. Symmetric or asymmetric binary variables should be numeric and only contain 0 and 1. character variables will be converted to factor. NAs are tolerated. With these variables are built, the principal coordinates which later become the regressors in the linear model.
<code>y</code>	the response variable used to fit the model
<code>sc</code>	the value of the correlation squared to select the principal coordinates more related to the response variable. The default value is 0.003.
<code>ev.min</code>	the minimum value to select the eigenvalues. These eigenvalues must be positive, the default value is 0.007
<code>...</code>	further parameters to be passed to the <code>gowdis</code> function (see <code>gowdis</code>) of low level.

Details

The `dblm` model builds; principal coordinates matrix, eigenvalues, and a linear regression model. `gowdis` function used in `dblm` compute the Gower (1971) similarity coefficient exactly as described by Podani (1999), then converts it to a dissimilarity coefficient by using $D = 1 - S$. It integrates variable weights as described by Legendre and Legendre (1998).

Value

A list containing the following components:

<code>table</code>	table with eigenvalues, correlations squared, and percentages of inertia associated with the most statistically significant principal coordinates (5%) with the response variable.
--------------------	--

ev	the n eigenvalues computed during the scaling process (see <code>cmdscale</code>).
cp	the k most statistically significant principal coordinates (5%) with the response variable.
dbmodel	returns a list of summary statistics of the fitted linear model.

References

- Cuadras, CM., Arenas C. and Fortiana, J. (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.
- Cuadras, CM. and Arenas, C. (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279
- Gower, J. C. (1971). *A general coefficient of similarity and some of its properties*. Biometrics 27:857-871.
- Legendre, P. and Legendre, L. (1998). *Numerical Ecology*. 2nd English edition. Amsterdam: Elsevier.
- Melo, C. E. (2012). *Analisis geoestadístico espacio tiempo basado en distancias y splines con aplicaciones*. PhD. Thesis. Universitat de Barcelona. 276 p. [\[link\]](#)
- Podani, J. (1999). *Extending Gower's general coefficient of similarity to ordinal characters*. Taxon 48:331-340.

See Also

See function `gowdis` in the FD package.

Examples

```
# considering 10 principal coordinates (constructed from a distance-based linear
# regression model)
## Not run:
data(croatia.temp)
data(croatiadb)
croatia.temp[,7] <- as.factor(croatia.temp[,7])
dblm1 <- dblm(data=croatia.temp,y=croatiadb$MTEMP)
str(dblm1)

## End(Not run)
```

extractFormula *geospt internal function*

Description

geospt internal function

Note

This function is not meant to be called by users directly

graph.rbfST

Graph that describes the behavior of the optimized eta and rho parameters, associated with a spatio-temporal radial basis function.

Description

Function for plotting the RMSPE for several values of the smoothing parameter *eta* with the same dataset. A curve is fitted to the points, and then the optimal *eta* that provides the smallest RMSPE is determined from the curve, by the `optimize` function from the `stats` package.

Usage

```
graph.rbfST(formula, data, eta.opt, rho.opt, n.neigh, func, np, xo, eta.dmax,
rho.dmax, P.T, iter, ...)
```

Arguments

formula	formula that defines the dependent variable as a linear model of independent variables (covariates or the principal coordinates); suppose the dependent variable has name z_{st} , for a <i>rbfST</i> detrended use $z_{st} \sim 1$, for a <i>rbfST</i> with trend, suppose z_{st} is linearly dependent on x and y , use the formula $z_{st} \sim x+y$ (linear trend).
data	SpatialPointsDataFrame: should contain the spatio-temporal dependent variable, independent variables (statics and/or dynamics), spatial coordinates and the time as an integer or numerical variable.
eta.opt	logical, indicating whether the parameter <i>eta</i> should be regarded as fixed (eta.opt = FALSE) or should be estimated (eta.opt = TRUE)
rho.opt	logical, indicating whether the parameter <i>rho</i> should be regarded as fixed (rho.opt = FALSE) or should be estimated (rho.opt = TRUE)
n.neigh	number of nearest observations that should be used for a <i>rbfST</i> prediction, where nearest is defined in terms of the spatio-temporal locations
func	function to be optimized. The following radial basis function spatio-temporal model types are currently available: gaussian "GAU", exponential "EXPON", trigonometric "TRI", thin plate spline "TPS", completely regularized spline "CRS", spline with tension "ST", inverse multiquadratic "IM", and multiquadratic "M", are currently available
np	number of points, where the radial basis function spatio-temporal is calculated
xo	starting point for searching the optimum. Defaults to <code>c(0.5, 0.5)</code> , <i>eta</i> and <i>rho</i> respectively. Use this statement only if <i>eta</i> and <i>rho</i> are equal to TRUE.
eta.dmax	maximum value of the range of the <i>eta</i> parameter that will be evaluated by the <code>optimize</code> function.
rho.dmax	maximum value of the range of the <i>rho</i> parameter that will be evaluated by the <code>optimize</code> function.
P.T	logical. Print table (TRUE) or not (FALSE). Default P:T=NULL.

`iter` The maximum allowed number of function evaluations.

`...` further parameters to be passed to the minimization functions `optimize` or `bobyqa`, typically arguments of the type `control()` which control the behavior of the minimization algorithm. See documentation about the selected minimization function for further details.

Value

Returns a graph that describes the behavior of the optimized *eta* or *rho* parameters and a table of values associated with the graph including optimal smoothing *eta* or *rho* parameters. If both *eta* and *rho* are FALSE simultaneously then the function returns a list with the best value obtained from the combinations smoothing *eta* and *rho* parameters and a lattice plot of class "trellis" with RMSPE pixel values associated with combinations of *eta* and *rho* parameters. Finally, if both *eta* and *rho* are TRUE, the function will return a list with the best combination of values of the smoothing *eta* or *rho* parameters and the RMSPE associated with these.

References

Johnston, K., Ver, J., Krivoruchko, K., Lucas, N. (2001). *Using ArcGIS Geostatistical Analysis*. ESRI.

Melo, C. E. (2012). *Analisis geoestadístico espacio tiempo basado en distancias y splines con aplicaciones*. PhD. Thesis. Universitat de Barcelona. 276 p. [\[link\]](#)

See Also

[rbfST](#), [rbfST.cv](#)

Examples

```
## Not run:
data(croatiadb)
coordinates(croatiadb)<--x+y
# optimizing eta
graph.rbfST(MTEMP~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10, data=croatiadb, eta.opt=TRUE,
            rho.opt=FALSE, n.neigh=30, func="TPS", np=40, eta.dmax=2, P.T=TRUE)
# optimizing rho
graph.rbfST(MTEMP~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10, data=croatiadb, eta.opt=FALSE,
            rho.opt=TRUE, n.neigh=30, func="M", np=20, rho.dmax=2, P.T=TRUE)
# optimizing eta and rho
tps.lo <- graph.rbfST(MTEMP~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10, data=croatiadb,
                    eta.opt=TRUE, rho.opt=TRUE, n.neigh=25, func="TPS", eta.dmax=0.2,
                    rho.dmax=0.2, xo=c(0.1,0.1), iter=50)
tps.lo # best combination of eta and rho obtained
# lattice of RMSPE values associated with a range of eta and rho, without optimization
tps.la <- graph.rbfST(MTEMP~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10, data=croatiadb,
                    eta.opt=FALSE, rho.opt=FALSE, n.neigh=30, func="TPS", np=10, eta.dmax=0.2,
                    rho.dmax=0.2)
tps.l$table # best combination of eta and rho obtained
tps.l$spplot # lattice of RMSPE

## End(Not run)
```

idwST	<i>Inverse Distance Weighting (IDW) function for spatio-temporal prediction.</i>
-------	--

Description

This function performs spatio-temporal interpolation. Here *idwST* is in a local neighborhood. This interpolation method considers the value of a point can be obtained from the weighted sum of values of the regionalized variable of closest neighbors. The general formula for the IDW is given by:

$$\hat{z}_0(st) = \sum_{i=1}^n \lambda_i z_i(st)$$

The expression for determining the weights is:

$$\lambda_i = \frac{d_{i0}^{-p}}{\sum_{i=1}^n d_{i0}^{-p}}$$

The weight is controlled by a factor p with each increment of the distance, d_{i0} is the distance between the prediction position and each of the measured positions.

The expression d_{i0} can be obtained by:

$$d_{i0} = \sqrt{(x_i - x_0)^2 + (y_i - y_0)^2 + C \cdot (t_i - t_0)^2}$$

x , y and t correspond to the spatio-temporal coordinates, p (factor.p) and C factors defined below.

Usage

```
idwST(formula, data, newdata, n.neigh, C, factor.p, progress)
```

Arguments

formula	formula that defines a detrended linear model, use $z_{st} \sim 1$.
data	SpatialPointsDataFrame: should contain the spatio-temporal dependent variable, independent variables (statics and/or dynamics), spatial coordinates and the time as an integer or numerical variable.
newdata	data frame or spatial object with prediction/simulation spatio-temporal locations; should contain attribute columns with the independent variables (if present) and (if locations is a formula) the coordinates and time with names, as defined in locations where you want to generate new predictions
n.neigh	number of nearest observations that should be used for a <i>idwST</i> prediction, where nearest is defined in terms of the spatio-temporal locations
C	numeric; associated to time factor, we recommend using the parameter found by minimizing the root-mean-square prediction errors using cross-validation. Using <i>idwST.cv</i> and optimize
factor.p	numeric; specify the inverse distance weighting power (p is the exponent that influences the weighting or optimal smoothing parameter)
progress	whether a progress bar shall be printed for spatio-temporal inverse-distance weighted function; default=TRUE

Details

idwST function generates individual spatio-temporal predictions from IDW spatio-temporal interpolation. IDW is a type of deterministic method for interpolation, the assigned values to unknown points are calculated with a weighted average of the values available at the known points.

Value

Attributes columns contain coordinates, time, predictions, and the variance column contains NA's

References

Li L, Losser T, Yorke C, Piltner R. (2014). *Fast inverse distance weighting-based spatiotemporal interpolation: a web-based application of interpolating daily fine particulate matter PM_{2.5} in the contiguous U.S. using parallel programming and k-d tree*. Int. J. Environ. Res. Public Health, 11: 9101-9141. [\[link\]](#)

Examples

```
# Loading Croatia data
data(croatia2008)
coordinates(croatia2008) <- ~x+y

# prediction case: one point
point <- data.frame(670863,5043464,5)
names(point) <- c("x","y","t")

coordinates(point) <- ~x+y
idwST(MTEMP~1, data=croatia2008, newdata=point, n.neigh=60, C=1, factor.p=2)

## Not run:
# prediction case: a grid of points Croatia (year 2008)
data(croatia)
points <- spsample(croatia, n=5000, type="regular")

data(croatia2008)
coordinates(croatia2008)<~x+y

GridsT <- vector(mode = "list", length = 12)

for(i in 1:12){
  GridsT[[i]] <- data.frame(points@coords,i)
  names(GridsT[[i]]) <- c("x","y","t")
}

idw.croatia <- data.frame(matrix(NA, ncol = 14, nrow=nrow(GridsT[[1]])))
pb <- txtProgressBar(min = 0, max = 12, char = "=", style = 3)
for(i in 1:12){
  coordinates(GridsT[[i]]) <- c("x", "y")
  idw.croatia[,i+2] <- idwST(MTEMP~1, croatia2008, newdata=GridsT[[i]], n.neigh=10, C=1,
    factor.p=2, progress=FALSE)[,4]
  setTxtProgressBar(pb, i)
}
```

```

}
close(pb)

idw.croatia[,1:2] <- GridsT[[1]]@coords
nam <- paste(c("ENE", "FEB", "MAR", "ABR", "MAY", "JUN", "JUL", "AGO", "SEP", "OCT", "NOV", "DIC"),
            2008, sep="")
names(idw.croatia) <- c("x", "y", nam)

coordinates(idw.croatia) <- c("x", "y")
gridded(idw.croatia) <- TRUE

# show prediction map
pal2 <- colorRampPalette(c("blue3", "wheat1", "red3"))

p1 <- splot(idw.croatia[,1:12], cuts=30, col.regions=pal2(35), colorkey=F,
           scales = list(draw =T, cex=0.6, abbreviate=TRUE, minlength=1), pch=0.3,
           cex.lab=0.3, cex.title=0.3, auto.key = F, main = "Earth's average
           temperature IDW map 2008", key.space=list(space="right", cex=0.8))

split.screen( rbind(c(0, 1,0,1), c(1,1,0,1)))
split.screen(c(1,2), screen=1)-> ind
screen( ind[1])
p1
screen( ind[2])
image.plot(legend.only=TRUE, legend.width=0.5, col=pal2(100),
           smallplot=c(0.7,0.75, 0.3,0.7), zlim=c(min(idw.croatia@data),
           max(idw.croatia@data)), axis.args = list(cex.axis = 0.7))
close.screen( all=TRUE)

## End(Not run)

```

idwST.cv

*IDW spatio-temporal leave-one-out cross validation***Description**

Generate the RMSPE value which is given by Inverse Distance Weighting (IDW) interpolation.

Usage

```
idwST.cv(formula, data, n.neigh, C, factor.p, progress)
```

Arguments

formula	formula that defines a detrended linear model, use $z_{st} \sim 1$.
data	SpatialPointsDataFrame: should contain the spatio-temporal dependent variable, independent variables (statics and/or dynamics), spatial coordinates and the time as an integer or numerical variable.

n.neigh	number of nearest observations that should be used for a <i>rbf.st</i> prediction, where nearest is defined in terms of the spatio-temporal locations
C	numeric; associated to time factor, we recommend using the parameter found by minimizing the root-mean-square prediction errors using cross-validation. Using <i>idwST.cv</i> and optimize
factor.p	numeric; specify the inverse distance weighting power (p is the exponent that influences the weighting or optimal smoothing parameter)
progress	whether a progress bar shall be printed for spatio-temporal inverse-distance weighted function; default=TRUE

Value

returns the RMSPE value

References

Melo, C. E. (2012). *Analisis geoestadístico espacio tiempo basado en distancias y splines con aplicaciones*. PhD. Thesis. Universitat de Barcelona. 276 p. [\[link\]](#)

See Also

[idwST](#), [rbfST](#)

Examples

```
## Not run:
data(croatiadb)
coordinates(croatiadb) <- ~x+y
idwST.cv(MTEMP~1, croatiadb[,1:2], n.neigh=10, C=1, factor.p=2)

## End(Not run)
```

idwST.cv1

Generate a RMSPE value, result of leave-one-out cross validation

Description

Generate the RMSPE value which is given by the radial basis function spatio-temporal with number of nearest observations *n.neigh* associated to time factor *C* and optimal smoothing parameter *factor.p*.

Usage

```
idwST.cv1(param, formula, data, n.neigh, progress)
```

Arguments

param	vector starting points (C and $factor.p$ respectively) for searching the <i>RMSPE</i> optimum.
formula	formula that defines a detrended linear model, use $z_{st} \sim 1$.
data	<code>SpatialPointsDataFrame</code> : should contain the spatio-temporal dependent variable, independent variables (statics and/or dynamics), spatial coordinates and the time as an integer or numerical variable.
n.neigh	number of nearest observations that should be used for a <i>rbf.st</i> prediction where nearest is defined in terms of the spatio-temporal locations
progress	whether a progress bar shall be printed for spatio-temporal inverse-distance weighted function; default=FALSE

Value

returns the RMSPE value

See Also

[idwST](#), [idwST.cv](#), [idwST.tcv](#)

Examples

```
require(minqa)
data(croatiadb)
coordinates(croatiadb) <- ~x+y

## Not run:
idwST.opt <- bobyqa(c(1, 2), idwST.cv1, lower=c(0,0.1), upper=c(2,4), formula=MTEMP~1,
  data=croatiadb[,1:2], n.neigh=10, progres=F, control=list(maxfun=50))

# obtained with optimal values previously estimated (33 iterations)
idwST.cv1(c(1.00538675066736,1.95853920335545), MTEMP~1, data=croatiadb[,1:2], n.neigh=10,
  progress=T)

## End(Not run)
```

idwST.tcv

table of idw spatio-temporal leave-one-out cross validation

Description

Generates a table with the results of inverse distance weighting spatio-temporal interpolation (*idwST*) from leave-one-out cross validation method.

Usage

```
idwST.tcv(formula, data, n.neigh, C, factor.p, progress)
```

Arguments

formula	formula that defines a detrended linear model, use $z_{st} \sim 1$.
data	SpatialPointsDataFrame: should contain the spatio-temporal dependent variable, independent variables (statics and/or dynamics), spatial coordinates and the time as an integer or numerical variable.
n.neigh	number of nearest observations that should be used for a <i>idwST</i> prediction where nearest is defined in terms of the spatio-temporal locations
C	numeric; associated to time factor, we recommend using the parameter found by minimizing the root-mean-square prediction errors using cross-validation. Using <i>idwST.cv</i> and optimize
factor.p	numeric; specify the inverse distance weighting power (p is the exponent that influences the weighting or optimal smoothing parameter)
progress	whether a progress bar shall be printed for spatio-temporal inverse-distance weighted function; default=TRUE

Details

Leave-one-out cross validation (LOOCV) consists of removing data, one at a time, and then trying to predict it. Next, the predicted value can be compared to the actual (observed) value to assess how well the prediction is working. The observed value is left out because *idwST* would otherwise predict the value itself.

Value

data frame contain prediction columns, observed values, residuals, the prediction variance, zscore (residual divided by standard error) which left with NA's, the fold column which is associated to cross-validation count, coordinates data and time. Prediction columns and residuals are obtained from cross-validation data points.

See Also

[idwST](#)

Examples

```
## Not run:
data(croatiadb)
coordinates(croatiadb) <- ~x+y
idw.t <- idwST.tcv(MTEMP~1, croatiadb, n.neigh=10, C=1.0054, factor.p=1.9585)
criteriaST.cv(idw.t)

## End(Not run)
```

rbfST	<i>gaussian, exponential, trigonometric, thin plate spline, inverse multiquadratic, and multiquadratic radial basis function for spatio-temporal prediction</i>
-------	---

Description

Function for spatio-temporal interpolation from radial basis function (*rbfST*), where *rbfST* is in a local neighbourhood.

exponential (EXPON)

$$\phi(\delta) = e^{-\eta\delta}, \eta > 0$$

gaussiano (GAU)

$$\phi(\delta) = e^{-\eta\delta^2}, \eta \neq 0$$

multiquadratic (M)

$$\phi(\delta) = \sqrt{\eta^2 + \delta^2}, \eta \neq 0$$

inverse multiquadratic (IM)

$$\phi(\delta) = 1/\sqrt{\eta^2 + \delta^2}, \eta \neq 0$$

thin plate spline (TPS)

$$\begin{aligned} \phi(\delta) &= (\eta \cdot \delta)^2 \log(\eta \cdot \delta), \text{ if } : \delta > 0, \eta > 0 \\ \phi(\delta) &= 0, \text{ otherwise} \end{aligned}$$

completely regularized spline (CRS)

$$\begin{aligned} \phi(\delta) &= \ln(\eta \cdot \delta/2)^2 + E_1(\eta \cdot \delta/2)^2 + C_E, \text{ if } : \delta > 0, \eta > 0 \\ \phi(\delta) &= 0, \text{ otherwise} \end{aligned}$$

where \ln is natural logarithm, $E_1(x)$ is the exponential integral function, and C_E is the Euler constant.

spline with tension (ST)

$$\begin{aligned} \phi(\delta) &= \ln(\eta \cdot \delta/2) + K_0(\eta \cdot \delta) + C_E, \text{ if } : \delta > 0 \\ \phi(\delta) &= 0, \text{ otherwise} \end{aligned}$$

where $K_0(x)$ is the modified Bessel function and C_E is the Euler constant.

Usage

rbfST(formula, data, eta, rho, newdata, n.neigh, func, progress)

Arguments

formula	formula that defines the dependent variable as a linear model of independent variables (covariates or principal coordinates); suppose the dependent variable has name z_{st} for a <i>rbfST</i> detrended use $z_{st} \sim 1$; for a <i>rbfST</i> with trend suppose z_{st} is linearly dependent on x and y , use the formula $z_{st} \sim x+y$ (linear trend).
data	SpatialPointsDataFrame: should contain the spatio-temporal dependent variable, independent variables (statics and/or dynamics), spatial coordinates and the time as an integer or numerical variable.
eta	the optimal smoothing parameter, we recommend using the parameter found by minimizing the root-mean-square prediction errors using cross-validation
rho	optimal robustness parameter, we recommend using the value obtained by minimizing the root-mean-square prediction errors with cross-validation. <i>eta</i> and <i>rho</i> parameters can be optimized simultaneously, through the bobyqa function from <code>nloptr</code> or <code>minqa</code> packages
newdata	data frame or spatial object with prediction/simulation spatio-temporal locations; should contain attribute columns with the independent variables (if present) and (if locations is a formula) the coordinates and time with names, as defined in locations where you want to generate new predictions
n.neigh	number of nearest observations that should be used for a <i>rbfST</i> prediction, where nearest is defined in terms of the spatio-temporal locations
func	spatio-temporal radial basis function; model type: "GAU", "EXPON", "TRI", "TPS", "CRS", "ST", "IM" and "M", are currently available
progress	whether a progress bar shall be printed for spatio-temporal radial basis functions; default=TRUE

Details

`rbf.st` function generates individual spatio-temporal predictions from gaussian (GAU), exponential (EXPON), trigonometric (TRI) thin plate spline (TPS), completely regularized spline (CRS), spline with tension (ST), inverse multiquadratic (IM), and multiquadratic (M) functions

Value

Attributes columns contain coordinates, time, predictions, and the variance column contains NA's

References

Melo, C. E. (2012). *Analisis geoestadístico espacio tiempo basado en distancias y splines con aplicaciones*. PhD. Thesis. Universitat de Barcelona. 276 p. [\[link\]](#)

Examples

```
## Not run:
# considering 10 principal coordinates (constructed from a distance-based regression model)
data(croatia.temp)
data(croatiadb)
```

```

# prediction case: one point
point <- data.frame(670863,5043464,5,170,200,15.7,3)
names(point) <- c("x","y","t","dem","dsea","twi","est")

croatia.temp[,7] <- as.factor(croatia.temp[,7])
dblm1 <- dblm(data=croatia.temp,y=croatiadb$MTEMP)
newdata1 <- t(cp.xnews(newdata=point,eigenvalues=dblm1$ev, data=croatia.temp,trend=dblm1$cp))
colnames(newdata1) <- c("X1","X2","X3","X4","X5","X6","X7","X8","X9","X10")
newdata1 <- data.frame(point[,1:3],newdata1)

data(croatiadb)
coordinates(croatiadb) <- ~x+y
coordinates(newdata1) <- ~x+y
rbfST(MTEMP~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10, data=croatiadb, eta=0.010076, rho=0.00004,
      newdata=newdata1, n.neigh=60, func="TPS")

# prediction case: a grid of points Croatia (month july)
data(croatia.grid7cp)
coordinates(croatia.grid7cp) <- ~x+y
rbf.t <- rbfST(MTEMP~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10, croatiadb, eta=0.01076, rho=0.00004,
              newdata=croatia.grid7cp, n.neigh=30, func="TPS")
coordinates(rbf.t) <- c("x", "y")
gridded(rbf.t) <- TRUE

# show prediction map
splot(rbf.t["var1.pred"], cuts=30, col.regions=bpy.colors(40), main = "Earth's average
      temperature TPS map\n (july month)", key.space=list(space="right", cex=0.8))

## End(Not run)

```

rbfST.cv

Leave-one-out cross validation for spatio-temporal radial basis function

Description

It generates the RMSPE value, which is given by the radial basis function with smoothing eta and robustness rho parameters.

Usage

```
rbfST.cv(formula, data, eta, rho, n.neigh, func)
```

Arguments

formula formula that defines the dependent variable as a linear model of independent variables (covariates or the principal coordinates); suppose the dependent variable has name z_{st} , for a *rbfST* detrended use $z_{st} \sim 1$, for a *rbfST* with trend, suppose z_{st} is linearly dependent on x and y, use the formula $z_{st} \sim x+y$ (linear trend).

data	SpatialPointsDataFrame: should contain the spatio-temporal dependent variable, independent variables (statics and/or dynamics), spatial coordinates and the time as an integer or numerical variable.
eta	the optimal smoothing parameter, we recommend using the parameter found by minimizing the root-mean-square prediction errors using cross-validation.
rho	optimal robustness parameter, we recommend using the value obtained by minimizing the root-mean-square prediction errors with cross-validation. <i>eta</i> and <i>rho</i> parameters can be optimized simultaneously, through the bobyqa function from <code>nloptr</code> or <code>minqa</code> packages.
n.neigh	number of nearest observations that should be used for a <i>rbfST</i> prediction, where nearest is defined in terms of the spatio-temporal locations.
func	spatio-temporal radial basis function; model type: "GAU", "EXPON", "TRI", "TPS", "CRS", "ST", "IM" and "M", are currently available

Value

returns the RMSPE value

References

Melo, C. E. (2012). *Analisis geoestadístico espacio tiempo basado en distancias y splines con aplicaciones*. PhD. Thesis. Universitat de Barcelona. 276 p. [\[link\]](#)

See Also

[rbfST](#), [graph.rbfST](#)

Examples

```
data(croatiadb)
coordinates(croatiadb) <- ~x+y
rbfST.cv(MTEMP~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10, croatiadb, eta=0.0108, rho=0.00004,
        n.neigh=25, func="TPS")
```

rbfST.cv1

RMSPE value result of leave-one-out cross validation for rbfST

Description

It generates the RMSPE value which is given by the spatio-temporal radial basis function with smoothing *eta* and robustness *rho* parameters.

Usage

```
rbfST.cv1(param, formula, data, n.neigh, func)
```

Arguments

param	vector starting points (<i>eta</i> and <i>rho</i> respectively) for searching the <i>RMSPE</i> optimum.
formula	formula that defines the dependent variable as a linear model of independent variables (covariates or the principal coordinates); suppose the dependent variable has name z_{st} , for a <i>rbfST</i> detrended use $z_{st} \sim 1$, for a <i>rbfST</i> with trend, suppose z_{st} is linearly dependent on x and y , use the formula $z_{st} \sim x+y$ (linear trend).
data	SpatialPointsDataFrame: should contain the spatio-temporal dependent variable, independent variables (statics and/or dynamics), spatial coordinates and the time as an integer or numerical variable.
n.neigh	number of nearest observations that should be used for a <i>rbfST</i> prediction, where nearest is defined in terms of the spatio-temporal locations.
func	spatio-temporal radial basis function; model type: "GAU", "EXPON", "TRI", "TPS", "CRS", "ST", "IM" and "M", are currently available

Value

returns the *RMSPE* value

See Also

[rbfST](#), [rbfST.cv](#), [graph.rbfST](#)

Examples

```
require(minqa)
data(croatiadb)
coordinates(croatiadb) <- ~x+y

## Not run:
rbf.im <- bobyqa(c(0.5, 0.5), rbfST.cv1, lower=c(1e-05,0), upper=c(2,2),
               formula=MTEMP~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10, data=croatiadb, n.neigh=25,
               func="IM", control=list(maxfun=50))

## End(Not run)

# obtained with the optimal values previously estimated
rbfST.cv1(c(0.847050095690357,0.104157855356128), MTEMP~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10,
          croatiadb, n.neigh=25, func="IM")
```


Description

It generates a table with the results of the evaluation of radial basis functions spatio-temporal (*rbfST*): gaussian (GAU), exponential (EXPON), trigonometric (TRI), thin plate spline (TPS), completely regularized spline (CRS), spline with tension (ST), inverse multiquadratic (IM), and multiquadratic (M) from the leave-one-out cross validation method.

Usage

```
rbfST.tcv(formula, data, eta, rho, n.neigh, func, progress)
```

Arguments

formula	formula that defines the dependent variable as a linear model of independent variables (covariates or the principal coordinates); suppose the dependent variable has name z_{st} , for a <i>rbf.st</i> detrended use $z_{st} \sim 1$, for a <i>rbf.st</i> with trend, suppose z_{st} is linearly dependent on x and y, use the formula $z_{st} \sim x+y$ (linear trend).
data	SpatialPointsDataFrame: should contain the spatio-temporal dependent variable, independent variables (statics and/or dynamics), spatial coordinates and the time as an integer or numerical variable.
eta	the optimal smoothing parameter; we recommend using the parameter found by minimizing the root-mean-square prediction errors using cross-validation
rho	optimal robustness parameter, we recommend using the value obtained by minimizing the root-mean-square prediction errors with cross-validation. <i>eta</i> and <i>rho</i> parameters can be optimized simultaneously, through the bobyqa function from <code>nloptr</code> or <code>minqa</code> packages
n.neigh	number of nearest observations that should be used for a <i>rbfST</i> prediction, where nearest is defined in terms of the spatio-temporal locations.
func	spatio-temporal radial basis function; model type: "GAU", "EXPON", "TRI", "TPS", "CRS", "ST", "IM" and "M", are currently available
progress	whether a progress bar shall be printed for spatio-temporal radial basis functions; default=TRUE

Details

Leave-one-out cross validation (LOOCV) visits a data point, predicts the value at that location by leaving out the observed value, and proceeds with the next data point. The observed value is left out because *rbf.st* would otherwise predict the value itself.

Value

data frame contain prediction columns, observed values, residuals, the prediction variance, zscore (residual divided by standard error) which left with NA's, the fold column which is associated to cross-validation count, coordinates data and time. Prediction columns and residuals are obtained from cross-validation data points.

See Also

[rbfST](#)

Examples

```
data(croatiadb)
coordinates(croatiadb) <- ~x+y
rbfST.tcv(MTEMP~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10, croatiadb, eta=0.0108, rho=0.00004,
          n.neigh=30, func="TPS")
```

standardize

standardize internal function

Description

standardize internal function

Note

This function is not meant to be called by users directly

Index

- * **datasets**
 - [croatia](#), [5](#)
 - [croatia.grid7cp](#), [6](#)
 - [croatia.temp](#), [7](#)
 - [croatia2008](#), [8](#)
 - [croatiadb](#), [9](#)
 - * **package**
 - [geosptdb-package](#), [2](#)
 - * **principal coordinates**
 - [dblm](#), [10](#)
 - * **spatial**
 - [cp.xnews](#), [3](#)
 - [criteriaST.cv](#), [4](#)
 - [extractFormula](#), [11](#)
 - [graph.rbfST](#), [12](#)
 - [idwST](#), [14](#)
 - [idwST.cv](#), [16](#)
 - [idwST.cv1](#), [17](#)
 - [idwST.tcv](#), [18](#)
 - [rbfST](#), [20](#)
 - [rbfST.cv](#), [22](#)
 - [rbfST.cv1](#), [23](#)
 - [rbfST.tcv](#), [24](#)
 - [standardize](#), [26](#)
 - * **spatio-temporal**
 - [geosptdb-package](#), [2](#)
- [bobyqa](#), [13](#), [21](#), [23](#), [25](#)
- [cmdscale](#), [3](#), [11](#)
- [cp.xnews](#), [3](#), [3](#)
- [criteriaST.cv](#), [4](#)
- [croatia](#), [5](#)
- [croatia.grid7cp](#), [6](#), [9](#)
- [croatia.temp](#), [6](#), [7](#), [9](#)
- [croatia2008](#), [8](#)
- [croatiadb](#), [3](#), [9](#)
- [dblm](#), [4](#), [10](#), [10](#)
- [extractFormula](#), [11](#)
- [geosptdb \(geosptdb-package\)](#), [2](#)
- [geosptdb-package](#), [2](#)
- [gowdis](#), [10](#), [11](#)
- [gower.dist](#), [4](#)
- [graph.rbfST](#), [3](#), [12](#), [23](#), [24](#)
- [idwST](#), [14](#), [17–19](#)
- [idwST.cv](#), [16](#), [18](#)
- [idwST.cv1](#), [17](#)
- [idwST.tcv](#), [18](#), [18](#)
- [lm](#), [10](#)
- [optimize](#), [12–14](#), [17](#), [19](#)
- [rbfST](#), [3](#), [4](#), [13](#), [17](#), [20](#), [23–25](#)
- [rbfST.cv](#), [13](#), [22](#), [24](#)
- [rbfST.cv1](#), [23](#)
- [rbfST.tcv](#), [5](#), [24](#)
- [standardize](#), [26](#)