Package: baygel (via r-universe)

October 23, 2026

Type Package

Title Bayesian Shrinkage Estimators for Precision Matrices in Gaussian Graphical Models

Version 0.3.0

Date 2023-11-11

Description This R package offers block Gibbs samplers for the Bayesian (adaptive) graphical lasso, ridge, and naive elastic net priors. These samplers facilitate the simulation of the posterior distribution of precision matrices for Gaussian distributed data and were originally proposed by: Wang (2012) <doi:10.1214/12-BA729>; Smith et al. (2022) <doi:10.48550/arXiv.2210.16290> and Smith et al. (2023) <doi:10.48550/arXiv.2306.14199>, respectively.

License GPL (>= 3)

Imports Rcpp (>= 1.0.8), RcppArmadillo (>= 0.11.1.1.0)

LinkingTo Rcpp, RcppArmadillo, RcppProgress

Suggests MASS, pracma

RoxygenNote 7.2.3

URL https://github.com/Jarod-Smithy/baygel

Encoding UTF-8

NeedsCompilation yes

Author Jarod Smith [aut, cre] (<https://orcid.org/0000-0003-4235-6147>), Mohammad Arashi [aut] (<https://orcid.org/0000-0002-5881-9241>), Andriette Bekker [aut] (<https://orcid.org/0000-0003-4793-5674>)

Maintainer Jarod Smith <jarodsmith706@gmail.com>

Repository https://cran.r-universe.dev

Date/Publication 2023-11-11 11:30:30 UTC

RemoteUrl https://github.com/cran/baygel

RemoteRef HEAD

RemoteSha 50a1991e2d166dcfe90da67597cf0ed532f436a0

Contents

blockBAGENI 2
blockBAGENII 3
blockBAGL 4
blockBAGR 5
blockBGEN 6
blockBGL 7
blockBGR 8

Index **10**

`blockBAGENI` *Type I naive Bayesian adaptive graphical elastic net block Gibbs sampler for Gaussian graphical models.*

Description

Implements the Type I naive Bayesian adaptive graphical elastic net block Gibbs sampler to simulate the posterior distribution of the precision matrix for Gaussian graphical models.

Usage

```
blockBAGENI(
  X,
  burnin,
  iterations,
  verbose = TRUE,
  r = 0.001,
  s = 0.01,
  a = 0.001,
  b = 0.1
)
```

Arguments

`X`	A numeric matrix, assumed to be generated from a multivariate Gaussian distribution.
`burnin`	An integer specifying the number of burn-in iterations.
`iterations`	An integer specifying the length of the Markov chain after the burn-in iterations.
`verbose`	A logical determining whether the progress of the MCMC sampler should be displayed.
`r`	A double specifying the value of the shape parameter for the gamma prior associated with the Bayesian graphical lasso penalty term.
`s`	A double specifying the value of the scale parameter for the gamma prior associated with the Bayesian graphical lasso penalty term.

a	A double specifying the value of the shape parameter for the inverse gamma prior associated with the Bayesian graphical ridge penalty term.
b	A double specifying the value of the scale parameter for the inverse gamma prior associated with the Bayesian graphical ridge penalty term.

Value

A list containing precision 'Omega' and covariance 'Sigma' matrices from the Markov chains.

Examples

```
# Generate true precision matrix:
p             <- 10
n             <- 500
OmegaTrue    <- pracma::Toeplitz(c(0.7^rep(1:p-1)))
SigTrue      <- pracma::inv(OmegaTrue)
# Generate expected value vector:
mu            <- rep(0,p)
# Generate multivariate normal distribution:
set.seed(123)
X             <- MASS::mvrnorm(n, mu = mu, Sigma = SigTrue)
# Generate posterior distribution:
posterior     <- blockBAGENI(X, iterations = 1000, burnin = 500)
# Estimated precision matrix using the mean of the posterior:
OmegaEst      <- apply(simplify2array(posterior$Omega), 1:2, mean)
```

---

blockBAGENII *Type II naive Bayesian adaptive graphical elastic net block Gibbs sampler for Gaussian graphical models.*

Description

Implements the Type II naive Bayesian adaptive graphical elastic net block Gibbs sampler to simulate the posterior distribution of the precision matrix for Gaussian graphical models.

Usage

```
blockBAGENII(X, burnin, iterations, verbose = TRUE, s = 0.1, b = 0.001)
```

Arguments

X	A numeric matrix, assumed to be generated from a multivariate Gaussian distribution.
burnin	An integer specifying the number of burn-in iterations.
iterations	An integer specifying the length of the Markov chain after the burn-in iterations.
verbose	A logical determining whether the progress of the MCMC sampler should be displayed.

s	A double specifying the value of the rate parameter for the exponential prior associated with the Bayesian graphical lasso penalty term.
b	A double specifying the value of the rate parameter for the exponential prior associated with the Bayesian graphical ridge penalty term.

Value

A list containing precision 'Omega' and covariance 'Sigma' matrices from the Markov chains.

Examples

```
# Generate true precision matrix:
p             <- 10
n             <- 500
OmegaTrue    <- pracma::Toeplitz(c(0.7^rep(1:p-1)))
SigTrue      <- pracma::inv(OmegaTrue)
# Generate expected value vector:
mu            <- rep(0,p)
# Generate multivariate normal distribution:
set.seed(123)
X             <- MASS::mvrnorm(n, mu = mu, Sigma = SigTrue)
# Generate posterior distribution:
posterior     <- blockBAGENII(X, iterations = 1000, burnin = 500)
# Estimated precision matrix using the mean of the posterior:
OmegaEst      <- apply(simplify2array(posterior$Omega), 1:2, mean)
```

---

blockBAGL *Bayesian adaptive graphical lasso block Gibbs sampler for Gaussian graphical models.*

Description

Implements a Bayesian adaptive graphical lasso block Gibbs sampler to simulate the posterior distribution of the precision matrix for Gaussian graphical models.

Usage

```
blockBAGL(X, burnin, iterations, verbose = TRUE, r = 0.01, s = 1e-06)
```

Arguments

X	A numeric matrix, assumed to be generated from a multivariate Gaussian distribution.
burnin	An integer specifying the number of burn-in iterations.
iterations	An integer specifying the length of the Markov chain after the burn-in iterations.
verbose	A logical determining whether the progress of the MCMC sampler should be displayed.
r	A double specifying the value of the shape parameter for the gamma prior.
s	A double specifying the value of the scale parameter for the gamma prior.

Value

A list containing precision 'Omega' and covariance 'Sigma' matrices from the Markov chains.

Examples

```
# Generate true precision matrix:
p             <- 10
n             <- 500
OmegaTrue    <- pracma::Toeplitz(c(0.7^rep(1:p-1)))
SigTrue      <- pracma::inv(OmegaTrue)
# Generate expected value vector:
mu            <- rep(0,p)
# Generate multivariate normal distribution:
set.seed(123)
X             <- MASS::mvrnorm(n, mu = mu, Sigma = SigTrue)
# Generate posterior distribution:
posterior     <- blockBAGL(X, iterations = 1000, burnin = 500)
# Estimated precision matrix using the mean of the posterior:
OmegaEst      <- apply(simplify2array(posterior$Omega), 1:2, mean)
```

`blockBAGR` *Bayesian adaptive graphical ridge block Gibbs sampler for Gaussian graphical models.*

Description

Implements a Bayesian adaptive graphical ridge block Gibbs sampler to simulate the posterior distribution of the precision matrix for Gaussian graphical models.

Usage

```
blockBAGR(X, burnin, iterations, verbose = TRUE, a = 1, b = 0.01)
```

Arguments

`X`	A numeric matrix, assumed to be generated from a multivariate Gaussian distribution.
`burnin`	An integer specifying the number of burn-in iterations.
`iterations`	An integer specifying the length of the Markov chain after the burn-in iterations.
`verbose`	A logical determining whether the progress of the MCMC sampler should be displayed.
`a`	A double specifying the value of the shape parameter for the inverse gamma prior.
`b`	A double specifying the value of the scale parameter for the inverse gamma prior.

Value

A list containing precision 'Omega' and covariance 'Sigma' matrices from the Markov chains.

Examples

```
# Generate true precision matrix:
p             <- 10
n             <- 500
OmegaTrue    <- pracma::Toeplitz(c(0.7^rep(1:p-1)))
SigTrue      <- pracma::inv(OmegaTrue)
# Generate expected value vector:
mu            <- rep(0,p)
# Generate multivariate normal distribution:
set.seed(123)
X             <- MASS::mvrnorm(n, mu = mu, Sigma = SigTrue)
# Generate posterior distribution:
posterior     <- blockBAGR(X, iterations = 1000, burnin = 500)
# Estimated precision matrix using the mean of the posterior:
OmegaEst      <- apply(simplify2array(posterior$Omega), 1:2, mean)
```

`blockBGEN` *Naive Bayesian graphical elastic net block Gibbs sampler for Gaussian graphical models.*

Description

Implements the Bayesian graphical elastic net block Gibbs sampler to simulate the posterior distribution of the precision matrix for Gaussian graphical models.

Usage

```
blockBGEN(X, burnin, iterations, lambda = 1, sig = 1, verbose = TRUE)
```

Arguments

`X`	A numeric matrix, assumed to be generated from a multivariate Gaussian distribution.
`burnin`	An integer specifying the number of burn-in iterations.
`iterations`	An integer specifying the length of the Markov chain after the burn-in iterations.
`lambda`	A numeric value representing the rate parameter for the double exponential and exponential prior associated with the Bayesian graphical lasso penalty term.
`sig`	A numeric value representing the standard deviation parameter for the double Gaussian and truncated Gaussian prior associated with the Bayesian graphical ridge penalty term.
`verbose`	A logical determining whether the progress of the MCMC sampler should be displayed.

Value

A list containing precision ‘Omega‘ and covariance ‘Sigma‘ matrices from the Markov chains.

Examples

```
# Generate true precision matrix:
p             <- 10
n             <- 500
OmegaTrue    <- pracma::Toeplitz(c(0.7^rep(1:p-1)))
SigTrue      <- pracma::inv(OmegaTrue)
# Generate expected value vector:
mu            <- rep(0,p)
# Generate multivariate normal distribution:
set.seed(123)
X             <- MASS::mvrnorm(n, mu = mu, Sigma = SigTrue)
# Generate posterior distribution:
posterior     <- blockBGEN(X, iterations = 1000, burnin = 500, lambda = 1, sig = 1)
# Estimated precision matrix using the mean of the posterior:
OmegaEst      <- apply(simplify2array(posterior$Omega), 1:2, mean)
```

---

blockBGL *Bayesian graphical lasso block Gibbs sampler for Gaussian graphical models.*

Description

Implements a Bayesian graphical lasso block Gibbs sampler to simulate the posterior distribution of the precision matrix for Gaussian graphical models.

Usage

```
blockBGL(X, burnin, iterations, lambda = 1, verbose = TRUE)
```

Arguments

X	A numeric matrix, assumed to be generated from a multivariate Gaussian distribution.
burnin	An integer representing the number of burn-in iterations.
iterations	An integer representing the length of the Markov chain post burn-in.
lambda	A numeric value representing the rate parameter for the double exponential and exponential prior.
verbose	A logical indicating if the MCMC sampler progress should be printed.

Value

A list containing precision ‘Omega‘ and covariance ‘Sigma‘ matrices from the Markov chains.

Examples

```
# Generate true precision matrix:
p             <- 10
n             <- 500
OmegaTrue    <- pracma::Toeplitz(c(0.7^rep(1:p-1)))
SigTrue      <- pracma::inv(OmegaTrue)
# Generate expected value vector:
mu            <- rep(0,p)
# Generate multivariate normal distribution:
set.seed(123)
X             <- MASS::mvrnorm(n, mu = mu, Sigma = SigTrue)
# Generate posterior distribution:
posterior     <- blockBGL(X, iterations = 1000, burnin = 500, lambda = 0.5)
# Estimated precision matrix using the mean of the posterior:
OmegaEst      <- apply(simplify2array(posterior$Omega), 1:2, mean)
```

---

blockBGR *Bayesian graphical ridge block Gibbs sampler for Gaussian graphical models.*

Description

Implements a Bayesian graphical ridge block Gibbs sampler to simulate the posterior distribution of the precision matrix for Gaussian graphical models.

Usage

```
blockBGR(X, burnin, iterations, sig = 1, verbose = TRUE)
```

Arguments

X	A numeric matrix, assumed to be generated from a multivariate Gaussian distribution.
burnin	An integer representing the number of burn-in iterations.
iterations	An integer representing the length of the Markov chain post burn-in.
sig	A numeric value representing the standard deviation parameter for the double Gaussian and truncated Gaussian prior.
verbose	A logical indicating if the MCMC sampler progress should be printed.

Value

A list containing precision ‘Omega‘ and covariance ‘Sigma‘ matrices from the Markov chains.

Examples

```
# Generate true precision matrix:
p             <- 10
n             <- 500
OmegaTrue    <- pracma::Toeplitz(c(0.7^rep(1:p-1)))
SigTrue      <- pracma::inv(OmegaTrue)
# Generate expected value vector:
mu            <- rep(0,p)
# Generate multivariate normal distribution:
set.seed(123)
X             <- MASS::mvrnorm(n, mu = mu, Sigma = SigTrue)
# Generate posterior distribution:
posterior     <- blockBGR(X, iterations = 1000, burnin = 500, sig = 0.5)
# Estimated precision matrix using the mean of the posterior:
OmegaEst      <- apply(simplify2array(posterior$Omega), 1:2, mean)
```

Index

blockBAGENI, 2
blockBAGENII, 3
blockBAGL, 4
blockBAGR, 5
blockBGEN, 6
blockBGL, 7
blockBGR, 8