

Package ‘mcvis’

June 3, 2020

Type Package

Title Multi-Collinearity Visualization

Version 1.0.4

Description

Visualize the relationship between linear regression variables and causes of multi-collinearity.

Encoding UTF-8

Imports assertthat, igraph, ggplot2, purrr, magrittr, reshape2, shiny,
dplyr, psych, rlang

RoxygenNote 7.1.0

License GPL-3

Suggests testthat (>= 2.1.0), covr, knitr, mplot, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Kevin Wang [aut, cre],
Chen Lin [aut],
Samuel Mueller [aut]

Maintainer Kevin Wang <kevin.wang09@gmail.com>

Repository CRAN

Date/Publication 2020-06-03 16:20:02 UTC

R topics documented:

ggplot.mcvis	2
mcvis	3
plot.mcvis	4
shiny_mcvis	5

Index	6
--------------	----------

Description

The `ggplot_mcvis` function first orders the MC-index matrix columns by the magnitude of the MC-index for the `tau1`, which is the inverse of the smallest eigenvalue. Under this ordering, the first entry of the matrix is classified as a "category 5" thickness in plotting. Subsequently, other lines are sequentially put in the categories of

- category 5: 0.5 or above
- category 4: 0.3 - 0.5
- category 3: 0.2 - 0.3
- category 2: 0.1 - 0.2
- category 1: 0.0 - 0.1

Usage

```
## S3 method for class 'mcvis'  
ggplot(x, eig_max = 1L, var_max = ncol(x$MC))
```

Arguments

<code>x</code>	Output of the <code>mcvis</code> function
<code>eig_max</code>	The maximum number of eigenvalues to be displayed on the plot.
<code>var_max</code>	The maximum number of variables (i.e. columns) to be displayed on the plot.

Value

A `ggplot`

Author(s)

Chen Lin, Kevin Wang, Samuel Mueller

Examples

```
library(ggplot2)  
set.seed(1)  
p = 10  
n = 100  
X = matrix(rnorm(n*p, 0, 5), ncol = p)  
X[,1] = X[,2] + rnorm(n, 0, 0.1)  
mcvis_result = mcvis(X)  
ggplot(mcvis_result)  
ggplot(mcvis_result, eig_max = p)
```

Description

Multi-collinearity Visualization

Usage

```
mcvis(  
  X,  
  sampling_method = "bootstrap",  
  standardise_method = "studentise",  
  times = 1000L,  
  k = 10L  
)
```

Arguments

X A matrix of regressors (without intercept terms).

sampling_method The resampling method for the data. Currently supports 'bootstrap' or 'cv' (cross-validation).

standardise_method The standardisation method for the data. Currently supports 'euclidean' (default, centered by mean and divide by Euclidean length) and 'studentise' (centred by mean and divide by standard deviation) and 'none' (no standardisation)

times Number of resampling runs we perform. Default is set to 1000.

k Number of partitions in averaging the MC-index. Default is set to 10.

Value

A list of outputs:

- **t_square**: The t^2 statistics for the regression between the VIFs and the tau's.
- **MC**: The MC-indices
- **col_names**: Column names (export for plotting purposes)

Author(s)

Chen Lin, Kevin Wang, Samuel Mueller

Examples

```

set.seed(1)
p = 10
n = 100
X = matrix(rnorm(n*p), ncol = p)
X[,1] = X[,2] + rnorm(n, 0, 0.1)
mcvis_result = mcvis(X = X)
mcvis_result

```

plot.mcvis

Multi-collinearity Visualization

Description

Multi-collinearity Visualization

Usage

```

## S3 method for class 'mcvis'
plot(x, thres = 2/3, eig.max = 1, var.max = ncol(x$MC), ...)

```

Arguments

x	Output of the mcvis function
thres	A parameter determining below which thickness the plotting lines hide.
eig.max	The maximum number of eigenvalues to be displayed on the plot.
var.max	The maximum number of variables to be displayed on the plot.
...	additional arguments (currently unused)

Author(s)

Chen Lin, Kevin Wang, Samuel Mueller

Examples

```

set.seed(1)
p = 10
n = 100
X = matrix(rnorm(n*p), ncol = p)
X[,1] = X[,2] + rnorm(n, 0, 0.1)
mcvis_result = mcvis(X)
plot(mcvis_result, thres = 1)

```

`shiny_mcvis`*Shiny app for mcvis exploration*

Description

Shiny app for mcvis exploration

Usage

```
shiny_mcvis(mcvis_result, X)
```

Arguments

<code>mcvis_result</code>	Output of the mcvis function.
<code>X</code>	The original X matrix

Author(s)

Chen Lin, Kevin Wang, Samuel Mueller

Examples

```
if(interactive()){  
  set.seed(1)  
  p = 10  
  n = 100  
  X = matrix(rnorm(n*p), ncol = p)  
  mcvis_result = mcvis(X)  
  shiny_mcvis(mcvis_result, X)  
}
```

Index

`ggplot.mcvis`, 2

`mcvis`, 3

`plot.mcvis`, 4

`shiny_mcvis`, 5