Package ‘clust.bin.pair’

February 15, 2018

Title Statistical Methods for Analyzing Clustered Matched Pair Data

Version 0.1.2

Description Tests, utilities, and case studies for analyzing significance in clustered binary matched-pair data. The central function clust.bin.pair uses one of several tests to calculate a Chi-square statistic. Implemented are the tests Eliasziw (1991) <doi:10.1002/sim.4780101211>, Obuchowski (1998) <doi:10.1002/(SICI)1097-0258(19980715)17:13%3C1495::AID-SIM863%3E3.0.CO;2-I>, Durkalski (2003) <doi:10.1002/sim.1438>, and Yang (2010) <doi:10.1002/bimj.201000035> with McNemar (1947) <doi:10.1007/BF02295996> included for comparison. The utility functions nested.to.contingency and paired.to.contingency convert data between various useful formats. Thyroids and psychiatry are the canonical datasets from Obuchowski and Petryshen (1989) <doi:10.1016/0165-1781(89)90196-0> respectively.

Depends R (>= 3.2.4)

License MIT + file LICENSE

Encoding UTF-8

LazyData true

URL `https://github.com/dgopstein/clust.bin.pair`

BugReports `https://github.com/dgopstein/clust.bin.pair/issues`

Imports

Suggests testthat

RoxygenNote 5.0.1

NeedsCompilation no

Author Dan Gopstein [aut, cre]

Maintainer Dan Gopstein <dan@gopstein.com>

Repository CRAN

Date/Publication 2018-02-15 17:44:57 UTC

R topics documented:

`clust.bin.pair` *Statistical test for clustered binary matched pair data*

Description

A single interface for several adjustments to the mcnemar test for marginal homogeneity that correct for clustered data.

Usage

```
clust.bin.pair(ak, bk, ck, dk, method = "yang")
```

Arguments

`ak`	vector containing counts per group of Success/Success results.
`bk`	vector containing counts per group of Success/Failure results.
`ck`	vector containing counts per group of Failure/Success results.
`dk`	vector containing counts per group of Failure/Failure results.
`method`	a character string specifying the method to calculate the statistic. Must be one of "yang" (default), "durkalski", "obuchowski", "eliasziw". A value of "mcnemar" can also be supplied for comparison.

Value

A list with class "htest" containing the following components:

`statistic`	the value of the test statistic.
`p.value`	the p-value for the test.
`method`	the type of test applied.
`data.name`	a character string giving the names of the data.

References

McNemar, Q. (1947). *Note on the sampling error of the difference between correlated proportions or percentages*. Psychometrika, 12(2), 153-157.

Eliasziw, M., & Donner, A. (1991). *Application of the McNemar test to non-independent matched pair data*. Statistics in medicine, 10(12), 1981-1991.

Obuchowski, N. A. (1998). *On the comparison of correlated proportions for clustered data*. Statistics in medicine, 17(13), 1495-1507.

Durkalski, V. L., Palesch, Y. Y., Lipsitz, S. R., & Rust, P. F. (2003). *Analysis of clustered matched-pair data*. Statistics in medicine, 22(15), 2417-2428.

Yang, Z., Sun, X., & Hardin, J. W. (2010). *A note on the tests for clustered matched-pair binary data*. Biometrical journal, 52(5), 638-652.

Examples

```
with(psychiatry, clust.bin.pair(ah, bh, ch, dh, method="eliasziw"))

tc <- nested.to.contingency(thyroids$x.pet, thyroids$x.spect)
clust.bin.pair(tc$ak, tc$bk, tc$ck, tc$dk, method="obuchowski")

oc <- with(obfuscation, paired.to.contingency(group = list(subject, atom),
                                              t1 = control, t2 = treatment))
clust.bin.pair(oc$ak, oc$bk, oc$ck, oc$dk, method="durkalski")
```

`nested.to.contingency` *Convert between nested results and the canonical contingency tables*

Description

Sum all concordant and discordant pairs from each nested group into a contingency table.

Usage

```
nested.to.contingency(t1, t2)
```

Arguments

`t1`	lists of pre-treatment measures
`t2`	lists of post-treatment measures

Value

Contingency tables represented in the rows of a matrix

Examples

```
nested.to.contingency(thyroids$x.pet, thyroids$x.spect)
```

---

obfuscation — *Obfuscated C code misinterpretation data*

Description

Data from Gopstein et. al.'s experiment on the misinterpretation of C code. Subjects were asked to hand evaluate pairs of functionally equivalent code. Half of the questions were intentionally obfuscated to elicit confusion.

Usage

```
data(obfuscation)
```

Format

A data frame with 57 rows and 4 variables:

subject the ID of the study participant

atom the type of obfuscation being evaluated

control whether the subject answered the un-obfuscated question correctly

treatment whether the subject answered the obfuscated question correctly

Source

Atoms of Confusion

Examples

```
data(obfuscation)

oc <- paired.to.contingency(group = obfuscation[,c("subject", "atom")],
                            t1    = obfuscation$control,
                            t2    = obfuscation$treatment)

clust.bin.pair(oc$ak, oc$bk, oc$ck, oc$dk, method="durkalski")
```

paired.to.contingency *Convert between paired results and the canonical contingency tables*

Description

Group results by common clustering then tally the concordant and discordant pairs.

Usage

```
paired.to.contingency(group, t1, t2)
```

Arguments

`group`	List of grouping values
`t1`	pre-treatment measures
`t2`	post-treatment measures

Value

Contingency tables represented in the rows of a matrix

Examples

```
paired.to.contingency(list(obfuscation$subject, obfuscation$atom),
                      obfuscation$control, obfuscation$treatment)
```

psychiatry *Psychiatrist and patient disagreement data*

Description

Psychiatrists and their patients were surveyed in pairs regarding patient concerns and treatment. Each psychiatrist was asked whether each question item was relevant to their patient and each of their patients were asked the same. The data can be evaluated to answer the question of whether there was patient/doctor agreement in each item. The sample was 29 psychiatrists, each with 1-8 patients, for a total of N = 135 matched pairs.

Usage

```
data(psychiatry)
```

Format

A data frame with 29 rows and 7 variables:

psychiatrist the ID of the psychiatrist

Nh the number of the psychiatrist's patients participating in the experiment

ah both participants answered 1

bh patient answered 1, psychiatrist answered 0

ch patient answered 0, psychiatrist answered 1

dh both participants answered 0

Wh Normalized difference: (bh - ch) / Nh

Source

Donner, A., & Petryshen, P. (1989). *The statistical analysis of matched data in psychiatric research*. Psychiatry research, 28(1), 41-46.

References

Eliasziw, M., & Donner, A. (1991). *Application of the McNemar test to non-independent matched pair data*. Statistics in medicine, 10(12), 1981-1991.

Examples

```
data(psychiatry)

psychiatry$Wh == round((psychiatry$bh - psychiatry$ch) / psychiatry$Nh, 2)

clust.bin.pair(psychiatry$ah, psychiatry$bh, psychiatry$ch, psychiatry$dh, method="eliasziw")
```

`thyroids` *PET and SPECT data for diagnosing hyperparathyroidism*

Description

Following surgery which confirmed the absence of hyperparathyroidism two diagnostic tests, PET and SPECT, were performed. Their measures of true negatives and false positives are reported. Data reported in Obuchowki 1998.

Usage

```
data(thyroids)
```

Format

A data frame with 21 rows and 6 variables:

patient ID of the patient

n.glands number of glands tested from the patient

n.pet number of true negatives from the PET test

x.pet individual results per gland from the PET test

n.spect number of true negatives from the SPECT test

x.spect individual results per gland from the SPECT test

Source

Obuchowski, N. A. (1998). *On the comparison of correlated proportions for clustered data.* Statistics in medicine, 17(13), 1495-1507.

Examples

```
data(thyroids)

thyroids$n.glands == sapply(thyroids$x.pet, length)
thyroids$n.glands == sapply(thyroids$x.spect, length)

thyroids$n.pet   == sapply(thyroids$x.pet,   function(x) length(which(x == 1)))
thyroids$n.spect == sapply(thyroids$x.spect, function(x) length(which(x == 1)))

tc <- nested.to.contingency(thyroids$x.pet, thyroids$x.spect)

clust.bin.pair(tc[,'ak'], tc[,'bk'], tc[,'ck'], tc[,'dk'], method="obuchowski")

do.call(clust.bin.pair, data.frame(tc))
```

Index