GGBase – infrastructure for GGtools, genetics of gene expression

Vincent J. Carey, stvjc at channing.harvard.edu

May 02, 2019

Contents

1 NOTA BENE: IF STARTING ANEW, USE gQTLBase/gQTLstats

This package was published in the dawn of eQTL analysis. It uses somewhat idiosyncratic data structures. gQTL* packages are more up to date.

2 Introduction

The GGBase package defines infrastructure for analysis of data on the genetics of gene expression. This document is primarily of concern to developers; for information on conducting analyses in genetics of expression, please see the vignette for the GGtools package.

3 Primary class structure, and associated methods

`smlSet` is used to denote "SNP matrix list" integrative container for expression plus genotype data. The `SnpMatrix` class is defined in Clayton's *snpStats* package.

```
library(GGBase)
## Loading required package: snpStats
## Loading required package: survival
## Loading required package: Matrix
getClass("smlSet")
## Class "smlSet" [package "GGBase"]
##
## Slots:
##
## Name:              smlEnv         annotation           organism
## Class:        environment          character          character
##
## Name:           assayData          phenoData        featureData
## Class:          AssayData AnnotatedDataFrame AnnotatedDataFrame
##
## Name:      experimentData       protocolData  .__classVersion__
## Class:              MIAxE AnnotatedDataFrame           Versions
##
## Extends:
## Class "eSet", directly
## Class "VersionedBiobase", by class "eSet", distance 2
## Class "Versioned", by class "eSet", distance 3
showMethods(class="smlSet", where="package:GGBase")
## Function: [ (package base)
## x="smlSet", i="ANY", j="ANY", drop="ANY"
##
## Function: clipPCs (package GGBase)
## x="smlSet", inds2drop="numeric", center="logical"
## x="smlSet", inds2drop="numeric", center="missing"
```

```
##
## Function: combine (package BiocGenerics)
## x="smlSet", y="smlSet"
##
## Function: exprs (package Biobase)
## object="smlSet"
##
## Function: nsFilter (package genefilter)
## eset="smlSet"
##
## Function: permEx (package GGBase)
## sms="smlSet"
##
## Function: plot_EvG (package GGBase)
## gsym="genesym", rsid="rsid", sms="smlSet"
## gsym="probeId", rsid="rsid", sms="smlSet"
##
## Function: smList (package GGBase)
## x="smlSet"
```

Genotype data are stored in a list in the `smlEnv` environment to diminish copying as functions are called on the `smlSet` instance.

4 Example data structure

Expression data were published by the Wellcome Trust GENEVAR project in 2007. Genotype data are from HapMap phase II.

```
if ("GGtools" %in% installed.packages()[,1]) {
 library(GGtools)
 s20 = getSS("GGtools", "20")
 s20
}
## Loading required package: data.table
## Loading required package: parallel
## Loading required package: Homo.sapiens
## Loading required package: AnnotationDbi
## Loading required package: stats4
## Loading required package: BiocGenerics
##
## Attaching package: 'BiocGenerics'
## The following objects are masked from 'package:parallel':
##
##     clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
##     clusterExport, clusterMap, parApply, parCapply, parLapply,
##     parLapplyLB, parRapply, parSapply, parSapplyLB
## The following object is masked from 'package:Matrix':
##
##     which
```

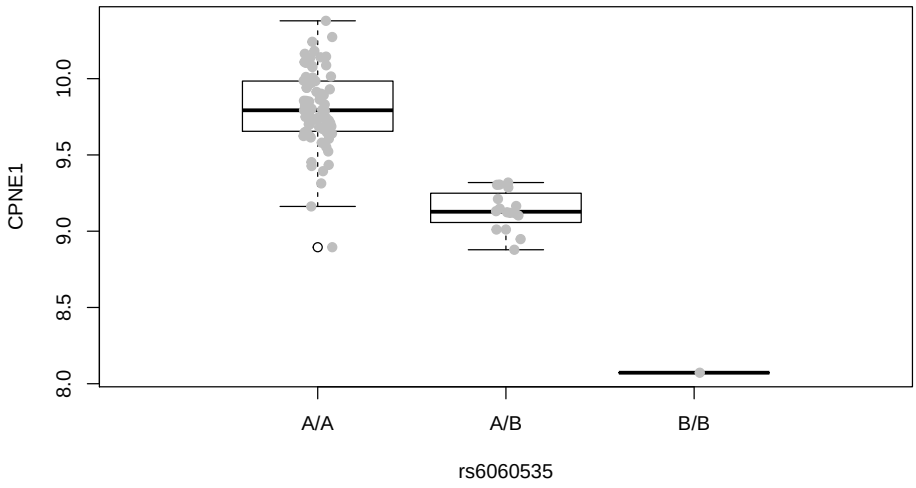
```
## The following objects are masked from 'package:stats':
##
##     IQR, mad, sd, var, xtabs
## The following objects are masked from 'package:base':
##
##     Filter, Find, Map, Position, Reduce, anyDuplicated, append,
##     as.data.frame, basename, cbind, colnames, dirname, do.call,
##     duplicated, eval, evalq, get, grep, grepl, intersect,
##     is.unsorted, lapply, mapply, match, mget, order, paste, pmax,
##     pmax.int, pmin, pmin.int, rank, rbind, rownames, sapply,
##     setdiff, sort, table, tapply, union, unique, unsplit, which,
##     which.max, which.min
## Loading required package: Biobase
## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.
## Loading required package: IRanges
## Loading required package: S4Vectors
##
## Attaching package: 'S4Vectors'
## The following objects are masked from 'package:data.table':
##
##     first, second
## The following object is masked from 'package:Matrix':
##
##     expand
## The following object is masked from 'package:base':
##
##     expand.grid
##
## Attaching package: 'IRanges'
## The following object is masked from 'package:data.table':
##
##     shift
## Loading required package: OrganismDbi
## Loading required package: GenomicFeatures
## Loading required package: GenomeInfoDb
## Loading required package: GenomicRanges
## Loading required package: GO.db
##
## Loading required package: org.Hs.eg.db
##
## Loading required package: TxDb.Hsapiens.UCSC.hg19.knownGene
## Registered S3 methods overwritten by 'ggplot2':
##   method         from
##   [.quosures     rlang
##   c.quosures     rlang
##   print.quosures rlang
##
```

```
## Attaching package: 'GGtools'
## The following object is masked from 'package:stats':
##
##     getCall
## SnpMatrix-based genotype set:
## number of samples:  90
## number of chromosomes present:  1
## annotation: illuminaHumanv1.db
## Expression data dims: 47293 x 90
## Total number of SNP: 119921
## Phenodata: An object of class 'AnnotatedDataFrame'
##   sampleNames: NA06985 NA06991 ... NA12892 (90 total)
##   varLabels: famid persid ... male (7 total)
##   varMetadata: labelDescription
```

5 Visualizing a specific gene-SNP relationship

The SNP rs6060535 was reported as an eQTL for CPNE1 by Cheung et al in a Nature paper of 2005.

```
if (exists("s20")) {
 plot_EvG(genesym("CPNE1"), rsid("rs6060535"), s20)
} else plot(1) # pdf must exist....
##
```



6 Genotype representations

The `SnpMatrix` class of the *snpStats* package is used to represent genotypes. Imputed genotypes and their uncertainties can be represented in this scheme, but the example does not depict this.

```
if (exists("s20")) {
# raw bytes
 as(smList(s20)[[1]], "matrix")[1:5,1:5]
# generic calls
 as(smList(s20)[[1]], "character")[1:5,1:5]
# risk allele (alphabetically later nucleotide) counts
 as(smList(s20)[[1]], "numeric")[1:5,1:5]
}
##         rs4814683 rs6076506 rs6139074 rs1418258 rs7274499
## NA06985         2         2         2         2         2
## NA06991         1         2         1         1         2
## NA06993         0         2         0         0         2
## NA06994         0         2         0         0         2
## NA07000         2         2         2         2         2
```

7 Reducing memory footprint of integrative data structures

When millions of genotypes are recorded, it can be cumbersome to work with all simultaneously in memory, and it is seldom scientifically relevant to do so. Thus a packaging protocol has been established in conjunction with the `getSS` function to allow chromosome-at-a-time loading of genotype data in conjunction with expression data.

To deploy the packaging protocol, use the `externalize` function on a "one-time" full smlSet representation of the data, or mimic the behavior of this function by creating a new package folder structure and populating the inst/parts with rda files representing a partition (usually by chromosome) of the genotype SnpMatrix instances.