Package ‘VisCov’

July 21, 2025

Title Visualizing of Distributions of Covariance Matrices

Version 1.6.0

Maintainer Kristof Meers <kristof.meers+cran@kuleuven.be>

Depends R (>= 2.12.0)

Imports bayesm, clusterGeneration, scatterplot3d, KernSmooth, trialr, methods

Description Visualizing of distributions of covariance matrices.
The package implements the methodology described in Tokuda, T., Goodrich, B., Van Mechelen, I., Gelman, A., & Tuerlinckx, F. (2012) <https://sites.stat.columbia.edu/gelman/research/unpublished/Visualization.pdf>.

License GPL (>= 3)

NeedsCompilation no

Language en-US

RoxygenNote 7.3.2

Encoding UTF-8

Author Tomoki Tokuda [aut],
Ben Goodrich [ctb],
Iven Van Mechelen [ctb],
Andrew Gelman [ctb],
Francis Tuerlinckx [ctb],
Kristof Meers [cre]

Repository CRAN

Date/Publication 2025-04-09 23:10:05 UTC

Contents

`panelSelect` *Selecting a Panel(s) using the Mouse*

Description

To select (using the mouse) one or more panels from a four-layered plot for a single distribution. The selected panels will be drawn in a separate graphics window.

Usage

```
panelSelect(panel.no, CovPlotData)
```

Arguments

`panel.no`	an integer indicating the number of panels that one wishes to select.
`CovPlotData`	the output list returned by `VisCov`.

Details

After drawing a graph by `VisCov`, one can select and draw the specified number of panels by clicking the mouse on the panels of interest.

Value

No return value, called for side effects.

References

Tokuda, T., Goodrich, B., Van Mechelen, I., Gelman, A. and Tuerlinckx, F. (submitted). Visualizing Distributions of Covariance Matrices.

Examples

```
set.seed(1234)
CovPlotData = VisCov()
panelSelect(panel.no = 1, CovPlotData) # Click once on the plot you want to select
panelSelect(panel.no = 4, CovPlotData) # Click four times on the plots you want to select
```

panelSelectCorr — *Selecting a Panel(s) using the Mouse*

Description

To visualize relevant parameters for correlation matrix, removing those panels related to variances.

Usage

```
panelSelectCorr(CovPlotData, range.logical.contour = FALSE,
                range.logical.all = TRUE)
```

Arguments

`CovPlotData` the output list returned by `VisCov`.

`range.logical.all`
: a logical value indicating whether the ranges in the panels are set as the same over different sets of samples. For the histograms, the ranges of frequencies are set as the same over the different sets of samples.

`range.logical.contour`
: a logical value indicating whether the ranges of the contour plots are set as the same over different sets of samples when `range.logical.all` is TRUE.

Details

After drawing a graph by `VisCov`, one can select and draw relevant panels for correlation matrix.

Value

No return value, called for side effects.

References

Tokuda, T., Goodrich, B., Van Mechelen, I., Gelman, A. and Tuerlinckx, F. (submitted). Visualizing Distributions of Covariance Matrices.

Examples

```
set.seed(1234)
distribution = "LKJ"
eta = 5
dim = 50
param = list(prob = 0.5, dim = dim, eta=eta, scaleCov = diag(1,dim))
CovPlotData = VisCov(distribution, param,title.logical = FALSE)
panelSelectCorr(CovPlotData)
```

panelSelectMultiple *Selecting Panels From Several Distributions*

Description

To draw the selected panels from several distributions generated by `VisCov`.

Usage

```
panelSelectMultiple(selected.condition, CovPlotDataMultiple,
                    range.logical.contour = FALSE, range.logical.all = TRUE,
                    row = FALSE)
```

Arguments

`selected.condition`
: a vector of strings containing types of the panels that one wishes to select: For layer 1, "vari", "cor"; for layer 2, "scatter1", "scatter2", "scatter3", "scatter4", "scatter5"; for layer 3, "contour", "threeD"; for layer 4, "Effective.Variance", "Effective.Dependence", "Effective.Dependence.submatrix".

`CovPlotDataMultiple`
: a list of objects returned by `VisCov`.

`range.logical.all`
: a logical value indicating whether the ranges in the panels are set as the same over different sets of samples. For the histograms, the ranges of frequencies are set as the same over the different sets of samples.

`range.logical.contour`
: a logical value indicating whether the ranges of the contour plots are set as the same over different sets of samples when `range.logical.all` is TRUE.

`row`
: a logical value whether the panels belonging to the same set of samples are arranged in the same row. If it is FALSE, they are arranged in the same column.

Value

No return value, called for side effects.

References

Tokuda, T., Goodrich, B., Van Mechelen, I., Gelman, A. and Tuerlinckx, F. (submitted). Visualizing Distributions of Covariance Matrices.

Examples

```
set.seed(1234)
distribution = "Inverse Wishart"
dim = 4
param = list(prob = 0.5, dim = dim, nu = dim+1, scaleCov = diag(1,dim))
```

```
CovPlotData1 = VisCov(distribution, param, title.logical = FALSE)
distribution = "Inverse Wishart"
dim = 4
param = list(prob = 0.5, dim = dim, nu = dim+50, scaleCov = diag(1,dim))
CovPlotData2 = VisCov(distribution, param, title.logical = FALSE)
selected.condition = c("scatter1", "scatter4", "contour", "Effective.Dependence")
panelSelectMultiple(selected.condition, list(CovPlotData1, CovPlotData2))
```

`VisCov` *Visualizing of Distributions of Covariance Matrices*

Description

To generate samples from a specific distribution of covariance matrices, draw the four-layered graphs and return the sampled matrices and the related information on the distribution.

Usage

```
VisCov(distribution = "Inverse Wishart", param = list(prob = 0.5, dim = 4,
       nu = 5, eta = 1, scaleCov = diag(1,4)), title = distribution,
       Nsamples = 1000, Ncontours = 100, logSD = TRUE,
       histogram.Variance = TRUE, histogram.Correlation = TRUE,
       histogram.Effective.Variance = TRUE,
       histogram.Effective.Dependence = TRUE,
       extreme.regio = "Effective Dependence", title.logical = TRUE)
```

Arguments

`distribution` a distribution to be plotted among:

"Inverse Wishart", "Scaled Inverse Wishart", "Scaled Inverse Wishart for correlation", "Scaled with uniform on correlation", "Wishart", "LKJ", "Scaled LKJ", and "User defined distribution".

`title.logical` a logical value indicating whether a title should be given to the graph.

`title` title of the graph.

`Nsamples` the number of samples for all plots except the contour plot.

`Ncontours` the number of samples for the contour plot.

`logSD` a logical value indicating whether the natural logarithm of the standard deviation is plotted.

`histogram.Variance`
a logical value indicating whether the univariate graph of the (log) standard deviation is a histogram (TRUE) or a density (FALSE).

`histogram.Correlation`
a logical value indicating whether the univariate graph of the correlation is a histogram (TRUE) or a density curve (FALSE).

histogram.Effective.Variance
: a logical value indicating whether the univariate graph of the effective variance is a histogram (TRUE) or a density curve (FALSE).

histogram.Effective.Dependence
: a logical value indicating whether the univariate graph of the effective dependence is a histogram (TRUE) or a density curve (FALSE).

extreme.regio
: indicates whether the coloring of extreme samples (in red and blue) is based on the "Effective Dependence" or "Effective Variance". If neither of them is specified, there is no coloring.

param
: a list of parameters consisting of the following objects: mat, a list of covariance matrices; applicable only when distribution = "User defined distribution"; prob, mass concentration in the equiprobability contour plot; dim, dimension of matrices; nu, degrees of freedom for the inverse Wishart or Wishart distributions; scaleCov, scale matrix for the inverse Wishart or Wishart distributions; mu0 and s0 are the mean and the standard deviation for the folded normal distribution, applicable when distribution is "Scaled Inverse Wishart", "Scaled Inverse Wishart for correlation" or "Scaled with uniform on correlation"; eta, shape parameter for the LKJ distribution.

Details

VisCov generates samples from a specific distribution of covariance matrices (or user defined distribution), draws the four-layered graphs and returns the sampled matrices and the related information on the distribution.

For more detail, see Tokuda et al. in the following reference.

Value

A list of variables to pass for selecting panels.

References

Tokuda, T., Goodrich, B., Van Mechelen, I., Gelman, A. and Tuerlinckx, F. (submitted). Visualizing Distributions of Covariance Matrices.

Examples

```
## Figures in the paper (Tokuda, Goodrich, Van Mechelen, Gelman, and Tuerlinckx)
## Example 1 (Figure 1 in the paper)
set.seed(1234)
distribution = "Inverse Wishart"
dim = 4
param = list(prob = 0.5, dim = dim, nu = dim+1, scaleCov = diag(1,dim))
CovPlotData = VisCov(distribution, param, title.logical = FALSE)

## Example 2  (Figure 2 in the paper)
set.seed(1234)
distribution = "Inverse Wishart"
dim = 4
```

```
param = list(prob = 0.5, dim = dim, nu = dim+1, scaleCov = diag(1,dim))
CovPlotData1 = VisCov(distribution, param, title.logical = FALSE)
distribution = "Inverse Wishart"
dim = 4
param = list(prob = 0.5, dim = dim, nu = dim+50, scaleCov = diag(1,dim))
CovPlotData2 = VisCov(distribution, param, title.logical = FALSE)
selected.condition = c("scatter1", "scatter4", "contour", "Effective.Dependence")
panelSelectMultiple(selected.condition, list(CovPlotData1, CovPlotData2))

## Example 3  (Figure 3 in the paper)
set.seed(1234)
distribution = "Inverse Wishart"
dim = 4
param = list(prob = 0.5, dim = dim, nu = dim+1, scaleCov = diag(1,dim))
CovPlotData1 = VisCov(distribution, param, title.logical = FALSE)
dim = 100
param = list(prob = 0.5, dim = dim, nu = dim+1, scaleCov = diag(1,dim))
CovPlotData2 = VisCov(distribution, param, title.logical = FALSE)
selected.condition = c("Effective.Dependence")
panelSelectMultiple(selected.condition, list(CovPlotData1, CovPlotData2))

## Example 4  (Figure 4 in the paper)
set.seed(1234)
distribution = "Scaled Inverse Wishart for correlation"
dim = 4
param = list(mu0 = 0, s0 = 1, prob = 0.5, dim = dim, nu = dim+1, scaleCov = diag(1,dim))
CovPlotData1 = VisCov(distribution, param, title.logical = FALSE)
param = list(mu0 = 0, s0 = 1,prob = 0.5, dim = dim, nu = dim+50, scaleCov = diag(1,dim))
CovPlotData2 = VisCov(distribution, param, title.logical = FALSE)
selected.condition = c("scatter1", "scatter2", "scatter4", "contour")
panelSelectMultiple(selected.condition, list(CovPlotData1, CovPlotData2))

## Example 5  (Figure 5 in the paper)
set.seed(1234)
distribution = "Scaled with uniform on correlation"
dim = 4
param = list(mu0 = 0, s0 = 1, prob = 0.5, dim = dim)
CovPlotData1 = VisCov(distribution, param, title.logical = FALSE)
dim = 50
param = list(mu0 = 0, s0 = 1, prob = 0.5, dim = dim)
CovPlotData2 = VisCov(distribution, param,title.logical = FALSE)
selected.condition = c("scatter2", "scatter4", "contour", "Effective.Dependence.submatrix")
panelSelectMultiple(selected.condition, list(CovPlotData1, CovPlotData2))

## Example 6  (Figure 6 in the paper)
set.seed(1234)
distribution = "LKJ"
eta = 5
dim = 100
param = list(prob = 0.5, dim = dim, eta=eta, scaleCov = diag(1,dim))
CovPlotData = VisCov(distribution, param,title.logical = FALSE)
panelSelectCorr(CovPlotData)
```

```
## Example 7  (Figure 7 in the paper)
set.seed(1234)
distribution = "Scaled Inverse Wishart"
dim = 4
param = list(mu0 = 0, s0 = 1, prob = 0.5, dim = dim, nu = dim+1, scaleCov = diag(1,dim))
CovPlotData1 = VisCov(distribution, param, title.logical = FALSE)
param = list(mu0 = 0, s0 = 1,prob = 0.5, dim = dim, nu = dim+50, scaleCov = diag(1,dim))
CovPlotData2 = VisCov(distribution, param, title.logical = FALSE)
selected.condition = c("scatter2", "scatter4", "contour", "Effective.Dependence")
panelSelectMultiple(selected.condition, list(CovPlotData1,CovPlotData2))

## Example 8  (Figure 8 in the paper)
distribution = "Wishart"
set.seed(1234)
dim = 4
param = list(prob = 0.5, dim = dim, nu = dim+1, scaleCov = diag(1,dim))
CovPlotData1 = VisCov(distribution, param, title.logical = FALSE)
dim = 50
param = list(prob = 0.5, dim = dim, nu = dim+1, scaleCov = diag(1,dim))
CovPlotData2 = VisCov(distribution, param,title.logical = FALSE)
selected.condition = c("scatter2", "scatter4", "contour", "Effective.Dependence")
panelSelectMultiple(selected.condition, list(CovPlotData1, CovPlotData2))

## Example 9  (Figure 9 in the paper)
set.seed(1234)
distribution = "User defined distribution"
param = list()
mat = list()
# Generation of covariance matrices
for (i in 1:1000){
  index = 0
  dim = 50
  while (index == 0){
    W = matrix(rnorm(dim * dim, 0, 1), nrow = dim)
    svdW= svd(W)
    Rho = svdW$u %*% t(svdW$v)
    D = diag(rbeta(dim, 0.5, 5))
    Sigma = diag(rnorm(dim, 0, 1))
    Sigma = abs(Sigma)
    S = Sigma %*% Rho %*% D %*% t(Rho) %*% Sigma
    # Checking positive definiteness
    eigenv = eigen(S)$values
    if (all(eigenv > 0)) index = 1
    if (index == 0){ print("zero") }
    mat[[i]] = S
  }
}
param = list(prob = 0.5, mat = mat)
CovPlotData = VisCov(distribution, param , title.logical = FALSE)
```

Index