# Package 'snplist'

April 21, 2025

**Type** Package

**Title** Tools to Create Gene Sets

**Version** 0.18.3

**Date** 2025-04-21

**Author** Chanhee Yi [aut],
Alexander Sibley [aut, cre],
Kouros Owzar [aut]

**Maintainer** Alexander Sibley <dcibioinformatics@duke.edu>

**Description** A set of functions to create SQL tables of gene and SNP information and compose them into a SNP Set, for example to export to a PLINK set.

**License** GPL-3

**Depends** R (>= 3.0.0), RSQLite (>= 1.1)

**Imports** biomaRt (>= 2.16.0), Rcpp (>= 0.10.5), R.utils (>= 1.27.1),
DBI (>= 0.3.1), methods (>= 3.6.2)

**Suggests** knitr

**LinkingTo** Rcpp

**VignetteBuilder** knitr

**NeedsCompilation** yes

**Repository** CRAN

**Date/Publication** 2025-04-21 18:40:02 UTC

## Contents

---

| snplist-package | *Tools to Create Gene Sets* |

---

#### Description

A set of functions to create SQL tables of gene and SNP information and compose them into a SNP Set, for example for use with the `RSNPset` package, or to export to a PLINK set.

#### Details

|          |            |
|----------|------------|
| Package: | snplist    |
| Type:    | Package    |
| Version: | 0.18.3     |
| Date:    | 2025-04-21 |
| License: | GPL-3      |

Please see the example function calls below, or refer to the individual function documentation or the included vignette for more information.

#### Author(s)

Authors: Chanhee Yi, Alexander Sibley, and Kouros Owzar Maintainer: Alexander Sibley <alexander.sibley@dm.duke.edu>

#### See Also

`RSQLite`, `Rcpp`

#### Examples

```
chromosome <- c(1,5,22,"X","Y","MT")

geneNum <- 5
snpNum <- 1200
annoDataNum <- 500

chrLength <- 1000
geneLength <- 100

gene <- paste("gene",1:geneNum,sep="")
chr <- sample(chromosome,geneNum,replace=TRUE)
start <- sample(chrLength,geneNum,replace=TRUE)
d <- sample(geneLength,geneNum,replace=TRUE)
end <- start+d
geneInfo <- data.frame(gene,chr,start,end)

rsid <- paste("rs",1:snpNum,sep="")
```

```
chr <- sample(chromosome,snpNum,replace=TRUE)
pos <- sample(chrLength+geneLength,snpNum,replace=TRUE)
snpInfo <- data.frame(rsid,chr,pos)

annoInfo <- data.frame("rsid"=sample(rsid,annoDataNum))

dim(geneInfo)
dim(snpInfo)
dim(annoInfo)

## Not run:
setGeneTable(geneInfo)
setSNPTable(snpInfo)
geneset <- makeGeneSet(annoInfo)
exportPLINKSet(geneset,"geneSet.set")
file.show("geneSet.set")

## End(Not run)
```

---

exportPLINKSet           *exportPLINKSet*

---

### Description

Simple function using Rcpp to write the gene set to a file in the PLINK set format.

### Usage

```
exportPLINKSet(geneSets, fname)
```

### Arguments

geneSets        An object created by the makeGeneSet() function.

fname           The name of the PLINK file to be created.

### Value

A Boolean indicating if the file was successfully written.

### See Also

[makeGeneSet](makeGeneSet)

### Examples

```
# Please see the vignette or the package description
# for an example of using this function.
```

getBioMartData                    *getBioMartData*

---

### Description

A function leveraging the `biomaRt` package to retrieve gene chromosome and start and end positions from Ensembl.

### Usage

```
getBioMartData(genes,verbose=FALSE,...)
```

### Arguments

genes        A vector of gene names matching `hgnc_symbol` in the Ensembl database.

verbose      A Boolean indicating whether to output the funcitons progress in terms of the dimensions of the `data.frame` being constructed. Default is FALSE.

...          Additional arguments passed on to the internal call to `biomaRt::useMart(...)`.
             If no such arguments are provided, `useMart("ensembl", dataset="hsapiens_gene_ensembl")`
             is run by default.

### Value

A `data.frame` object with columns 'gene','chr','start', and 'end', suitable for input to the `setGeneTable` function.

### Note

At the time of package release, the BioMart community portal is temporarily unavailable. See www.biomart.org for updated status or more information. To access alternative hosts, pass additional arguments to the internal call to `biomaRt::useMart(...)`, as in the second example below.

### References

Durinck S., Spellman P.T., Birney E. and Huber W. (2009) Mapping identifiers for the integration of genomic datasets with the R/Bioconductor package biomaRt, *Nature Protocols*, **4**, 1184–1191.

### See Also

[setGeneTable](#)

## Examples

```
## Not run:
getBioMartData(c("BRCA1","BRCA2"))
getBioMartData(c("BRCA1","BRCA2"),
               host="www.ensembl.org",
               biomart="ENSEMBL_MART_ENSEMBL",
               dataset="hsapiens_gene_ensembl")

## End(Not run)
```

---

makeGeneSet                    *makeGeneSet*

---

## Description

This function uses existing SQLite tables (from setGeneTable and setSNPTable) to make SNP sets. The SNP Set for each gene is the collection of SNPs located either between the start and end locations of the gene, or within a specified neighborhood around the gene. The SNP Sets are stored in the SQLite database, and returned as a list object.

## Usage

```
makeGeneSet(annoInfo=NULL,margin=0,annoTable='anno',geneTable='gene',
                 allTable='allchrpos',db='snplistdb',dbCleanUp=FALSE)
```

## Arguments

| | |
|---|---|
| annoInfo | A vector of rsIDs, a data.frame with an 'rsid' column, or a file with one rsID per line. The SNP sets will be restricted to contain only the SNPs listed here. Default is NULL, in which case all SNPs present in the SNP table in the SQLite database will be used. |
| margin | A number, indicating the size of the neighborhood (in base pairs) surrounding a genes start and end positions in which a SNP will be included in that genes SNP set. Default is 0. |
| annoTable | A string indicating the name of the SQLite table for the rsIDs from annoInfo. Also used in naming the resulting table of SNP sets ('<name>ToGene'). Default is 'anno'. |
| geneTable | Name of the SQLite table containing chromosome, start and end positions for each gene, as previously created by setGeneTable. Default is 'gene'. |
| allTable | Name of the SQLite table containing chromosome and position for each SNP, as previously created by setSNPTable. Default is 'allchrpos'. |
| db | Name of the SQLite database in which to find the gene and SNP tables and create the SNP set table. Default is 'snplistdb'. |
| dbCleanUp | Boolean indicating if the tables and views created by the function should be dropped after the SNP set is returned. Default is FALSE. |

## Details

Note: This function relies on the prior execution of the `setGeneTable` and `setSNPTable` functions and the SQLite databes and tables they create. If the `table` or `db` argument in either of those functions is changed from the default value, it must also be changed here.

## Value

Returns a `list` of SNP sets of the form:

<gene name>        Vector of rsIDs of SNPs within <gene> (or the neighborhood around it)

## See Also

[setGeneTable](), [setSNPTable](), [snplist-package]()

## Examples

```
    # Please see the vignette or the package description
    # for an example of using this function.
```

---

setGeneTable                    *setGeneTable*

---

## Description

Takes a `data.frame` object with columns 'gene','chr','start', and 'end', and creates an SQLite table of the information. Returns a count of the number of genes in the table.

## Usage

```
    setGeneTable(geneInfo,table='gene',db='snplistdb')
```

## Arguments

| | |
|---|---|
| geneInfo | A `data.frame` object of gene location info with columns 'gene','chr','start', and 'end'. |
| table | Name of the SQLite table to be created. Default is 'gene'. |
| db | Name of the SQLite database in which to create `table`. Default is 'snplistdb'. |

## Value

Count of genes included in `table`.

## Examples

```
geneInfo <- cbind(c('BRCA1','BRCA2'),c(17,13),c(41196312,32889611),c(41277500,32973805))
 colnames(geneInfo) <- c('gene','chr','start','end')
 ## Not run:
 setGeneTable(as.data.frame(geneInfo))

## End(Not run)
```

---

setSNPTable *setSNPTable*

---

## Description

Takes a file or `data.frame` object with columns 'chr','pos', and 'rsid', and creates an SQLite table of the information. Returns a count of the number of SNPs in the table.

## Usage

```
setSNPTable(snpInfo,table='allchrpos',db='snplistdb')
```

## Arguments

snpInfo    A `data.frame` object of SNP location info with columns 'chr','pos', and 'rsid', or a tab-delimited file with those columns and one record per row.

table      Name of the SQLite table to be created. Default is 'allchrpos'.

db         Name of the SQLite database in which to create `table`. Default is 'snplistdb'.

## Value

Count of genes included in `table`.

## Examples

```
snpInfo <- cbind(c(17,17,13,13),
 c(41211653, 41213996, 32890026,32890572),
 c("rs8176273","rs8176265","rs9562605","rs1799943") )
colnames(snpInfo) <- c('chr','pos','rsid')
## Not run:
setSNPTable(as.data.frame(snpInfo))

## End(Not run)
```

# Index