Package ‘BRISC’

August 19, 2019

Type Package

Title Fast Inference for Large Spatial Datasets using BRISC

Version 0.2.0

Maintainer Arkajyoti Saha <arkajyotisaha93@gmail.com>

Author Arkajyoti Saha [aut, cre],
Abhirup Datta [aut],
Jorge Nocedal [ctb],
Naoaki Okazaki [ctb]

Depends R (>= 3.3.0), RANN, parallel, stats, rdist, matrixStats,
pbapply, graphics

Description Fits Bootstrap with univariate spatial regression models using Bootstrap for Rapid Inference on Spatial Covariances (BRISC) for large datasets using Nearest Neighbor Gaussian Processes detailed in Saha and Datta (2018) <doi:10.1002/sta4.184>.

License GPL (>= 2)

URL https://github.com/ArkajyotiSaha/BRISC

BugReports https://github.com/ArkajyotiSaha/BRISC/issues

Encoding UTF-8

NeedsCompilation yes

RoxygenNote 6.1.1

Repository CRAN

Date/Publication 2019-08-19 16:00:02 UTC

R topics documented:

BRISC_bootstrap *Function for performing bootstrap with BRISC*

Description

The function BRISC_bootstrap performs bootstrap to provide confidence intervals for parameters of univariate spatial regression models using outputs of BRISC_estimation. The details of the bootstrap method can be found in BRISC (Saha & Datta, 2018). The optimization is performed with C library of limited-memory BFGS libLBFGS: a library of Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS), http://www.chokkan.org/software/liblbfgs/ (Naoaki Okazaki). For user convenience the soure codes of the package libLBFGS are provided in the package. Some code blocks are borrowed from the R package: spNNGP: Spatial Regression Models for Large Datasets using Nearest Neighbor Gaussian Processes https://CRAN.R-project.org/package=spNNGP .

Usage

```
BRISC_bootstrap(BRISC_Out, n_boot = 100, h = 1, n_omp = 1,
                init = "Initial", verbose = TRUE, nugget_status = 1)
```

Arguments

BRISC_Out an object of class "BRISC_Out", obtained as an output of BRISC_estimation.

n_boot number of bootstrap samples. Default value is 100.

h Number of core to be used in parallel computing setup for bootstrap samples. If h = 1, there is no parallelization. Default value is 1.

n_omp number of threads to be used, value can be more than 1 if source code is compiled with OpenMP support. Default is 1.

init Keyword that specifies initialization scheme to be used. Supported keywords are: "Initial" and "Estimate" for initialization of parameter values for bootstrap samples with initial values used in BRISC_estimate and estimated values of parameters in BRISC_estimate respectively.

verbose if TRUE, model specifications along with information regarding OpenMP support and progress of the algorithm is printed to the screen. Otherwise, nothing is printed to the screen. Default value is FALSE.

nugget_status If nugget_status = 0, tau square is fixed to 0, if nugget_status = 1 tau square is variable and is estimated. Default value is 1.

Value

A list comprising of the following:

boot.Theta estimates of spatial covariance parameters corresponding to bootstrap samples.

boot.Beta estimates of beta corresponding to bootstrap samples.

confidence.interval
confidence intervals corresponding to the parameters.

boot.time time (in seconds) required to perform the bootstrapping after preprocessing data in R, reported using proc.time().

Author(s)

Arkajyoti Saha <arkajyotisaha93@gmail.com>,
Abhirup Datta <abhidatta@jhu.edu>

References

Saha, A., & Datta, A. (2018). BRISC: bootstrap for rapid inference on spatial covariances. Stat, e184, DOI: 10.1002/sta4.184.

Okazaki N. libLBFGS: a library of Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS), http://www.chokkan.org/software/liblbfgs/ .

Andrew Finley, Abhirup Datta and Sudipto Banerjee (2017). spNNGP: Spatial Regression Models for Large Datasets using Nearest Neighbor Gaussian Processes. R package version 0.1.1. https://CRAN.R-project.org/package=spNNGP

Examples

```
rmvn <- function(n, mu = 0, V = matrix(1)){
  p <- length(mu)
  if(any(is.na(match(dim(V),p))))
    stop("Dimension not right!")
  D <- chol(V)
  t(matrix(rnorm(n*p), ncol=p)%*%D + rep(mu,rep(n,p)))
}

set.seed(1)
n <- 300
coords <- cbind(runif(n,0,1), runif(n,0,1))

beta <- c(1,5)
x <- cbind(rnorm(n), rnorm(n))

sigma.sq = 1
phi = 5
tau.sq = 0.1

B <- as.matrix(beta)
D <- as.matrix(dist(coords))
R <- exp(-phi*D)
w <- rmvn(1, rep(0,n), sigma.sq*R)

y <- rnorm(n, x%*%B + w, sqrt(tau.sq))

estimation_result <- BRISC_estimation(coords, x, y)
bootstrap_result <- BRISC_bootstrap(estimation_result, n_boot = 10)
```

BRISC_estimation *Function for estimation with BRISC*

Description

The function BRISC_estimation fits univariate spatial regression models for large spatial data using Vecchia's approximate likelihood (Vecchia, 1988). BRISC_estimation uses the sparse Cholesky representation of Vecchia's likelihood developed in Datta et al., 2016. The Maximum Likelihood Estimates (MLE) of the parameters are used later for calculating the confidence interval via the BRISC_bootstrap (BRISC, Saha & Datta, 2018). We recommend using BRISC_estimation followed by BRISC_bootstrap to obtain the cofidence intervals for the model parameters.

The optimization is performed with C library of limited-memory BFGS libLBFGS: a library of Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS), http://www.chokkan.org/software/liblbfgs/ (Naoaki Okazaki). For user convenience the soure codes of the package libLBFGS are provided in the package. The code for the coordinate ordering method, approximate Maximum Minimum Distance (Guinness, 2018) is available in https://github.com/joeguinness/gp_reorder/tree/master/R and is adopted with minor modification. Some code blocks are borrowed from the R package: spNNGP: Spatial Regression Models for Large Datasets using Nearest Neighbor Gaussian Processes https://CRAN.R-project.org/package=spNNGP .

Usage

```
BRISC_estimation(coords, x, y, sigma.sq = 1, tau.sq = 0.1, phi = 1,
                 nu = 0.5, n.neighbors = 15, n_omp = 1, order = "Sum_coords",
                 cov.model = "exponential", search.type = "tree",
                 verbose = TRUE, eps = 2e-05, nugget_status = 1)
```

Arguments

coords — an $n \times 2$ matrix of the observation coordinates in R^2 (e.g., easting and northing).

x — an $n \times p$ matrix of the covariates in the observation coordinates.

y — a n length vector of response at the observed coordinates.

sigma.sq — starting value of sigma square. Default value is 1.

tau.sq — starting value of tau square. Default value is 0.1.

phi — starting value of phi. Default value is 1.

nu — starting value of nu, only required for matern covariance model. Default value is 0.5.

n.neighbors — number of neighbors used in the NNGP. Default value is 15.

n_omp — number of threads to be used, value can be more than 1 if source code is compiled with OpenMP support. Default is 1.

order — Keyword that specifies the ordering scheme to be used in ordering the observations. Supported keywords are: "AMMD" and "Sum_coords" for approximate Maximum Minimum Distance and sum of coordinate based ordering, respectively. Default value is "Sum_coords". $n > 65$ is required for "AMMD".

`cov.model` Keyword that specifies the covariance function to be used in modelling the spatial dependence structure among the observations. Supported keywords are: "exponential", "matern", "spherical", and "gaussian" for exponential, matern, spherical and gaussian covariance function respectively. Default value is "exponential".

`search.type` Keyword that specifies type of nearest neighbor search algorithm to be used. Supported keywords are: "tree" and "brute". Both of them provide the same result, though "tree" should be faster. Default value is "tree".

`verbose` if TRUE, model specifications along with information regarding OpenMP support and progress of the algorithm is printed to the screen. Otherwise, nothing is printed to the screen. Default value is FALSE.

`eps` The tolerance to be used in centred finite difference approximation of derivatives. Default value is 2e-05.

`nugget_status` If nugget_status = 0, tau square is fixed to 0, if nugget_status = 1 tau square is variable and is estimated. Default value is 1.

Value

An object of class `BRISC_Out`, which is a list comprising:

`ord` the vector of indices used to order data necessary for fitting the NNGP model.

`coords` the matrix `coords[ord,]`.

`y` the vector `y[ord]`.

`X` the matrix `x[ord,,drop=FALSE]`.

`n.neighbors` the used value of `n.neighbors`.

`cov.model` the used covariance model.

`eps` value of used `eps`.

`init` Initial values of the parameters of the covariance model.

`Beta` estimate of beta.

`Theta` estimate of parameters of covarinace model.

`estimation.time`
time (in seconds) required to perform the model fitting after ordering and preprocessing data in R, reported using `proc.time()`.

`BRISC_Object` Object required for bootstrap and prediction.

Author(s)

Arkajyoti Saha <`arkajyotisaha93@gmail.com`>,
Abhirup Datta <`abhidatta@jhu.edu`>

References

Saha, A., & Datta, A. (2018). BRISC: bootstrap for rapid inference on spatial covariances. Stat, e184, DOI: 10.1002/sta4.184.

Datta, A., S. Banerjee, A.O. Finley, and A.E. Gelfand. (2016) Hierarchical Nearest-Neighbor Gaussian process models for large geostatistical datasets. Journal of the American Statistical Association, 111:800-812.

Guinness, J. (2018) Permutation and Grouping Methods for Sharpening Gaussian Process Approximations, Technometrics, DOI: 10.1080/00401706.2018.1437476, https://github.com/joeguinness/gp_reorder/tree/master/R .

Vecchia, A. V. (1988) Estimation and model identification for continuous spatial processes. Journal of the Royal Statistical Society. Series B (Methodological), 297-312.

Okazaki N. libLBFGS: a library of Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS), http://www.chokkan.org/software/liblbfgs/ .

Andrew Finley, Abhirup Datta and Sudipto Banerjee (2017). spNNGP: Spatial Regression Models for Large Datasets using Nearest Neighbor Gaussian Processes. R package version 0.1.1. https://CRAN.R-project.org/package=spNNGP

Examples

```
rmvn <- function(n, mu = 0, V = matrix(1)){
  p <- length(mu)
  if(any(is.na(match(dim(V),p))))
    stop("Dimension not right!")
  D <- chol(V)
  t(matrix(rnorm(n*p), ncol=p)%*%D + rep(mu,rep(n,p)))
}

set.seed(1)
n <- 1000
coords <- cbind(runif(n,0,1), runif(n,0,1))

beta <- c(1,5)
x <- cbind(rnorm(n), rnorm(n))

sigma.sq = 1
phi = 5
tau.sq = 0.1

B <- as.matrix(beta)
D <- as.matrix(dist(coords))
R <- exp(-phi*D)
w <- rmvn(1, rep(0,n), sigma.sq*R)

y <- rnorm(n, x%*%B + w, sqrt(tau.sq))

estimation_result <- BRISC_estimation(coords, x, y)
estimation_result$Theta ## Gives estimation of covariance model parameters.
estimation_result$Beta ## Gives estimation of Beta
```

BRISC_prediction *Function for performing prediction with BRISC*

Description

The function `BRISC_prediction` performs fast prediction on a set of new locations with univariate spatial regression models using Nearest Neighbor Gaussian Processes (NNGP) (Datta et al., 2016). BRISC_prediction uses the parameter estimates from `BRISC_estimation` for the prediction. Some code blocks are borrowed from the R package: spNNGP: Spatial Regression Models for Large Datasets using Nearest Neighbor Gaussian Processes https://CRAN.R-project.org/package=spNNGP .

Usage

```
BRISC_prediction(BRISC_Out, X.0, coords.0, n_omp = 1, verbose = TRUE)
```

Arguments

`BRISC_Out` an object of class "BRISC_Out", obtained as an output of `BRISC_estimation`.

`X.0` the covariates for prediction locations. Its Structure should be identical (including intercept) with that of covariates provided for estimation purpose in `BRISC_estimation`.

`coords.0` the spatial coordinates corresponding to prediction locations. Its structure should be same as that of coords in `BRISC_estimation`.

`n_omp` number of threads to be used, value can be more than 1 if source code is compiled with OpenMP support. Default is 1.

`verbose` if TRUE, model specifications along with information regarding OpenMP support and progress of the algorithm is printed to the screen. Otherwise, nothing is printed to the screen. Default value is FALSE.

Value

A list comprising of the following:

`prediction` predicted response corresponding to X.0 and coords.0.

`prediction.ci` confidence intervals corresponding to the predictions.

`prediction.time` time (in seconds) required to perform the prediction after preprocessing data in R, reported using `proc.time()`.

Author(s)

Arkajyoti Saha <arkajyotisaha93@gmail.com>,
Abhirup Datta <abhidatta@jhu.edu>

References

Datta, A., S. Banerjee, A.O. Finley, and A.E. Gelfand. (2016) Hierarchical Nearest-Neighbor Gaussian process models for large geostatistical datasets. Journal of the American Statistical Association, 111:800-812.

Andrew Finley, Abhirup Datta and Sudipto Banerjee (2017). spNNGP: Spatial Regression Models for Large Datasets using Nearest Neighbor Gaussian Processes. R package version 0.1.1. https://CRAN.R-project.org/package=spNNGP

Examples

```
rmvn <- function(n, mu = 0, V = matrix(1)){
  p <- length(mu)
  if(any(is.na(match(dim(V),p))))
    stop("Dimension not right!")
  D <- chol(V)
  t(matrix(rnorm(n*p), ncol=p)%*%D + rep(mu,rep(n,p)))
}

set.seed(1)
n <- 500
coords <- cbind(runif(n,0,1), runif(n,0,1))

beta <- c(1,5)
x <- cbind(rnorm(n), rnorm(n))

sigma.sq = 1
phi = 5
tau.sq = 0.1

B <- as.matrix(beta)
D <- as.matrix(dist(coords))
R <- exp(-phi*D)
w <- rmvn(1, rep(0,n), sigma.sq*R)

y <- rnorm(n, x%*%B + w, sqrt(tau.sq))

estimation_result <- BRISC_estimation(coords[1:400,], x[1:400,], y[1:400])
prediction_result <- BRISC_prediction(estimation_result, x[401:500,], coords[401:500,])
```

BRISC_variogram.ci *Function for plotting estimated Variogram and confidence region*

Description

The function BRISC_variogram.ci plots estimated Variogram and associated confience region. BRISC_variogram.ci uses the parameter estimates from BRISC_estimation and associated confidence interval from BRISC_bootstrap.

Usage

```
BRISC_variogram.ci(BRISC_Out, confidence_est, plot.variogram = FALSE)
```

Arguments

`BRISC_Out`	an object of class "BRISC_Out", obtained as an output of `BRISC_estimation`.
`confidence_est`	Bootstrp sample of the Theta parameters, obtained from `BRISC_bootstrap`.
`plot.variogram`	if TRUE, plots the variogram and the associated confidence region. Default is FALSE

Value

A list comprising of the following:

`variogram`	Variogram and associated confidence region corresponding to lag ranging from 0 to 20, evaluated at 0.01 frequency.
`Plot`	Plots the Variogram and associated confidence region with legends.

Author(s)

Arkajyoti Saha <`arkajyotisaha93@gmail.com`>,
Abhirup Datta <`abhidatta@jhu.edu`>

Examples

```
rmvn <- function(n, mu = 0, V = matrix(1)){
  p <- length(mu)
  if(any(is.na(match(dim(V),p))))
    stop("Dimension not right!")
  D <- chol(V)
  t(matrix(rnorm(n*p), ncol=p)%*%D + rep(mu,rep(n,p)))
}

set.seed(1)
n <- 300
coords <- cbind(runif(n,0,1), runif(n,0,1))

beta <- c(1,5)
x <- cbind(rnorm(n), rnorm(n))

sigma.sq = 1
phi = 5
tau.sq = 0.1

B <- as.matrix(beta)
D <- as.matrix(dist(coords))
R <- exp(-phi*D)
w <- rmvn(1, rep(0,n), sigma.sq*R)
```

```
y <- rnorm(n, x%*%B + w, sqrt(tau.sq))

estimation_result <- BRISC_estimation(coords, x, y)
bootstrap_result <- BRISC_bootstrap(estimation_result, n_boot = 10)
varg <- BRISC_variogram.ci(estimation_result, bootstrap_result$boot.Theta, plot.variogram = TRUE)
```

Index