Package 'msigdb'

April 24, 2025

Title An ExperimentHub Package for the Molecular Signatures Database (MSigDB)

Version 1.16.0

Description This package provides the Molecular Signatures Database (MSigDB) in a R accessible objects. Signatures are stored in GeneSet class objects form the GSEABase package and the entire database is stored in a GeneSetCollection object. These data are then hosted on the ExperimentHub. Data used in this package was obtained from the MSigDB of the Broad Institute. Metadata for each gene set is stored along with the gene set in the GeneSet class object.

biocViews ExperimentHub, Homo_sapiens_Data, Mus_musculus_Data

License CC BY 4.0

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Depends R (>= 4.1)

Imports ExperimentHub, utils, GSEABase, org.Mm.eg.db, org.Hs.eg.db, AnnotationDbi, methods, stats, AnnotationHub

Suggests singscore, vissE, knitr, prettydoc, BiocStyle, rmarkdown, testthat (>= 3.0.0), BiocFileCache, GO.db, stringr, limma

URL https://davislaboratory.github.io/msigdb

BugReports https://github.com/DavisLaboratory/msigdb/issues

NeedsCompilation no

VignetteBuilder knitr

Config/testthat/edition 3

git_url https://git.bioconductor.org/packages/msigdb

git_branch RELEASE_3_21

git_last_commit d73d327

git_last_commit_date 2025-04-15

Repository Bioconductor 3.21

Date/Publication 2025-04-24

Author Dharmesh D. Bhuva [aut, cre] (ORCID:
<https://orcid.org/0000-0002-6398-9157>),
Gordon K. Smyth [aut] (ORCID: <https://orcid.org/0000-0001-9221-2892>),
Alexandra Garnham [aut] (ORCID:
<https://orcid.org/0000-0002-8312-8450>)

Maintainer Dharmesh D. Bhuva <bhuva.d@wehi.edu.au>

Contents

appendKEGG 2
getIMEX 3
getIMEXVersions 3
getMsigdb 4
getMsigdbIDF 5
getMsigdbVersions 5
getMsigIdType 6
getMsigOrganism 6
listCollections 7
listSubCollections 8
msigdb 8
subsetCollection 10

Index **11**

`appendKEGG` *Add KEGG pathway derived gene sets to a collection*

Description

This function adds gene sets derived from KEGG pathways to the MSigDB data stored in this package. Direct user-end download from the MSigDB is required to ensure KEGG licenses are adhered to.

Usage

```
appendKEGG(gsc, version = getMsigdbVersions())
```

Arguments

`gsc`	a GeneSetCollection object, containing MSigDB genesets in the form of GeneSet objects.
`version`	a character, stating the version of MSigDB to be retrieved (should be >= 7.2). See `getMsigdbVersions()`.

Value

a GeneSetCollection object, storing gene sets from the MSigDB including the downloaded KEGG gene sets.

`getIMEX` *Retrieve IMEx PPI hosted on the hub*

Description

Download International Molecular Exchange (IMEx) protein-protein interaction (PPI) hosted on the ExperimentHub or retrieve pre-downloaded version from cache. This package currently hosts versions for human and mouse with both symbol and Entrez identifiers.

Usage

```
getIMEX(org = c("hs", "mm"), inferred = FALSE, version = getIMEXVersions())
```

Arguments

`org`	a character, representing the organism whose PPI database needs to be retrieved ("hs" for human and "mm" for mouse).
`inferred`	a logical, indicating whether inference from other organisms should be included in the PPI.
`version`	a character, stating the version of IMEX to be retrieved. See `getMsigdbVersions()`.

Value

a data.frame, containing the IMEx PPI.

Examples

```
imex = getIMEX("hs")
```

`getIMEXVersions` *Get IMEX versions included in the msigdb package*

Description

Get IMEX versions included in the msigdb package

Usage

```
getIMEXVersions()
```

Value

a character, stating the IMEX versions available in this package.

Examples

```
getIMEXVersions()
```

`getMsigdb` *Retrieve MSigDB data hosted on the hub*

Description

Download molecular signatures database (MSigDB) hosted on the ExperimentHub or retrieve pre-downloaded version from cache. This package currently hosts versions greater than 7.2 for human and mouse with both symbol and Entrez identifiers.

Usage

```
getMsigdb(
  org = c("hs", "mm"),
  id = c("SYM", "EZID"),
  version = getMsigdbVersions()
)
```

Arguments

`org`	a character, representing the organism whose signature database needs to be retrieved ("hs" for human and "mm" for mouse).
`id`	a character, representing the ID type to use ("SYM" for gene symbols and "EZID" for Entrez IDs).
`version`	a character, stating the version of MSigDB to be retrieved (should be >= 7.2). See `getMsigdbVersions()`.

Value

a GeneSetCollection, containing GeneSet objects from the specified version of the molecular signatures database (MSigDB).

Examples

```
gsc = getMsigdb('hs', 'SYM')
```

getMsigdbIDF *Retrieve MSigDB data hosted on the hub*

Description

Download molecular signatures database (MSigDB) hosted on the ExperimentHub or retrieve pre-downloaded version from cache. This package currently hosts versions greater than 7.2 for human and mouse with both symbol and Entrez identifiers.

Usage

```
getMsigdbIDF(org = c("hs", "mm"), version = getMsigdbVersions())
```

Arguments

org	a character, representing the organism whose signature database needs to be retrieved ("hs" for human and "mm" for mouse).
version	a character, stating the version of MSigDB to be retrieved (should be >= 7.2). See `getMsigdbVersions()`.

Value

a list of named numeric vectors, containing inverse document frequency (IDF) weights. Names represent terms that the IDF is computed for. IDFs are computed using gene-set names ("Name") and short descriptions ("Short").

Examples

```
gsc = getMsigdbIDF("hs")
```

getMsigdbVersions *Get MSigDB versions included in the msigdb package*

Description

Get MSigDB versions included in the msigdb package

Usage

```
getMsigdbVersions()
```

Value

a character, stating the MSigDB versions available in this package.

Examples

```
getMsigdbVersions()
```

`getMsigIdType` *Infer gene identifier type for the gene set collection*

Description

The gene identifier (Symbol or Entrez ID) of a gene set collection is inferred from the IDs present in the data. A collection should ideally store gene sets using a single identifier type. This function returns the identifier type (either SymbolIdentifier or EntrezIdentifier) of the collection. It returns an error if the identifier is neither of these.

Usage

```
getMsigIdType(gsc)
```

Arguments

`gsc`	a GeneSetCollection object, containing MSigDB genesets in the form of GeneSet objects.

Value

a GSEABase::SymbolIdentifier or GSEABASE::EntrezIdentifier object, specifying the gene identifier type (gene symbols or Entrez IDs respectively).

Examples

```
gsc <- getMsigdb()
id <- getMsigIdType(gsc)
```

`getMsigOrganism` *Infer organism type for the gene set collection*

Description

Since both Human and Mouse MSigDB collections are hosted in this package, this function infers the type of organism represented in a gene set collection based on the gene IDs present. If not all gene IDs belong to the same organism, the organism with more than 50% gene IDs present in the collection is returned. In any other case, the function returns an error.

Usage

```
getMsigOrganism(gsc, idType)
```

Arguments

gsc	a GeneSetCollection object, containing MSigDB genesets in the form of GeneSet objects.
idType	a GSEABase::SymbolIdentifier or GSEABASE::EntrezIdentifier object, representing the ID type inferred from the `getMsigIdType()` function. Avoid providing this manually.

Value

a character, either "mm" (representing Mus musculus - mouse) or "hs" (representing Homo sapiens - human).

Examples

```
gsc <- getMsigdb()
id <- getMsigIdType(gsc)
getMsigOrganism(gsc, id)
```

---

listCollections *List all collection types within a MSigDB gene set collection*

Description

This function lists all the collection types present in a MSigDB gene set collection. Descriptions of collections can be found at the MSigDB website.

Usage

```
listCollections(gsc)
```

Value

a character vector, containing character codes for all collections present in the GeneSetCollection object.

Examples

```
gsc = getMsigdb('hs', 'SYM')
listCollections(gsc)
```

listSubCollections *List all sub-collection types within a MSigDB gene set collection*

Description

This function lists all the sub-collection types present in a MSigDB gene set collection. Descriptions of sub-collections can be found at the MSigDB website.

Usage

```
listSubCollections(gsc)
```

Value

a character vector, containing character codes for all sub-collections present in the GeneSetCollection object.

Examples

```
gsc = getMsigdb('hs', 'SYM')
listSubCollections(gsc)
```

msigdb *The Molecular Signatures Database (MSigDB)*

Description

This ExperimentHub package contains gene expression signatures from the molecular signatures database (MSigDB) for versions >= 7.2. Collections for human and mouse are currently supported in this package. The mouse version was developed in conjunction with Gordon K Smyth and Alexandra Garnham, and reflects the collections available from WEHI (https://bioinf.wehi.edu.au/MSigDB/).

Format

A GeneSetCollection object composed of GeneSet objects representing all non-empty gene expression signatures from the molecular signatures database (MSigDB).

Details

The molecular signatures database (MSigDB) is a collection of over 25000 gene expression signatures that are grouped into collections and sub-collections. Metadata associated with signatures is collected and stored in the data in this package.

All data in this package are stored in a `GeneSetCollection` object from the `GSEABase` package. Each gene expression signature in the collection is stored in a `GeneSet` object from the `GSEABase` package. This data does not include KEGG gene sets due to copyrights. Users can download this data using functions provided in the package (see Details).

Data in this package does not include gene sets from the KEGG database due to licensing limitations. Users can use the `appendKEGG()` function in this package to download KEGG gene sets directly from the MSigDB and append to existing data objects.

The mouse MSigDB is created by translating human genes to mouse homologs using annotations from the Mouse Genome Informatics (MGI) database for most gene sets. Gene sets in the collections c1 (positional gene sets) and c5 (ontologies) are recreated as information in these gene sets is organism specific. Positional gene sets are created using gene information from NCBI. Gene sets representing gene ontologies are derived from the mouse R/Bioconductor organism database (org.Mm.eg.db).

Acknowledgement

MSigDB is protected by copyright © 2004-2020 Broad Institute, Inc., Massachusetts Institute of Technology, and Regents of the University of California. Use of MSigDB is subject to the terms and conditions of the Creative Commons Attribution 4.0 International License - https://creativecommons.org/licenses/by/4.0/.

MSigDB gene sets derived from BioCarta pathways are the subject of copyright © 2000-2017 BioCarta, and are subject to Biocarta's Disclaimer of Liability and of Warranties - https://data.broadinstitute.org/gsea-msigdb/msigdb/biocarta/biocarta_disclaimer_of_liability_and_of_warranties.txt

References

Subramanian, A., Tamayo, P., Mootha, V. K., Mukherjee, S., Ebert, B. L., Gillette, M. A., ... & Mesirov, J. P. (2005). Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. Proceedings of the National Academy of Sciences, 102(43), 15545-15550.

Liberzon, A., Subramanian, A., Pinchback, R., Thorvaldsdóttir, H., Tamayo, P., & Mesirov, J. P. (2011). Molecular signatures database (MSigDB) 3.0. Bioinformatics, 27(12), 1739-1740.

Liberzon, A., Birger, C., Thorvaldsdóttir, H., Ghandi, M., Mesirov, J. P., & Tamayo, P. (2015). The molecular signatures database hallmark gene set collection. Cell systems, 1(6), 417-425.

Examples

```
library(ExperimentHub)
eh <- ExperimentHub()
msigdb_datasets <- query(eh, "msigdb")

#load data using different approaches
```

```
msigdb <- getMsigdb('hs', 'SYM')
msigdb <- eh[["EH5421"]]
```

`subsetCollection` *Subset collections and sub-collections of MSigDB*

Description

The molecular signatures database (MSigDB) is composed of collections and sub-collection. Many analyses (e.g. gene-set enrichment using limma::fry) are best carried out within specific collections rather than across the entire database of signatures. This function allows subsetting of MSigDB data objects within this package using collection and sub-collection types.

Usage

```
subsetCollection(gsc, collection = c(), subcollection = c())
```

Arguments

`gsc`	a GeneSetCollection object, containing MSigDB genesets in the form of GeneSet objects.
`collection`	a character, stating the collection(s) to be retrieved. The collection(s) must be one from the `listCollections()` function.
`subcollection`	a character, stating the sub-collection(s) to be retrieved. The sub-collection(s) must be one from the `listSubCollections()` function.

Value

a GeneSetCollection object, containing gene sets belonging to the queries collection and/or sub-collection.

Examples

```
gsc = getMsigdb('hs', 'SYM')
subsetCollection(gsc, collection = "h")
```

Index

appendKEGG, 2
appendKEGG(), *9*

getIMEX, 3
getIMEXVersions, 3
getMsigdb, 4
getMsigdbIDF, 5
getMsigdbVersions, 5
getMsigIdType, 6
getMsigIdType(), *7*
getMsigOrganism, 6

listCollections, 7
listCollections(), *10*
listSubCollections, 8
listSubCollections(), *10*

msigdb, 8
msigdb-package *(*msigdb*)*, 8

subsetCollection, 10