Package ‘CompRandFld’

February 19, 2015

Version 1.0.3-4
Date 2015-01-31
Title Composite-Likelihood Based Analysis of Random Fields
Author Simone Padoan[aut, cre] <simone.padoan@unibocconi.it>, Moreno Bevilacqua[aut] <moreno.bevilacqua@uv.cl>
Maintainer Simone Padoan <simone.padoan@unibocconi.it>
Depends R (>= 2.13)
Description A set of procedures for the analysis of Random Fields using likelihood and non-standard likelihood methods is provided. Spatial analysis often involves dealing with large dataset. Therefore even simple studies may be too computationally demanding. Composite likelihood inference is emerging as a useful tool for mitigating such computational problems. This methodology shows satisfactory results when compared with other techniques such as the tapering method. Moreover, composite likelihood (and related quantities) have some useful properties similar to those of the standard likelihood.
Suggests RandomFields, spam, scatterplot3d, fields, mapproj
License GPL (>= 2)
URL http://faculty.unibocconi.it/simonepadoan
Repository CRAN
Repository/R-Forge/Project comprandfld
Repository/R-Forge/Revision 258
Repository/R-Forge/DateTimeStamp 2015-02-01 15:14:09
Date/Publication 2015-02-01 19:46:28
NeedsCompilation yes

R topics documented:

  CheckCorrModel ................................................. 2
  CheckInput ..................................................... 3
  CheckLikelihood ................................................ 4
  CheckModel ..................................................... 5
CheckCorrModel

Description

Subroutine called by InitParam. The procedure controls if the correlation model inserted is correct.

Usage

CheckCorrModel(corrmodel)

Arguments

corrmodel  String; the name of a correlation model, for the description see Covmatrix.

Author(s)

Simone Padoan, <simone.padoan@unibocconi.it>, http://faculty.unibocconi.it/simonepadoan;
Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, https://sites.google.com/a/uv.cl/moreno-bevilacqua/home.
CheckInput

See Also

FitComposite

Checking Input

Description

Subroutine called by the fitting procedures. The procedure controls the validity of the input inserted by the users.

Usage

CheckInput(coordx, coordy, coordt, corrmodel, data, distance, 
fcall, fixed, grid, likelihood, margins, maxdist, 
maxtime, model, numblock, optimizer, param, replicates, 
start, taper, tapsep, threshold, type, varest, vartype, 
weighted)

Arguments

coordx  A numeric \((d \times 2)\)-matrix (where \(d\) is the number of points) assigning 2-dimensions of coordinates or a numeric vector assigning 1-dimension of coordinates.

coordy  A numeric vector assigning 1-dimension of coordinates; coordy is interpreted only if coordx is a numeric vector otherwise it will be ignored.

coordt  A numeric vector assigning 1-dimension of temporal coordinates.

corrmodel  String; the name of a correlation model, for the description see FitComposite.

data  A numeric vector or a \((n \times d)\)-matrix or \((d \times d \times n)\)-matrix of observations.

distance  String; the name of the spatial distance. The default is Eucl, the euclidean distance. See the Section Details.

fcall  String; Fitting to call the fitting procedure and simulation to call the simulation.

fixed  A named list giving the values of the parameters that will be considered as known values. The listed parameters for a given correlation function will be not estimated, i.e. if list(nugget=0) the nugget effect is ignored.

grid  Logical; if FALSE (the default) the data are interpreted as a vector or a \((n \times d)\)-matrix, instead if TRUE then \((d \times d \times n)\)-matrix is considered.

likelihood  String; the configuration of the composite likelihood. Marginal is the default.

margins  String; the type of the marginal distribution of the max-stable field.

maxdist  Numeric; an optional positive value indicating the maximum spatial distance considered in the composite-likelihood computation.

maxtime  Numeric; an optional positive value indicating the maximum temporal lag separation in the composite-likelihood.
model String; the density associated to the likelihood objects. Gaussian is the default.
numblock Numeric; the number of observation of the underlying random field (only for max-stable simulations). See RFsim.
optimizer String; the optimization algorithm (see optim for details). 'Nelder-Mead' is the default.
param A numeric vector of parameters, needed only in simulation. See RFsim.
replicates Logical; if FALSE (the default) one spatial random field is considered, instead if TRUE the data are considered as iid replicates of a field.
start A named list with the initial values of the parameters that are used by the numerical routines in maximization procedure. NULL is the default.
taper String; the name of the tapered correlation function.
tapsep Numeric; an optional value indicating the separable parameter in the space time quasi taper (see Details).
threshold Numeric; a value indicating a threshold for the binary random field.
type String; the type of the likelihood objects. If pairwise (the default) then the marginal composite likelihood is formed by pairwise marginal likelihoods.
varest Logical; if TRUE the estimate variances and standard errors are returned. FALSE is the default.
vartype String; the type of estimation method for computing the estimate variances, see FitComposite.
weighted Logical; if TRUE the likelihood objects are weighted. If FALSE (the default) the composite likelihood is not weighted.

Author(s)
Simone Padoan, <simone.padoan@unibocconi.it>, http://faculty.unibocconi.it/simonepadoan;
Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, https://sites.google.com/a/uv.cl/moreno-bevilacqua/home.

See Also
FitComposite

CheckLikelihood (likelihood)

Description
Subroutine called by InitParam. The procedure controls the type of the composite-likelihood inserted by the users.

Usage
CheckLikelihood(likelihood)
CheckModel

Arguments

  likelihood  String; the configuration of the composite likelihood. Marginal is the default.

Author(s)

  Simone Padoan, <simone.padoan@unibocconi.it>, http://faculty.unibocconi.it/simonepadoan;
  Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, https://sites.google.com/a/uv.cl/moreno-bevilacqua/home.

See Also

  FitComposite

CheckModel  Checking Random Field type

Description

  Subroutine called by InitParam. The procedure controls the type of random field inserted by the users.

Usage

  CheckModel(model)

Arguments

  model  String; the density associated to the likelihood objects. Gaussian is the default.

Author(s)

  Simone Padoan, <simone.padoan@unibocconi.it>, http://faculty.unibocconi.it/simonepadoan;
  Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, https://sites.google.com/a/uv.cl/moreno-bevilacqua/home.

See Also

  FitComposite
CheckVarType

### Description
Subroutine called by InitParam. The procedure controls the type of likelihood objects inserted by the users.

### Usage
```
CheckVarType(type)
```

### Arguments
| type | String; the type of the likelihood objects. If Pairwise (the default) then the marginal composite likelihood is formed by pairwise marginal likelihoods. |

### Author(s)
Simone Padoan, <simone.padoan@unibocconi.it>, http://faculty.unibocconi.it/simonepadoan; Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, https://sites.google.com/a/uv.cl/moreno-bevilacqua/home.

### See Also
FitComposite

CheckVarType

### Description
Subroutine called by InitParam. The procedure controls the method used to compute the estimates’ variances.

### Usage
```
CheckVarType(type)
```

### Arguments
| type | String; the method used to compute the estimates’ variances. If SubSamp (the default) the estimates’ variances are computed by the sub-sampling method, see FitComposite. |
Author(s)
Simone Padoan, <simone.padoan@unibocconi.it>, http://faculty.unibocconi.it/simonepadoan; Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, https://sites.google.com/a/uv.cl/moreno-bevilacqua/home.

See Also
FitComposite

CompLikelihood  Optimizes the Composite log-likelihood

Description
Subroutine called by FitComposite. The procedure estimates the model parameters by maximisation of the composite log-likelihood.

Usage
CompLikelihood(coordx, coordy, corrmodel, data, distance, flagcorr, flagnuis, fixed, grid, likelihood, lower, model, namescorr, namesnuis, namesparam, numparam, numparamcorr, optimizer, param, spacetime, threshold, type, upper, varest, vartype, winconst, winstp)

Arguments
coordx
A numeric \((d \times 2)\)-matrix (where \(d\) is the number of points) assigning 2-dimensions of coordinates or a numeric vector assigning 1-dimension of coordinates.

coordy
A numeric vector assigning 1-dimension of coordinates; \(\text{coordy}\) is interpreted only if \(\text{coordx}\) is a numeric vector otherwise it will be ignored.

corrmodel
Numeric; the id of the correlation model.

data
A numeric vector or a \((n \times d)\)-matrix or \((d \times d \times n)\)-matrix of observations.

distance
String; the name of the spatial distance. The default is Eucl, the euclidean distance. See the Section Details.

flagcorr
A numeric vector of binary values denoting which parameters of the correlation function will be estimated.

flagnuis
A numeric vector of binary values denoting which nuisance parameters will be estimated.

fixed
A numeric vector of parameters that will be considered as known values.

grid
Logical; if FALSE (the default) the data are interpreted as a vector or a \((n \times d)\)-matrix, instead if TRUE then \((d \times d \times n)\)-matrix is considered.

likelihood
String; the configuration of the composite likelihood, see FitComposite.
**CorrelationParam**

**lower**
A numeric vector with the lower bounds of the parameters’ ranges.

**model**
Numeric; the id value of the density associated to the likelihood objects.

**namescorr**
String; the names of the correlation parameters.

**namesnuis**
String; the names of the nuisance parameters.

**namesparam**
String; the names of the parameters to be maximised.

**numparam**
Numeric; the number of parameters to be maximised.

**numparamcorr**
Numeric; the number of correlation parameters.

**optimizer**
String; the optimization algorithm (see `optim` for details). 'Nelder-Mead' is the default.

**param**
A numeric vector of parameters’ values.

**spacetime**
Logical; if TRUE the random field is spatial-temporal otherwise is a spatial field.

**threshold**
Numeric; a value indicating a threshold for the binary random field, see `FitComposite`.

**type**
String; the type of the likelihood objects. If Pairwise (the default) then the marginal composite likelihood is formed by pairwise marginal likelihoods.

**upper**
A numeric vector with the upper bounds of the parameters’ ranges.

**varest**
Logical; if TRUE the estimate’ variances and standard errors are returned. FALSE is the default.

**vartype**
String; the type of estimation method for computing the estimate variances, see `FitComposite`.

**winconst**
Numeric; a positive value – if vartype=SubSamp – determines the window size in the sub-sampling estimates of the variances, see `FitComposite`.

**winstp**
Numeric; a positive value – if vartype=SubSamp – determines the window step in the sub-sampling estimates of the variances, see `FitComposite`.

**Author(s)**
Simone Padoan, <simone.padoan@unibocconi.it>, http://faculty.unibocconi.it/simonepadoan; Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, https://sites.google.com/a/uv.cl/moreno-bevilacqua/home.

**See Also**

`FitComposite`

**CorrelationParam**  
*Lists the Parameters of a Correlation Model*

**Description**
Subroutine called by InitParam and other procedures. The procedure returns a list with the parameters of a given correlation model.
Covariogram

Usage
CorrelationParam(corrmodel)

Arguments

corrmodel String; the name of a correlation model. See Covmatrix for details

Author(s)
Simone Padoan, <simone.padoan@unibocconi.it>, http://faculty.unibocconi.it/simonepadoan;
Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, https://sites.google.com/a/uv.cl/moreno-bevilacqua/home.

See Also
FitComposite

Examples

require(CompRandFld)

### Example 1. Parameters of the Matern correlation model
CorrelationParam("matern")

### Example 2. Parameters of the Gneiting correlation model
CorrelationParam("gneiting")

Covariogram Computes covariance, variogram and extremal coefficient functions

Description
The procedure computes and/or plots the covariance, the variogram or the extremal coefficient functions and the practical range estimated fitting a Gaussian or max-stable random field with the composite-likelihood or using the weighted least square method. Allows to add to the variogram or extremal coefficient plots the empirical estimates.
Covariogram

Usage

Covariogram(fitted, lags=NULL, lagt=NULL, answer.cov=FALSE, answer.vario=FALSE, answer.extc=FALSE, answer.range=FALSE, fix.lags=NULL, fix.lagt=NULL, show.cov=FALSE, show.vario=FALSE, show.extc=FALSE, show.range=FALSE, add.cov=FALSE, add.vario=FALSE, add.extc=FALSE, pract.range=95, vario, ...)

Arguments

fitted A fitted object obtained from the `FitComposite` or `WLeastSquare` procedures.
lags A numeric vector of distances.
lagt A numeric vector of temporal separations.
answer.cov Logical; if TRUE a vector with the estimated covariance function is returned; if FALSE (the default) the covariance is not returned.
answer.vario Logical; if TRUE a vector with the estimated variogram is returned; if FALSE (the default) the variogram is not returned.
answer.extc Logical; if TRUE a vector with the estimated extremal coefficient is returned; if FALSE (the default) the variogram is not returned.
answer.range Logical; if TRUE the estimated practical range is returned; if FALSE (the default) the practical range is not returned.
fix.lags Integer; a positive value denoting the spatial lag to consider for the plot of the temporal profile.
fix.lagt Integer; a positive value denoting the temporal lag to consider for the plot of the spatial profile.
show.cov Logical; if TRUE the estimated covariance function is plotted; if FALSE (the default) the covariance function is not plotted.
show.vario Logical; if TRUE the estimated variogram is plotted; if FALSE (the default) the variogram is not plotted.
show.extc Logical; if TRUE the estimated extremal coefficient is plotted; if FALSE (the default) the extremal coefficient is not plotted.
show.range Logical; if TRUE the estimated practical range is added on the plot; if FALSE (the default) the practical range is not added.
add.cov Logical; if TRUE the vector of the estimated covariance function is added on the current plot; if FALSE (the default) the covariance is not added.
add.vario Logical; if TRUE the vector with the estimated variogram is added on the current plot; if FALSE (the default) the correlation is not added.
add.extc Logical; if TRUE the vector with the estimated extremal coefficient is added on the current plot; if FALSE (the default) the correlation is not added.
pract.range Numeric; the percent of the sill to be reached.
vario A Variogram object obtained from the `EVariogram` procedure.
... other optional parameters which are passed to plot functions.
Covariogram

Value

The returned object is eventually a list with:

- covariance: The vector of the estimated covariance function;
- variogram: The vector of the estimated variogram function;
- extrcoeff: The vector of the estimated extremal coefficient function;
- practical.range: The estimated practical range.

Author(s)

Simone Padoan, <simone.padoan@unibocconi.it>, http://faculty.unibocconi.it/simonepadoan;
Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, https://sites.google.com/a/uv.cl/moreno-bevilacqua/home.

References


See Also

FitComposite, WLeastSquare

Examples

library(CompRandFld)
library(RandomFields)
library(scatterplot3d)
set.seed(31231)

# Set the coordinates of the points:
x <- runif(100, 0, 10)
y <- runif(100, 0, 10)
coords<-cbind(x,y)

# Example 1. Plot of covariance and variogram functions
# estimated from a Gaussian random field with exponent
correlation. One spatial replication is simulated.

#
# Set the model's parameters:
corrmodel <- "exponential"
mean <- 0
sill <- 1
nugget <- 0
scale <- 2

# Simulation of the Gaussian random field:
data <- RFsim(coordx=coords, corrmodel=corrmodel, param=list(mean=mean, 
sill=sill, nugget=nugget, scale=scale))

# Maximum composite-likelihood fitting of the Gaussian random field:
start<.list(scale=scale,sill=sill,mean=mean(data))
fixed<.list(nugget=nugget)

# Maximum composite-likelihood fitting of the random field:
fit <- FitComposite(data, coordx=coords, corrmodel=corrmodel,likelihood="Marginal", 
type="Pairwise",start=start,fixed=fixed,maxdist=6)

# Results:
print(fit)

# Empirical estimation of the variogram:
vario <- EVariogram(data, x, y)

# Plot of covariance and variogram functions:
par(mfrow=c(1,2))
Covariogram(fit, show.cov=TRUE, show.range=TRUE, 
show.vario=TRUE, vario=vario,pch=20)

# Example 2. Plot of covariance and extremal coefficient
# functions estimated from a max-stable random field with
# exponential correlation. n iid spatial replications are
# simulated.

set.seed(1156)
# Simulation of the max-stable random field:
data <- RFsim(coordx=coords, corrmodel=corrmodel, model="ExtGauss", replicates=20, 
param=list(mean=mean,sill=sill,nugget=nugget, scale=scale))

start<list(sill=sill, scale=scale)

# Maximum composite-likelihood fitting of the max-stable random field:
fit <- FitComposite(data, coordx=coords, corrmodel=corrmodel, model='ExtGauss', 
replicates=20, varest=TRUE, vartype='Sampling', 
margins="Frechet", start=start)
data <- Dist2Dist(data, to='Gumbel')

# Empirical estimation of the madogram:
vario <- EVariogram(data, coordx=coords, type='madogram', replicates=20)

# Plot of correlation and extremal coefficient functions:
par(mfrow=c(1,2))
Covariogram(fit, show.cov=TRUE, show.range=TRUE, show.extc=TRUE,
           vario=vario, pract.range=84, pch=20)

###############################################################################
### Example 3. Plot of covariance and variogram functions
### estimated from a Gaussian spatio-temporal random field with
### double-exp correlation.
### One spatio-temporal replication is simulated.
###
# Define the spatial-coordinates of the points:
x <- runif(20, 0, 1)
y <- runif(20, 0, 1)
# Define the temporal sequence:
time <- seq(0, 30, 1)

# Simulation of the spatio-temporal Gaussian random field:
data <- RfSim(x, y, time, corrmodel="exp_exp", param=list(mean=mean,
               nugget=nugget, scale_s=0.5, scale_t=1, sill=sill))
data

# Maximum composite-likelihood fitting of the space-time Gaussian random field:
fit <- FitComposite(data, x, y, time, corrmodel="exp_exp", maxtime=5,
                       likelihood="Marginal", type="Pairwise", fixed=list(
                       nugget=nugget, mean=mean), start=list(scale_s=0.2,
                       scale_t=1, sill=sill))

# Empirical estimation of spatio-temporal covariance:
vario <- EVariogram(data, x, y, time, maxtime=10)

# Plot of the fitted space-time covariance
#Covariogram(fit, show.cov=TRUE)

# Plot of the fitted space-time variogram
#Covariogram(fit, vario=vario, show.vario=TRUE)

# Plot of covariance, variogram and spatio and temporal profiles:
#Covariogram(fit, vario=vario, fix.lagt=1, fix.lags=1, show.vario=TRUE, pch=20)

###############################################################################
### Example 4. Plot of parametric and empirical lorelograms
### estimated from a Binary Gaussian random fields with
### exponential correlation. One spatial replication is
### simulated.
###

```r
#set.seed(1240)

# Define the spatial-coordinates of the points:
x <- seq(0, 0.1)
y <- seq(0, 0.1)

# Simulation of the Binary Gaussian random field:
data <- RFsim(x, y, corrm=model="BinaryGauss",
              threshold=0, param=list(nugget=0, mean=0,
              scale=0.6, sill=0.8))

data

# Maximum composite-likelihood fitting of the Binary Gaussian random field:
fit <- fitComposite(data, x, y, corrm=model="BinaryGauss",
                    maxdist=0.8, likelihood="Marginal", type="Pairwise",
                    start=list(mean=0, scale=0.1, sill=0.1))

# Empirical estimation of the lorelogram:
vario <- EVariogram(data, x, y, type="lorelogram", maxdist=2)

# Plot of fitted and empirical lorelograms:
Covariogram(fit, vario=vario, show.vario=TRUE, lags=seq(0.1, 2, 0.1), pch=20)
```

---

**Covmatrix**

**Spatio-temporal (tapered) Covariance Matrix**

---

**Description**

The function computes the (tapered) covariance matrix for a spatial (temporal or spatio-temporal) covariance model and a set of spatial (temporal or spatio-temporal) points.

**Usage**

```r
Covmatrix(coordx, coordy=NULL, coordt=NULL, corrm, distance="Eucl", grid=FALSE, iskrig=FALSE,
          maxdist=NULL, maxtime=NULL, param ,
          taper=NULL, tapsep=NULL, type="Standard")
```

**Arguments**

- **coordx** A numeric \((d \times 2)\)-matrix (where \(d\) is the number of spatial sites) giving 2-dimensions of spatial coordinates or a numeric \(d\)-dimensional vector giving 1-dimension of spatial coordinates.
- **coordy** A numeric vector giving 1-dimension of spatial coordinates; coordy is interpreted only if coordx is a numeric vector or grid=TRUE otherwise it will be ignored. Optional argument, the default is NULL then coordx is expected to be numeric a \((d \times 2)\)-matrix.
coordt A numeric vector giving 1-dimension of temporal coordinates. At the moment implemented only for the Gaussian case. Optional argument, the default is NULL then a spatial random field is expected.

corrmodel String; the name of a correlation model, for the description see the Section Details.

distance String; the name of the spatial distance. The default is Eucl, the euclidean distance. See FitComposite.

grid Logical; if FALSE (the default) the data are interpreted as spatial or spatial-temporal realisations on a set of non-equispaced spatial sites (irregular grid). See FitComposite.

iskrig Logical: the default value is FALSE. It is TRUE if the function is called by the function Kri.

maxdist Numeric; an optional positive value indicating the marginal spatial compact support. See FitComposite.

maxtime Numeric; an optional positive value indicating the marginal temporal compact support. See FitComposite.

param A list of parameter values required for the correlation model. See FitComposite and CorrelationParam.

taper String; the name of the taper correlation function if type is tapering, see the Section Details.

tapsep Numeric; an optional value indicating the separable parameter in the space-time quasi taper (see Details).

type String; the type of covariance matrix Standard (the default) or Tapering for tapered covariance matrix

Details

The parameter param is a list including all the parameters of a covariance function model. In particular, the covariance models share the following parameters: the sill that represents the common variance of the random field, the nugget that represents the local variation (white noise) at the origin. For each correlation model you can check the list of the specific parameters using CorrelationParam.

Here there is the list of all the implemented space and space-time correlation models. The list of space-time correlation functions includes separable and non-separable models.

- Purerly spatial correlation models:
  1. cauchy
     \[ R(h) = (1 + h^2)^{-\beta} \]
     The parameter \( \beta \) is positive. It is a special case of the gencauchy model.
  2. exponential
     \[ R(h) = e^{-h}, \quad h \geq 0 \]
     This model is a special case of the whittle and the stable model.
  3. gauss
     \[ R(h) = e^{-h^2} \]
     This model is a special case of the stable model.
4. gencauchy (generalised cauchy)

\[ R(h) = (1 + h^\alpha)^{-\beta} \]

The parameter \( \alpha \) is in \((0,2]\), and \( \beta \) is positive.

5. spherical

\[ R(h) = (1 - 1.5h + 0.5h^3)1_{[0,1]}(h) \]

This isotropic covariance function is valid only for dimensions less than or equal to 3.

6. stable

\[ R(h) = e^{-h^\alpha}, \quad h \geq 0 \]

The parameter \( \alpha \) is in \((0,2]\).

7. wave

\[ R(h) = \frac{\sin h}{h}, \quad h > 0 \quad \text{and} \quad R(0) = 1 \]

This isotropic covariance function is valid only for dimensions less than or equal to 3.

8. matern

\[ R(h) = 2^{1-\nu} \Gamma(\nu)^{-1} h^{\nu} K_\nu(h) \]

The parameter \( \nu \) is positive.

This is the model of choice if the smoothness of a random field is to be parameterised: if \( \nu > m \) then the graph is \( m \) times differentiable.

- Spatio-temporal correlation models:
  - Non-separable models:
    1. gneiting (non-separable space time model)

\[ R(h, u) = e^{\frac{-h^n}{1 + u^\lambda}} \]

The parameters \( \nu \) and \( \lambda \) take values in \([0,2]\); the parameter \( \gamma \) takes values in \([0,1]\). For \( \gamma = 0 \) it is a separable model.

2. gneiting GC (non-separable space time model with great circle distances)

\[ R(h, u) = e^{\frac{-h^n}{1 + u^\lambda}} \]

3. iacocesare (non-separable space time model)

\[ R(h, u) = (1 + h^n + u^\lambda)^{-\delta} \]

The parameters \( \nu \) and \( \lambda \) take values in \([1,2]\); the parameters \( \delta \) must be greater than or equal to half the space-time dimension.

4. porcu (non-separable space time model)

\[ R(h, u) = (0.5(1 + h^n)^\gamma + 0.5(1 + u^\lambda)^\gamma)^{-\gamma^{-1}} \]

The parameters \( \nu \) and \( \lambda \) take values in \([0,2]\); the parameter \( \gamma \) takes values in \([0,1]\). The limit of the correlation model as \( \gamma \) tends to zero leads to a separable model.
5. porcu2 (non-separable space time model)

\[ R(h, u) = e^{-h^\nu (1 + u^\lambda)^{0.5\gamma\nu}} \]

The parameters \( \nu \) and \( \lambda \) take values in \([0, 2]\); the parameter \( \gamma \) take values in \([0, 1]\). For \( \gamma = 0 \) it is a separable model.

- Separable models.

Space-time separable correlation models are easily obtained as the product of a spatial and a temporal correlation model, that is

\[ R(h, u) = R(h)R(u) \]

Several combinations are possible:
1. exp_exp: spatial exponential model and temporal exponential model
2. exp_cauchy: spatial exponential model and temporal cauchy model
3. matern_cauchy: spatial matern model and temporal cauchy model
4. stable_stable: spatial stabel model and temporal stable model

Note that some models are nested. (The exp_exp with the stable_stable for instance.)

- Spatial taper function models.

For spatial covariance tapering the tapered correlation functions are:
1. Wendland1

\[ R(h) = (1 - h)^2(1 + 0.5h)1_{[0,1]}(h) \]

2. Wendland2

\[ R(h) = (1 - h)^4(1 + 4h)1_{[0,1]}(h) \]

3. Wendland3

\[ R(h) = (1 - h)^6(1 + 6h + 35h^2/3)1_{[0,1]}(h) \]

- Spatio-tempora tapered correlation models.

For space-time covariance tapering likelihood the taper functions are obtained as the product of a spatial and a temporal taper (Separable taper). Several combinations are possible:
- Wendlandi_Wendlandj: spatial Wendlandi taper and temporal Wendlandj taper with \( i,j=1,2,3 \).

- Space-time non separable adaptive-taper with dynamically space-time compact support is:
  - qt_time and qt_space. In The case of qt_time the space-time quasi taper is:

\[ T(h, u) = (arg)^{-6}(1 + 7x)(1 - x)^71_{x \leq maxtime \leq arg}(u) \]

\[ arg = (1 + \frac{h}{maxdist})^\beta, x = u \frac{arg}{maxtime} \]

where \( 0 \leq \beta \leq 1 \) is a fixed parameter of separability (tapsep), maxtime the fixed temporal compact support and maxdist the fixed spatial scale parameter. The adaptive-taper qt_space is the same taper but changing the time with the space.
Remarks:

Let $R(h)$ be a spatial correlation model given in standard notation. Then the covariance model applied with arbitrary variance and scale equals to:

$$C(h) = \text{sill} + \text{nugget}, \quad \text{if} \quad h = 0$$

$$C(h) = \text{sill} \times R(\frac{h}{\text{scale}}), \quad \text{if} \quad h > 0$$

Similarly if $R(h,u)$ is a spatio-temporal correlation model given in standard notation, then the covariance model is:

$$C(h,u) = \text{sill} + \text{nugget}, \quad \text{if} \quad h = 0, u = 0$$

$$C(h,u) = \text{sill} \times R(\frac{h}{\text{scale}_s}, \frac{u}{\text{scale}_t}), \quad \text{if} \quad h > 0 \quad \text{or} \quad u > 0$$

Here ‘...’ stands for additional parameters.

Let $R(h)$ be a spatial taper given in standard notation. Then the taper function applied with an arbitrary compact support (maxdist) equals to:

$$T(h) = R(\frac{h}{\text{maxdist}})$$

Similarly if $R(h,u)$ is a spatio-temporal taper given in standard notation, then the taper function applied with arbitrary compact supports (maxdist, maxtime) equals to:

$$T(h,u) = R(\frac{h}{\text{maxdist}}, \frac{u}{\text{maxtime}})$$

Then the tapered covariance matrix is obtained as:

$$C_{\text{tap}}(h,u) = T(h,u)C(h,u)$$

Value

Returns an object of class CovMat. An object of class CovMat is a list containing at most the following components:

- **coordx** A $d$-dimensional vector of spatial coordinates;
- **coory** A $d$-dimensional vector of spatial coordinates;
- **coordt** A $t$-dimensional vector of temporal coordinates;
- **covmatrix** The covariance matrix if type is Standard. An object of class spam if type is Tapering
- **corrmodel** String: the correlation model;
- **distance** String: the type of spatial distance;
Covmatrix

grid Logical: TRUE if the spatial data are in a regular grid, otherwise FALSE;

nozero In the case of tapered matrix the percentage of non zero values in the covariance matrix. Otherwise is NULL.

maxdist Numeric: the marginal spatial compact support if type is Tapering;

maxtime Numeric: the marginal temporal compact support if type is Tapering;

namescorr String: The names of the correlation parameters;

numcoord Numeric: the number of spatial coordinates;

numtime Numeric: the number the temporal coordinates;

param Numeric: The covariance parameters;

tapmod String: the taper model if type is Tapering. Otherwise is NULL.

spacetime TRUE if spatio-temporal and FALSE if spatial covariance model;

In the space-time case covmatrix is the covariance matrix of the random vector

\[ Z(s_1, t_1), \ldots, Z(s_n, t_1), \ldots, Z(s_1, t_m), \ldots, Z(s_n, t_m) \]

for \( n \) spatial locatione sites and \( m \) temporal instants.

Author(s)

Simone Padoan, <simone.padoan@unibocconi.it>, http://faculty.unibocconi.it/simonepadoan;

Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, https://sites.google.com/a/uv.cl/moreno-bevilacqua/home.

References


See Also

kri, RFsim, FitComposite
library(CompRandFld)
library(spam)

### Example 1. Covariance matrix associated to a Matern correlation model

define the spatial coordinates of the points:
x <- runif(500, 0, 2)
y <- runif(500, 0, 2)

matrix1 <- Covmatrix(x, y, corrmmodel="matern", param=list(smooth=0.5,
sill=1, scale=0.2, mean=0))
dim(matrix1$covmatrix)

### Example 2. Tapered Covariance matrix associated to a Matern correlation model

### Example 3. Covariance matrix associated to a space-time double exponential correlation model

#define the temporal coordinates:
times <- c(1,2,3)

#define correlation model
corrmmodel="exp_exp"

#define covariance parameters
param=list(scale_s=0.3, scale_t=0.5, sill=1, mean=0)

# Simulation of a spatial Gaussian random field:
matrix3 <- Covmatrix(x, y, times, corrmmodel=corrmmodel,
                      param=param)
dim(matrix3$covmatrix)
Dist2Dist

Switches from an EV to Another EV Distribution

Description

The function transforms observations belonging to the GEV class from one model to another.

Usage

Dist2Dist(data, from='Gev', to='sfrechet', loc=NULL, scale=NULL, shape=NULL)

Arguments

data A numeric vector or a matrix of extreme values.
from The name of the original extreme value distribution, i.e. `Gev` (the default), see the Details section.
to The name of the desired extreme value distribution, i.e. `sfrechet` (the default), see the Details section.
loc A numeric value or vector of location parameters.
scale A numeric value or vector of scale parameters.
shape A numeric value or vector of shape parameters.
Details

If `data` is a numeric vector of length `n` then the dataset is considered as a realisation from an univariate extreme value distribution. Instead, if `data` is a `(n x d)`-matrix then the columns represent the different variables with extreme value distributions and the rows represent the iid replications. Finally, if `data` is a `(d x d x n)`-matrix then the columns and rows represent the different variables and the third dimension represents the iid replications.

The parameters `from` and `to` indicate the original extreme value distribution(s) from which the observations are drawn and the target extreme value distribution(s) that the transformed data will follow. The options are:

1. `from=Gev` (generalised extreme value distribution):
   - `to=Uniform`, which means uniform distribution;
   - `to=sFrechet`, which means standard (or unit) Frechet distribution, that is GEV(1,1,1);
   - `to=sGumbel`, which means standard Gumbel distribution, that is GEV(0,1,1);
   - `to=sWeibull`, which means standard Weibull distribution, that is GEV(1,1,-1);
   - `to=Gev`, which means generalised extreme value distribution. Note, that in this case, it is required to insert vectors of location, scale and shape parameters with dimension `n` in the univariate case, dimension `d` when `data` is `(n x d)`-matrix and dimension `n x d` when `data` is `(d x d x n)`-matrix.

2. `from=sFrechet`
   - `to=Gev`.

3. `from=sGumbel`
   - `to=Gev`.

4. `from=sWeibull`
   - `to=Gev`.

Value

A numeric vector or matrix of transformed values following the desired distribution.

Author(s)

Simone Padoan, <simone.padoan@unibocconi.it>, http://faculty.unibocconi.it/simonepadoan;
Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, https://sites.google.com/a/uv.cl/moreno-bevilacqua/
home.

References


See Also

FitGev
**Empirical Variogram(variants) of Gaussian, Binary and Max-Stable Fields**

**Description**

The function returns an empirical estimate of the variogram (or its variants) for Gaussian, Binary and max-stable random field.

**Usage**

```r
EVariogram(data, coordx, coordy, coordt=NULL, cloud=FALSE, distance='Eucl', grid=FALSE, gev=c(0,1,0), maxdist=NULL, maxtime=NULL, numbins=NULL, replicates=1, type='variogram')
```

**Arguments**

- `data` A d-dimensional vector (a single spatial realisation) or a (n x d)-matrix (n iid spatial realisations) or a (d x d)-matrix (a single spatial realisation on regular grid) or an (d x d x n)-array (n iid spatial realisations on regular grid) or a (t x d)-matrix (a single spatial-temporal realisation) or an (t x d x n)-array (n iid spatial-temporal realisations) or or an (d x d x t x n)-array (a single spatial-temporal realisation on regular grid) or an (d x d x t x n)-array (n iid spatial-temporal realisations on regular grid). See `FitComposite` for details.

- `coordx` A numeric (d x 2)-matrix (where d is the number of spatial sites) assigning 2-dimensions of spatial coordinates or a numeric d-dimensional vector assigning 1-dimension of spatial coordinates.

- `coordy` A numeric vector assigning 1-dimension of spatial coordinates; coordy is interpreted only if coordx is a numeric vector or grid=TRUE otherwise it will be ignored. Optional argument, the default is NULL then coordx is expected to be numeric a (d x 2)-matrix.

- `coordt` A numeric vector assigning 1-dimension of temporal coordinates. Optional argument, the default is NULL then a spatial random field is expected.

- `cloud` Logical; if TRUE the variogram cloud is computed, otherwise if FALSE (the default) the empirical (binned) variogram is returned.

- `distance` String; the name of the spatial distance. The default is Eucl, the euclidean distance. See the Section Details of `FitComposite`.

- `grid` Logical; if FALSE (the default) the data are interpreted as spatial or spatial-temporal realisations on a set of non-equispaced spatial sites.

- `gev` A numeric vector with the three GEV parameters;

- `maxdist` A numeric value denoting the spatial maximum distance, see the Section Details.

- `maxtime` A numeric value denoting the temporal maximum distance, see the Section Details.
numbins  A numeric value denoting the numbers of bins, see the Section Details.
replicates Numeric; a positive integer denoting the number of independent and identically distributed (iid) replications of a spatial or spatial-temporal random field. Optional argument, the default value is 1 then a single realisation is considered.
type A String denoting the type of variogram. Four options are available: variogram, madogram, Fmadogram and 1orelogram. It is returned respectively, the standard variogram with the first (Gaussian responses), the madogram with the second and third (extreme values), the lorelogram with the fourth (Binary data).

Details

We briefly report the definitions of variogram used in this function.

In the case of a spatial Gaussian random field the sample variogram estimator is defined by
\[
\hat{\gamma}(h) = 0.5 \sum_{x_i, x_j \in N(h)} (Z(x_i) - Z(x_j))^2 / |N(h)|
\]
where \(N(h)\) is the set of all the sample pairs whose distances fall into a tolerance region with size \(h\) (equispaced intervalls are considered). Observe, that in the literature often the above definition is termed semivariogram (see e.g. the first reference). Nevertheless, here this defition has been used in order to be consistent with the variogram defition used for the extremes (see e.g. the third reference).

In the case of a spatial max-stable random field, the sample madogram estimator is defined similarly to the Gaussian case by
\[
\hat{\nu}(h) = 0.5 \sum_{x_i, x_j \in N(h)} |Z(x_i) - Z(x_j)| / |N(h)|.
\]

In the case of a spatial binary random field, the sample lorelogram estimator (the analogue of the correlation) is defined by
\[
\hat{L}(h) = \frac{N_{11}(h)N_{00}(h)}{(N_{01}(h)N_{10}(h))},
\]
where \(N_{11}(h)\) is the number of pairs who are both equal to 1 and that falls in the bin \(h\). Similarly are defined the other quantities.

In the case of a spatio-temporal Gaussian random field the sample variogram estimator is defined by
\[
\hat{\gamma}(h, u) = 0.5 \sum_{(x_i, l), (x_j, k) \in N(h, u)} (Z(x_i, l) - Z(x_j, k))^2 / |N(h, u)|
\]
where \(N(h, u)\) is the set of all the sample pairs whose spatial distances fall into a tolerance region with size \(h\) and \(|k - l| = u\). Note, that \(Z(x_i, l)\) is the observation at site \(x_i\) and time \(l\). Taking this in mind and given the above definition of lorelogram, the spatio-temporal extention is straightforward.

The numbins parameter indicates the number of adjacent intervals to consider in order to grouped distances with which to compute the (weighted) lest squares.

The maxdist parameter indicates the maximum spatial distance below which the shorter distances will be considered in the calculation of the (weigthed) least squares.

The maxtime parameter indicates the maximum temporal distance below which the shorter distances will be considered in the calculation of the (weigthed) least squares.
Value

Returns an object of class `variogram`. An object of class `variogram` is a list containing at most the following components:

- **bins**: Adjacent intervals of grouped spatial distances if `cloud=FALSE`. Otherwise if `cloud=TRUE` all the spatial pairwise distances;
- **bint**: Adjacent intervals of grouped temporal distances if `cloud=FALSE`. Otherwise if `cloud=TRUE` all the temporal pairwise distances;
- **cloud**: If the variogram cloud is returned (TRUE) or the empirical variogram (FALSE);
- **centers**: The centers of the spatial bins;
- **distance**: The type of spatial distance;
- **extcoeff**: The spatial extremal coefficient function. Available only if `type` is equal to `madogram` or `fmadogram` (for the moment available only for a spatial random field);
- **lenbins**: The number of pairs in each spatial bin;
- **lenbinst**: The number of pairs in each spatial-temporal bin;
- **lenbint**: The number of pairs in each temporal bin;
- **srange**: The maximum and minimum spatial distances used for the calculation of the variogram;
- **variograms**: The empirical spatial variogram;
- **variogramst**: The empirical spatial-temporal variogram;
- **variogramt**: The empirical temporal variogram;
- **trange**: The maximum and minimum temporal distance used for the calculation of the variogram;
- **type**: The type of estimated variogram: the standard variogram or the madogram.

Author(s)

Simone Padoan, <simone.padoan@unibocconi.it>, http://faculty.unibocconi.it/simonepadoan; Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, https://sites.google.com/a/uv.cl/moreno-bevilacqua/home.

References


See Also

FitComposite

Examples

library(CompRandFld)
library(RandomFields)
set.seed(514)

# Set the coordinates of the sites:
x <- runif(150, 0, 10)
y <- runif(150, 0, 10)

# Set the model's parameters:
corrmodel <- "exponential"
mean <- 0
sill <- 1
nugget <- 0
scale <- 3

# Simulation of the spatial Gaussian random field:
data <- RFsim(x, y, corrmmodel=corrmodel, param=list(mean=mean,
sill=sill, nugget=nugget, scale=scale))$data

# Empirical spatial variogram estimation:
fit <- EVariogram(data, x, y)

# Results:
plot(fit$centers, fit$variograms, xlab='h', ylab=expression(gamma(h)),
     ylim=c(0, max(fit$variograms)), xlim=c(0, fit$srange[2]), pch=20,
     main="variogram")

# Example 2. Empirical estimation of the variogram from a
# spatio-temporal Gaussian random fields with Gneiting
# correlation function.
# One spatio-temporal replication is simulated
#
set.seed(331)
# Define the temporal sequence:
times <- seq(1,7,1)

# Simulation of a spatio-temporal Gaussian random field:
data <- RFsim(x, y, times, corrmodel="gneiting",
               param=list(mean=0, scale_s=0.4, scale_t=1, sill=sill,
                            nugget=0, power_s=1, power_t=1, sep=0.5))$data

# Empirical spatio-temporal variogram estimation:
fit <- EVariogram(data, x, y, times, maxtime=5, maxdist=4)

# Results: Marginal spatial empirical variogram
par(mfrow=c(2,2), mai=c(.5,.5,.3,3), mgp=c(1.4,.5,0))
plot(fit$centers, fit$variograms, xlab='h', ylab=expression(gamma(h)),
     ylim=c(0, max(fit$variograms)), xlim=c(0, max(fit$centers)),
     pch=20, main="Marginal spatial Variogram",cex.axis=.8)

# Results: Marginal temporal empirical variogram
plot(fit$bin, fit$variogramt, xlab='t', ylab=expression(gamma(t)),
     ylim=c(0, max(fit$variograms)), xlim=c(0, max(fit$bin)),
     pch=20, main="Marginal temporal Variogram",cex.axis=.8)

# Building space-time variogram
st.vario <- matrix(fit$variogramst,length(fit$centers),length(fit$bin))

st.vario <- cbind(c(0,fit$variograms), rbind(fit$variogramt, st.vario))

# Results: 3d Spatio-temporal variogram
require(scatterplot3d)
scatterplot3d(st.grid[,1], st.grid[,2], c(st.vario),
              highlight.3d=TRUE, xlab="h",ylab="t",
              zlab=expression(gamma(h,t)), pch=20,
              main="Space-time variogram",cex.axis=.7,
              mar=c(2,2,2,2), mgp=c(0,0,0),
              cex.lab=.7)

# A smoothed version
par(mai=c(.2,.2,.2,.2),mgp=c(1,.3, 0))
persp(c(0,fit$centers), c(0,fit$bin), st.vario,
      xlab="h", ylab="u", zlab=expression(gamma(h,u)),
      ltheta=90, shade=0.75, ticktype="detailed", phi=30,
      theta=30,main="Space-time variogram",cex.axis=.8,
      cex.lab=.8)


### Example 3. Empirical estimation of the madogram from a max-stable random field (Extremal Gaussian model) with exponential correlation.
## n iid spatial replications are simulated.
###
set.seed(7273)
# Simulation of the max-stable random field:
data <- RFsim(x, y, corrmodel=corrmodel, model="ExtGauss",
               param=list(mean=mean, sill=sill, nugget=nugget,
                           scale=scale), replicates=40)$data
# Tranform data from from common unit Frechet to standard Gumbel margins:
data <- Dist2Dist(data, to='sGumbel')
# Empirical madogram estimation:
fit <- EVariogram(data, x, y, type='madogram', replicates=40, cloud=FALSE)
# Results:
par(mfrow=c(1,2), mai=c(.6,.6,.3,.3), mgp=c(1.6,.6,0))
plot(fit$centers, fit$variograms, xlab='h', ylab=expression(nu(h)),
ylim=c(0, max(fit$variograms)), xlim=c(0, fit$srange[2]), pch=20,
      main="madogram")
plot(fit$centers, fit$extcoeff, xlab='h', ylab=expression(theta(h)),
ylim=c(1, 2), xlim=c(0, fit$srange[2]), pch=20,
      main="extremal coefficient")

---

FitComposite

Max-Likelihood-Based Fitting of Gaussian, Binary and Max-Stable Fields

Description

Maximum weighted composite-likelihood fitting of spatio-temporal Gaussian, binary and spatial max-stable random fields. For the spatio-temporal Gaussian random field, (restricted) maximum likelihood and tapered likelihood fitting is also available. The function returns the model parameters’ estimates and the estimates’ variances and allows to fix any of the parameters.

Usage

FitComposite(data, coordx, coordy=NULL, coordt=NULL, corrmodel,
distance='Eucl', fixed=NULL, grid=FALSE,
likelihood='Marginal', margins='Gev', maxdist=NULL,
maxtime=NULL, model='Gaussian', optimizer='Nelder-Mead',
replicates=1, start=NULL, taper=NULL, tapsep=NULL,
threshold=NULL, type='Pairwise', varest=FALSE,
vartype='SubSamp', weighted=FALSE, winconst, winstp)
Arguments

**data**
A \(d\)-dimensional vector (a single spatial realisation) or a \((n \times d)\)-matrix (\(n\) iid spatial realisations) or a \((d \times d)\)-matrix (a single spatial realisation on regular grid) or an \((d \times d \times n)\)-array (\(n\) iid spatial realisations on regular grid) or a \((t \times d)\)-matrix (a single spatial-temporal realisation) or an \((t \times d \times n)\)-array (\(n\) iid spatial-temporal realisations on regular grid) or an \((d \times d \times t \times n)\)-array (\(n\) iid spatial-temporal realisations on regular grid). For the description see the Section Details.

**coordx**
A numeric \((d \times 2)\)-matrix (where \(d\) is the number of spatial sites) assigning 2-dimensions of spatial coordinates or a numeric \(d\)-dimensional vector assigning 1-dimension of spatial coordinates.

**coordy**
A numeric vector assigning 1-dimension of spatial coordinates; `coordy` is interpreted only if `coordx` is a numeric vector or `grid=TRUE` otherwise it will be ignored. Optional argument, the default is NULL then `coordx` is expected to be numeric a \((d \times 2)\)-matrix.

**coordt**
A numeric vector assigning 1-dimension of temporal coordinates. At the moment implemented only for the Gaussian case. Optional argument, the default is NULL then a spatial random field is expected.

**corrmodel**
String; the name of a correlation model, for the description see the Section Details.

**distance**
String; the name of the spatial distance. The default is `eucl`, the euclidean distance. See the Section Details.

**fixed**
An optional named list giving the values of the parameters that will be considered as known values. The listed parameters for a given correlation function will be not estimated, i.e. if `list(nugget=0)` the nugget effect is ignored.

**grid**
Logical; if `FALSE` (the default) the data are interpreted as spatial or spatial-temporal realisations on a set of non-equispaced spatial sites (irregular grid).

**likelihood**
String; the configuration of the composite likelihood. `marginal` is the default, see the Section Details.

**margins**
String; the type of the marginal distribution of the max-stable field. `gev` is the default, see the Section Details.

**maxdist**
Numeric; an optional positive value indicating the maximum spatial distance considered in the composite or tapered likelihood computation. See the Section Details for more information.

**maxtime**
Numeric; an optional positive value indicating the maximum temporal separation considered in the composite or tapered likelihood computation (see Details).

**model**
String; the type of random field and therefore the densities associated to the likelihood objects. `gaussian` is the default, see the Section Details.

**optimizer**
String; the optimization algorithm (see optim for details). `Nelder-Mead` is the default.

**replicates**
Numeric; a positive integer denoting the number of independent and identically distributed (iid) replications of a spatial or spatial-temporal random field. Optional argument, the default value is 1 then a single realisation is considered.
An optional named list with the initial values of the parameters that are used by the numerical routines in maximization procedure. NULL is the default (see Details).

taper

String; the name of the type of covariance matrix. It can be Standard (the default value) or Tapering for tapered covariance matrix.

tapsep

Numeric; an optional value indicating the separable parameter in the space time quasi taper (see Details).

threshold

Numeric; a value indicating a threshold for the binary random field. Optional in the case that model is BinaryGauss, see the Section Details.

type

String; the type of the likelihood objects. If Pairwise (the default) then the marginal composite likelihood is formed by pairwise marginal likelihoods (see Details).

varest

Logical; if TRUE the estimates’ variances and standard errors are returned. FALSE is the default.

vartype

String; (SubSamp the default) the type of method used for computing the estimates’ variances, see the Section Details.

weighted

Logical; if TRUE the likelihood objects are weighted, see the Section Details. If FALSE (the default) the composite likelihood is not weighted.

winconst

Numeric; a positive value for computing the sub-window size where observations are sampled in the sub-sampling procedure (if vartype=SubSamp). For increasing winconst increasing sub-window sizes are obtained. Optional argument, the default is 1. See Details for more information.

winstp

Numeric; a value in (0, 1] for computing the sub-window step (in the sub-sampling procedure). This value denote the proportion of the sub-window size. Optional argument, the default is 0.5. See Details for more information.

Details

Note, that the standard likelihood may be seen as particular case of the composite likelihood. In this respect FitComposite provides maximum (restricted) likelihood fitting. Only composite likelihood estimation based on pairs are considered. Specifically marginal pairwise, conditional pairwise and difference pairwise. Covariance tapering is considered only for Gaussian random fields.

With data, coordx, coordy, coordt, grid and replicates parameters:

- If data is a numeric \( d \)-dimensional vector, coordx and coordy are two numeric \( d \)-dimensional vectors (or coordx is \( (d \times 2) \)-matrix and coordy=NULL), coordt=NULL, grid=FALSE and replicates=1, then the data are interpreted as a single spatial realisation observed on \( d \) spatial sites;

- If data is a numeric \( (n \times d) \)-matrix, coordx and coordy are two numeric \( d \)-dimensional vectors (or coordx is \( (d \times 2) \)-matrix and coordy=NULL), coordt=NULL, grid=FALSE and replicates=n, then the data are interpreted as \( n \) iid replications of a spatial random field observed on \( d \) spatial sites.

- If data is a numeric \( (d \times d) \)-matrix, coordx and coordy are two numeric \( d \)-dimensional vectors, coordt=!=NULL, grid=TRUE and replicates=1, then the data are interpreted as a single spatial random field realisation observed on \( d \) equispaced spatial sites (named regular grid).
• If data is a numeric \((d \times d \times n)\)-array, coordx and coordy are two numeric \(d\)-dimensional vectors, coordt=NULL, grid=TRUE and replicates=n, then the data are interpreted as \(n\) iid realisations of a spatial random field observed on \(d\) equispaced spatial sites.

• If data is a numeric \((t \times d)\)-matrix, coordx and coordy are two numeric \(d\)-dimensional vectors (or coordx is \((d \times 2)\)-matrix and coordy=NULL), coordt is a numeric \(t\)-dimensional vector, grid=FALSE and replicates=1, then the data are interpreted as a single spatial-temporal realisation of a random field observed on \(d\) spatial sites and for \(t\) times.

• If data is a numeric \((t \times d \times n)\)-array, coordx and coordy are two numeric \(d\)-dimensional vectors (or coordx is \((d \times 2)\)-matrix and coordy=NULL), coordt is a numeric \(t\)-dimensional vector, grid=FALSE and replicates=n, then the data are interpreted as \(n\) iid realisations of a spatial-temporal random field observed on \(d\) spatial sites and for \(t\) times.

• If data is a numeric \((d \times d \times t)\)-array, coordx and coordy are two numeric \(d\)-dimensional vectors, coordt is a numeric \(t\)-dimensional vector, grid=TRUE and replicates=1, then the data are interpreted as a single spatial-temporal realisation of a random field observed on \(d\) equispaced spatial sites and for \(t\) times.

• If data is a numeric \((d \times d \times t \times n)\)-array, coordx and coordy are two numeric \(d\)-dimensional vectors, coordt is a numeric \(t\)-dimensional vector, grid=TRUE and replicates=n, then the data are interpreted as \(n\) iid realisation of a spatial-temporal random field observed on \(d\) equispaced spatial sites and for \(t\) times.

The corrmodel parameter allows to select a specific correlation function for the random field. (See Covmatrix).

The distance parameter allows to consider different kinds of spatial distances. The settings alternatives are:

1. Eucl, the euclidean distance (default value);
2. Chor, the chordal distance;
3. Geod, the geodesic distance;

The likelihood parameter represents the composite-likelihood configurations. The settings alternatives are:

1. Conditional, the composite-likelihood is formed by conditionals likelihoods;
2. Marginal, the composite-likelihood is formed by marginals likelihoods;
3. Full, the composite-likelihood turns out to be the standard likelihood;

The margins parameter concerns only max-stable fields and indicates how the margins are considered. The options are Gev or Frechet, where in the former case the marginals are supposed generalized extreme value distributed and in the latter case unit Frechet distributed.

The maxdist parameter set the maximum spatial distance below which pairs of sites with inferior distances are considered in the composite-likelihood. This can be inferior of the effective maximum spatial distance. Note that this corresponds to use a weighted composite-likelihood with binary weights. Pairs with distance less than maxdist have weight 1 and are included in the likelihood computation, instead those with greater distance have weight 0 and then excluded. The default is NULL, in this case the effective maximum spatial distance between sites is considered.

The same arguments of maxdist are valid for maxtime but here the weighted composite-likelihood regards the case of spatial-temporal field. At the moment is implemented only for Gaussian random
fields. The default is NULL, in this case the effective maximum temporal lag between pairs of observations is considered.

In the case of tapering likelihood maxdist and maxtime describes the spatial and temporal compact support of the taper model (see Covmatrix). If they are not specified then the maximum spatial and temporal distances are considered. In the case of space time quasi taper the tapsep parameter allows to specify the spatio temporal compact support (see Covmatrix).

The model parameter indicates the type of random field considered, for instance model=Gaussian denotes a Gaussian random field. Accordingly, this also determines the analytical expression of the finite dimensional distribution associated with the random field. The available options are:

- Gaussian, for a Gaussian random field (see i.e. Wackernagel, H. 1998);
- BinaryGauss, for a Binary random field (see Heagerty and Lele 1998)
- BrowResn, for a Brown-Resnick max-stable random field (see Kabluchko, Z. et al. 2009);
- ExtGauss, for an Extremal Gaussian max-stable random field (known also as Schlather model) (see Schlather, M. 2002);
- ExtT, for an Extremal t max-stable random field (see Davison, A. C. et al. 2012);

Note, that only for the Gaussian case the estimation procedure is implemented for spatial and spatial-temporal random fields.

The start parameter allows to specify starting values. If start is omitted the routine is computing the starting values using the weighted moment estimator.

The taper parameter, optional in case that type=Tapering, indicates the type of taper correlation model. (See Covmatrix)

The threshold parameter indicates the value (common for all the spatial sites) above which the values of the underlying Gaussian latent process are considered successes events (values below are instead failures). See e.g. Heagerty and Lele (1998) for more details.

The type parameter represents the type of likelihood used in the composite-likelihood definition. The possible alternatives are listed in the following scheme.

1. If a Gaussian random field is considered (model=Gaussian):
   - If the composite is formed by marginal likelihoods (likelihood=Marginal):
     - Pairwise, the composite-likelihood is defined by the pairwise likelihoods;
     - Difference, the composite-likelihood is defined by likelihoods which are obtained as difference of the pairwise likelihoods.
   - If the composite is formed by conditional likelihoods (likelihood=Conditional)
     - Pairwise, the composite-likelihood is defined by the pairwise conditional likelihoods.
   - If the composite is formed by a full likelihood (likelihood=Full):
     - Standard, the objective function is the classical multivariate likelihood;
     - Restricted, the objective function is the restricted version of the full likelihood (e.g. Harville 1977, see References);
     - Tapering, the objective function is the tapered version of the full likelihood (e.g. Kaufman et al. 2008, see References).
The varest parameter specifies if the standard error estimation of the estimated parameters must be computed. For Gaussian random field and standard (restricted) likelihood estimation, standard errors are computed as square root of the diagonal elements of the Fisher Information matrix (asymptotic covariance matrix of the estimates under increasing domain). For Gaussian random field and tapered and composite likelihood estimation, standard errors estimate are computed as square root of the diagonal elements of the Godambe Information matrix. (asymptotic covariance matrix of the estimates under increasing domain (see Shaby, B. and D. Ruppert (2012) for tapering and Bevilacqua et. al. (2012) Bevilacqua and Gaetan (2013) for weighted composite likelihood)). The vartype parameter specifies the method used to compute the estimates’ variances in the composite likelihood case. In particular for estimating the variability matrix $J$ in the Godambe expression matrix. This parameter is considered if $\text{varest} = \text{TRUE}$. The options are:

- **SubSamp** (the default), indicates the Sub-Sampling method;
- **Sampling**, indicates that the variability matrix is estimated by the sample contro-part (available only for $n$ iid replications of the random field, i.e. \text{replicates=n});

The weighted parameter specifies if the likelihoods forming the composite-likelihood must be weighted. If $\text{TRUE}$ the weights are selected by opportune procedures that improve the efficient of the maximum composite-likelihood estimator (not implemented yet). If $\text{FALSE}$ the efficient improvement procedure is not used.

For computing the standard errors by the sub-sampling procedure, winconst and winstp parameters represent respectively a positive constant used to determine the sub-window size and the the step with which the sub-window moves.

In the spatial case (subset of $R^2$), the domain is seen as a rectangle $B \times H$, therefore the size of the sub-window side $b$ is given by $b = winconst \times \sqrt{B}$ (similar is of $h$). For a complete description see Lee and Lahiri (2002). By default $\text{winconst}$ is set $B/(2 \times \sqrt{B})$. The $\text{winstp}$ parameter is used to determine the sub-window step. The latter is given by the proportion of the sub-window size, so that when $\text{winstp}=1$ there is not overlapping between contiguous sub-windows. In the spatial case by default $\text{winstp}=0.5$. The sub-window is moved by successive steps in order to cover the entire spatial domain. Observations, that fall in disjoint or overlapping windows are considered independent samples.

In the spatio-temporal case the subsampling is meant only in time as described by Li et al. (2007). Thus, $\text{winconst}$ represents the lengh of the temporal sub-window. By default the size of the sub-window is computed following the rule established in Li et al. (2007). By default $\text{winstp}$ is the time step.

Observe that in the spatio-temporal case, the returned values by $\text{srange}$ and $\text{trange}$, represent respectively the minimum and maximum of the marginal spatial distances and those of the temporal separations. Thus, the minimum being not the overall (i.e. considering the spatio-temporal coordinates) is not zero, as one could be expect and the latter can be easily added by the user.

### Value

Returns an object of class `FitComposite`. An object of class `FitComposite` is a list containing at most the following components:

- **clic** The composite information criterion, if the full likelihood is considered then it coincides with the Akaike information criterion;
- **coordx** A $d$-dimensional vector of spatial coordinates;
coordy A \(d\)-dimensional vector of spatial coordinates;

coordt A \(t\)-dimensional vector of temporal coordinates;

convergence A string that denotes if convergence is reached;

corrmmodel The correlation model;

data The vector or matrix or array of data;

distance The type of spatial distance;

fixed The vector of fixed parameters;

iterations The number of iteration used by the numerical routine;

likelihood The configuration of the composite likelihood;

logCompLik The value of the log composite-likelihood at the maximum;

message Extra message passed from the numerical routines;

model The density associated to the likelihood objects;

nozero In the case of tapered likelihood the percentage of non zero values in the covariance matrix. Otherwise is NULL.

numcoord The number of spatial coordinates;

numrep The number of the iid replicatations of the random field;

numtime The number the temporal realisations of the random field;

param The vector of parameters’ estimates;

srange The minimum and maximum spatial distance (see Details). The maximum is \(\text{maxdist}\), if inserted, rather the effective maximum distance;

stderr The vector of standard errors;

sensmat The sensitivity matrix;

varcov The matrix of the variance-covariance of the estimates;

varimat The variability matrix;

vartype The method used to compute the variance of the estimates;

trange The minimum and maximum temporal separation (see Details). The maximum is \(\text{maxtime}\), if inserted, rather then the effective maximum separation;

threshold The threshold used in the binary random field.

type The type of the likelihood objects.

winconst The constant use to compute the window size in the sub-sampling procedure;

winstp The step used for moving the window in the sub-sampling procedure

Author(s)

Simone Padoan, <simone.padoan@unibocconi.it>, http://faculty.unibocconi.it/simonepadoan;

Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, https://sites.google.com/a/uv.cl/moreno-bevilacqua/

home.
References


Maximum Restricted Likelihood Estimator:


Tapered likelihood:


Composite-likelihood:


Weighted Composite-likelihood for binary random fields:


Weighted Composite-likelihood for max-stable random fields:


Weighted Composite-likelihood for Gaussian random fields:


Spatial Extremes:


Sub-sampling estimation:

See Also
*Covmatrix, GaussRF, MaxStableRF, WLeastSquare, optim*

Examples
```r
library(CompRandFld)
library(RandomFields)
library(spam)
set.seed(3132)

# Define the spatial-coordinates of the points:
x <- runif(100, 0, 10)
y <- runif(100, 0, 10)

# Set the covariance model's parameters:
corrmodel <- "exponential"
mean <- 0
sill <- 1
nugget <- 0
scale <- 1.5
param <- list(mean=mean, sill=sill, nugget=nugget, scale=scale)
coords <- cbind(x,y)
# Simulation of the spatial Gaussian random field:
data <- RFSim(coordx=coords, corrmodel=corrmodel, param=param)$data

# Fixed parameters
fixed <- list(mean=mean, nugget=nugget)

# Starting value for the estimated parameters
start <- list(scale=scale, sill=sill)
```
### Example 1. Maximum likelihood fitting of
### Gaussian random fields with exponential correlation.
### One spatial replication.
### Likelihood setting: composite with
### marginal pairwise likelihood objects.
###

```r
# Maximum composite-likelihood fitting of the random field:
fit <- FitComposite(data, coord=coords, corrmodel=corrmodel, maxdist=2,
                     likelihood="Marginal", type="Pairwise", varest=TRUE,
                     start=start, fixed=fixed)

# Results:
print(fit)
```

### Example 2. Maximum likelihood fitting of
### Gaussian random fields with exponential correlation.
### One spatial replication.
### Likelihood setting: standard full likelihood.
###

```r
# Maximum composite-likelihood fitting of the random field:
fit <- FitComposite(data, coord=coords, corrmodel=corrmodel, maxdist=2,
                     likelihood="Full", type="Standard", varest=TRUE,
                     start=start, fixed=fixed)

# Results:
print(fit)
```

### Example 3. Maximum likelihood fitting of
### Gaussian random fields with exponential correlation.
### One spatial replication.
### Likelihood setting: tapered full likelihood.
###

```r
# Maximum tapered likelihood fitting of the random field:
fit <- FitComposite(data, coord=coords, corrmodel=corrmodel, maxdist=2,
                     type="Tapering", taper="Wendland1", varest=TRUE,
                     start=start, fixed=fixed)

# Results:
print(fit)
```
### Likelihood setting: composite with marginal pairwise likelihood objects.
### Simulation of a max-stable random field in the specified points:
```r
# Simulation of a max-stable random field in the specified points:
data <- RFSim(x, y, corrm=corrm, model="ExtGauss", replicates=30,
               param=list(mean=mean, sill=sill, nugget=nugget, scale=scale))$data
# Maximum composite-likelihood fitting of the random field:
fit <- FitComposite(data, x, y, corrm=corrm, model="ExtGauss",
                    replicates=30, varest=TRUE, vartype="Sampling",
                    margins="Frechet", start=list(sill=sill, scale=scale))
# Results:
print(fit)
```

### Example 5. Maximum likelihood fitting of Binary-Gaussian random fields with exponential correlation. One spatial replication.
### Likelihood setting: composite with marginal pairwise likelihood objects.
### Simulation of the spatial Binary-Gaussian random field:
```r
# Simulation of the spatial Binary-Gaussian random field:
data <- RFSim(coordx=coords, corrm=corrm, model="BinaryGauss",
               threshold=0, param=list(mean=mean, sill=.8,
                        nugget=nugget, scale=scale))$data
# Maximum composite-likelihood fitting of the random field:
#fit <- FitComposite(data, coordx=coordx, corrm=corrm, threshold=0,
#                    model="BinaryGauss", fixed=list(nugget=nugget,
#                        mean=0), start=list(scale=.1, sill=.1))
# Results:
#print(fit)
```
Examples of spatio-temporal random fields

# Define the temporal sequence:
#time <- seq(1, 80, 1)

# Define the spatial coordinates of the points:
x <- runif(10, 0, 10)
y <- runif(10, 0, 10)
coords<-cbind(x,y)

# Set the covariance model's parameters:
corrmodel="exp_exp"
scale_s=0.5
scale_t=1
sill=1
nugget=0
mean=0

# Simulation of the spatial-temporal Gaussian random field:
data <- RFSim(coordx=coords,coordt=time,corrmodel=corrmodel, param=param)$data

# Fixed parameters
fixed<-list(mean=mean,nugget=nugget)

# Starting value for the estimated parameters
start<-list(scale_s=scale_s,scale_t=scale_t,sill=sill)

# Maximum composite-likelihood fitting of the random field:
fit <- FitComposite(data=data,coordx=coords,coordt=time,corrmodel="exp_exp",
maxtime=2,maxdist=1,likelihood="Marginal",type="Pairwise",
start=start,fixed=fixed)

# Results:
print(fit)
Description
the function returns the parameters’ estimates and the variances of the estimates (if required) of the
generalized extreme value distribution for a given dataset of extreme values.

Usage
FitGev(data, method='Nelder-Mead', start, varest=FALSE)

Arguments
data A vector of extreme values.
method The optimization method (see optim for details). 'Nelder-Mead' is the default.
start A named list with the initial values for the parameters over which the likelihood
is to be maximized.
varest Logical; if TRUE the estimate’ variances and the standard errors are returned,
instead if FALSE (the default) only the estimate are computed.

Details
If start is omitted the routine is computing the starting values using moment estimators.

Value
The returned object is a list with:

param The vector of parameters’ estimates.
varcov The matrix of the variance-covariance of the estimates.
stderr The vector of the standard errors.

Author(s)
Simone Padoan, <simone.padoan@unibocconi.it>, http://faculty.unibocconi.it/simonepadoan;
Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, https://sites.google.com/a/uv.cl/moreno-bevilacqua/home.

References
York.
72, 67–90.

See Also
GevLogLik, optim
**GevLogLik**  

*Log-Likelihood of the GEV Distribution*

**Description**

The function returns the log-likelihood value of the Generalized Extreme Value Distribution for a given set of data and parameters.

**Usage**

```r
data  numdata  param
```

**Arguments**

- `data`: A vector of extreme values.
- `numdata`: The number of data observations.
- `param`: The vector of GEV parameters (location, scale and shape).

**Value**

The log-likelihood value is returned.

**Author(s)**

Simone Padoan, <simone.padoan@unibocconi.it>, [http://faculty.unibocconi.it/simonepadoan](http://faculty.unibocconi.it/simonepadoan); Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, [https://sites.google.com/a/uv.cl/moreno-bevilacqua/home](https://sites.google.com/a/uv.cl/moreno-bevilacqua/home).

**References**


---

**HypoTest**  

*Statistical Hypothesis Tests for Nested Models*

**Description**

The function performs statistical hypothesis tests for nested models based on composite likelihood versions of: Wald-type, score-type and Wilks-type (likelihood ratio) statistics.

**Usage**

```r
data  numdata  param
```

**Arguments**

- `object1`: 
- `object2`: 
- `...`: 
- `statistic`: 

**Value**

The log-likelihood value is returned.
Arguments

object1  An object of class `fitcomposite`.
object2  An object of class `fitcomposite` that is a nested model within object1.
...  Further successively nested objects.
statistic  String; the name of the statistic used within the hypothesis test (see Details).

Details

The implemented hypothesis tests for nested models are based on the following statistics:

1. Wald-type (\texttt{wald});
2. Score-type, also known as Rao-type (\texttt{rao});
3. Wilks-type; also known as the composite likelihood ratio statistic. Available are variants of the basic version, in particular:
   - Rotnitzky and Jewell adjustment (\texttt{wilksRJ});
   - Satterhwaite adjustment (\texttt{wilksS});
   - Chandler and Bate adjustment (\texttt{wilksCB});
   - Pace, Salvan and Sartori adjustment (\texttt{wilksPSS});

More specifically, consider an \(p\)-dimensional random vector \(Y\) with probability density function \(f(y; \theta)\), where \(\theta \in \Theta\) is a \(q\)-dimensional vector of parameters. Suppose that \(\theta = (\psi, \tau)\) can be partitioned in a \(q'\)-dimensional subvector \(\psi\) and \(q''\)-dimensional subvector \(\tau\). Assume also to be interested in testing the specific values of the vector \(\psi\). Then, one can use some statistical hypothesis tests for testing the null hypothesis \(H_0: \psi = \psi_0\) against the alternative \(H_1: \psi \neq \psi_0\). Composite likelihood versions of 'Wald' and 'score' statistics have the usual asymptotic chi-square distribution with \(q'\) degree of freedom. The Wald-type statistic is

\[
W = (\hat{\psi} - \psi_0)^T (G_{\psi\psi})^{-1}(\hat{\theta})(\hat{\psi} - \psi_0),
\]

where \(G_{\psi\psi}\) is the \(q' \times q'\) submatrix of the Godambe information pertaining to \(\psi\) and \(\hat{\theta}\) is the maximum likelihood estimator from the full model. The score-type statistic (Rao-type) is

\[
W = s_\psi(\psi_0, \hat{\tau}(\psi_0))^T H^{\psi\psi}(\hat{\theta}_\psi)(G_{\psi\psi}(\hat{\theta}_\psi))^{-1} H^{\psi\psi}(\hat{\theta}_\psi)s_\psi(\psi_0, \hat{\tau}(\psi_0)),
\]

where \(H^{\psi\psi}\) is the \(q' \times q'\) submatrix of the inverse of \(H(\theta)\) pertaining to \(\psi\) (the same for \(G\)) and \(\hat{\theta}_\psi\) is the constrained maximum likelihood estimate of \(\theta\) for fixed \(\psi\). These two statistics can be called from the routine \texttt{HypoTest} assigning at the argument \texttt{statistic} respectively the values: \texttt{wald} and \texttt{rao}.

Alternatively to the Wald-type and score-type statistics one can use the composite version of the Wilks-type or likelihood ratio statistic, given by

\[
W = 2[C(\hat{\theta}; y) - C(\psi_0, \hat{\tau}(\psi_0); y)].
\]

The asymptotic distribution of the composite likelihood ratio statistic is given by

\[
W \sim \sum_i \lambda_i \chi^2.
\]
for \( i = 1, \ldots, q' \), where \( \chi^2_i \) are \( q' \) iid copies of a chi-square one random variable and \( \lambda_1, \ldots, \lambda_{q'} \) are the eigenvalues of the matrix \((H^\psi\psi)^{-1}G^\psi\psi\). There exist several adjustments to the composite likelihood ratio statistic in order to get an approximated \( \chi^2_{q'} \). For example, Rotnitzky and Jewell (1990) proposed the adjustment \( W' = W/\bar{\lambda} \) where \( \bar{\lambda} \) is the average of the eigenvalues \( \lambda_i \). This statistic can be called within the routine by the value: \texttt{WilksRJ}. A better solution is proposed by Satterthwaite (1946) defining \( W'' = \nu W/(q' \bar{\lambda}) \), where \( \nu = (\sum_i \lambda_i^2)/\sum_i \lambda_i^2 \) for \( i = 1, \ldots, q' \), is the effective number of the degree of freedom. Note that in this case the distribution of the likelihood ratio statistic is a chi-square random variable with \( \nu \) degree of freedom. This statistic can be called from the routine assigning the value: \texttt{WilksS}. For the adjustments suggested by Chandler and Bate (2007) and Pace, Salvan and Sartori (2011) we refere to the articles (see \textbf{References}), these versions can be called from the routine assigning respectively the values: \texttt{WilksCB} and \texttt{WilksPSS}.

\textbf{Value}

An object of class \texttt{c("data.frame")}. The object contain a table with the results of the tested models. The rows represent the responses for each model and the columns the following results:

- \texttt{Num.Par}: The number of the model’s parameters.
- \texttt{Diff.Par}: The difference between the number of parameters of the model in the previous row and those in the actual row.
- \texttt{Df}: The effective number of degree of freedom of the chi-square distribution.
- \texttt{Chisq}: The observed value of the statistic.
- \texttt{Pr(>chisq)}: The p-value of the quantile Chisq computed using a chi-squared distribution with \texttt{Df} degrees of freedom.

\textbf{Author(s)}

Simone Padoan, \texttt{<simone.padoan@unibocconi.it>}, \url{http://faculty.unibocconi.it/simonepadoan};
Moreno Bevilacqua, \texttt{<moreno.bevilacqua@uv.cl>}, \url{https://sites.google.com/a/uv.cl/moreno-bevilacqua/home}.

\textbf{References}


\textbf{See Also}

\texttt{FitComposite}.
Examples

# Please remove the symbol hashtag to run the code

```r
library(CompRandFld)
library(RandomFields)
set.seed(3451)

# Define the spatial-coordinates of the points:
x <- runif(300, 0, 10)
y <- runif(300, 0, 10)

# Example 1. Composite likelihood-based hypothesis testing.
# Simulation of a Gaussian spatial random field with
# stable correlation.
# Estimation by composite likelihood using the setting:
# marginal pairwise likelihood objects.
#
# Set the model's parameters:
corrmodel <- "stable"
mean <- 0
sill <- 1
nugget <- 1
scale <- 1
power <- 1.3

data <- RFsim(x, y, corrmodel=corrmodel, param=list(mean=mean,
    sill=sill,nugget=nugget,scale=scale,power=power))$data

# Simulation of the spatial Gaussian random field:
data <- RFsim(x, y, corrmodel=corrmodel, param=list(mean=mean,
    sill=sill,nugget=nugget,scale=scale,power=power))$data

# Maximum composite-likelihood fitting of the random field, full model:
fit1 <- FitComposite(data, x, y, corrmodel=corrmodel, maxdist=5,
    varest=TRUE,start=list(mean=mean,power=power,scale=scale,sill=sill),
    fixed=list(nugget=1))

# Maximum composite-likelihood fitting of the random field, first nested model:
fit2 <- FitComposite(data, x, y, corrmodel=corrmodel, maxdist=5,
    varest=TRUE,start=list(mean=mean,power=power,scale=scale),
    fixed=list(nugget=1,sill=1))

# Maximum composite-likelihood fitting of the random field, second nested model:
fit3 <- FitComposite(data, x, y, corrmodel=corrmodel, maxdist=5,
    varest=TRUE,start=list(scale=scale),
    fixed=list(nugget=1,sill=1,mean=0,power=1.3))

# Hypothesis testing results:
# composite Wald-type statistic:
HypoTest(fit1, fit2, fit3, statistic='Wald')
```
# composite score-type statistic:
HypoTest(fit1, fit2, fit3, statistic='Rao')

# composite likelihood ratio statistic with RJ adjustment:
HypoTest(fit1, fit2, fit3, statistic='WilksRJ')

# composite likelihood ratio statistic with S adjustment:
HypoTest(fit1, fit2, fit3, statistic='WilksS')

# composite likelihood ratio statistic with CB adjustment:
HypoTest(fit1, fit2, fit3, statistic='WilksCB')

# composite likelihood ratio statistic with PSS adjustment:
HypoTest(fit1, fit2, fit3, statistic='WilksPSS')

---

InitParam

**Initializes the Parameters for Estimation Procedures**

**Description**

Subroutine called by the fitting procedures. The procedure initializes the parameters for the fitting procedure.

**Usage**

InitParam(coordx, coordy, coordt, corrmodel, data, distance, fcall, fixed, grid, likelihood, margins, maxdist, maxtime, model, numblock, param, parscale, paramrange, replicates, start, taper, tapsep, threshold, type, typereal, varest, vartype, weighted, winconst, winstp)

**Arguments**

- **coordx**: A numeric $(d\times 2)$-matrix (where $d$ is the number of points) assigning 2-dimensions of coordinates or a numeric vector assigning 1-dimension of coordinates.
- **coordy**: A numeric vector assigning 1-dimension of coordinates; coordy is interpreted only if coordx is a numeric vector otherwise it will be ignored.
- **coordt**: A numeric vector assigning 1-dimension of temporal coordinates.
- **corrmodel**: String; the name of a correlation model.
- **data**: A numeric vector or a $(n \times d)$-matrix or $(d \times d \times n)$-matrix of observations.
- **distance**: String; the name of the spatial distance. The default is Eucl, the euclidean distance. See the Section Details.
- **fcall**: String; "fitting" to call the fitting procedure and "simulation" to call the simulation procedure.
- **fixed**: A named list giving the values of the parameters that will be considered as known values.
grid Logical; if FALSE (the default) the data are interpreted as a vector or a \((n \times d)\)-matrix, instead if TRUE then \((d \times d \times n)\)-matrix is considered.

likelihood String; the configuration of the composite likelihood.

margins String; the type of the marginal distribution of the max-stable field.

maxdist Numeric; an optional positive value indicating the maximum spatial distance considered in the composite-likelihood computation.

maxtime Numeric; an optional positive value indicating the maximum temporal lag considered in the composite-likelihood computation.

model String; the density associated to the likelihood objects. Gaussian is the default.

numblock Numeric; the observation size of the underlying random field. Only in case of max-stable random fields and in the simulation.

param A numeric vector of parameter values required in the simulation procedure of random fields.

parscale A numeric vector of scaling factor to improve the maximizing procedure, see optim.

paramrange A numeric vector of parameters ranges, see optim.

replicates Logical; if FALSE (the default) one spatial random field is considered, instead if TRUE the data are considered as iid replicates of a field.

start A named list with the initial values of the parameters that are used by the numerical routines in maximization procedure.

taper String; the name of the type of covariance matrix. It can be Standard (the default value) or Tapering for tapered covariance matrix.

tapsep Numeric; an optional value indicating the separable parameter in the space time quasi taper (see Details).

threshold Numeric; a value indicating a threshold for the binary random field.

type String; the type of likelihood objects. Temporary value set to be "WLeast-Square" (weighted least-square) in order to compute the starting values.

typereal String; the real type of likelihood objects. See FitComposite.

varest Logical; if TRUE the estimates’ variances and standard errors are returned. FALSE is the default.

vartype String; the type of estimation method for computing the estimate variances, see the Section Details.

weighted Logical; if TRUE the likelihood objects are weighted, see FitComposite.

winconst Numeric; a positive real value indicating the sub-window in the sub-sampling procedure. See FitComposite.

winstp Numeric; a value in \((0, 1]\) for defining the window step. See FitComposite.

Acknowledgements

We thank the CRAN team for having helped us in finding a bug in the code.
See Also

FitComposite

Description

The function computes simple or ordinary (tapered) kriging, in addition, for a set of unknown spatial location sites and temporal instants and a given space or space-time covariance model, it computes the Kriging variance.

Usage

Kri(data, coordx, coordy=NULL, coordt=NULL, corrmodel, distance="Eucl", grid=FALSE, loc, maxdist=NULL, maxtime=NULL, param, taper=NULL, tapsep=NULL, time=NULL, type="Standard", type_krig="Simple")

Arguments

data A d-dimensional vector (a single spatial realisation) or a (d x d)-matrix (a single spatial realisation on regular grid) or a (t x d)-matrix (a single spatial-temporal realisation) or an (d x d x t x n)-array (a single spatial-temporal realisation on regular grid) giving the data used for prediction.

coordx A numeric (d x 2)-matrix (where d is the number of spatial sites) giving 2-dimensions of spatial coordinates or a numeric d-dimensional vector giving 1-dimension of spatial coordinates used for prediction.

coordy A numeric vector giving 1-dimension of spatial coordinates used for prediction; coordy is interpreted only if coordx is a numeric vector or grid=TRUE otherwise it will be ignored. Optional argument, the default is NULL then coordx is expected to be numeric a (d x 2)-matrix.

coordt A numeric vector giving 1-dimension of temporal coordinates used for prediction; the default is NULL then a spatial random field is expected.

corrmodel String; the name of a correlation model, for the description see the Section Details.

distance String; the name of the spatial distance. The default is Eucl, the euclidean distance. See the Section Details of FitComposite.

grid Logical; if FALSE (the default) the data used for prediction are interpreted as spatial or spatial-temporal realisations on a set of non-equispaced spatial sites (irregular grid).
A numeric \((n \times 2)\)-matrix (where \(n\) is the number of spatial sites) giving 2-dimensions of spatial coordinates to be predicted.

Numeric; an optional positive value indicating the maximum spatial compact support in the case of covariance tapering kriging.

Numeric; an optional positive value indicating the maximum temporal compact support in the case of covariance tapering kriging.

A list of parameter values required for the correlation model. See the Section Details.

String; the name of the taper correlation function, see the Section Details.

Numeric; an optional value indicating the separable parameter in the space time quasi taper (see Details).

A numeric \((m \times 1)\) vector (where \(m\) is the number of temporal instants) giving the temporal instants to be predicted; the default is NULL then only spatial prediction is performed.

String; if standard then standard kriging is performed; if Tapering then kriging with covariance tapering is performed.

String; the type of kriging. If Simple (the default) then simple kriging is performed. If ordinary then ordinary kriging is performed. (See the Section Details).

For a spatial or spatio-temporal dataset, given a set of locations and temporal instants and a correlation model \(\text{corrmodel}\) with some fixed parameters, the function computes simple or ordinary kriging, for the specified spatial locations \(\text{loc}\) and temporal instants \(\text{time}\), providing also the respective standard error. For the choice of the spatial or spatio temporal correlation model see details in \text{Covmatrix} function. The parameter \(\text{param}\) specifies the covariance parameters, see \text{CorrelationParam} and \text{Covmatrix} for details. The \(\text{type_krig}\) parameter indicates the type of kriging. In the case of simple kriging, the known mean can be specified by the parameter mean within list \(\text{param}\) (See examples). In addition, it is possible to perform kriging based on covariance tapering for simple kriging (Furrer et al., 2008). In this case, space or space-time tapered function and spatial or spatio- temporal compact support must be specified. For the choice of a space or space-time tapered function see \text{Covmatrix}. When performing kriging with covariance tapering, sparse matrix algorithms are exploited using the package \text{spam}.

Returns an object of class \(\text{Kg}\). An object of class \(\text{Kg}\) is a list containing at most the following components:

- \(\text{coordx}\) A \(d\)-dimensional vector of spatial coordinates used for prediction;
- \(\text{coordy}\) A \(d\)-dimensional vector of spatial coordinates used for prediction;
- \(\text{coordt}\) A \(t\)-dimensional vector of temporal coordinates used for prediction;
- \(\text{corrmodel}\) String: the correlation model;
- \(\text{covmatrix}\) The covariance matrix if \(\text{type}\) is Standard. An object of class spam if \(\text{type}\) is Tapering.
data: The vector or matrix or array of data used for prediction.
distance: String: the type of spatial distance;
grid: TRUE if the spatial data used for prediction are observed in a regular grid, otherwise FALSE;
loc: A \((n \times 2)\)-matrix of spatial locations to be predicted.
nozero: In the case of tapered simple kriging the percentage of non zero values in the covariance matrix. Otherwise is NULL.
numcoord: Numeric: the number \(d\) of spatial coordinates used for prediction;
umloc: Numeric: the number \(n\) of spatial coordinates to be predicted;
umtime: Numeric: the number \(d\) of the temporal instants used for prediction;
umt: Numeric: the number \(m\) of the temporal instants to be predicted;
param: Numeric: The covariance parameters;
pred: A \((m \times n)\)-matrix of spatio or spatio temporal kriging prediction;
spacetime: TRUE if spatio-temporal kriging and FALSE if spatial kriging;
tapmod: String: the taper model if type is Tapering. Otherwise is NULL.
time: A \(m\)-dimensional vector of temporal coordinates to be predicted;
type: String: the type of kriging (Standard or Tapering).
type_krig: String: the type of kriging (Simple or Ordinary).
varpred: A \((m \times n)\)-matrix of spatio or spatio temporal variance kriging prediction;

Author(s)
Simone Padoan, <simone.padoan@unibocconi.it>, http://faculty.unibocconi.it/simonepadoan;
Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, https://sites.google.com/a/uv.cl/moreno-bevilacqua/home.

References

See Also
Covmatrix

Examples

library(CompRandFld)
library(fields)
Example of Spatial kriging

# Define the spatial coordinates of the points:
x <- runif(50, 0, 1)
y <- runif(50, 0, 1)

# Set the model's parameters:
corrmodel <- "exponential"
mean<-0
sill<-1
nugget<-0
scale<-.5
param<-list(mean=mean,sill=sill,nugget=nugget,scale=scale)

# Spatial matrix location sites
coords<-cbind(x,y)

# Simulation of the spatial Gaussian random field:
set.seed(3132)
data <- RFsim(coordx=coords, corrmodel=corrmodel,
    param=param)$data
start<-list(scale=scale,sill=sill)
fixed<-list(mean=mean,nugget=nugget)
# Maximum likelihood fitting:
fit <- FitComposite(data, coordx=coords, corrmodel=corrmodel,
    likelihood='Full', type='Standard',
    start=start,fixed=fixed)

# Locations to predict
xx<-seq(0,.1,0.02)
loc_to_pred<-as.matrix(expand.grid(xx,xx))

pr<-Kri(loc=loc_to_pred,coordx=coords,corrmodel=corrmodel,
    param= as.list(c(fit$param,fit$fixed)), data=data)

# Example 1. Spatial simple kriging of n sites of a
# Gaussian random fields with exponential correlation.
#
pr_tap<-Kri(loc=loc_to_pred,coordx=coords,corrmodel=corrmodel, data=data,
    type="Tapering",
### Examples of Spatio-temporal Kriging

```r
### Define the spatial coordinates of the points:
x <- runif(15, 0, 1)
y <- runif(15, 0, 1)
coords <- cbind(x, y)
times <- 1:7

### Define the times to predict
times_to_pred <- 8:10

### Define model correlation and associated parameters
corrmodel <- "exp_exp"
param <- list(nugget = 0, mean = 1, scale_s = 1, scale_t = 2, sill = 2)

### Simulation of the space time Gaussian random field:
set.seed(31)
data <- RFsim(coordx = coords, coordt = times, corrmodel = corrmodel, param = param)$data

### Maximum likelihood fitting of the space time random field:
start <- list(scale_s = 1, scale_t = 2, sill = 2)
fixed <- list(mean = 1, nugget = 0)
fit <- FitComposite(data, coordx = coords, coordt = times, corrmodel = corrmodel, likelihood = 'Marginal', type = 'Pairwise', start = start, fixed = fixed, maxdist = 0.5, maxtime = 3)
```

---

```r
## 
## maxdist=0.15, taper="Wendland1")
##
colour <- rainbow(100)

#simple kriging map prediction
image.plot(xx, xx, matrix(pr$pred, ncol=length(xx)), col=colour,
##
## xlab="", ylab="", main="Simple Kriging")

# simple kriging map prediction variance
image.plot(xx, xx, matrix(pr$varpred, ncol=length(xx)), col=colour,
##
## xlab="", ylab="", main="Std error")

# simple tapered kriging map prediction
image.plot(xx, xx, matrix(pr_tap$pred, ncol=length(xx)), col=colour,
##
## xlab="", ylab="", main="Simple Tapered Kriging")

# simple tapered kriging map prediction variance
image.plot(xx, xx, matrix(pr_tap$varpred, ncol=length(xx)), col=colour,
##
## xlab="", ylab="", main="Std error")
```
### Example 3. Spatio temporal simple kriging of n locations and m temporal instants for a Gaussian random fields with estimated double exponential correlation.

```r
param <- as.list(c(fit$param, fit$fixed))
pr <- Kri(loc=loc_to_pred, time=time_to_pred, coordx=coords, coordt=times, corrmmodel=corrmmodel, param=param, data=data)
par(mfrow=c(3,2))
colour <- rainbow(100)
for(i in 1:3){
    image.plot(xx, xx, matrix(pr$pred[i,], ncol=length(xx)), col=colour, main = paste("Kriging Time" , i),ylab="")
    image.plot(xx, xx, matrix(pr$varpred[i,], ncol=length(xx)), col=colour, main = paste("Std error Time" , i),ylab="")
}
```

### Example 4. Spatio temporal tapered simple kriging of n locations and m temporal instants for a Gaussian random fields with estimated double exponential correlation.

```r
#pr_tap <- Kri(loc=loc_to_pred, time=time_to_pred, coordx=coords, coordt=times,
  #corrmmodel=corrmmodel, param=param, type="Tapering", maxdist=0.4, maxtime=4,
  #taper="Wendland2_Wendland2", data=data)
#par(mfrow=c(3,2))

#for(i in 1:3){
  #image.plot(xx, xx, matrix(pr_tap$pred[i,], ncol=length(xx)), col=colour, main = paste("Tapered Kriging Time" , i),ylab="")
  #image.plot(xx, xx, matrix(pr_tap$varpred[i,], ncol=length(xx)), col=colour, main = paste("Tapered Std error Time" , i),ylab="")
#}
```

---

**Likelihood**

*Optimizes the Log Likelihood*
Description

Subroutine called by FitComposite. The procedure estimates the model parameters by maximization of the log-likelihood.

Usage

Likelihood(corrmodel, data, fixed, flagcor, flagnuis, grid, lower,
          model, namescorr, namesnuis, namesparam, numcoord, numpairs,
          numparamcor, numrep, numtime, optimizer, param, setup,
          spacetime, varest, taper, type, upper)

Arguments

corrmodel    Numeric; the id of the correlation model.
data         A numeric vector or a \((n \times d)\)-matrix or \((d \times d \times n)\)-matrix of observations.
flagcor      A numeric vector of flags denoting which correlation parameters have to be estimated.
flagnuis     A numeric vector of flags denoting which nuisance parameters have to be estimated.
fixed        A numeric vector of parameters that will be considered as known values.
grid         Logical; if FALSE (the default) the data are interpreted as a vector or a \((n \times d)\)-matrix, instead if TRUE then \((d \times d \times n)\)-matrix is considered.
lower        A numeric vector with the lower bounds of the parameters’ ranges.
model        Numeric; the id value of the density associated to the likelihood objects.
namescorr    String; the names of the correlation parameters.
namesnuis    String; the names of the nuisance parameters.
namesparam   String; the names of the parameters to be maximised.
numcoord     Numeric; the number of coordinates.
numpairs     Numeric; the number of pairs.
umparamcor   Numeric; the number of the correlation parameters.
umrep        Numeric; the number of iid replications.
umtime       Numeric; the number of temporal observations.
optimizer    String; the optimization algorithm (see optim for details). ’Nelder-Mead’ is the default.
param        A numeric vector of parameters.
setup        A List of useful components for the estimation based on the maximum tapered likelihood.
spacetime    Logical; if the random field is spatial (FALSE) or spatio-temporal (TRUE).
varest       Logical; if TRUE the estimate’ variances and standard errors are returned. FALSE is the default.
taper        String; the name of the taper correlation function.
type         String; the type of the likelihood objects. If Pairwise (the default) then the marginal composite likelihood is formed by pairwise marginal likelihoods.
upper        A numeric vector with the upper bounds of the parameters’ ranges.
MomEst

**Author(s)**
Simone Padoan, <simone.padoan@unibocconi.it>, [http://faculty.unibocconi.it/simonepadoan](http://faculty.unibocconi.it/simonepadoan);
Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, [https://sites.google.com/a/uv.cl/moreno-bevilacqua/home](https://sites.google.com/a/uv.cl/moreno-bevilacqua/home).

**See Also**
FitComposite

---

**MomEst**

*Estimation of the GEV parameters by the Method of Moments*

**Description**

Using the moment estimator, the function returns the parameter estimates of the generalized extreme value distribution for a given dataset of extreme values.

**Usage**

MomEst(data, n)

**Arguments**

data A vector of extreme values.
n The number of observations.

**Value**

The returned object is a list with:

- location The location estimate.
- scale The scale estimate.
- shape The shape estimate.

**Author(s)**
Simone Padoan, <simone.padoan@unibocconi.it>, [http://faculty.unibocconi.it/simonepadoan](http://faculty.unibocconi.it/simonepadoan);
Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, [https://sites.google.com/a/uv.cl/moreno-bevilacqua/home](https://sites.google.com/a/uv.cl/moreno-bevilacqua/home).

**References**


**See Also**

GevLogLik, FitGev
NuisanceParam

Lists the Nuisance Parameters of a Random Field

Description

Subroutine called by InitParam and other procedures. The procedure returns a list with the nuisance
parameters of a given random field model.

Usage

NuisanceParam(model)

Arguments

  model            String; the name of a random field.

Author(s)

Simone Padoan, <simone.padoan@unibocconi.it>, http://faculty.unibocconi.it/simonepadoan;
Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, https://sites.google.com/a/uv.cl/moreno-bevilacqua/home.

See Also

FitComposite

Rfsim

Simulation of Gaussian, Binary and Max-stable Random Fields

Description

Simulation of spatial and spatio-temporal Gaussian, binary and max-stable random fields. The func-
tion returns one or more replications of a random field for a given covariance model and covariance
parameters.

Usage

Rfsim(coordx, coordy=NULL, coordt=NULL, corrmmodel, distance="Eucl",
   grid=FALSE, model='Gaussian', numbloc=NULL, param,
   replicates=1, threshold=NULL)
Arguments

coordx  A numeric \((d \times 2)\)-matrix (where \(d\) is the number of spatial sites) giving 2-dimensions of spatial coordinates or a numeric \(d\)-dimensional vector giving 1-dimension of spatial coordinates.

coordy  A numeric vector giving 1-dimension of spatial coordinates; coordy is interpreted only if coordx is a numeric vector or grid=TRUE otherwise it will be ignored. Optional argument, the default is NULL then coordx is expected to be numeric a \((d \times 2)\)-matrix.

coordt  A numeric vector giving 1-dimension of temporal coordinates. At the moment implemented only for the Gaussian case. Optional argument, the default is NULL then a spatial random field is expected.

corrmodel  String; the name of a correlation model, for the description see the Section Details.

distance  String; the name of the spatial distance. The default is EuCl, the euclidean distance. See the Section Details of FitComposite.

grid  Logical; if FALSE (the default) the data are interpreted as spatial or spatial-temporal realisations on a set of non-equispaced spatial sites (irregular grid).

model  String; the type of random field and therefore the densities associated to the likelihood objects. Gaussian is the default, see the Section Details.

numblock  Numeric; the observation size of the underlying random field. Only in case of max-stable random fields.

param  A list of parameter values required in the simulation procedure of random fields, see Examples.

replicates  Numeric; a positive integer denoting the number of independent and identically distributed (iid) replications of a spatial or spatial-temporal random field. Optional argument, the default value is 1 then a single realisation is considered.

threshold  Numeric; a value indicating a threshold for the binary random field. Optional in the case that model is BinaryGauss, see the Section Details.

Details

Note that this function is also interfaced to the R package RandomFields, using fast routines therein developed for the simulation of random fields, see for example GaussRF, MaxStableRF, ect.

Value

Returns an object of class RFsim. An object of class RFsim is a list containing at most the following components:

coordx  A \(d\)-dimensional vector of spatial coordinates;

coordy  A \(d\)-dimensional vector of spatial coordinates;

coordt  A \(t\)-dimensional vector of temporal coordinates;

corrmodel  The correlation model; see Covmatrix.

data  The vector or matrix or array of data, see FitComposite;
**RFsim**

- **distance** The type of spatial distance;
- **model** The type of random field, see `FitComposite`.
- **numcoord** The number of spatial coordinates;
- **numtime** The number the temporal realisations of the random field;
- **param** The vector of parameters' estimates;
- **randseed** The seed used for the random simulation;
- **replicates** The number of the iid replicatations of the random field;
- **spacetime** TRUE if spatio-temporal and FALSE if spatial random field;
- **threshold** The threshold for deriving the binary random field.

**Author(s)**

Simone Padoan, <simone.padoan@unibocconi.it>, http://faculty.unibocconi.it/simonepadoan; Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, https://sites.google.com/a/uv.cl/moreno-bevilacqua/home.

**References**


**See Also**

`Covmatrix, GaussRF, MaxStableRF`

**Examples**

```r
library(CompRandFld)
library(RandomFields)
library(mapproj)
library(fields)

# Define the spatial-coordinates of the points:
x <- runif(500, 0, 2)
y <- runif(500, 0, 2)
set.seed(261)
# Simulation of a spatial Gaussian random field:
data <- RFsim(x, y, corrm=matern, param=list(smooth=0.5,
            mean=0, sill=1, scale=0.2, nugget=0))$data
```
Example 2. Simulation of a binary random field based on the latent Gaussian random field with exponential correlation.

One spatial replication on a regular grid

# Define the spatial-coordinates of the points:
x <- seq(0, 1, 0.05)
y <- seq(0, 1, 0.05)
set.seed(251)

# Simulation of a spatial binary random field:
sim <- RFsim(x, y, corrmodel="exponential", grid=TRUE,
model="BinaryGauss", threshold=0,
param=list(nugget=0, mean=0, scale=1, sill=1))
image(x,y,sim$data,col=terrain.colors(100))

Example 3. Simulation of a max-stable random extremal-t type with exponential correlation.

One spatial replication on a regular grid

set.seed(341)
x <- seq(0, 1, 0.02)
y <- seq(0, 1, 0.02)

# Simulation of a spatial binary random field:
sim <- RFsim(x, y, corrmodel="exponential", grid=TRUE, model="ExtT",
numblock=500, param=list(nugget=0, mean=0, scale=1,
sill=1, df=5))
image.plot(x,y,log(sim$data))

Example 4. Simulation of a Gaussian random field.

with double exponential correlation.

One spatio-temporal replication.

# Define the spatial-coordinates of the points:
x <- seq(0, 1, 0.1)
y <- seq(0, 1, 0.1)
# Define the temporal-coordinates:
times <- seq(1, 3, 1)
#
# Simulation of a spatial Gaussian random field:
sim <- RFsim(x, y, times, corrmobile="exp", grid=TRUE,
            param=list(nugget=0, mean=0, scale_s=0.3,
                        scale_t=0.5, sill=1))$data
# Spatial simulated data at first temporal instant
sim[,1]

--------------------------------------------------------------------------------
### Example 5. Simulation of a Gaussian random field
### with exponential correlation on a portion of the earth surface
### One spatial replication.
###
lon_region<-c(-40,40)
lat_region<-c(-40,40)
#
lon<-seq(min(lon_region),max(lon_region),2)
lat<-seq(min(lat_region),max(lat_region),2)
#
data<-RFsim(coordx=lon,coordy=lat,corrmobile="exponential",
            distance="Geod",grid=TRUE,param=list(nugget=0,mean=0
            ,scale=8000,sill=1))$data
image.plot(lon,lat,data,xlab="Longitude",ylab="Latitude")
map(database="world",xlim=lon_region,ylim=lat_region,add=TRUE)

Gauging Stations of the Annual Maximum Rainfalls in U.S.

Description

A (46 × 3)-matrix containing the coordinates of the 46 gauging stations in U.S. where the daily precipitation are recorded.

Usage

data(usrain)

Format

A numerical matrix containing the longitude, the latitude and the altitude of the gauging stations.
**Source**


---

**usrain**

*Annual Maximum Rainfalls in U.S.*

**Description**

A $(91 \times 46)$-matrix containing annual maximum of daily rainfalls, in millimeters, for 91 years at 46 gauging stations in U.S.

**Usage**

`data(usrain)`

**Format**

A numerical matrix containing 4186 observations.

**Source**


---

**winds**

*Irish Daily Wind Speeds*

**Description**

A matrix containing daily wind speeds, in kilometers per hour, from 1961 to 1978 at 12 sites in Ireland.

**Usage**

`data(irishwinds)`

**Format**

A $(6574 \times 11)$-matrix containing wind speed observations.

**Source**

winds.coords

Weather Stations of the Irish Daily Wind Speeds

Description

A data frame containing information about the weather stations where the data are recorded in Ireland.

Usage

data(irishwinds)

Format

A data frame containing site - the name of the city (character), abbr - the abbreviation (character), elev - the elevation (numeric), lat - latitude (numeric) and lon - longitude.

Source


wleastsquare

WLS of Gaussian and Max-Stable Random Fields

Description

the function returns the parameters’ estimates and the estimates’ variances of a random field obtained by the weighted least squares estimator.

Usage

wleastsquare(data, coordx, coordy=NULL, coordt=NULL, corrmatrix, distance="Eucl", fixed=NULL, grid=FALSE, maxdist=NULL, maxtime=NULL, model='Gaussian', optimizer='Nelder-Mead', numbins=NULL, replicates=1, start=NULL, weighted=FALSE)

Arguments

data A \(d\)-dimensional vector (a single spatial realisation) or a \((n \times d)\)-matrix (\(n\) iid spatial realisations) or a \((d \times d)\)-matrix (a single spatial realisation on regular grid) or an \((d \times d \times n)\)-array (\(n\) iid spatial realisations on regular grid) or a \((t \times d)\)-matrix (a single spatial-temporal realisation) or an \((t \times d \times n)\)-array (\(n\) iid spatial-temporal realisations on regular grid) or a \((d \times d \times t \times n)\)-array (a single spatial-temporal realisation on regular grid) or an \((d \times d \times t \times n)\)-array (\(n\) iid spatial-temporal realisations on regular grid). See FitComposite for details.
coordx | A numeric \((d \times 2)\)-matrix (where \(d\) is the number of spatial sites) giving 2-dimensions of spatial coordinates or a numeric \(d\)-dimensional vector giving 1-dimension of spatial coordinates.

coordy | A numeric vector giving 1-dimension of spatial coordinates; coordy is interpreted only if coordx is a numeric vector or grid=TRUE otherwise it will be ignored. Optional argument, the default is NULL then coordx is expected to be numeric a \((d \times 2)\)-matrix.

coordt | A numeric vector giving 1-dimension of temporal coordinates. Optional argument, the default is NULL then a spatial random field is expected.

corrmodel | String; the name of a correlation model, for the description (see FitComposite).

distance | String; the name of the spatial distance. The default is Eucl, the euclidean distance. See the Section Details of FitComposite.

fixed | A named list giving the values of the parameters that will be considered as known values. The listed parameters for a given correlation function will be not estimated, i.e. if list(nugget=0) the nugget effect is ignored.

grid | Logical; if FALSE (the default) the data are interpreted as a vector or a \((n \times d)\)-matrix, instead if TRUE then \((d \times d \times n)\)-matrix is considered.

maxdist | A numeric value denoting the maximum distance, see Details and FitComposite.

maxtime | Numeric; an optional positive value indicating the maximum temporal lag considered in the composite-likelihood computation (see FitComposite).

model | String; the type of random field. Gaussian is the default, see FitComposite for the different types.

optimizer | String; the optimization algorithm (see optim for details). 'Nelder-Mead' is the default.

numbins | A numeric value denoting the numbers of bins, see the Section Details.

replicates | Numeric; a positive integer denoting the number of independent and identically distributed (iid) replications of a spatial or spatial-temporal random field. Optional argument, the default value is 1 then a single realisation is considered.

start | A named list with the initial values of the parameters that are used by the numerical routines in maximization procedure. NULL is the default (see FitComposite).

weighted | Logical; if TRUE then the weighted least square estimator is considered. If FALSE (the default) then the classic least square is used.

Details

The numbins parameter indicates the number of adjacent intervals to consider in order to grouped distances with which to compute the (weighted) least squares.

The maxdist parameter indicates the maximum distance below which the shorter distances will be considered in the calculation of the (weighted) least squares.

Value

Returns an object of class WLS. An object of class WLS is a list containing at most the following components:
bins
Adjacent intervals of grouped distances;
bint
Adjacent intervals of grouped temporal separations;
centers
The centers of the bins;
coordx
The vector or matrix of spatial coordinates;
coordy
The vector of spatial coordinates;
coordt
The vector of temporal coordinates;
convergence
A string that denotes if convergence is reached;
corrmodel
The correlation model;
data
The vector or matrix of data;
distance
The type of spatial distance;
fixed
The vector of fixed parameters;
iterations
The number of iteration used by the numerical routine;
message
Extra message passed from the numerical routines;
model
The type of random fields;
numcoord
The number of spatial coordinates;
numrep
The number of the iid replicatations of the random field;
umtime
The number the temporal realisations of the random field;
param
The vector of parameters’ estimates;
srange
The minimum and maximum spatial distance;
trange
The minimum and maximum temporal separations;
variograms
The empirical spatial variogram;
variogramt
The empirical temporal variogram;
variogramst
The empirical spatial-temporal variogram;
weighted
A logical value indicating if its the weighted method;
wls
The value of the least squares at the minimum.

Author(s)
Simone Padoan, <simone.padoan@unibocconi.it>, http://faculty.unibocconi.it/simonepadoan;
Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, https://sites.google.com/a/uv.cl/moreno-bevilacqua/home.

References
See Also

FitComposite, optim

Examples

library(CompRandFld)
library(RandomFields)
set.seed(2111)

# Set the coordinates of the sites:
x <- runif(100, 0, 10)
y <- runif(100, 0, 10)

# Set the model's parameters:
corrmodel <- "exponential"
mean <- 0
sill <- 1
nugget <- 0
scale <- 2

# Simulation of the Gaussian random field:
data <- RFsim(x, y, corrm=corrmodel, param=list(mean=mean,
sill=sill, nugget=nugget, scale=scale))

fix<-list(nugget=0)
ini<-list(scale=scale, sill=sill)
# Least square fitting of the random field:
fit <- WLeastSquare(data, x, y, corrm=corrmodel, fixed=fix, start=ini)

# Results:
print(fit)

# Example 2. Least square fitting of a max-stable random field
# (Extremal Gaussian model) with exponential correlation
# n iid spatial replications.
# Unweighted version (all weights equals to 1).
#
Computes Starting Values based on Weighted Least Squares

WlsInit

**Description**

Subroutine called by FitComposite. The function returns opportune starting values for the composite-likelihood fitting procedure based on weighted least squares.

**Usage**

```
WlsInit(coordx, coordy, coorbd, corrmodel, data, distance, fcall, fixed,
        grid, likelihood, margins, maxdist, maxtime, model, numblock,
        **par**)
```
\texttt{WlsInit}

\begin{verbatim}
param, parscale, paramrange, replicates, start, taper, tapsep, threshold, type, varest, vartype, weighted, winconst, winstp)
\end{verbatim}

\textbf{Arguments}

- \texttt{coordx} A numeric \((d\times2)\)-matrix (where \(d\) is the number of points) assigning 2-dimensions of coordinates or a numeric vector assigning 1-dimension of coordinates.

- \texttt{coordy} A numeric vector assigning 1-dimension of coordinates; \texttt{coordy} is interpreted only if \texttt{coordx} is a numeric vector otherwise it will be ignored.

- \texttt{coordt} A numeric vector assigning 1-dimension of temporal coordinates.

- \texttt{corrmodel} String; the name of a correlation model, for the description.

- \texttt{data} A numeric vector or a \((n \times d)\)-matrix or \((d \times d \times n)\)-matrix of observations.

- \texttt{distance} String; the name of the spatial distance. The default is \texttt{Eucl}, the euclidean distance. See the Section \texttt{Details}.

- \texttt{fcall} String; "fitting" to call the fitting procedure and "simulation" to call the simulation procedure.

- \texttt{fixed} A named list giving the values of the parameters that will be considered as known values.

- \texttt{grid} Logical; if \texttt{FALSE} (the default) the data are interpreted as a vector or a \((n \times d)\)-matrix, instead if \texttt{TRUE} then \((d \times d \times n)\)-matrix is considered.

- \texttt{likelihood} String; the configuration of the composite likelihood.

- \texttt{margins} String; the type of the marginal distribution of the max-stable field.

- \texttt{maxdist} Numeric; an optional positive value indicating the maximum spatial distance considered in the composite-likelihood computation.

- \texttt{maxtime} Numeric; an optional positive value indicating the maximum temporal separation considered in the composite-likelihood computation.

- \texttt{model} String; the name of the model. Here the default is \texttt{NULL}.

- \texttt{numblock} Numeric; the observation size of the underlying random field. Only in case of max-stable random fields and in the simulation.

- \texttt{param} A numeric vector of parameter values required in the simulation procedure of random fields.

- \texttt{parscale} A numeric vector with scaling values for improving the maximisation routine.

- \texttt{paramrange} A numeric vector with the range of the parameter space.

- \texttt{replicates} Logical; if \texttt{FALSE} (the default) one spatial random field is considered, instead if \texttt{TRUE} the data are considered as iid replicates of a field.

- \texttt{start} A numeric vector with starting values.

- \texttt{taper} String; the name of the type of covariance matrix. It can be \texttt{Standard} (the default value) or \texttt{Tapering} for taperd covariance matrix.

- \texttt{tapsep} Numeric; an optional value indicating the separabe parameter in the space time quasi taper (see \texttt{Details}).

- \texttt{threshold} Numeric; a value indicating a threshold for the binary random field.

- \texttt{type} String; the type of estimation method.
varest Logical; if TRUE the estimates' variances and standard errors are returned. FALSE is the default.

vartype String; the type of estimation method for computing the estimate variances, see the Section Details.

weighted Logical; if TRUE the likelihood objects are weighted, see FitComposite.

winconst Numeric; a positive real value indicating the window size used from the sub-sampling method for the estimation of the parameters variances.

winstp Numeric; a value in (0, 1] for defining the window step. See FitComposite.

Author(s)

Simone Padoan, <simone.padoan@unibocconi.it>, http://faculty.unibocconi.it/simonepadoan;
Moreno Bevilacqua, <moreno.bevilacqua@uv.cl>, https://sites.google.com/a/uv.cl/moreno-bevilacqua/home.

See Also

FitComposite, WLeastSquare.
Index

*Topic Composite
  CheckCorrModel, 2
  CheckInput, 3
  CheckLikelihood, 4
  CheckModel, 5
  CheckType, 6
  CheckVarType, 6
  CompLikelihood, 7
  CorrelationParam, 8, 15, 48
  Covariogram, 9
  Covmatrix, 2, 9, 14, 31, 32, 36, 48, 49, 56, 57
  Dist2Dist, 21
  EVariogram, 10, 23
  FitComposite, 3–11, 15, 19, 23, 26, 28, 43, 46, 47, 54–57, 61, 62, 64, 67
  FitGev, 22, 39, 54
  GaussRF, 36, 56, 57
  GevLogLik, 40, 41, 54
  HypoTest, 41
  InitParam, 45
  Kri, 15, 19, 47
  Likelihood, 52
  NuisanceParam, 55

*Topic LeastSquare
  WLeastSquare, 61
  WlsInit, 65

*Topic Simulation
  Covmatrix, 14
  RFSim, 55

*Topic Variogram
  EVariogram, 23

*Topic datasets
  us.coords, 59
  usrain, 60
  winds, 60
  winds.coords, 61

*Topic extremes
  Dist2Dist, 21
  FitGev, 40
  GevLogLik, 41
  MomEst, 54

*Topic spatial
  HypoTest, 41

CheckCorrModel, 2
CheckInput, 3
CheckLikelihood, 4
CheckModel, 5
CheckType, 6
CheckVarType, 6
CompLikelihood, 7
CorrelationParam, 8, 15, 48
Covariogram, 9
Covmatrix, 2, 9, 14, 31, 32, 36, 48, 49, 56, 57
Dist2Dist, 21
EVariogram, 10, 23
FitComposite, 3–11, 15, 19, 23, 26, 28, 43, 46, 47, 54–57, 61, 62, 64, 67
FitGev, 22, 39, 54
GaussRF, 36, 56, 57
GevLogLik, 40, 41, 54
HypoTest, 41
InitParam, 45
Kri, 15, 19, 47
Likelihood, 52
NuisanceParam, 55
optim, 4, 8, 29, 36, 40, 46, 53, 62, 64
print.FitComposite(FitComposite), 28
print.RFSim(RFSim), 55
print.WLS(WLeastSquare), 61
RFSim, 4, 19, 55
us.coords, 59
usrain, 60
winds, 60
winds.coords, 61
WLeastSquare, 10, 11, 36, 61, 67
WlsInit, 65