Package ‘BGmisc’

October 12, 2022

Title Behavior Genetic Modeling Functions

Version 0.1

Date 2020-11-15

Description
Functions for behavior genetic modeling, including model identification, calculating relatedness, and various others (e.g., Hunter, Garrison, et al, 2019 <doi:10.1007/s10519-019-09973-8>).

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1.9000

Imports Matrix, stats

NeedsCompilation no

Author S. Mason Garrison [aut],
Michael D. Hunter [aut],
S. Alexandra Burt [aut],
Jonathan D. Trattner [aut, cre]

Maintainer Jonathan D. Trattner `<code@jdtrat.com>`

Repository CRAN

Date/Publication 2020-12-04 09:10:05 UTC

R topics documented:

`BGmisc_package` *Behavior Genetic Miscellaneous functions in R*

Description

This collection contains functions for behavior genetic modeling. These functions include model identification, calculating relatedness, and various others (e.g. Hunter, Garrison, et al, 2019 <doi:10.1007/s10519-019-09973-8>).

Author(s)

S. Mason Garrison, Michael D. Hunter, S. Alexandra Burt, and Jonathan D. Trattner

---

`comp2vech` *Turn a variance component relatedness matrix into its half-vectorization*

Description

Turn a variance component relatedness matrix into its half-vectorization

Usage

```
comp2vech(x, include.zeros = FALSE)
```

Arguments

`x`	relatedness component matrix
`include.zeros`	logical. Whether to include all-zero rows.

Details

This is a wrapper around the `vech` function for producing the half-vectorization of a matrix. The extension here is to allow for blockwise matrices.

Examples

```
comp2vech(list(matrix(c(1, .5, .5, 1), 2, 2), matrix(1, 2, 2)))
```

`fitComponentModel` *Fit the estimated variance components of a model to covariance data*

Description

Fit the estimated variance components of a model to covariance data

Usage

```
fitComponentModel(covmat, ...)
```

Arguments

`covmat`	the covariance matrix of the raw data, possibly blockwise.
`...`	Comma-separated relatedness component matrices.

Details

Returns a regression (linear model fitted with `lm`). The coefficients of the regression are the estimated variance components.

Examples

```
## Not run:
# install.packages("OpenMX")
data(twinData, package = "OpenMx")
sellVars <- c("ht1", "ht2")
mzData <- subset(twinData, zyg %in% c(1), c(selVars, 'zyg'))
dzData <- subset(twinData, zyg %in% c(3), c(selVars, 'zyg'))

fitComponentModel(
covmat = list(cov(mzData[,selVars], use = "pair"), cov(dzData[,selVars], use = "pair")),
A = list(matrix(1, nrow = 2, ncol = 2), matrix(c(1, 0.5, 0.5, 1), nrow = 2, ncol = 2)),
C = list(matrix(1, nrow = 2, ncol = 2), matrix(1, nrow = 2, ncol = 2)),
E = list(diag(1, nrow = 2), diag(1, nrow = 2))
)

## End(Not run)
```

`identifyComponentModel` *Determine if a variance components model is identified*

Description

Determine if a variance components model is identified

Usage

```
identifyComponentModel(..., silent = FALSE)
```

Arguments

`...`	Comma-separated relatedness component matrices.
`silent`	logical. Whether to print messages about identification.

Details

Returns of list of length 2. The first element is a single logical value: TRUE if the model is identified, FALSE otherwise. The second list element is the vector of non-identified parameters. For instance, a model might have 5 components with 3 of them identified and 2 of them not. The second list element will give the names of the components that are not simultaneously identified.

Examples

```
identifyComponentModel(A=list(matrix(1, 2, 2)), C=list(matrix(1, 2, 2)), E= diag(1, 2))
```

`relatedness` *Estimate Relatedness based on Observed Correlation*

Description

Estimate Relatedness based on Observed Correlation

Usage

```
relatedness(cor_obs, ace_A = 0.9, ace_C = 0, shared_c = 0)
```

Arguments

`cor_obs`	Observed Correlation
`ace_A`	proportion of variance attributable to additive genetic variance
`ace_C`	proportion of variance attributable to shared environmental variance
`shared_c`	proportion of shared environment shared. Typically takes zero or 1.

Value

estimated relatedness Coefficient `est_r`

Examples

```
# Using the ACE framework, we can estimate the relatedness between two
# individuals based on the observed correlation between their additive genetic
# variance, shared environmental variance, and proportion of shared environment.

relatedness(cor_obs = 0.5, ace_A = 0.9, ace_C = 0, shared_c = 0)
```

`related_coef` *Relatedness Coefficient Calculation*

Description

Relatedness Coefficient Calculation based on Wright (1922)

Usage

```
related_coef(generations = 2, path = NULL, full = TRUE)
```

Arguments

`generations`	Specifies the number of generations back of common ancestors the pair share
`path`	A Traditional method to count common ancestry, which is 2 times the number of generations removed from common ancestors
`full`	Full or half kin. Do the kin share both parents at the common ancestor's generation?

Details

$r_{bc} = \sum \left(\frac{1}{2}\right)^{n+n'+1} (1 + f_a)$ where the relatedness coefficient between two people (b & c) is defined in relation to their common ancestors.

Value

Relatedness Coefficient `coef`

Examples

```
# For two full siblings, we would expect a relatedness of 0.5. Using the
# default method to count common ancestry, and looking back one generation
# (i.e. towards the full siblings' parents), we get a relatedness coefficient
# of 0.5:
related_coef(generations = 1, path = NULL, full = TRUE)
# Similarly, for half siblings, we would expect a relatedness coefficient of 0.25:
related_coef(generations = 1, path = NULL, full = FALSE)
```

`vech` *Create the half-vectorization of a matrix*

Description

Create the half-vectorization of a matrix

Usage

```
vech(x)
```

Arguments

`x`	a matrix, the half-vectorization of which is desired

Details

Returns the vector of the lower triangle of a matrix, including the diagonal. The upper triangle is ignored with no checking that the provided matrix is symmetric.

Examples

```
vech(matrix(c(1, 0.5, 0.5, 1), nrow = 2, ncol = 2))
```

Index