# Package 'hscovar'

August 6, 2020

**Type** Package

**Title** Calculation of Covariance Between Markers for Half-Sib Families

**Version** 0.4.0

**Date** 2020-08-05

**Description** The theoretical covariance between pairs of markers is calculated from either paternal haplotypes and maternal linkage disequilibrium (LD) or vise versa. A genetic map is required. Grouping of markers is based on the correlation matrix and a representative marker is suggested for each group. Employing the correlation matrix, optimal sample size can be derived for association studies based on a SNP-BLUP approach.
The implementation relies on paternal half-sib families and biallelic markers. If maternal half-sib families are used, the roles of sire/dam are swapped. Multiple families can be considered.
Wittenburg, Bonk, Doschoris, Reyer (2019) ``Design of Experiments for Fine-Mapping Quantitative Trait Loci in Livestock Populations" <doi:10.1101/2019.12.17.879106>.
Carlson, Eberle, Rieder, Yi, Kruglyak, Nickerson (2004) ``Selecting a maximally informative set of single-nucleotide polymorphisms for association analyses using linkage disequilibrium" <doi:10.1086/381000>.

**Depends** R (>= 3.5.0)

**Imports** parallel, Matrix, foreach, rlist, pwr

**License** GPL (>= 2)

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.0.2

**NeedsCompilation** no

**Author** Dörte Wittenburg [aut, cre],
Michael Doschoris [aut],
Jan Klosa [ctb]

**Maintainer** Dörte Wittenburg <wittenburg@fbn-dummerstorf.de>

**Repository** CRAN

**Date/Publication** 2020-08-06 12:30:06 UTC

# R topics documented:



---

AR1 *Correlation matrix of an autoregressive model of order 1*

---

## Description

An order-1 autoregressive correlation matrix is set up which is used for the examples on power and sample size calculation.

## Usage

```
AR1(p, rho)
```

## Arguments

| | |
|---|---|
| p | dimension |
| rho | correlation |

## Value

(p x p) matrix

## Examples

```
AR1(10, 0.2)
```

calcvar *Variance of estimator*

## Description

Calculation of variance of estimator and residual degrees of freedom

## Usage

```
calcvar(lambda, eigendec, n, weights = 1)
```

## Arguments

| | |
|---|---|
| lambda | shrinkage parameter |
| eigendec | eigenvalue decomposition of (p x p) correlation matrix R |
| n | sample size |
| weights | vector (LEN p) of SNP-specific weights or scalar if weights are equal for all SNPs; default value 1 |

## Details

The variance of estimator beta (regression coefficient of SNP-BLUP approach) and the residual degrees of freedom are calculated based on the eigenvalue decomposition of correlation matrix R

## Value

df residual degrees of freedom

var.beta vector (LEN p) of variance of estimator beta up to a constant (i.e. residual variance / n)

## Examples

```
### correlation matrix (should depend on sire haplotypes)
R <- AR1(100, rho = 0.1)
eigendec <- eigen(R)
out <- calcvar(1200, eigendec, 100)
```

## coeff.beta.k *Ratio of expected value to variance of estimator*

### Description

The ratio of expected value to standard deviation is calculated for the estimator of a selected regression coefficient.

### Usage

```
coeff.beta.k(k, beta.true, lambda, eigendec, n, weights = 1)
```

### Arguments

| | |
|---|---|
| k | index of selected regression coefficient |
| beta.true | (LEN p) vector of regression coefficients |
| lambda | shrinkage parameter |
| eigendec | eigenvalue decomposition of (p x p) correlation matrix R |
| n | sample size |
| weights | vector (LEN p) of SNP-specific weights or scalar if weights are equal for all SNPs; default value 1 |

### Value

ratio

## CovarMatrix *Calculation of covariance matrices from maternal and paternal LD*

### Description

The covariance matrix is set as maternal plus paternal LD matrix where the paternal part is a weighted average of sire-specific LD matrices.

### Usage

```
CovarMatrix(exp_freq_mat, LDDam, LDSire, Ns)
```

### Arguments

| | |
|---|---|
| exp_freq_mat | [MATRIX] paternal EXPECTATION matrix |
| LDDam | [MATRIX] maternal Linkage Disequilibrium matrix |
| LDSire | [LIST] Linkage disequilibrium matrices for the sires; each element of the list corresponds to a family |
| Ns | [VECTOR] family size for each sire s |

## Details

The internal suMM function works on lists!

## Value

covK (p x p) matrix of covariance between markers

## Examples

```
data(testdata)
G <- Haplo2Geno(H.sire)
E <- ExpectMat(G)
LDfam2 <- LDsire(H.sire, pos.chr, family = 3:4)
LDfam3 <- LDsire(H.sire, pos.chr, family = 5:6)
## covariance matrix based on sires 2 and 3 only, each with 100 progeny
K <- CovarMatrix(E[2:3, ], LDDam = matLD, LDSire = list(LDfam2, LDfam3), Ns = c(100, 100))
```

---

# CovMat *Calculation of covariance or correlation matrix*

---

## Description

The theoretical covariance between pairs of markers is calculated from either paternal haplotypes and maternal linkage disequilibrium (LD) or vise versa. A genetic map is required. The implementation relies on paternal half-sib families and biallelic markers such as single nucleotide polymorphisms (SNP).

## Usage

```
CovMat(linkMat, haploMat, nfam, pos_chr, map_fun = "haldane", corr = F)
```

## Arguments

| | |
|---|---|
| linkMat | (p x p) matrix of maternal LD between pairs of p markers; matrix is block diagonal in case of multiple chromosomes |
| haploMat | (2N x p) matrix of sires haplotypes for all chromosomes (2 lines per sire) codes as 0's and 1's reflecting reference and alternate alleles |
| nfam | vector (LEN N) containing number of progeny per sire or scalar value in case of equal family size |
| pos_chr | list (LEN number of chromosomes) of vectors (LEN number of markers) of genetic positions in Morgan per chromosome |
| map_fun | character string of mapping function used; so far "haldane" (default) and "kosambi" are enabled |
| corr | logical; FALSE (default) if output is covariance matrix or TRUE if output is correlation matrix |

## Value

list (LEN 2) of matrix (DIM $p1$ x $p1$) and vector (LEN $p1$) with $p1 \leq p$

`K` covariance matrix OR

`R` correlation matrix

`valid.snps` vector of SNP indices considered for covariance/ correlation matrix

## Note

If maternal half-sib families are used, the roles of sire/dam are swapped. Multiple families can be considered.

Family size is used for weighting covariance terms in case of multiple half-sib families. It only matters if number of progeny differs.

If you have maternal haplotypes (H.mothers; same format as H.sire) instead of maternal LD (matLD) then LD can be estimated from counting haplotype frequencies as:

`matLD <-LDdam(inMat = H.mother,pos.chr)`

If multiple chromosomes are considered, then, for instance:

`pos.chr <-list(pos.snp.chr1,pos.snp.chr2,pos.snp.chr3)`

## References

Wittenburg, Bonk, Doschoris, Reyer (2019) "Design of Experiments for Fine-Mapping Quantitative Trait Loci in Livestock Populations" https://doi.org/10.1101/2019.12.17.879106

## Examples

```
### 1: INPUT DATA
data(testdata)
### 2: COVARIANCE/CORRELATION MATRIX
corrmat <- CovMat(matLD, H.sire, 100, pos.chr, corr = TRUE)
### 3: TAGSNPS FROM CORRELATION MATRIX
bin <- tagSNP(corrmat$R)
bin <- tagSNP(corrmat$R, 0.5)
```

---

# ExpectMat *Expected value of paternally inherited allele*

---

## Description

Expected value is +/-0.5 if sire is homozygous reference/ alternate allele or 0 if sire is heterozygous at the investigated marker

## Usage

`ExpectMat(inMat)`

### Arguments

| | |
|---|---|
| inMat | [MATRIX] The paternal genotype matrix |

### Value

ExP.Fa (N x p) matrix of expected values

### Examples

```
data(testdata)
G <- Haplo2Geno(H.sire)
E <- ExpectMat(G)
```

---

## H.sire *testdata: sire haplotypes*

---

### Description

(2N x p) matrix of sire haplotypes for all chromosomes (2 lines per sire); unknown alleles are marked as 9

### Usage

```
H.sire
```

### Format

An object of class matrix with 10 rows and 300 columns.

---

## Haplo2Geno *Conversion of haplotypes into genotypes*

---

### Description

Haplotypes are converted into into genotypes without checking for missing values.

### Usage

```
Haplo2Geno(inpMat)
```

### Arguments

| | |
|---|---|
| inpMat | [MATRIX] haplotype matrix (2 lines per individual) |

### Value

outMa (N x p) genotype matrix

## Examples

```
data(testdata)
G <- Haplo2Geno(H.sire)
```

---

LDdam *Calculation of maternal LD matrix*

---

## Description

Matrix containing linkage disequilibrium between marker pairs on maternal gametes is set up by counting haplotypes frequencies.

## Usage

```
LDdam(inMat, pos_chr)
```

## Arguments

| | |
|---|---|
| inMat | [MATRIX] The maternal HAPLOTYPE matrix. |
| pos_chr | [LIST] The marker positions in Morgan on chromosomes. |

## Details

The function generates a block diagonal sparse matrix based on Matrix::bdiag. Use as.matrix() to obtain a regular one.

## Value

Dd (p x p) matrix of maternal LD

## Examples

```
## haplotype matrix of n individuals at p SNPs
p <- 10; n <- 4
mat <- matrix(ncol = p, nrow = 2 * n, sample(c(0, 1), size = 2 * n * p, replace = TRUE))
LDdam(mat, list(1:p))
```

LDsire *Calculation of paternal LD matrix*

## Description

Matrix containing linkage disequilibrium between marker pairs on paternal gametes is set up from sire haplotypes and genetic-map information for each half-sib family.

## Usage

```
LDsire(inMat, pos_chr, family, map_fun = "haldane")
```

## Arguments

| | |
|---|---|
| inMat | [MATRIX] Haplotype matrix for sires for all chromosomes. |
| pos_chr | [LIST] The marker positions in Morgan on chromosomes. |
| family | [VECTOR] Which family (sire) should be processed? Vector with consecutive entries of the form 1:2, 3:4, 5:6 and so on, linking to haplotypes (rows in inMat) of the corresponding sire |
| map_fun | ["haldane" or "kosambi"] The mapping function applied. |

## Details

The function generates a block diagonal sparse matrix based on Matrix::bdiag. Use as.matrix() to obtain a regular one.

## Value

Ds

Ds (p x p) matrix of paternal LD

## Examples

```
data(testdata)
LDfam2 <- LDsire(H.sire, pos.chr, family = 3:4)
```

## matLD *testdata: maternal linkage disequilibrium*

### Description

(p x p) matrix of maternal LD between pairs of p markers; matrix is block diagonal in case of multiple chromosomes

### Usage

```
matLD
```

### Format

An object of class `matrix` with 300 rows and 300 columns.

## pos.chr *testdata: genetic map positions*

### Description

list of vectors of genetic map positions per chromosome

### Usage

```
pos.chr
```

### Format

An object of class `list` of length 1.

## pwr.normtest *Probability under alternative hypothesis (power)*

### Description

Calculation of power is based on normal distribution. At each selected QTL position, the probability of the corresponding regression coefficient being different from zero is calculated using a t-like test statistic which has normal distribution with mean `E(beta_k)/sqrt{Var(beta_k)}` and variance 1. Under the null hypothesis `beta_k = 0`, `E(beta_k) = 0`. Then, the mean value is returned as power.

### Usage

```
pwr.normtest(R, n, betaSE, lambda, pos, weights = 1, alpha = 0.01)
```

### Arguments

| | |
|---|---|
| R | (p x p) matrix containing theoretical correlation between SNP pairs |
| n | sample size |
| betaSE | effect size relative to residual standard deviation |
| lambda | shrinkage parameter |
| pos | vector (LEN nqtl) of SNP indices for assumed QTL positions |
| weights | weights vector (LEN p) of SNP-specific weights or scalar if weights are equal for all SNPs; default value 1 |
| alpha | type-I error level; default value 0.01 |

### Value

`result` mean power at selected QTL positions

`h2.le` QTL heritability under linkage-equilibrium assumption

`h2.ld` QTL heritability under linkage-disequilibrium assumption

### References

Wittenburg, Bonk, Doschoris, Reyer (2019) Design of Experiments for Fine-Mapping Quantitative Trait Loci in Livestock Populations https://doi.org/10.1101/2019.12.17.879106

### Examples

```
### correlation matrix (should depend on sire haplotypes)
R <- AR1(100, rho = 0.1)
### positions of putative QTL signals
pos <- c(14, 75)
### power at given sample size and other parameters
pwr.normtest(R, 100, 0.35, 1200, pos)
```

---

## pwr.snpblup *Wrapper function for sample size calculation*

---

### Description

Given parameters specified by the experimenter, optimal sample size is estimated by repeatedly applying `search.best.n.bisection`.

## Usage

```
pwr.snpblup(
  nfathers,
  nqtl,
  h2,
  R,
  rep = 10,
  nmax = 5000,
  weights = 1,
  typeII = 0.2,
  alpha = 0.01
)
```

## Arguments

| | |
|---|---|
| nfathers | number of half-sib families |
| nqtl | number of QTL assumed |
| h2 | heritability captured by QTL |
| R | (p x p) matrix containing theoretical correlation between SNP pairs |
| rep | number of repetitions; default value 10 |
| nmax | maximum value for grid search; default value 5000 |
| weights | vector (LEN p) of SNP-specific weights or scalar if weights are equal for all SNPs; default value 1 |
| typeII | type-II error level; default value 0.2 |
| alpha | type-I error level; default value 0.01 |

## Details

Sample size depends on parameters specified by the experimenter (number of half-sib families, number of QTL, heritability, correlation matrix). These values are converted into parameters required for the probability density function under the alternative hypothesis (beta_k !=0, for k selected QTL positions). As power depends on the selected QTL positions, these are sampled at random and power calculations are repeated. Afterwards the mean value is a plausible estimate of optimal sample size.

Linear model for SNP-BLUP approach: `y = X beta + e` with `t(beta) = (beta_1,ldots,beta_p)`
Ridge approach: `hat{beta} = (Xt X + I lambda)^{-1} Xt y`

The identity matrix `I` can be replaced by a diagonal matrix containing SNP-specific weights yielding a generalised ridge approach.

## Value

vector of optimal sample size over all repetitions

## References

Wittenburg, Bonk, Doschoris, Reyer (2019) Design of Experiments for Fine-Mapping Quantitative Trait Loci in Livestock Populations https://doi.org/10.1101/2019.12.17.879106

## Examples

```
### input parameters specified by experimenter
# number of half-sib families
nfathers <- 10
# number of assumed QTL
nqtl <- 2
# QTL heritability
h2 <- 0.2
### correlation matrix (should depend on sire haplotypes)
R <- AR1(100, rho = 0.1)
### optimal sample size in a multi-marker approach
set.seed(11)
pwr.snpblup(nfathers, nqtl, h2, R, rep = 1)
```

---

# search.best.n.bisection

*Method of bisection for estimating optimal sample size*

---

## Description

A grid [nstart,nmax] for possible sample size is considered. Instead of executing a time-consuming grid search, the method of bisection is applied to this interval. For each step, the function pwr.normtest is called for the given set of parameters.

## Usage

```
search.best.n.bisection(
  R,
  betaSE,
  lambda,
  pos,
  nstart,
  nmax,
  weights = 1,
  typeII = 0.2,
  alpha = 0.01
)
```

## Arguments

| | |
|---|---|
| R | (p x p) matrix containing theoretical correlation between SNP pairs |
| betaSE | effect size relative to residual standard deviation |
| lambda | shrinkage parameter |
| pos | vector (LEN nqtl) of SNP indices for assumed QTL positions |
| nstart | minimum value for grid search |
| nmax | maximum value for grid search |

| | |
|---|---|
| weights | vector (LEN p) of SNP-specific weights or scalar if weights are equal for all SNPs; default value 1 |
| typeII | type-II error level; default value 0.2 |
| alpha | type-I error level; default value 0.01 |

## Value

integer of optimal sample size

## Examples

```
### correlation matrix (should depend on sire haplotypes)
R <- AR1(100, rho = 0.1)
### positions of putative QTL signals
pos <- c(14, 75)
### optimal sample size
search.best.n.bisection(R, 0.35, 1200, pos, 10, 5000)
```

---

# simpleM *Calculation of effective number of independent tests*

---

## Description

Adapted simpleM method which considers theoretical correlation between SNP pairs instead of composite LD values. Principal component decomposition yields the effective number of independent tests. This value is needed for the Bonferroni correction of type-I error when testing SNP effects based on a single-marker model.

## Usage

```
simpleM(mat, quant = 0.995)
```

## Arguments

| | |
|---|---|
| mat | correlation matrix |
| quant | percentage cutoff, variation of SNP data explained by eigenvalues; default value 0.995 |

## Value

effective number of independent tests

## References

Gao, Starmer & Martin (2008) A multiple testing correction method for genetic association studies using correlated single nucleotide polymorphisms. Genet Epidemiol, 32:361-369.

## Examples

```
### correlation matrix (should depend on sire haplotypes)
R <- AR1(100, rho = 0.1)
### effective number of tests
Meff <- simpleM(R)
### relative effect size given heritability and number of QTL signals
h2 <- 0.2
nqtl <- 2
betaSE <- sqrt(h2 / (nqtl - nqtl * h2))
### optimal sample size in a single-marker approach
pwr::pwr.t.test(d = betaSE, sig.level = 0.01 / Meff, power = 0.8,
 alternative = "two.sided", type = "one.sample")
```

---

`startvalue` *Start value for estimating optimal sample size*

---

## Description

Calculation of start value for estimating optimal sample size

## Usage

```
startvalue(lambda, R, nfam, weights = 1)
```

## Arguments

| | |
|---|---|
| `lambda` | shrinkage parameter |
| `R` | (p x p) matrix containing theoretical correlation between SNP pairs |
| `nfam` | number of half-sib families |
| `weights` | vector (LEN p) of SNP-specific weights or scalar if weights are equal for all SNPs; default value 1 |

## Details

Minimum sample size that exceeds residual degrees of freedom; this value can be used as start value in grid search for optimal sample size

## Value

start value

## Examples

```
### correlation matrix (should depend on sire haplotypes)
R <- AR1(100, rho = 0.1)
startvalue(1200, R, 10)
```

tagSNP *tagSNP*

## Description

Grouping of markers depending on correlation structure

## Usage

```
tagSNP(mat, threshold = 0.8)
```

## Arguments

| | |
|---|---|
| mat | (p x p) correlation matrix |
| threshold | lower value of correlation considered for grouping |

## Details

Grouping of markers is based on the correlation matrix. Apart from this, the strategy for grouping is similar to Carlson et al. (2004). A representative marker is suggested for each group.

## Value

list (LEN number of groups) of lists (LEN 2); marker names correspond to column names of mat

snps vector of marker IDs in group

tagsnp representative marker suggested for this group

## References

Carlson, Eberle, Rieder, Yi, Kruglyak & Nickerson (2004) Selecting a maximally informative set of single-nucleotide polymorphisms for association analyses using linkage disequilibrium. Am J Hum Genet, 74:106-120.

## Examples

```
### 1: INPUT DATA
data(testdata)
### 2: COVARIANCE/CORRELATION MATRIX
corrmat <- CovMat(matLD, H.sire, 100, pos.chr, corr = TRUE)
### 3: TAGSNPS FROM CORRELATION MATRIX
bin <- tagSNP(corrmat$R)
bin <- tagSNP(corrmat$R, 0.5)
as.numeric(unlist(rlist::list.select(bin, tagsnp)))
```

testdata *Description of the testdata*

## Description

The data set contains paternal haplotypes, maternal LD and genetic map positions that are required to calculate the covariance between pairs of markers.

The raw data can be downloaded at the source given below. Then, executing the following R code leads to the data that have been provided as `testdata.RData`.

**H.sire** (2N x p) haplotype matrix for sires for all chromosomes (2 lines per sire)

**matLD** (p x p) matrix of maternal LD between pairs of p markers; matrix is block diagonal in case of multiple chromosomes

**pos.chr** list of vectors of genetic map positions per chromosome

## Source

The data are available from the RADAR repository https://dx.doi.org/10.22000/280

## Examples

```
## data.frame of estimates of paternal recombination rate and maternal LD
load('Result.RData')
## list of haplotypes of sires for each chromosome
load('sire_haplotypes.RData')
## physical map
map <- read.table('map50K_ARS_reordered.txt', header = T)
## select target region
chr <- 1
window <- 301:600
## map information of target region
map.target <- map[map$Chr == chr, ][window, ]
Result.target <- Result[(Result$Chr == chr) & (Result$SNP1 %in% window) &
  (Result$SNP2 %in% window), ]
## SNP position in Morgan approximated from recombination rate
part <- Result.target[Result.target$SNP1 == window[1], ]
sp <- smooth.spline(x = map.target$locus_Mb[part$SNP2 - window[1] + 1], y = part$Theta, df = 4)
pos.snp <- predict(sp, x =  map.target$locus_Mb[window - window[1] + 1])$y
## list of SNPs positions
pos.chr <- list(pos.snp)
## haplotypes of sires (mating candidates) in target region
H.sire <- rlist::list.rbind(haps[[chr]])[, window]
## matrix of maternal LD (block diagonal if multiple chromosome)
matLD <- matrix(0, ncol = length(window), nrow = length(window))
## off-diagonal elements
for(l in 1:nrow(Result.target)){
  id1 <- Result.target$SNP1[l] - window[1] + 1
  id2 <- Result.target$SNP2[l] - window[1] + 1
```

```
  matLD[id1, id2] <- matLD[id2, id1] <- Result.target$D[l]
}
## diagonal elements
for(k in unique(Result.target$SNP1)){
  id <- k - window[1] + 1
 p <- Result.target$fAA[Result.target$SNP1 == k] + Result.target$fAB[Result.target$SNP1 == k]
  matLD[id, id] <- max(p * (1 - p))
}
```

# Index