Package ‘plinkFile’

April 5, 2020

Title 'PLINK' (and 'GCTA') File Helpers

Version 0.1.0

Description Provide function that reads binary genotype produced by 'PLINK' <https://www.cog-genomics.org/plink/1.9/input#bed> into a R matrix, or scan the genotype one variant at a time like apply(), it also provides functions that reads and writes genotype relatedness/kinship matrices created by 'PLINK' <https://www.cog-genomics.org/plink/1.9/distance#make_rel> or 'GCTA' <https://cnsgenomics.com/software/gcta/#MakingaGRM>. Currently it does not support writing back into 'PLINK' binary, it is best used for bringing data produced by 'PLINK' and 'GCTA' into R environment.

Depends R (>= 3.1)

License GPL (>= 2)

Encoding UTF-8

LazyData true

RoxygenNote 7.0.2

NeedsCompilation no

Author Xiaoran Tong [aut, cre]

Maintainer Xiaoran Tong <tongxia1@msu.edu>

Repository CRAN

Date/Publication 2020-04-05 15:30:02 UTC

R topics documented:

dbd *Decompress Byte Data*

Description

For each SNP (i.e., a row in the BED), a byte encodes the up to 4 genotype samples (2 bits each).

Usage

```
dbd(B, N, quiet = TRUE)
```

Arguments

`B`	byte data in R "raw" mode
`N`	number of individuals in the byte data.
`quiet`	do not report (def=TRUE)

Details

The function decodes bytes read from a BED to allele dosage or NA.

Value

a N x P matrix of genotype, where P is the number of variants.

`gid` *Infer Sample ID from a symmetric matrix*

Description

Exam the row name for family and individual id.

Usage

```
gid(x, sep = ".")
```

Arguments

`x`	matrix
`sep`	separator between FID and IID forming the sample ID

Details

For matrices without rowname, id are automatically generated.

By common practice, the row names or a matrix are in the form of [FID.]IID. Samples without family ID are given one identical to their individual ID.

Value

data.frame of inferred family ID and individual ID.

`lc` *Line Count*

Description

Count the occurance of '\n', much faster than `readLine` and `read.table`. Although slower than the unix command "wc -l", it upholds platform independency.

Usage

```
lc(f)
```

Arguments

`f`	the file name, or a connection.

`readBED` *Read BED file*

Description

Read a BED file into a R matrix. This is meant for in-of-memory process of moderate to small sized genotype.

Usage

```
readBED(pfx, row = NULL, col = NULL, quiet = TRUE)
```

Arguments

pfx	prefix of PLINK file set, or the fullname of a BED file.
row	the row names: 1 = use individual ID, 2 = family and individual ID, def = NULL.
col	the column names: 1 = use variant ID (i.e., rsID), 2 = CHR:POS, 3 = CHR:POS_A1_A2
quiet	suppress screen printing? (def=TRUE)

Details

To scan a huge BED one varant at time without reading it into the memoty, see `scanBED` instead.

A BED (*binary biallelic genotype table*) is comprised of three files (usually) sharing identical prefix:

- pfx.fam: table of N typed individuals
- pfx.bim: table of P typed genomic variants (i.e., SNPs);
- pfx.bed: genotype matrix of N rows and P columns stored in condensed binary format.

The three files are commonly referred by their common prefix, e.g.:

chrX.bed, chrX.fam, and chrX.bim, are jointly specified by "chrX".

Value

genotype matrix with row individuals and column variants.

See Also

readBED

Examples

```
bed <- system.file("extdata", 'm20.bed', package="plinkFile")
pfx <- sub("[.]bed$", "", bed)
bed <- readBED(pfx, quiet=FALSE)
```

`readBIM` *Read BIM file*

Description

Read BIM file

Usage

```
readBIM(pfx)
```

Arguments

`pfx`	prefix of a PLINK file set.

Value

data frame describing genome variants, loaded from the BIM file.

`readBSM` *Read Binary Symmetric Matrix (BSM)*

Description

Read BSM represented by a pair of files suffixed by ".bin" and ".id", usually produced by PLINK and GCTA.

Usage

```
readBSM(pfx, dgv = 1, fid = NULL, id = NULL, bin = NULL)
```

Arguments

`pfx`	prefix of data files pfx.id and pfx.bin
`dgv`	diagonal value for matrix without a diagonal (def=1.0)
`fid`	separator between FID and IID (def=NULL, use IID only)
`id`	use id file instead of the default `{pfx}.id`
`bin`	use bin file instead of the default `{pfx}.bin`

Details

The ".bin" is a binary file storing the matrix entries, which can be

- the N x N symmetric matrix in full
- the lower triangle with diagonal
- the lower triangle w/o diagonal

, saved as either single or double precision.

The ".id" a text file of family ID (FID) and individual ID (IID) in two columns. by default, IID is used as matix row and column names.

PLINK option `--make-red bin`, `--distance bin`, and GCTA option `--make-grm` all creats binary symmetric matrices, widely used in linear mixed model or kernel based models for genetics.

Value

symmetric matrix loaded from file, with sample ID in the row and column names.

Examples

```
pfx <- file.path(system.file("extdata", package="plinkFile"), "m20.rel")
(readBSM(pfx, fid=":"))
```

`readFAM` *Read FAM file*

Description

Read FAM file

Usage

```
readFAM(pfx)
```

Arguments

`pfx` prefix of a PLINK file set.

Value

data frame describing individuals, loaded from the FAM file.

Examples

```
pfx <- file.path(system.file("extdata", package="plinkFile"), "m20")
bed <- readBED(pfx, row=1, col=1, quiet=FALSE)
bed
```

readGRM — *Read Genetic Related Matrix (GRM) of GCTA*

Description

GRM is the core formt of GCTA, which is an binary symmetric matrix with an extra variant count matrix (VCM), this function reads the binary sysmmetric matrix.

Usage

```
readGRM(pfx, fid = ".")
```

Arguments

`pfx`	prefix of GRM file set
`fid`	separator after family ID (def=NULL, use IID only)

Details

GCTA GRM is represented by a set of three files:

- .grm.bin :GRM matrix in binary
- .grm.id :sample FID and IID in text
- .grm.N.bin :number of valid variants for each GRM entry

and it always uses single precision (4 bytes per entry).

To read the extra the extra VCM (grm.N.bin), use `readVCM`.

Value

matrix of relatedness with sample ID in row and column names.

Examples

```
pfx <- file.path(system.file("extdata", package="plinkFile"), "m20")
(readGRM(pfx))
```

readIBS *Read PLINK Binary IBS matrix*

Description

A PLINK IBS (Identity by State) matrix is represented by

- .mibs.bin:IBS matrix in binary
- .mibs.id :FID and IID in text

A binary IBS matrix is the result of PLINK `--distance ibs bin`

Usage

```
readIBS(pfx, fid = ".")
```

Arguments

`pfx`	prefix of the IBS file set.
`fid`	seperate after family ID (def=NULL, use IID only)

Value

IBS matrix with row and column names set to sample ID.

Examples

```
pfx <- file.path(system.file("extdata", package="plinkFile"), "m20")
(readIBS(pfx))
```

readREL *Read PLINK Binary REL matrix*

Description

A PLINK REL (Relatedness) matrix is represented by

- .rel.bin:REL matrix in binary
- .rel.id :FID and IID in text

A binary REL matrix is the result of PLINK `--make-rel bin`

Usage

```
readREL(pfx, fid = ".")
```

Arguments

pfx	prefix of the REL file set
fid	separate after family ID. (def=NULL, use IID only)

Value

relatedness matrix with row and column names set to sample ID.

Examples

```
pfx <- file.path(system.file("extdata", package="plinkFile"), "m20")
(readREL(pfx))
```

readVCM *Read Variant Count Matrix (VCM) accompanying a GCTA GRM*

Description

GRM (Genetic Relatedness Matrix) is the core formt of GCTA, which is a PLINK binary symmetric matrix with an extra variant count matrix (VCM), this function reads the VCM.

Usage

```
readVCM(pfx, fid = NULL)
```

Arguments

pfx	prefix of GRM file set
fid	seperate after family ID (def=NULL, use IID only)

Value

matrix of variant count with sample ID in row and column names.

Examples

```
pfx <- file.path(system.file("extdata", package="plinkFile"), "m20")
(readVCM(pfx))
```

saveBSM *Save Symmetric Matrix to Binary*

Description

Save symmetric matrix to a binary core file (.bin), and a text file of IDs (.id), recognizable by PLINK.

Usage

```
saveBSM(pfx, x, ltr = TRUE, diag = TRUE, unit = 4L, fid = ".")
```

Arguments

`pfx`	prefix of output files
`x`	symmetric matrix to save
`ltr`	store the lower triangle only? (def=TRUE)
`diag`	save diagnal? (def=TRUE) ignored if `ltr` is FALSE.
`unit`	numerical unit, (def=4, single precision)
`fid`	separator between FID and IID (def=".").

Examples

```
pfx <- file.path(system.file("extdata", package="plinkFile"), "m20.rel")
rel <- readBSM(pfx)  # relatedness kernel matrix
re2 <- rel^2         # 2nd order polynomial kernel

tmp <- tempdir()
dir.create(tmp, FALSE)
out <- file.path(tmp, 'm20.re2')
saveBSM(out, re2)    # save the polynomial kernel
dir(tmp)             # show new files, then clean up
unlink(tmp, recursive=TRUE)
```

saveGRM *Save symmetic matrix to GCTA GRM format.*

Description

GRM (Genetic Relatedness Matrix) is the core formt of GCTA, this function saves a R symmetric matrix to a file set recgnizable by GCTA.

Usage

```
saveGRM(pfx, grm, vcm = NULL, fid = ".")
```

Arguments

pfx	prefix of data files
grm	genome relatedness matrix to save
vcm	variant counts matrix to save (def=1).
fid	separator after family ID. (def=".")

Details

Three files will be saved:

- .grm.bin :genetic relatedness matrix in binary
- .grm.id :FID and IID for N individuals in text
- .grm.N.bin :variant count matrix (VCM) in binary

FID and IID will be generated if the grm to be saved has no row names.

When save the vcm, if a single number is given, this number is used as the variant count for all entries in the GRM.

saveGRM is useful in exporting customized kinship matrices (such as a Gaussian or a Laplacian kernel) to a GRM acceptable by GCTA, which are not supported by GCTA's own GRM builder.

Examples

```
pfx <- file.path(system.file("extdata", package="plinkFile"), "m20")
gmx <- readBED(pfx)  # read genotype matrix from PLINK BED.
gmx <- scale(gmx)    # standardize
tmp <- tempdir()     # for example outputs
dir.create(tmp, FALSE)

# kinship matrix as Gaussian kernel, built from the first 10 variants
gmx.gau <- gmx[, +(1:10)]                 # the first 10 variants
not.na.gau <- tcrossprod(!is.na(gmx.gau)) # variant count matrix
kin.gau <- exp(as.matrix(-dist(gmx.gau, "euc")) / not.na.gau)
print(kin.gau)                            # the Gaussian kernel
out.gau <- file.path(tmp, "m20.gau")
saveGRM(out.gau, kin.gau, not.na.gau)     # gau.grm.* should appear

# kinship matrix as Laplacian kernel, built without the first 10 variants
gmx.lap <- gmx[, -(1:10)]                 # drop the first 10 variants
not.na.lap <- tcrossprod(!is.na(gmx.lap)) # variant count matrix
kin.lap <- exp(as.matrix(-dist(gmx.lap, "man")) / not.na.lap)
out.lap <- file.path(tmp, "m20.lap")
print(kin.lap)                            # the Laplacian kernel
saveGRM(out.lap, kin.lap, not.na.lap)     # lap.grm.* should appear

# merge kinship in R language for a radius based function kernel matrix
not.na.rbf <- not.na.gau + not.na.lap
kin.rbf <- (kin.gau * not.na.gau + kin.lap * not.na.lap) / not.na.rbf
print(kin.rbf)
out.rbf <- file.path(tmp, "m20.rbf")
```

```
saveGRM(out.rbf, kin.rbf, not.na.rbf)     # rbf.grm.* should appear

# show saved matrices, then clean up
dir(tmp, "(gau|lap|rbf)")
unlink(tmp, recursive=TRUE)
```

scanBED *Scan genotypes in PLINK BED(s)*

Description

Go through a BED file set and visit one variant at a time. This is meant for out-of-memory screening of huge genotype, such as a GWAS study.

Usage

```
scanBED(pfx, FUN, ..., simplify = TRUE)
```

Arguments

pfx	prefix of PLINK BED.
FUN	the function to process each variant.
...	additional argument to pass to FUN.
simplify	TRUE to simplify the result as an array.

Details

To read an entire BED into a R matrix, see readBED instead.

A BED (*binary biallelic genotype table*) is comprised of three files (usually) sharing identical prefix:

- pfx.fam: table of N typed individuals
- pfx.bim: table of P typed genomic variants (i.e., SNPs);
- pfx.bed: genotype matrix of N rows and P columns stored in condensed binary format.

The three files are commonly referred by their common prefix, e.g.:

chrX.bed, chrX.fam, and chrX.bim, are jointly specified by "chrX".

Value

an array with each row corresponding to a variant if simplify is set to TRUE; otherwise, a list with each element corresponding to a variant is returned.

A context vaiable ".i" is assigned to the environment of FUN, therefore, one can access the index of variant current being processed from within the body of FUN.

See Also

readBED

Examples

```
pfx <- file.path(system.file("extdata", package="plinkFile"), "000")
ret <- scanBED(pfx, function(g)
{
    af <- mean(g, na.rm=TRUE) / 2
    maf <- min(af, 1 - af)
    c(idx=.i, mu=mean(g, na.rm=TRUE), maf=maf, nas=sum(is.na(g)))
})
print(ret[1:5, ])
```

`testReadBED` *Test BED Reader*

Description

Read m20 (bed, bim, and fam) under "extdata" and compare with the content in text file "i10.txt" converted from m20 by PLINK.

Usage

```
testReadBED()
```

`testReadBSM` *Test Genetic Relatedness Matrix Reader*

Description

Compare the read from genetic relatedness matrix created from the same genome segment but stored in different shapes and types.

Usage

```
testReadBSM()
```

testScanBED *Test BED Scanner*

Description

Go through file set "000" under "extdata", summerize every SNP.

Usage

```
testScanBED()
```

Index