Package ‘agreement’

October 27, 2008

Type Package

LazyData yes

LazyLoad yes

Encoding latin1

Depends R (>= 2.7.0), stats, utils, tools

Title agreement: Analyse the agreement between two measurement methods

Version 1.0-1

Date 2008-09-18

Author Fabio Frascati, Bruno Mario Cesana

Maintainer Fabio Frascati <fabio.frascati@studenti.unimi.it>

Description The agreement is a package to investigate agreement between two measurement methods using a simulation approach. This approach is the same as that presented in literature.

License GPL-2

URL http://www.r-project.org/, http://cran.r-project.org/

R topics documented:

`agreement-package` *agreement: Analyse the agreement between two measurement methods*

Description

The agreement is a package to investigate agreement between two measurement methods using a simulation approach. This approach is the same as that presented in literature.

Details

Package:	agreement
Type:	Package
LazyData:	yes
LazyLoad:	yes
Encoding:	latin1
Depends:	R (>= 2.7.0), stats, utils, tools
Version:	1.0-1
Date:	2008-09-18
License:	GPL-2
URL:	http://www.r-project.org/, http://cran.r-project.org/

Index:

```
lin.simulation          Performs a summary of the simulation
```

Further information is available in the following vignettes:

`agreement`	agreement: Analyse the agreement between two measurement methods (source, pdf)

Author(s)

Fabio Frascati, Bruno Mario Cesana

Maintainer: Fabio Frascati <fabio.frascati@studenti.unimi.it>

`lin.simulation` *Performs a summary of the simulation*

Description

In order to analyze the problem we take a bivariate gaussian distribution on *(X,Y)*. It is true under the null hypothesis (no agreement) and the alternative (yes agreement).

Usage

```
lin.simulation(NUM_CAMP = 5000, NUM = 30, matH0, matH1, underH0 = TRUE, ALPHA_CI
```

Arguments

`NUM_CAMP`	number of samples to simulate. Its default value is *5000*
`NUM`	sample size. Its default value is *30*
`matH0`	matrix of parameters under null hypothesis. It has *2* rows and *3* columns. See **Details**.
`matH1`	matrix of parameters under alternative hypothesis . It has exactly the same structure of `matH0`
`underH0`	logical parameter to determine what condition to simulate. Its default value is `TRUE` (simulation under null hypothesis)
`ALPHA_CI`	level of significance. Its default value is *0.05*
`la_CP1`	the threshold used for TDI. Its default value is *0.9*

Details

The first column of `table` object is the theoretical value of the indices (**Th val**) while the second column (**Thr**) represent the threshold used to determine the rejection region. Theoretical values of α and $1-\beta$ are reported in the third column (**Th prob**). The fourth column (**Mean of est**) represents the inverse transformation of the mean estimate of the agreement measure. We expect the first and the fourth columns to be similar in order to consider the estimate robust. The same conclusion is made between the fifth and the sixth columns which represent the standard deviation of the transformation (**Std of est**) and the mean of the standard deviation (**Mean of std**) respectively. In the seventh column (**Prop rej**) it is calculated the proportion between `NUM_CAMP` runs fall in the rejection region. If we simulate under H_0 then we expect that this value is about $\alpha = 0.05$ (type one error probability) while we expect it is about the true value $1-\beta$ (power) if we simulate under H_1.

`matH0` is a matrix of six elements. In the first row we have the variance of *X*, the covariance between *X* and *Y* and the expected value of *X* respectively. In the second row we have the covariance between *X* and *Y*, the variance of *Y* and the expected value of *Y* respectively. `matH1` has exactly the same structure of `matH0`.

Value

`table`	it is a matrix with *6* rows and *7* columns. Each row represents a measure of agreement and each column a summary for the simulation. See **Details**.
`underH0`	see above
`matH0`	see above
`matH1`	see above
`NUM_CAMP`	see above
`NUM`	see above
`alpha`	see above
`rho`	is the value for the correlation coefficient under null hypothesis (if `underH0 = TRUE`) or alternative hypothesis (if `underH0 = FALSE`)

Author(s)

Fabio Frascati ⟨fabio.frascati@studenti.unimi.it⟩

References

D. G. Altman, J. M. Bland (1983): Measurement in Medicine: The Analysis of Method Comparison Studies, The Statistician, **32**, 302–317

L. Lin (1989): A Concordance Correlation Coefficient to Evaluate Reproducibility, Biometrics, **45**, 255–258

L. Lin, A. Heyadat, B. Sinha, M. Yang (2002): Statistical methods in assessing agreement: Models, issues, and tools, JASA, **97**, 257–270

Examples

```
## we define the matrix of parameters under H0
sigma2x0   <- 1 / 1.15
sigma2y0   <- 1.15
covxy0     <- 0.95 * sqrt(1 / 1.15 * 1.15)
```

```
mux0       <- 0
muy0       <- 0.15
matH0      <- matrix(0,nrow = 2,ncol = 3)
matH0[1,1] <- sigma2x0
matH0[1,2] <- covxy0
matH0[1,3] <- mux0
matH0[2,1] <- covxy0
matH0[2,2] <- sigma2y0
matH0[2,3] <- muy0
## we define the matrix of parameters under H1
sigma2x1   <- 1 / 1.1
sigma2y1   <- 1.1
covxy1     <- 0.9662055 * sqrt(1 / 1.1 * 1.1)
mux1       <- 0
muy1       <- 0.1
matH1      <- matrix(0,nrow = 2,ncol = 3)
matH1[1,1] <- sigma2x1
matH1[1,2] <- covxy1
matH1[1,3] <- mux1
matH1[2,1] <- covxy1
matH1[2,2] <- sigma2y1
matH1[2,3] <- muy1
## we run lin.simulation()
lin.simulation(matH0 = matH0,matH1 = matH1,NUM = 30,NUM_CAMP = 5000,underH0 = TRUE)
```

Index