

# The sbgcop Package

February 16, 2008

**Title** Semiparametric Bayesian Gaussian copula estimation

**Version** 0.95

**Date** 2007-03-09

**Author** Peter Hoff

**Maintainer** Peter Hoff <hoff@stat.washington.edu>

**Description** This package estimates parameters of a Gaussian copula, treating the univariate marginal distributions as nuisance parameters as described in Hoff(2007). It also provides a semiparametric imputation procedure for missing multivariate data.

**License** GPL Version 2 or later

**URL** <http://www.stat.washington.edu/hoff>

## R topics documented:

ldmvnorm . . . . .	1
plotci.sA . . . . .	2
qM.sM . . . . .	3
rwish . . . . .	3
sR.sC . . . . .	4
sbgcop.mcmc . . . . .	5
sbgcop-package . . . . .	6
<b>Index</b>	<b>7</b>

ldmvnorm

*Log Multivariate Normal Density*

---

**Description**

Computes the log of the multivariate normal density

**Usage**

```
ldmvnorm(Y, S)
```

**Arguments**

Y                    an n x p matrix  
S                    a p x p positive definite matrix

**Details**

This function computes the log density of the data matrix Y under the model that the rows are independent samples from a mean-zero multivariate normal distribution with covariance matrix S.

**Value**

A real number.

**Author(s)**

Peter Hoff

**Examples**

```
Y<-matrix(rnorm(9*7),9,7)  
ldmvnorm(Y,diag(7))
```

---

plotci.sA*Plot Confidence Bands for Association Parameters*

---

**Description**

Plots 95 parameters

**Usage**

```
plotci.sA(sA, ylabs = colnames(sA[, , 1]), mgp = c(1.75, 0.75, 0))
```

**Arguments**

<code>sA</code>	a $p \times p \times nsamp$ array
<code>ylabs</code>	a $p \times 1$ vector of names for plotting labels
<code>mgp</code>	margin parameters

**Author(s)**

Peter Hoff

---

`qM.sM`

*Matrix Quantiles*

---

**Description**

Computes quantiles along the third dimension of a 3-d array.

**Usage**

```
qM.sM(sM, quantiles = c(0.025, 0.5, 0.975))
```

**Arguments**

<code>sM</code>	an $m \times n \times s$ array
<code>quantiles</code>	quantiles to be computed

**Value**

an array of dimension  $m \times n \times l$ , where  $l$  is the length of `quantiles`

**Author(s)**

Peter Hoff

---

 rwish

*Sample from the Wishart Distribution*


---

**Description**

Generate a random sample from the Wishart distribution.

**Usage**

```
rwish(S0, nu)
```

**Arguments**

S0            a positive definite matrix  
 nu            a positive integer

**Details**

Return the sum of nu i.i.d. rank-one matrices generated as  $z z^t$  ( $z$ ), where  $z$  is a sample from a multivariate normal distribution with covariance  $S0$ . The resulting random variable has mean  $nu * S0$ .

**Value**

a positive definite matrix.

**Author(s)**

Peter Hoff

---

 sR.sC

*Compute Regression Parameters*


---

**Description**

Compute an array of regression parameters from an array of correlation parameters.

**Usage**

```
sR.sC(sC)
```

**Arguments**

sC            a  $p \times p \times nsamp$  array of, made up of  $nsamp$  correlation matrices.

**Details**

For each of the nsamp correlation matrices C, a matrix of regression parameters is computed via  $R[j, -j] \leftarrow C[j, -j] \%*\% \text{solve}(C[-j, -j])$

**Value**

a  $p \times p \times \text{nsamp}$  array of regression parameters.

**Author(s)**

Peter Hoff

---

sbgcop.mcmc

*Semiparametric Bayesian Gaussian copula estimation*


---

**Description**

sbgcop.mcmc is used to semiparametrically estimate the parameters of a Gaussian copula. It can be used for posterior inference on the copula parameters, or for imputation of missing values in matrix-valued data.

**Usage**

```
sbgcop.mcmc(Y, S0 = diag(dim(Y)[2]), n0 = dim(Y)[2] + 2, nsamp = 100,
  odens = max(1, round(nsamp/1000)), seed = 1, verb = TRUE)
```

**Arguments**

Y	an $n \times p$ matrix. Missing values are allowed.
S0	a $p \times p$ positive definite matrix
n0	a positive integer
nsamp	number of iterations of the Markov chain.
odens	output density: number of iterations between saved samples.
seed	an integer for the random seed
verb	print progress of MCMC(TRUE/FALSE)?

**Details**

This function produces MCMC samples from the posterior distribution of a correlation matrix, using a scaled inverse-Wishart prior distribution and an extended rank likelihood. It also provides imputation for missing values in a multivariate dataset.

**Value**

An object of class `psgc` containing the following components:

<code>C.psamp</code>	an array of size $p \times p \times nsamp/odens$ , consisting of posterior samples of the correlation matrix.
<code>Y.pmean</code>	the original datamatrix with imputed values replacing missing data
<code>LPC</code>	the log-probability of the latent variables at each saved sample. Used for diagnostic purposes.

**Author(s)**

Peter Hoff

**References**

<http://www.stat.washington.edu/hoff/>

**Examples**

```
fit<-sbgcop.mcmc(swiss)
summary(fit)
plot(fit)
```

---

sbgcop-package

*Semiparametric Bayesian Gaussian copula estimation*

---

**Description**

This package estimates parameters of a Gaussian copula, treating the univariate marginal distributions as nuisance parameters as described in Hoff(2007). It also provides a semiparametric imputation procedure for missing multivariate data.

**Details**

Package:	sbgcop
Type:	Package
Version:	0.95
Date:	2007-03-09
License:	GPL Version 2 or later

This function produces MCMC samples from the posterior distribution of a correlation matrix, using a scaled inverse-Wishart prior distribution and an extended rank likelihood. It also provides imputation for missing values in a multivariate dataset.

**Author(s)**

Peter Hoff <hoff@stat.washington.edu>

**References**

Hoff (2007) “Extending the rank likelihood for semiparametric copula estimation”

**Examples**

```
fit<-sbgcop.mcmc(swiss)
summary(fit)
plot(fit)
```

# Index

\*Topic **array**

plotci.sA, 2

qM.sM, 3

sR.sC, 4

\*Topic **datagen**

rwish, 3

\*Topic **distribution**

ldmvnorm, 1

rwish, 3

\*Topic **models**

sbgcop.mcmc, 5

\*Topic **multivariate**

ldmvnorm, 1

qM.sM, 3

rwish, 3

sbgcop-package, 6

sbgcop.mcmc, 5

sR.sC, 4

\*Topic **regression**

sR.sC, 4

ldmvnorm, 1

plot.psgc (*sbgcop.mcmc*), 5

plotci.sA, 2

print.sum.psgc (*sbgcop.mcmc*), 5

qM.sM, 3

rwish, 3

sbgcop (*sbgcop-package*), 6

sbgcop-package, 6

sbgcop.mcmc, 5

sR.sC, 4

summary.psgc (*sbgcop.mcmc*), 5