

# Package ‘scp’

April 27, 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** 7.0.2

**URL** <https://github.com/mhuiying/scp>

**BugReports** <https://github.com/mhuiying/scp/issues>

**Suggests** knitr,  
rmarkdown,  
covr,  
testthat

**VignetteBuilder** knitr

**Imports** geoR

## R topics documented:

krige_pred . . . . .	2
plausibility . . . . .	3
plausibility_contour . . . . .	4
scp . . . . .	6

<b>Index</b>	<b>9</b>
--------------	----------

---

krige_pred	<i>Kriging prediction function</i>
------------	------------------------------------

---

## 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 observations `s` and `Y`.

## Usage

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

## Arguments

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

<code>yhat</code>	point prediction for $Y(s0)$
<code>sd</code>	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	<i>calculate plausibility for <math>Y_0</math></i>
--------------	--

---

## Description

This function provides the plausibility of  $Y(s_0)$  being  $Y_0$ , 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,
  dfun = c("residual2", "abs_residual", "std_residual2", "std_abs_residual")
)
```

## Arguments

<b><math>Y_0</math></b>	a scalar or a vector
<b><math>s_0</math></b>	prediction location, a numeric vector with <code>length = 2</code> .
<b><math>s</math></b>	an $n \times 2$ matrix or data.frame with two coordinates of $n$ locations.
<b><math>Y</math></b>	a vector with $n$ values corresponding to $Y(s)$ .
<b>global</b>	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 $\eta < \text{Inf}$ or $m \leq n$ . Defaults to TRUE.
<b>eta</b>	kernel bandwidth for weight schema, a positive scalar with smaller value meaning more localized procedure. Defaults to Inf, which puts equal weight on surrounding $m$ points.
<b>m</b>	an positive integer representing the number of nearest locations to use for prediction. Default to NULL. If <code>global = TRUE</code> , $m = n$ ; if <code>global = FALSE</code> and $m$ is not specified, $m$ would be determined by <code>eta</code> .
<b>pred_fun</b>	spatial prediction function with inputs being $s_0, s, Y$ and outputs being predicted $Y(s_0)$ (and its standard error). Defaults to <a href="#">krige_pred</a> .
<b>dfun</b>	non-conformity measure with four options. In which, "residual2" (default) represents squared residual, "std_residual2" represents standardized squared residual, "abs_residual" represents absolute residual, and "std_abs_residual" represents standardized absolute residual.

**Value**

The output is a scalar or a vector with plausibility values for  $Y_0$ . 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  $Y(s_0)$ , where  $s_0 = c(0.5, 0.5)$ , using sample data
## What's the plausibility if  $Y(s_0) = 0$ ?  $Y(s_0) = 1.5$ ?

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

# plausibility for  $Y(s_0) = 0$ : 0.8744795
plausibility(Y0=0, s0=s0, s=s, Y=Y)

# plausibility for  $Y(s_0) = 1.5$ : 0.4669839
plausibility(Y0=1.5, s0=s0, s=s, Y=Y)

# plausibility for a sequence of  $Y_0$ '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  $Y(s_0)$ , 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,
  dfun = c("residual2", "abs_residual", "std_residual2", "std_abs_residual"),
  precision = NULL
)
```

## Arguments

<code>s0</code>	prediction location, a numeric vector with <code>length = 2</code> .
<code>s</code>	an $n \times 2$ matrix or data.frame with two coordinates of $n$ locations.
<code>Y</code>	a vector with $n$ values corresponding to $Y(s)$ .
<code>global</code>	logical; if TRUE, <code>scp</code> function returns the result of global spatial conformal prediction (GSCP); if FALSE, <code>scp</code> function returns the result of local spatial conformal prediction (LSCP) and users need to specify <code>eta &lt; Inf</code> or $m \leq n$ . Defaults to TRUE.
<code>eta</code>	kernel bandwidth for weight schema, a positive scalar with smaller value meaning more localized procedure. Defaults to Inf, which puts equal weight on surrounding $m$ points.
<code>m</code>	an positive integer representing the number of nearest locations to use for prediction. Default to NULL. If <code>global = TRUE</code> , $m = n$ ; if <code>global = FALSE</code> and $m$ is not specified, $m$ would be determined by <code>eta</code> .
<code>pred_fun</code>	spatial prediction function with inputs being <code>s0,s,Y</code> and outputs being predicted $Y(s0)$ (and its standard error). Defaults to <a href="#">krige_pred</a> .
<code>dfun</code>	non-conformity measure with four options. In which, "residual2" (default) represents squared residual, "std_residual2" represents standardized squared residual, "abs_residual" represents absolute residual, and "std_abs_residual" represents standardized absolute residual.
<code>precision</code>	a positive scalar represents how dense $Y(s)$ candidates ( <code>Y_cand</code> ) are. Defaults to NULL.

## Value

The output is a data.frame of `Y_cand` and corresponding plausibility values.

## Author(s)

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

## References

to be entered

**See Also**[plausibility](#)**Examples**

```
## generate plausibility contour for  $Y(s_0)$ , where  $s_0 = c(0.5, 0.5)$ , using sample data

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

p_df = plausibility_contour(s0=s0, s=s, Y=Y)
plot(p_df$Y_cand, p_df$p_y, type = "l", lwd = 2, las = 1, xlab = "Y candidates", ylab = "plausibility")

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

---

 scp

*Spatial conformal prediction at a single input location*


---

**Description**

This function provides the spatial conformal prediction interval for location  $s_0$ , given observations  $s$  and  $Y$ .

**Usage**

```
scp(
  s0,
  s,
  Y,
  global = TRUE,
  eta = Inf,
  m = NULL,
  pred_fun = krige_pred,
  dfun = c("residual2", "abs_residual", "std_residual2", "std_abs_residual"),
  precision = NULL,
  alpha = 0.05
)
```

**Arguments**

$s_0$	prediction location, a numeric vector with <code>length = 2</code> .
$s$	an $n \times 2$ matrix or data.frame with two coordinates of $n$ locations.
$Y$	a vector with $n$ values corresponding to $Y(s)$ .

<b>global</b>	logical; if TRUE , <code>scp</code> function returns the result of global spatial conformal prediction (GSCP); if FALSE, <code>scp</code> function returns the result of local spatial conformal prediction (LSCP) and users need to specify <code>eta</code> . Defaults to TRUE.
<b>eta</b>	kernel bandwidth for weight schema, a positive scalar with smaller value meaning more localized procedure. Defaults to Inf, which puts equal weight on surrounding $m$ points.
<b>m</b>	an positive integer representing the number of nearest locations to use for prediction. Default depends on <code>eta</code> .
<b>pred.fun</b>	spatial prediction function with inputs being $s_0, s, Y$ and outputs being predicted $Y(s_0)$ (and its standard error). Defaults to <a href="#">krige-pred</a> representing Kriging prediction.
<b>dfun</b>	non-conformity measure with four options. In which, "residual2" (default) represents squared residual, "std_residual2" represents standardized squared residual, "abs_residual" represents absolute residual, and "std_abs_residual" represents standardized absolute residual.
<b>precision</b>	a positive scalar represents how dense the candidates for $Y(s)$ are. Defaults to NULL.
<b>alpha</b>	significance level. Defaults to 0.05.

### Value

The output is a vector of lower and upper bounds of the conformal prediction interval.

### 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  $s_0 = c(0.5, 0.5)$  using sample data

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

# default prediction interval
scp(s0=s0, 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="abs_residual")

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



# Index

krige\_pred, [2](#), [3](#), [5](#), [7](#)

plausibility, [2](#), [3](#), [6](#), [7](#)

plausibility\_contour, [2](#), [4](#), [4](#), [7](#)

scp, [2](#), [6](#)