

Package ‘polycor’

April 19, 2009

Title Polychoric and Polyserial Correlations

Date 2009/01/26

Version 0.7-7

Depends R (>= 1.9.0), mvtnorm, sfsmisc

Author John Fox

Description Computes polychoric and polyserial correlations by quick “two-step” methods or ML, optionally with standard errors; tetrachoric and biserial correlations are special cases.

Maintainer John Fox <jfox@mcmaster.ca>

License GPL (>= 2)

Revision 5

Repository CRAN

Date/Publication 2009-01-29 19:36:17

R topics documented:

hetcor	2
polychor	4
polyserial	5
print.polycor	7

Index	8
--------------	----------

 hetcor

Heterogeneous Correlation Matrix

Description

Computes a heterogenous correlation matrix, consisting of Pearson product-moment correlations between numeric variables, polyserial correlations between numeric and ordinal variables, and polychoric correlations between ordinal variables.

Usage

```

hetcor(data, ..., ML = FALSE, std.err = TRUE, bins=4, pd=TRUE)
## S3 method for class 'data.frame':
hetcor(data, ML = FALSE, std.err = TRUE,
  use = c("complete.obs", "pairwise.complete.obs"), bins=4, pd=TRUE, ...)
## Default S3 method:
hetcor(data, ..., ML = FALSE, std.err = TRUE, bins=4, pd=TRUE)
## S3 method for class 'hetcor':
print(x, digits = max(3, getOption("digits") - 3), ...)
## S3 method for class 'hetcor':
as.matrix(x, ...)

```

Arguments

<code>data</code>	a data frame consisting of factors and/or numeric variables, or the first of several variables.
<code>...</code>	variables and/or arguments to be passed down.
<code>ML</code>	if TRUE, compute maximum-likelihood estimates; if FALSE, compute quick two-step estimates.
<code>std.err</code>	if TRUE, compute standard errors.
<code>bins</code>	number of bins to use for continuous variables in testing bivariate normality; the default is 4.
<code>pd</code>	if TRUE and if the correlation matrix is not positive-definite, an attempt will be made to adjust it to a positive-definite matrix, using the nearcor function in the <code>sfsmisc</code> package. Note that default arguments to <code>nearcor</code> are used; for more control call <code>nearcor</code> directly.
<code>use</code>	if "complete.obs", remove observations with any missing data; if "pairwise.complete.obs", compute each correlation using all observations with valid data for that pair of variables.
<code>x</code>	an object of class "hetcor" to be printed, or from which to extract the correlation matrix.
<code>digits</code>	number of significant digits.

Value

Returns an object of class "hetcor" with the following components:

correlations	the correlation matrix.
type	the type of each correlation: "Pearson", "Polychoric", or "Polyserial".
std.errors	the standard errors of the correlations, if requested.
n	the number (or numbers) of observations on which the correlations are based.
tests	p-values for tests of bivariate normality for each pair of variables.
NA.method	the method by which any missing data were handled: "complete.obs" or "pairwise.complete.obs".
ML	TRUE for ML estimates, FALSE for two-step estimates.

Note

Although the function reports standard errors for product-moment correlations, transformations (the most well known is Fisher's z -transformation) are available that make the approach to asymptotic normality much more rapid.

Author(s)

John Fox (jfox@mcmaster.ca)

References

- Drasgow, F. (1986) Polychoric and polyserial correlations. Pp. 68-74 in S. Kotz and N. Johnson, eds., *The Encyclopedia of Statistics, Volume 7*. Wiley.
- Olsson, U. (1979) Maximum likelihood estimation of the polychoric correlation coefficient. *Psychometrika* **44**, 443-460.
- Rodriguez, R.N. (1982) Correlation. Pp. 193-204 in S. Kotz and N. Johnson, eds., *The Encyclopedia of Statistics, Volume 2*. Wiley.
- Ghosh, B.K. (1966) Asymptotic expansion for the moments of the distribution of correlation coefficient. *Biometrika* **53**, 258-262.
- Olkin, I., and Pratt, J.W. (1958) Unbiased estimation of certain correlation coefficients. *Annals of Mathematical Statistics* **29**, 201-211.

See Also

[polychor](#), [polyserial](#), [nearcor](#)

Examples

```
set.seed(12345)
R <- matrix(0, 4, 4)
R[upper.tri(R)] <- runif(6)
diag(R) <- 1
R <- cov2cor(t(R) %**% R)
round(R, 4) # population correlations
```

```

data <- rmvnorm(1000, rep(0, 4), R)
round(cor(data), 4) # sample correlations
x1 <- data[,1]
x2 <- data[,2]
y1 <- cut(data[,3], c(-Inf, .75, Inf))
y2 <- cut(data[,4], c(-Inf, -1, .5, 1.5, Inf))
data <- data.frame(x1, x2, y1, y2)
hetcor(data) # Pearson, polychoric, and polyserial correlations, 2-step est.
hetcor(x1, x2, y1, y2, ML=TRUE) # Pearson, polychoric, polyserial correlations, ML est.

```

polychor

Polychoric Correlation

Description

Computes the polychoric correlation (and its standard error) between two ordinal variables or from their contingency table, under the assumption that the ordinal variables dissect continuous latent variables that are bivariate normal. Either the maximum-likelihood estimator or a (possibly much) quicker “two-step” approximation is available. For the ML estimator, the estimates of the thresholds and the covariance matrix of the estimates are also available.

Usage

```
polychor(x, y, ML = FALSE, control = list(), std.err = FALSE, maxcor=.9999)
```

Arguments

<code>x</code>	a contingency table of counts or an ordered categorical variable; the latter can be numeric, a factor, or an ordered factor, but if a factor, its levels should be in proper order.
<code>y</code>	if <code>x</code> is a variable, a second ordered categorical variable.
<code>ML</code>	if <code>TRUE</code> , compute the maximum-likelihood estimate; if <code>FALSE</code> , the default, compute a quicker “two-step” approximation.
<code>control</code>	optional arguments to be passed to the <code>optim</code> function.
<code>std.err</code>	if <code>TRUE</code> , return the estimated variance of the correlation (for the two-step estimator) or the estimated covariance matrix (for the ML estimator) of the correlation and thresholds; the default is <code>FALSE</code> .
<code>maxcor</code>	maximum absolute correlation (to insure numerical stability).

Value

If `std.err` is `TRUE`, returns an object of class “polychor” with the following components:

<code>type</code>	set to “polychoric”.
<code>rho</code>	the polychoric correlation.
<code>row.cuts</code>	estimated thresholds for the row variable (<code>x</code>), for the ML estimate.

<code>col.cuts</code>	estimated thresholds for the column variable (y), for the ML estimate.
<code>var</code>	the estimated variance of the correlation, or, for the ML estimate, the estimated covariance matrix of the correlation and thresholds.
<code>n</code>	the number of observations on which the correlation is based.
<code>chisq</code>	chi-square test for bivariate normality.
<code>df</code>	degrees of freedom for the test of bivariate normality.
<code>ML</code>	TRUE for the ML estimate, FALSE for the two-step estimate.

Othewise, returns the polychoric correlation.

Author(s)

John Fox (jfox@mcmaster.ca)

References

- Drasgow, F. (1986) Polychoric and polyserial correlations. Pp. 68–74 in S. Kotz and N. Johnson, eds., *The Encyclopedia of Statistics, Volume 7*. Wiley.
- Olsson, U. (1979) Maximum likelihood estimation of the polychoric correlation coefficient. *Psychometrika* **44**, 443-460.

See Also

[hetcor](#), [polyserial](#), [print.polycor](#), [optim](#)

Examples

```
set.seed(12345)
data <- rmvnorm(1000, c(0, 0), matrix(c(1, .5, .5, 1), 2, 2))
x <- data[,1]
y <- data[,2]
cor(x, y) # sample correlation
x <- cut(x, c(-Inf, .75, Inf))
y <- cut(y, c(-Inf, -1, .5, 1.5, Inf))
polychor(x, y) # 2-step estimate
polychor(x, y, ML=TRUE, std.err=TRUE) # ML estimate
```

polyserial

Polyserial Correlation

Description

Computes the polyserial correlation (and its standard error) between a quantitative variable and an ordinal variables, based on the assumption that the joint distribution of the quantitative variable and a latent continuous variable underlying the ordinal variable is bivariate normal. Either the maximum-likelihood estimator or a quicker “two-step” approximation is available. For the ML estimator the estimates of the thresholds and the covariance matrix of the estimates are also available.

Usage

```
polyserial(x, y, ML = FALSE, control = list(), std.err = FALSE, maxcor=.9999, bins=
```

Arguments

<code>x</code>	a numerical variable.
<code>y</code>	an ordered categorical variable; can be numeric, a factor, or an ordered factor, but if a factor, its levels should be in proper order.
<code>ML</code>	if <code>TRUE</code> , compute the maximum-likelihood estimate; if <code>FALSE</code> , the default, compute a quicker “two-step” approximation.
<code>control</code>	optional arguments to be passed to the <code>optim</code> function.
<code>std.err</code>	if <code>TRUE</code> , return the estimated variance of the correlation (for the two-step estimator) or the estimated covariance matrix of the correlation and thresholds (for the ML estimator); the default is <code>FALSE</code> .
<code>maxcor</code>	maximum absolute correlation (to insure numerical stability).
<code>bins</code>	the number of bins into which to dissect <code>x</code> for a test of bivariate normality; the default is 4.

Value

If `std.err` is `TRUE`, returns an object of class "polycor" with the following components:

<code>type</code>	set to "polyserial".
<code>rho</code>	the polyserial correlation.
<code>cuts</code>	estimated thresholds for the ordinal variable (<code>y</code>), for the ML estimator.
<code>var</code>	the estimated variance of the correlation, or, for the ML estimator, the estimated covariance matrix of the correlation and thresholds.
<code>n</code>	the number of observations on which the correlation is based.
<code>chisq</code>	chi-square test for bivariate normality.
<code>df</code>	degrees of freedom for the test of bivariate normality.
<code>ML</code>	<code>TRUE</code> for the ML estimate, <code>FALSE</code> for the two-step estimate.

Othwise, returns the polyserial correlation.

Author(s)

John Fox (jfox@mcmaster.ca)

References

Dragow, F. (1986) Polychoric and polyserial correlations. Pp. 68–74 in S. Kotz and N. Johnson, eds., *The Encyclopedia of Statistics, Volume 7*. Wiley.

See Also

[hetcor](#), [polychor](#), [print.polycor](#), [optim](#)

Examples

```

set.seed(12345)
data <- rmvnorm(1000, c(0, 0), matrix(c(1, .5, .5, 1), 2, 2))
x <- data[,1]
y <- data[,2]
cor(x, y) # sample correlation
y <- cut(y, c(-Inf, -1, .5, 1.5, Inf))
polyserial(x, y) # 2-step estimate
polyserial(x, y, ML=TRUE, std.err=TRUE) # ML estimate

```

```

print.plycor      Print Method for polycor Objects

```

Description

print method for objects of class `polycor`, produced by `polychor` and `polyserial`.

Usage

```

## S3 method for class 'polycor':
print(x, digits = max(3, getOption("digits") - 3), ...)

```

Arguments

<code>x</code>	an object of class <code>polycor</code> , as returned by <code>polychor</code> or <code>polyserial</code> .
<code>digits</code>	number of significant digits to be printed.
<code>...</code>	not used.

Value

Invisibly returns `x`; used for its side effect — i.e., printing.

Author(s)

John Fox <jfox@mcmaster.ca>

See Also

[polychor](#), [polyserial](#)

Examples

```

set.seed(12345)
data <- rmvnorm(1000, c(0, 0), matrix(c(1, .5, .5, 1), 2, 2))
x <- data[,1]
y <- data[,2]
cor(x, y) # sample correlation
x <- cut(x, c(-Inf, .75, Inf))
y <- cut(y, c(-Inf, -1, .5, 1.5, Inf))
polychor(x, y, ML=TRUE, std.err=TRUE) # polychoric correlation, ML estimate

```

Index

*Topic **methods**

`print.polycor`, 7

*Topic **models**

`hetcor`, 1

`polychor`, 4

`polyserial`, 5

*Topic **print**

`print.polycor`, 7

`as.matrix.hetcor` (*hetcor*), 1

`hetcor`, 1, 5, 6

`nearcor`, 2, 3

`optim`, 5, 6

`polychor`, 3, 4, 6, 7

`polyserial`, 3, 5, 5, 7

`print.hetcor` (*hetcor*), 1

`print.polycor`, 5, 6, 7