# Package 'MotifDb'

April 28, 2025

**Type** Package

**Title** An Annotated Collection of Protein-DNA Binding Sequence Motifs

**Version** 1.50.0

**Date** 2025-03-12

**Author** Paul Shannon, Matt Richards

**Maintainer** Paul Shannon <pshannon@systemsbiology.org>

**Depends** R (>= 3.5.0), methods, BiocGenerics, S4Vectors, IRanges, GenomicRanges, Biostrings

**Suggests** RUnit, seqLogo, BiocStyle, knitr, rmarkdown, formatR, markdown

**Imports** rtracklayer, splitstackshape

**Description** More than 9900 annotated position frequency matrices from 14 public sources, for multiple organisms.

**License** Artistic-2.0 | file LICENSE

**License_is_FOSS** no

**License_restricts_use** yes

**LazyLoad** yes

**biocViews** MotifAnnotation

**VignetteBuilder** knitr, rmarkdown, formatR, markdown

**Encoding** UTF-8

**git_url** https://git.bioconductor.org/packages/MotifDb

**git_branch** RELEASE_3_21

**git_last_commit** 9b7c5d1

**git_last_commit_date** 2025-04-15

**Repository** Bioconductor 3.21

**Date/Publication** 2025-04-27

# Contents

associateTranscriptionFactors . . . . . 2
export . . . . . 3
geneToMotif . . . . . 4
MotifDb . . . . . 5
MotifList-class . . . . . 9
motifToGene . . . . . 9
query . . . . . 10
subset . . . . . 11

**Index** **13**

---

`associateTranscriptionFactors` *associateTranscriptionFactors*

---

## Description

In the analysis of, or exploration of gene regulatory networks, one often creates a data.frame of possible genomic regulatory sites, genomic locations where a TF binding motif matches some DNA sequence. A common next step is to associate each of these motifs with its related transcription factor/s. We provide two sources for those relationships. When you specify the "MotifDb" source, we return the motif/TF relationships provided by each of the constituent public MotifDb sources. When you specify the "TFClass" source, transcription factor family memberships (described in https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4383905/) are - sometimes expansively - provided for each motif you supply.

This method uses, and therefore expects, different columns of the incoming data.frame to be used with each method. The MotifDb source uses the "motifName" column of the incoming data.frame. The TFClass source expects a "shortName" column in the incoming database.

A new column, "geneSymbol", is added to the incoming data.frame. This new column identifies the transcription factor associated with the motif for each row in the data.frame.

## Usage

```
## S4 method for signature 'MotifList'
associateTranscriptionFactors(object, tbl.withMotifs, source, expand.rows, motifColumnName="motifName
```

## Arguments

| | |
|---|---|
| `object` | a `MotifList` object. |
| `tbl.withMotifs` | a `data.frame` |
| `source` | a `character` string, either "MotifDb" or "TFClass" (case insensitive) |
| `expand.rows` | a `logical` value, recommended especially for the TFClass source, in which sometimes many TFs are mapped to the same motif |
| `motifColumnName` | a `character` string identifying the column in tbl.withMotifs which contains the motifs to be associated with transcription factors |

## Value

A data.frame with one column ("geneSymbol") and possibly multiple rows added

## Author(s)

Paul Shannon

## See Also

MotifDb, geneToMotif, motifToGene, subset, query

## Examples

```
tbl.tfClassExample <- data.frame(motifName=c("MA0006.1", "MA0042.2", "MA0043.2"),
                                 chrom=c("chr1", "chr1", "chr1"),
                                 start=c(1000005, 1000085, 1000105),
                                 start=c(1000013, 1000092, 1000123),
                                 score=c(0.85, 0.92, 0.98),
                                 stringsAsFactors=FALSE)
    # here we illustrate how to add a column with the required name:
tbl.tfClassExample$shortMotif <- tbl.tfClassExample$motifName
tbl.out <- associateTranscriptionFactors(MotifDb, tbl.tfClassExample, source="TFClass",
                                         expand.rows=TRUE)
dim(tbl.out)   #    MANY tfs mapped, mostly FOX family genes
tbl.motifDbExample <- data.frame(motifName=c("Mmusculus-jaspar2016-Ahr::Arnt-MA0006.1",
                                             "Hsapiens-jaspar2016-FOXI1-MA0042.2",
                                             "Hsapiens-jaspar2016-HLF-MA0043.2"),
                                 chrom=c("chr1", "chr1", "chr1"),
                                 start=c(1000005, 1000085, 1000105),
                                 start=c(1000013, 1000092, 1000123),
                                 score=c(0.85, 0.92, 0.98),
                                 stringsAsFactors=FALSE)

tbl.out <- associateTranscriptionFactors(MotifDb, tbl.motifDbExample, source="MotifDb",
                                         expand.rows=TRUE)
 dim(tbl.out)   # one new column ("geneSymbol"), no new rows
```

---

`export` *export*

---

## Description

Writes all matrices in the supplied list, in the specified format, to the specified connection.

### Usage

```
## S4 method for signature 'MotifList,connection,character'
export(object, con, format, ...)
## S4 method for signature 'MotifList,character,character'
export(object, con, format, ...)
## S4 method for signature 'MotifList,missing,character'
export(object, con, format, ...)
```

### Arguments

| | |
|---|---|
| object | a MotifList object. |
| con | either a file connection or a filename or missing, implying stdout. |
| format | a character string, currently only 'meme' and 'transfac', which both produce the same result |
| ... | ignore this |

### Value

The matrices list is written to the specified connection in the specified format.

### Author(s)

Paul Shannon

### See Also

MotifDb, query, subset, flyFactorSurvey, hPDI, jaspar, ScerTF, uniprobe

### Examples

```
library (MotifDb)
  # identify all the SOX genes
sox.indices = grep ('^sox', values (MotifDb)$geneSymbol, ignore.case=TRUE)
matrices = MotifDb [sox.indices]
export (matrices, con='SoxGenes-meme.txt', format='meme')
```

---

## geneToMotif *geneToMotif*

---

### Description

Using either of our two sources ("MotifDb" or "TFClass") retrieve the names of the transcription factor binding motifs associated with the gene symbol for each transcription factor. Slightly different information is returned in each case but the columns "geneSymbol", "motif", "pubmedID", "source" are returned by both sources. The TFClass source is described here: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4383905/. The MotifDb source is in fact the usually 1:1 gene/motif mapping provided by each of the data sources upon which MotifDb is built.

### Usage

```
## S4 method for signature 'MotifList'
geneToMotif(object, geneSymbols, source, ignore.case)
```

### Arguments

| | |
|---|---|
| object | a MotifList object. |
| geneSymbols | a character string |
| source | a character string, either 'MotifDb' or 'TFclass' (case insensitive) |
| ignore.case | a logical variable, default FALSE, guiding gene name matching |

### Value

A data.frame with these columns: geneSymbol, motif, pubmedID, source. The MotifDb source alos include dataSource and organism.

### Author(s)

Paul Shannon

### See Also

MotifDb, motifToGene, associateTranscriptionFactors, subset, query

### Examples

```
genes <- c("ATF5", "FOS")
geneToMotif(MotifDb, genes, source="TFClass")
geneToMotif(MotifDb, genes, source="MotifDb")
```

---

## MotifDb *MotifDb: An Annotated Collection of DNA-binding sequence motifs*

---

### Description

Approximately 2000 position frequency matrices collected from public sources, with ample accompanying metadata, and search and export capabilities provided.

### Details

MotifDb is an R object of class MotifList, whose entries are numeric matrices, accompanied by a 'parallel' metadata structure, a DataFrame, in which each row provides information about the corresponding matrix. This object is automatically created and fully populated by data from five public sources (see below) when the package is loaded into your R environment via the library call. The matrices are obtained from six public sources:

| | |
|---|---|
| FlyFactorSurvey: | 614 |
| hPDI: | 437 |
| JASPAR_CORE: | 459 |
| jolma2013: | 843 |
| ScerTF: | 196 |
| stamlab: | 683 |
| UniPROBE: | 380 |
| cisbp 1.02 | 874 |

Representing primarily five organsisms (and 49 total):

| | |
|---|---|
| Hsapiens: | 2328 |
| Dmelanogaster: | 1008 |
| Scerevisiae: | 701 |
| Mmusculus: | 660 |
| Athaliana: | 160 |
| Celegans: | 44 |
| other: | 177 |

All the matrices are stored as position frequency matrices, in which each columm (each position) sums to 1.0. When the number of sequences which contributed to the motif are known, that number will be found in the matrix's metadata. With this information, one can transform the matrices into either PCM (position count matrices), or PWM (position weight matrices), also known as PSSM (position-specific-scoring matrices). The latter transformation requires that a model of the background distribution be known, or assumed.

The names of the matrices are the same as rownames of the metadata DataFrame, and have been chosen to balance the needs of concision and full description, including the organism in which the motif was discovered, the data source, and the name of the motif in the data source from which it was obtained. For example: "Hsapiens-JASPAR_CORE-SP1-MA0079.2" and "Scerevisiae-ScerTF-GSM1-badis".

Subsets of the Matrices may be obtainted in several ways:

- By integer index, eg, `MotifDb [[1]]`
- By query, eg, `as.list (query (MotifDb, 'FBgn0000014'))`
- (Interactively only) by subset `as.list (subset (MotifDb, geneSymbol=='Abda' & !is.na (pubmedID)))`

The matrices are stored in a `SimpleList` which has semantics very similar to the familiar list of R base. To examine a matrix, however, you must sidestep the MotifDb show method. These three commands display quite different results:

```
> MotifDb [1]
MotifDb object of length 1
| Created from downloaded public sources: 2012-Jul6
| 1 position frequency matrices from 1 source:
```

```
|    FlyFactorSurvey:    1
| 1 organism/s
|      Dmelanogaster:    1
Dmelanogaster-FlyFactorSurvey-ab_SANGER_10_FBgn0259750

> MotifDb [[1]]
   1    2    3    4 5 6 7 8 9   10   11   12   13   14   15   16   17   18   19   20   21
A 0.0 0.50 0.20 0.35 0 0 1 0 0 0.55 0.35 0.05 0.20 0.45 0.20 0.10 0.40 0.40 0.25 0.50 0.30
C 0.3 0.15 0.25 0.00 1 1 0 0 0 0.10 0.65 0.70 0.45 0.25 0.10 0.25 0.25 0.10 0.10 0.25 0.25
G 0.4 0.05 0.50 0.65 0 0 0 1 1 0.00 0.00 0.05 0.05 0.15 0.05 0.20 0.05 0.15 0.55 0.15 0.45
T 0.3 0.30 0.05 0.00 0 0 0 0 0 0.35 0.00 0.20 0.30 0.15 0.65 0.45 0.30 0.35 0.10 0.10 0.00

> as.list (MotifDb [1])
$`Dmelanogaster-FlyFactorSurvey-ab_SANGER_10_FBgn0259750`
   1    2    3    4 5 6 7 8 9   10   11   12   13   14   15   16   17   18   19   20   21
A 0.0 0.50 0.20 0.35 0 0 1 0 0 0.55 0.35 0.05 0.20 0.45 0.20 0.10 0.40 0.40 0.25 0.50 0.30
C 0.3 0.15 0.25 0.00 1 1 0 0 0 0.10 0.65 0.70 0.45 0.25 0.10 0.25 0.25 0.10 0.10 0.25 0.25
G 0.4 0.05 0.50 0.65 0 0 0 1 1 0.00 0.00 0.05 0.05 0.15 0.05 0.20 0.05 0.15 0.55 0.15 0.45
T 0.3 0.30 0.05 0.00 0 0 0 0 0 0.35 0.00 0.20 0.30 0.15 0.65 0.45 0.30 0.35 0.10 0.10 0.00
```

There are fifteen kinds of metadata – though not all matrices have a full complement: not all of the public sources are complete in this regard. The information falls into these categories, using the *Dmelanogaster-FlyFactorSurvey-ab_SANGER_10_FBgn0259750* entry as an example (see below for the associated position frequency matrix):

1. providerName: "ab_SANGER_10_FBgn0259750"
2. providerId: "FBgn0259750"
3. dataSource: "FlyFactorSurvey"
4. geneSymbol: "Ab"
5. geneId: "FBgn0259750"
6. geneIdType: "FLYBASE"
7. proteinId: "E1JHF4"
8. proteinIdType: "UNIPROT"
9. organism: "Dmelanogaster"
10. sequenceCount: 20
11. bindingSequence: NA
12. bindingDomain: NA
13. tfFamily: NA
14. experimentType: "bacterial 1-hybrid, SANGER sequencing"
15. pubmedID: NA

## References


- Neph S, Stergachis AB, Reynolds A, Sandstrom R, Borenstein E, Stamatoyannopoulos JA. Circuitry and dynamics of human transcription factor regulatory networks. Cell. 2012 Sep 14;150(6):1274-86.
- Portales-Casamar E, Thongjuea S, Kwon AT, Arenillas D, Zhao X, Valen E, Yusuf D, Lenhard B, Wasserman WW, Sandelin A. JASPAR 2010: the greatly expanded open-access database of transcription factor binding profiles. Nucleic Acids Res. 2010 Jan;38(Database issue):D105-10. Epub 2009 Nov 11.
- Robasky K, Bulyk ML. UniPROBE, update 2011: expanded content and search tools in the online database of protein-binding microarray data on protein-DNA interactions. Nucleic Acids Res. 2011 Jan;39(Database issue):D124-8. Epub 2010 Oct 30.
- Spivak AT, Stormo GD. ScerTF: a comprehensive database of benchmarked position weight matrices for Saccharomyces species. Nucleic Acids Res. 2012 Jan;40(Database issue):D162-8. Epub 2011 Dec 2.
- Xie Z, Hu S, Blackshaw S, Zhu H, Qian J. hPDI: a database of experimental human protein-DNA interactions. Bioinformatics. 2010 Jan 15;26(2):287-9. Epub 2009 Nov 9.
- Zhu LJ, et al. 2011. FlyFactorSurvey: a database of Drosophila transcription factor binding specificities determined using the bacterial one-hybrid system. Nucleic Acids Res. 2011 Jan;39(Database issue):D111-7. Epub 2010 Nov 19.
- Jolma A, et al. 2013. DNA-binding specificities of human transcription factors. Cell 2013 Jan 17.


## See Also

query, subset, export, flyFactorSurvey, hPDI, jaspar, ScerTF, uniprobe

## Examples

```
  # are there any matrices for Sox4?  we find two
mdb.sox4 <- MotifDb [grep ('sox4', values (MotifDb)$geneSymbol, ignore.case=TRUE)]
  # the same two matrices can be obtained this way also
if (interactive ())
  mdb.sox4 <- subset (MotifDb, tolower(geneSymbol)=='sox4')
  # and like this
mdb.sox4 <- query (MotifDb, 'sox4')  # matches against all fields in the metadata
  # implicitly invoke the 'show' method
mdb.sox4
  # get their full names
names (mdb.sox4)
  # examine their metadata
values (mdb.sox4)
  # examine the matrices with names include
as.list (mdb.sox4)
  # export the matrices in meme format
destination.file = tempfile ()
export (mdb.sox4, destination.file, 'meme')
```

## MotifList-class *MotifList*

### Description

A direct subclass of SimpleList, having no extra slots, in which listData is a list of position frequency matrices (PFMs), and the elementMetadata slot is a DataFrame with fifteen columns describing each matrix. Upon loading the MotifDb class, one MotifList object is instantiated and filled with matrices and their metadata. There should be no need for users to explicitly create objects of this class. When you load the MotifDb package, a fully-populated instance of this class is created, with > 2000 matrices with metadata

### Methods

`subset(x)`: extract matrices by metadata.

`export(x)`: write matrices

`show(x)`: describe matrices compactly

`query(x)`: find matrices

### Author(s)

Paul Shannon

### Examples

```
  # Examine the number of matrices contributed by each source.
print (table (values (MotifDb)$dataSource))
```

## motifToGene *motifToGene*

### Description

Using either of our two sources ("MotifDb" or "TFClass") this method retrieves the the transcription factor (its gene symbol) for each of the supplied motifs. Slightly different information is returned in each case but the columns "geneSymbol", "motif", "pubmedID", "source" are returned by both. The TFClass source is described here: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4383905/. The MotifDb source is in fact the (typically) 1:1 gene/motif mapping provided by each of the data sources upon which MotifDb is built.

### Usage

```
## S4 method for signature 'MotifList'
motifToGene(object, motifs, source)
```

### Arguments

| | |
|---|---|
| object | a MotifList object. |
| motifs | a character string |
| source | a character string, either 'MotifDb' or "TFclass' (case insensitive) |

### Value

A data.frame with these columns: geneSymbol, motif, pubmedID, source. The MotifDb source also include dataSource and organism.

### Author(s)

Paul Shannon

### See Also

MotifDb, geneToMotif, associateTranscriptionFactors, subset, query

### Examples

```
motifs <- c("MA0592.2", "ELF1.SwissRegulon", "UP00022")
motifToGene(MotifDb, motifs, source="TFClass")
motifToGene(MotifDb, motifs, source="MotifDb")
```

---

## query *query*

---

### Description

A very general search tool, returning all matrices whose metadata, in ANY column, is matched by the query string.

### Usage

```
## S4 method for signature 'MotifList'
query(object, andStrings, orStrings, notStrings, ignore.case=TRUE)
```

### Arguments

| | |
|---|---|
| object | a MotifList object. |
| andStrings | a character string vector, length one or more, every element of which must be found in the metadata |
| orStrings | a character string vector, length one or more, any element of which must be found in the metadata |
| notStrings | a character string vector, length one or more, none of which may appear in the metadata |
| ignore.case | a logical value, default TRUE |

### Value

A list of the matrices

### Author(s)

Paul Shannon

### See Also

MotifDb, subset, export, flyFactorSurvey, hPDI, jaspar, ScerTF, uniprobe

### Examples

```
 matrices.human <- query(MotifDb, 'hsapiens')
 matrices.sox4 <- query(MotifDb, 'sox4')
 uniprobe.sox.matrices <- query(MotifDb, c('uniprobe', 'sox'))
   # two approaches to selectinve extraction of TFEB matrices
tfeb.human.1 <- query(MotifDb, andStrings=c("TFEB", "hsapiens"), notStrings=c("hpdi", "jolma", "cisbp"))
tfeb.human.2 <- query(MotifDb, andStrings=c("TFEB", "hsapiens"), orStrings=c("hocomoco", "jaspar", "swissregulo
                      notStrings="2016")
```

---

## subset *subset*

---

### Description

An analog of the base package subset method, this version will return all the matrices whose metadata match the (possibly intricate) logical expression in the "subset" argument.

Note: just as with the base subset method, this method is unreliable except when used interactively. Batch, script or other programmatic use of this function is to be avoided.

### Usage

```
## S4 method for signature 'MotifList'
subset(x, subset, select, drop=FALSE, ...)
```

### Arguments

| | |
|---|---|
| `x` | a `MotifList` object. |
| `subset` | a `logical` expression whose terms are predicates on the column names of the metadata table |
| `select, drop, ...` | these are ignored, appearing here only in fidelity to the generic definition of the method. |

## Value

A list of the matrices whose metadata satisfies the supplied subset

## Author(s)

Paul Shannon

## See Also

MotifDb, query, export, flyFactorSurvey, hPDI, jaspar, ScerTF, uniprobe

## Examples

```
mdb <- MotifDb
if (interactive ()) {
  matrices <- subset (mdb, dataSource=='UniPROBE')
  egr1.matrices <- subset (mdb, geneSymbol=='Egr1')
  jaspar.egr1.matrices <- subset (mdb, geneSymbol=='Egr1' &
                                       dataSource == 'JASPAR_CORE')
    # one of the mouse egr1 matrices has a geneSymbol 'Zif268', but
    # has the proper entrez geneId.
  all.egr1.matrices <- subset (mdb, geneId=='13653')
  }
```

# Index


∗ **classes**
MotifList-class, 9
∗ **datasets**
MotifDb, 5
∗ **methods**
MotifList-class, 9
∗ **utilities**
associateTranscriptionFactors, 2
export, 3
geneToMotif, 4
motifToGene, 9
query, 10
subset, 11

associateTranscriptionFactors, 2
associateTranscriptionFactors,MotifList-method *(associateTranscriptionFactors)*, 2

class:MotifList *(MotifList-class)*, 9

export, 3
export,MotifList,character,character-method *(export)*, 3
export,MotifList,connection,character-method *(export)*, 3
export,MotifList,missing,character-method *(export)*, 3

geneToMotif, 4
geneToMotif,MotifList-method *(geneToMotif)*, 4

MotifDb, 5
MotifDb-package *(MotifDb)*, 5
MotifList-class, 9
motifToGene, 9
motifToGene,MotifList-method *(motifToGene)*, 9

query, 10
query,MotifList-method *(query)*, 10

show,MotifList-method *(MotifList-class)*, 9
subset, 11
subset,MotifList-method *(subset)*, 11