

Package ‘mirf’

April 17, 2009

Version 1.0

Date 2008-09-03

Title MULTIPLE IMPUTATION AND RANDOM FORESTS FOR UNOBSERVABLE PHASE,
HIGH-DIMENSIONAL DATA

Author Yimin Wu, B. Aletta S. Nonyane and Andrea S. Foulkes

Maintainer Yimin Wu <yiminwu@cs.umass.edu>

Depends R (>= 2.5.1), haplo.stats, randomForest

Description This package applies a combination of missing haplotype imputation via the EM algorithm of Excoffier and Slatkin(1995) and modelling trait-haplotype associations via the Random Forest algorithm. The EM algorithm is implemented by the function haplo.em (of the haplo.stats package) and the Random Forest algorithm is implemented by the randomForest function (of the randomForest package). This method is described in the published manuscript: B.A.S. Nonyane and A.S. Foulkes (2007) Multiple imputation and random forests (MIRF) for unobservable high-dimensional data. The International Journal of Biostatistics 3(1): Article 12.

License BSD

Citation Support for developing this R package was provided by the National Institute of Allergy and Infectious Diseases (NIAID) research award (PI:Foulkes, R01 AI056983).

URL <http://www-unix.oit.umass.edu/~foulkes/>

Repository CRAN

Date/Publication 2008-09-10 07:18:18

R topics documented:

FMSmirfpckg	2
mirf	2
sepGeno	5

Index	7
--------------	----------

`FMSmirfpckg`*Functional Muscle Size and Strength*

Description

Data example used here comes from "Functional SNPs Associated with Muscle Size and Strength study" (FAMuSS) (Thompson et al., 2004, Clarkson et al. 2005). The FAMuSS study was conducted to determine the effects of single nucleotide polymorphisms (SNPs) on skeletal muscle size and strength before and after exercise training. For the purposes of illustration, we use a subset of data including 2 genes namely ACTN3 and RESISTIN which were shown to be associated with muscle size and strength. ACTN3 has 4 SNPs and RESISTIN has 6 SNPs. The trait under study is percentage change in muscle strength and is labelled NDRM.CH in the dataset. Covariates are Center, Age and Gender. The covariate used for HWE grouping is Race, which has 5 categories.

Usage

```
data(FMSmirfpckg)
```

Format

A data frame containing 991 rows and 16 columns (the first column is ID, column 2 to column 15 are genotypes, followed by Center, Gender, Age, Race, and NDRM.CH)

Source

Thompson et al.

References

Functional polymorphisms associated with human muscle size and strength. Thompson, P.D., Moyna, N. Seip, R., Clarkson, P., Angelopoulos, T., Gordon, P., Pescatello, L., Visich, P., Zoeller, R., Devaney, J.M., Gordish, H., Bilbie, S., Hoffman, E.P. *Med Sci Sports Exerc.* 2004; 36(7): 1132-9

The grant number for FAMuSS is R01NS40606-02-Hoffman.

`mirf`*MULTIPLE IMPUTATION AND RANDOM FORESTS FOR UNOBSERVABLE PHASE, HIGH-DIMENSIONAL DATA*

Description

mirf combines multiple imputation and random forests to characterize haplotype-trait associations. This approach involves estimation of allelic phase, imputing data according to these estimates and then combining the results of random forests across multiply imputed datasets. The method depends on two existing R packages: haplo.em and randomForest. Any parameters in these two packages can also be set in this function.

Usage

```
mirf(geno, y, gene.column=NULL, gene.names=NULL, SNPnames=NULL,
M=10, hwe.group=NULL, covariates=NULL, residualize=NULL,
miss.val=c(0, NA, "NA"), weight=NULL, control=haplo.em.control(), ...)
```

Arguments

<code>geno</code>	a data frame or matrix of alleles, such that each locus has a pair of adjacent columns of alleles, and the order of columns corresponds to the order of loci on a chromosome. If there are C loci, then $\text{ncol}(\text{geno}) = 2 * C$. Rows represent the alleles for each subject.
<code>y</code>	a vector of responses for the subjects, $\text{length}(y) = \text{nrow}(\text{geno})$. Both categorical and continuous y are acceptable. (NOTE: missing values are not allowed. If y is categorical, the argument "residualize" is not permitted)
<code>gene.column</code>	a vector with kth element equal to the number of columns in the geno matrix for gene k. If there are K genes and C loci, then $\text{length}(\text{gene.column}) = K$, and $\text{sum}(\text{gene.column}) = \text{ncol}(\text{geno}) = 2 * C$. If user doesn't input this variable, then we assume all columns in the geno matrix belong to one gene. In another word, we set the default value to $c(2 * C)$.
<code>gene.names</code>	vector of names for genes. The length of gene.names is equal to the length of gene.column
<code>SNPnames</code>	vector of names for SNP.
<code>M</code>	number of haplotype-imputations. The default value is set to 10. This value may be set higher for more accuracy, and is recommended for small sample sizes. (NOTE: increasing this value will increase computation time)
<code>hwe.group</code>	a vector containing a categorical variable which defines grouping according to the assumption of Hardy Weinberg equilibrium (HWE). HWE is assumed within each level of hwe.group. The default value is set to NULL. If specified, haplotype frequency estimates are obtained within each group separately.
<code>covariates</code>	a matrix of covariates that are predictors of the response. Default set to NULL. There are two options for dealing with covariates: the first is to include them in the random forest analysis and the second one is to residualize the response (y), see residualize option below. (NOTE: the missing data in the covariates are omitted by default.)
<code>residualize</code>	a vector indicating which columns in the covariates matrix are to be used in residualizing the response. Default set to NULL. Example: if $\text{ncol}(\text{covariates}) = 3$ and $\text{residualize} = c(1,3)$, then the first and the third columns in the covariates matrix are used to residualize y, and the second column is included in random-Forest analysis together with haplotype predictors. If $\text{residualize} = \text{NULL}$, then all covariates are included in the randomForest analysis.
<code>miss.val</code>	This is a parameter in "haplo.em". Please refer to the documentation of "haplo.em".
<code>weight</code>	This is a parameter in "haplo.em". Please refer to the documentation of "haplo.em".
<code>control</code>	This is a parameter in "haplo.em". Please refer to the documentation of "haplo.em".
<code>...</code>	You can pass all possible parameters defined in randomForest method. Please refer to the documentation of "randomForest".

Value

An object of class mirf which has the following component:

score	it should read as a data frame with four(or more) columns. The first column is gene name to which the haplotype predictor corresponds. The second column is imputed haplotypes and covariates included as predictors in randomForest analysis. The third column contains importance scores for the predictors, averaged over imputations taking into account the within- and between-imputation variance. Importance score=NAN implies that the sample size was too small to estimate the average importance score for that predictor. The remaining columns are the estimated haplotype frequencies from haplo.em for different hwe groups.
-------	--

Note

You can also set all remaining parameters accepted by haplo.em and randomForest. Please check their documentation for more information.

Author(s)

Yimin Wu, B. Aletta S. Nonyane and Andrea S. Foulkes

References

B.A.S. Nonyane and A.S. Foulkes (2007) Multiple imputation and random forests (MIRF) for unobservable high-dimensional data. The International Journal of Biostatistics 3(1): Article 12

See Also

[haplo.em](#), [randomForest](#)

Examples

```
## Not run:
library(mirf)
data(FMSmirfpckg)

# ACTN3 and RESISTIN are genes with 4 and 6 SNPs, respectively.
genotype <- FMSmirfpckg[,c(2:11)]
gene.names <- c("actn3", "resistin")

# Make a genotype matrix with one letter per column.
# See help(sepGeno) for more detail.
genoSingleColsResult <- sepGeno(genotype)
genoSingleCols <- genoSingleColsResult$geno
SNPnames <- genoSingleColsResult$SNPnames

# Now ACTN3 gene has 8 columns and RESISTIN has 12 columns
gene.column <- c(8,12)
trait <- FMSmirfpckg$"NDRM.CH"

# Assuming HWE for entire cohort and no covariates
```

```

# Note: this takes several minutes to run
mirf(geno=genoSingleCols, y=trait, gene.column=gene.column,
gene.names = gene.names, SNPnames=SNPnames, M=4)

# Specifying groups within which HWE is expected to hold
# and specifying covariates to include as predictors
hwe.group <- FMSmirfpckg$"Race"

# HWE assumed within groups defined by RACE
covariates<-cbind(FMSmirfpckg$"Center",FMSmirfpckg$"Gender",FMSmirfpckg$"Age")
dimnames(covariates)[[2]] <- c("Center","Gender","Age")

mirf(geno=genoSingleCols, y=trait, gene.column=gene.column,
gene.names = gene.names, SNPnames=SNPnames, M=4, hwe.group=hwe.group, covariates=covariates)

# Residualizing by Center
residualize <-c(1)
mirf(geno=genoSingleCols, y=trait, gene.column=gene.column,
gene.names = gene.names, SNPnames=SNPnames, M=4, hwe.group=hwe.group,
covariates=covariates, residualize=residualize)
## End(Not run)

```

sepGeno

CHANGE A GENOTYPE MATRIX THAT HAS TWO LETTERS IN EACH COLUMN TO A NEW GENOTYPE MATRIX WITH ONE COLUMN PER LETTER

Description

This function is a preprocess function that takes a geno matrix that has two letters in each column (with no separation or any one character as separation) and return a new geno matrix with one column per letter. This is useful because mirf and haplo.em only take a geno matrix with one column per letter.

Usage

```
sepGeno(inputGeno)
```

Arguments

`inputGeno` a data frame or matrix of alleles, each column has two letters (with no separation or one symbol as separation such as "/"). Rows represent the alleles for each subject.

Value

An object of class sepGeno which has the following component:

`geno` a new geno matrix with one column per letter, which can be used in mirf or haplo.em

SNPnames a vector of names for SNPs. If it is NULL, that means the inputGeno matrix doesn't have column names. This vector can be input as SNPnames in mirf or as locus.label in haplo.em

Author(s)

Yimin Wu, B. Aletta S. Nonyane and Andrea S. Foulkes

References

B.A.S. Nonyane and A.S. Foulkes (2007) Multiple imputation and random forests (MIRF) for unobservable high-dimensional data. *The International Journal of Biostatistics* 3(1): Article 12

See Also

[mirf](#), [haplo.em](#), [randomForest](#)

Examples

```
library(mirf)
data(FMSmirfpckg)

inputGeno <- FMSmirfpckg[1:4, c(2:5)]

genoSingleCols <- sepGeno(inputGeno)
genoSingleCols$geno
genoSingleCols$SNPnames
```

Index

*Topic **datasets**

FMSmirfpckg, [2](#)

*Topic **models**

mirf, [2](#)

sepGeno, [5](#)

FMSmirfpckg, [2](#)

haplo.em, [4, 6](#)

mirf, [2, 6](#)

randomForest, [4, 6](#)

sepGeno, [5](#)