Package ‘cmvnorm’

May 20, 2019

Type Package
Title The Complex Multivariate Gaussian Distribution
Version 1.0-6
Depends emulator (>= 1.2-15)
Suggests knitr
Imports elliptic
Maintainer Robin K. S. Hankin <hankin.robin@gmail.com>
Description Various utilities for the complex multivariate Gaussian distribution.
VignetteBuilder knitr
License GPL-2
URL https://github.com/RobinHankin/cmvnorm.git
BugReports https://github.com/RobinHankin/cmvnorm/issues
NeedsCompilation no
Author Robin K. S. Hankin [aut, cre] (<https://orcid.org/0000-0001-5982-0415>)
Repository CRAN
Date/Publication 2019-05-20 21:20:03 UTC

R topics documented:

cmvnorm-package ........................................ 2
corr_complex ......................................... 3
isHermitian .......................................... 5
Mvnorm ............................................... 6
setreal ............................................. 7
var ................................................. 8

Index 10
The Complex Multivariate Gaussian Distribution

Description

Various utilities for the complex multivariate Gaussian distribution.

Details

The DESCRIPTION file:

<table>
<thead>
<tr>
<th>Package</th>
<th>cmvnorm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Package</td>
</tr>
<tr>
<td>Title</td>
<td>The Complex Multivariate Gaussian Distribution</td>
</tr>
<tr>
<td>Version</td>
<td>1.0-6</td>
</tr>
<tr>
<td>Authors@R</td>
<td>person(given=c(&quot;Robin&quot;, &quot;K. S.&quot;), family=&quot;Hankin&quot;, role = c(&quot;aut&quot;,&quot;cre&quot;), email=&quot;<a href="mailto:hankin.robin@gmail.com">hankin.robin@gmail.com</a>&quot;)</td>
</tr>
<tr>
<td>Depends</td>
<td>emulator (&gt;= 1.2-15)</td>
</tr>
<tr>
<td>Suggests</td>
<td>knitr</td>
</tr>
<tr>
<td>Imports</td>
<td>elliptic</td>
</tr>
<tr>
<td>Maintainer</td>
<td>Robin K. S. Hankin <a href="mailto:hankin.robin@gmail.com">hankin.robin@gmail.com</a></td>
</tr>
<tr>
<td>Description</td>
<td>Various utilities for the complex multivariate Gaussian distribution.</td>
</tr>
<tr>
<td>VignetteBuilder</td>
<td>knit</td>
</tr>
<tr>
<td>License</td>
<td>GPL-2</td>
</tr>
<tr>
<td>URL</td>
<td><a href="https://github.com/RobinHankin/cmvnorm.git">https://github.com/RobinHankin/cmvnorm.git</a></td>
</tr>
<tr>
<td>BugReports</td>
<td><a href="https://github.com/RobinHankin/cmvnorm/issues">https://github.com/RobinHankin/cmvnorm/issues</a></td>
</tr>
<tr>
<td>Author</td>
<td>Robin K. S. Hankin [aut, cre] (<a href="https://orcid.org/0000-0001-5982-0415">https://orcid.org/0000-0001-5982-0415</a>)</td>
</tr>
</tbody>
</table>

Index of help topics:

<table>
<thead>
<tr>
<th>Help Topic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Im&lt;-</td>
<td>Manipulate real or imaginary components of an object</td>
</tr>
<tr>
<td>Mvcmnorm</td>
<td>Multivariate complex Gaussian density and random deviates</td>
</tr>
<tr>
<td>cmvnorm-package</td>
<td>The Complex Multivariate Gaussian Distribution</td>
</tr>
<tr>
<td>corr_complex</td>
<td>Complex Gaussian processes</td>
</tr>
<tr>
<td>isHermitian</td>
<td>Is a Matrix Hermitian?</td>
</tr>
<tr>
<td>var</td>
<td>Variance and standard deviation of complex vectors</td>
</tr>
</tbody>
</table>

Generalizing the real multivariate Gaussian distribution to the complex case is not straightforward but one common approach is to replace the real symmetric variance matrix with a Hermitian positive-definite matrix. The `cmvnorm` package provides some functionality for the resulting density function.

Author(s)

NA
Maintainer: Robin K. S. Hankin <hankin.robin@gmail.com>

References


Examples

```r
S1 <- 4+diag(5)
S2 <- S1
S2[1,5] <- 4+1i
S2[5,1] <- 4-1i # Hermitian
rcmvnorm(10,sigma=S1)
rcmvnorm(10,mean=rep(1i,5),sigma=S2)
dcmvnorm(rep(1,5),sigma=S2)
```

---

corr_complex  
*Complex Gaussian processes*

Description

Various utilities for investigating complex Gaussian processes

Usage

```r
corr_complex(z1, z2 = NULL, distance.function = complex_CF, means = NULL, scales = NULL, pos.def.matrix = NULL)
complex_CF(z1,z2, means, pos.def.matrix)
scales.likelihood.complex(pos.def.matrix, scales, means, zold, z, give_log = TRUE, func = regressor.basis)
interpolant.quick.complex(x, d, zold, Ainv, scales = NULL, pos.def.matrix = NULL, means=NULL, func = regressor.basis, give.Z = FALSE, distance.function = corr_complex, ...)
```

Arguments

- `z, z1, z2` Points in $C^n$
- `distance.function` Function giving the (complex) covariance between two points in $C^n$
means, pos.def.matrix, scales

In function complex_cf(), the mean and covariance matrix of the distribution whose characteristic function is used as to give the covariance matrix; scales is used to specify the diagonal of pos.def.matrix if the off-diagonal elements are zero.

zold, d, give_log, func, x, Ainv, give.Z,...

Direct analogues of the arguments in interpolant() and scales.likelihood() in the emulator package.

Details

- Function complex_cf() returns a (slightly reparameterized) characteristic function of a complex Gaussian distribution. The covariance is given by

\[ c(t) = \exp(i \text{Re}(t^* \mu) - t^* B t) \]

where \( t = x - x' \) is interpreted as the distance between two observations, \( \mu \) is the mean of the distribution (which is in general a complex vector), and \( B \) a positive-definite matrix.

- Function corr_complex() is the complex analogue of corr.matrix(). It returns a matrix with entry \((i,j)\) equal to the covariance of the process at observation \(i\) and observation \(j\), or \(\text{cov}(\eta(x_i), \eta(x_j))\). The elements are calculated by complex_cf().

This function includes only a single method, that of nested calls to apply(). I could not figure out how to generalize method 1 of corr.matrix() to the complex case.

- Function scales.likelihood.complex() is a complex version of scales.likelihood() which takes a positive definite matrix and a mean. The formula used is

\[ (\sigma^2)^{-(n-q)} |A|^{-1} |H A^{-1} H^*|^\frac{1}{2} \]

. Here and elsewhere, \(A^*\) means the complex conjugate of the transpose.

- Function interpolant.quick.complex() is a complex version of interpolant.quick().

\[ h(x)^* \hat{\beta} + t(x)^* A^{-1} (y - H \hat{\beta}) \]

This is the complex version of Oakley’s equation 2.30 or Hankin’s equation 5.

More details are given in the package vignette.

Author(s)

Robin K. S. Hankin

References


**isHermitian**

## Examples

```r
complex_CF(c(1,1i),c(1,-1i),means=c(1i,1i),pos.def.matrix=diag(2))
V <- latin.hypercube(7,2,complex=TRUE)

cm <- c(1,1+1i)  # "complex mean"
cs <- matrix(c(2,1i,-1i,1,2,2))  # "complex scales"
tb <- c(1,1i,1-1i)  # "true beta"

A <- corr_complex(V,means=cm,pos.def.matrix=cs)
Ainv <- solve(A)
z <- drop(rcmvnorm(n=1,mean=regressor.multi(V) %*% tb, sigma=A))

betahat.fun(V,Ainv,z)  # should be close to 'tb'
#scales.likelihood.complex(cs,cm,V,z)  # log-likelihood evaluated true parameters

interpolant.quick.complex(x=0.1i+V[1:3],d=z,zold=V,Ainv=Ainv,pos.def.matrix=cs,means=cm)
```

---

### isHermitian

**Is a Matrix Hermitian?**

## Description

Returns TRUE if a matrix is Hermitian or Hermitian positive-definite

## Usage

```r
ishermitian(x, tol = 100 * .Machine$double.eps)
ishpd(x, tol = 100 * .Machine$double.eps)
zapim(x, tol = 100 * .Machine$double.eps)
```

## Arguments

- **x**  
  A square matrix

- **tol**  
  Tolerance for numerical scruff

## Details

Functions `ishermitian()` and `ishpd()` return a Boolean. Function `zapim()` zaps small imaginary parts of components vector, returning real if all elements are so zapped.
Author(s)

Robin K. S. Hankin

Examples

```r
v <- 2*(1:30)
zapim(v+1i*exp(-v))

ishpd(matrix(c(1,0.1i,-0.1i,1,2,2))) # should be TRUE
ishermitian(matrix(c(1,3i,-3i,1,2,2))) # should be TRUE
```

---

Mvncnorm

*Multivariate complex Gaussian density and random deviates*

Description

Density function and a random number generator for the multivariate complex Gaussian distribution.

Usage

```r
rcnorm(n)
dcmvnorm(z, mean, sigma, log = FALSE)
rcmvnorm(n, mean = rep(0, nrow(sigma)), sigma = diag(length(mean)),
  method = c("svd", "eigen", "chol"),
  tol = 100 * .Machine$double.eps)
```

Arguments

- `z`  
  Complex vector or matrix of quantiles. If a matrix, each row is taken to be a quantile

- `n`  
  Number of observations

- `mean`  
  Mean vector

- `sigma`  
  Covariance matrix, Hermitian positive-definite

- `tol`  
  Numerical tolerance term for verifying positive definiteness

- `log`  
  In dcmvnorm(), Boolean with default TRUE meaning to return the Gaussian density function, and FALSE meaning to return the logarithm

- `method`  
  Matrix decomposition used to determine the positive-definite matrix square root of sigma, possible methods are eigenvalue decomposition ("eigen", default), and singular value decomposition ("svd")
setreal

Details

Function dcmvnorm() is the density function of the complex multivariate normal (Gaussian) distribution:

\[ p(z) = \frac{\exp(-z^* \Gamma z)}{|\pi \Gamma|} \]

Function rcnorm() is a low-level function designed to generate observations drawn from a standard complex Gaussian. Function rcmvnorm() is a user-friendly wrapper for this.

Author(s)

Robin K. S. Hankin

References


Examples

```r
S <- emulator::cprod(rcmvnorm(3, mean = c(1, 1i), sigma = diag(2)))
rcmvnorm(10, sigma = S)
rcmvnorm(10, mean = c(0, 1+10i), sigma = S)

# Now try and estimate the mean (viz 1, 1i) and variance (S) from a
# random sample:

n <- 101
z <- rcmvnorm(n, mean = c(0, 1+10i), sigma = S)
xbar <- colMeans(z)
Sbar <- cprod(sweep(z, 2, xbar))/n
```

setreal

Manipulate real or imaginary components of an object

Description

Manipulate real or imaginary components of an object
var

Usage

\[
\begin{align*}
\text{Im}(x) & \leftarrow \text{value} \\
\text{Re}(x) & \leftarrow \text{value}
\end{align*}
\]

Arguments

- \textit{x} \hspace{1em} \text{Complex-valued object}
- \textit{value} \hspace{1em} \text{Real-valued object}

Author(s)

Robin K. S. Hankin

Examples

\[
\begin{align*}
A & \leftarrow \text{matrix}(c(1,0.1i,-0.1i,1),2,2) \\
\text{Im}(A) & \leftarrow \text{Im}(A)*3 \\
\text{Re}(A) & \leftarrow \text{matrix}(c(5,2,2,5),2,2)
\end{align*}
\]

---

\begin{tabular}{ll}
\textbf{var} & \textit{Variance and standard deviation of complex vectors} \\
\end{tabular}

Description

Complex generalizations of \texttt{stats::sd()} and \texttt{stats::var()}

Usage

\[
\begin{align*}
\text{var}(x, \ y=\text{NULL}, \ \text{na.rm}=\text{FALSE}, \ \text{use}) \\
\text{sd}(x, \ \text{na.rm}=\text{FALSE})
\end{align*}
\]

Arguments

- \textit{x, y} \hspace{1em} \text{Complex vector or matrix}
- \textit{na.rm} \hspace{1em} \text{Boolean with default FALSE meaning to leave NA values present and TRUE meaning to remove them}
- \textit{use} \hspace{1em} \text{Ignored}
Details

Intended to be broadly compatible with stats::sd() and stats::var().

If given real values, \texttt{var()} and \texttt{sd()} return the variance and standard deviation as per ordinary real analysis. If given complex values, returns the complex generalization in which Hermitian transposes are used.

If \( z \) is a complex matrix, \texttt{var}(z) returns the variance of the rows.

These functions use \( n - 1 \) on the denominator purely for consistency with \texttt{stats::var()} (for the record, I disagree with the rationale for \( n - 1 \)).

Author(s)

Robin K. S. Hankin

Examples

\begin{verbatim}
sd(rclnorm(10))  # imaginary component suppressed by zapim()

var(rcmvnorm(1e5,mean=c(0,0)))
\end{verbatim}
Index

*Topic complex
  isHermitian, 5
  Mvcnorm, 6
  setreal, 7
*Topic distribution
  Mvcnorm, 6
*Topic math
  setreal, 7
*Topic multivariate
  Mvcnorm, 6
*Topic package
  cmvnorm-package, 2

  cmvnorm (cmvnorm-package), 2
  cmvnorm-package, 2
  complex_CF (corr_complex), 3
  corr_complex, 3

dcmvnorm (Mvcnorm), 6
Im<-(setreal), 7
interpolant.quick.complex
  (corr_complex), 3
isHermitian, 5
ishpd (isHermitian), 5

Mvcnorm, 6

rcmvnorm (Mvcnorm), 6
rcnorm (Mvcnorm), 6
Re<-(setreal), 7

scales.likelihood.complex
  (corr_complex), 3
sd (var), 8
setreal, 7

var, 8

zapim (isHermitian), 5