Package ‘CompRandFld’

November 8, 2010

Version 0.5

Date 2010-11-04

Title Composite-likelihood based Analysis of Random Fields

Author Simone Padoan <simone.padoan@epfl.ch>, Moreno Bevilacqua <moreno.bevilacqua@unive.it>

Maintainer Simone Padoan <simone.padoan@epfl.ch>

Depends R (>= 2.9.0)

Description The aim of this package is to collect a set of procedures for the analysis of Random Fields by Composite Likelihood methods. Spatial analysis often involves dealing with large dataset. Therefore even simple studies may be too computationally demanding. Composite likelihood based methods are emerging as useful tools for mitigating such computational problems and show satisfactory results when compared with other techniques such as, for example the tapering method. Moreover, composite likelihood (and related quantities) have some good properties similar to those of the standard likelihood.

Suggests RandomFields

URL http://people.epfl.ch/simone.padoan

R topics documented:

`CheckCorrModel` *Check of the Correlation Model*

Description

Subroutine called by InitParam. The procedure controls if the correlation model inserted has been implemented.

Usage

```
CheckCorrModel(corrmodel)
```

Arguments

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

Author(s)

Simone Padoan, `<simone.padoan@epfl.ch>`, `http://eflum.epfl.ch/people/padoan.en.php`.

See Also

`FitComposite`

`CheckInput` *Check of the input*

Description

Subroutine called by the fitting procedures. The procedure controls the input passed to the fitting procedures.

Usage

```
CheckInput(coordx, coordy, corrmodel, data, fixed,
           grid, likelihood, lonlat, model, optimizer,
           start, time, type, varest, vartype, weighted,
           weights, winconst)
```

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.
`corrmodel`	String; the name of a correlation model, for the description see `CovarianceFct`.
`data`	A numeric vector or a ($n \times d$)-matrix or ($d \times d \times n$)-matrix of observations.
`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.
`lonlat`	Logical; if `FALSE` (the default), `coordx` and `coordy` are interpreted as Cartesian coordinates otherwise they are considered as longitude and latitude.
`model`	String; the density associated to the likelihood objects. `Gaussian` is the default.
`optimizer`	String; the optimization algorithm (see `optim` for details). 'Nelder-Mead' is the default.
`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.
`time`	Logical; if `FALSE` (the default) a spatial random field is considered (one temporal realisation), if `TRUE` a spatial-temporal random field is considered.
`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.
`weights`	A numeric vector of weights.
`winconst`	Numeric; a positive real value – if `vartype=Sub-Samp` – that determines the window size in the sub-sampling estimates of the variances, see `FitComposite`.

Author(s)

Simone Padoan, `<simone.padoan@epfl.ch>`, http://eflum.epfl.ch/people/padoan.en.php.

See Also

`FitComposite`

CheckLikelihood *Check of the type of Composite-likelihood*

Description

Subroutine called by InitParam. The procedure controls the type of the composite-likelihood passed to the FitComposite procedure.

Usage

```
CheckLikelihood(likelihood)
```

Arguments

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

Author(s)

Simone Padoan, `<simone.padoan@epfl.ch>`, `http://eflum.epfl.ch/people/padoan.en.php`.

See Also

`FitComposite`

CheckModel *Check of the type of Random Field*

Description

Subroutine called by InitParam. The procedure controls the type of random field passed to the fitting procedures.

Usage

```
CheckModel(model)
```

Arguments

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

Author(s)

Simone Padoan, `<simone.padoan@epfl.ch>`, `http://eflum.epfl.ch/people/padoan.en.php`.

See Also

`FitComposite`

CheckParam — *Check of the Parameters*

Description

Subroutine called by InitParam. The procedure controls the validity of the correlation's parameters.

Usage

```
CheckParam(corrmodel, namesparam, numparam)
```

Arguments

`corrmodel`	String; the name of a correlation model.
`namesparam`	String; the names of the parameters.
`numparam`	Numeric; the number of the parameters.

Author(s)

Simone Padoan, `<simone.padoan@epfl.ch>`, `http://eflum.epfl.ch/people/padoan.en.php`.

See Also

`FitComposite`

CheckParamRange — *Check of the Parameters' Ranges*

Description

Subroutine called by the fitting procedures. The procedure controls the range of the correlation's parameters.

Usage

```
CheckParamRange(param)
```

Arguments

`param`	Numeric; a vector of correlation's parameters.

Author(s)

Simone Padoan, `<simone.padoan@epfl.ch>`, `http://eflum.epfl.ch/people/padoan.en.php`.

See Also

`FitComposite`

CheckType *Check of the type of likelihood objects*

Description

Subroutine called by InitParam. The procedure controls the type of likelihood objects that form the composite-likelihood .

Usage

```
CheckType(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@epfl.ch>, http://eflum.epfl.ch/people/padoan.en.php.

See Also

FitComposite

CheckVarType *Check of the method for the computation of the estimates' variances*

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@epfl.ch>, http://eflum.epfl.ch/people/padoan.en.php.

See Also

FitComposite

CompLikelihood *Computation of the Composite-likelihood*

Description

Subroutine called by OptimCompLik. The procedure computes the composite-likelihood for a given set of data and parameters.

Usage

```
CompLikelihood(coordx, coordy, corrmodel, data, dista, fixed, lags,
               likelihood, model, namescorr, namesnuis, numcoord,
               numdata, param, type)
```

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.
`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.
`dista`	A numeric value specifying the cut-off distance.
`fixed`	A numeric vector with the parameters that will be considered as known values.
`lags`	A numeric vector of distances between points.
`likelihood`	Numeric; the configuration of the composite likelihood, see `FitComposite`.
`model`	Numeric; the id of the random field.
`namescorr`	String; the names of the correlation parameters.
`namesnuis`	String; the names of the nuisance parameters.
`numcoord`	Numeric; the number of coordinates.
`numdata`	Numeric; the number of the data in time.
`param`	A numeric vector with the parameter values.
`type`	Numeric; the type of the likelihood objects, see `FitComposite`.

Author(s)

Simone Padoan, <simone.padoan@epfl.ch>, http://eflum.epfl.ch/people/padoan.en.php.

See Also

`FitComposite`

CorrelationFct — *Computation of the Correlation function*

Description

Subroutine called by Covariogram. The procedure computes the estimated correlation function for a given fitted model.

Usage

```
CorrelationFct(corrmodel, lags, numpairs, param)
```

Arguments

`corrmodel`	Numeric; the id of the correlation model.
`lags`	A numeric vector of distances between points.
`numpairs`	Numeric; the number of distances.
`param`	A numeric vector with the parameter values.

Author(s)

Simone Padoan, `<simone.padoan@epfl.ch>`, `http://eflum.epfl.ch/people/padoan.en.php`.

See Also

`FitComposite`

CorrelationParam — *Lists the Parameters of the Correlation function*

Description

Subroutine called by DetectParam and InitParam. The procedure returns the list of the parameter for a given correlation model.

Usage

```
CorrelationParam(corrmodel)
```

Arguments

`corrmodel`	String; the name of a correlation model.

Author(s)

Simone Padoan, `<simone.padoan@epfl.ch>`, `http://eflum.epfl.ch/people/padoan.en.php`.

See Also

`FitComposite`

`Covariogram` *Computation of the covariance function and the variogram*

Description

The procedure computes and plots the estimated covariance function and the variogram from a fitted model obtained fitting a random field with the composite-likelihood or using the weighted least square method.

Usage

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

Arguments

`fitted`	The fitted object obtained from the `FitComposite` procedure.
`lags`	A numeric vector of distances.
`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.range`	Logical; if `TRUE` the estimated pratical range is returned; if `FALSE` (the default) the pratical range is not returned.
`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.range`	Logical; if `TRUE` the estimated pratical range is added on the plot; if `FALSE` (the default) the pratical 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.
`pract.range`	Numeric; the percent of the sill to be reached.
`...`	other optional parameters which are passed to plot function.

Value

The returned object is a list with:

`covariance`	The vector of the estimated covariance function;
`variogram`	The vector of the estimated variogram function;
`pratical.range`	The estimated practial range.

Author(s)

Simone Padoan, <simone.padoan@epfl.ch>, http://eflum.epfl.ch/people/padoan.en.php.

References

Gaetan, C. and Guyon, X. (2009) Spatial Statistics and Modelling. *Spring Verlang, New York*.

Examples

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

# Set the coordinates of the points:
x <- runif(100, 0, 20)
y <- runif(100, 0, 20)


#################################################################
###
### Example 1. Plot of the estimated correlation function
### from a spatial realisation of a Gaussian random field.
###
###
################################################################

# Set the model's parameters:
corrmodel <- "stable"
mean <- 0
variance <- 3
nugget <- 1
scale <- 10
power <- 1.5

# Simulation of the Gaussian random field in the specified points:
sim <- GaussRF(x=x, y=y, model=corrmodel, grid=FALSE,
               param=c(mean, variance, nugget, scale, power))

# Maximum composite-likelihood fitting of the random field:
fit <- FitComposite(x, y, corrmodel, sim)

# Plot of the Correlation function:
par(mfrow=c(1,2))
Covariogram(fit, show.cov=TRUE, show.range=TRUE, show.vario=TRUE)
```

DetectParam *Identification of the Parameters of the Correlation function*

Description

Subroutine called by Covariogram and others The procedure returns a list with the correlation model and the list of parameters.

Usage

```
DetectParam(corrmodel, fixed, param)
```

Arguments

`corrmodel`	String; the name of a correlation model.
`fixed`	A numeric vector with the fixed parameters.
`param`	A numeric vector with the parameters.

Author(s)

Simone Padoan, `<simone.padoan@epfl.ch>`, `http://eflum.epfl.ch/people/padoan.en.php`.

See Also

`FitComposite`

`FitComposite` *Maximum Composite-likelihood Fitting of Random Fields*

Description

Maximum composite-likelihood fitting for random fields. The function returns the parameters' estimates and the estimates' variances of random fields obtained by maximisation of the composite-likelihood and allows to fix any of the parameters.

Usage

```
FitComposite(coordx, coordy=NULL, corrmodel, data, fixed=NULL,
             grid=FALSE, likelihood='Marginal', lonlat=FALSE,
             model='Gaussian', optimizer='Nelder-Mead', start=NULL,
             time=FALSE, type='Pairwise', varest=FALSE,
             vartype='SubSamp', weighted=FALSE, weights=NULL,
             winconst=NULL)
```

Arguments

`coordx`	A numeric (d×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.
`corrmodel`	String; the name of a correlation model, for the description see the Section **Details**.
`data`	A numeric vector or a $(n \times d)$-matrix or $(d \times d \times n)$-matrix of observations (see **Details**).
`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, see the Section **Details**.
`lonlat`	Logical; if `FALSE` (the default), `coordx` and `coordy` are interpreted as Cartesian coordinates otherwise they are considered as longitude and latitude.
`model`	String; the density 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.
`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 **Details**).
`time`	Logical; if `FALSE` (the default) a spatial random field is considered (one temporal realisation), if `TRUE` a spatial-temporal random field is considered, 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 estimate' variances and standard errors are returned. `FALSE` is the default.
`vartype`	String; (`Sub-Samp` 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.
`weights`	A numeric vector of weights (see **Details**).
`winconst`	Numeric; a positive real value – if `vartype=Sub-Samp` – that determines the window size in the sub-sampling estimates of the variances (see **Details**).

Details

The `corrmodel` parameter allows to select a specific correlation function for the random field. The implemented correlation models are:

1. `cauchy`;
2. `exponential`;
3. `gauss` (Gaussian);
4. `gencauchy` (generalised Cauchy);
5. `stable` (or powered exponential);
6. `whittlematern` (Whittle-Matern).

See for more details `CovarianceFct`.

With the `data` parameter:

- If a numeric vector, the data are interpreted as one spatial realisation;
- If a numeric $(n \times d)$-matrix, the columns represent the data observed at different points and the rows represent the data for different time steps.
- If a numeric $(d \times d \times n)$-matrix the data are observed at $(d \times d)$ points for `n` time steps.

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

1. `Conditional`, the composite-likelihood is formed by conditionals likelihoods (not implemented yet);
2. `Marginal`, the composite-likelihood is formed by marginals likelihoods;
3. `Full`, the composite-likelihood turns out to be the standard likelihood;

The `model` paramter represents the density function underlying the definition of the likelihoods which form the composite-likelihood. The settings alternatives are:

- `Gaussian`, the Gaussian density.

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 `time` parameter allows to specify the type of random field. If `FALSE` a spatial random field is considered, if `TRUE` a spatial-temporal random field is used. For the moment the case of i.i.d. time replications is implemented. Soon will be possible to specify also dependence structure for the temporal component.

The `type` parameter represents the type of likelihood used in the composite-likelihood definition. The settings alternatives are:

1. If the composite is formed by `Marginal` likelihoods:
   - If each likelihood is obtained by the `Gaussian` density then with:
     – `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.
2. If the `Full` likelihood is considered:
   - If the likelihood is obtained by the `Gaussian` density then with:
     – `Standard`, the likelihood used is the standard version;
     – `Restricted`, the likelihood used is the restricted version.

The `vartype` parameter – if the `varest` is `TRUE` – specifies the method used to compute the estimates' variances. The default `Sub-Samp` uses the Sub-Sampling method to estimate the variability matrix in the Godambe matrix. Other options are: `Theoretical` where for the variability matrix it is used the exact expression (for the moment it has been implemented only for the `Difference` likelihood, see the `type` field) and `Sampling` where the variability matrix in this case is estimated by the sample contro-part (available only for i.i.d. replicates on time, see the `time` field).

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

The `weights` parameter allows to weight the composite-likelihood by weights insert by the users. These do not imply any gain in efficiency of the maximum composite-likelihood estimator but still be a reasonable setting (not implemented yet!).

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 conicide with the Akaike information criterion;
`coord`	The vector of coordinates;
`convergence`	A string that denotes if convergence is reached;
`corrmodel`	The correlation model;
`data`	The vector or matrix of data;
`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;
`lonlat`	The type of coordinates;
`message`	Extra message passed from the numerical routines;
`model`	The density associated to the likelihood objects;
`param`	The vector of parameters' estimates;
`stderr`	The vector of standard errors;
`sensmat`	The sensitivity matrix;
`varcov`	The matrix of the variance-covariance of the estimates;
`varimat`	The variability matrix;
`type`	The type of the likelihood objects.

Author(s)

Simone Padoan, <simone.padoan@epfl.ch>, http://eflum.epfl.ch/people/padoan.en.php; Moreno Bevilacqua, <moreno.bevilacqua@unive.it>.

References

Harville, D. A. (1977) Maximum Likelihood Approaches to Variance Component Estimation and to Related Problems. *Journal of the American Statistical Association*, **72**, 320–338.

Varin, C. and Vidoni, P. (2005) A Note on Composite Likelihood Inference and Model Selection. *Biometrika*, **92**, 519–528.

Varin, C. (2008) On Composite Marginal Likelihoods. *Advances in Statistical Analysis*, **92**, 1–28.

Gaetan, C. and Guyon, X. (2009) Spatial Statistics and Modelling. *Spring Verlang, New York*.

Padoan, S. A. Ribatet, M and Sisson, S. A. (2010) Likelihood-Based Inference for Max-Stable Processes. *Journal of the American Statistical Association, Theory & Methods*, **105**, 263–277.

See Also

`CovarianceFct`, `WLeastSquare`, `optim`

Examples

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

# Set the coordinates of the points:
x <- runif(100, 0, 20)
y <- runif(100, 0, 20)


################################################################
###
### Example 1. Maximum composite-likelihood fitting of one
### spatial realisation of a Gaussian random field.
### Composite-likelihood setting: pairwise marginal likelihoods.
###
###############################################################

# Set the model's parameters:
corrmodel <- "stable"
mean <- 0
variance <- 1
nugget <- 0
scale <- 10
power <- 1.5

# Simulation of the Gaussian random field in the specified points:
sim <- GaussRF(x=x, y=y, model=corrmodel, grid=FALSE,
               param=c(mean, variance, nugget, scale, power))

# Maximum composite-likelihood fitting of the random field:
fit <- FitComposite(x, y, corrmodel, sim)

# Results:
print(fit)


################################################################
###
### Example 2. Maximum composite-likelihood fitting of one
### spatial realisation of a Gaussian random field.
### Composite-likelihood setting: difference likelihoods.
###
###############################################################

# Set the model's parameters:
corrmodel <- "stable"
mean <- 0
variance <- 1
nugget <- 0
scale <- 10
power <- 1.5

# Simulation of the Gaussian random field in the specified points:
sim <- GaussRF(x=x, y=y, model=corrmodel, grid=FALSE,
               param=c(mean, variance, nugget, scale, power))
```

```
# Maximum composite-likelihood fitting of the random field:
fit <- FitComposite(x, y, corrmodel, sim, type='Difference')

# Results:
print(fit)


################################################################
###
### Example 3. Maximum likelihood fitting of one
### spatial realisation of a Gaussian random field.
### Likelihood setting: restricted likelihood.
###
###############################################################

# Set the model's parameters:
corrmodel <- "stable"
mean <- 0
variance <- 1
nugget <- 0
scale <- 10
power <- 1.5

# Simulation of the Gaussian random field in the specified points:
sim <- GaussRF(x=x, y=y, model=corrmodel, grid=FALSE,
               param=c(mean, variance, nugget, scale, power))

# Maximum composite-likelihood fitting of the random field:
fit <- FitComposite(x, y, corrmodel, sim, likelihood='Full',
                    type='Restricted')

# Results:
print(fit)
```

InitParam *Initialization of the Fitting Procedures*

Description

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

Usage

```
InitParam(coordx, coordy, corrmodel, data, fixed, grid,
          likelihood, lonlat, model, parscale, paramrange,
          start, time, type, vartype)
```

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.

`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.
`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.
`lonlat`	Logical; if `FALSE` (the default), `coordx` and `coordy` are interpreted as Cartesian coordinates otherwise they are considered as longitude and latitude.
`model`	String; the density associated to the likelihood objects. `Gaussian` is the default.
`parscale`	A numeric vector of scaling factor to improve the maximizing procedure, see `optim`.
`paramrange`	A numeric vector of parameters ranges, see `optim`.
`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 **Details**).
`time`	Logical; if `FALSE` (the default) a spatial random field is considered (one temporal realisation), if `TRUE` a spatial-temporal random field is considered.
`type`	String; the type of the likelihood objects. If `Pairwise` (the default) then the marginal composite likelihood is formed by pairwise marginal likelihoods.
`vartype`	String; the type of estimation method for computing the estimate variances, see the Section **Details**.

Author(s)

Simone Padoan, `<simone.padoan@epfl.ch>`, `http://eflum.epfl.ch/people/padoan.en.php`.

See Also

`FitComposite`

`Likelihood` *Computation of the likelihood*

Description

Subroutine called by OptimLik. The procedure computes the likelihood function for a given set of data and parameters.

Usage

```
Likelihood(corrmodel, data, fixed, grid, lags, model,
           namescorr, namesnuis, numcoord, numdata,
           numpairs, param, type)
```

Arguments

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.
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.
lags	A numeric vector of distances between points.
model	String; the density associated to the likelihood objects. Gaussian is the default.
namescorr	String; the names of the correlation parameters.
namesnuis	String; the names of the nuisance parameters.
numcoord	Numeric; the number of coordinates;
numdata	Numeric; the number of data replications in time.
numpairs	Numeric; the number of pairwise points.
param	A numeric vector of parameters' values.
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@epfl.ch>, http://eflum.epfl.ch/people/padoan.en.php.

See Also

FitComposite

LogNormDen *Computation of the multivariate log-normal density*

Description

Subroutine called by the Likelihood procedure. The procedure compute the multivariate log-normal density for a given set of data and parameters.

Usage

```
LogNormDen(stdata, detvarcov, ivarcov, numcoord, type)
```

Arguments

stdata	A numeric vector $(d \times 1)$ of data.
detvarcov	Numeric; the determinant of the variance-covariance matrix.
ivarcov	The inverse of the variance-covariance matrix $(d \times d)$.
numcoord	The number of point's coordinates.
type	The numeric id denoting the type of likelihood.

Author(s)

Simone Padoan, <simone.padoan@epfl.ch>, http://eflum.epfl.ch/people/padoan.en.php.

See Also

FitComposite

OptimCompLik *Optimization of the Composite log-likelihood*

Description

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

Usage

```
OptimCompLik(coordx, coordy, corrmodel, data, flagcorr,
             flagnuis, fixed, grid, lags, likelihood,
             lonlat, lower, model, namescorr, namesnuis,
             namesparam, numcoord, numdata, numparam,
             numparamcorr, optimizer, param, type, upper,
             varest, vartype, weighted, winconst)
```

Arguments

`coordx`	A numeric (d×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.
`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.
`flagcorr`	A numeric vector of binary values denoting which paramerters of the correlation function will be estimated.
`flagnuis`	A numeric vector of binary values denoting which nuisance paramerters 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.
`lags`	A numeric vector of distances between points.
`likelihood`	String; the configuration of the compositelikelihood, see `FitComposite`.
`lonlat`	Logical; if `FALSE` (the default), `coordx` and `coordy` are interpreted as Cartesian coordinates otherwise they are considered as longitude and latitude.
`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.
numdata	Numeric; the number of data replications in time.
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.
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.
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 real value – if vartype=Sub-Samp – that determines the window size in the sub-sampling estimates of the variances, see FitComposite.

Author(s)

Simone Padoan, <simone.padoan@epfl.ch>, http://eflum.epfl.ch/people/padoan.en.php.

See Also

FitComposite

OptimLik *Optimization of the log-likelihood*

Description

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

Usage

```
OptimLik(corrmodel, data, fixed, grid, lags, lower, model,
         namescorr, namesnuis, namesparam, numcoord, numdata,
         numpairs, optimizer, param, varest, 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.
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.
lags	A numeric vector of distances between points.
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.
numdata	Numeric; the number of data replications in time.
numpairs	Numeric; the number of pairwise distances.
optimizer	String; the optimization algorithm (see optim for details). 'Nelder-Mead' is the default.
param	A numeric vector of parameters' values.
varest	Logical; if TRUE the estimate' variances and standard errors are returned. FALSE is the default.
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.

Author(s)

Simone Padoan, <simone.padoan@epfl.ch>, http://eflum.epfl.ch/people/padoan.en.php.

See Also

FitComposite

SetRangeParam *Identification of the Parameters' Range*

Description

Subroutine called by InitParam and the fitting procedures. The procedure returns the range of the parameters for a given vector of parameters.

Usage

```
SetRangeParam(namesparam, numparam)
```

Arguments

`namesparam`	String; the names of the parameters.
`numparam`	Numeric; the numer of parameters.

Author(s)

Simone Padoan, <simone.padoan@epfl.ch>, http://eflum.epfl.ch/people/padoan.en.php.

See Also

FitComposite

`WLeastSquare` *Weighted Least Square Estimation of Random Fields*

Description

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

Usage

```
WLeastSquare(coordx, coordy, corrmodel, data, fixed=NULL,
             grid=FALSE, lonlat=FALSE, maxdist=NULL,
             optimizer='Nelder-Mead', numbins=NULL, start=NULL,
             time=FALSE, weighted=FALSE)
```

Arguments

`coordx`	A numeric (d×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.
`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 (see `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.
`lonlat`	Logical; if `FALSE` (the default), `coordx` and `coordy` are interpreted as Cartesian coordinates otherwise they are considered as longitude and latitude.
`maxdist`	A numeric value denoting the maximum distance, see the Section **Details**.
`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**

`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`).
`time`	Logical; if `FALSE` (the default) a spatial random field is considered (one temporal realisation), if `TRUE` a spatial-temporal random field is considered (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

Insert description of `maxdist` and `numbins`.

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;
`coord`	The vector of coordinates;
`convergence`	A string that denotes if convergence is reached;
`corrmodel`	The correlation model;
`data`	The vector or matrix of data;
`fixed`	The vector of fixed parameters;
`iterations`	The number of iteration used by the numerical routine;
`message`	Extra message passed from the numerical routines;
`param`	The vector of parameters' estimates;
`variogram`	The empirical variogram.

Author(s)

Simone Padoan, <simone.padoan@epfl.ch>, http://eflum.epfl.ch/people/padoan.en.php; Moreno Bevilacqua, <moreno.bevilacqua@unive.it>.

References

Cressie, N. A. C. (1993) Statistics for Spatial Data. New York: Wiley.

Barry, J. T., Crowder, M. J. and Diggle, P. J. (1997) Parametric estimation of the variogram. Tech. Report, Dept Maths & Stats, Lancaster University.

Gaetan, C. and Guyon, X. (2009) Spatial Statistics and Modelling. *Spring Verlang, New York*.

See Also

`FitComposite`, `optim`

Examples

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

# Set the coordinates of the sites:
x <- runif(100, 0, 20)
y <- runif(100, 0, 20)


################################################################
###
### Example 1. Least square fitting of one
### spatial realisation of a Gaussian random field.
### Non weighted version (all weights equals to 1)
###
###############################################################

# Set the model's parameters:
corrmodel <- "stable"
mean <- 0
variance <- 1
nugget <- 0
scale <- 10
power <- 1.5

# Simulation of the Gaussian random field in the specified points:
sim <- GaussRF(x=x, y=y, model=corrmodel, grid=FALSE,
              param=c(mean, variance, nugget, scale, power))

# Least square fitting of the random field:
fit <- WLeastSquare(x, y, corrmodel, sim)

# Results:
print(fit)


################################################################
###
### Example 1. Weighted least square fitting of one
### spatial realisation of a Gaussian random field.
### Weighted version.
###
###############################################################

# Set the model's parameters:
corrmodel <- "stable"
mean <- 0
variance <- 1
nugget <- 0
scale <- 10
power <- 1.5

# Simulation of the Gaussian random field in the specified points:
sim <- GaussRF(x=x, y=y, model=corrmodel, grid=FALSE,
              param=c(mean, variance, nugget, scale, power))
```

```
# Least square fitting of the random field:
fit <- WLeastSquare(x, y, corrmodel, sim, weighted=TRUE)

# Results:
print(fit)
```

Wls *Computation of the Weighted Least Squares*

Description

Subroutine called by WLeastSquare. The procedure computes the weighted least squares for a given set of data and parameters.

Usage

```
Wls(bins, corrmodel, fixed, lenbins, moments, numbins, param, weighted)
```

Arguments

`bins`	A numeric vector with adjacent intervals of grouped distances
`corrmodel`	Numeric; the id of the correlation mode.
`fixed`	A named list giving the values of the parameters that will be considered as known values.
`lenbins`	A numeric vector with the number of observations that fall in each bin.
`moments`	A vector with the estimated means.
`numbins`	Numeric; the number of bins.
`param`	A numeric vector of parameters' values.
`weighted`	Logical; if `TRUE` the least squares are weighted.

Author(s)

Simone Padoan, `<simone.padoan@epfl.ch>`, http://eflum.epfl.ch/people/padoan.en.php.

See Also

`WLeastSquare`, `FitComposite`

WlsInit *Computation of Starting Values based on Weighted Least Squares*

Description

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

Usage

```
WlsInit(coordx, coordy, corrmodel, data, fixed, grid,
        likelihood, lonlat, model, parscale, paramrange,
        start, time, type, vartype)
```

Arguments

`coordx`	A numeric (d×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.
`corrmodel`	String; the name of a correlation model, for the description.
`data`	A numeric vector or a $(n \times d)$-matrix or $(d \times d \times n)$-matrix of observations.
`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.
`lonlat`	Logical; if `FALSE` (the default), `coordx` and `coordy` are interpreted as Cartesian coordinates otherwise they are considered as longitude and latitude.
`model`	String; the name of the model. Here the default is `NULL`.
`parscale`	A numeric vector with scaling values for improving the maximisation routine.
`paramrange`	A numeric vector with the range of the parameter space.
`start`	A numeric vector with starting values.
`time`	Logical; if `FALSE` (the default) a spatial random field is considered (one temporal realisation), if `TRUE` a spatial-temporal random field is considered.
`type`	String; the type of estimation method.
`vartype`	String; the type of estimation method for computing the estimate variances, see the Section **Details**.

Author(s)

Simone Padoan, `<simone.padoan@epfl.ch>`, `http://eflum.epfl.ch/people/padoan.en.php`.

See Also

`FitComposite`, `WLeastSquare`.

Index