fbat

April 19, 2010

`checkMarkers` *Basic data quality checks for markers*

Description

Basic data quality checks for markers.

Usage

```
checkMarkers(geneSetObj, founderOnly=TRUE, thrsh=0.05, quiet=TRUE)
checkMarkers.default(pedObj, founderOnly=TRUE, thrsh=0.05, quiet=TRUE)
```

Arguments

`geneSetObj`	a `geneSet` object.
`pedObj`	a list with five elements: `ped`, `columns`, `markerNames`, `Position`, and `filename`. `ped` is a pedigree data frame whose first 6 columns are family (pedigree id), pid (patient id), father (father id), mother (mother id), sex, affected (affection status). The remaining columns are pairs of marker alleles. Each row corresponds to an individual; `columns` are the names of the first 5 (or 6) columns of ped file. It should be either equal to c("family","pid","father","mother","sex","affected") or equal to c("family","pid","father","mother","sex"); `founderOnly` indicates if using only founder info; `markerNames` is a vector of marker names; `Position` is a vector of marker positions; `fileName` is the pedigree file name
`founderOnly`	indicates if using only founder info
`thrsh`	threshold for Hardy-Weinberg test. If the pvalue of the HW test for a marker is greater than `thrsh`, then the marker is a good marker.
`quiet`	print intermediate results if `quiet=FALSE`.

Value

a data frame contains components:

`Name`	marker names.
`Position`	marker positions.
`ObsHET`	marker's observed heterozygosity (i.e., proportion of heterozygotes at markes). Missing alleles are excluded in the calculation.

PredHET	marker's predicted heterozygosity (i.e., 2*MAF*(1-MAF)). Missing alleles are excluded in the calculation.
HWpval	pvalues for Hardy-Weinberg test
pGeno	percentage of non-missing genotypes for markes
MAF	minor allele frequencies. missing allele are excluded from calculation
Rating	Rating[i]=1 means that the i-th marker passes HW test (do not reject H0 that HW equilibrium holds). Rating[i]=0 means HW equilibrum does hold for the i-th marker.

Author(s)

Weiliang Qiu <stwxq@channing.harvard.edu>, Ross Lazarus <ross.lazarus@channing.harvard.ed
Gregory Warnes <warnes@bst.rochester.edu>, Nitin Jain <nitin.jain@pfizer.com>

Examples

```
data(CAMP)
res<-checkMarkers(CAMP)
print(res)
```

---

checkMendelian *Check Mendelian Errors*

Description

Check Mendelian errors.

Usage

```
checkMendelian(geneSetObj, quiet = TRUE)

checkMendelian.default(pedObj, quiet=TRUE)
```

Arguments

geneSetObj	a geneSet object.
pedObj	a list with five elements: ped, columns, markerNames, Position, and filename. ped is a pedigree data frame whose first 6 columns are family (pedigree id), pid (patient id), father (father id), mother (mother id), sex, affected (affection status). The remaining columns are pairs of marker alleles. Each row corresponds to an individual; columns are the names of the first 5 (or 6) columns of ped file. It should be either equal to c("family","pid","father","mother","sex","affected") or equal to c("family","pid","father","mother","sex"); founderOnly indicates if using only founder info; markerNames is a vector of marker names; Position is a vector of marker positions; fileName is the pedigree file name
quiet	print intermediate results if quiet=FALSE.

Details

check the following errors:

1 father id = subject id

2 mother id = subject id

3 could not determine if an individual is a parent or a child in a family

4 inconsistent parental sex in a family

5 parental genotypes are not compatible with childrens' genotypes in a family

6 all childrens' genotypes are missing in a family

7 inconsistent sib genotypes in a family

Value

A list with following elements:

`errorFlag` `errorFlag=1` indicates the occurence of errors; `errorFlag=0` indicates no error.

`compatibleFlag` `compatibleFlag=0` indicates the occurence of non-compatibility; `compatibleFlag=1` indicates compatibility.

`nMerrMarker` A $nMarkers x 1$ vector records the numbers of families with non-compatible genotypes, where $nMarkers$ is the number of markers.

`nMerrFamily` A $nFamily x 1$ vector records the numbers of markers with non-compatible genotypes, where $nFamily$ is the number of families.

`nErrFamilySample` A $nFamily x 1$ vector records the numbers of times that father id is equal to subject id or mother id is equal to subject id in a family.

Author(s)

Weiliang Qiu <stwxq@channing.harvard.edu>, Ross Lazarus <ross.lazarus@channing.harvard.ed
Gregory Warnes <warnes@bst.rochester.edu>, Nitin Jain <nitin.jain@pfizer.com>

Examples

```
data(CAMP)
checkMendelian(CAMP, quiet = TRUE)
```

`fbat` *Family-Based Association Tests*

Description

Family-Based Assoiciation Tests for biallelic markers.

Usage

```
fbat(geneSetObj,
     model="a",
     traitMethod=3,
     traitOffset=0,
     quiet=TRUE)

fbat.default(pedObj,
             model="a",
             traitMethod=3,
             traitOffset=0,
             quiet=TRUE)
```

Arguments

geneSetObj	an object of geneSet.
pedObj	a list with five elements: ped, columns, markerNames, Position, and filename. ped is a pedigree data frame whose first 6 columns are family (pedigree id), pid (patient id), father (father id), mother (mother id), sex, affected (affection status). The remaining columns are pairs of marker alleles. Each row corresponds to an individual; columns are the names of the first 5 (or 6) columns of ped file. It should be either equal to c("family","pid","father","mother","sex","affected") or equal to c("family","pid","father","mother","sex"); founderOnly indicates if using only founder info; markerNames is a vector of marker names; Position is a vector of marker positions; fileName is the pedigree file name
model	Genotype coding method. model="d" means GDOM (dominante) coding; model="r" means GREC (recessive) coding; model="g" means GEN (genotype) coding; model="a" or otherwise means GTDT (additive) coding.
traitMethod	Trait coding method. traitMethod=1 means T=y-offset, where y is the trait and offset is an offset. In a .ped file, y=2 if affected; y=1 if unaffected; and y=0 if unknown. traitMethod=2 means T=1 if affected, T=0 otherwise.
traitOffset	Offset if traitMethod=1.
quiet	Print some intermediate results if quiet=FALSE.

Value

statPvalue	A m by 3 matrix with the 3 columns: test statistics, degree of freedom and pvalues, where m is the number of markers.
S.list	A list of S scores for markers.
ES.list	A list of expected S scores for markers.
CovS.list	A list of covariance matrix of S scores for markers.
alleles.list	A list of alleles for markers
familySize	size of nuclear families
flagMarkers	A vector of flags. flagMarkers[i]=1 if for marker i, all children genotypes in all families are missing. Otherwise flagMarkers[i]=0.
numInfoFamily	number of informative families at each marker

Author(s)

Weiliang Qiu <stwxq@channing.harvard.edu>, Ross Lazarus <ross.lazarus@channing.harvard.ed
Gregory Warnes <warnes@bst.rochester.edu>, Nitin Jain <nitin.jain@pfizer.com>

References

Horvath et al. The family based association test method: computing means and variances for general statistics. *Technical report* http://www.biostat.harvard.edu/~fbat/fbattechreport.ps.

Rabinowitz and Laird (2000). A Unified Approach to Adjusting Association Tests for Population Admixture with Arbitrary Pedigree Structure and Arbitrary Missing Marker Information. *Human Heredity* ***50****:211-223.*

Laird et al. (2000). Implementing a Unified Approach to Family-Based Tests of Association. *Genetic Epidemiology* ***19(Suppl 1)****:S36-S42.*

Schaid (1996). General Score Tests for Associations of Genetic Markers With Disease Using Cases and Their Parents. *Genetic Epidemiology* ***13****:423-449.*

Examples

```
data(CAMP)
tmp<-fbat(CAMP)
summaryPvalue(tmp)
```

---

getFounders *Get founders' information*

Description

Get a subset of pedigree object containing only founders' information.

Usage

```
getFounders(pedObj)
```

Arguments

pedObj	a list with five elements: ped, columns, markerNames, Position, and filename. ped is a pedigree data frame whose first 6 columns are family (pedigree id), pid (patient id), father (father id), mother (mother id), sex, affected (affection status). The remaining columns are pairs of marker alleles. Each row corresponds to an individual; columns are the names of the first 5 (or 6) columns of ped file. It should be either equal to c("family","pid","father","mother","sex","affected") or equal to c("family","pid","father","mother","sex"); founderOnly indicates if using only founder info; markerNames is a vector of marker names; Position is a vector of marker positions; fileName is the pedigree file name

Value

An pedigree object contains only founders' information.

Author(s)

Weiliang Qiu <stwxq@channing.harvard.edu>, Ross Lazarus <ross.lazarus@channing.harvard.ed
Gregory Warnes <warnes@bst.rochester.edu>, Nitin Jain <nitin.jain@pfizer.com>

References

~put references to the literature/web site here ~

---

missGFreq *Count frequencies of missing genotypes*

Description

Count frequencies of missing genotypes

Usage

```
missGFreq(geneSetObj, founderOnly = TRUE, quiet = FALSE)

missGFreq.default(pedObj, founderOnly=TRUE)
```

Arguments

geneSetObj	a geneSet object.
pedObj	a list with five elements: ped, columns, markerNames, Position, and filename. ped is a pedigree data frame whose first 6 columns are family (pedigree id), pid (patient id), father (father id), mother (mother id), sex, affected (affection status). The remaining columns are pairs of marker alleles. Each row corresponds to an individual; columns are the names of the first 5 (or 6) columns of ped file. It should be either equal to c("family","pid","father","mother","sex","affected") or equal to c("family","pid","father","mother","sex"); founderOnly indicates if using only founder info; markerNames is a vector of marker names; Position is a vector of marker positions; fileName is the pedigree file name
founderOnly	indicates if using only founder info
quiet	print intermediate results if quiet=FALSE.

Value

A matrix with the following three columns:

column 1	counts of genotypes, of which both alleles are missing.
column 2	counts of genotypes, of which the first allele is missing and the second allele is not missing.
column 3	counts of genotypes, of which the first allele is not missing and the second allele is missing.

Author(s)

Weiliang Qiu <stwxq@channing.harvard.edu>, Ross Lazarus <ross.lazarus@channing.harvard.ed
Gregory Warnes <warnes@bst.rochester.edu>, Nitin Jain <nitin.jain@pfizer.com>

Examples

```
data(CAMP)
res<-missGFreq(CAMP,founderOnly=FALSE)
# number of missing genotypes per marker
print(res$nMissMarkers)
# number of missing genotypes per subject
print(res$nMissSubjects[1:10,])
```

---

pedAFreq *get allele frequencies*

Description

Get allele frequencies (missing alleles allowed).

Usage

```
pedAFreq(geneSetObj, founderOnly=TRUE, missingOutput=FALSE, quiet=FALSE)

pedAFreq.default(pedObj,
         founderOnly=TRUE,
         missingOutput=FALSE,
         quiet=FALSE)
```

Arguments

`geneSetObj`	a `geneSet` object.
`pedObj`	a list with five elements: `ped`, `columns`, `markerNames`, `Position`, and `filename`. `ped` is a pedigree data frame whose first 6 columns are family (pedigree id), pid (patient id), father (father id), mother (mother id), sex, affected (affection status). The remaining columns are pairs of marker alleles. Each row corresponds to an individual; `columns` are the names of the first 5 (or 6) columns of ped file. It should be either equal to c("family","pid","father","mother","sex","affected") or equal to c("family","pid","father","mother","sex"); `founderOnly` indicates if using only founder info; `markerNames` is a vector of marker names; `Position` is a vector of marker positions; `fileName` is the pedigree file name
`founderOnly`	indicates if using only founder info.
`missingOutput`	indicates if missing allele frequency should be output.
`quiet`	print intermediate results if `quiet=FALSE`.

Value

`aFreqMat`	allele frequencies.
`aPercMat`	allele percentages.
`missingOutput`	indicates if missing allele frequency should be output.

Author(s)

Weiliang Qiu <stwxq@channing.harvard.edu>, Ross Lazarus <ross.lazarus@channing.harvard.ed
Gregory Warnes <warnes@bst.rochester.edu>, Nitin Jain <nitin.jain@pfizer.com>

Examples

```
data(CAMP)
res<-pedAFreq(CAMP)
res$aFreqMat
res$aPercMat
res$missingOutput
```

---

pedFlagHomo *flag homo/heterozygotes*

Description

Flag homo/heterozygotes.

Usage

```
pedFlagHomo(geneSetObj, founderOnly=TRUE, quiet=FALSE)

pedFlagHomo.default(pedObj,
            founderOnly=TRUE,
            quiet=FALSE)
```

Arguments

geneSetObj	a geneSet object.
pedObj	a list with five elements: ped, columns, markerNames, Position, and filename. ped is a pedigree data frame whose first 6 columns are family (pedigree id), pid (patient id), father (father id), mother (mother id), sex, affected (affection status). The remaining columns are pairs of marker alleles. Each row corresponds to an individual; columns are the names of the first 5 (or 6) columns of ped file. It should be either equal to c("family","pid","father","mother","sex","affected") or equal to c("family","pid","father","mother","sex"); founderOnly indicates if using only founder info; markerNames is a vector of marker names; Position is a vector of marker positions; fileName is the pedigree file name
founderOnly	indicates if using only founder info
quiet	print intermediate results if quiet=FALSE.

Value

countMat	Count the number of homo/heterozygotes.
flagHomoMat	Flag homo/heterozygotes. 1 – homozygotes; 0 – heterozygotes; -1 – genotype contains one missing allele; -2 – genotype contains two missing alleles.
markerNames	marker names.

Author(s)

Weiliang Qiu <stwxq@channing.harvard.edu>, Ross Lazarus <ross.lazarus@channing.harvard.ed
Gregory Warnes <warnes@bst.rochester.edu>, Nitin Jain <nitin.jain@pfizer.com>

Examples

```
data(CAMP)
res<-pedFlagHomo(CAMP)
res$countMat
res$flagHomoMat
res$markerNames
```

---

pedGFreq *get genotype frequencies*

Description

Get genotype frequencies (missing alleles allowed).

Usage

```
pedGFreq(geneSetObj, founderOnly=TRUE, missingOutput=FALSE, quiet=FALSE)

pedGFreq.default(pedObj,
         founderOnly=TRUE,
         missingOutput=FALSE,
         quiet=FALSE)
```

Arguments

geneSetObj	a geneSet object.
pedObj	a list with five elements: ped, columns, markerNames, Position, and filename. ped is a pedigree data frame whose first 6 columns are family (pedigree id), pid (patient id), father (father id), mother (mother id), sex, affected (affection status). The remaining columns are pairs of marker alleles. Each row corresponds to an individual; columns are the names of the first 5 (or 6) columns of ped file. It should be either equal to c("family","pid","father","mother","sex","affected") or equal to c("family","pid","father","mother","sex"); founderOnly indicates if using only founder info; markerNames is a vector of marker names; Position is a vector of marker positions; fileName is the pedigree file name
founderOnly	indicates if using only founder info.
missingOutput	indicates if missing allele frequency should be output.
quiet	print intermediate results if quiet=FALSE.

Value

gFreqMat	genotype frequencies.
gPercMat	genotype percentages.
missingOutput	indicates if missing allele frequency should be output.

Author(s)

Weiliang Qiu <stwxq@channing.harvard.edu>, Ross Lazarus <ross.lazarus@channing.harvard.ed
Gregory Warnes <warnes@bst.rochester.edu>, Nitin Jain <nitin.jain@pfizer.com>

Examples

```
data(CAMP)
res<-pedGFreq(CAMP)
res$gFreqMat
res$gPercMat
```

---

pedHardyWeinberg *Test Hardy-Weinberg equilibrium for each marker based on parental data*

Description

Test Hardy-Weinberg equilibrium for each marker based on parental data.

Usage

```
pedHardyWeinberg(geneSetObj, threshold=3, founderOnly=TRUE, quiet=FALSE)

pedHardyWeinberg.default(pedObj,
          threshold=3, founderOnly=TRUE, quiet=FALSE)
```

Arguments

geneSetObj	a geneSet object.
pedObj	a list with five elements: ped, columns, markerNames, Position, and filename. ped is a pedigree data frame whose first 6 columns are family (pedigree id), pid (patient id), father (father id), mother (mother id), sex, affected (affection status). The remaining columns are pairs of marker alleles. Each row corresponds to an individual; columns are the names of the first 5 (or 6) columns of ped file. It should be either equal to c("family","pid","father","mother","sex","affected") or equal to c("family","pid","father","mother","sex"); founderOnly indicates if using only founder info; markerNames is a vector of marker names; Position is a vector of marker positions; fileName is the pedigree file name
threshold	a threshold to check if expected frequencies of genotypes are too small.
founderOnly	indicates if using only founder info
quiet	print intermediate results if quiet=FALSE.

Value

resMat	A matrix records the following quantities for all markers (rows correspond to markers): nInfoInd (number of informative individuals, i.e. individuals whose genotypes contain no missing alleles for the specified marker), nGenotype (number of possible genotypes), nHET (number of heterozygous genotypes), nHOM (number of homozygous genotypes), nAllele (number of alleles), nMissing (number of missing alleles), chi2 (chi square test statistic), df (degree of freedom of the chi square test statistic under H0), p-value (pvalue of the test).

genotype	A list of possible genotypes and their frequencies for all markers.
nGenotype.vec	A vector of numbers of possible genotypes for all markers.
piVec	Allele frequencies for all markers.

Author(s)

Weiliang Qiu <stwxq@channing.harvard.edu>, Ross Lazarus <ross.lazarus@channing.harvard.ed
Gregory Warnes <warnes@bst.rochester.edu>, Nitin Jain <nitin.jain@pfizer.com>

Examples

```
data(CAMP)
res<-pedHardyWeinberg(CAMP)
viewHW(res, "m709")
viewHW(res, "m654")
viewHW(res, "m47")
viewHW(res, "p46")
viewHW(res, "p79")
viewHW(res, "p252")
viewHW(res, "p491")
viewHW(res, "p523")
```

---

readHapMap *Import HapMap data*

Description

Import HapMap data and convert it to pedigree format.

Usage

```
readHapMap(hapmapfile, race="CEU", skip = 2, comment.char = "&", quiet = FALSE)
```

Arguments

hapmapfile	the hapmap file name
race	can take values CEU, YRI, CHB, and JPT
skip	first skip lines in the file hapmapfile will be skipped.
comment.char	hapmapfile snp names contain the symbol \# which is the comment command of R. So by default, we set comment.char as "&".
quiet	print intermediate results if quiet=FALSE.

Details

HapMap files are those snp files output by HapMap browsers.

Value

A list with five elements: ped, columns, markerNames, Position, and fileName. ped is a pedigree data frame whose first 6 columns are family (pedigree id), pid (patient id), father (father id), mother (mother id), sex, affected (affection status). The remaining columns are pairs of marker alleles. Each row corresponds to an individual; columns are the names of the first 5 (or 6) columns of ped file. It should be either equal to c("family","pid","father","mother","sex","affected") or equal to c("family","pid","father","mother","sex"); founderOnly indicates if using only founder info; markerNames is a vector of marker names; Position is a vector of marker positions; fileName is the pedigree file name.

Author(s)

Weiliang Qiu <stwxq@channing.harvard.edu>, Ross Lazarus <ross.lazarus@channing.harvard.ed
Gregory Warnes <warnes@bst.rochester.edu>, Nitin Jain <nitin.jain@pfizer.com>

---

readLink *Import file with PLINK data format*

Description

Import file with PLINK data format and convert it to a pedigree object.

Usage

```
readLink(pedfile, gmfile, columns = c("family", "pid", "father", "mother", "sex"
```

Arguments

pedfile	pedigree data file with no marker info
gmfile	marker info files. It contains three columns: marker IDs, marker names, and marker positions
columns	By default, the first five columns of pedfile are sample information: family id, patient id, father id, mother id, patient sex.
quiet	print intermediate results if quiet=FALSE.

Details

The data format is used by the software PLINK.

Value

A list with five elements: ped, columns, markerNames, Position, and fileName. ped is a pedigree data frame whose first 6 columns are family (pedigree id), pid (patient id), father (father id), mother (mother id), sex, affected (affection status). The remaining columns are pairs of marker alleles. Each row corresponds to an individual; columns are the names of the first 5 (or 6) columns of ped file. It should be either equal to c("family","pid","father","mother","sex","affected") or equal to c("family","pid","father","mother","sex"); founderOnly indicates if using only founder info; markerNames is a vector of marker names; Position is a vector of marker positions; fileName is the pedigree file name.

Author(s)

Weiliang Qiu <stwxq@channing.harvard.edu>, Ross Lazarus <ross.lazarus@channing.harvard.ed
Gregory Warnes <warnes@bst.rochester.edu>, Nitin Jain <nitin.jain@pfizer.com>

---

readPed *Import pedigree file from standard pedigree file format*

Description

Import pedigree file from standard pedigree file format.

Usage

```
readPed (
        filename,
        columns=c("family", "pid", "father", "mother",
                  "sex", "affected"),
        quiet=FALSE
        )
```

Arguments

filename	File containing genotype data
columns	column names for sample info
quiet	indicates if intermediate output should be printed

Value

A list with five elements: ped, columns, markerNames, Position, and fileName. ped is a pedigree data frame whose first 6 columns are family (pedigree id), pid (patient id), father (father id), mother (mother id), sex, affected (affection status). The remaining columns are pairs of marker alleles. Each row corresponds to an individual; columns are the names of the first 5 (or 6) columns of ped file. It should be either equal to c("family","pid","father","mother","sex","affected") or equal to c("family","pid","father","mother","sex"); founderOnly indicates if using only founder info; markerNames is a vector of marker names; Position is a vector of marker positions; fileName is the pedigree file name.

Author(s)

Weiliang Qiu <stwxq@channing.harvard.edu>, Ross Lazarus <ross.lazarus@channing.harvard.ed
Gregory Warnes <warnes@bst.rochester.edu>, Nitin Jain <nitin.jain@pfizer.com>

See Also

read.table, etc

sampleInfoCEU *Information about HapMap CEU subjects*

Description

Information about HapMap CEU subjects.

Usage

```
data(sampleInfoCEU)
```

Format

A data frame with 90 observations on the following 6 variables.

`pedID` a numeric vector

`id` a numeric vector

`fid` a numeric vector

`mid` a numeric vector

`sex` a numeric vector

`sampleID` a factor with levels `NA06985` `NA06991` `NA06993` `NA06994` `NA07000` `NA07019` `NA07022` `NA07029` `NA07034` `NA07048` `NA07055` `NA07056` `NA07345` `NA07348` `NA07357` `NA10830` `NA10831` `NA10835` `NA10838` `NA10839` `NA10846` `NA10847` `NA10851` `NA10854` `NA10855` `NA10856` `NA10857` `NA10859` `NA10860` `NA10861` `NA10863` `NA11829` `NA11830` `NA11831` `NA11832` `NA11839` `NA11840` `NA11881` `NA11882` `NA11992` `NA11993` `NA11994` `NA11995` `NA12003` `NA12004` `NA12005` `NA12006` `NA12043` `NA12044` `NA12056` `NA12057` `NA12144` `NA12145` `NA12146` `NA12154` `NA12155` `NA12156` `NA12234` `NA12236` `NA12239` `NA12248` `NA12249` `NA12264` `NA12707` `NA12716` `NA12717` `NA12740` `NA12750` `NA12751` `NA12752` `NA12753` `NA12760` `NA12761` `NA12762` `NA12763` `NA12801` `NA12802` `NA12812` `NA12813` `NA12814` `NA12815` `NA12864` `NA12865` `NA12872` `NA12873` `NA12874` `NA12875` `NA12878` `NA12891` `NA12892`

Examples

```
data(sampleInfoCEU)
```

sampleInfoCHB *Information about HapMap CHB subjects*

Description

Information about HapMap CHB subjects

Usage

```
data(sampleInfoCHB)
```

Format

A data frame with 45 observations on the following 6 variables.

`pedid` a numeric vector

`id` a numeric vector

`fid` a numeric vector

`mid` a numeric vector

`sex` a numeric vector

`sampleID` a factor with levels `NA18524` `NA18526` `NA18529` `NA18532` `NA18537` `NA18540` `NA18542` `NA18545` `NA18547` `NA18550` `NA18552` `NA18555` `NA18558` `NA18561` `NA18562` `NA18563` `NA18564` `NA18566` `NA18570` `NA18571` `NA18572` `NA18573` `NA18576` `NA18577` `NA18579` `NA18582` `NA18592` `NA18593` `NA18594` `NA18603` `NA18605` `NA18608` `NA18609` `NA18611` `NA18612` `NA18620` `NA18621` `NA18622` `NA18623` `NA18624` `NA18632` `NA18633` `NA18635` `NA18636` `NA18637`

Examples

```
data(sampleInfoCHB)
```

---

sampleInfoJPT *Information about HapMap JPT subjects*

Description

Information about HapMap JPT subjects

Usage

```
data(sampleInfoJPT)
```

Format

A data frame with 45 observations on the following 6 variables.

`pedid` a numeric vector

`id` a numeric vector

`fid` a numeric vector

`mid` a numeric vector

`sex` a numeric vector

`sampleID` a factor with levels `NA18940` `NA18942` `NA18943` `NA18944` `NA18945` `NA18947` `NA18948` `NA18949` `NA18951` `NA18952` `NA18953` `NA18956` `NA18959` `NA18960` `NA18961` `NA18964` `NA18965` `NA18966` `NA18967` `NA18968` `NA18969` `NA18970` `NA18971` `NA18972` `NA18973` `NA18974` `NA18975` `NA18976` `NA18978` `NA18980` `NA18981` `NA18987` `NA18990` `NA18991` `NA18992` `NA18994` `NA18995` `NA18997` `NA18998` `NA18999` `NA19000` `NA19003` `NA19005` `NA19007` `NA19012`

Examples

```
data(sampleInfoJPT)
```

sampleInfoYRI *Information about HapMap YRI subjects*

Description

Information about HapMap YRI subjects

Usage

```
data(sampleInfoYRI)
```

Format

A data frame with 90 observations on the following 6 variables.

`pedID` a numeric vector

`id` a numeric vector

`fid` a numeric vector

`mid` a numeric vector

`sex` a numeric vector

`sampleID` a factor with levels `NA18500` `NA18501` `NA18502` `NA18503` `NA18504` `NA18505` `NA18506` `NA18507` `NA18508` `NA18515` `NA18516` `NA18517` `NA18521` `NA18522` `NA18523` `NA18852` `NA18853` `NA18854` `NA18855` `NA18856` `NA18857` `NA18858` `NA18859` `NA18860` `NA18861` `NA18862` `NA18863` `NA18870` `NA18871` `NA18872` `NA18912` `NA18913` `NA18914` `NA19092` `NA19093` `NA19094` `NA19098` `NA19099` `NA19100` `NA19101` `NA19102` `NA19103` `NA19116` `NA19119` `NA19120` `NA19127` `NA19128` `NA19129` `NA19130` `NA19131` `NA19132` `NA19137` `NA19138` `NA19139` `NA19140` `NA19141` `NA19142` `NA19143` `NA19144` `NA19145` `NA19152` `NA19153` `NA19154` `NA19159` `NA19160` `NA19161` `NA19171` `NA19172` `NA19173` `NA19192` `NA19193` `NA19194` `NA19200` `NA19201` `NA19202` `NA19203` `NA19204` `NA19205` `NA19206` `NA19207` `NA19208` `NA19209` `NA19210` `NA19211` `NA19221` `NA19222` `NA19223` `NA19238` `NA19239` `NA19240`

Examples

```
data(sampleInfoYRI)
```

summaryPvalue *summarize the test statistics and p-values*

Description

summarize the test statistics and p-values

Usage

```
summaryPvalue(fbatObject)
```

Arguments

fbatObject	Object for Family Based Association Tests. See references.

Details

Print summary of test statistics and p-value.

Author(s)

Weiliang Qiu <stwxq@channing.harvard.edu>, Ross Lazarus <ross.lazarus@channing.harvard.ed
Gregory Warnes <warnes@bst.rochester.edu>, Nitin Jain <nitin.jain@pfizer.com>

References

http://www.biostat.harvard.edu/~fbat/fbat.htm

Examples

```
data(CAMP)
tmp<-fbat(CAMP)
summaryPvalue(tmp)
```

---

viewFlagHomo *flag homo/heterozygotes for specified marker*

Description

Flag homo/heterozygoter for specified marker.

Usage

```
viewFlagHomo(flagHomo.object, markerName)
```

Arguments

flagHomo.object	object returned by the function pedFlagHomo().
markerName	name of the specified marker.

Value

countMatM	Count the number of homo/heterozygotes for the specified marker.
flagHomoMatM	Flag homo/heterozygotes for the specified marker. 1 – homozygotes; 0 – heterozygotes; -1 – genotype contains one missing allele; -2 – genotype contains two missing alleles.

Author(s)

Weiliang Qiu <stwxq@channing.harvard.edu>, Ross Lazarus <ross.lazarus@channing.harvard.ed
Gregory Warnes <warnes@bst.rochester.edu>, Nitin Jain <nitin.jain@pfizer.com>

Examples

```
data(CAMP)
res<-pedFlagHomo(CAMP)
viewFlagHomo(res, "p79")
```

`viewHW` *View allele frequencies, Hardy-Weinberg equilibrium test statistics for specified marker*

Description

View allele frequencies, Hardy-Weinberg equilibrium test statistics for specified marker.

Usage

```
viewHW(HW.object, markerName)
```

Arguments

`HW.object`	object returned by the function 'pedHardyWeinberg'.
`markerName`	a character string indicating the name of marker whose statistics are to be viewed

Value

`resM`	A vector records the following quantities for the specified marker: `nInfoInd` (number of informative individuals, i.e. individuals whose genotypes contain no missing alleles for the specified marker), `nGenotype` (number of possible genotypes), `nHET` (number of heterozygous genotypes), `nHOM` (number of homozygous genotypes), `nAllele` (number of alleles), `nMissing` (number of missing alleles), `chi2` (chi square test statistic), `df` (degree of freedom of the chi square test statistic under H0), `p-value` (pvalue of the test).
`nGenotypeM`	number of possible genotypes for the specified marker.
`genotypeM`	possible genotypes and their frequencies.
`piVecM`	allele frequencies.

Author(s)

Weiliang Qiu <stwxq@channing.harvard.edu>, Ross Lazarus <ross.lazarus@channing.harvard.ed
Gregory Warnes <warnes@bst.rochester.edu>, Nitin Jain <nitin.jain@pfizer.com>

Examples

```
data(CAMP)
res<-pedHardyWeinberg(CAMP)
viewHW(res, "m709")
viewHW(res, "m654")
viewHW(res, "m47")
viewHW(res, "p46")
viewHW(res, "p79")
viewHW(res, "p252")
viewHW(res, "p491")
viewHW(res, "p523")
```

viewstat *view statistics for a marker*

Description

view statistics for a marker

Usage

```
viewstat(fbatObject, markerName)
```

Arguments

fbatObject	Object for Family Based Association Tests. See references.
markerName	name(s) of the marker(s) for which statistics is needed

Details

Print various stats for a marker, such as: family size, number of people in the family, number of informative families in the marker, the alleles of marker, scores for marker, expected score for marker, covariance matrix of the score for the marker, Moore-Penrose generealized inverse of covariance matrix and P-value.

Author(s)

Weiliang Qiu <stwxq@channing.harvard.edu>, Ross Lazarus <ross.lazarus@channing.harvard.ed
Gregory Warnes <warnes@bst.rochester.edu>, Nitin Jain <nitin.jain@pfizer.com>

References

http://www.biostat.harvard.edu/~fbat/fbat.htm

Examples

```
data(CAMP)
res<-fbat(CAMP)
viewstat(res, "m709")
viewstat(res, "m654")
viewstat(res, "m47")
viewstat(res, "p46")
viewstat(res, "p79")
viewstat(res, "p252")
viewstat(res, "p491")
viewstat(res, "p523")
```

Index