

Package ‘comparison’

February 19, 2015

Version 1.0-4

Date 05/11/2013

Title Multivariate likelihood ratio calculation and evaluation

Author David Lucy <d.lucy@lancaster.ac.uk>

Maintainer David Lucy <d.lucy@lancaster.ac.uk>

Description Functions for calculating and evaluating likelihood ratios from uni/multivariate continuous observations

Imports isotone, methods

License GPL-2

LazyLoad yes

NeedsCompilation no

Repository CRAN

Date/Publication 2013-11-05 15:52:30

R topics documented:

comparison-package	2
calc.ece	3
calibrate.set	4
compcovar-class	5
compitem-class	7
ece-class	8
glass	9
plot-method	10
two.level.comparison.items	11
two.level.components	12
two.level.density.LR	14
two.level.lindley.LR	15
two.level.normal.LR	17

Index	20
--------------	-----------

comparison-package	<i>Calculates and compares likelihood ratios for uni/multivariate comparison problems</i>
--------------------	---

Description

Takes uni/multivariate observations of a physical object, for example trace element measurements, and a similar set of uni/multivariate observations from another object, and calculates a likelihood ratio for the propositions that the first item came from the same source as the second given some populational data.

Details

Package:	comparison
Type:	Package
Version:	1.0
Date:	11-07-2012
License:	GPL2
LazyLoad:	yes

Acknowledgements

In a package of functions such as these which have undergone a long development over a number of years, it is inevitable that a number of people, besides those directly cited, have helped to correct and add to the code. These people are (in alphabetical order): Ivo Alberink (NFI), Anabel Bolck (NFI), Sonja Menges (BKA), Geoff Morrison (UNSW), Tereza Neocleous (Glasgow), Anders Nordgaard (SKL), Brad Patterson (George Mason), Phil Rose (ANU), Agnieszka Rzepecka (Jagiellonian), Marjan Sjerps (NFI) and Hanjing Zhang (Edinburgh).

Author(s)

David Lucy <d.lucy@lancaster.ac.uk> - <http://www.maths.lancs.ac.uk/~lucy>

References

Aitken, C.G.G. & Lucy, D. (2004) Evaluation of trace evidence in the form of multivariate data. *Applied Statistics*: **53**(1); 109-122.

See Also

[compcovar](#)
[compitem](#)
[two.level.comparison.items](#)

```
two.level.components
two.level.density.LR
two.level.normal.LR
two.level.lindley.LR
gpava
ece
calibrate.set
calc.ece
plot
```

Examples

```
ls()
```

calc.ece	<i>Empirical cross-entropy calculation</i>
----------	--

Description

Calculates the empirical cross-entropy for likelihood ratios from a sequence same and different item comparisons

Usage

```
calc.ece(LR.ss, LR.ds, prior=seq(from=0.01, to=0.99, length=99))
```

Arguments

LR.ss	array of likelihood ratios for same source item comparisons
LR.ds	array of likelihood ratios for different source item comparisons
prior	array of ordinates for the prior in ascending order, and between 0 and 1. Default is 99 divisions of 0.01 to 0.99

Value

Returns an S4 object of class ece

Acknowledgements

The function to calculate the values of the likelihood ratio for the calibrated.set draws heavily upon the opt_loglr.m function from Niko Brummer's FoCal package for Matlab.

Note

The empirical cross-entropy for a set of comparisons for items of known origin can be used as a measure of performance of the comparisons. This function takes the likelihood ratios.

Author(s)

David Lucy <d.lucy@lancaster.ac.uk> - <http://www.maths.lancs.ac.uk/~lucy>

References

Ramos, D. & Gonzalez-Rodriguez, J. (2008) Cross-entropy analysis of the information in forensic speaker recognition; IEEE Odyssey.

Zadora, G. & Ramos, D. (2010) Evaluation of glass samples for forensic purposes - an application of likelihood ratio model and information-theoretical approach. *Chemometrics and Intelligent Laboratory: 102*; 63-83.

See Also

[gpava](#)
[ece](#)
[calibrate.set](#)
[calc.ece](#)

Examples

```
#library(comparison)
LR.same <- c(0.5, 2, 4, 6, 8, 10) # the same has 1 LR < 1
LR.different <- c(0.2, 0.4, 0.6, 0.8, 1.1) # the different has 1 LR > 1
ece.1 <- calc.ece(LR.same, LR.different) # simplest invocation
plot(ece.1) # use plot method
```

calibrate.set

Internal function - calibrate the set

Description

Produces an calibrated set of idealised values

Usage

```
calibrate.set(LR.ss, LR.ds, method = "raw")
```

Arguments

LR.ss	an array of likelihood ratios for the comparisons of items known to be from the same items
LR.ds	an array of likelihood ratios for the comparisons of items known to be from different items
method	either "raw" or "laplace" - default "raw"

Details

An internal function not really to be used directly.

Value

LR.cal.ss calibrated lrs for the comparison for same set
LR.cal.ds calibrated lrs for the comparison for different set

Note

Do not use this function directly.

Author(s)

David Lucy <d.lucy@lancaster.ac.uk> - <http://www.maths.lancs.ac.uk/~lucy>

References

Ramos, D. & Gonzalez-Rodriguez, J. (2008) Cross-entropy analysis of the information in forensic speaker recognition; IEEE Odyssey.

See Also

[gpava](#)
[ece](#)
[calibrate.set](#)
[calc.ece](#)

Examples

```
ls()
```

compcovar-class *Class of object compcovar*

Description

A set of estimates for the hierarchical covariance matrices, and means, calculated for a background population needed for the comparison of two items.

Objects from the Class

Objects can be created by calls of the form `new("compcovar", ...)`. The function `two.level.components` is recommended for the creation of these objects.

Slots

v.within: estimate of the within item covariance matrix - object of class "matrix".

v.between: estimate of the between item covariance matrix - object of class "matrix"

n.observations: count of the total number of observations in the sample from the background population - object of class "numeric"

n.items: count of the number of items in the sample from the background population of - object class "numeric"

item.n: an array of the number of replicated observations for each item in the sample - object of class "matrix"

item.means: an array of the means of the replicated observations for each item in the sample - object of class "matrix"

n.vars: count of the number of properties measured - object of class "numeric" ~~~

overall.means: means of all the properties measured for each item in the background population - object of class "numeric"

multivariate: flag indicating whether the observations are multivariate - object of class "logical"

balanced: flag indicating whether the observations are balanced - Object of class "logical"

s.within: sum of squared deviations for within all items - object of class "matrix"

s.between: sum of squared deviations between all items - object of class "matrix"

warn.type: whether any warnings were issued "character"

Methods

two.level.density.LR signature(control = "compitem", recovered = "compitem", background = "compcovar")

Note

ML estimates for variance components - need something a bit more robust

Author(s)

David Lucy <d.lucy@lancaster.ac.uk> - <http://www.maths.lancs.ac.uk/~lucy>

References

Aitken, C.G.G. & Lucy, D. (2004) Evaluation of trace evidence in the form of multivariate data. *Applied Statistics*: **53**(1); 109-122.

See Also

[compcovar](#)
[compitem](#)
[two.level.comparison.items](#)
[two.level.components](#)
[two.level.density.LR](#)

Examples

```
showClass("compcovar")
```

compitem-class	<i>Class of object compitem</i>
----------------	---------------------------------

Description

An object which represents the information required for comparison calculated from either the control, or recovered items.

Objects from the Class

Objects can be created by calls of the form `new("compitem", ...)`. Using `two.level.comparison.items` is recommended.

Slots

`item.means`: a vector of the means for the observations of the item - object of class "numeric"
`n.replicates`: count of the number of replicated observations of the item - object of class "numeric"
`n.vars`: count of the number of properties observed for the item - object of class "numeric"
`multivariate`: flag indicating whether the observations are multivariate - object of class "logical"
`observed`: the raw observations for the item - object of class "matrix"
`warn.type`: whether any warnings were issued "character"

Methods

two.level.density.LR signature(control = "compitem", recovered = "compitem", background = "compcovar"): ...

Note

Best not to create these directly, use `two.level.comparison.items` instead.

Author(s)

David Lucy <d.lucy@lancaster.ac.uk> - <http://www.maths.lancs.ac.uk/~lucy>

References

Aitken, C.G.G. & Lucy, D. (2004) Evaluation of trace evidence in the form of multivariate data. *Applied Statistics*: **53**(1); 109-122.

See Also

[compcovar](#)
[compitem](#)
[two.level.comparison.items](#)
[two.level.components](#)
[two.level.density.LR](#)

Examples

```
showClass("compitem")
```

ece-class

Class of object ece

Description

Objects of class "ece" are empirical cross-entropies.

Objects from the Class

Objects can be created by calls of the form `new("ece", ...)` or, more commonly, via the [calc.ece](#) function which is recommended.

Slots

prior: an arbitrary array of ascending values between zero and one representing each value for which the ece is to be evaluated

ece.null: ece values calculated assuming the observations give precisely no information about the membership of the

ece: ece values calculated from the likelihood ratios derived from the observations which are related to membership

ece.cal: values calculated from the likelihood ratios derived from a calibrated set using the function `calibrate_set`

Methods

plot variation on the generic `plot` function to plot up all three sets of ece values

References

Ramos, D. & Gonzalez-Rodriguez, J. (2008) Cross-entropy analysis of the information in forensic speaker recognition; IEEE Odyssey.

Zadora, G. & Ramos, D. (2010) Evaluation of glass samples for forensic purposes - an application of likelihood ratio model and information-theoretical approach. *Chemometrics and Intelligent Laboratory*: **102**; 63-83.

See Also

[gpava](#)
[ece](#)
[calibrate.set](#)
[calc.ece](#)
[plot](#)

 glass

Data - elemental composition of glass

Description

Greg Zadora's glass composition data for seven elements from 200 glass items.

Usage

```
data(glass)
```

Format

item	factor -	200 levels - which item the measurements came from
fragment	factor -	4 levels - which of the four fragments from each item the observations were made upon
logNaO	numeric -	log of sodium concentration to oxygen concentration
logMgO	numeric -	log of magnesium concentration to oxygen concentration
logAlO	numeric -	log of aluminium concentration to oxygen concentration
logSiO	numeric -	log of silicon concentration to oxygen concentration
logKO	numeric -	log of potassium concentration to oxygen concentration
logCaO	numeric -	log of calcium concentration to oxygen concentration
logFeO	numeric -	log of iron concentration to oxygen concentration

Details

These data are from Greg Zadora at the [Institute of Forensic Research](#) in Cracow, Poland. They are the log of the ratios of each element to oxygen, so logNaO is the log(10) of the Sodium to Oxygen ratio, and logAlO is the log of the Aluminium to Oxygen ratio. The instrumental method was SEM-EDX.

The item indicates the object the glass came from. The levels for each item are unique to that item. The fragment can be considered a sub-item. When collecting these observations Greg took a glass object, say a jam jar, he would then break it, and extract four fragments. Each fragment would be measured three times upon different parts of that fragment. The fragment labels are repeated, so, for example, fragment "f1" from item "s2" has nothing whatsoever to do with fragment "f1" from item "s101".

These data are a list. If some object more matrix like is needed convert using `as.data.frame()`.

For two level models use "item" as the lower level - three level models can use the additional information from the individual fragments.

Source

Grzegorz Zadora [Institute of Forensic Research](#) Cracow, Poland.

References

Aitken, C.G.G. Zadora, G. & Lucy, D. (2007) A Two-Level Model for Evidence Evaluation. *Journal of Forensic Sciences*: **52**(2); 412-419.

Examples

```
data(glass)
```

plot-method

Methods for plot in Package **graphics**

Description

Methods for function plot in package **graphics**

Methods

```
signature(x = "ANY")  
signature(x = "ece")
```

Usage

```
plot(x) # where x is an ece object generated from calc.ece
```

References

Ramos, D. & Gonzalez-Rodriguez, J. (2008) Cross-entropy analysis of the information in forensic speaker recognition; IEEE Odyssey.

Zadora, G. & Ramos, D. (2010) Evaluation of glass samples for forensic purposes - an application of likelihood ratio model and information-theoretical approach. *Chemometrics and Intelligent Laboratory*: **102**; 63-83.

See Also

[gpava](#)
[ece](#)
[calibrate.set](#)
[calc.ece](#)

`two.level.comparison.items`*Creates a compitem object*

Description

This function creates a `compitem` object from a dataframe or matrix of observations from an item to be deemed a control, or a recovered, item.

Usage

```
two.level.comparison.items(dat, data.columns)
```

Arguments

<code>dat</code>	a matrix or dataframe of observed properties from either the control, or recovered, item
<code>data.columns</code>	a vector of integers giving which columns in <code>dat</code> are the observations of the properties

Details

Takes observations from the control, or recovered, item and makes them into an object which can be passed to either: `two.level.density.LR`, or, `two.level.normal.LR`, or, `two.level.lindley.LR`, for likelihood ratio calculation.

Value

Returns a `compitem` object

Note

This function, and class it generates, exist to make sure that all the necessary bits and pieces are there to put into the likelihood ratio calculation function

Author(s)

David Lucy <d.lucy@lancaster.ac.uk> - <http://www.maths.lancs.ac.uk/~lucy>

References

Aitken, C.G.G. & Lucy, D. (2004) Evaluation of trace evidence in the form of multivariate data. *Applied Statistics*: **53**(1); 109-122.

See Also

[compcovar](#)
[compitem](#)
[two.level.comparison.items](#)
[two.level.components](#)
[two.level.density.LR](#)
[two.level.normal.LR](#)
[two.level.lindley.LR](#)

Examples

```
# load this library
library(comparison)

# load Greg Zadora's glass data
data(glass)

# make it into a data frame
dat <- as.data.frame(glass)

# calculate a compcovar object based upon dat
# using K, Ca and Fe - warning - could take time
# on slower machines
Z <- two.level.components(dat, c(7,8,9), 1)

# calculate a compitem object representing the control item
control <- two.level.comparison.items(dat[37:42,], c(7,8,9))

# calculate a compitem object representing the recovered item
# known to be from the same item (item 4)
recovered.1 <- two.level.comparison.items(dat[43:48,], c(7,8,9))

# calculate a compitem object representing the recovered item
# known to be from a different item (item 6)
recovered.2 <- two.level.comparison.items(dat[67:72,], c(7,8,9))
```

two.level.components *Creates a compcovar object*

Description

Takes a large sample from the background population and calculates the within and between covariance matrices, a vector of means, a vector of the counts of replicates for each item from the sample, and other bits needed to make up a [compcovar](#) object

Usage

```
two.level.components(dat, data.columns, item.column)
```

Arguments

<code>dat</code>	a matrix, or <code>data.frame</code> , of observations, with cases in rows, and properties as columns.
<code>data.columns</code>	an array indicating which columns are the properties.
<code>item.column</code>	an integer indicating which column gives the item.

Details

Uses ML estimation at the moment - hopefully give some alternative ways of estimating the covariance matrices in the future.

Value

Returns a `compcovar` object

Note

Can be used directly for variance component estimation.

Author(s)

David Lucy <d.lucy@lancaster.ac.uk> - <http://www.maths.lancs.ac.uk/~lucy>

References

Aitken, C.G.G. & Lucy, D. (2004) Evaluation of trace evidence in the form of multivariate data. *Applied Statistics*: **53**(1); 109-122.

See Also

[compcovar](#)
[compitem](#)
[two.level.comparison.items](#)
[two.level.components](#)
[two.level.density.LR](#)

Examples

```
# load this library
library(comparison)

# load Greg Zadora's glass data
data(glass)

# make it into a data frame
dat <- as.data.frame(glass)

# calculate a compcovar object based upon dat
# using K, Ca and Fe - warning - could take time
# on slower machines
```

```
Z <- two.level.components(dat, c(7,8,9), 1)
```

two.level.density.LR *Likelihood ratio calculation - kernel density*

Description

Takes a `compitem` object which represents some control item, and a `compitem` object which represents a recovered item, then uses information from a `compcovar` object, which represents the information from the population, to calculate a likelihood ratio as a measure of the evidence given by the observations for the same/different source propositions.

Usage

```
two.level.density.LR(control, recovered, background)
```

Arguments

control	a <code>compitem</code> object with the control item information
recovered	a <code>compitem</code> object with the recovered item information
background	a <code>compcovar</code> object with the populational information

Details

Does the likelihood ratio calculations for a two-level model without assuming that the between items distribution is uni/multivariate normal.

Value

Returns an estimate of the likelihood ratio

Note

Do not even think about using this function without the proper `compcovar` and `compitem` objects - it will not work.

Author(s)

David Lucy <d.lucy@lancaster.ac.uk> - <http://www.maths.lancs.ac.uk/~lucy>

References

Aitken, C.G.G. & Lucy, D. (2004) Evaluation of trace evidence in the form of multivariate data. *Applied Statistics*: **53**(1); 109-122.

See Also

[compcovar](#)
[compitem](#)
[two.level.comparison.items](#)
[two.level.components](#)
[two.level.normal.LR](#)

Examples

```
# load this library
library(comparison)

# load Greg Zadora's glass data
data(glass)

# make it into a data frame
dat <- as.data.frame(glass)

# calculate a compcovar object based upon dat
# using K, Ca and Fe - warning - could take time
# on slower machines
Z <- two.level.components(dat, c(7,8,9), 1)

# calculate a compitem object representing the control item
control <- two.level.comparison.items(dat[1:6,], c(7,8,9))

# calculate a compitem object representing the recovered item
# known to be from the same item (item 1)
recovered.1 <- two.level.comparison.items(dat[7:12,], c(7,8,9))

# calculate a compitem object representing the recovered item
# known to be from a different item (item 2)
recovered.2 <- two.level.comparison.items(dat[19:24,], c(7,8,9))

# calculate the likelihood ratio for a known
# same source comparison - should be 20.59322
lr.1 <- two.level.density.LR(control, recovered.1, Z)

# calculate the likelihood ratio for a known
# different source comparison - should be 0.02901532
lr.2 <- two.level.density.LR(control, recovered.2, Z)
```

two.level.lindley.LR *Likelihood ratio calculation - normal - Lindley's approach*

Description

Takes a [compitem](#) object which represents some control item, and a [compitem](#) object which represents a recovered item, then uses information from a [compcovar](#) object, which represents the

information from the population, to calculate a likelihood ratio as a measure of the evidence given by the observations for the same/different source propositions.

Usage

```
two.level.lindley.LR(control, recovered, background)
```

Arguments

control	a <code>compitem</code> object with the control item information
recovered	a <code>compitem</code> object with the recovered item information
background	a <code>compcovar</code> object with the populational information

Details

Does the likelihood ratio calculations for a two-level model assuming that the between item distribution is univariate normal. This function is taken from the approach devised by Denis Lindley in his 1977 paper (details below) and represents the progenitor of all the functions in this package.

Value

Returns an estimate of the likelihood ratio

Note

Do not even think about using this function without the proper `compcovar` and `compitem` objects - it will not work.

Author(s)

David Lucy <d.lucy@lancaster.ac.uk> - <http://www.maths.lancs.ac.uk/~lucy>

References

Lindley, D. (1977) A problem in forensic Science. *Biometrika*: **64**; 207-213.

See Also

`compcovar`
`compitem`
`two.level.comparison.items`
`two.level.components`
`two.level.density.LR`
`two.level.normal.LR`

Examples

```
# load this library
library(comparison)

# load Greg Zadora's glass data
data(glass)

# make it into a data frame
dat <- as.data.frame(glass)

# calculate a compcovar object based upon dat
# using K
Z <- two.level.components(dat, 7, 1)

# calculate a compitem object representing the control item
control <- two.level.comparison.items(dat[1:6,], 7)

# calculate a compitem object representing the recovered item
# known to be from the same item (item 1)
recovered.1 <- two.level.comparison.items(dat[7:12,], 7)

# calculate a compitem object representing the recovered item
# known to be from a different item (item 2)
recovered.2 <- two.level.comparison.items(dat[19:24,], 7)

# calculate the likelihood ratio for a known
# same source comparison - should be 6.323941
lr.1 <- two.level.lindley.LR(control, recovered.1, Z)

# calculate the likelihood ratio for a known
# different source comparison - should be 0.004422907
lr.2 <- two.level.lindley.LR(control, recovered.2, Z)
```

two.level.normal.LR *Likelihood ratio calculation - normal*

Description

Takes a `compitem` object which represents some control item, and a `compitem` object which represents a recovered item, then uses information from a `compcovar` object, which represents the information from the population, to calculate a likelihood ratio as a measure of the evidence given by the observations for the same/different source propositions.

Usage

```
two.level.normal.LR(control, recovered, background)
```

Arguments

control a `compitem` object with the control item information
recovered a `compitem` object with the recovered item information
background a `compcovar` object with the populational information

Details

Does the likelihood ratio calculations for a two-level model assuming that the between item distribution is uni/multivariate normal.

Value

Returns an estimate of the likelihood ratio

Note

Do not even think about using this function without the proper `compcovar` and `compitem` objects - it will not work.

Author(s)

Agnieszka Martyna <ag.rzepecka@gmail.com> and David Lucy <d.lucy@lancaster.ac.uk> - <http://www.maths.lancs.ac.uk/~lucy>.

References

Aitken, C.G.G. & Lucy, D. (2004) Evaluation of trace evidence in the form of multivariate data. *Applied Statistics*: **53**(1); 109-122.

See Also

`compcovar`
`compitem`
`two.level.comparison.items`
`two.level.components`
`two.level.density.LR`

Examples

```
# load this library
library(comparison)

# load Greg Zadora's glass data
data(glass)

# make it into a data frame
dat <- as.data.frame(glass)

# calculate a compcovar object based upon dat
# using K, Ca and Fe - warning - could take time
```

```
# on slower machines
Z <- two.level.components(dat, c(7,8,9), 1)

# calculate a compitem object representing the control item
control <- two.level.comparison.items(dat[1:6,], c(7,8,9))

# calculate a compitem object representing the recovered item
# known to be from the same item (item 1)
recovered.1 <- two.level.comparison.items(dat[7:12,], c(7,8,9))

# calculate a compitem object representing the recovered item
# known to be from a different item (item 2)
recovered.2 <- two.level.comparison.items(dat[19:24,], c(7,8,9))

# calculate the likelihood ratio for a known
# same source comparison - should be 51.16539
lr.1 <- two.level.normal.LR(control, recovered.1, Z)

# calculate the likelihood ratio for a known
# different source comparison - should be 0.02901532
lr.2 <- two.level.normal.LR(control, recovered.2, Z)
```

Index

- *Topic **classes**
 - compcovar-class, 5
 - compitem-class, 7
 - ece-class, 8
 - *Topic **datasets**
 - glass, 9
 - *Topic **methods**
 - plot-method, 10
 - *Topic **models**
 - calc.ece, 3
 - calibrate.set, 4
 - *Topic **multivariate**
 - comparison-package, 2
 - compcovar-class, 5
 - compitem-class, 7
 - two.level.comparison.items, 11
 - two.level.components, 12
 - two.level.density.LR, 14
 - two.level.normal.LR, 17
 - *Topic **package**
 - comparison-package, 2
 - *Topic **univariate**
 - two.level.lindley.LR, 15
- calc.ece, 3, 3, 4, 5, 8–10
- calibrate.set, 3, 4, 4, 5, 9, 10
- comparison-package, 2
- compcovar, 2, 6, 8, 12–18
- compcovar-class, 5
- compitem, 2, 6, 8, 11–18
- compitem-class, 7
- ece, 3–5, 9, 10
- ece-class, 8
- glass, 9
- gpava, 3–5, 9, 10
- plot, 3, 9
- plot, ANY-method (plot-method), 10
- plot, ece-method (plot-method), 10
- plot-method, 10
- two.level.comparison.items, 2, 6–8, 11, 12, 13, 15, 16, 18
- two.level.components, 3, 5, 6, 8, 12, 12, 13, 15, 16, 18
- two.level.density.LR, 3, 6, 8, 11–13, 14, 16, 18
- two.level.lindley.LR, 3, 11, 12, 15
- two.level.normal.LR, 3, 11, 12, 15, 16, 17