

Package ‘gamlss.spatial’

February 17, 2018

Type Package

Title Spatial Terms in Generalized Additive Models for Location Scale and Shape Models

Version 1.3.5

Date 2018-02-15

Description It allows us to fit Gaussian Markov Random Field within the Generalized Additive Models for Location Scale and Shape algorithms.

License GPL-2 | GPL-3

URL <http://www.gamlss.org/>

Depends R (>= 2.15.0), gamlss.dist, gamlss (>= 4.2-7), gamlss.add, spam, mgcv

Imports stats, grDevices, graphics, methods

Suggests knitr, spdep, maptools,

Repository CRAN

NeedsCompilation no

Author Fernanda De Bastiani [aut, cre, cph],
Mikis Stasinopoulos [aut],
Robert Rigby [aut]

Maintainer Fernanda De Bastiani <fernandadebastiani@gmail.com>

RoxygenNote 5.0.1

Date/Publication 2018-02-17 12:30:09 UTC

R topics documented:

gamlss.spatial-package	2
draw.polys	3
gamlss.gmrf	5
MRF	7

Index	10
--------------	-----------

gamlss.spatial-package

Spatial models within gamlss

Description

This package provides a set of functions to facilitate the fitting of spatial models within GAMLSS. At the moment it allows only Gaussian Markov Random Fiels (GMRF) but other facilities will be added soon.

Details

Package: gamlss.spatial
Type: Package
Version: 1.0
Date: 2013-10-21
License: GPL-2 | GPL-3

Author(s)

Fernanda De Bastiani, Mikis Stasinopoulos, Robert Rigby and Vlasios Voudouris.

Maintainer: Fernanda <fernandadebastiani@gmail.com>

References

Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, **54**, part 3, pp 507-554.

Rue and Held (2005) *Gaussian markov random fields: theory and applications*, Chapman & Hall, USA.

Stasinopoulos D. M., Rigby R.A. and Akantziliotou C. (2006) Instructions on how to use the GAMLSS package in R. Accompanying documentation in the current GAMLSS help files, (see also <http://www.gamlss.org/>).

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <http://www.jstatsoft.org/v23/i07>.

Examples

```
library(mgcv)
data(columb)
data(columb.polys)
m1 <- MRFA(columb$crime, columb$district, polys=columb.polys)
draw.polys(columb.polys, m1)
```

Description

This set of functions are useful to get information and to plot maps.

Usage

```
draw.polys(polys, object = NULL, scheme = NULL,
           swapcolors = FALSE, n.col = 100, ...)
polys2nb(polys)
nb2prec(neighbour, x, area=NULL)
polys2polys(object, neighbour.nb)
nb2nb(neighbour.nb)
```

Arguments

polys	an object containing the polygon information for the area
object	are either the values to plot in the draw.polys() function or a polygons information for a shape file for function polys2polys
scheme	scheme of colours to use, it can be "heat", "rainbow", "terrain", "topo", "cm" or any colour
swapcolors	to reverse the colours, it just work for "heat", "rainbow", "terrain", "topo", "cm" options
n.col	range for the colours
neighbour.nb	neighbour information for a shape file for function nb2nb
neighbour	the neighbour information, and if the neighbour is from S4 shape file than use nb2nb to transfer it to the appropriate neighbour for MRF(), MRFA(), mrf() and mrfa().
x	the factor defining the areas
area	all possible areas involved
...	for extra options

Details

draw.polys() plots the fitted values of fitted MRF object.

polys2nb() gets the neighbour information from the polygons.

nb2prec() creates the precision matrix from the neighbour information.

polys2polys() transforms a shape file polygons (S4 object) to the polygons required form for the functions MRF() and MRFA().

nb2nb() transforms from a shape file neighbour (S4 object) to the neighbour required form for functions MRF().

Value

The `draw.polys()` produces a plot while the rest of the functions produce required object for fitting or plotting.

Author(s)

Fernanda De Bastiani, Mikis Stasinopoulos, Robert Rigby and Vlasios Voudouris
 Maintainer: Fernanda <fernandadebastiani@gmail.com>

References

- Stasinopoulos, D. M., Rigby, R. A., Heller, G. Z., Voudouris, V. and De Bastiani, F. (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*. Chapman and Hall, Boca Raton.
- De Bastiani, F. Rigby, R. A., Stasinopoulos, D. M., Cysneiros, A. H. M. A. and Uribe-Opazo, M. A. (2016) Gaussian Markov random eld spatial models in GAMLSS. *Journal of Applied Statistics*, pp 1-19.
- Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, **54**, part 3, pp 507-554.
- Rue and Held (2005) *Gaussian Markov random fields: theory and applications*, Chapman & Hall, USA.
- Stasinopoulos D. M., Rigby R.A. and Akantziliotou C. (2006) Instructions on how to use the GAMLSS package in R. Accompanying documentation in the current GAMLSS help files, (see also <http://www.gamlss.org/>).
- Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <http://www.jstatsoft.org/v23/i07>.

See Also

[MRF](#), [MRFA](#)

Examples

```
# bringing the required libraries
library(spdep)
library(maptools)
# reading the shape file from package spdep
bh <- readShapePoly(system.file("etc/shapes/bhcv.shp",
                             package="spdep")[1])
# pick up parts of the data and scale them
BhData <- data.frame(scale(bh@data[,5:8]))
# getting the neighbourhood and the polygons using the package
# spdep functions
bh.nb <- poly2nb(bh) # neighbourhood
bh.polys <- bh@polygons # polygons
# now getting the information for the S4 object to required format
# from object S4 to object S3
newpolys <- polys2polys(bh.polys,bh.nb)
```

```

newnb <- nb2nb(bh.nb)
# drawing the map
draw.polys(newpolys[[1]])
# plotting one of the variables in BhData
poo <-BhData$HLCI
names(poo) <- row.names(BhData)
draw.polys(newpolys[[1]], poo)
# now get the precision matrix
Prec <- nb2prec(newnb, x=as.factor(row.names(BhData)),
               area=as.factor(row.names(BhData)))

```

gamlss.gmrf

*Gaussian Markov Random Field fitting within GAMLSS***Description**

The function `gmrf()` can be used to fit Markov Random Field additive terms within GAMLSS.

Usage

```

gamlss.gmrf(x, y, w, xeval = NULL, ...)
gmrf(x, precision = NULL, neighbour = NULL, polys = NULL,
     area = NULL, adj.weight = 1000, df = NULL, lambda =
     NULL, start = 10, method = c("Q", "A"), control =
     gmrf.control(...), ...)

```

Arguments

<code>x</code>	a factor containing the areas
<code>precision</code>	the precision matrix if set
<code>neighbour</code>	an object containing the neighbour information for the area if set
<code>polys</code>	the polygon information if set
<code>area</code>	this argument is here to allow more areas than the levels of the factor <code>x</code> , see example below
<code>adj.weight</code>	a value to adjust the iterative weight if necessary
<code>df</code>	degrees of freedom for fitting if required, only for <code>method="A"</code>
<code>lambda</code>	The smoothing parameter <code>lambda</code> if known, only for <code>method="A"</code>
<code>start</code>	starting value for the smoothing parameter <code>lambda</code>
<code>method</code>	"Q" for Q-function, or "A" for alternating method
<code>y</code>	working response variable
<code>w</code>	iterative weights
<code>xeval</code>	whether to predict or not
<code>control</code>	to be use for some of the argument of <code>MRF()</code> .
<code>...</code>	for extra arguments

Details

The function `gmrf()` is to support the function `MRF()` and `MRFA()` within GAMLSS. It is intended to be called within a GAMLSS formula. The function `gmrf()` is not intended to be used directly. It is calling the function `MRFA()` and `MRF()` within the GAMLSS fitting algorithm. The results using the option `method="Q"` or `method="A"` should produce identical results.

Value

a fitted `gamlss` object

Author(s)

Fernanda De Bastiani, Mikis Stasinopoulos, Robert Rigby and Vlasios Voudouris.

Maintainer: Fernanda <fernandadebastiani@gmail.com>

References

Stasinopoulos, D. M., Rigby, R. A., Heller, G. Z., Voudouris, V. and De Bastiani, F. (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*. Chapman and Hall, Boca Raton. (see also <http://www.gamlss.org/>)

De Bastiani, F. Rigby, R. A., Stasinopoulos, D. M., Cysneiros, A. H. M. A. and Uribe-Opazo, M. A. (2016) Gaussian Markov random spatial models in GAMLSS. *Journal of Applied Statistics*, pp 1-19.

Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, **54**, part 3, pp 507-554.

Rue and Held (2005) *Gaussian markov random fields: theory and applications*, Chapman & Hall, USA.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <http://www.jstatsoft.org/v23/i07>.

See Also

[MRF](#), [MRFA](#)

Examples

```
library(gamlss)
library(mgcv)
data(columb)
data(columb.polys)
vizinhos=polys2nb(columb.polys)
precisionC <- nb2prec(vizinhos,x=columb$district)
# MRFA
m1<- gamlss(crime~ gmrf(district, polys=columb.polys, method="Q"), data=columb)
m2<- gamlss(crime~ gmrf(district, polys=columb.polys, method="A"), data=columb)
AIC(m1,m2, k=0)
draw.polys(columb.polys, getSmo(m2), scheme="topo")
```

Description

The functions `MRF()` and `MRFA()` are used to fit a Gaussian Markov Random Fields (MRF) model. They are used by the functions `mrf()` and `mrfa()` respectively to fit a MRF additive term within GAMLSS

Usage

```
MRF(y, x, precision = NULL, neighbour = NULL, polys = NULL,
     area = NULL, weights = rep(1, length(y)), sig2e = 1,
     sig2b = 1, sig2e.fix = FALSE,
     sig2b.fix = FALSE, penalty = FALSE,
     delta = c(0.01, 0.01), shift = c(0, 0))
```

```
MRFA(y, x, precision = NULL, neighbour = NULL, polys = NULL,
     area = NULL, weights = rep(1, length(y)),
     lambda = NULL, df = NULL, start = 10)
```

Arguments

<code>y</code>	response variable
<code>x</code>	a factor containing the areas
<code>precision</code>	the precision matrix if set
<code>neighbour</code>	an object containing the neighbour information for the area if set
<code>polys</code>	the polygon information if set
<code>area</code>	this argument is here to allow more areas than the levels of the factor <code>x</code> , see example below.
<code>weights</code>	prior weights
<code>sig2e</code>	starting values for the error variance
<code>sig2b</code>	starting values for the random field variance
<code>sig2e.fix</code>	whether <code>sig2e</code> is fixed in the fitting, default equals <code>FALSE</code>
<code>sig2b.fix</code>	whether <code>sig2B</code> is fixed in the fitting, default equals <code>FALSE</code>
<code>penalty</code>	whether quadratic penalty is required to help convergence in for flat likelihoods, this is equivalent of putting a normal prior distribution for the log-sigmas e.g. <code>logsig2e ~ N(shift, 1/delta)</code>
<code>delta</code>	the precision of the prior
<code>shift</code>	the mean of the prior
<code>lambda</code>	smoothing parameter for MRFA function
<code>start</code>	starting value for the smoothing parameter <code>lambda</code> for MRFA function
<code>df</code>	for fixing the degrees of freedom (only in <code>MRFA()</code>)

Details

There are two functions for fitting Markov random fields: i) `MRF()` which uses the Q-function (marginal likelihood) for estimating the `sig2e` and `sig2b` parameters and ii) `MRFA()` which estimates the smoothing parameter $\lambda = \text{sig2e}/\text{sig2b}$ using the "alternating" method.

Value

a fitted MRF object

Author(s)

Fernanda De Bastiani, Mikis Stasinopoulos, Robert Rigby and Vlasios Voudouris.

Maintainer: Fernanda <fernandadebastiani@gmail.com>

References

Stasinopoulos, D. M., Rigby, R. A., Heller, G. Z., Voudouris, V. and De Bastiani, F. (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*. Chapman and Hall, Boca Raton. (see also <http://www.gamlss.org/>)

De Bastiani, F. Rigby, R. A., Stasinopoulos, D. M., Cysneiros, A. H. M. A. and Uribe-Opazo, M. A. (2016) Gaussian Markov random field spatial models in GAMLSS. *Journal of Applied Statistics*, pp 1-19.

Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, **54**, part 3, pp 507-554.

Rue and Held (2005) *Gaussian markov random fields: theory and applications*, Chapman & Hall, USA.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <http://www.jstatsoft.org/v23/i07>.

See Also

[mrf](#)

Examples

```
library(mgcv)
data(columb)
data(columb.polys)
vizinhos=polys2nb(columb.polys)
precisionC <- nb2prec(vizinhos,x=columb$district)
# MRFA
m1<-MRFA(columb$crime, columb$district, polys=columb.polys)
m11<-MRFA(columb$crime, columb$district, precision=precisionC)
m12<-MRFA(columb$crime, columb$district, neighbour=vizinhos)
draw.polys(columb.polys, m12, scheme="heat",swapcolors=TRUE)
## Not run:
# MRF
```



```
m2<-MRF(columb$crime, columb$district, polys=columb.polys)
m21<-MRF(columb2$crime, columb2$district, precision=precisionC)
m22<-MRF(columb2$crime, columb2$district, neighbour=vizinhos)
AIC(m1, m11,m12,m2, m21, m22, k=0)
draw.polys(columb.polys, m12, scheme="heat",swapcolors=TRUE)
# removing one area
columb2 <- columb[-5,]
# creating new precision matrix
precisionC2 <- nb2prec(vizinhos,x=columb2$district,area=columb2$district)
# MRFA
# new data but declaring area
m11<-MRFA(columb2$crime, columb2$district, polys=columb.polys, area=columb2$district)
# new data old polys
m112<-MRFA(columb2$crime, columb2$district, polys=columb.polys)
# new data old precision old area
m111<-MRFA(columb2$crime, columb2$district, precision=precisionC,area=columb2$district)
# new data old neighbour old area
m121<-MRFA(columb2$crime, columb2$district, neighbour=vizinhos,area=columb2$district)
# new data new precision old area
m113<-MRFA(columb2$crime, columb2$district, precision=precisionC2,area=columb2$district)
AIC(m11,m112,m111,m121,m113, k=0)
m11<-MRFA(columb2$crime, columb2$district, polys=columb.polys, area=columb2$district)
# new data old polys
m112<-MRFA(columb2$crime, columb2$district, polys=columb.polys)
# new data old precision old area
m111<-MRFA(columb2$crime, columb2$district, precision=precisionC,area=columb2$district)
# new data old neighbour old area
m121<-MRFA(columb2$crime, columb2$district, neighbour=vizinhos,area=columb2$district)
# new data new precision old area
m113<-MRFA(columb2$crime, columb2$district, precision=precisionC2,area=columb2$district)
AIC(m11,m112,m111,m121,m113, k=0)
draw.polys(columb.polys, fitted(m11))

## End(Not run)
```

Index

- *Topic **package**
 - gamlss.spatial-package, 2
- *Topic **regression**
 - draw.polys, 3
 - gamlss.gmrf, 5
 - MRF, 7
- *Topic **spatial**
 - draw.polys, 3
 - gamlss.gmrf, 5

draw.polys, 3

gamlss.gmrf, 5

gamlss.spatial

- (gamlss.spatial-package), 2

gamlss.spatial-package, 2

gmrf (gamlss.gmrf), 5

MRF, 4, 6, 7

mrf, 8

MRFA, 4, 6

MRFA (MRF), 7

nb2nb (draw.polys), 3

nb2prec (draw.polys), 3

polys2nb (draw.polys), 3

polys2polys (draw.polys), 3