

# Package ‘homologene’

March 29, 2019

**Type** Package

**Title** Quick Access to Homologene and Gene Annotation Updates

**Version** 1.4.68.19.3.27

**Depends** R (>= 3.1.2)

**Imports** dplyr (>= 0.7.4), magrittr (>= 1.5), purrr (>= 0.2.5), readr (>= 1.3.1), R.utils (>= 2.8.0)

**Suggests** testthat (>= 1.0.2)

**Date** 2019-03-28

**BugReports** <https://github.com/oganm/homologene/issues>

**URL** <https://github.com/oganm/homologene>

**Description** A wrapper for the homologene database by the National Center for Biotechnology Information ('NCBI'). It allows searching for gene homologs across species. Data in this package can be found at <<ftp://ftp.ncbi.nih.gov/pub/HomoloGene/build68/>>. The package also includes an updated version of the homologene database where gene identifiers and symbols are replaced with their latest (at the time of submission) version and functions to fetch latest annotation data to keep updated.

**License** MIT + file LICENSE

**LazyData** true

**RoxygenNote** 6.1.1

**NeedsCompilation** no

**Author** Ogan Mancarci [aut, cre],  
Leon French [ctb]

**Maintainer** Ogan Mancarci <[ogan.mancarci@gmail.com](mailto:ogan.mancarci@gmail.com)>

**Repository** CRAN

**Date/Publication** 2019-03-28 23:10:03 UTC

## R topics documented:

autoTranslate . . . . .	2
getGeneHistory . . . . .	3
getGeneInfo . . . . .	3
getHomologene . . . . .	4
homologene . . . . .	5
homologeneData . . . . .	5
homologeneData2 . . . . .	6
homologeneVersion . . . . .	6
human2mouse . . . . .	7
mouse2human . . . . .	7
taxData . . . . .	8
updateHomologene . . . . .	8
updateIDs . . . . .	9

<b>Index</b>	<b>10</b>
--------------	-----------

---

autoTranslate	<i>Attempt to automatically translate a gene list</i>
---------------	---

---

### Description

Given a list of query gene list and a target gene list, the function tries find the homology pairing that matches the query list to the target list. The query list is a short list of genes while the target list is supposed to represent a large number of genes from the target species. The default output will be the largest possible list. If `returnAllPossible = TRUE` then all possible pairings with any matches are returned. It is possible to limit the search by setting `possibleOrigins` and `possibleTargets`. Note that gene symbols of some species are more similar to each other than others. Using this with small gene lists and without providing any `possibleOrigins` or `possibleTargets` might return multiple hits, or if `returnAllPossible = TRUE` a wrong match can be returned.

### Usage

```
autoTranslate(genes, targetGenes, possibleOrigins = NULL,
             possibleTargets = NULL, returnAllPossible = FALSE,
             db = homologene::homologeneData)
```

### Arguments

<code>genes</code>	A list of genes to match the target. Symbols or NCBI ids
<code>targetGenes</code>	The target list. This list is supposed to represent a large number of genes from the target species.
<code>possibleOrigins</code>	Taxonomic identifiers of possible origin species
<code>possibleTargets</code>	Taxonomic identifiers of possible target species

returnAllPossible  
                   if TRUE returns all possible pairings with non zero gene matches. If FALSE  
                   (default) returns the best match

db                Homologene database to use.

**Value**

A data frame if returnAllPossible = FALSE and a list of data frames if TRUE

---

getGeneHistory            *Download gene history file*

---

**Description**

Downloads and reads the gene history file from NCBI website. This file is needed for other functions

**Usage**

```
getGeneHistory(destfile = NULL, justRead = FALSE)
```

**Arguments**

destfile            Path of the output file. If NULL a temp file will be used

justRead            If TRUE and destfile exists, it reads the file instead of downloading the latest one from NCBI

**Value**

A data frame with latest gene history information

---

getGeneInfo             *Download gene symbol information*

---

**Description**

This function downloads the gene\_info file from NCBI website and returns the gene symbols for current IDs.

**Usage**

```
getGeneInfo(destfile = NULL, justRead = FALSE, chunk_size = 1e+06)
```

**Arguments**

<code>destfile</code>	Path of the output file. If NULL a temp file will be used
<code>justRead</code>	If TRUE and <code>destfile</code> exists, it reads the file instead of downloading the latest one from NCBI
<code>chunk_size</code>	Chunk size to be used with <code>link[readr]{read_tsv_chunked}</code> . The <code>gene_info</code> file is big enough to make its intake difficult. If you don't have large amounts of free memory you may have to reduce this number to read the file in smaller chunks

**Value**

A data frame with gene symbols for each current gene id

---

<code>getHomologene</code>	<i>Get the latest homologene file</i>
----------------------------	---------------------------------------

---

**Description**

This function downloads the latest homologene file from NCBI. Note that Homologene has not been updated since 2014 so the output will be identical to [homologeneData](#) included in this package. This function is here for futureproofing purposes.

**Usage**

```
getHomologene(destfile = NULL, justRead = FALSE)
```

**Arguments**

<code>destfile</code>	Path of the output file. If NULL a temp file will be used
<code>justRead</code>	If TRUE and <code>destfile</code> exists, it reads the file instead of downloading the latest one from NCBI

**Value**

A data frame with homology groups, gene ids and gene symbols

---

homologene	<i>Get homologues of given genes</i>
------------	--------------------------------------

---

**Description**

Given a list of genes and a taxid, returns a data frame including the genes and their corresponding homologues

**Usage**

```
homologene(genes, inTax, outTax, db = homologene::homologeneData)
```

**Arguments**

genes	A vector of gene symbols or NCBI ids
inTax	taxid of the species that the input genes are coming from
outTax	taxid of the species that you are seeking homology
db	Homologene database to use.

**Examples**

```
homologene(c('Eno2', '17441'), inTax = 10090, outTax = 9606)
```

---

homologeneData	<i>homologeneData</i>
----------------	-----------------------

---

**Description**

List of gene homologues used by homologene functions

**Usage**

```
homologeneData
```

**Format**

An object of class `data.frame` with 275237 rows and 4 columns.

---

homologeneData2	<i>homologeneData2</i>
-----------------	------------------------

---

**Description**

A modified copy of the homologene database. Homologene was updated at 2014 and many of its gene IDs and symbols are out of date. Here the IDs and symbols are replaced with their most current version Last update: Wed Mar 27 16:34:11 2019

**Usage**

homologeneData2

**Format**

An object of class data.frame with 269592 rows and 4 columns.

---

homologeneVersion	<i>Version of homologene used</i>
-------------------	-----------------------------------

---

**Description**

Version of homologene used

**Usage**

homologeneVersion

**Format**

An object of class integer of length 1.

---

human2mouse	<i>Human/mouse wrapper for homologue</i>
-------------	--

---

**Description**

Human/mouse wrapper for homologue

**Usage**

```
human2mouse(genes, db = homologue::homologueData)
```

**Arguments**

genes	A vector of gene symbols or NCBI ids
db	Homologue database to use.

**Examples**

```
human2mouse(c('EN02', '4340'))
```

---

mouse2human	<i>Mouse/human wrapper for homologue</i>
-------------	--

---

**Description**

Mouse/human wrapper for homologue

**Usage**

```
mouse2human(genes, db = homologue::homologueData)
```

**Arguments**

genes	A vector of gene symbols or NCBI ids
db	Homologue database to use.

**Examples**

```
mouse2human(c('Eno2', '17441'))
```

---

taxData	<i>Names and ids of included species</i>
---------	--

---

**Description**

Names and ids of included species

**Usage**

```
taxData
```

**Format**

An object of class `data.frame` with 21 rows and 2 columns.

---

updateHomologene	<i>Update homologene database</i>
------------------	-----------------------------------

---

**Description**

Creates an updated version of the homologene database. This is done by downloading the latest gene annotation information and tracing changes in gene symbols and identifiers over history. [homologeneData2](#) was created using this function over the original [homologeneData](#). This function requires downloading large amounts of data from the NCBI ftp servers.

**Usage**

```
updateHomologene(destfile = NULL,
  baseline = homologene::homologeneData2, gene_history = NULL,
  gene_info = NULL)
```

**Arguments**

<code>destfile</code>	Optional. Path of the output file.
<code>baseline</code>	The baseline homologene file to be used. By default uses the <a href="#">homologeneData2</a> that is included in this package. The more ids to update, the more time is needed for the update which is why the default option uses an already updated version of the original database.
<code>gene_history</code>	A gene history data frame, possibly returned by <a href="#">getGeneHistory</a> function. Use this if you want to have a static <code>gene_history</code> file to update up to a specific date. An up to date <code>gene_history</code> object can be set to update to a specific date by trimming rows that have recent dates. Note that the same is not possible for the <code>gene_info</code> If not provided, the latest file will be downloaded.



gene\_info      A gene info data frame that contains ID-symbol matches, possibly returned by [getGeneInfo](#). Use this if you want a static version. Should be in sync with the gene\_history file. Note that there is no easy way to track changes in gene symbols back in time so if you want to update it up to a specific date, make sure you don't lose that file.

**Value**

Homologene database in a data frame with updated gene IDs and symbols

---

updateIDs	<i>Update gene IDs</i>
-----------	------------------------

---

**Description**

Given a list of gene ids and gene history information, traces changes in the gene's name to get the latest valid ID

**Usage**

```
updateIDs(ids, gene_history)
```

**Arguments**

ids              Gene ids  
gene\_history     Gene history information, probably returned by [getGeneHistory](#)

**Value**

A character vector. New ids for genes that changed ids, or "-" for discontinued genes. the input itself.

**Examples**

```
## Not run:  
gene_history = getGeneHistory()  
updateIDs(c("4340964", "4349034", "4332470", "4334151", "4323831"),gene_history)  
  
## End(Not run)
```

# Index

## \*Topic **datasets**

- homologeneData, [5](#)
- homologeneData2, [6](#)
- homologeneVersion, [6](#)
- taxData, [8](#)

autoTranslate, [2](#)

getGeneHistory, [3](#), [8](#), [9](#)

getGeneInfo, [3](#), [9](#)

getHomologene, [4](#)

homologene, [5](#)

homologeneData, [4](#), [5](#), [8](#)

homologeneData2, [6](#), [8](#)

homologeneVersion, [6](#)

human2mouse, [7](#)

mouse2human, [7](#)

taxData, [8](#)

updateHomologene, [8](#)

updateIDs, [9](#)