

Package ‘IASbreeding’

August 21, 2024

Type Package

Title Intelligent Animal Solutions of breeding

Version 0.1.0

Depends R (>= 3.5.0)

Author Wentao Cai <caiwentao@caas.cn>

Maintainer Wentao Cai <caiwentao@caas.cn>

Description IASbreeding can calculate kinship matrix, variance component analysis using linear mixed model, breeding value using BLUP.

License GPL-3

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Imports Rcpp (>= 0.12.16),
RcppEigen

LinkingTo Rcpp, RcppEigen

R topics documented:

IASbreeding-package	2
Akin	2
GetEBV	3
getEBV_cpp	4
Gkin	4
gkin_cpp	5
IASBLUP	5
IASREML	6
iasreml_cpp	7
pedcheck	8
prepareped	8
read_bed	9
read_bed2_cpp	9

Index	11
--------------	-----------

IASbreeding-package *A short title line describing what the package does*

Description

A more detailed description of what the package does. A length of about one to five lines is recommended.

Details

This section should provide a more detailed overview of how to use the package, including the most important functions.

Author(s)

Your Name, email optional.

Maintainer: Your Name <your@email.com>

References

This optional section can contain literature or other references for background information.

See Also

Optional links to other man pages

Examples

```
## Not run:
## Optional simple examples of the most important functions
## These can be in \dontrun{} and \donttest{} blocks.

## End(Not run)
```

Akin *Get kinship from pedigree*

Description

Get kinship from three columns pedigree (ID, Sire, Dam)

Usage

```
Akin(data = NULL)
```

Arguments

data pedigree

Details

Title

Value

pedigree kinship

Examples

```
newped<-data.frame(ID=c("E","G","F","A","B","C","D"),Sire=c("0","0","0","E","G","E","B"),
                   Dam=c("0","0","0","F","A","A","C"))
A<-Akin(newped)
```

GetEBV	<i>Calculate Breeding value</i>
--------	---------------------------------

Description

Calculate heritability and breeding value using AI-REML

Usage

```
GetEBV(formula, random, kinship, data, h2 = 0.3)
```

Arguments

formula	y~ 1+fixed_effects
random	column name of random effect
kinship	kinship(pedigree kinship A or genomic kinship G)
data	phenotype data with both phenoytpe and covariate
h2	the heritability of this trait, default: 0.3

Details

Title

Value

Variance component, Heritability.

Examples

```
phe_file <- system.file("extdata", "Phenotype.txt", package = "IASbreeding", mustWork = TRUE)
pheno <- read.table(phe_file,header=TRUE)
ped_file <- system.file("extdata", "Pedigree.txt", package = "IASbreeding", mustWork = TRUE)
ped <- read.table(ped_file,header=TRUE)
newped<-pedcheck(ped)
A<-Akin(newped)
Results <- GetEBV(BW56 ~ 1 + Sex + Condition, random = "ID", kinship = A, data = pheno, h2 = 0.3)
```

getEBV_cpp	<i>Estimate Breeding Values (EBV)</i>
------------	---------------------------------------

Description

This function calculates the estimated breeding values (EBV) for individuals based on a given kinship matrix, fixed effects, and observed phenotypes.

Usage

```
getEBV_cpp(Kin, X, y, H2)
```

Arguments

Kin	A numeric matrix representing the kinship matrix.
X	A numeric matrix representing fixed effects.
y	A numeric matrix (or vector) representing the observed phenotypes.
H2	A numeric value representing the heritability (H2).

Value

A list containing the fixed effects and the random effects (EBV).

Gkin	<i>Get kinship from genotype</i>
------	----------------------------------

Description

Get kinship from genotype

Usage

```
Gkin(Genotype)
```

Arguments

Genotype	genotype matrix
----------	-----------------

Details

Title

Value

Genomic kinship

Examples

```
bed_file <- system.file("extdata", "Genotype.bed", package = "IASbreeding", mustWork = TRUE)
file_prefix <- sub("\\.bed$", "", bed_file)
Genotype <- read_bed(file_prefix)
kin<-Gkin(Genotype)
```

gkin_cpp

*Compute the G Matrix from Genotype Data***Description**

This function computes the G matrix (genomic relationship matrix) from a given genotype matrix. The function handles missing data by replacing missing values with the column mean and normalizes the resulting matrix.

Usage

```
gkin_cpp(matrix)
```

Arguments

matrix	M numeric matrix where each row represents an individual and each column represents a genetic marker. The values in the matrix are expected to be 0, 1, or 2, corresponding to the number of minor alleles. Missing values (NA) are allowed and will be replaced by the mean of the corresponding column.
--------	---

Value

M numeric matrix representing the G matrix, which is the genomic relationship matrix. The matrix is normalized based on allele frequencies.

IASBLUP

*Calculate Heritability and Breeding Value***Description**

Calculates heritability and breeding values using AI-REML.

Usage

```
IASBLUP(  
  formula,  
  random,  
  kinship,  
  data,  
  init = 0.3,  
  max_iter = 30,  
  cc = 1e-05,  
  allblup = FALSE  
)
```

Arguments

formula	A formula specifying the fixed effects (e.g., $y \sim 1 + \text{fixed_effects}$).
random	A character string specifying the column name of the random effect.
kinship	A matrix or data frame representing kinship (pedigree kinship A or genomic kinship G).
data	A data frame containing phenotype and covariate information.
init	The pre-estimated heritability (default: 0.3).
max_iter	The maximum number of iterations for AI-REML (default: 30).
cc	The threshold for changed variance to stop iteration (default: $1.0e-5$).
allblup	Logical. If TRUE, predict all animals in the pedigree (default: FALSE). Note that this may use a large amount of memory.

Details

Title

Value

A list containing variance components, heritability estimates, and breeding values.

Examples

```
phe_file <- system.file("extdata", "Phenotype.txt", package = "IASbreeding", mustWork = TRUE)
pheno <- read.table(phe_file, header=TRUE)
ped_file <- system.file("extdata", "Pedigree.txt", package = "IASbreeding", mustWork = TRUE)
ped <- read.table(ped_file, header=TRUE)
newped <- pedcheck(ped)
A <- Akin(newped)
Results <- IASBLUP(BW56 ~ 1 + Sex + Condition,
  random = "ID", kinship = A, max_iter = 20, data = pheno)
```

IASREML

Calculate heritability and variance component

Description

Calculate heritability and breeding value using AI-REML

Usage

```
IASREML(formula, random, kinship, data, init = 0.3, max_iter = 30, cc = 1e-05)
```

Arguments

formula	$y \sim 1 + \text{fixed_effects}$
random	column name of random effect
kinship	kinship(pedigree kinship A or genomic kinship G)
data	phenotype data with both phenotype and covariate
init	the pre-estimated heritability, default: 30.
max_iter	the maximum iteration for AI-REML, default: $1.0e-5$.
cc	the threshold of changed variance when stop iteration, default: $1.0e-5$.

Details

Title

Value

Variance component, Heritability.

Examples

```
phe_file <- system.file("extdata", "Phenotype.txt", package = "IASbreeding", mustWork = TRUE)
pheno <- read.table(phe_file, header=TRUE)
ped_file <- system.file("extdata", "Pedigree.txt", package = "IASbreeding", mustWork = TRUE)
ped <- read.table(ped_file, header=TRUE)
newped <- pedcheck(ped)
A <- Akin(newped)
Results <- IASREML(BW56 ~ 1 + Sex + Condition, random = "ID",
  kinship = A, max_iter = 20, data = pheno)
```

iasreml_cpp	<i>Restricted Maximum Likelihood (REML) Estimation</i>
-------------	--

Description

This function performs the restricted maximum likelihood (REML) estimation to obtain variance components and heritability estimates.

Usage

```
iasreml_cpp(ZKZ, X, y, int_h2, cc, max_iter)
```

Arguments

ZKZ	A numeric matrix representing the kinship matrix multiplied by Z (design matrix).
X	A numeric matrix representing fixed effects.
y	A numeric matrix (or vector) representing the observed phenotypes.
int_h2	A numeric value representing the initial heritability estimate.
cc	A numeric value for the convergence criterion.
max_iter	An integer indicating the maximum number of iterations.

Value

A list containing the variance components, heritability estimates, and their standard errors.

pedcheck	<i>Check pedigree</i>
----------	-----------------------

Description

Check of pedigree and rebuild the unlisted individuals.

Usage

```
pedcheck(ped)
```

Arguments

ped	pedigree input file
-----	---------------------

Value

The checked pedigree

Examples

```
ped_file <- system.file("extdata", "Pedigree.txt", package = "IASbreeding", mustWork = TRUE)
ped <- read.table(ped_file, header=TRUE)
newped<-pedcheck(ped)
```

prepareped	<i>prepare pedigree</i>
------------	-------------------------

Description

renumber pedigree.

Usage

```
prepareped(data = NULL, n.max = 50)
```

Arguments

data	input file of checked pedigree
n.max	the maximum iteration for reorder sire and dam

Details

Title

Value

renumber pedigree

Examples

```
ped_test<-data.frame(ID=c("E","F","G","A","B","C","D"),Sire=c("0","0","E","E","G","E","B"),
                    Dam=c("0","0","0","F","A","A","C"))
ped_test1<-prepareped(ped_test)
```

read_bed	<i>Read genotype from PLINK binary file</i>
----------	---

Description

Read genotype from PLINK binary file (bed, bim and fam)

Usage

```
read_bed(file_prefix, verbose = TRUE)
```

Arguments

file_prefix	The prefix of genotype file.
verbose	remind log of reading bed file.

Details

Title

Value

Genotype matrix

Examples

```
bed_file <- system.file("extdata", "Genotype.bed", package = "IASbreeding", mustWork = TRUE)
file_prefix <- sub("\\.bed$", "", bed_file)
Genotype <-read_bed(file_prefix)
```

read_bed2_cpp	<i>Fast read Plink binary genotype file (bed) This function read the genotype from given Plink binary files with bim, fam and bed suffix.</i>
---------------	---

Description

Fast read Plink binary genotype file (bed) This function read the genotype from given Plink binary files with bim, fam and bed suffix.

Usage

```
read_bed2_cpp(file, m_loci, n_ind)
```

Arguments

<code>file</code>	where bed file of PLINK.
<code>m_loci</code>	represents number of genetic markers.
<code>n_ind</code>	represents number of individuals.

Value

genotype matrix with each row represents an individual and each column represents a genetic marker.

Index

* **package**

IASbreeding-package, [2](#)

Akin, [2](#)

GetEBV, [3](#)

getEBV_cpp, [4](#)

Gkin, [4](#)

gkin_cpp, [5](#)

IASBLUP, [5](#)

IASbreeding (IASbreeding-package), [2](#)

IASbreeding-package, [2](#)

IASREML, [6](#)

iasreml_cpp, [7](#)

pedcheck, [8](#)

prepareped, [8](#)

read_bed, [9](#)

read_bed2_cpp, [9](#)