Package 'GWASpoly'

May 4, 2022

Title Genome-wide Association Studies for Autopolyploids

Version 2.11

Author Jeffrey B. Endelman, Umesh R. Rosyara

Maintainer Jeffrey Endelman <endelman@wisc.edu>

Description Designed for genome-wide association studies in autopolyploids.

License GPL-3

LazyData true

RoxygenNote 7.1.1

Encoding UTF-8

Imports rrBLUP, methods, ggplot2, tidyr, stats, parallel, rlang, scam, Matrix

Suggests knitr, rmarkdown

VignetteBuilder knitr

Collate 'GWASpoly.data.R'
'GWASpoly.K.R'
'GWASpoly.R'
'GWASpoly.fitted.R'
'GWASpoly.thresh.R'
'Keff.R'
'LD.plot.R'
'VCF2dosage.R'
'design.score.R'
'fit.QTL.R'
'get.QTL.R'
'get_x.R'
'make.full.R'
'makeLOCO.R'
'manhattan.plot.R'
'qq.plot.R'
'qvalue.R'
'read.GWASpoly.R'
'score.calc.R'
'set.K.R'
'set.params.R'
'set.threshold.R'
'write.GWASpoly.R'

R topics documented:

`fit.QTL` *Test markers as QTL under backward elimination*

Description

Test markers as QTL under backward elimination

Usage

```
fit.QTL(data, trait, qtl, fixed = NULL)
```

Arguments

`data`	variable inheriting from class `GWASpoly.K`
`trait`	name of trait
`qtl`	data frame to specify the multi-QTL model (see Details)
`fixed`	data frame to specify the fixed effects (see Details)

Details

`qtl` is a data frame with columns "Marker" and "Model", where each row corresponds to a QTL. `fixed` is a data frame with columns "Effect" and "Type": the first column is the name of the effect, which must match a column in the phenotype input file, and the second column is either "factor" or "numeric". The p-value and R2 for each marker are based on the likelihood ratio test under backward elimination, comparing the deviance to the chi-squared distribution.

Value

data frame with partial r2 and p-values

get.QTL *Extract significant QTL*

Description

Output a table with significant markers

Usage

```
get.QTL(data, traits = NULL, models = NULL, bp.window = NULL)
```

Arguments

data	Output from set.threshold
traits	Vector of trait names (by default, all traits)
models	Vector of model names (by default, all models)
bp.window	prune output to return only the most significant marker within this window size

Details

To return all significant markers (original behavior of the function), use bp.window=NULL. Assumes input map position in bp.

Value

Data frame with results. Score = -log10(p). Effect = marker effect (not available for the general and diplo-general models because there are multiple effects).

GWASpoly *Compute marker significance scores*

Description

Compute marker significance scores

Usage

```
GWASpoly(data, models, traits = NULL, params = NULL, n.core = 1, quiet = F)
```

Arguments

data	Output from set.K
models	Vector of model names
traits	Vector trait names (by default, all traits)
params	Optional list of params created by set.params
n.core	Number of cores for parallel computing
quiet	TRUE/FALSE whether to suppress output charting progress

Details

The following marker-effect models are available:

- "additive": Indicates the marker effect is proportional to the dosage of the alternate allele
- "X-dom": where X can be any integer between 1 and ploidy/2 and refers to the allele dosage needed for complete dominance (e.g., "1-dom" = simplex dominance, "2-dom" = duplex dominance). The software tries both dominance patterns for a given dosage model, e.g., whether the reference or alternate allele is dominant
- "diplo-general": All heterozygotes have the same effect
- "diplo-additive": All heterozygotes have the same effect, constrained to be halfway between the homozygous effects
- "general": There are no constraints on the effects of the different dosage levels

To specify additional model parameters, such as the inclusion of fixed effects (Q matrix) and the minimum minor allele frequency, use `set.params`

Value

Variable of class `GWASpoly.fitted`

GWASpoly.data-class *S4 class with genotype and phenotype data*

Description

S4 class with genotype and phenotype data

Slots

`map` data frame with columns Marker,Chrom,Position,Ref,Alt

`pheno` data frame of phenotypes

`geno` matrix (individuals x markers) of allele dosages (0,1,2,...ploidy)

`fixed` data frame of fixed effects

`ploidy` ploidy

GWASpoly.fitted-class *S4 class with results from genome-wide scan*

Description

S4 class with results from genome-wide scan

Slots

`map` data frame with columns Marker,Chrom,Position,Ref,Alt
`pheno` data frame of phenotypes
`geno` matrix with allele dosages
`fixed` data frame of fixed effects
`ploidy` ploidy
`K` list of covariance matrices
`scores` -log10(p) results
`effects` estimated marker effects
`params` parameters used for the analysis

`GWASpoly.K-class` *S4 class with genotypes, phenotypes, and polygenic covariance*

Description

S4 class with genotypes, phenotypes, and polygenic covariance

Slots

`map` data frame with columns Marker,Chrom,Position,Ref,Alt
`pheno` data frame of phenotypes
`geno` matrix with allele dosages
`fixed` data frame of fixed effects
`ploidy` ploidy
`K` list of covariance matrices (one for each chromosome)

`GWASpoly.thresh-class` *S4 class with results from genome-wide scan and detection threshold*

Description

S4 class with results from genome-wide scan and detection threshold

Slots

`map` data frame with columns Marker,Chrom,Position,Ref,Alt
`pheno` data frame of phenotypes
`geno` matrix with allele dosages
`fixed` data frame of fixed effects
`ploidy` ploidy
`K` list of covariance matrices
`scores` -log10(p) results
`effects` estimated marker effects
`params` parameters used for the analysis
`threshold` thresholds for significance

LD.plot *Plot LD vs distance*

Description

Plot LD vs distance

Usage

```
LD.plot(data, max.pair = 10000, dof = 8, max.loci = NULL)
```

Arguments

data	variable inheriting from class GWASpoly
max.pair	maximum number of r2 pairs for the spline
dof	degrees of freedom for the spline
max.loci	maximum number of markers to use per chromosome

Details

A monotone decreasing, convex spline is fit using R package scam.

Value

ggplot2 object

manhattan.plot *Create Manhattan plot*

Description

Create Manhattan plot

Usage

```
manhattan.plot(data, traits = NULL, models = NULL, chrom = NULL)
```

Arguments

data	Variable of class GWASpoly.fitted
traits	Vector of trait names (by default, all traits plotted)
models	Vector of model names (by default, all models plotted)
chrom	optional, to plot only one chromosome

Details

Results for the ref and alt versions of the dominance model are combined. If `data` is the output from `set.threshold`, then the threshold is displayed as a horizontal dashed line when `models` contains a single model. Because the threshold varies between models, it is not drawn when multiple models are included. Although the ref and alt versions of each dominance model are slightly different (as seen with `qq.plot`), they are treated as a single model for the Manhattan plot, and the average threshold is shown.

Value

ggplot2 object

`qq.plot` *Quantile-Quantile (QQ) Plot*

Description

Inspect p-value inflation using a QQ plot

Usage

```
qq.plot(data, trait, models = NULL)
```

Arguments

`data`	Variable of class `GWASpoly.fitted`
`trait`	Trait name
`models`	Vector of model names (by default, all models plotted)

Value

ggplot2 object

`read.GWASpoly` *Read in marker and phenotype data*

Description

Read in marker and phenotype data

Usage

```
read.GWASpoly(ploidy, pheno.file, geno.file, format, n.traits, delim = ",")
```

Arguments

ploidy	Ploidy (e.g., 2 for diploid, 4 for tetraploid)
pheno.file	Name of the phenotype file
geno.file	Name of the genotype file
format	Format for the marker data. See details.
n.traits	Number of traits
delim	Character to indicate the delimiter in the data files (e.g., "," for csv, "\t" for tab-delimited)

Details

The first column of the phenotype file contains the genotype identifier, columns 2 through (n.traits + 1) contain trait values, and subsequent columns contain the levels (for factors) or numeric values (for covariates) of any fixed effects. The first three columns of the genotype file are (1) marker name, (2) chromosome, and (3) position. Optionally, columns 4 and 5 can be REF and ALT, respectively. Subsequent columns contain the marker data for each individual in the population. Marker data can be coded in one of three formats:

- "numeric": markers are coded based on the dosage of the alternate allele, taking on values between 0 and ploidy
- "AB": e.g., AAAB, ABBB for tetraploids
- "ACGT": e.g., AAAT, GGCC for tetraploids

Only bi-allelic markers are allowed. As of version 2.02 of the package, fractional values of dosage are allowed for the "numeric" format, with missing values imputed by the population mean for each marker. The fractional values are only used for the additive genetic model; for the other models, dosages are rounded to the nearest whole number. If the input allele dosages are whole numbers, then missing values are imputed with the population mode (most frequent value) for each marker.

Value

Variable of class GWASpoly.data

---

set.K *Set covariance matrix for polygenic effect*

Description

Set covariance matrix for polygenic effect

Usage

```
set.K(data, K = NULL, n.core = 1, LOCO = NULL)
```

Arguments

data	Output from read.GWASpoly
K	Optional: user-supplied matrix
n.core	Number of cores for parallel computing
LOCO	TRUE/FALSE, whether to use leave-one-chromosome-out

Details

When LOCO = TRUE, K is computed for each chromosome as $K=MM'$, where M is the centered genotype matrix (lines x markers), and scaled to have unit diagonal (the overall scaling is not important for GWAS). When LOCO = FALSE, a single K matrix is computed for all markers (this was the original behavior of the function). Alternatively, the user can supply their own positive semidefinite K, with row.names that match the genotype identifiers (this option cannot be used with LOCO).

Value

Variable of class GWASpoly.K

---

set.params *Set parameters*

Description

Set parameters

Usage

```
set.params(
  fixed = NULL,
  fixed.type = NULL,
  n.PC = 0,
  MAF = NULL,
  geno.freq = NULL,
  P3D = TRUE
)
```

Arguments

fixed	Vector of names of fixed effects
fixed.type	Vector of effect types ("numeric" or "factor"), corresponding to the effects listed in "fixed"
n.PC	Number of principal components to include as covariates
MAF	Minimum minor allele frequency
geno.freq	Maximum genotype frequency (after applying dominance relations)
P3D	TRUE/FALSE whether to use the P3D approximation (variance components not re-estimated for every marker)

Details

The list returned by the function should be passed to GWASpoly function.

Value

A list with the following components

fixed	Names of fixed effects
fixed.type	Types of fixed effects
n.PC	Number of principal components to include as covariates
min.MAF	Minimum minor allele frequency
max.geno.freq	Maximum genotype frequency (after applying dominance relations)
P3D	TRUE/FALSE whether to use the P3D approximation

---

set.threshold *Set the significance threshold*

Description

Set the significance threshold

Usage

```
set.threshold(
  data,
  method = "M.eff",
  level = 0.05,
  n.permute = 1000,
  n.core = 1
)
```

Arguments

data	Variable of class GWASpoly.fitted
method	One of the following: "M.eff","Bonferroni","FDR","permute"
level	Genome-wide false positive or false discovery rate (depending on method).
n.permute	Number of permutations for method "permute"
n.core	Number of cores to use for multicore processing

Details

The default method, "M.eff", is a Bonferroni-type correction but using an effective number of markers that accounts for LD between markers (Moskvina and Schmidt, 2008). The FDR method is based on version 1.30.0 of the qvalue package.

Value

Variable of class GWASpoly.thresh

References

Moskvina V, Schmidt KM (2008) On multiple-testing correction in genome-wide association studies. Genetic Epidemiology 32:567-573. doi:10.1002/gepi.20331

VCF2dosage *Prepare GWASpoly dosage input file from VCF file*

Description

Prepare GWASpoly dosage input file from VCF file

Usage

```
VCF2dosage(
  VCF.file,
  dosage.file,
  geno.code,
  ploidy,
  samples = NULL,
  min.DP = 1,
  max.missing,
  min.minor = 5
)
```

Arguments

VCF.file	VCF filename to read in
dosage.file	name of CSV file to output with allele dosage
geno.code	genotype code in the FORMAT field: either "GT" or "DS" (see Details)
ploidy	ploidy
samples	optional vector of sample names, to export subset of the population
min.DP	minimum per sample depth (DP) to export genotype. Default is 1, for no filtering.
max.missing	threshold for missing data per marker, as a proportion.
min.minor	minimum number of samples with the minor allele. Default is 5.

Details

The "GT" option for geno.code is the posterior maximum genotype (e.g., 0/0/1/1), while "DS" represents the posterior mean dosage of the alternate allele. (At present, only bi-allelic variants are supported.) The VCF file must conform to version 4.1 of the VCF specification or later.

write.GWASpoly *Write results to file*

Description

Write results to file

Usage

```
write.GWASpoly(data, trait, filename, what = "scores", delim = ",")
```

Arguments

`data`	Variable of class `GWASpoly.fitted`
`trait`	Trait name
`filename`	Filename
`what`	Either "scores" or "effects"
`delim`	Delimiter to use in the output file (default is comma)

Details

Score = -log10(p). Effect = marker effect (not available for the general and diplo-general models).

Index