Package 'scp'

May 8, 2020

Type Package

Title Spatial Conformal Prediction

Version 0.1.0

Author Huiying Mao <hmao@samsi.info>, Brian Reich <bjreich@ncsu.edu>, Ryan Martin <rgmarti3@ncsu.edu>

Maintainer Huiying Mao <hmao@samsi.info>

Description Provide robust, valid, and model-free spatial prediction intervals using Spatial Conformal Prediction (SCP) algorithms

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

URL https://github.com/mhuiying/scp

BugReports https://github.com/mhuiying/scp/issues

Suggests knitr,
rmarkdown,
covr,
testthat

VignetteBuilder knitr

Imports geoR

Depends R (¿= 2.10)

R topics documented:

krige_pred *Kriging prediction function*

Description

This function provides an example for pred_fun in scp, plausibility, and plausibility_contour, which provides a point prediction for location s0 (and corresponding standard error), given obserations s and Y.

Usage

```
krige_pred(s0, s, Y, return_sd = FALSE)
```

Arguments

s0	prediction location, a numeric vector with length = 2.
s	an $n \times 2$ matrix or data.frame with two coordinates of n locations.
Y	a vector with n values corresponding to Y(s).
return_sd	logical. if TRUE, Krige_pred function returns the standard error of Y(s0) along with the point prediction; if FALSE, Krige_pred function only returns the point prediction. Defaults to FALSE.

Value

If return_sd is FALSE (default), the output is a value of point prediction for Y(s0); If return_sd is TRUE, the output is a list with the following elements:

yhat	point prediction for Y(s0)
sd	standard error for Y(s0)

Examples

```
#?sample_data
s0 = c(0.5,0.5)
s  = sample_data$s
Y  = sample_data$Y

krige_pred(s0,s,Y)
krige_pred(s0,s,Y,return_sd=TRUE)
```

plausibility *calculate plausibility of* Y(s0) *being* Y0

Description

This function provides the plausibility of Y(s0) being Y0, given observations s and Y, using spatial conformal prediction algorithms.

Usage

```
plausibility(
  Y0,
  s0,
  s,
  Y,
  global = TRUE,
  eta = Inf,
  m = NULL,
  pred_fun = krige_pred,
  thetaHat = NULL,
  dfun = c("residual2", "std_residual2")
)
```

Arguments

Y0	a scalar or a vector
s0	prediction location, a numeric vector with length = 2 or a matrix with 1 row and 2 cols, or a data.frame with 1 row and 2 cordinates.
s	an $n \times 2$ matrix or a data.frame with two coordinates of n locations.
Y	a vector with n values corresponding to Y(s).
global	logical; if TRUE , scp function returns the result of global spatial conformal prediction (GSCP); if FALSE, scp function returns the result of local spatial conformal prediction (LSCP) and users need to specify eta < Inf or m $\leq$ n. Defaults to TRUE.
eta	kernel bandwidth for weight schema, a positve scalar with smaller value meaning more localized procedure. Defauls to Inf, which puts equal weight on surrounding m points.
m	an postive integer representing the number of nearest locations to use for prediction. Default to NULL. If global = TRUE, m = n; if global = FALSE and m is not specified, m would be determined by eta.
pred_fun	spatial prediction function with inputs being s0,s,Y and ouputs being predicted Y(s0) (and its standard error). Defaults to krige_pred.
thetaHat	a vector of Matern parameters, representing nugget, partial sill, range, and smoothness as in Mao. et al. (2020). Defaults to NULL. It will be ignored if pred_fun is not krige_pred.

`dfun` non-conformity measure with four options. In which, `"residual2"` (default) represents squared residual and `"std_residual2"` represents standardized squared residual.

Value

The output is a scalar or a vector with plausibility values for `Y0`. The numbers are between 0 and 1.

Author(s)

Huiying Mao, <hmao@samsi.info>, Brian Reich <bjreich@ncsu.edu>

References

to be entered

See Also

plausibility_contour

Examples

```
## To predict Y(s0), where s0 = c(0.5,0.5), using sample data
## What's the plausibility if Y(s0) = 0? Y(s0) = 1.5?

#?sample_data
s0 = c(0.5,0.5)
s  = sample_data$s
Y  = sample_data$Y

# plausibility for Y(s0) = 0
plausibility(Y0=0,s0=s0,s=s,Y=Y)

# plausibility for Y(s0) = 1.5
plausibility(Y0=1.5,s0=s0,s=s,Y=Y)

# plausibility for a sequence of Y0's
plausibility(Y0=seq(0,1,0.1),s0=s0,s=s,Y=Y)
```

---

plausibility_contour *generate plausibility contour*

Description

This function provides the plausibility contour for `Y(s0)`, given observations `s` and `Y`, using spatial conformal prediction algorithms.

Usage

```
plausibility_contour(
  s0,
  s,
  Y,
  global = TRUE,
  eta = Inf,
  m = NULL,
  pred_fun = krige_pred,
  thetaHat = NULL,
  dfun = c("residual2", "std_residual2"),
  precision = NULL
)
```

Arguments

s0	prediction location, a numeric vector with `length = 2` or a `matrix` with 1 row and 2 cols, or a data.frame with 1 row and 2 cordinates.
s	an $n \times 2$ `matrix` or a `data.frame` with two coordinates of n locations.
Y	a vector with n values corresponding to `Y(s)`.
global	logical; if `TRUE` , `scp` function returns the result of global spatial conformal prediction (GSCP); if `FALSE`, `scp` function returns the result of local spatial conformal prediction (LSCP) and users need to specify `eta < Inf` or `m` $\leq$ `n`. Defaults to `TRUE`.
eta	kernel bandwidth for weight schema, a positve scalar with smaller value meaning more localized procedure. Defauls to `Inf`, which puts equal weight on surrounding `m` points.
m	an postive integer representing the number of nearest locations to use for prediction. Default to `NULL`. If `global = TRUE`, `m = n`; if `global = FALSE` and `m` is not specified, `m` would be determined by `eta`.
pred_fun	spatial prediction function with inputs being `s0,s,Y` and ouputs being predicted `Y(s0)` (and its standard error). Defaults to `krige_pred`.
thetaHat	a vector of Matern parameters, representing nugget, partial sill, range, and smoothness as in Mao. et al. (2020). Defaults to `NULL`. It will be ignored if `pred_fun` is not `krige_pred`.
dfun	non-conformity measure with four options. In which, `"residual2"` (default) represents squared residual and `"std_residual2"` represents standardized squared residual.
precision	a positive scalar represents how dense `Y(s)` candidates (`Y_cand`) are. Defaults to `NULL`.

Value

The output is a "plausibility_contour" object.

Author(s)

Huiying Mao, <hmao@samsi.info>, Brian Reich <bjreich@ncsu.edu>

References

to be entered

See Also

plausibility

Examples

```
## generate plausibility contour for Y(s0), where s0 = c(0.5,0.5), using sample data

#?sample_data
s0 = c(0.5,0.5)
s  = sample_data$s
Y  = sample_data$Y

pc = plausibility_contour(s0=s0,s=s,Y=Y)
plot(pc)

idx = which(s[, 1] == s0[1] & s[, 2] == s0[2])
abline(v = Y[idx], col = "red", lty = 2)
legend("topright", col=1:2, lty=1:2, c("plausibility", "true value"))
```

---

scp *Spatial conformal prediction at input location(s)*

Description

This function provides the spatial conformal prediction interval for location(s) `s0`, given obserations `s` and `Y`.

Usage

```
scp(
  s0,
  s,
  Y,
  global = TRUE,
  eta = Inf,
  m = NULL,
  pred_fun = krige_pred,
  thetaHat = NULL,
  dfun = c("residual2", "std_residual2"),
```

```
  precision = NULL,
  alpha = 0.05
)
```

Arguments

s0	prediction location(s), a numeric vector with length = 2, or a matrix with ncol = 2, or a data.frame with two coordinates.
s	an $n \times 2$ matrix or a data.frame with two coordinates of n locations.
Y	a vector with n values corresponding to Y(s).
global	logical; if TRUE, scp function returns the result of global spatial conformal prediction (GSCP); if FALSE, scp function returns the result of local spatial conformal prediction (LSCP) and users need to specify eta. Defaults to TRUE.
eta	kernel bandwidth for weight schema, a positve scalar with smaller value meaning more localized procedure. Defauls to Inf, which puts equal weight on surrounding m points.
m	an postive integer representing the number of nearest locations to use for prediction. Default depands on eta.
pred_fun	spatial prediction function with inputs being $s0, s, Y$ and ouputs being predicted Y(s0) (and its standard error). Defaults to krige_pred representing Kriging prediction.
thetaHat	a vector of Matern parameters, representing nugget, partial sill, range, and smoothness as in Mao. et al. (2020). Defaults to NULL. It will be ignored if pred_fun is not krige_pred.
dfun	non-conformity measure with four options. In which, "residual2" (default) represents squared residual and "std_residual2" represents standardized squared residual.
precision	a positive scalar represents how dense the candidates for Y(s) are. Defaults to NULL.
alpha	significance level. Defaults to 0.05.

Value

The output is a data.frame of lower and upper bounds of the conformal prediction interval(s) for s0.

Author(s)

Huiying Mao, <hmao@samsi.info>, Brian Reich <bjreich@ncsu.edu>

References

to be entered

See Also

plausibility, plausibility_contour

Examples

```
## generate prediction interval for prediction locations(s) s0(s) using sample data

#?sample_data
s  = sample_data$s
Y  = sample_data$Y

# locations to predict
s0  = c(0.5,0.5)
s0s = rbind(c(0.4, 0.4), c(0.5,0.5), c(0.6, 0.6))

# default prediction interval
scp(s0=s0,s=s,Y=Y)
scp(s0=s0s,s=s,Y=Y)

# user define eta=0.1, where LSCP is considered
scp(s0=s0,s=s,Y=Y,eta=0.1)

# user define non-conformity measure
scp(s0=s0,s=s,Y=Y,dfun="std_residual2")

# user define prediction function
fun = function(s0,s,Y) return(mean(Y))
scp(s0=s0,s=s,Y=Y,pred_fun=fun)
```

`select_eta` *Select the optimal tunning parameter for* `scp`

Description

Select the optimal tunning parameter for `scp`

Usage

```
select_eta(s, Y, eta_cand = NULL, plot = TRUE)
```

Arguments

`s`	an $n \times 2$ `matrix` or a `data.frame` with two coordinates of n locations.
`Y`	a vector with n values corresponding to `Y(s)`.
`eta_cand`	a vector of candidate `eta` values. Defaults to `NULL`.
`plot`	logical; if `TRUE`, `select_eta` plots the empirical interval score for the candidate `eta` values. if `FALSE`, no plot will be generated. Defaults to `TRUE`.

Value

The optimum `eta` based on minimizing empirical interval score.

Examples

```
#?sample_data
s  = sample_data$s
Y  = sample_data$Y

# CAUTION: the following command may take one hour to run.
opt_eta = select_eta(s = s, Y = Y)

use optimal eta to calculate prediction interval
s0  = c(0.5,0.5)
scp(s0 = s0, s = s, Y = Y, eta = opt_eta)
```

Index