

Package: magree (via r-universe)

August 20, 2024

Type Package

Title Implements the O'Connell-Dobson-Schouten Estimators of Agreement for Multiple Observers

Version 1.2

Date 2024-01-22

Depends graphics

Author Mark Clements, Dianne O'Connell

Maintainer Mark Clements <mark.clements@ki.se>

Description Implements an interface to the legacy Fortran code from O'Connell and Dobson (1984) <[DOI:10.2307/2531148](https://doi.org/10.2307/2531148)>. Implements Fortran 77 code for the methods developed by Schouten (1982) <[DOI:10.1111/j.1467-9574.1982.tb00774.x](https://doi.org/10.1111/j.1467-9574.1982.tb00774.x)>. Includes estimates of average agreement for each observer and average agreement for each subject.

License GPL-3 | GPL-2

LazyData yes

NeedsCompilation yes

Date/Publication 2024-01-22 19:00:03 UTC

Repository <https://mclements.r-universe.dev>

RemoteUrl <https://github.com/cran/magree>

RemoteRef HEAD

RemoteSha 3f1ba26f8f2600e2a306417c8e84a2cf59569ef2

Contents

landis	2
magree	2
oconnell	4
plot.magree	6
print.magree	7
print.summary.magree	8

schouten	9
summary.magree	11

Index	12
--------------	-----------

landis	<i>Landis and Koch dataset.</i>
--------	---------------------------------

Description

Canonical dataset for agreement for multiple observers described in Landis and Koch (Biometrics 1977; 33: 363-374).

Usage

```
data("landis")
```

Format

The format is: int [1:118, 1:7] 4 1 3 4 3 2 1 3 2 1 ... - attr(*, "dimnames")=List of 2 ..\$: chr [1:118] "1" "2" "3" "4"\$: chr [1:7] "A" "B" "C" "D" ...

Source

Landis and Koch (Biometrics 1977; 33: 363-374)

Examples

```
data(landis)
## maybe str(landis) ; plot(landis) ...
```

magree	<i>O'Connell-Dobson-Schouten estimators for multiobserver agreement.</i>
--------	--

Description

Use the O'Connell-Dobson-Schouten estimators of agreement for nominal or ordinal data.

Usage

```
magree(X, weights=c("unweighted", "linear", "quadratic"), score = NULL)
```

Arguments

X	A matrix or data-frame with observations/subjects as rows and raters as columns.
weights	<p>"unweighted" For nominal categories - only perfect agreement is counted.</p> <p>"linear" For ordinal categories where disagreement is proportional to the distance between the categories. This is analogous to the agreement weights $w_{i,j} = 1 - score[i] - score[j] / (max(score) - min(score))$.</p> <p>"quadratic" For ordinal categories where disagreement is proportional to the square of the distance between the categories. This is analogous to the agreement weights $w_{i,j} = 1 - (score[i] - score[j])^2 / (max(score) - min(score))^2$.</p>
score	The scores that are to be assigned to the categories. Currently, this defaults to a sorted list of the unique values.

Details

The Fortran code from Professor Dianne O'Connell was adapted for R.

The output object is very similar to the Fortan code. Not all of the variance terms are currently used in the print and summary methods.

Value

oconnell	object from the oconnell function
schouten	object from the schouten function
call	As per <code>sys.call()</code> , to allow for using update

See Also

[oconnell](#), [schouten](#).

Examples

```
## Table 1 (O'Connell and Dobson, 1984)
summary(fit <- magree(landis, weights="unweighted"))
update(fit, weights="linear")
update(fit, weights="quadratic")

## Table 5, O'Connell and Dobson (1984)
magree(landis==1)
magree(landis==2)
magree(landis==3)
magree(landis==4)
magree(landis==5)

## Plot of the marginal distributions
plot(fit)

## Plot of the average agreement by observer
plot(fit, type="kappa by observer")
```

 oconnell

O'Connell-Dobson estimators for multiobserver agreement.

Description

Use the O'Connell-Dobson estimator of agreement for nominal or ordinal data. This includes a range of statistics on agreement for assuming either distinct or homogeneous items.

Usage

```
oconnell(X, weights=c("unweighted", "linear", "quadratic"), i=NULL, score = NULL)
```

Arguments

X	A matrix or data-frame with observations/subjects as rows and observers as columns.
weights	<p>"unweighted" For nominal categories - only perfect agreement is counted.</p> <p>"linear" For ordinal categories where disagreement is proportional to the distance between the categories. This is analogous to the agreement weights $w_{i,j} = 1 - score[i] - score[j] / (max(score) - min(score))$.</p> <p>"quadratic" For ordinal categories where disagreement is proportional to the square of the distance between the categories. This is analogous to the agreement weights $w_{i,j} = 1 - (score[i] - score[j])^2 / (max(score) - min(score))^2$.</p>
i	<ol style="list-style-type: none"> For nominal categories - only perfect agreement is counted. For ordinal categories where disagreement is proportional to the distance between the categories. This is analogous to the agreement weights $w_{i,j} = 1 - score[i] - score[j] / (max(score) - min(score))$. For ordinal categories where disagreement is proportional to the square of the distance between the categories. This is analogous to the agreement weights $w_{i,j} = 1 - (score[i] - score[j])^2 / (max(score) - min(score))^2$. <p>This argument takes precedence over weights if it is specified.</p>
score	The scores that are to be assigned to the categories. Currently, this defaults to 1:L, where L is the number of categories.

Details

The Fortran code from Professor Dianne O'Connell was adapted for R.

The output object is very similar to the Fortan code. Not all of the variance terms are currently used in the print, summary and plot methods.

Value

X	As input
i	As input
nrater	Number of observers
nscore	Number of categories
nsubj	Number of subjects
p1[j, k]	Probability of observer j giving score k when observers are distinct
p2[k]	Probability of score k when observers are homogeneous
w1[j, k]	Weighted average of d[] for observer j, score k
w2[k]	Weighted average of d[] for score k when observers are homogeneous
d[j]	Amount of disagreement for subject j
s1[j]	Chance-corrected agreement statistic for subject j when observers are distinct
s2[j]	Chance-corrected agreement statistic for subject j when observers are homogeneous; $s[j]=1-d[j]/expdel$.
delta[j, k]	j<k: amount of disagreement expected by chance for observers j and k; j>k amount of disagreement expected by chance for observers j and k when observers are homogeneous
expd1	Amount of disagreement expected by chance in null case when observers are distinct
expd2	Amount of disagreement expected by chance when observers are homogeneous
dbar	Average value of d[] over all subjects
sav1	Chance-corrected agreement statistic over all subjects when observers are distinct
sav2	Chance-corrected agreement statistic over all subjects when observers are homogeneous
var0s1	Null variance of S when observers are distinct
var0s2	Null variance of S when observers are homogeneous
vars1	Unconstrained variance of S when observers are distinct
vars2	Unconstrained variance of S when observers are homogeneous
v0sav1	Null variance of Sav when observers are distinct
v0sav2	Null variance of Sav when observers are homogeneous
vsav1	Unconstrained variance of Sav when observers are distinct
vsav2	Unconstrained variance of Sav when observers are homogeneous
p0sav1	Probability of overall agreement due to chance when observers are distinct
p0sav2	Probability of overall agreement due to chance when observers are homogeneous
resp[i, j]	Response for observer i on subject j; transpose of X (BEWARE)
score(i)	Score associated with i'th category
call	As per <code>sys.call()</code> , to allow for using update

See Also

[magree, schouten.](#)

Examples

```
## Table 1 (O'Connell and Dobson, 1984)
summary(fit <- oconnell(landis, weights="unweighted"))
update(fit, weights="linear")
update(fit, weights="quadratic")

## Table 3 (O'Connell and Dobson, 1984)
slideTypeGroups <-
  list(c(2,3,5,26,31,34,42,58,59,67,70,81,103,120),
       c(7,10:13,17,23,30,41,51,55,56,60,65,71,73,76,86,87,105,111,116,119,124),
       c(4,6,24,25,27,29,39,48,68,77,79,94,101,102,117),
       c(9,32,36,44,52,62,84,95),
       c(35,53,69,72),
       c(8,15,18,19,47,64,82,93,98,99,107,110,112,115,121),
       c(1,16,22,49,63,66,78,90,100,113),
       c(28,37,40,61,108,114,118),
       106,
       43,
       83,
       c(54,57,88,91,126),
       c(74,104),
       38,
       46,
       c(89,122),
       c(80,92,96,123),
       85)
data.frame(SlideType=1:18,
          S1=sapply(slideTypeGroups,
                   function(ids) mean(fit$s1[as.character(ids)])),
          S2=sapply(slideTypeGroups,
                   function(ids) mean(fit$s2[as.character(ids)])))

## Table 5, O'Connell and Dobson (1984)
oconnell(landis==1)
oconnell(landis==2)
oconnell(landis==3)
oconnell(landis==4)
oconnell(landis==5)

## Plot of the marginal distributions
plot(fit)
```

Description

plot methods for magree, oconnell and schouten objects

Usage

```
## S3 method for class 'magree'
plot(x, type = c("p1", "kappa by observer"),
     xlab = NULL, ylab = NULL, main = NULL, ...)
## S3 method for class 'oconnell'
plot(x, type = c("p1"), xlab = NULL, ylab = NULL, main = NULL, ...)
## S3 method for class 'schouten'
plot(x, type = c("kappa by observer"), xlab = NULL,
     ylab = NULL,
     main = NULL, xdelta = 0.1, axes = TRUE, ...)
```

Arguments

x	magree, oconnell or schouten object.
type	Type of plot. For "p1", plot the probabilities by observer. For "kappa by observer", plot the kappas for each observer.
xlab	x-axis label
ylab	y-axis label
main	main label
xdelta	For plot.schouten and "kappa by observer", specifies the width of the brackets for the confidence intervals.
axes	Bool for whether to plot the axes.
...	other arguments passed to plot

Examples

```
fit <- schouten(landis)
plot(fit)
fit <- oconnell(landis)
plot(fit,type="p1")
```

print.magree

print methods for magree objects

Description

print methods for magree objects

Usage

```
## S3 method for class 'magree'  
print(x, ...)  
## S3 method for class 'oconnell'  
print(x, ...)  
## S3 method for class 'schouten'  
print(x, ...)
```

Arguments

x	the object to print
...	other arguments

Examples

```
print(magree(landis))
```

print.summary.magree *print method for summary.magree objects*

Description

print method for summary.magree objects

Usage

```
## S3 method for class 'summary.magree'  
print(x, ...)  
## S3 method for class 'summary.oconnell'  
print(x, ...)  
## S3 method for class 'summary.schouten'  
print(x, ...)
```

Arguments

x	summary object
...	other arguments passed to print.oconnell, which is passed to summary (if x is not already a summary object).

Examples

```
summary(magree(landis))
```


schouten

*Schouten estimators for multiobserver agreement.***Description**

Use the Schouten estimator of agreement for nominal or ordinal data. This includes a range of statistics on agreement.

Usage

```
schouten(X, weights=c("unweighted", "linear", "quadratic", "user"), w=NULL,
score=NULL)
```

Arguments

X	A matrix or data-frame with subjects as rows and observers as columns.
weights	<p>"unweighted" For nominal categories - only perfect agreement is counted.</p> <p>"linear" For ordinal categories where disagreement is proportional to the distance between the categories. This is analogous to the agreement weights $w_{i,j} = 1 - i - j / (c - 1)$.</p> <p>"quadratic" For ordinal categories where disagreement is proportional to the square of the distance between the categories. This is analogous to the agreement weights $w_{i,j} = 1 - (i - j)^2 / (c - 1)^2$.</p> <p>"user" An indicator for a user-defined weight matrix. The weights argument will be defined as "user" if the w argument is specified.</p>
w	A user-defined weights matrix. This argument takes precedence over weights and score if it is specified and the weight argument will be defined as "user".
score	A user-defined set of scores for each category. If this is not specified, it is assumed that score=1:L, where L is the number of categories. This is used with the weights argument to define the w matrix.

Details

Fortran code was written by Mark Clements based on the algorithms in Schouten (1982).

The output object is closely related to the Fortan code. Not all of the variance terms are currently used in the print, summary and plot methods.

Value

N	Number of subjects
M	Number of observers
L	Number of categories
data	Re-formatted X
w	Weight matrix

<code>kab</code>	Kappas between each pair of observers
<code>ka</code>	Average kappas for each observer
<code>kappa</code>	Average kappa
<code>pab, pa, p, ma, qab, qa, q, oab, eab, oa, ea, o, e, wa, wab</code>	Working fields
<code>varkab</code>	Variances for <code>kab</code>
<code>varka</code>	Variances for <code>ka</code>
<code>vark</code>	Variance for the kappa
<code>covkka</code>	Covariance term between the overall average kappa and the average kappas for each observer
<code>chi</code>	Chi-squared statistics comparing the overall average kappa and the average kappa for each observer (df=1 under the null hypothesis)
<code>pchi</code>	P-values that the overall average kappa equals the average kappa for each observer
<code>var0kab</code>	Variance for <code>kab</code> under the null hypothesis
<code>var0ka</code>	Variance for <code>ka</code> under the null hypothesis
<code>var0k</code>	Variance for the overall average kappa under the null hypothesis
<code>p0</code>	P-value for <code>kappa=0</code>
<code>p0a</code>	P-values that the average kappa for an observer equals zero (i.e. <code>ka=0</code>)
<code>weights</code>	As input
<code>X</code>	As input
<code>call</code>	As per <code>sys.call()</code> , to allow for using update

See Also

[magree](#), [oconnell](#).

Examples

```
## Weights matrix used by Schouten (1982)
w <- outer(1:5,1:5,function(x,y) ((x<=2 & y<=2) | (x>=3 & y>=3))+0)
fit <- schouten(landis,w=w) # user-defined weights

summary(fit) # Schouten (1982), Tables 2 and 5

## we can fit the same model with oconnell() or magree() using the score argument
magree(landis,score=c(1,1,2,2,2))

## plot of the average kappas by observer
plot(fit, type="kappa by observer")
```

summary.magree	<i>summary method for magree objects</i>
----------------	--

Description

summary method for magree objects

Usage

```
## S3 method for class 'magree'  
summary(object, ...)  
## S3 method for class 'oconnell'  
summary(object, ci.transform = c("logit", "identity"), ci.p = 0.95, ...)  
## S3 method for class 'schouten'  
summary(object, ci.transform = c("logit", "identity"), ci.p = 0.95, ...)
```

Arguments

object	object to summarise
ci.transform	transformation used to calculate the confidence intervals. Either "logit" for a logit transform or "identity" for no transform.
ci.p	p value for the confidence interval.
...	unused additional arguments

Examples

```
summary(magree(landis))
```

Index

* Agreement

magree, 2
oconnell, 4
schouten, 9

* Kappa

magree, 2
oconnell, 4
schouten, 9

* datasets

landis, 2

* design

plot.magree, 6
print.magree, 7
print.summary.magree, 8
summary.magree, 11

landis, 2

magree, 2, 6, 10

oconnell, 3, 4, 10

plot.magree, 6

plot.oconnell (plot.magree), 6

plot.schouten (plot.magree), 6

print.magree, 7

print.oconnell (print.magree), 7

print.schouten (print.magree), 7

print.summary.magree, 8

print.summary.oconnell

(print.summary.magree), 8

print.summary.schouten

(print.summary.magree), 8

schouten, 3, 6, 9

summary.magree, 11

summary.oconnell (summary.magree), 11

summary.schouten (summary.magree), 11