SNPlocs.Hsapiens.dbSNP.20100427
March 26, 2015

getSNPlocs Accessing the data stored in SNPlocs.Hsapiens.dbSNP.20100427

Description
The data sets stored in SNPlocs.Hsapiens.dbSNP.20100427 and how to access them.

Usage
## Datasets:
data(SNPcount)
data(all_rsid)
data(ch1.snplocs)
data(ch2.snplocs)
data(ch3.snplocs)
data(ch4.snplocs)
data(ch5.snplocs)
data(ch6.snplocs)
data(ch7.snplocs)
data(ch8.snplocs)
data(ch9.snplocs)
data(ch10.snplocs)
data(ch11.snplocs)
data(ch12.snplocs)
data(ch13.snplocs)
data(ch14.snplocs)
data(ch15.snplocs)
data(ch16.snplocs)
data(ch17.snplocs)
data(ch18.snplocs)
data(ch19.snplocs)
data(ch20.snplocs)
data(ch21.snplocs)
data(ch22.snplocs)
getSNPlocs

data(chX_snplocs)
data(chY_snplocs)
data(chMT_snplocs)

## Convenience wrappers for loading the SNP data:
getSNPcount()
getSNPlocs(seqname, as.GRanges=FALSE, caching=TRUE)

## Extract SNP information for a set of rs ids:
rsid2loc(rsids, caching=TRUE)
rsid2alleles(rsids, caching=TRUE)
rsidsToGRanges(rsids, caching=TRUE)

Arguments

seqname  The name of the sequence for which to get the SNP locations and alleles.
  If as.GRanges is FALSE, only one sequence can be specified (i.e. seqname must
  be a single string). If as.GRanges is TRUE, an arbitrary number of sequences
  can be specified (i.e. seqname can be a character vector of arbitrary length).

as.GRanges  TRUE or FALSE. If TRUE, then the SNP locations and alleles are returned in a
  GRanges object. Otherwise (the default), they are returned in a data frame (see
  below).

caching  Should the loaded SNPs be cached in memory for faster further retrieval but at
  the cost of increased memory usage?

rsids  A vector of rs ids. Can be integer or character vector, with or without the "rs"
  prefix. NAs are not allowed.

Details

See SNPlocs.Hsapiens.dbSNP.20100427 for general information about this package.

The SNP data are split by chromosome i.e. the package contains one data set per chromosome, each
of them being a serialized data frame with 1 row per SNP and the 2 following columns:

- loc: The 1-based location of the SNP relative to the first base at the 5' end of the plus strand
  of the reference sequence.
- alleles: A raw vector with no NAs which can be converted into a character vector contain-
  ing the alleles for each SNP represented by an IUPAC nucleotide ambiguity code (see
  ?IUPAC_CODE_MAP in the Biostrings package for more information).

Note that those data sets are not intended to be used directly but the user should instead use the
getSNPcount and getSNPlocs convenience wrappers for loading the SNP data. When used with
as.GRanges=FALSE (the default), getSNPlocs returns a data frame with 1 row per SNP and the 3
following columns:

- RefSNP_id: RefSNP ID (aka "rs id") with "rs" prefix removed. Character vector with no
  NAs and no duplicates.
- alleles_as_ambig: A character vector with no NAs containing the alleles for each SNP
  represented by an IUPAC nucleotide ambiguity code.
- loc: Same as for the 2-col serialized data frame described previously.
getSNPlocs

Value

getSNPcount returns a named integer vector containing the number of SNPs for each sequence in the reference genome.

By default (as.GRanges=FALSE), getSNPlocs returns the 3-col data frame described above containing the SNP data for the specified chromosome. Otherwise (as.GRanges=TRUE), it returns a GRanges object with extra columns "RefSNP_id" and "alleles_as_ambig". Note that all the elements (genomic ranges) in this GRanges object have their strand set to * and that all the sequence lengths are set to NA.

rsid2loc and rsid2alleles both return a named vector (integer vector for the former, character vector for the latter) where each (name, value) pair corresponds to a supplied rs id. For both functions the name in (name, value) is the chromosome of the rs id. The value in (name, value) is the position of the rs id on the chromosome for rsid2loc, and a single IUPAC code representing the associated alleles for rsid2alleles.

rsidsToGRanges returns a GRanges object similar to the one returned by getSNPlocs (when used with as.GRanges=TRUE) and where each element corresponds to a supplied rs id.

Author(s)

H. Pages

See Also

SNPlocs.Hsapiens.dbSNP.20100427, IUPAC_CODE_MAP, GRanges-class, injectSNPs

Examples

```r
## A. BASIC USAGE
## -----------------------------------------------

getSNPcount()

## Get the locations and alleles of all SNPs on chromosomes 22:
ch22snps <- getSNPlocs("ch22")
dim(ch22snps)
colnames(ch22snps)
head(ch22snps)

## Get the locations and alleles of all SNPs on chromosomes 22 and MT
## as a GRanges object:
getSNPlocs(c("ch22", "chMT"), as.GRanges=TRUE)

## B. EXTRACT SNP INFORMATION FOR A SET OF RS IDS...
## -----------------------------------------------

## ... and return it in a GRanges object:
myrsids <- c("rs2639606", "rs75264089", "rs73396229", "rs55871206",
"rs10932221", "rs56219727", "rs73709730", "rs55838886",
"rs3734153", "rs79381275", "rs75350930", "rs1516535")
```
getSNPlocs

rsidsToGRanges(myrsids)

## C. INJECTION IN THE REFERENCE GENOME

library(BSgenome.Hsapiens.UCSC.hg19)
Hsapiens

## Note that the chromosome names in BSgenome.Hsapiens.UCSC.hg19
## are those used by UCSC and they differ from those used by dbSNP.

## Inject the SNPs in hg19 (injectSNPs() "knows" how to map dbSNP
## chromosome names to UCSC names):
Hs2 <- injectSNPs(Hsapiens, "SNPlocs.Hsapiens.dbSNP.20100427")
Hs2

## Get the number of nucleotides that were modified by this injection:

## D. SOME BASIC QUALITY CONTROL (WITH SURPRISING RESULTS!)

## Note that dbSNP can assign distinct ids to SNPs located at the same
## position: any(duplicated(ch22snps$RefSNP_id)) # rs ids are all distinct...
any(duplicated(ch22snps$loc)) # but some locations are repeated!
ch22snps <- ch22snps[order(ch22snps$loc), ] # sort by location
which(duplicated(ch22snps$loc))[1] # 639
ch22snps[637:640, ] # rs75258394 and rs78180314 have same locations

## Also note that not all SNP alleles are consistent with the hg19 genome
## i.e. the alleles reported for a given SNP are not always compatible
## with the nucleotide found at the SNP location in hg19.
## For example, to get the number of inconsistent SNPs in chr1:
ch1snps <- getSNPlocs("chr1")
all_alleles <- paste(ch1snps$alleles_as_ambig, collapse="")
nchar(all_alleles) # 1369185 SNPs on chr1
neditAt(all_alleles, unmasked(Hsapiens$chr1)[ch1snps$loc], fixed=FALSE)

## Finally, let’s check that no SNP falls in an assembly gap:
agaps <- masks(Hsapiens$chr1)$AGAPS
agaps # the assembly gaps

## Looping over the assembly gaps:
sapply(1:length(agaps),
    function(i)
        any(ch1snps$loc >= start(agaps)[i] &
            ch1snps$loc <= end(agaps)[i]))

## Or, in a more efficient way:
The SNPlocs.Hsapiens.dbSNP.20100427 package

Description

This package contains SNP locations and alleles for Homo sapiens extracted from dbSNP BUILD 131.

Details

SNPs from dbSNP were filtered to keep only those satisfying the 3 following criteria:

- The SNP is a single-base substitution i.e. its type is "snp". Other types used by dbSNP are: "in-del", "mixed", "microsatellite", "named-locus", "multinucleotide-polymorphism", etc... All those SNPs were dropped.
- The SNP is marked as not withdrawn.
- A single location on the reference genome (GRCh37) is reported for the SNP.

Note

The source data files used for this package were created by NCBI on 24-26 March 2010.

WARNING: The SNPs in this package are based on the GRCh37 genome. Note that the GRCh37 genome is the same as the hg19 genome from UCSC except for the mitochondrion chromosome. Therefore, the SNPs in this package can be "injected" in BSgenome.Hsapiens.UCSC.hg19 but this injection will exclude chrM (i.e. nothing will be injected in that sequence).

See http://www.ncbi.nlm.nih.gov/snp, the SNP Home at NCBI, for more information about dbSNP.

See ?injectSNPs in the BSgenome software package for more information about the SNP injection mechanism.

See http://genome.ucsc.edu/cgi-bin/hgGateway?clade=mammal&org=Human&db=hg19 for more information about the Human Feb. 2009 (GRCh37/hg19) assembly used by the UCSC Genome Browser.

Author(s)

H. Pages

References


About the Human Feb. 2009 (GRCh37/hg19) assembly used by the UCSC Genome Browser: http://genome.ucsc.edu/cgi-bin/hgGateway?clade=mammal&org=Human&db=hg19
See Also

getSNPlocs for how to access the data stored in this package.
injectSNPs in the BSgenome package for more information about SNP injection.
Index

*Topic data
  getSNPlocs.1
*Topic package
  getSNPlocs.1
    SNPlocs.Hsapiens.dbSNP.20100427.5
  .loadAlleles(getSNPlocs), 1
  .loadLoc (getSNPlocs), 1

all_rsids (getSNPlocs), 1

ch10_snplocs (getSNPlocs), 1
ch11_snplocs (getSNPlocs), 1
ch12_snplocs (getSNPlocs), 1
ch13_snplocs (getSNPlocs), 1
ch14_snplocs (getSNPlocs), 1
ch15_snplocs (getSNPlocs), 1
ch16_snplocs (getSNPlocs), 1
ch17_snplocs (getSNPlocs), 1
ch18_snplocs (getSNPlocs), 1
ch19_snplocs (getSNPlocs), 1
ch1_snplocs (getSNPlocs), 1
ch20_snplocs (getSNPlocs), 1
ch21_snplocs (getSNPlocs), 1
ch22_snplocs (getSNPlocs), 1
ch2_snplocs (getSNPlocs), 1
ch3_snplocs (getSNPlocs), 1
ch4_snplocs (getSNPlocs), 1
ch5_snplocs (getSNPlocs), 1
ch6_snplocs (getSNPlocs), 1
ch7_snplocs (getSNPlocs), 1
ch8_snplocs (getSNPlocs), 1
ch9_snplocs (getSNPlocs), 1
chMT_snplocs (getSNPlocs), 1
chX_snplocs (getSNPlocs), 1
chY_snplocs (getSNPlocs), 1
COMPATIBLE_BSGENOMES
  (SNPlocs.Hsapiens.dbSNP.20100427), 5

getSNPcount (getSNPlocs), 1