Package ‘MVNtestchar’

July 25, 2020

Type Package

Title Test for Multivariate Normal Distribution Based on a Characterization

Version 1.1.3

Date 2020-07-14

Description Provides a test of multivariate normality of an unknown sample that does not require estimation of the nuisance parameters, the mean and covariance matrix. Rather, a sequence of transformations removes these nuisance parameters and results in a set of sample matrices that are positive definite. These matrices are uniformly distributed on the space of positive definite matrices in the unit hyper-rectangle if and only if the original data is multivariate normal (Fairweather, 1973, Doctoral dissertation, University of Washington). The package performs a goodness of fit test of this hypothesis. In addition to the test, functions in the package give visualizations of the support region of positive definite matrices for bivariate samples.

Depends R (>= 2.10)

Imports graphics, grDevices, Hmisc, stats, utils, knitr, ggplot2

License GPL (>= 2)

NeedsCompilation no

Suggests markdown

VignetteBuilder knitr, markdown

Author William Fairweather [aut, cre]

Maintainer William Fairweather <wrf343@flowervalleyconsulting.com>

Repository CRAN

Date/Publication 2020-07-25 21:30:26 UTC

R topics documented:

MVNtestchar-package .................................................. 2
maxv12 ......................................................................... 4
slice.v1 ........................................................................ 5
MVNtestchar-package

Description

Provides a test of multivariate normality of an unknown sample that does not require estimation of the nuisance parameters, the mean and covariance matrix. Rather, a sequence of transformations removes these nuisance parameters and results in a set of sample matrices that are positive definite. These matrices are uniformly distributed on the space of positive definite matrices in the unit hyper-rectangle if and only if the original data is multivariate normal (Fairweather, 1973, Doctoral dissertation, University of Washington). The package performs a goodness of fit test of this hypothesis. In addition to the test, functions in the package give visualizations of the support region of positive definite matrices for bivariate samples.

Details

The DESCRIPTION file:

Package: MVNtestchar
Type: Package
Title: Test for Multivariate Normal Distribution Based on a Characterization
Version: 1.1.3
Date: 2020-07-14
Authors@R: person("William", "Fairweather", email = "wrf343@flowervalleyconsulting.com", role = c("aut", "cre"))
Description: Provides a test of multivariate normality of an unknown sample that does not require estimation of the nuisance parameters, the mean and covariance matrix. Rather, a sequence of transformations removes these nuisance parameters and results in a set of sample matrices that are positive definite. These matrices are uniformly distributed on the space of positive definite matrices in the unit hyper-rectangle if and only if the original data is multivariate normal (Fairweather, 1973, Doctoral dissertation, University of Washington). The package performs a goodness of fit test of this hypothesis. In addition to the test, functions in the package give visualizations of the support region of positive definite matrices for bivariate samples.
Depends: R (>= 2.10)
Imports: graphics, grDevices, Hmisc, stats, utils, knitr, ggplot2
License: GPL (>=2)
NeedsCompilation: no
Suggests: markdown
VignetteBuilder: knitr, markdown
Packaged: 2020-03-11 18:35:57 UTC; No
Author: William Fairweather [aut, cre]
Maintainer: William Fairweather <wrf343@flowervalleyconsulting.com>

Index of help topics:
MVNtestchar-package

The MVNtestchar-package provides a test of multivariate normality of a sample which does not require estimation of the nuisance parameters, the mean vector and covariance matrix. Rather, a sequence of transformations removes these nuisance parameters, resulting in a set of sample matrices that are positive definite. If, and only if the original data is multivariate normal, these matrices are uniformly distributed on the space of positive definite matrices in the unit hyper-rectangle. The package performs a goodness of fit test of this hypothesis. In addition to the test, functions in the package give visualizations of the support region of positive definite matrices for p equals 2.

**Author(s)**

person("Fairweather", "William", email = "wrf343@flowervalleyconsulting.com", role = c("aut", "cre"))

**References**


maxv12  

Rotatable Plot of Surface of Possible Maximum Values of Off-diagonal Variable

Description

Rotatable plot of surface of possible maximum values of off-diagonal variable v12 in positive definite 2 x 2 matrix

Usage

maxv12(theta = 30, phi = 30, inc = 25, lseq = 200, ticktype="detailed", diagnose = FALSE, verbose = TRUE)

Arguments

theta  
left-right plot rotation parameter in degrees

phi  
up-down plot rotation parameter in degrees

inc  
increment in degrees of plot rotations

lseq  
number of cut points in v1 and in v2

ticktype  
simple or detailed ticks on variables

diagnose  
Logical. T causes printing of diagnostic content

verbose  
Logical. T causes printing of program ID before and after running

Value

Output is a plot that is rotatable via keyboard input. Upon exit, the latest values of the rotation parameters is listed to facilitate return to the latest plot

Author(s)

William R. Fairweather

See Also

support.p2()

Examples

## Not run: maxv12(theta = 30, phi = 30, inc = 25, lseq = 200, 
ticktype = "detailed", diagnose = FALSE, verbose = TRUE)

## End(Not run)
Description

Rotatable plot of slice through support region in positive definite 2 x 2 matrix at fixed value of diagonal variable v1

Usage

slice.v1(level3 = 0.6, theta = 0, phi = 60, inc = 25, lseq = 100, ticktype="detailed", diagnose = FALSE, verbose = TRUE)

Arguments

level3 Level of V1 where slice is taken
theta left-right plot rotation parameter in degrees
phi up-down plot rotation parameter in degrees
lseq number of cut points in v1 and in v2
inc increment in degrees of plot rotations
ticktype simple or detailed ticks on variables
diagnose Logical. T causes printing of diagnostic content
verbose Logical. T causes printing of program ID before and after running

Value

Output is a plot that is rotatable via keyboard input. Upon exit, the latest values of the rotation parameters is listed to facilitate return to the latest plot

Author(s)

William R. Fairweather

See Also

support.p2()

Examples

```r
## Not run: slice.v1(level3 = 0.6, theta = 0, phi = 60, inc = 25, lseq = 100,
   ticktype = "detailed")
```  
```
## End(Not run)```
### slice.v12

**Rotatable Plot of Slice Through Support Region in Positive Definite 2 x 2 Matrix**

**Description**

Rotatable plot of slice through support region in positive definite 2 x 2 matrix at fixed value of off-diagonal variable v12

**Usage**

```
slice.v12(level3 = 0.3, theta = 30, phi = 10, inc = 25, lseq = 100, ticktype="detailed", diagnose = FALSE, verbose = TRUE)
```

**Arguments**

- `level3`: Level of V1 where slice is taken
- `theta`: left-right plot rotation parameter in degrees
- `phi`: up-down plot rotation parameter in degrees
- `inc`: increment in degrees of plot rotations
- `lseq`: number of cut points in v1 and in v2
- `ticktype`: simple or detailed ticks on variables
- `diagnose`: Logical. T causes printing of diagnostic content
- `verbose`: Logical. T causes printing of program ID before and after running

**Value**

Output is a plot that is rotatable via keyboard input. Upon exit, the latest values of the rotation parameters is listed to facilitate return to the latest plot

**Author(s)**

William R. Fairweather

**See Also**

support.p2()

**Examples**

```r
## Not run: slice.v12(level3 = 0.3, theta = 30, phi = 10, inc = 25, lseq = 100, ticktype = "detailed")
## End(Not run)```
**support.p2**

Show Support Region of Positive Definite Matrices with Rank 2

**Description**
Rotatable plot of support region for positive definite matrix with p=2

**Usage**
```
support.p2(theta = 110, phi = 10, lseq = 150, inc = 25, ticktype="detailed",
diagnose = FALSE, verbose = TRUE)
```

**Arguments**
- `theta`: left-right plot rotation parameter in degrees
- `phi`: up-down plot rotation parameter in degrees
- `lseq`: number of cut points in v1 and in v2
- `inc`: increment in degrees of plot rotations
- `ticktype`: simple or detailed ticks on variables
- `diagnose`: Logical. T causes printing of diagnostic content
- `verbose`: Logical. T causes printing of program ID before and after running

**Details**
Support region for p-variate positive definite matrix distributions is difficult to envision except for p=2. The diagonals of the matrix are V1 and V2 and the off-diagonal variable is V12. In our application 0<=V1,V2<=1, and -1<=V12<=1, so the bounded space is a hyper-rectangle. Each point in this region represents a symmetric pxp matrix, but not all of these are positive definite. This function shades the region of positive definite matrices.

**Value**
Output is a plot that is rotatable via keyboard input. Upon exit, the latest values of the rotation parameters is listed to facilitate return to the latest plot

**Author(s)**
William R. Fairweather

**Examples**
```
## Not run: support.p2(theta = 110, phi = 10, lseq = 150, inc = 25,
ticktype = "detailed")
```
```
## End(Not run)
```
Process the Samples Whose Distribution is to be Tested

Description

Create positive definite matrices without nuisance parameters. Tabulate distribution. Calculate goodness of fit

Usage

testunknown(x, pvector, k, diagnose.s = FALSE, diagnose = FALSE, verbose = TRUE)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>Name of matrix or array.</td>
</tr>
<tr>
<td>pvector</td>
<td>Dimensionality of random vectors</td>
</tr>
<tr>
<td>k</td>
<td>Number of cuts per unit for diagonal elements of matrix. Program uses 2k cuts per unit for off-diagonal elements</td>
</tr>
<tr>
<td>diagnose.s</td>
<td>Logical. T causes printing of diagnostic terms in internal called function(s)</td>
</tr>
<tr>
<td>diagnose</td>
<td>Logical. T causes printing of diagnostic content</td>
</tr>
<tr>
<td>verbose</td>
<td>Logical. T causes printing of function ID before and after running</td>
</tr>
</tbody>
</table>

Value

a list including elements

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distribution</td>
<td>List. Count of pd matrices within individual subcubes of pd space, 1 for each layer of list</td>
</tr>
<tr>
<td>Goodness of fit</td>
<td>List. Chi square test of goodness of fit to uniform distribution, 1 for each layer of list</td>
</tr>
<tr>
<td>Call</td>
<td>Call to testunknown function</td>
</tr>
</tbody>
</table>

Author(s)

William R. Fairweather

References

unknown.Bp2

A Sample From an Unknown Bivariate Distribution

Description
A 3600 x 2 x 1 array generated from 7200 modified Bernoulli(0,1) variables.

Usage
data("unknown.Bp2")

Format
3600 x 2 x 1 array

Source
Generated by the author

Examples
data("unknown.Bp2")

data(unknown.Np2)
testunknown(x=unknown.Np2, pvector=2, k=20,
    diagnose.s = FALSE, diagnose = FALSE, verbose = TRUE)

unknown.Bp4

A Sample From an Unknown Four-variate Distribution

Description
A 6000 x 4 matrix generated from 24,000 Bernoulli(0,1) variables

Usage
data("unknown.Bp4")

Format
6000 x 4 x 1 array

Source
Generated by the author
unknown.Np2  A Sample From an Unknown Bivariate Distribution

**Description**

A 2500 x 2 matrix generated from 5000 normal(0,1) variables

**Usage**

data("unknown.Np2")

**Format**

2500 x 2 matrix

**Source**

Generated by the author

**Examples**

data("unknown.Np2")

unknown.Np4  A Sample From an Unknown Four-variate Distribution

**Description**

A 6000 x 4 x 1 array generated from 24000 normal(0,1) variables

**Usage**

data("unknown.Np4")

**Format**

6000 x 4 x 1 array

**Source**

Generated by the author

**Examples**

data("unknown.Np4")
Index

* ~distribution
  maxv12, 4
  slice.v1, 5
  slice.v12, 6
* ~hplot
  maxv12, 4
  slice.v1, 5
  slice.v12, 6
* ~iplot
  maxv12, 4
  slice.v1, 5
  slice.v12, 6
* ~multivariate
  maxv12, 4
  slice.v1, 5
  slice.v12, 6
* array
  testunknown, 8
* datasets
  unknown.Bp2, 9
  unknown.Bp4, 9
  unknown.Np2, 10
  unknown.Np4, 10
* distribution
  MVNtestchar-package, 2
  support.p2, 7
  testunknown, 8
* hplot
  MVNtestchar-package, 2
  support.p2, 7
* iplot
  MVNtestchar-package, 2
  support.p2, 7
* math
  testunknown, 8
* multivariate
  MVNtestchar-package, 2
  support.p2, 7
  testunknown, 8