GGtools

April 19, 2009

`gwSnpTests`	*methods for iterating association tests (expression vs SNP) across genomes or chromosomes*

Description

methods for iterating association tests (expression vs SNP) across genomes or chromosomes

Usage

```
gwSnpTests(sym, sms, cnum, ...)
```

Arguments

`sym`	genesym, probeId, or formula instance
`sms`	smlSet instance
`cnum`	chrnum instance or missing
`...`	...

Details

invokes snpMatrix test procedures as appropriate

gwSnpScreen is deprecated and simply throws a message indicating this.

Value

gwSnpScreenResult or cwSnpScreenResult instance

Author(s)

Vince Carey <stvjc@channing.harvard.edu>

Examples

```
if (!exists("hmceuB36.2021")) data(hmceuB36.2021)
# condense to founders only
hmFou = hmceuB36.2021[, which(hmceuB36.2021$isFounder)]
# show basic formula fit
f1 = gwSnpTests(genesym("CPNE1")~male, hmFou, chrnum(20))
f1
plot(f1)
# show how to avoid adjusted fit
f1b = gwSnpTests(genesym("CPNE1")~1-1, hmFou, chrnum(20))
# show gene set modeling on chromosome
library(GSEABase)
gs1 = GeneSet(c("CPNE1", "ADA"))
geneIdType(gs1) = SymbolIdentifier()
f2 = gwSnpTests(gs1~male, hmFou, chrnum(20))
f2
names(f2)
plot(f2[["ADA"]])
# show 'smlSet-wide' fit
f3 = gwSnpTests(gs1~male, hmFou)
f3
```

---

hbTestResults-class

Class "hbTestResults" holds results of tests of association of expression levels with haplotype within haplotype block

---

Description

Class "hbTestResults" holds results of tests of association of expression levels with haplotype within haplotype block

Objects from the Class

Objects can be created by calls of the form `new("hbTestResults", ...)`.

Slots

hscores: Object of class `"list"` series of haplo.stats:::haplo.score results for blocks

locs: Object of class `"numeric"` locations at which blocks were found (mean location within each block)

chrnum: Object of class `"chrnum"` chromosome being analyzed

smlSetName: Object of class `"character"` name of the smlSet harboring data in use

rsid: Object of class `"ANY"` can be a dbSNP id to use as an anchor, or a number constituting absolute chromosomal location at which blocks will be sought

rad: Object of class `"numeric"` radius in base pairs around the `rsid` to be searched for blocks

ldStruc: Object of class `"ANY"` the result of the mapLD:::mapLD function

Methods

pvals `signature(x = "hbTestResults")`: extracts p-values for global score tests, one per block

locs `signature(x = "hbTestResults")`: extracts locations of haplotype blocks found (average SNP location within block)

hscores `signature(x = "hbTestResults")`: extracts `haplo.score` results as a list, for all blocks

Author(s)

VJ Carey <stvjc@channing.harvard.edu>

Examples

```
showClass("hbTestResults")
```

`hbTests-methods` *haplotype-block based tests for structured expression variation*

Description

haplotype-block based tests for structured expression variation

Methods

fmla = "genesym", sms = "smlSet", cnum = "chrnum", rsid = "numeric", rad = "numeric" expression data for gene identified by `genesym` is extracted from `sms`, and genotype data within `rad` base pairs of `rsid` are obtained and processed by `mapLD` to define haplotype blocks and the SNP tagging these blocks. Score tests are then computed for the association of expression of the gene identified by `genesym` with haplotype copy number (additive model by default, but options captured by ... are passed to `haplo.score`.)

Examples

```
library(GGtools)
data(hmceuB36.2021)
hmFou = hmceuB36.2021[, hmceuB36.2021$isFounder==TRUE]
hh = hbTests(genesym("CPNE1"), hmFou, chrnum(20), 33600000, 2e4 )
hh
pvals(hh)
plot(locs(hh), -log10(pvals(hh)))
hscores(hh)[[which.min(pvals(hh))]]
```

hmceuB36.2021 *two chromosomes of genotype data and full expression data for CEPH CEU hapmap data*

Description

two chromosomes of genotype data and full expression data for CEPH CEU hapmap data

Usage

```
data(hmceuB36.2021)
```

Format

The format is: Formal class 'smlSet' [package "GGBase"] with 9 slots ..@ smlEnv :<environment: 0x3902e98> ..@ annotation : chr "illuminaHumanv1.db" ..@ chromInds : num [1:2] 20 21 ..@ organism : chr "Hs" ..@ assayData :<environment: 0x3c96504> ..@ phenoData :Formal class 'AnnotatedDataFrame' [package "Biobase"] with 4 slots ..@ featureData :Formal class 'AnnotatedDataFrame' [package "Biobase"] with 4 slots ..@ experimentData :Formal class 'MIAME' [package "Biobase"] with 13 slots ..@ ...classVersion..:Formal class 'Versions' [package "Biobase"] with 1 slots

Examples

```
#data(hmceuB36.2021)
```

invokePhase-methods *~~ Methods for Function invokePhase in Package 'GGtools' ~~*

Description

~~ Methods for function `invokePhase` in Package 'GGtools' ~~

Methods

x = "snp.matrix", cnum = "chrnum", parmstring = "character", globpname = "character", where2run = "chara
transform snp.matrix entity to phaseInput (uses tempfile()) and invokes PHASE

x = "phaseInput", cnum = "chrnum", parmstring = "character", globpname = "character", where2run = "char
for prepared 'phaseInput' structure, invoke PHASE

Examples

```
## Not run:
data(smtest)
invokePhase(smtest, chrnum(20), "", Sys.getenv("PHASE_LOC"),
   ".", TRUE)
## End(Not run)
```

plot-methods *~~ Methods for Function plot in Package 'GGtools' ~~*

Description

~~ Methods for function `plot` in Package 'GGtools' ~~

Methods

x = "cwSnpScreenResult", y = "missing" shows results of chromosome-wide screen for expression-associated SNP

x = "filteredGwSnpScreenResult", y = "ANY" shows results of genome-wide screen for expression-associated SNP

x = "filteredMultiGwSnpScreenResult", y = "ANY" fails, need to pick gene at this time

snpm2mapLD *prepare input to mapLD function for haplotype block identification*

Description

prepare input to mapLD function for haplotype block identification

Usage

```
snpm2mapLD(x, chrnum, runMAP=TRUE, ...)
```

Arguments

`x`	snp.matrix instance
`chrnum`	chromosome number
`runMAP`	logical indicating whether or not to run mapLD
`...`	additional arguments to mapLD

Details

sets up a data frame suitable for mapLD, and will invoke with appropriate arguments identifying columns for alleles and other identifiers if runMAP is TRUE (default).

smtest is a small snp.matrix instance

Value

a list with element struc holding the data frame, and mapLD output if requested. Note that mapLD writes an eps file to disk *sigh*.

Author(s)

Vince Carey <stvjc@channing.harvard.edu>

Examples

```
data(smtest)
ss = snpm2mapLD(smtest, chrnum=20, runMAP=FALSE)
ss
# you could run mapLD on ss[[1]]
```

`snpm2phase` *convert information in a snp.matrix to PHASE input format; invokePhase can run a suitably installed version of PHASE*

Description

convert information in a snp.matrix to PHASE input format; invokePhase can run a suitably installed version of PHASE

Usage

```
snpm2phase(snpm, cnum, outfilename)
parsePh.out(fn)
personalHap(x)
```

Arguments

`snpm`	snp.matrix instance
`cnum`	chromosome number as chrnum instance
`outfilename`	character name of file to write
`fn`	character name of PHASE .out file to read
`x`	output of parsePh.out

Details

follows phase 2.1 documentation for input format

a phaseInput container class can store relevant metadata

Value

writes to a file and gives a message

Author(s)

Vince Carey <stvjc@channing.harvard.edu>

Examples

```
data(smtest)
tt = tempfile()
pin = snpm2phase(smtest, chrnum(20), tt)

class(pin)
getClass("phaseInput")
pin
readLines(tt)
unlink(tt)
pp = parsePh.out(system.file("phaseOut/cpne1_20k.out", package="GGtools"))
pp[[1]][1:3]
personalHap(pp)
```

`strMultPop` *serialization of a table from Stringer's multipopulation eQTL report*

Description

serialization of a table from Stringer's multipopulation eQTL report

Usage

```
data(strMultPop)
```

Format

A data frame with 39649 observations on the following 12 variables.

`rsid` a factor with levels rs...

`genesym` a factor with levels `37865` `39692` `ABC1` `ABCD2` `ABHD4` `ACAS2` ...

`illv1pid` a factor with levels `GI_10047105-S` `GI_10092611-A` `GI_10190705-S` `GI_10567821-S` `GI_10835118-S` `GI_10835186-S` ...

`snpChr` a numeric vector

`snpCoordB35` a numeric vector

`probeMidCoorB35` a numeric vector

`snp2probe` a numeric vector

`minuslog10p` a numeric vector

`adjR2` a numeric vector

`assocGrad` a numeric vector

`permThresh` a numeric vector

`popSet` a factor with levels `CEU-CHB-JPT` `CEU-CHB-JPT-YRI` `CHB-JPT`

Details

imported from the PDF(!) distributed by Stranger et al as supplement to PMID 17873874

Source

PMID 17873874 supplement

References

PMID 17873874 supplement

Examples

```
data(strMultPop)
strMultPop[1:2,]
```

`topSnps-methods` *report on most significant SNP with gwSnpTests results*

Description

report on most significant SNP with gwSnpTests results

Methods

x = "cwSnpScreenResult" also takes argument n for number to report

x = "gwSnpScreenResult" also takes argument n for number to report

---

`GGtools-trackSet-methods`
create a browser track from a chromosome-wide SNP screen

Description

create a browser track from a chromosome-wide SNP screen

Methods

object = "cwSnpScreenResult" returns a track set with genomic coordinates for x axis and -log10 p-values for y axis

Index