Package 'diaQTL'

June 27, 2020

Title QTL Analysis in Diallel Populations

Version 0.60

Author Jeffrey B. Endelman and Rodrigo R. Amadeu

Maintainer Jeffrey Endelman <endelman@wisc.edu>

Description QTL analysis of diploid and autotetraploid diallel populations. Phenotypes are regressed on genotype probabilities, and the regression coefficients are random effects.

Depends R (>= 3.5.0)

License GPL-3

LazyData true

RoxygenNote 7.1.0

Roxygen list(markdown = TRUE)

Encoding UTF-8

Imports BGLR, ggplot2, methods, coda, Matrix, scam

Suggests knitr, rmarkdown

VignetteBuilder knitr

R topics documented:

Amat *A matrix*

Description

Calculates the additive (A) relationship matrix from founder genotype probabilities

Usage

```
Amat(data, chrom = NULL, exclude.marker = NULL)
```

Arguments

data	Variable inheriting from class diallel_geno
chrom	Only use markers from these chromosomes
exclude.marker	Markers to exclude

Details

Additive relationships are calculated from kinship coefficients of order 2 (Gallais 2003). Can be subset to one or more chromosomes (which is useful for the leave-one-chromosome-out kinship method), or specific markers can be excluded after QTL detection.

Value

A matrix

References

Gallais, A. 2003. Quantitative Genetics and Breeding Methods in Autopolyploid Plants. Institut National de la Recherche Agronomique, Paris.

Examples

```
## Not run:
  Amat_example = Amat(data = diallel_example)
  Amat_example = Amat(data = diallel_example, chrom=c(1:11)) #leave chromosome 12 out
  Amat_example = Amat(data = diallel_example, exclude.marker = "solcap_snp_c2_25522")

## End(Not run)
```

diallel_geno-class *S4 class with genotype data*

Description

S4 class with genotype data

Slots

`ploidy` Either 2 or 4

`ped` data frame with pedigree information. Variables are id,population,mother,father

`map` data frame with marker,chrom,and position (either bp or cM)

`geno` list of length 2. The first element (named "A") is a list of sparse matrices, one for each marker, with dimensions (id x alleles), containing the allele dosages, which are the regression variables for additive effects. The second (optional) element (named "D") has the same structure (list of sparse matrices), but each matrix contains the dosage of allele-pairs, which are the regression variables for digenic dominance effects.

diallel_geno_pheno-class *S4 class with genotype and phenotype data*

Description

S4 class with genotype and phenotype data

Slots

`ploidy` Either 2 or 4

`ped` data frame with pedigree information. Variables are id,population,mother,father

`map` data frame with marker,chrom,and position (either bp or cM)

`geno` list of length 2. The first element (named "A") is a list of sparse matrices, one for each marker, with dimensions (id x alleles), containing the allele dosages, which are the regression variables for additive effects. The second (optional) element (named "D") has the same structure (list of sparse matrices), but each matrix contains the dosage of allele-pairs, which are the regression variables for digenic dominance effects.

`pheno` data frame of phenotypes

`X` incidence matrix for fixed effects

`Z` incidence matrix for individuals

Dmat *D matrix*

Description

Calculates the dominance (D) relationship matrix from founder genotype probabilities

Usage

```
Dmat(data, chrom = NULL, exclude.marker = NULL)
```

Arguments

data	Variable inheriting from class diallel_geno
chrom	Only use markers from these chromosomes
exclude.marker	Markers to exclude

Details

Dominance relationships are calculated from kinship coefficients of order 4 (Gallais 2003). An entire chromosome or specific markers can be excluded.

Value

Digenic dominance relationship matrix

References

Gallais, A. 2003. Quantitative Genetics and Breeding Methods in Autopolyploid Plants. Institut National de la Recherche Agronomique, Paris.

Examples

```
## Not run:
  Dmat_example = Dmat(data = diallel_example)
  Dmat_example = Dmat(data = diallel_example, chrom=c(10,11)) #leave chromosome 12 out
  Dmat_example = Dmat(data = diallel_example, exclude.marker = "solcap_snp_c2_25522")

## End(Not run)
```

dosage *Get parental allele dosage estimates*

Description

Get parental allele dosage estimates

Usage

```
dosage(data, marker = NULL, id = NULL)
```

Arguments

data	Variable inheriting from class diallel_geno
marker	Name of marker
id	Name of individual

Details

Function can be used to get parental allele dosage estimates at a single marker for all individuals (in which case id should be NULL) or for a single individual for all markers (in which case marker should be NULL)

Value

Matrix of (id or markers) x parental alleles

Examples

```
## Not run:
  dosage_example = dosage(data = diallel_example,
                          marker = "solcap_snp_c2_25522")
  dosage_example = dosage(data = diallel_example,
                          id = "W15263-8R")

## End(Not run)
```

F1codes *Genotype codes for F1 populations*

Description

Character vector with the 100 possible tetraploid genotypes for a F1 population. Maternal alleles are denoted 1,2,3,4 and paternal alleles 5,6,7,8.

Usage

```
data(F1codes)
```

Format

character vector

`fitQTL` *Fit a single QTL model*

Description

Fit a single QTL model

Usage

```
fitQTL(data, trait, marker, params, dominance = FALSE, cofactor = NULL)
```

Arguments

`data`	Variable of class `diallel_geno_pheno`
`trait`	Name of trait
`marker`	Name of marker to fit as QTL
`params`	List containing the number of burn-in (burnIn) and total iterations (nIter)
`dominance`	Logical variable whether to include digenic dominance effects
`cofactor`	Name of marker to fit as cofactor

Details

Standard errors of the posterior mean estimates are calculated by dividing the SD of the Markov Chain by the square root of the effective number of iterations, which is calculated by function `effectiveSize` in R package `coda`. The error bars on the plot of additive effects correspond to +/- 1.96*SE (95 percent confidence interval). For binary traits, R2 = the squared phi correlation. The additive and digenic dominance variances are reported as a proportion of the total variance: h2=Va/(Va+Vd+Vresid) and d2=Vd/(Va+Vd+Vresid).

Value

List containing

R2 Coefficient of determination

DIC Deviance Information Criterion

h2 Mean and SE for proportion of variance due to additive effects

effectsA Mean and SE of the additive effects for parental alleles

plotA ggplot object for additive effects

If dominance=T the list also contains

d2 Mean and SE for proportion of variance due to digenic dominance effects

effectsD Mean and SE of the dominance effects for parental allele pairs

plotD ggplot object for dominance effects

Examples

```
## Not run:
## additive effects
params1 <- set_params( diallel_example, trait = "tuber_shape" )

fit1 <- fitQTL( data = diallel_example,
                trait = "tuber_shape",
                params = params1,
                marker = "solcap_snp_c2_25522")

## additive + dominance effects
params2 <- set_params( diallel_example, trait = "tuber_shape", dominance = TRUE )

fit2 <- fitQTL( data = diallel_example,
                trait = "tuber_shape",
                params = params2,
                marker = "solcap_snp_c2_25522",
                dominance = TRUE)


## End(Not run)
```

`LODthresh` *LOD thresholds for scan1*

Description

LOD thresholds for scan1

Usage

```
LODthresh(genome.size, num.parents, ploidy)
```

Arguments

`genome.size`	Genome size in Morgans (not centiMorgans)
`num.parents`	Number of parents
`ploidy`	2 or 4

Details

LOD thresholds to control the genome-wide false positive rate at 0.05 were determined via simulation for up to 20 parents and genome sizes up to 12 Morgans. A monotone increasing concave curve was fit to these results using R package `scam` and is used for prediction. (The LOD threshold does not depend on population size.)

Value

LOD threshold

plot_dosage *Plot parental allele dosage*

Description

Plot parental allele dosages across the chromosome for one individual

Usage

```
plot_dosage(data, id, chrom, distance, marker = NULL)
```

Arguments

data	Variable inheriting from class diallel_geno
id	Name of individual
chrom	Name of chromosome
distance	Either "cM" for centiMorgans or "bp" for base pairs (depending on position used in the input map)
marker	Optional, position of marker indicated with dashed line

Value

ggplot object

Examples

```
## Not run:
plot_dosage(data = diallel_example,
            id = "W15263-8R",
            chrom = 10)

plot_dosage(data = diallel_example,
            id = "W15263-8R",
            chrom = 10,
            marker = "solcap_snp_c2_25522")

## End(Not run)
```

read_data *Read data files*

Description

Reads genotype, pedigree, and phenotype data files

Usage

```
read_data(genofile, ploidy = 4, pedfile, phenofile = NULL, fixed = NULL)
```

Arguments

genofile	File with map and genotype probabilities
ploidy	Allowable values are 2 or 4
pedfile	File with pedigree information (four column format: id,population,mother,father)
phenofile	File with phenotype data (optional)
fixed	If there are fixed effects, this is a character vector of "factor" or "numeric"

Details

First three columns of the genotype file are marker, chromosome, and position. Columns 4 through (n+4) correspond to the n individuals of the population. The genotype information for each marker x individual combination is a string with the format "state|state|state...=>prob|prob|prob...", where "state" refers to the genotype state and "prob" is the genotype probability in decimal format. Only states with nonzero probabilities need to be listed. The encoding for the states in tetraploids is described in the documentation for the F1codes and S1codes datasets that come with the package. For diploids, there are 4 F1 genotype codes, 1,2,3,4, which correspond to allele combinations 1-3,1-4,2-3,2-4, respectively; the S1 genotype codes 1,2,3 correspond to 1-1,1-2,2-2, respectively. For the phenotype file, first column is id, followed by traits, and then any fixed effects. Pass a character vector for the function argument "fixed" to specify whether each effect is a factor or numeric covariate. The number of traits is deduced based on the number of columns. Binary traits must be coded "N"/"Y" and are converted to 0/1 internally for analysis by probit regression.

Value

Variable of class diallel_geno if phenofile is NULL, otherwise diallel_geno_pheno

Examples

```
## Not run:
  ## Get the location of raw csv files examples
  genocsv = system.file( "tutorial", "potato_geno.csv", package = "diaQTL" )
  pedcsv = system.file( "tutorial", "potato_ped.csv", package = "diaQTL" )
  phenocsv = system.file( "tutorial", "potato_pheno.csv", package = "diaQTL" )

  ## Check their location in the system
  print(genocsv)
  print(pedcsv)
  print(phenocsv)

  ## Load them in R
  diallel_example <- read_data(genofile = genocsv,
                               ploidy = 4,
                               pedfile = pedcsv,
                               phenofile = phenocsv)

## End(Not run)
```

S1codes *Genotype codes for S1 populations*

Description

Character vector with the 35 possible tetraploid genotypes for a S1 population. Alleles are denoted 1,2,3,4.

Usage

```
data(S1codes)
```

Format

character vector

scan1 *Single QTL scan*

Description

Performs a linear regression for each position in the map.

Usage

```
scan1(
  data,
  trait,
  params,
  chrom = NULL,
  dominance = F,
  cofactor = NULL,
  n.core = 1
)
```

Arguments

data	Variable of class diallel_geno_pheno
trait	Name of trait
params	List containing burnIn and nIter, use set_params function to estimate it
chrom	Names of chromosomes to scan (default is all)
dominance	Logical variable whether to include digenic dominance effects
cofactor	Optional name of marker to include as cofactor in the scan
n.core	Number of cores to use for parallel execution by forking (do not use with GUI)

Details

For non-binary traits, R2 is the proportion of variance explained by the regression. For binary traits, R2 is the squared phi correlation. LOD score is the difference between the log-likelihood of the model with QTL and the null model (no QTL); higher values are better. deltaDIC is the difference between the DIC of the model with QTL minus the DIC of the null model; lower values are better.

Value

Data frame containing the map, LOD, R2 and deltaDIC results.

Examples

```
## Not run:
  par1 <- set_params(data = diallel_example,
                     trait = "tuber_shape")

  scan1_example <- scan1(data = diallel_example,
                chrom = 10,
                trait = "tuber_shape",
                params = par1)

## End(Not run)
```

---

scan1_permute *Permutation test for scan1*

Description

Permutation test for scan1

Usage

```
scan1_permute(
  data,
  trait,
  params,
  n.permute = 1000,
  chrom = NULL,
  dominance = F,
  cofactor = NULL,
  n.core = 1
)
```

Arguments

data	Variable of class diallel_geno_pheno
trait	Name of trait
params	Number of burn-in and total iterations
n.permute	Number of permutations
chrom	Names of chromosomes to scan (default is all)

dominance	Logical variable whether to include digenic dominance
cofactor	Optional name of marker to include as cofactor in the scan
n.core	Number of cores to use, allows for parallel execution

Value

Data frame with maximum LOD and minimum deltaDIC for each iteration

Examples

```
## Not run:
  par1 <- set_params(data = diallel_example,
                     trait = "tuber_shape")

  ans1_permut <- scan1_permute(data = diallel_example,
                               chrom = 10,
                               trait = "tuber_shape",
                               params = par1,
                               n.permute = 100)


## End(Not run)
```

---

scan1_summary *Summary of scan1 result*

Description

Summary of scan1 result

Usage

```
scan1_summary(
  scan1_data,
  display = T,
  thresh = NULL,
  chromosome = NULL,
  distance = "cM"
)
```

Arguments

scan1_data	output from scan1
display	Logical variable whether to plot the LOD score
thresh	optional, LOD threshold for plotting
chromosome	string with chrom name(s) to plot. By default, all chromosomes are plotted
distance	string with "cM" for centiMorgans or "bp" for basepairs

Value

List containing

peaks Data frame of the markers with the highest LOD score per chromosome

plot ggplot object

Examples

```
## Not run:
  scan1_summary( scan1_example )
  scan1_summary( scan1_example, chromosome = "10" )
  scan1_summary( scan1_example, chromosome = c( "10", "12" ) )

## End(Not run)
```

`set_params` *Determine parameters for scan1*

Description

Determine parameters for scan1

Usage

```
set_params(data, trait, tol = 0.1, burnIn = 50, nIter = 1000)
```

Arguments

`data`	Variable of class `diallel_geno_pheno`
`trait`	Name of trait
`tol`	tolerance for estimating the median
`burnIn`	initial value for burnIn parameter
`nIter`	initial value for nIter parameter

Details

The burn-in and total number of iterations are determined using the Raftery and Lewis diagnostic from R package `coda`, based on a 95% probability that the estimated median of the additive effects is between the quantiles (0.5-tol) to (0.5+tol). For greater precision, decrease the `tol` parameter.

Value

List containing

burnIn Number of burn-in iterations

nIter Total number of iterations

Examples

```
## Not run:
  par1 <- set_params(data = diallel_example,
                     trait = "tuber_shape")

## End(Not run)
```

Index