Package: snpReady (via r-universe)

August 20, 2024

Version 0.9.7

Date 2018-04-11

Title Preparing Genotypic Datasets in Order to Run Genomic Analysis

Description Three functions to clean, summarize and prepare genomic datasets to Genome Selection and Genome Association analysis and to estimate population genetic parameters.

Depends Matrix, matrixcalc, stringr, rgl, impute

License GPL-3

URL https://github.com/italo-granato/snpReady

BugReports https://github.com/italo-granato/snpReady/issues

RoxygenNote 6.0.1

Suggests knitr, rmarkdown, reshape2

VignetteBuilder knitr

Repository https://italo-granato.r-universe.dev

RemoteUrl https://github.com/italo-granato/snpready

RemoteRef HEAD

RemoteSha ddb7eee245a2ab8933ddc9503f31cee40037232f

Contents

G.matrix *Estimation of Genomic Relationship Matrix*

Description

It generates four different types of Genomic Relationship Matrix (GRM)

Usage

```
G.matrix(M, method=c("VanRaden", "UAR", "UARadj", "GK"), format=c("wide", "long"),
         plot = FALSE)
```

Arguments

M	matrix. Matrix of markers in which n individuals are in rows and p markers in columns. This matrix do not need to be centered.
method	Method to built the GRM. Four methods are currently supported. "VanRaden" indicates the method proposed by Vanraden (2008) for additive genomic relationship and its counterpart for dominance genomic relationship. "UAR" and "UARadj" are methods proposed by Yang et al. (2010) for additive genomic relationship. "GK" represents the Gaussian kernel for additive genomic. See Details
format	Type of object to be returned. wide returns a $n \times n$ matrix. long returns the low diagonal from GRM as a table with 3 columns. See Details
plot	If TRUE, a graphical output is produced. See Details

Details

G.matrix provides four different types of relationship matrix. The VanRaden represents the relationship matrix estimated as proposed by Vanraden (2008):

$$G = \frac{XX'}{trace(XX')/n}$$

X is the centered marker matrix. For any marker locus i, $x_i = m_i - 2p_i$ where m_i is the vector of SNP genotypes coded as allele couting (0, 1 and 2).

UAR is the genomic relationship matrices proposed by Yang et al. (2010) and named as Unified Additive Relationship (UAR). This matrix is then obtained by

$$G_{UAR} = A_{jk} = \frac{1}{N}\sum_i A_{ijk} = \begin{cases} \frac{1}{N}\sum_i \frac{(x_{ij}-2p_i)(x_{ik}-2p_i)}{2p_i(1-p_i)}, j \neq k \\ 1 + \frac{1}{N}\sum_i \frac{x_{ij}^2(1+2p_i)x_{ij}+2p_i^2}{2p_i(1-p_i)}, j = k \end{cases}$$

where p_i is the allele frequency at SNP i and x_{ij} is the SNP genotype that takes a value of 0, 1 or 2 for the genotype of the j^{th} individual at SNP i. The same authors proposed an adjustment in the original UAR matrix (UARadj) to reduce the bias in estimation of variance in the relationship in causal loci. Thus:

$$G_{UARadj} = \begin{cases} \beta A_{jk}, j \neq k \\ 1 + \beta(A_{jk} - 1), j = k \end{cases}$$

where $\beta = 1 - fracc + 1/Nvar(A_{jk}$, c is a constant dependent on MAF of causal variants assumed as 0. `GK` represents the Gaussian kernel, obtained by

$$K(x_i, x_{i'}) = \frac{exp(-d^2_{ii'})}{quantile(d^2, 0.5)}$$

where $d^2_{ii'}$ is the square of euclidian distance between two individuals The `format` argument is the desired output format. For `"wide"`, the relationship output produced is in matrix format, with $n \times n$ dimension. If `"long"` is the chosen format, the inverse of the relationship matrix is computed and converted to a table. In this case, the low triangular part of the relationship matrix is changed to a table with three columns representing the respective rows, columns, and values (Used mainly by ASReml)

If the relationship matrix is not positive definite, a near positive definite matrix is created and solved, followed by a warning message.

Value

It returns the GRM. If the method is `VanRaden`, additive and dominance matrices are produced. Otherwise, only the additive form. If `plot` is `TRUE` a heat map of the pairwise relationship is save as pdf into the working directory . Also, a 3D plot with the three first principal components is generated.

References

Pérez-Elizalde, S.,Cuevas, J.; Pérez-Rodríguez, P.; Crossa, J. (2015) Selection of The Bandwidth Parameter in a Bayesian Kernel Regression Model for Genomic-Enabled Prediction. J Agr Biol Envir S, 20-4:512-532

Yang, J., Benyamin, B., McEvoy, B.P., et al (2010) Common SNPs explain a large proportion of the heritability for human height. Nature Genetics 42:565-569

VanRaden, P.M. (2008) Efficient Methods to Compute Genomic Predictions. Journal of Dairy Science, 91:4414-4423

Examples

```
#(1) Additive and dominance relationship matrix
data(maize.hyb)
x <- G.matrix(maize.hyb, method = "VanRaden", format = "wide")
A <- x$Ga
D <- x$Gd
```

`maize.hyb` *maize hybrids*

Description

50 hybrids of maize genotyped with 492 SNP markers

Format

A matrix with 50 rows and 492 columns, where hybrids are in rows and SNP markers in columns

Examples

```
#' data(maize.hyb)
```

`maize.line` *maize lines*

Description

A raw dataset of maize lines genotyped with 768 markers

Format

A matrix with 70656 observations on the following 4 variables.

- sample: identification of samples (name of individuals)
- marker: identification of SNP markers
- allele.1: Allele 1
- allele.2: Allele 2

Source

Lines genotyped from allogamous breeding laboratory - ESALQ/USP http://www.genetica.esalq.usp.br/alogamas/index2.html

Examples

```
data(maize.line)
## str(maize.line)
```

popgen *Population genetics from genomic data*

Description

Allows for estimating parameters of population genetics from genomic data. Besides, it also allows the estimate of same parameters considering subpopulations.

Usage

```
popgen(M, subgroups, plot = FALSE)
```

Arguments

M	Object of class `matrix`. A (non-empty) matrix of molecular markers, considering the count of reference alleles per loci (0, 1 or 2). Markers must be in columns and individuals in rows. Missing data should be assigned as `NA`
subgroups	A `vector` with information for subgroups or subpopulations.
plot	If `TRUE`, a graphical output is produced. See `details`

Details

The number of subgroups is defined by the user and accepts any data type (`character`, `integer` ...) to distinguish subpopulations. These two inputs must have the same sort for rows (genotypes).

Value

Two-level lists are returned (`whole` and `bygroup`), one with general information for markers and individuals and another by subgroups (if applicable).

For `whole`, a list containing estimates parameters for

$Markers For each marker it presents the allelic frequency (p and q), Minor Allele Frequency (MAF), expected heterozygosity (H_e), observed heterozygosity (H_o), Nei's Genetic Diversity (DG) and Polymorphism Informative Content(PIC), proportion of missing ($Miss$), χ^2 statistic for the Hardy-Weinberg equilibrium test and its pvalue

$Genotypes It presents observed heterozygosity (H_o) and coefficient of inbreeding (F_i) estimated as excess of homozygous relative to the expected (Keller et al. (2011))

$Population The same parameters as those for markers except PIC are returned for general population along with lower and upper boundaries

$Variability shows estimates of effective population size (Ne), additive (Va) and dominance (Vd) variances components, and a summary of number of groups, genotypes and markers

In the presence of `subgroups`, the same populational parameters are estimated considering each subpopulation accompanied by its exclusive and fixed alleles. Moreover, a list with the F-statistics (F_IT, F_IS and F_ST) for genotypes and markers are exhibited. For genotypes, it shows the statistics considering all subpopulations and a pairwise framework, and for markers loci, the parameters are presented only considering all subpopulations.

The plot produces a histogram for the estimates of MAF, GD, PIC and He considering the whole population and subpopulations, when it is available. Also, a heat map of the pairwise F_ST between populations is produced.

References

Weir, B.S. and C.C. Cockerham. (1984). Estimating F-Statistics for the Analysis of Population Structure. Evolution 38: 1358-1370. doi:10.2307/2408641.

Keller M.C., Visscher P.M., Goddard M.E. (2011) Quantification of inbreeding due to distant ancestors and its detection using dense single nucleotide polymorphism data. Genetics 189:237-249. doi: 10.1534/genetics.111.130922

Examples

```
# hybrid maize data
data(maize.hyb)
x <- popgen(maize.hyb)

# using subpopulations
PS<-c(rep(1,25), rep(2,25))
x <- popgen(maize.hyb, subgroups=PS)
```

---

raw.data *Preparation of genomic data*

Description

This function gets genomic data ready to be used in packages or softwares that perform genomic predictions

Usage

```
raw.data(data, frame = c("long","wide"), hapmap=NULL, base=TRUE, sweep.sample=1,
     call.rate=0.95, maf=0.05, imput=TRUE, imput.type=c("wright", "mean","knni"),
        outfile=c("012","-101","structure"), plot = FALSE)
```

Arguments

data	object of class matrix
frame	Format of genomic data to be imputed. Two formats are currently supported. "long" is used for objects with sample ID (1st column), marker ID (2nd column), fist allele (3rd column) and second allele (4th column). While "wide" is for objects with dimension $n \times m$ where markers must be in columns and individuals in rows
hapmap	matrix. Object with information on SNPs, chromosome and position
base	logical. Are genotypes coded as nitrogenous bases? if TRUE, data are converted to numeric. If FALSE, it follows to clean up

sweep.sample	numeric. Threshold for removing samples from data by missing rate. Samples with missing rate above the defined threshold are removed from dataset
call.rate	numeric. Threshold for removing marker by missing genotype rate. SNP with call rate below threshold are removed from dataset. Must be between 0, 1
maf	Threshold for removing SNP by minor allele frequency. Must be between 0, 1
imput	Should imputation of missing data be performed?. Default is TRUE
imput.type	Type of imputation. It can be "wright", "mean" or "knni". See details
outfile	character. Type of output to be produced. "012" outputs matrix coded as 0 to AA, 1 to Aa and 2 to aa. "-101" presents marker matrix coded as -1, 0 and 1 to aa, Aa and AA, respectively. "structure" returns a matrix suitable for STRUCTURE Software. For this, each individual is splited in two rows, one for each allele. Nitrogenous bases are then recoded to a specific number, so A is 1, C is 2, G is 3 and T is 4. This format is only acceptable if base are TRUE
plot	If TRUE, a graphical output of quality control is produced.

Details

The function allows flexible input of genomic data. Data might be in long format with 4 columns or in wide format where markers are in columns and individuals in rows. Both numeric and nitrogenous bases are accepted. Samples and markers can be eliminated based on missing data rate. Markers can also be eliminated based on the frequency of the minor allele. Three methods of imputation are currently implemented. One is carried out through combination of allele frequency and individual observed heterozygosity estimated from markers.

$$p(x_{ij}) = \begin{cases} 0 = (1-p_j)^2 + p_j(1-p_j)F_i \\ 1 = 2p_j(1-p_j) - 2p_j(1-p_j)F_i \\ 2 = p_j^2 + p_j(1-p_j)F_i \end{cases}$$

Hence, for missing values, genotypes are imputed based on their probability of occurrence. This probability depends both on genotype frequency and inbreeding of the individual a specific locus. The second method is based on mean of SNP. Thus, each missing point in a SNP j is replaced by mean of SNP j

$$x_{ij} = 2p_j$$

The "knni" imputes missing markers using the mean of the k-nearest markers. Nearest markers are found by computing the Euclidian distance between markers. If you use this option, please refer to the package impute (Hastie et al. 2017) in publications.

Value

Returns a properly coded marker matrix output and a report specifying which individuals are removed by sweep.sample and which markers are removed by "call.rate" and maf. Also, a plot with proportion of removed markers and imputed data, for each chromosome, when the map is included, is produced when plot is TRUE

Examples

```
data(maize.line)
M <- as.matrix(maize.line)
mrc <- raw.data(M, frame="long", base=TRUE, sweep.sample= 0.8,
         call.rate=0.95, maf=0.05, imput=FALSE, outfile="-101")
```

Index