

Package ‘StempCens’

June 11, 2025

Type Package

Title Spatio-Temporal Estimation and Prediction for Censored/Missing Responses

Version 1.2.0

Description It estimates the parameters of spatio-temporal models with censored or missing data using the SAEM algorithm (Delyon et al., 1999). This algorithm is a stochastic approximation of the widely used EM algorithm and is particularly valuable for models in which the E-step lacks a closed-form expression. It also provides a function to compute the observed information matrix using the method developed by Louis (1982). To assess the performance of the fitted model, case-deletion diagnostics are provided.

License GPL (>= 2)

Imports Rcpp, stats, utils, mvtnorm, tmvtnorm, MCMCglmm, ggplot2, grid, Rdpack

RdMacros Rdpack

Encoding UTF-8

RoxygenNote 7.3.2

LinkingTo Rcpp, RcppArmadillo

Suggests testthat

NeedsCompilation yes

Author Larissa A. Matos [aut, cre] (ORCID: <https://orcid.org/0000-0002-2635-0901>),
Katherine L. Valeriano [aut] (ORCID: <https://orcid.org/0000-0001-6388-4753>),
Victor H. Lachos [ctb] (ORCID: <https://orcid.org/0000-0002-7239-2459>)

Maintainer Larissa A. Matos <larissa.amatos@gmail.com>

Repository CRAN

Date/Publication 2025-06-11 21:50:02 UTC

Contents

StempCens-package	2
-----------------------------	---

CovarianceM	3
CrossStempCens	4
DiagStempCens	6
EffectiveRange	7
EstStempCens	9
PredStempCens	12
rnStempCens	13
Index	15

StempCens-package	<i>Spatio-Temporal Estimation and Prediction for Censored/Missing Responses</i>
-------------------	---

Description

It estimates the parameters of spatio-temporal models with censored or missing data using the SAEM algorithm (Delyon et al., 1999). This algorithm is a stochastic approximation of the widely used EM algorithm and is particularly valuable for models in which the E-step lacks a closed-form expression. It also provides a function to compute the observed information matrix using the method developed by Louis (1982). To assess the performance of the fitted model, case-deletion diagnostics are provided.

Details

The functions provided are:

- [CovarianceM](#): computes the spatio-temporal covariance matrix for balanced data.
- [EffectiveRange](#): computes the effective range for an isotropic spatial correlation function.
- [EstStempCens](#): returns the maximum likelihood estimates of the unknown parameters.
- [PredStempCens](#): performs spatio-temporal prediction in a set of new S spatial locations for fixed time points.
- [CrossStempCens](#): performs cross-validation, which measure the performance of the predictive model on new test dataset.
- [DiagStempCens](#): returns measures and graphics for diagnostic analysis.

Author(s)

Larissa A. Matos ([ORCID](#)), Katherine L. Valeriano ([ORCID](#)) and Victor H. Lachos ([ORCID](#))

Maintainer: Larissa A. Matos (<larissa.amatos@gmail.com>).

References

- Cook R (1977). “Detection of influential observation in linear regression.” *Technometrics*, **19**(1), 15–18. doi:[10.1080/00401706.1977.10489493](https://doi.org/10.1080/00401706.1977.10489493).
- Delyon B, Lavielle M, Moulines E (1999). “Convergence of a stochastic approximation version of the EM algorithm.” *Annals of Statistics*, **27**(1), 94–128. doi:[10.1214/aos/1018031103](https://doi.org/10.1214/aos/1018031103).
- Louis T (1982). “Finding the observed information matrix when using the EM algorithm.” *Journal of the Royal Statistical Society: Series B (Methodological)*, **44**(2), 226–233. doi:[10.1111/j.2517-6161.1982.tb01203.x](https://doi.org/10.1111/j.2517-6161.1982.tb01203.x).
- Zhu H, Lee S, Wei B, Zhou J (2001). “Case-deletion measures for models with incomplete data.” *Biometrika*, **88**(3), 727–737. doi:[10.1093/biomet/88.3.727](https://doi.org/10.1093/biomet/88.3.727).

CovarianceM

Covariance matrix for spatio-temporal model

Description

It computes the spatio-temporal covariance matrix for balanced data, i.e., when we have the same temporal indexes per location. To compute the spatial correlation it provides 5 functions: exponential, gaussian, matern, spherical and power exponential. To compute the temporal correlation is used an autocorrelation function of an AR(1) process.

Usage

```
CovarianceM(phi, rho, tau2, sigma2, distSpa, disTemp, kappa, type.S)
```

Arguments

phi	value of the spatial scaling parameter.
rho	value of the time scaling parameter.
tau2	value of the the nugget effect parameter.
sigma2	value of the partial sill.
distSpa	$n \times n$ spatial distance matrix without considering repetitions.
disTemp	$T \times T$ temporal distance matrix without considering repetitions.
kappa	parameter for all spatial covariance functions. In the case of exponential, gaussian and spherical function κ is equal to zero. For the power exponential function κ is a number between 0 and 2. For the matern correlation function is upper than 0.
type.S	type of spatial correlation function: 'exponential' for exponential, 'gaussian' for gaussian, 'matern' for matern, 'pow.exp' for power exponential and 'spherical' for spherical function, respectively. See the analytical form of these functions in EffectiveRange .

Value

The function returns the $nT \times nT$ spatio-temporal covariance matrix for balanced data.

Author(s)

Katherine L. Valeriano, Victor H. Lachos and Larissa A. Matos

Examples

```
set.seed(1000)
# Parameter values
phi <- 5
rho <- 0.45
tau2 <- 0.80
sigma2 <- 2

# Coordinates and time points
coords <- matrix(runif(20, 0, 10), ncol=2) # Cartesian coordinates without repetitions
time <- as.matrix(1:5) # Time index without repetitions
Ms <- as.matrix(dist(coords)) # Spatial distances
Mt <- as.matrix(dist(time)) # Temporal distances

# Covariance matrix
Cov <- CovarianceM(phi, rho, tau2, sigma2, distSpa=Ms, disTemp=Mt,
                  kappa=0, type.S="exponential")
```

CrossStempCens	<i>Cross-Validation in spatio-temporal model with censored/missing responses</i>
----------------	--

Description

This function performs cross-validation in spatio-temporal model with censored/missing responses, which measure the performance of the predictive model on new test dataset. The cross-validation method for assessing the model performance is validation set approach (or data split).

Usage

```
CrossStempCens(Pred.StempCens, yObs.pre)
```

Arguments

Pred.StempCens	an object of class Pred.StempCens given as output by the PredStempCens function.
yObs.pre	a vector of the observed responses, the test data.

Value

Bias	bias prediction error.
Mspe	mean squared prediction error.
Rmspe	root mean squared prediction error.
Mae	mean absolute error.

Author(s)

Katherine L. Valeriano, Victor H. Lachos and Larissa A. Matos

See Also

[EstStempCens](#), [PredStempCens](#)

Examples

```

set.seed(400)
# Parameter values
beta <- c(-1,1.50)
phi <- 5
rho <- 0.6
tau2 <- 0.80
sigma2 <- 2

# Coordinates and covariates
coords <- matrix(round(runif(14, 0, 10), 9), ncol=2) # Coordinates without repetitions
time <- as.matrix(seq(1, 5)) # Time index without repetitions
x <- cbind(rexp(35, 2), rnorm(35, 2, 1))

# Data
data <- rnStempCens(x, time, coords, beta, phi, rho, tau2, sigma2,
                    type.S="gaussian", kappa=0, cens="left", pcens=0.2)

# Splitting the dataset
train <- data[1:32,]
test <- data[33:35,]

# Estimation
x <- cbind(train$x1, train$x2)
cc <- train$ci

est_teste <- EstStempCens(y=train$yObs, x, cc=train$ci, time=train$time, coord=train[, 1:2],
                          LI=train$lcl, LS=train$ucl, init.phi=3.5, init.rho=0.5,
                          init.tau2=1, type.Data="unbalanced", method="nlminb", kappa=0,
                          type.S="gaussian", IMatrix=TRUE, M=20, perc=0.25,
                          MaxIter=300, pc=0.20)

# Prediction
xPre <- cbind(test$x1, test$x2)
pre_teste <- PredStempCens(est_teste, test[,1:2], test$time, xPre)
class(pre_teste)

# Cross-validation
cross_teste <- CrossStempCens(pre_teste, test$yObs)
cross_teste$Mspe # MSPE

```

DiagStempCens	<i>Diagnostic in spatio-temporal model with censored/missing responses</i>
---------------	--

Description

Return measures and graphics for diagnostic analysis in spatio-temporal model with censored/missing responses.

Usage

```
DiagStempCens(Est.StempCens, type.diag = "individual", diag.plot = TRUE,
              ck)
```

Arguments

Est.StempCens	an object of class Est.StempCens given as output by the EstStempCens function. In the EstStempCensfunction, IMatrix must be TRUE.
type.diag	type of diagnostic: 'individual' is related when one observation is deleted, 'time' is related when an entire time is deleted, 'location' is related when an entire location is deleted and 'all' the three cases ('individual', 'time' and 'location'). By default type.diag is individual.
diag.plot	TRUE or FALSE. It indicates if diagnostic plots must be showed. By default = TRUE.
ck	the value for ck considered in the benchmark value for the index plot: $mean(GD) + ck * sd(GD)$, where GD is the vector with all values of the diagnostic measures.

Details

This function uses the case deletion approach to study the impact of deleting one or more observations from the dataset on the parameters estimates, using the ideas of Cook (1977) and Zhu et al. (2001). The measure is defined by

$$GD_i(\theta^*) = (\theta^* - \theta^*[i])' [-Q^*(\theta|\theta^*)] (\theta^* - \theta^*[i]), i = 1, \dots, m,$$

where θ^* is the estimate of θ using the complete data, $\theta^*[i]$ are the estimates obtained after deletion of the i -th observation (or group of observations) and $Q^*(\theta|\theta^*)$ is the Hessian matrix.

We can eliminate an observation, an entire location or an entire time index.

Value

The function returns a list with the diagnostic measures.

If type.diag == individual | time | location: GD is a data.frame with the index value of the observation and the GD measure.

If type.diag == all: GDind is a data.frame with the index value of the observation and the GD measure for individual.

GDtime is a data.frame with the time index value and the GD measure for time.

GDloc is a data.frame with the side index value and the GD measure for location.

Author(s)

Katherine L. Valeriano, Victor H. Lachos and Larissa A. Matos

See Also

[EstStempCens](#)

Examples

```
set.seed(12345)
# Parameter values
beta <- c(-1,1.5)
phi <- 3
rho <- 0.40
tau2 <- 1
sigma2 <- 2

# Simulating data
coord <- matrix(runif(10, 0, 10), ncol=2) # Cartesian coordinates without repetitions
time <- as.matrix(1:5) # Time index without repetitions
x <- cbind(rexp(25,2), rnorm(25,2,1))
data <- rnStempCens(x, time, coord, beta, phi, rho, tau2, sigma2,
                   type.S="exponential", cens="right", pcens=0.20)
data$yObs[17] <- abs(data$yObs[17]) + 2*sd(data$yObs) # perturbed observation

# Estimation
est <- EstStempCens(y=data$yObs, x, cc=data$ci, data$time, cbind(data$x.coord,data$y.coord),
                   LI=data$lcl, LS=data$ucl, init.phi=2.5, init.rho=0.5, init.tau2=0.8,
                   type.Data="balanced", method="nlminb", kappa=0, type.S="exponential",
                   IMatrix=TRUE, lower.lim=c(0.01,-0.99,0.01), upper.lim=c(30,0.99,20), M=20,
                   perc=0.25, MaxIter=300, pc=0.20)

# Diagnostic
set.seed(12345)
diag <- DiagStempCens(est, type.diag="time", diag.plot = TRUE, ck=1)
```

EffectiveRange

Effective range for some spatial correlation functions

Description

It computes the effective range for an isotropic spatial correlation function, which is commonly defined to be the distance from which the correlation becomes small, typically below 0.05.

Usage

```
EffectiveRange(cor = 0.05, phi, kappa = 0, Sp.model = "exponential")
```

Arguments

<code>cor</code>	effective correlation to check for. By default = 0.05.
<code>phi</code>	spatial scaling parameter.
<code>kappa</code>	smoothness parameter, required by the matern and power exponential functions. By default = 0.
<code>Sp.model</code>	type of spatial correlation function: 'exponential' for exponential, 'gaussian' for gaussian, 'matern' for matern, 'pow.exp' for power exponential and 'spherical' for spherical function, respectively. By default = exponential.

Details

The available isotropic spatial correlation functions are:

Exponential: $Corr(d) = \exp(-d/\phi)$,

Gaussian: $Corr(d) = \exp(-(d/\phi)^2)$,

Matern: $Corr(d) = 1/(2^{\kappa-1}\Gamma(\kappa))(d/\phi)^{\kappa}K_{\kappa}(d/\phi)$,

Power exponential: $Corr(d) = \exp(-(d/\phi)^{\kappa})$,

Spherical: $Corr(d) = 1 - 1.5d/\phi + 0.5(d/\phi)^3$,

where d is the Euclidean distance between two observations, ϕ is the spatial scaling parameter, $\Gamma(\cdot)$ is the gamma function, κ is the smoothness parameter and $K_{\kappa}(\cdot)$ is the modified Bessel function of the second kind of order κ .

Value

The function returns the effective range, i.e., the approximate distance from which the spatial correlation is lower than `cor`.

Author(s)

Katherine L. Valeriano, Victor H. Lachos and Larissa A. Matos

Examples

```
phi <- 164.60
range1 <- EffectiveRange(0.05, phi, kappa=0, Sp.model="exponential")
range2 <- EffectiveRange(0.05, phi, kappa=1, Sp.model="pow.exp")
# Note that these functions are equivalent.
```

EstStempCens	<i>ML estimation in spatio-temporal model with censored/missing responses</i>
--------------	---

Description

Return the maximum likelihood estimates of the unknown parameters of spatio-temporal model with censored/missing responses. The estimates are obtained using SAEM algorithm. The function also computes the observed information matrix using the method developed by Louis (1982). The types of censoring considered are left, right, interval or missing values.

Usage

```
EstStempCens(y, x, cc, time, coord, LI, LS, init.phi, init.rho, init.tau2,
  tau2.fixo = FALSE, type.Data = "balanced", method = "nlminb",
  kappa = 0, type.S = "exponential", IMatrix = TRUE,
  lower.lim = c(0.01, -0.99, 0.01), upper.lim = c(30, 0.99, 20), M = 20,
  perc = 0.25, MaxIter = 300, pc = 0.2, error = 1e-06)
```

Arguments

y	a vector of responses.
x	a matrix or vector of covariates.
cc	a vector of censoring indicators. For each observation: 1 if censored/missing and 0 if non-censored/non-missing.
time	a vector of time.
coord	a matrix of coordinates of the spatial locations.
LI	lower limit of detection. For each observation: if non-censored/non-missing =y, if left-censored/missing =-Inf or =LOD if right/interval-censored.
LS	upper limit of detection. For each observation: if non-censored/non-missing =y, if right-censored/missing =Inf or =LOD if left/interval-censored.
init.phi	initial value of the spatial scaling parameter.
init.rho	initial value of the time scaling parameter.
init.tau2	initial value of the the nugget effect parameter.
tau2.fixo	TRUE or FALSE. Indicate if the nugget effect (τ^2) parameter must be fixed. By default = FALSE.
type.Data	type of the data: 'balanced' for balanced data and 'unbalanced' for unbalanced data. By default = balanced.
method	optimization method used to estimate (ϕ , ρ and τ^2): 'optim' for the function optim and 'nlminb' for the function nlminb . By default = nlminb.
kappa	parameter for all spatial covariance functions. In the case of exponential, gaussian and spherical function κ is equal to zero. For the power exponential function κ is a number between 0 and 2. For the matern correlation function is upper than 0.

type.S	type of spatial correlation function: 'exponential' for exponential, 'gaussian' for gaussian, 'matern' for matern, 'pow.exp' for power exponential and 'spherical' for spherical function, respectively. Default is exponential function.
IMatrix	TRUE or FALSE. Indicate if the observed information matrix will be computed. By default = TRUE.
lower.lim, upper.lim	vectors of lower and upper bounds for the optimization method. If unspecified, the default is $c(0.01, -0.99, 0.01)$ for the lower bound and $c(30, 0.99, 20)$ for the upper bound if tau2.fxo=FALSE.
M	number of Monte Carlo samples for stochastic approximation. By default = 20.
perc	percentage of burn-in on the Monte Carlo sample. By default = 0.25.
MaxIter	the maximum number of iterations of the SAEM algorithm. By default = 300.
pc	percentage of iterations of the SAEM algorithm with no memory. By default = 0.20.
error	the convergence maximum error. By default = 1e-6.

Details

The spatio-temporal Gaussian model is giving by:

$$Y(s_i, t_j) = \mu(s_i, t_j) + Z(s_i, t_j) + \epsilon(s_i, t_j),$$

where the deterministic term $\mu(s_i, t_j)$ and the stochastic terms $Z(s_i, t_j)$, $\epsilon(s_i, t_j)$ can depend on the observed spatio-temporal indexes for $Y(s_i, t_j)$. We assume Z is normally distributed with zero-mean and covariance matrix $\Sigma_z = \sigma^2 \Omega_{\phi\rho}$, where σ^2 is the partial sill, $\Omega_{\phi\rho}$ is the spatio-temporal correlation matrix, ϕ and ρ are the spatial and time scaling parameters; $\epsilon(s_i, t_j)$ is an independent and identically distributed measurement error with $E[\epsilon(s_i, t_j)] = 0$, variance $Var[\epsilon(s_i, t_j)] = \tau^2$ (the nugget effect) and $Cov[\epsilon(s_i, t_j), \epsilon(s_k, t_l)] = 0$ for all $s_i \neq s_k$ or $t_j \neq t_l$.

In particular, we define $\mu(s_i, t_j)$, the mean of the stochastic process as

$$\mu(s_i, t_j) = \sum_{k=1}^p x_k(s_i, t_j) \beta_k,$$

where $x_1(s_i, t_j), \dots, x_p(s_i, t_j)$ are known functions of (s_i, t_j) , and $\beta_1, \dots, \beta_p$ are unknown parameters to be estimated. Equivalently, in matrix notation, we have the spatio-temporal linear model as follows:

$$Y = X\beta + Z + \epsilon,$$

$$Z \sim N(0, \sigma^2 \Omega_{\phi\rho}),$$

$$\epsilon \sim N(0, \tau^2 I_m).$$

Therefore the spatio-temporal process, Y , has normal distribution with mean $E[Y] = X\beta$ and variance $\Sigma = \sigma^2 \Omega_{\phi\rho} + \tau^2 I_m$. We assume that Σ is non-singular and X has full rank.

The estimation process was computed via SAEM algorithm initially proposed by Delyon et al. (1999).

Value

The function returns an object of class `Est.StempCens` which is a list given by:

`m.data` Returns a list with all data components given in input.

m.results A list given by:

theta	final estimation of $\theta = (\beta, \sigma^2, \tau^2, \phi, \rho)$.
Theta	estimated parameters in all iterations, $\theta = (\beta, \sigma^2, \tau^2, \phi, \rho)$.
beta	estimated β .
sigma2	estimated σ^2 .
tau2	estimated τ^2 .
phi	estimated ϕ .
rho	estimated ρ .
Eff.range	estimated effective range.
PsiInv	estimated Ψ^{-1} , where $\Psi = \Sigma/\sigma^2$.
Cov	estimated Σ .
SAEMy	stochastic approximation of the first moment for the truncated normal distribution.
SAEMyy	stochastic approximation of the second moment for the truncated normal distribution.
Hessian	Hessian matrix, the negative of the conditional expected second derivative matrix given the observed values.
Louis	the observed information matrix using the Louis' method.
loglik	log likelihood for SAEM method.
AIC	Akaike information criteria.
BIC	Bayesian information criteria.
AICcorr	corrected AIC by the number of parameters.
iteration	number of iterations needed to convergence.

Author(s)

Katherine L. Valeriano, Victor H. Lachos and Larissa A. Matos

Examples

```
set.seed(12345)
# Initial parameter values
beta <- c(-1,1.50)
phi <- 5
rho <- 0.45
tau2 <- 0.80
sigma2 <- 1.5
coord <- matrix(runif(10, 0, 10), ncol=2)
time <- matrix(1:5, ncol=1)
x <- cbind(rexp(25,2), rnorm(25,2,1))
# Data simulation
data <- rnStempCens(x, time, coord, beta, phi, rho, tau2, sigma2,
                    type.S="matern", kappa=1.5, cens="left", pcens=0.20)
# Estimation
```

```
est_teste <- EstStempCens(data$yObs, x, data$ci, data$time, cbind(data$x.coord, data$y.coord),
  data$lcl, data$ucl, init.phi=3.5, init.rho=0.5, init.tau2=0.7,
  tau2.fixo=FALSE, kappa=1.5, type.S="matern", IMatrix=TRUE, M=20,
  perc=0.25, MaxIter=300, pc=0.2)
```

PredStempCens	<i>Prediction in spatio-temporal model with censored/missing responses</i>
---------------	--

Description

This function performs spatio-temporal prediction in a set of new S spatial locations for fixed time points.

Usage

```
PredStempCens(Est.StempCens, locPre, timePre, xPre)
```

Arguments

Est.StempCens	an object of class Est.StempCens given as output by the EstStempCens function.
locPre	a matrix of coordinates for which prediction is performed.
timePre	the time point vector for which prediction is performed.
xPre	a matrix of covariates for which prediction is performed.

Value

The function returns an object of class Pred.StempCens which is a list given by:

predValues	predicted values.
VarPred	predicted covariance matrix.

Author(s)

Katherine L. Valeriano, Victor H. Lachos and Larissa A. Matos

See Also

[EstStempCens](#)

Examples

```

set.seed(12345)
# Parameter values
beta <- c(-1,1.50)
phi <- 5
rho <- 0.60
tau2 <- 0.80
sigma2 <- 2
coord <- matrix(runif(34, 0, 10), ncol=2)
time <- matrix(1:5, ncol=1)
x <- cbind(rexp(85,2), rnorm(85,2,1))

# Data simulation
data <- rnStempCens(x, time, coord, beta, phi, rho, tau2, sigma2,
                    type.S="pow.exp", kappa=0.5, cens="left", pcens=0.10)
# Splitting the dataset
train <- data[11:85,]
test <- data[1:10,]

# Estimation
x <- cbind(train$x1, train$x2)
coord2 <- cbind(train$x.coord, train$y.coord)
est_teste <- EstStempCens(train$yObs, x, train$ci, train$time, coord2, train$lcl,
                          train$ucl, init.phi=3.5, init.rho=0.5, init.tau2=1, kappa=0.5,
                          type.S="pow.exp", IMatrix=FALSE, M=20, perc=0.25, MaxIter=300,
                          pc=0.20)

# Prediction
xPre <- cbind(test$x1, test$x2)
pre_test <- PredStempCens(est_teste, test[,1:2], test$time, xPre)

```

rnStempCens

*Censored spatio-temporal data simulation***Description**

It simulates balanced censored spatio-temporal data with a linear structure for an established censoring rate or limit of detection.

Usage

```

rnStempCens(x, time, coords, beta, phi, rho, tau2, sigma2,
            type.S = "exponential", kappa = 0, cens = "left", pcens = 0.1,
            lod = NULL)

```

Arguments

x	design matrix of dimensions $ntxp$.
time	vector containing the unique time points at which the observations are made, of length t .

coords	2D unique spatial coordinates of dimension $n \times 2$.
beta	linear regression parameters.
phi	value of the spatial scaling parameter.
rho	value of the time scaling parameter.
tau2	value of the the nugget effect parameter.
sigma2	value of the partial sill.
type.S	type of spatial correlation function: 'exponential' for exponential, 'gaussian' for gaussian, 'matern' for matern, 'pow.exp' for power exponential and 'spherical' for spherical function, respectively. See the analytical expressions of these functions in EffectiveRange .
kappa	parameter for all spatial covariance functions. In the case of exponential, gaussian and spherical function κ is equal to zero. For the power exponential function κ is a number between 0 and 2. For the matern correlation function is upper than 0.
cens	'left' or 'right' censoring. By default='left'.
pcens	desired censoring rate. By default=0.10.
lod	desired detection limit for censored observations. By default=NULL.

Value

The function returns a data.frame containing the simulated data.

Author(s)

Katherine L. Valeriano, Victor H. Lachos and Larissa A. Matos

Examples

```
set.seed(1000)
# Initial parameter values
phi <- 5
rho <- 0.45
tau2 <- 0.80
sigma2 <- 2
beta <- c(1, 2.5)
x <- cbind(1, rnorm(50))

# Coordinates
coords <- matrix(runif(20, 0, 10), ncol=2) # Cartesian coordinates without repetitions
time <- 1:5 # Time index without repetitions

# Data simulation
data <- rnStempCens(x, time, coords, beta, phi, rho, tau2, sigma2,
                  type.S="exponential", cens="left", pcens=0.10)
```

Index

CovarianceM, [2](#), [3](#)
CrossStempCens, [2](#), [4](#)

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

EffectiveRange, [2](#), [3](#), [7](#), [14](#)
EstStempCens, [2](#), [5–7](#), [9](#), [12](#)

nlminb, [9](#)

optim, [9](#)

PredStempCens, [2](#), [4](#), [5](#), [12](#)

rnStempCens, [13](#)

StempCens-package, [2](#)