Package 'GOVS'

June 6, 2021

Title Genome optimization via virtual simulation

Version 1.0

Author Qian Cheng, Shuqin Jiang and Xiangfeng Wang

Maintainer Qian Cheng <qchengray@gmail.edu>

Depends R (>= 3.6.0), rrBLUP, lsmeans, readr, pbapply, ggplot2, pheatmap

Imports grid, stats, scales

Description GOVS is an integrative R package for maize breeding that streamlines genome optimization via virtual simulation achieve guidance of lines selection and population development. GOVS describes a promising strategy that can help breeders to select materials in a purposeful and directional manner, with the purpose of enabling breeders to combine others technologies with rapidly genetic gain benefit by genome optimization.

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1

License GPL-3

R topics documented:

bins . . . 2
binsInfo . . . 2
binsPlot . . . 2
extractGenome . . . 3
GBLUP . . . 5
gcaCompute . . . 6
genomeOptimization . . . 7
GOVS . . . 8
IBDConstruct . . . 9
IBDTestData . . . 11
mosaicPlot . . . 12
MZ . . . 13
phe . . . 13
reviseFunc . . . 13
scaCompute . . . 14
SNPrrBLUP . . . 15
statDesign . . . 16
transHapmap2numeric . . . 18

Index 19

`bins` *bins data for GOVS*

Description

bins data of 3515 bins for GOVS.

Usage

```
data(bins)
```

References

Liu H J, Wang X, Xiao Y, et al. CUBIC: an atlas of genetic architecture promises directed maize improvement[J]. Genome biology, 2020, 21(1): 1-17.

`binsInfo` *bins information data for GOVS*

Description

bins information data of 3515 bins for GOVS.

Usage

```
data(binsInfo)
```

References

Liu H J, Wang X, Xiao Y, et al. CUBIC: an atlas of genetic architecture promises directed maize improvement[J]. Genome biology, 2020, 21(1): 1-17.

`binsPlot` *Visualization of IBD map results*

Description

Visualization of IBD map results

Usage

```
binsPlot(IBDRes,color,parentInfo,parentNum)
```

Arguments

IBDRes	The results of IBDConstruct, see IBDConstruct.
color	A named vector for defining color of parents.
parentInfo	A named vector for defining label of parents.
parentNum	The number of parents.

Author(s)

Qian Cheng, Shuqin Jiang, Xiangfeng Wang

See Also

IBDConstruct

Examples

```
## load example data
data(IBDTestData)

## compute rou from genetic position
rou = IBDTestData$posGenetic
rou = diff(rou)
rou = ifelse(rou<0,0,rou)

## constract IBD map of chr10 for one progeny
IBDRes <- IBDConstruct(snpParents = IBDTestData$snpParents,
markerInfo = IBDTestData$markerInfo,
snpProgeny = IBDTestData$snpProgeny,q = 0.97,G = 9,rou = rou)

## plot
# color
color <- c("#DA053F","#FC0393","#C50F84","#D870D4","#DCA0DC","#4A0380",
           "#9271D9","#0414FB","#2792FC","#4883B2","#2CFFFE","#138B8A",
           "#42B373","#9BFB9C","#84FF2F","#566B32","#FED62D","#FD8A21",
           "#F87E75","#B01D26","#7E0006","#A9A9A9","#FFFE34","#FEBFCB")
names(color) <- 1:24

# parent label
parentInfo <- c("5237","E28","Q1261","CHANG7-2","DAN340","HUANGC","HYS",
                "HZS","TY4","ZI330","ZONG3","LX9801","XI502","81515",
                "F349","H21","JI853","JI53","LV28","YUANFH","SHUANG741",
                "K12","NX110","ZONG31")
names(parentInfo) <- 1:24

# plot
binsPlot(IBDRes,color,parentInfo,24)
```

---

extractGenome *Extraction & assembly module*

Description

Extracting genome fragment from candidates based the results of genome optimization and then assembling all fragments so that produce optimized genome(virtual genome).

Usage

```
extractGenome(hmp,binsInfo,ID = NULL,designInfo,output = NULL,
              bins,extractContent ="Genotype")
```

Arguments

hmp	The genetic data in hapmap format.
binsInfo	Data frame, including bins index, start, end, length of bins locus.
ID	A character array regarding sample IDs for 'hmp', if NULL, the 'hmp' data must involve header.
designInfo	Outputs of genomeOptimization, a matrix consists of sample IDs regarding the fragment source among condidates at each bin locus.
output	The prefix of output files regarding the assembled genome.
bins	Results of IBD analysis (bins matrix), each row represents a bin fragment as well as each column represents each sample.
extractContent	Character, the content of virtual genome, "Bin" for bin source well "Genotype" for genetic data, default "Genotype".

Value

A matrix involves of three optimal genomes (optimal: max, moderate: med and poor: min) If output is defined, the assembled genome will be written to files with suffix ".G" or ".bin".

Author(s)

Qian Cheng, Shuqin Jiang, Xiangfeng Wang

See Also

GOVS

Examples

```
## Not run !
## load test data
# Phenotypic data:
data(phe)
# bins data:
data(bins)
# genomic data:
data(MZ)
# bins info
data(binsInfo)

# 1.run example for EW trait (write result files to local)
genomeOptimization(pheno = phe,trait = "EW",bins = bins,output = "MZ_test_1404")

# 2.run example for EW trait (get the results of genomeOptimization in R)
```

```
GO_Res <- genomeOptimization(pheno = phe,trait = "EW",bins = bins,output = NULL)

# 3.extract and assemble virtual genome (genomic genome)
Vgenome <- extractGenome(hmp = MZ,binsInfo = binsInfo,designInfo = GO_Res$overall)

# 4.extract and assemble virtual genome (bin source genome)
Vbin <- extractGenome(hmp = MZ,binsInfo = binsInfo,designInfo = GO_Res$overall,
                      bins = bins,extractContent = "BINsource")
```

GBLUP *Genomic prediction by GBLUP (kinship matrix input)*

Description

Genotpye-to-phenotype prediciton via genomic best linear unbiased prediction (GBLUP) model. The inputs is genotypes.

Usage

```
GBLUP(amat,y,idx1,idx2,fix = NULL,model = FALSE)
```

Arguments

amat	Additive relationship matrix, which compute from genetic matrix (See A.mat).
y	An numeric array of phenotype.
idx1	An array of index for training set.
idx2	An array of index for testing (predicted) set.
fix	A matrix containing other variables as fixed effects in mixed model.
model	Boolean, if output the model, default FALSE.

Value

A array for predicted value.

Author(s)

Qian Cheng, Shuqin Jiang, Xiangfeng Wang

References

Endelman, J.B. 2011. Ridge regression and other kernels for genomic selection with R package rrBLUP. Plant Genome 4:250-255. doi: 10.3835/plantgenome2011.08.0024

See Also

rrBLUP:
A.mat:
kin.blup

Examples

```
## Not run!
## load hapmap data (genomic data) of MZ hybrids
data(MZ)

## load phenotypic data of MZ hybrids
data(phe)

## pre-process for G2P prediction
rownames(MZ) <- MZ[,1]
MZ <- MZ[,-c(1:11)]
MZ.t <- t(MZ)

## conversion
MZ.n <- transHapmap2numeric(MZ.t)
dim(MZ.t)
## Additive relationship matrix compute
library(rrBLUP)
amat <- A.mat(MZ.n)

## prediction
idx1 <- sample(1:1404,1000)
idx2 <- setdiff(1:1404,idx1)
predRes <- GBLUP(amat,phe$EW,idx1,idx2,fix = NULL,model = FALSE)
```

`gcaCompute` *GCA computation*

Description

Calculate parental general combining ability (GCA) based F1 phenotypic values.

Usage

```
gcaCompute(phe_df,which,trait)
```

Arguments

`phe_df`	Phenotypic data frame, row represents F1 combination and includes the paternal information in columns.
`which`	The column index of male or female to compute parternal or maternal GCA.
`trait`	A character string to define which trait GCA will be computed, this function support two or more phenotypic GCA be computed at the same time.

Value

A data frame involves of target trait GCA.

Author(s)

Qian Cheng, Shuqin Jiang, Xiangfeng Wang

Examples

```
## simulation data
df <- data.frame(seqname = paste0("sample",1:500),
                 female_ID = sample(paste0("female",1:20), size = 500, replace = TRUE),
                 male_ID = sample(paste0("male",1:10), size = 500, replace = TRUE),
                 trait = rnorm(500, mean = 225, sd = 20))
plot(density(df$trait))

## compute GCA of female lines
gcaRes <- gcaCompute(df,which = "female_ID",trait = "trait")
```

---

genomeOptimization *Genome optimization function*

Description

Virtual genome optimization based IBD(bins) data

Usage

```
genomeOptimization(pheno,bins,trait,output = NULL)
```

Arguments

pheno	Phenotypic data frame, the first column describes sample names.
bins	Results of IBD analysis (bins matrix), each row represents a bin fragment as well as each column represents each sample.
trait	The names of interest trait (The trait must be included in 'Pheno' data frame
output	The prefix of output files regarding the scheme of virtual genome.

Value

Genome optimization results

max	Details for "optimal" simulation
med	Details for "moderate" simulation
min	Details for "poor" simulation
overall	Lines' names for genome optimization results of three simulated genome. Used for extractGenome
pvalue	P-value for ANOVA

If output is defined, all above five elements will be written to five files with suffix ".max", ".med",".min", ".merge",".pvalue".

Author(s)

Qian Cheng, Shuqin Jiang, Xiangfeng Wang

See Also

GOVS

Examples

```
## Not run !
## load test data
# Phenotypic data:
data(phe)
# bins data:
data(bins)

# 1.run example for EW trait (write result files to local)
genomeOptimization(pheno = phe,trait = "EW",bins = bins,output = "MZ_test_1404")

# 2.run example for EW trait (get the results of genomeOptimization in R)
GO_Res <- genomeOptimization(pheno = phe,trait = "EW",bins = bins,output = NULL)
```

GOVS *Genome optimization via virtual simulation*

Description

One-stop function for a complete progress of genome optimization.

Usage

```
GOVS(hmp,ID = NULL,pheno,trait,bins,binsInfo,which = "max",
     output = NULL,module = "DES",designInfo,extractContent = "Genotype")
```

Arguments

hmp	The genetic data in hapmap format.
ID	A character array regarding sample IDs for "hmp", if NULL, the "hmp" data must involve header.
pheno	Phenotypic data frame, the first column describes sample IDs.
trait	The names of interest trait (The trait must be included in "Pheno" data frame).
bins	Results of IBD analysis (bins matrix), each row represents a bin fragment as well as each column represents each sample.
binsInfo	Data frame, including bins index, start, end, length of bins locus.
which	A character, defining which virtual genome used for statistics, default "max".
output	The prefix of output files.
module	Character represents the module combination for analysis, default "DES", "D" for genome optimization module, "E" for extraction & assembly module. "S" for statistic module. "D","E","S","DE","DES" and "ES" are alternative for different module combinations. Note that different combination need different essential inputs.
designInfo	Data frame, the results of genome optimization module, it's necessary for "ES" and "S" module.
extractContent	Character, the content of virtual genome, "Bin" for bin source well "Genotype" for genetic data, default "Genotype".

Value

A list regarding defined module.
If output is defined, all files will be written to file with prefix of "output".

Details see:
genomeOptimization
extractGenome
statDesign

Author(s)

Qian Cheng, Shuqin Jiang, Xiangfeng Wang

See Also

genomeOptimization
extractGenome
statDesign

Examples

```
## Not run !
## load test data
# Phenotypic data:
data(phe)
# genomic data:
data(MZ)
# bins data:
data(bins)
# bins information:
data(binsInfo)
# example for one-stop solution for GOVS
GOVS_res <- GOVS(MZ,pheno = phe,trait = "EW",which = "max",bins = bins,
                 binsInfo = binsInfo,module = "DES")
```

IBDConstruct *Construct IBD map*

Description

A IBD map was constructed of contributions from the parents onto the progeny lines using a hidden Markov model (HMM).

Usage

```
IBDConstruct(snpParents,snpProgeny,markerInfo,q,
             rou,G,threshold = NULL,omit = T)
```

Arguments

snpParents	A matrix for the parents' genotype, lines in column and marker in row.
snpProgeny	An array for the progeny' genotype, marker number must equal to snpParents.
markerInfo	A matrix or dataframe with four cols(marker ID, allele, chromsome and physical position) regarding genotypic information.
q	The quality of sequencing, range 0 to 1 to define the quality of marker.
rou	Correlations between any pairs of flanking markers, that estimated with the offspring-LD level after corrected by parent-LD level, it can be obtained by genetic location.
G	Generations that the offsprings decented from the parents.
threshold	The threshold of posterior.
omit	Whether to omit untraceable segments, default True.

Details

The details see:
https://genomebiology.biomedcentral.com/articles/10.1186/s13059-020-1930-x#MOESM14
CUBIC: an atlas of genetic architecture promises directed maize improvement
https://github.com/heroalone/HMM-IBD

Value

A list regarding Constructed bin map.

bin	Results of IBD analysis (bins matrix), each row represents a bin fragment.
binsInfo	Data frame, including bins index, start, end, length of bins locus.

References

Liu H J, Wang X, Xiao Y, et al. CUBIC: an atlas of genetic architecture promises directed maize improvement[J]. Genome biology, 2020, 21(1): 1-17.

https://github.com/heroalone/HMM-IBD

Examples

```
## load example data
data(IBDTestData)
## compute rou from genetic position
rou = IBDTestData$posGenetic
rou = diff(rou)
rou = ifelse(rou<0,0,rou)
## constract IBD map of chr10 for one progeny
IBDRes <- IBDConstruct(snpParents = IBDTestData$snpParents,
markerInfo = IBDTestData$markerInfo,
snpProgeny = IBDTestData$snpProgeny,q = 0.97,G = 9,rou = rou)
```

IBDTestData *Example data for IBD map construction*

Description

Example data for IBD map construction

Usage

```
data("IBDTestData")
```

Format

A list:

`snpParents` a matrix regarding genotype of parents.

`markerInfo` a matrix or dataframe with four cols regarding information of markers.

`snpProgeny` a character vector involes genotypes of progeny.

`posGenetic` a numeric vector regarding genetic positions.

Details

Details see:
Liu H J, Wang X, Xiao Y, et al. CUBIC: an atlas of genetic architecture promises directed maize improvement[J]. Genome biology, 2020, 21(1): 1-17.

https://github.com/heroalone/HMM-IBD

Source

Liu H J, Wang X, Xiao Y, et al. CUBIC: an atlas of genetic architecture promises directed maize improvement[J]. Genome biology, 2020, 21(1): 1-17.

https://github.com/heroalone/HMM-IBD

References

Liu H J, Wang X, Xiao Y, et al. CUBIC: an atlas of genetic architecture promises directed maize improvement[J]. Genome biology, 2020, 21(1): 1-17.

https://github.com/heroalone/HMM-IBD

Examples

```
data(IBDTestData)
```

mosaicPlot *Visualization of overall bins data*

Description

Mosasic plot for overall bins data

Usage

```
mosaicPlot(bins,binsInfo,chr,resolution = 500,list,parentNum = 24,color,
           clust = T,methods = "ward.D2",
           dist_method = "euclidean")
```

Arguments

bins	Results of IBD analysis (bins matrix), each row represents a bin fragment as well as each column represents each sample.
binsInfo	Data frame, including bins index, start, end, length of bins locus.
chr	Which chromesome will be used to plot mosaic.
resolution	To set the resolution of mosaic plot, default 500.
list	The names of lines to visualize mosaic plot.
parentNum	The number of parent, if color not defined, this parameter is used to auto generate color palette.
color	A array to define color palette.
clust	Boolean values determining if lines should be hclust object.
methods	Clustering method used. Accepts the same values as hclust.
dist_method	The distance measure to be used for clustering. See dist

Author(s)

Qian Cheng, Shuqin Jiang, Xiangfeng Wang

See Also

bins

Examples

```
## load data
data(bins)
data(binsInfo)
## color
color <- c("#DA053F","#FC0393","#C50F84","#D870D4","#DCA0DC","#4A0380",
           "#9271D9","#0414FB","#2792FC","#4883B2","#2CFFFE","#138B8A",
           "#42B373","#9BFB9C","#84FF2F","#566B32","#FED62D","#FD8A21",
           "#F87E75","#B01D26","#7E0006","#A9A9A9","#FFFE34","#FEBFCB")

mosaicPlot(bins = bins,binsInfo = binsInfo,chr = 1,resolution = 500,
                 color = color,
                 list = colnames(bins)[1:200])
```

MZ *hapmap data (genotypic data) for GOVS*

Description

hapmap data of Zheng58 F1 progeny (genotypic data) for GOVS.

Usage

```
data(MZ)
```

References

Xiao Y, Jiang S, Cheng Q, et al. The genetic mechanism of heterosis utilization in maize improvement[J]. Genome Biology, 2021, 22(1): 1-29.

phe *Phenotypic data for GOVS*

Description

Phenotypic data of Zheng58 F1 progeny for GOVS.

Usage

```
data(phe)
```

References

Xiao Y, Jiang S, Cheng Q, et al. The genetic mechanism of heterosis utilization in maize improvement[J]. Genome Biology, 2021, 22(1): 1-29.

reviseFunc *Data correct function*

Description

Scale two sets of data to a uniform distribution.

Usage

```
reviseFunc(ori,aim,cut = 10,sample_names)
```

Arguments

ori	A numeric array, as reference for correction.
aim	A numeric array, which is the object to implement correction.
cut	Number of intervals to cut, default 10.
sample_names	A character array consits of the names of 'aim'.

Author(s)

Qian Cheng, Shuqin Jiang, Xiangfeng Wang

Examples

```
## Not run!
## simulation data
df <- data.frame(seqname = paste0("sample",1:500),
                 female_ID = sample(paste0("female",1:20), size = 500,
                 replace = TRUE),
                 male_ID = sample(paste0("male",1:10), size = 500,
                 replace = TRUE),
                 trait = rnorm(500, mean = 225, sd = 20))
plot(density(df$trait))

## correct data distribution
df1 <- data.frame(seqname = paste0("sample",1:500),
                  trait = rnorm(500, mean = 225, sd = 20))
df2 <- data.frame(seqname = paste0("sample",1:300),
                  trait = rnorm(300, mean = 170, sd = 30))

## comparison of original distribution
plot(density(df1$trait),xlim = c(30,300),xlab = "Value range",main = "")
lines(density(df2$trait),col = "red")
legend(30,0.02,legend = c("df1","df2"),col = c("black","red"),lty = c(1,1))

## scale distribution
correct_df2 <- reviseFunc(sample_names = df2$seqname,ori = df1$trait,
                          aim = df2$trait,cut = 500)

## comparison of corrected distribution and original distribution
plot(density(df1$trait),xlim = c(30,300),xlab = "Value range",main = "")
lines(density(df2$trait),col = "red")
lines(density(correct_df2),col = "blue")
legend(30,0.02,legend = c("df1","df2","correct_df2"),
       col = c("black","red","blue"),lty = c(1,1,1))
```

`scaCompute` *SCA computation*

Description

Calculate hybrid special combining ability (SCA) based F1 phenotypic values.

Usage

```
scaCompute(phe_df,which_male,which_female,trait,seqname)
```

Arguments

`phe_df`	Phenotypic data frame, row represents F1 combination and includes the paternal information in columns.
`which_male`	The column index of paternal IDs.

which_female	The column index of maternal IDs.
trait	A character string to define which trait GCA will be computed, this function support two or more phenotypic GCA be computed at the same time.
seqname	A character array regarding hybrid IDs

Value

A data frame involves of target trait SCA.

Author(s)

Qian Cheng, Shuqin Jiang, Xiangfeng Wang

See Also

gcaCompute

Examples

```
## simulation data
df <- data.frame(seqname = paste0("sample",1:500),
                 female_ID = sample(paste0("female",1:20), size = 500, replace = TRUE),
                 male_ID = sample(paste0("male",1:10), size = 500, replace = TRUE),
                 trait = rnorm(500, mean = 225, sd = 20))
plot(density(df$trait))

## compute SCA of hybrid lines
scaRes <- scaCompute(df,which_female = "female_ID",which_male = "male_ID",
                     trait = "trait",seqname = df$seqname)
```

SNPrrBLUP *Genomic prediction by rrBLUP*

Description

Genotpye-to-phenotype prediciton via ridge regression best linear unbiased prediction (rrBLUP) model. The inputs is genotypes.

Usage

```
SNPrrBLUP(x,y,idx1,idx2,fix = NULL,model = FALSE)
```

Arguments

x	Genotypic matrix in numberic format (See transHapmap2numeric), row represents sample well column represents feature (SNP).
y	An numeric array of phenotype.
idx1	An array of index for training set.
idx2	An array of index for testing (predicted) set.
model	Boolean, if output the model, default FALSE.
fix	A matrix containing other variables as fixed effects in mixed model.

Value

A array for predicted value

Author(s)

Qian Cheng, Shuqin Jiang, Xiangfeng Wang

References

Endelman, J.B. 2011. Ridge regression and other kernels for genomic selection with R package rrBLUP. Plant Genome 4:250-255. doi: 10.3835/plantgenome2011.08.0024

See Also

transHapmap2numeric

rrBLUP:
mixed.solve

Examples

```
## Not run!
## load hapmap data (genomic data) of MZ hybrids
data(MZ)

## load phenotypic data of MZ hybrids
data(phe)

## pre-process for G2P prediction
rownames(MZ) <- MZ[,1]
MZ <- MZ[,-c(1:11)]
MZ.t <- t(MZ)

## conversion
MZ.n <- transHapmap2numeric(MZ.t)
dim(MZ.t)

## prediction
idx1 <- sample(1:1404,1000)
idx2 <- setdiff(1:1404,idx1)
predRes <- SNPrrBLUP(MZ.n,phe$EW,idx1,idx2,fix = NULL,model = FALSE)
```

---

statDesign *Statistic of optimal genome*

Description

Statistic summary for analysis of the contribution of all candidates to optimal genome. The results directly guide the lines selection and population improvement route.

Usage

```
statDesign(designInfo,which = "max",binsInfo,pheno,trait,output = NULL)
```

Arguments

designInfo	Outputs of genomeOptimization, a matrix consists of sample IDs regarding the fragment source among condidates at each bin locus.
binsInfo	Data frame, including bins index, start, end, length of bins locus.
which	A character, defining which virtual genome used for statistics, default "max".
pheno	Phenotypic data frame, the first column describes sample names.
trait	The names of interest trait (The trait must be included in 'Pheno' data frame).
output	The prefix of output files regarding the summary and statistic infomation via the process of genome optimization.

Value

A data frame regarding statistics results of virtual simulation.

Lines	Lines
Bins(#)	The number (#) of bins that a line contributed to the simulated genome.
Bins(%)	The number of bins that a line contributed accounting for the proportion (%) of simulated genome.
Fragments(%)	The total length of genomic fragments that a line contributed accounting for the proportion (%) of simulated genome.
phenotype	The phenotypic value of the corresponding lines or their offspring
phenotypeRank	The phenotype rank
Cumulative(%)	The cumulative percentage of fragments contributing to the simulated genome.

If output is defined, the statistics results will be written to file with suffix "statRes.csv".

Author(s)

Qian Cheng, Shuqin Jiang, Xiangfeng Wang

See Also

GOVS

Examples

```
## Not run !
## load test data
# Phenotypic data:
data(phe)
# bins data:
data(bins)

# 1.run example for EW trait (write result files to local)
genomeOptimization(pheno = phe,trait = "EW",bins = bins,output = "MZ_test_1404")

# 2.run example for EW trait (get the results of genomeOptimization in R)
```

```
GO_res <- genomeOptimization(pheno = phe,trait = "EW",bins = bins,output = NULL)

# 3.Statistics for genome optimization
sta_res <- statDesign(designInfo = GO_Res$overall,binInfo = binsInfo,pheno = phe,
                      trait = "EW")
```

`transHapmap2numeric` *Convert genotypic data in character to number for training models*

Description

This function help users to transform genetic matrix from character format to numeric format. AA-0, Aa-1, aa-2, A is major allele and a is minor allele.

Usage

```
transHapmap2numeric(G)
```

Arguments

`G`	Genetic matrix of character, row represents sample and column represents SNP.

Details

0: AA
1: Aa
2: aa

A is major allele and a is minor allele

Value

A matrix for genotypic data in numeric format.

Author(s)

Qian Cheng, Shuqin Jiang, Xiangfeng Wang

Examples

```
## Not run!
## load hapmap data (genomic data) of MZ hybrids
data(MZ)

## pre-process for input of transHapmap2numeric
rownames(MZ) <- MZ[,1]
MZ <- MZ[,-c(1:11)]
MZ.t <- t(MZ)

## conversion
MZ.n <- transHapmap2numeric(MZ.t)
dim(MZ.t)
```

Index

A.mat, *5*

bins, 2, *12*
binsInfo, 2
binsPlot, 2

dist, *12*

extractGenome, 3, *7*, *9*

GBLUP, 5
gcaCompute, 6, *15*
genomeOptimization, *4*, 7, *9*, *17*
GOVS, *4*, *7*, 8, *17*

hclust, *12*

IBDConstruct, *3*, 9
IBDTestData, 11

kin.blup, *5*

mixed.solve, *16*
mosaicPlot, 12
MZ, 13

phe, 13

reviseFunc, 13
rrBLUP, *5*, *16*

scaCompute, 14
SNPrrBLUP, 15
statDesign, *9*, 16

transHapmap2numeric, *15*, *16*, 18