Package ‘pcgen’

October 14, 2022

Type  Package
Title  Reconstruction of Causal Networks for Data with Random Genetic Effects
Version  0.2.0
Author  Willem Kruijer, Pariya Behrouzi, Maria Xose Rodriguez-Alvarez
Maintainer  Pariya Behrouzi <pariya.behrouzi@gmail.com>
Depends  R (>= 3.1.0)
Imports  pcalg, graph, Matrix, stats, MASS, utils, Hmisc, methods, lme4, sommer, ggm
Description  Implements the pcgen algorithm, which is a modified version of the standard pc-algorithm, with specific conditional independence tests and modified orientation rules. pcgen extends the approach of Valente et al. (2010) <doi:10.1534/genetics.109.112979> with reconstruction of direct genetic effects.
License  GPL-3
Date  2019-02-18
NeedsCompilation  no
Repository  CRAN
Date/Publication  2019-02-18 14:50:03 UTC

R topics documented:

checkG .................................................. 2
gencovTest ........................................ 3
getResiduals ....................................... 4
pcgen .................................................. 5
pcgenFast ......................................... 7
pcgenTest ......................................... 9
pcRes ............................................... 11
simdata ......................................... 13

Index 14
checkG

Check for consistency in genetic effects

Description

Given output from pcgen or pcgenFast, this function checks whether the estimated graph is consistent with the set of traits having significant genetic variance. The function detects traits that have significant genetic variance but for which there is no partially directed path from G.

Usage

checkG(pcgen.output, suffStat, alpha = 0.01, covariates = NULL)

Arguments

- pcgen.output: A graph with nodes G (genotype) and a number of traits. Typically output from pcgen or pcgenFast.
- suffStat: A data.frame, of which the first column is the factor G (genotype), and subsequent columns contain the traits, and optionally some QTLs. The name of the first column should be G.
- alpha: The significance level used in each conditional independence test. Default is 0.01.
- covariates: A data.frame containing covariates, to be used in each conditional independence test. Cannot contain factors. Should be either NULL (default) or a data.frame with the same number of rows as suffStat. An intercept is already included for each trait in suffStat; covariates should not contain a column of ones.

Value

A logical matrix of dimension \((p + 1) \times (p + 1)\), \(p\) being the number of traits. Most entries are FALSE, except those in the first row and column for which there are conflicts. Entries \([1, j]\) and \([j, 1]\) are TRUE if the \(j\)th trait has significant genetic variance, but there is no partially directed path from G towards that trait. The matrix can then be used in a subsequent run of pcgen or pcgenFast, in the fixedEdges argument. The arguments suffStat, alpha and covariates should stay the same throughout (first run of pcgen, checkG, second run of pcgen).

Author(s)

Willem Kruijer and Pariya Behrouzi. Maintainers: Willem Kruijer <willem.kruijer@wur.nl> and Pariya Behrouzi <pariya.behrouzi@gmail.com>

References

gencovTest

**Estimate genetic covariances between all pairs of traits, and test their significance**

**Description**

For each pair of traits in suffStat, we fit a bivariate mixed model, and perform a likelihood ratio test for the null-hypothesis of zero genetic covariance.

**Usage**

```r
gencovTest(suffStat, max.iter = 200, out.cor = TRUE)
```

**Arguments**

- `suffStat`: A data.frame with \((p + 1)\) columns, of which the first column is the factor \(G\) (genotype), and subsequent \(p\) columns contain traits. It should not contain covariates or QTLs.
- `max.iter`: Maximum number of iterations in the EM-algorithm, used to fit the bivariate mixed model.
- `out.cor`: If `TRUE`, the output will contain estimates of genetic correlations; otherwise covariances. The pvalues are always for genetic covariance.

**Value**

A list with elements `pvalues` and `out.cor`, which are both \(p \times p\) matrices.

**Author(s)**

Willem Kruijer and Pariya Behrouzi. Maintainers: Willem Kruijer <willem.kruijer@wur.nl> and Pariya Behrouzi <pariya.behrouzi@gmail.com>

**References**


**Examples**

```r
data(simdata)
test <- gencovTest(suffStat = simdata, max.iter = 200, out.cor = TRUE)
```
**getResiduals**

**Residuals from the GBLUP**

**Description**

Residuals from the best linear unbiased predictor of the genetic effects (GBLUP), which is computed given REML-estimates of the variance components.

**Usage**

```r
getResiduals(suffStat, covariates = NULL, cov.method = "uni", K = NULL)
```

**Arguments**

- `suffStat`: A data.frame, of which the first column is the factor `G` (genotype), and subsequent columns contain the traits. The name of the first column should be `G`.
- `covariates`: A data.frame containing covariates, that should always be used in each conditional independence test. Should be either `NULL` (default) or a data.frame with the same number of rows as `suffStat`. An intercept is already included for each trait in `suffStat`; covariates should not contain a column of ones.
- `cov.method`: (A string, specifying which method should be used to compute the GBLUP. Options are "us" (unstructured multi-trait model fitted using sommer) and "uni" (based on univariate GBLUPs). Default is "uni").
- `K`: A genetic relatedness matrix. If `NULL` (default), independent genetic effects are assumed.

**Details**

If `cov.method = "uni"`, the GBLUP and the residuals are computed separately for each trait in `suffStat`. The covariance of each trait is then assumed to be

\[
\sigma^2_G ZKZ^T + \sigma^2_E I_n
\]

where `Z` is a binary incidence matrix, assigning plants or plots to genotypes. `Z` is based on the first column in `suffStat`. If there is a single observation per genotype (typically a genotypic mean), `Z` is the identity matrix, and the relatedness matrix `K` should be specified. If there are replicates for at least some of the genotypes, and no `K` is provided, independent genetic effects are assumed (`K` will be the identity matrix). It is also possible to have replicates and specify a non-diagonal `K`. Whenever `K` is specified, sommer (mmer2) will be used; otherwise lmer (lme4). The mmer2 is also used when `cov.method = "us"`, in which case the multivariate GBLUP is computed, for all traits in `suffStat` simultaneously. This is only possible for a limited number of traits.

**Value**

A data-frame with the residuals.
pcgen

Author(s)

Willem Kruijer and Pariya Behrouzi. Maintainers: Willem Kruijer <willem.kruijer@wur.nl> and Pariya Behrouzi <pariya.behrouzi@gmail.com>

References


Examples

data(simdata)
rs <- getResiduals(suffStat= simdata)

Description

Reconstruction of directed networks with random genetic effects, based on phenotypic observations. The pcgen algorithm is a modification of the pc-stable algorithm of Colombo & Maathuis (2014) . It is assumed that there are replicates, and independent genetic effects.

Usage

pcgen(suffStat, covariates = NULL, QTLs = integer(), alpha = 0.01, m.max = Inf, fixedEdges = NULL, fixedGaps = NULL, verbose = FALSE, use.res = FALSE, res.cor = NULL, max.iter = 50, stop.if.significant = TRUE, return.pvalues = FALSE)

Arguments

suffStat A data.frame, of which the first column is the factor G (genotype), and subsequent columns contain the traits, and optionally some QTLs. The name of the first column should be G. Should not contain covariates.
covariates A data.frame containing covariates, that should always be used in each conditional independence test Should be either NULL (default) or a data.frame with the same number of rows as suffStat. An intercept is already included for each trait in suffStat; covariates should not contain a column of ones.
QTLs Column numbers in suffStat that correspond to QTLs.
alpha The significance level used in each conditional independence test. Default is 0.01
m.max Maximum size of the conditioning sets
fixedEdges A logical matrix of dimension \((p + 1) \times (p + 1)\), where \(p\) is the number of traits. The first row and column refer to the node \(G\), and subsequent rows and columns to the traits. As in the pcalg package, the edge \(i - j\) is never considered for removal if the entry \([i, j]\) or \([j, i]\) (or both) are TRUE. In that case, the edge is guaranteed to be present in the resulting graph.

fixedGaps A logical matrix of dimension \((p + 1) \times (p + 1)\), where \(p\) is the number of traits. The first row and column refer to the node \(G\), and subsequent rows and columns to the traits. As in the pcalg package, the edge \(i - j\) is removed before starting the algorithm if the entry \([i, j]\) or \([j, i]\) (or both) are TRUE. In that case, the edge is guaranteed to be absent in the resulting graph.

verbose If TRUE, p-values for the conditional independence tests are printed.

use.res If TRUE, the test for conditional independence of 2 traits given a set of other traits and \(G\) is based on residuals from GBLUP. If FALSE (the default), it is based on bivariate mixed models.

res.cor If use.res = TRUE, res.cor should be the correlation matrix of the residuals from the GBLUP. These can be obtained with the getResiduals function. See the example below.

max.iter Maximum number of iterations in the EM-algorithm, used to fit the bivariate mixed model (when use.res = FALSE).

stop.if.significant If TRUE, the EM-algorithm used in some of the conditional independence tests (when use.res = FALSE) will be stopped whenever the p-value becomes significant, i.e. below alpha. This will speed up calculations, and can be done because (1) the PC algorithm only needs an accept/reject decision (2) In EM the likelihood is nondecreasing. Should be put to FALSE if the precise p-values are of interest.

return.pvalues If TRUE, the maximal p-value for each edge is returned.

Details

The pcgen function is based on the pc function from the pcalg package (Kalisch et al. (2012) and Hauser and Buhlmann (2012)).

Value

If return.pvalues = FALSE, the output is a graph (an object with S3 class "pcgen"). If return.pvalues = TRUE, the output is a list with elements gr (the graph) and pMax (a matrix with the p-values).

Author(s)

Willem Kruijer and Pariya Behrouzi. Maintainers: Willem Kruijer <willem.kruijer@wur.nl> and Pariya Behrouzi <pariya.behrouzi@gmail.com>

References

effects.

### See Also
- `getResiduals`

### Examples
```r
data(simdata)
out <- pcgen(simdata)

data(simdata)
rs <- getResiduals(suffStat = simdata)
pc.fit1 <- pcgen(suffStat = simdata, alpha = 0.01, verbose = TRUE, use.res = TRUE, res.cor = cor(rs))
```

### Description
The `pcgenFast` algorithm starting with a skeleton estimated using the standard pc-algorithm, based on residuals from the GBLUP.

### Usage
```r
pcgenFast(suffStat, alpha = 0.01, m.max = Inf, res.m.max = Inf, verbose = FALSE, covariates = NULL, fixedEdges = NULL, QTLs = integer(), max.iter = 50, stop.if.significant = TRUE, cov.method = 'uni', use.res = FALSE, return.pvalues = FALSE)
```

### Arguments
- **suffStat**: A data.frame, of which the first column is the factor G (genotype), and subsequent columns contain the traits, and optionally some QTLs. The name of the first column should be G.
- **alpha**: The significance level used in each conditional independence test. Default is 0.01.
m.max
Maximum size of the conditioning set, in the pcgen algorithm.

res.m.max
Maximum size of the conditioning set, in the pc-algorithm on the residuals (used for prior screening).

verbose
If TRUE, p-values for the conditional independence tests are printed.

covariates
A data.frame containing covariates, to be used in each conditional independence test. Cannot contain factors. Should be either NULL (default) or a data.frame with the same number of rows as suffStat. An intercept is already included for each trait in suffStat; covariates should not contain a column of ones.

fixedEdges
A logical matrix of dimension \((p + 1) \times (p + 1)\), where \(p\) is the number of traits. The first row and column refer to the node \(G\), and subsequent rows and columns to the traits. As in the pcalg package, the edge \(i - j\) is never considered for removal if the entry \([i,j]\) or \([j,i]\) (or both) are TRUE. In that case, the edge is guaranteed to be present in the resulting graph.

QTLs
Column numbers in suffStat that correspond to QTLs.

max.iter
Maximum number of iterations in the EM-algorithm, used to fit the bivariate mixed model (when use.res = FALSE).

stop.if.significant
If TRUE, the EM-algorithm used in some of the conditional independence tests (when use.res = FALSE) will be stopped whenever the p-value becomes significant, i.e. below alpha. This will speed up calculations, and can be done because (1) the PC algorithm only needs an accept/reject decision (2) In EM the likelihood is nondecreasing. Should be put to FALSE if the precise p-values are of interest.

cov.method
A string, specifying which method should be used to compute the GBLUP. Options are ’us’ (unstructured multi-trait model fitted using sommer) and ’uni’ (based on univariate GBLUPs). Default is ’uni’.

use.res
If FALSE, residuals from GBLUP are only used for screening with the standard pc algorithm. After that, the standard pcgen algorithm is run on the remaining edges; the test for conditional independence of 2 traits given a set of other traits and \(G\) is based on bivariate mixed models. If TRUE, this test is based on the residuals. In this case, no further edges between traits are removed after screening and pcgen will only infer the orientation, and the direct genetic effects.

return.pvalues
If TRUE, the maximal p-value for each edge is returned.

Value
If return.pvalues = FALSE, the output is a graph (an object with S3 class "pcgen"). If return.pvalues = TRUE, the output is a list with elements gr (the graph) and pMax (a matrix with the p-values).

Author(s)
Willem Kruijer and Pariya Behrouzi. Maintainers: Willem Kruijer <willem.kruijer@wur.nl> and Pariya Behrouzi <pariya.behrouzi@gmail.com>
References


See Also

generateResiduals

Examples

data(simdata)
out <- pcgenFast(suffStat = simdata, alpha = 0.01, verbose = FALSE, use.res = TRUE)

pcgenTest

The conditional independence test in pcgen

Description

This performs the conditional independence test used in the pcgen algorithm, assuming there are replicates, and independent genetic effects.

Usage

pcgenTest(x, y, S, suffStat, QTLs = integer(), covariates = NULL, alpha = 0.01, max.iter = 50, stop.if.significant = TRUE, use.res = FALSE, res.cor = NULL)

Arguments

x, y Column numbers in suffStat that should be tested for conditional independence given the variables in S.
S vector of integers defining the conditioning set, where the integers refer to column numbers in suffStat. May be numeric(), i.e. the empty set.
suffStat A data.frame, of which the first column is the factor G(genotype), and subsequent columns contain the traits, and optionally some QTLs. The name of the first column should be G. It should not contain covariates.
QTLs Column numbers in suffStat that correspond to QTLs. These may be partly in S and x and y, but x and y cannot be both QTLs.
covariates A data.frame containing covariates. It should be either NULL (default) or a data.frame with the same number of rows as suffStat. An intercept is already included for each trait in suffStat; covariates should not contain a column of ones.
alpha The significance level used in the test. The test itself of course does not depend
on this, but it is used in the EM-algorithm to speed up calculations. When
stop.if.significant = TRUE, the EM-algorithm is stopped once the p-value
is below the significance level. Default is 0.01.

max.iter Maximum number of iterations in the EM-algorithm, used to fit the bivariate
mixed model (when use.res = FALSE).

stop.if.significant If TRUE, the EM-algorithm used in some of the conditional independence tests
(when use.res = FALSE) will be stopped whenever the p-value becomes sig-
nificant, i.e. below alpha. This will speed up calculations, and can be done
because (1) the PC algorithm only needs an accept/reject decision (2) In EM the
likelihood is nondecreasing. It should be put to FALSE if the precise p-value is
of interest.

use.res If TRUE, the test for conditional independence of 2 traits given a set of other traits
and G is based on residuals from GBLUP. If FALSE (the default), it is based on
bivariate mixed models.

res.cor If use.res = TRUE, res.cor should be the correlation matrix of the residuals from
the GBLUP. These can be obtained with the getResiduals function. See the
example below.

Details

pcgenTest tests for conditional independence between \( x \) and \( y \) given \( S \). It distinguishes 2 situa-
tions: (i) if one of \( x \) and \( y \) (say \( x \)) is the factor G, pcgenTest will test if the genetic variance in \( y \) is
zero, given the traits in \( S \). (ii) if \( x \) and \( y \) are both traits, pcgenTest tests if the residual covariance
between them is zero, given the traits in \( S \) and the factor G. The factor G is automatically included
in the conditioning set \( S \) (\( S \) does not need to contain the integer 1). This test is either based on a bi-
variate mixed model (when use.res=FALSE), or on residuals from GBLUP (use.res=T), obtained
with the getResiduals function. In the latter case, res.cor must be provided.

Value

A p-value

Author(s)

Willem Kruijer and Pariya Behrouzi. Maintainers: Willem Kruijer <willem.kruijer@wur.nl>
and Pariya Behrouzi <pariya.behrouzi@gmail.com>

References

Kruijer, W., Behrouzi, P., Rodriguez-Alvarez, M. X., Wit, E. C., Mahmoudi, S. M., Yandell, B.,
Van Eeuwijk, F., (2018, in preparation), Reconstruction of networks with direct and indirect genetic
effects.

See Also

getResiduals
Examples

data(simdata)
rs <- getResiduals(suffStat= simdata)
pcgenTest(suffStat= simdata, x= 2, y= 3, S= 4)
pcgenTest(suffStat= simdata, x= 2, y= 3, S= c(1,4))
pcgenTest(suffStat= simdata, x= 2, y= 3, S= 4, use.res= TRUE, res.cor= cor(rs))
pcgenTest(suffStat= simdata, x= 2, y= 1, S= 4)

---

The pc algorithm applied to residuals

Description

The standard pc algorithm applied to GBLUP residuals, or to the GBLUP itself.

Usage

pcRes(suffStat, alpha= 0.01, K = NULL, m.max = Inf, verbose = FALSE, covariates = NULL, QTLs = integer(), cov.method = "uni", use.GBLUP = FALSE, return.pvalues = FALSE)

Arguments

suffStat A data.frame, of which the first column is the factor G (genotype), and subsequent columns contain the traits, and optionally some QTLs. The name of the first column should be G.
alpha The significance level used in the test. Default is 0.01.
K A genetic relatedness matrix. If NULL (the default), independent genetic effects are assumed.
m.max Maximum size of the conditioning set, in the pc-algorithm on the residuals.
verbose If TRUE, p-values for the conditional independence tests are printed.
covariates A data.frame containing covariates, that should always be used in each conditional independence test. Should be either NULL (default) or a data.frame with the same number of rows as suffStat. An intercept is already included for each trait in suffStat; covariates should not contain a column of ones.
QTLs Column numbers in suffStat that correspond to QTLs.
cov.method A string, specifying which method should be used to compute the GBLUP. Options are 'us' (unstructured multi-trait model fitted using sommer) and 'uni' (based on univariate GBLUPs).
use.GBLUP Use the GBLUP itself, instead of the residuals
return.pvalues If TRUE, the maximal p-value for each edge is returned.
Details

If `use.GBLUP = FALSE`, GBLUP residuals are used as input for the pc-stable algorithm of Colombo and Maathuis (2014). This closely resembles the residual networks of Valente et al., (2010) and Topner et al., (2017) (who used different ways to predict the genetic effects, and applied other causal inference algorithms to the residuals). When `use.GBLUP = TRUE`, pc-stable is applied to the GBLUP itself, which resembles the genomic networks of Topner et al., (2017). If `cov.method = "uni"`, the GBLUP and the residuals are computed separately for each trait in suffStat. The covariance of each trait is assumed to be

\[
\sigma^2_G Z K Z^T + \sigma^2_E I_n
\]

where \( Z \) is a binary incidence matrix, assigning plants or plots to genotypes. \( Z \) is based on the first column in suffStat. If there is a single observation per genotype (typically a genotypic mean), \( Z \) is the identity matrix, and the relatedness matrix \( K \) should be specified. If there are replicates for at least some of the genotypes, and no \( K \) is provided, independent genetic effects are assumed (\( K \) will be the identity matrix). It is also possible to have replicates and specify a non-diagonal \( K \). Whenever \( K \) is specified, sommer (mmer2) will be used; otherwise lmer (lme4). mmer2 is also used when `cov.method = "us"`, in which case the multivariate GBLUP is computed, for all traits in suffStat simultaneously. This is only possible for a limited number of traits.

Value

If `return.pvalues = FALSE`, the output is a graph (an object with S3 class "pcgen"). If `return.pvalues = TRUE`, the output is a list with elements `gr` (the graph) and `pMax` (a matrix with the p-values).

Author(s)

Willem Kruijer and Pariya Behrouzi. Maintainers: Willem Kruijer <willem.kruijer@wur.nl> and Pariya Behrouzi <pariya.behrouzi@gmail.com>

References


Examples

data(simdata)
out <- pcRes(suffStat = simdata, alpha = 0.01, verbose= FALSE)
Description

Simulated data, for two replicates of genotypes g1,...,g200. Three traits were simulated (Y1, Y2 and Y3), using a structural equation model defined by Y1 -> Y2 -> Y3, and direct genetic effects on Y1 and Y3.

Usage

data(simdata)

Format

A data frame of dimension 4 x 400. The first column is the factor G (genotype); the subsequent columns contain \( y_1 \), \( y_2 \) and \( y_3 \).

Examples

data(simdata)
out <- pcgen(simdata)
out2 <- pcRes(suffStat = simdata, alpha = 0.01, verbose= FALSE)
Index

* datasets
  simdata, 13

checkG, 2

gencovTest, 3
getResiduals, 4, 7, 9, 10

pcgen, 5
pcgenFast, 7
pcgenTest, 9
pcRes, 11

simdata, 13