Package 'polyBreedR'

October 14, 2020

Title Using genome-wide markers for polyploid breeding

Version 0.12

Author Jeffrey B. Endelman

Maintainer Jeffrey Endelman <endelman@wisc.edu>

Description Using genome-wide markers for polyploid breeding

Depends R (>= 3.5.0)

License GPL-3

LazyData true

RoxygenNote 7.1.0

Encoding UTF-8

Imports AGHmatrix, ggplot2, ggrepel, pedigree, grDevices, utils, tidyr, Matrix

Suggests knitr, rmarkdown, asreml

VignetteBuilder knitr

R topics documented:

A_mat *Additive relationship matrix from pedigree*

Description

Additive relationship matrix from pedigree

Usage

```
A_mat(ped, ploidy, order.ped = TRUE)
```

Arguments

ped	Pedigree in three column format: id, mother, father
ploidy	2 or 4
order.ped	TRUE/FALSE does the pedigree need to be ordered so that progeny follow parents

Details

This is a wrapper that prepares the pedigree in the format required for R package AGHmatrix by Amadeu et al. (2016) (cite them if you use this function). A random bivalents model for tetraploid meiosis is assumed.

Value

Additive relationship matrix (dim: indiv x indiv)

References

Amadeu et al. (2016) Plant Genome 9, doi:10.3835/plantgenome2016.01.0009

check_ploidy *Check ploidy*

Description

Fraction of simplex or triplex markers

Usage

```
check_ploidy(geno, map)
```

Arguments

geno	Genotype matrix (markers x indiv)
map	Data frame with marker map (Marker, Chrom, Position)

Details

For every indiv in the genotype matrix, the fraction of markers per chromosome called as simplex or triplex is calculated, which should be low for diploids. A small amount of missing genotype data can be tolerated.

Value

List containing

mat Matrix (indiv x chrom) of results

plot ggplot2 barplot

`check_trio` *Check markers for parent-offspring trio*

Description

Check markers for parent-offspring trio

Usage

```
check_trio(parentage, geno, ploidy)
```

Arguments

`parentage`	Data frame with three columns: id, mother, father
`geno`	Matrix of allele dosages: markers x indiv
`ploidy`	2 or 4

Details

Computes the percentage of markers at which the two parents and offspring have incompatible allele dosages (for tetraploids, the random bivalents model is used). For dihaploid offspring of a single tetraploid parent, use `ploidy = 4` and "haploid" for the father in `parentage`, as well as a diploid (0,1,2) genotype for the offspring. A small amount of missing genotype data can be tolerated.

Value

Data frame with the percentage of incompatible markers for each trio

direct_sum *Direct Sum*

Description

Direct Sum

Usage

```
direct_sum(x)
```

Arguments

x	list of matrices

Value

Sparse Matrix

D_mat *Dominance genomic relationships*

Description

Coefficients and relationship matrix for digenic dominance effects with bi-allelic markers

Usage

```
D_mat(geno, ploidy)
```

Arguments

geno	Matrix of allele dosages: markers x indiv
ploidy	2 or 4

Details

Digenic dominance effects are based on the traditional orthogonal decomposition of genetic variance in panmictic populations (Fisher 1918; Kempthorne 1957; Endelman et al. 2018). The D matrix is computed from the coefficients and scaling factor according to D = tcrossprod(coeff/scale). Missing genotype data is replaced with the population mean.

Value

List containing

coeff Coefficients of the marker effects (dim: indiv x marker)

scale Scaling factor between markers and indiv

mat D matrix

References

Fisher (1918) Trans. Roy. Soc. Edin. 52:399-433.

Kempthorne (1957) An Introduction to Genetic Statistics.

Endelman et al. (2018) Genetics 209:77-87.

`geno_call` *Genotype calls*

Description

Genotype calls based on a normal mixture model

Usage

```
geno_call(
  data,
  filename,
  model.ploidy = 4,
  sample.ploidy = 4,
  min.posterior = 0,
  transform = TRUE
)
```

Arguments

`data`	matrix (markers x id) of input values for the normal mixture model
`filename`	CSV filename with the model parameters
`model.ploidy`	2 or 4 (default)
`sample.ploidy`	2 or 4 (default)
`min.posterior`	minimum posterior probability (default 0) for genotype call
`transform`	TRUE (default) or FALSE whether to apply arcsin square root transformation

Details

The first column of the CSV input file should be the SNP ID, followed by columns for the normal distribution means, standard deviations, and mixture probabilities. Genotype calls are based on the maximum a posteriori (MAP) method. If the posterior probability of the MAP genotype is less than `min.posterior`, then `NA` is returned for that sample. By default, an arcsin square root transformation is applied to the input values to match the approach used by R package fitPoly. To use a tetraploid mixture model for diploid samples, set `sample.ploidy = 2` and `model.ploidy = 4`.

Value

matrix of allele dosages (0,1,2,..ploidy) with dimensions markers x individuals

get_pedigree *Generate pedigree*

Description

Generate pedigree for a set of individuals

Usage

```
get_pedigree(id, pedfile, delim = ",", na.string = "NA", trim = TRUE)
```

Arguments

id	Vector of names of individuals
pedfile	Name of pedigree file
delim	Delimiter for the pedigree file (default is "," for CSV)
na.string	String used for NA in the pedigree file (default is "NA")
trim	TRUE/FALSE whether to trim pedigree (see Details)

Details

Finds ancestors of individuals in a three-column pedigree file (id,mother,father). The id column can be the identifier for an individual or cross. String matches must be exact or based on the naming convention crossID-progenyID. The returned pedigree is ordered using R package `pedigree` so that offspring follow parents. When `trim` is TRUE (default), the pedigree is trimmed to remove ancestors with only one offspring (which are not needed to compute the pedigree relationship matrix).

Value

Data frame with columns id, mother, father

GvsA *Plot G vs. A*

Description

Plot marker-based vs. pedigree-based additive relationship coefficients

Usage

```
GvsA(
  parentage,
  G,
  A,
  filename = NULL,
  thresh.G = Inf,
  thresh.A = 0.5,
  Gmax = NULL,
  Amax = NULL
)
```

Arguments

parentage	Data frame of individuals to plot, with 3 columns: id,mother,father
G	Genomic relationship matrix
A	Pedigree relationship matrix
filename	Name of PDF file to save the results (optional for one individual)
thresh.G	Threshold above which names are displayed (default Inf)
thresh.A	Threshold above which names are displayed (default 0.5)
Gmax	Upper limit for y-axis for plotting. If NULL, maximum value in G is used.
Amax	Upper limit for x-axis for plotting. If NULL, maximum value in A is used.

Details

Useful for finding and correcting pedigree errors. If the G or A coefficient for an individual exceeds the threshold, its name is displayed in the figure. If parentage contains one individual, by default a ggplot2 variable will be returned, but the result can also be written to file. If multiple individuals are present, a filename is required.

G_mat *Additive genomic relationships*

Description

Coefficients and relationship matrix for additive effects with bi-allelic markers

Usage

```
G_mat(geno, ploidy)
```

Arguments

geno	Matrix of allele dosages: markers x indiv
ploidy	2 or 4

Details

Additive effects are based on the traditional orthogonal decomposition of genetic variance in panmictic populations (Fisher 1918; Kempthorne 1957; Endelman et al. 2018). The G matrix is computed from the coefficients and scaling factor according to G = tcrossprod(coeff/scale). Missing genotype data is replaced with the population mean.

Value

List containing

coeff Coefficients of the marker effects (dim: indiv x marker)

scale Scaling factor between markers and indiv

mat G matrix

References

Fisher (1918) Trans. Roy. Soc. Edin. 52:399-433.

Kempthorne (1957) An Introduction to Genetic Statistics.

Endelman et al. (2018) Genetics 209:77-87.

`impute` *Impute missing marker data*

Description

Impute marker data based on the population mean or mode

Usage

```
impute(geno, method)
```

Arguments

`geno`	Matrix of allele dosages with dimensions markers x indiv
`method`	Either "mean" or "mode"

Details

Missing values are imputed with either the population mean or mode (most frequent value) for each marker

Value

Imputed genotype matrix (markers x indiv)

`merge_impute` *Merge two genotype matrices and impute missing data*

Description

Merge two genotype matrices and impute missing data by BLUP

Usage

```
merge_impute(geno1, geno2, ploidy)
```

Arguments

`geno1`	Genotype matrix (coded 0...ploidy) with dimensions markers x indiv
`geno2`	Genotype matrix (coded 0...ploidy) with dimensions markers x indiv
`ploidy`	Either 2 or 4

Details

Designed to impute from low to high density markers. The BLUP method is equivalent to Eq. 4 of Poland et al. (2012), but this function is not iterative. Additional shrinkage toward the mean is applied if needed to keep the imputed values within the range [0,ploidy]. Missing data in the input matrices are imputed with the population mean for each marker. If an individual appears in both input matrices, it is renamed with suffixes ".1" and ".2" and treated as two different individuals. Monomorphic markers are removed.

Value

Imputed genotype matrix (markers x indiv)

References

Poland et al. (2012) Plant Genome 5:103-113.

---

readXY *Read SNP array intensity data*

Description

Read SNP array intensity data

Usage

```
readXY(filename, skip, output = "ratio")
```

Arguments

filename	filename
skip	number of lines to skip before the header line with the column names
output	Either "ratio" or "theta"

Details

The first two columns of the tab-delimited input file should be the SNP and Sample ID. Columns labeled "X" and "Y" contain the signal intensities for the two alleles. Use output to specify whether to return the ratio = Y/(X+Y) or theta = atan(Y/X)*2/pi.

Value

matrix with dimensions markers x individuals

Step2 *Genomic prediction of marker effects from multi-environment trials*

Description

Additive and dominance marker effects predicted by BLUP, based on variance components estimated with ASReml-R (must have license)

Usage

```
Step2(pheno, G, D = NULL, AIC = TRUE, silent = FALSE, workspace = "128mb")
```

Arguments

pheno	List of matrices containing BLUEs and variance-covariance matrix
G	Returned object from G_mat for additive effects
D	Optional, returned object from D_mat for dominance effects (default is NULL)
AIC	Boolean variable, whether to select A vs. AD model based on AIC (default is TRUE)
silent	Boolean variable, whether to suppress ASReml-R convergence monitoring (default is FALSE)
workspace	Workspace memory for ASReml-R (default is "128mb")

Details

Best practice for the analysis of datasets comprising multiple experiments follows a two-step approach (Damesa et al. 2017). Step 1 generates a vector of genotype BLUEs (and variance-covariance matrix) for each experiment, taking its design and potentially spatial factors into account. In Step 2, the BLUEs are used as a response variable in a mixed model, and the covariance matrix for the residuals is constrained to equal the direct sum of the covariance matrices of the BLUEs. In the current implementation, the Step 2 model contains three genetic effects: additive, digenic dominant, and independent (to capture higher-order non-additive effects). The digenic dominant term has nonzero mean to allow for heterosis/inbreeding depression (Varona et al. 2018). The model also contains a fixed effect for each experiment and independent random effects for the genotype x environment interaction (an environment may contain multiple experiments). Each element of pheno corresponds to an experiment and is a list with components named "env" and "y". Env is the name of the environment for that experiment, and "y" is a matrix of dimensions n x (n+1), where n is the number of genotypes in the trial. The first column of the matrix contains the BLUEs, and the remaining n columns are the variance-covariance matrix for the BLUEs. ASReml-R is used to estimate variance components, and BLUPs are computed based on standard formulas (Searle et al. 1992). When D is not NULL, variances are first estimated for the strictly additive (A) model and used as initial values when fitting the additive + dominance (AD) model. If AIC = TRUE, then whichever model (A vs. AD) has lower AIC is used for BLUP; otherwise, BLUPs are computed based on the AD model regardless of the AIC values. Individuals in pheno but not in G are removed from the analysis; to make predictions for unphenotyped individuals, include them in G (and D).

Value

List containing

params Matrix with AIC and estimated parameters

indiv Matrix of predicted values for the individuals in geno

markers Matrix of predicted effects for the markers in geno

References

Damesa et al. 2017. Agronomy Journal 109: 845-857. doi:10.2134/agronj2016.07.0395

Varona et al. 2018. Frontiers in Genetics 9: 78. doi:10.3389/fgene.2018.00078

Searle et al. 1992. Variance Components. doi:10.1002/9780470316856

`update_alias` *Update names based on alias*

Description

Update names based on data frame with alias and preferred name

Usage

```
update_alias(x, alias, remove.space = TRUE)
```

Arguments

`x`	Vector of names to update
`alias`	Data frame with two columns: first is the preferred name and second is the alias
`remove.space`	TRUE/FALSE

Details

Parameter `remove.space` indicates whether blank spaces should be removed before string matching

Value

Vector with updated names

Index