Package 'yulab.utils'

November 17, 2026

Title Supporting Functions for Packages Maintained by 'YuLab-SMU'

Version 0.2.6

Description Miscellaneous functions commonly used by 'YuLab-SMU'.

Depends R (>= 4.2.0)

Imports cli, digest, fs, methods, rappdirs, rlang, tools, utils

Suggests httr2, jsonlite, openssl, testthat (>= 3.0.0)

ByteCompile true

License Artistic-2.0

URL https://yulab-smu.top/

BugReports https://github.com/YuLab-SMU/yulab.utils/issues

Encoding UTF-8

Config/testthat/edition 3

Config/roxygen2/version 8.1.1

NeedsCompilation no

Author Guangchuang Yu [aut, cre] (ORCID:
<https://orcid.org/0000-0002-6485-8781>)

Maintainer Guangchuang Yu <guangchuangyu@gmail.com>

Repository CRAN

Date/Publication 2026-11-17 16:00:02 UTC

Contents

as_chunked_array 2
c2 3
check_directory 4
check_file 4
check_input 5
check_packages 6
check_range 6

combinations . . . 7
CRANpkg . . . 7
download_yulab_file . . . 8
exec . . . 8
get_dependencies . . . 9
get_fun_from_pkg . . . 10
Githubpkg . . . 10
has_internet . . . 11
initial_cache . . . 12
install_zip . . . 13
install_zip_gh . . . 14
is.installed . . . 15
load_OrgDb . . . 15
ls2df . . . 16
mat2df . . . 16
mat2list . . . 17
mypkg . . . 17
o . . . 18
packageTitle . . . 19
parse_ratio . . . 19
pload . . . 20
quiet . . . 21
rbindlist . . . 21
read.cb . . . 22
scale_range . . . 22
scihub_dl . . . 23
set_PCRE . . . 23
show_in_excel . . . 24
str_detect . . . 25
str_extract . . . 25
str_starts . . . 26
str_wrap . . . 27
user_dir . . . 27
yread_tsv . . . 28
yulab_abort . . . 29
yulab_msg . . . 29

Index **31**

`as_chunked_array` *Convert to chunked_array*

Description

Convert a numeric or character vector to a chunked_array object. If the input is already a chunked_array, it is returned as-is.

Usage

```
as_chunked_array(x)
```

Arguments

x	a vector or chunked_array object

Value

A chunked_array object

---

c2 *Concatenate Chunked Arrays*

Description

Concatenate vectors into a chunked array

Usage

```
c2(x, y)
```

Arguments

x	a vector or chunked_array object
y	a vector or chunked_array object

Details

Combine two vector or chunked_array objects into a single chunked_array without copying the underlying data.

Value

A chunked_array object

Author(s)

Guangchuang Yu

Examples

```
a <- c2(1:100, 101:200)
length(a)
a[150]
```

check_directory *Check if directory exists and is accessible*

Description

Validates directory existence and accessibility with options to create if missing

Usage

```
check_directory(
  path,
  create_if_missing = FALSE,
  check_write_permission = TRUE,
  arg_name = "directory"
)
```

Arguments

path	Directory path
create_if_missing	Whether to create directory if it doesn't exist
check_write_permission	Whether to verify write permissions
arg_name	Name of the argument for error messages

Value

Invisible TRUE if valid, throws error otherwise

See Also

Other validate-utils: check_file(), check_input(), check_packages(), check_range()

check_file *Handle file operations with proper error messages*

Description

Enhanced file validation with comprehensive checks and better error messages

Usage

```
check_file(path, operation = "read", must_exist = TRUE)
```

Arguments

path	File path
operation	Operation being performed (read, write, etc.)
must_exist	Whether the file must exist

Value

Invisible TRUE if operation can proceed, throws error otherwise

See Also

Other validate-utils: check_directory(), check_input(), check_packages(), check_range()

---

check_input *Validate input with type/length constraints*

Description

Enhanced input validation supporting base types and class checks.

Usage

```
check_input(
  x,
  type = NULL,
  length = NULL,
  min_length = NULL,
  max_length = NULL,
  allow_null = FALSE,
  arg_name = "input"
)
```

Arguments

x	Object to check
type	Expected type (e.g., "numeric", "character", or class name)
length	Expected length
min_length	Minimum length
max_length	Maximum length
allow_null	Whether NULL is allowed
arg_name	Argument name for messages

Value

Invisible TRUE on success

See Also

Other validate-utils: check_directory(), check_file(), check_packages(), check_range()

---

check_packages *Check if required packages are installed with informative errors*

Description

Enhanced package checking with better error messages and validation

Usage

```
check_packages(packages, reason = "for this functionality")

check_pkg(packages, reason = "for this functionality")
```

Arguments

packages	Character vector of package names
reason	Reason why these packages are needed

Value

Invisible TRUE if all packages are available, throws error otherwise

See Also

Other validate-utils: check_directory(), check_file(), check_input(), check_range()

---

check_range *Check if value is within specified range*

Description

Validates that a numeric value falls within the specified range

Usage

```
check_range(x, min = NULL, max = NULL, inclusive = TRUE, arg_name = "value")
```

Arguments

x	Numeric value to check
min	Minimum allowed value (optional)
max	Maximum allowed value (optional)
inclusive	Whether bounds are inclusive (default: TRUE)
arg_name	Name of the argument for error messages

Value

Invisible TRUE if valid, throws error otherwise

See Also

Other validate-utils: `check_directory()`, `check_file()`, `check_input()`, `check_packages()`

`combinations` *combinations*

Description

all possible combinations of n sets

Usage

```
combinations(n)
```

Arguments

`n`	number of sets

Value

a list of all combinations

`CRANpkg` *Markdown link to CRAN/Bioconductor*

Description

Markdown link to CRAN/Bioconductor

Usage

```
CRANpkg(pkg)

Biocpkg(pkg)
```

Arguments

`pkg`	package name

Value

md text string

Author(s)

Guangchuang Yu

See Also

Other pkg-utils: `Githubpkg()`, `get_dependencies()`, `get_fun_from_pkg()`, `is.installed()`, `mypkg()`, `packageTitle()`, `pload()`

`download_yulab_file` *Process YuLab File Download*

Description

Process YuLab File Download

Usage

```
download_yulab_file(destfile, urls, gzfile = FALSE, appname = NULL)
```

Arguments

`destfile`	character. Local file path.
`urls`	character vector. Base URLs for download. Remote is is `urls/basename(destfile)`
`gzfile`	logical. Whether the remote file is gzipped.
`appname`	character. R package name.

Author(s)

Guangchuang Yu

`exec` *exec*

Description

run system command

Usage

```
exec(command)
```

Arguments

`command`	system command to run

Value

An exec instance that stores system command outputs

Author(s)

Guangchuang Yu

See Also

Other os-utils: has_internet(), user_dir()

---

get_dependencies *get_dependencies*

Description

Get reverse dependencies

Usage

```
get_dependencies(pkg, repo = c("CRAN", "BioC"))
```

Arguments

pkg	package name
repo	'CRAN' and/or 'BioC'

Value

reverse dependencies

Author(s)

Guangchuang Yu

See Also

Other pkg-utils: CRANpkg(), Githubpkg(), get_fun_from_pkg(), is.installed(), mypkg(), packageTitle(), pload()

get_fun_from_pkg *get_fun_from_pkg*

Description

load function from package

Usage

```
get_fun_from_pkg(pkg, fun)
```

Arguments

pkg	package
fun	function

Value

function

Author(s)

Guangchuang Yu

See Also

Other pkg-utils: CRANpkg(), Githubpkg(), get_dependencies(), is.installed(), mypkg(), packageTitle(), pload()

Examples

```
get_fun_from_pkg('utils', 'zip')
```

Githubpkg *Markdown link to GitHub*

Description

Markdown link to GitHub

Usage

```
Githubpkg(user, pkg)
```

Arguments

user	github user
pkg	package name

Value

md text string

Author(s)

Guangchuang Yu

See Also

Other pkg-utils: CRANpkg(), get_dependencies(), get_fun_from_pkg(), is.installed(), mypkg(), packageTitle(), pload()

---

has_internet *has_internet*

Description

test for internect connection via reading lines from a URL

Usage

```
has_internet(site = "https://www.baidu.com/")
```

Arguments

site	URL to test connection

Value

logical value

Author(s)

Guangchuang Yu

See Also

Other os-utils: exec(), user_dir()

initial_cache *Cache intermediate data*

Description

Utilities to cache intermediate data: initialize items, update items, remove items, and retrieve elements.

Usage

```
initial_cache()

get_cache()

rm_cache()

initial_cache_item(item)

get_cache_item(item)

rm_cache_item(item)

update_cache_item(item, elements, ttl = NULL)

get_cache_element(item, elements, default = NULL, prune_expired = TRUE)

prune_cache_item(item)

cache_list_items()

cache_size()

cache_save(path)

cache_load(path)

with_cache(item, key, compute, ttl = NULL)
```

Arguments

item	Cache item name
elements	Elements to cache
ttl	Time-to-live in seconds
default	Default value if cache element is missing
prune_expired	Logical, whether to prune expired items
path	File path to save or load cache

key	Element key
compute	Function to compute value when missing

Value

Cache environment, item, or selected elements

Examples

```
## Not run:
 slow_fib <- function(x) {
     if (x < 2) return(1)
     slow_fib(x-2) + slow_fib(x-1)
 }

 fast_fib <- function(x) {
     if (x < 2) return(1)
     res <- get_cache_element('fibonacci', as.character(x))
     if (!is.null(res)) {
         return(res)
     }
     res <- fast_fib(x-2) + fast_fib(x-1)
     e <- list()
     e[[as.character(x)]] <- res
     update_cache_item('fibonacci', e)
     return(res)
 }

 system.time(slow_fib(30))
 system.time(fast_fib(30))


## End(Not run)
```

---

install_zip *install_zip*

Description

install R package from zip file of source codes

Usage

```
install_zip(file, subdir = NULL, args = "--no-build-vignettes")
```

Arguments

file	zip file
subdir	sub directory that contains R package files, default is NULL
args	argument to build package

Value

No return value, called for install R package from zip file of source codes

Author(s)

Guangchuang Yu

`install_zip_gh` *install_zip_gh*

Description

install github package

Usage

```
install_zip_gh(
  repo,
  ref = "HEAD",
  subdir = NULL,
  args = "--no-build-vignettes"
)
```

Arguments

`repo`	github repo
`ref`	github branch, default is HEAD, which means the default branch of the GitHub repo
`subdir`	sub directory that contains R package files, default is NULL
`args`	argument to build package

Details

it download the zip file first and use `install_zip` to install it

Value

No return value, called for installing github package

Author(s)

Guangchuang Yu

is.installed *is.installed*

Description

Check whether packages are installed

Usage

```
is.installed(packages)
```

Arguments

packages	package names

Value

logical vector

Author(s)

Guangchuang Yu

See Also

Other pkg-utils: CRANpkg(), Githubpkg(), get_dependencies(), get_fun_from_pkg(), mypkg(), packageTitle(), pload()

Examples

```
is.installed(c("dplyr", "ggplot2"))
```

load_OrgDb *load_OrgDb*

Description

load OrgDb

Usage

```
load_OrgDb(OrgDb)
```

Arguments

OrgDb	OrgDb object or OrgDb name

Value

OrgDb object

Author(s)

Guangchuang Yu https://yulab-smu.top

---

ls2df *Convert a list of vectors (e.g., gene IDs) to* data.frame

---

Description

Convert a list of vectors to a data.frame

Usage

```
ls2df(inputList)
```

Arguments

inputList	List of vectors

Value

data.frame

---

mat2df *mat2df*

Description

convert a matrix to a tidy data frame (from wide to long format as described in the tidyverse concept)

Usage

```
mat2df(x)
```

Arguments

x	the input matrix

Value

a data.frame in long format with the 'value' column stores the original values and 'row' and 'col' columns stored in row and column index as in x

Author(s)

Guangchuang Yu

Examples

```
x <- matrix(1:15, nrow = 3)
mat2df(x)
```

---

mat2list *mat2list*

Description

convert a matrix to a list

Usage

```
mat2list(x)
```

Arguments

x	the input matrix

Value

a list that contains matrix columns as its elements

Examples

```
x <- matrix(1:15, nrow = 3)
mat2list(x)
```

---

mypkg *mypkg*

Description

Markdown link to a package

Usage

```
mypkg(pkg, url)
```

Arguments

pkg	package name
url	package url

Value

md text string

Author(s)

Guangchuang Yu

See Also

Other pkg-utils: CRANpkg(), Githubpkg(), get_dependencies(), get_fun_from_pkg(), is.installed(), packageTitle(), pload()

---

o *o*

Description

open selected directory or file

Usage

```
o(file = ".")
```

Arguments

file	to be open; open working directory by default

Value

No return value, called for opening specific directory or file

Author(s)

Guangchuang Yu

See Also

Other io-utils: read.cb(), show_in_excel(), yread_tsv()

Examples

```
## Not run:
## to open current working directory
o()

## End(Not run)
```

packageTitle *packageTitle*

Description

Extract package title

Usage

```
packageTitle(pkg, repo = "CRAN")
```

Arguments

pkg	package name
repo	'CRAN' and/or 'BioC'

Value

reverse dependencies

Author(s)

Guangchuang Yu

See Also

Other pkg-utils: CRANpkg(), Githubpkg(), get_dependencies(), get_fun_from_pkg(), is.installed(), mypkg(), pload()

parse_ratio *parse_ratio*

Description

Parse character ratio to double, e.g., `1/5` → `0.2`

Usage

```
parse_ratio(ratio)
```

Arguments

ratio	Character vector of ratios

Value

Numeric vector

Author(s)

Guangchuang Yu

`pload` *pload*

Description

Load a package

Usage

```
pload(package, action = "auto")
```

Arguments

`package`	package name
`action`	Installation function; `"auto"` tries `BiocManager::install()` if available

Details

Uses `library()` to load package. If not installed, attempts installation via `rlang::check_installed()` (optionally using `BiocManager::install()`).

Value

the selected package loaded to the R session

Author(s)

Guangchuang Yu

See Also

Other pkg-utils: `CRANpkg()`, `Githubpkg()`, `get_dependencies()`, `get_fun_from_pkg()`, `is.installed()`, `mypkg()`, `packageTitle()`

quiet *quiet*

Description

Suppress messages and output from x

Usage

```
quiet(x)
```

Arguments

x	some code

Value

the result of x

rbindlist *rbindlist*

Description

Row-bind a list

Usage

```
rbindlist(x)
```

Arguments

x	List with similar elements that can be row-bound

Value

data.frame

Author(s)

Guangchuang Yu

read.cb *read.cb*

Description

read clipboard

Usage

```
read.cb(reader = read.table, ...)
```

Arguments

reader	function to read the clipboard
...	parameters for the reader

Value

clipboard content, output type depends on the output of the reader

Author(s)

Guangchuang Yu

See Also

Other io-utils: o(), show_in_excel(), yread_tsv()

scale_range *scale-range*

Description

normalized data by range

Usage

```
scale_range(data)
```

Arguments

data	the input data.

Value

normalized data

Author(s)

Guangchuang Yu

`scihub_dl` *download publication via scihub*

Description

using scihub to download publication using doi

Usage

```
scihub_dl(doi, scihub = "sci-hub.tw", download = TRUE)
```

Arguments

`doi`	doi
`scihub`	scihub website
`download`	whether download the pdf file

Value

pdf url

Author(s)

Guangchuang Yu

`set_PCRE` *Switch regular expression style (PCRE vs TRE)*

Description

- `set_regexpr_style()` selects the style explicitly.
- `auto_set_regexpr_style()` chooses based on OS (TRE on Windows; PCRE elsewhere).
- `set_PCRE()` and `set_TRE()` force the style.

Usage

```
set_PCRE()

set_TRE()

use_perl()

set_regexpr_style(style)

auto_set_regexpr_style()
```

Arguments

`style`	one of 'PCRE' or 'TRE'

Details

These functions do not change the behavior of `gsub()`/`regexpr()` directly. They set a global option that you can read via `use_perl()` and pass to `gsub()`/`regexpr()`.

Value

Logical indicating whether to use `perl`

Author(s)

Guangchuang Yu

References

https://stackoverflow.com/questions/47240375/regular-expressions-in-base-r-perl-true-vs-the-default-pcre-vs-tre

---

`show_in_excel` *show_in_excel*

Description

Open data frame in Excel. It can be used in pipe.

Usage

```
show_in_excel(.data)
```

Arguments

`.data`	a data frame to be open

Value

original .data

Author(s)

Guangchuang Yu

See Also

Other io-utils: `o()`, `read.cb()`, `yread_tsv()`

`str_detect` *str_detect*

Description

Detect presence/absence of a match

Usage

```
str_detect(string, pattern, negate = FALSE)
```

Arguments

`string`	Input string
`pattern`	Pattern to look for
`negate`	If `TRUE`, invert the result

Value

Logical vector

Author(s)

Guangchuang Yu

See Also

Other str-utils: `str_extract()`, `str_starts()`, `str_wrap()`

`str_extract` *str_extract*

Description

Extract a substring using a pattern

Usage

```
str_extract(string, pattern)
```

Arguments

`string`	Input string
`pattern`	Regular expression to extract

Value

Substring

Author(s)

Guangchuang Yu

See Also

Other str-utils: `str_detect()`, `str_starts()`, `str_wrap()`

`str_starts` *str_starts*

Description

Detect patterns at the beginning or end of strings

Usage

```
str_starts(string, pattern, negate = FALSE)

str_ends(string, pattern, negate = FALSE)
```

Arguments

`string`	Input string
`pattern`	Pattern to match
`negate`	If `TRUE`, return non-matching elements

Value

a logical vector

Author(s)

Guangchuang Yu

See Also

Other str-utils: `str_detect()`, `str_extract()`, `str_wrap()`

str_wrap *str_wrap*

Description

Wrap long strings to multiple lines

Usage

```
str_wrap(string, width = getOption("width"))
```

Arguments

string	Input string
width	Maximum characters before wrapping

Value

Updated strings with "\n" inserted

Author(s)

Guangchuang Yu

See Also

Other str-utils: str_detect(), str_extract(), str_starts()

user_dir *user_dir*

Description

get the user dir to save app caches, logs and data (a wrapper function of rappdirs::user_cache_dir())

Usage

```
user_dir(appname = NULL, appauthor = NULL, ...)
```

Arguments

appname	App name
appauthor	App author
...	additional parameters

Value

a directory (created if not exists)

Author(s)

Guangchuang Yu

See Also

Other os-utils: `exec()`, `has_internet()`

`yread_tsv` *yread*

Description

read file with caching

Usage

```
yread_tsv(
  file,
  reader = utils::read.delim,
  params = list(),
  cache_dir = tempdir()
)

yread(file, reader = readLines, params = list(), cache_dir = NULL)
```

Arguments

`file`	a file or url
`reader`	a function to read the 'file_url'
`params`	a list of parameters that passed to the 'reader'
`cache_dir`	a folder to store cache files. If set to NULL will disable cache.

Details

This function read a file (local or url) and cache the content.

Value

the output of using the 'reader' to read the 'file_url' with parameters specified by the 'params'

Author(s)

Yonghe Xia and Guangchuang Yu

See Also

Other io-utils: `o()`, `read.cb()`, `show_in_excel()`

`yulab_abort` *Standardized error handling*

Description

Provides `rlang`-based wrappers for messaging: `yulab_abort()`, `yulab_warn()`, and `yulab_inform()`.

Usage

```
yulab_abort(message, class = "yulab_error", ...)

yulab_warn(message, class = "yulab_warning", ...)

yulab_inform(message, class = "yulab_info", ...)
```

Arguments

`message`	Message string
`class`	Custom class for categorization
`...`	Additional context

Value

No return value

`yulab_msg` *yulab_msg*

Description

Messages for YuLab packages

Usage

```
yulab_msg(pkgname = NULL, n = 1)
```

Arguments

`pkgname`	Package name
`n`	Number of citation messages

Value

Package message

Author(s)

Guangchuang Yu

Index

∗ **cache-utils**
  initial_cache, 12
∗ **io-utils**
  o, 18
  read.cb, 22
  show_in_excel, 24
  yread_tsv, 28
∗ **messages**
  yulab_abort, 29
∗ **os-utils**
  exec, 8
  has_internet, 11
  user_dir, 27
∗ **pkg-utils**
  CRANpkg, 7
  get_dependencies, 9
  get_fun_from_pkg, 10
  Githubpkg, 10
  is.installed, 15
  mypkg, 17
  packageTitle, 19
  pload, 20
∗ **regex-utils**
  set_PCRE, 23
∗ **str-utils**
  str_detect, 25
  str_extract, 25
  str_starts, 26
  str_wrap, 27
∗ **validate-utils**
  check_directory, 4
  check_file, 4
  check_input, 5
  check_packages, 6
  check_range, 6

as_chunked_array, 2
auto_set_regexpr_style *(*set_PCRE*)*, 23

Biocpkg *(*CRANpkg*)*, 7

c2, 3
cache_list_items *(*initial_cache*)*, 12
cache_load *(*initial_cache*)*, 12
cache_save *(*initial_cache*)*, 12
cache_size *(*initial_cache*)*, 12
check_directory, 4
check_directory(), *5–7*
check_file, 4
check_file(), *4*, *6, 7*
check_input, 5
check_input(), *4–7*
check_packages, 6
check_packages(), *4–7*
check_pkg *(*check_packages*)*, 6
check_range, 6
check_range(), *4–6*
combinations, 7
CRANpkg, 7
CRANpkg(), *9–11*, *15*, *18–20*

download_yulab_file, 8

exec, 8
exec(), *11*, *28*

get_cache *(*initial_cache*)*, 12
get_cache_element *(*initial_cache*)*, 12
get_cache_item *(*initial_cache*)*, 12
get_dependencies, 9
get_dependencies(), *8*, *10, 11*, *15*, *18–20*
get_fun_from_pkg, 10
get_fun_from_pkg(), *8, 9*, *11*, *15*, *18–20*
Githubpkg, 10
Githubpkg(), *8–10*, *15*, *18–20*

has_internet, 11
has_internet(), *9*, *28*

initial_cache, 12
initial_cache_item *(*initial_cache*)*, 12
install_zip, 13

install_zip_gh, 14
is.installed, 15
is.installed(), *8–11*, *18–20*

load_OrgDb, 15
ls2df, 16

mat2df, 16
mat2list, 17
mypkg, 17
mypkg(), *8–11*, *15*, *19, 20*

o, 18
o(), *22*, *24*, *29*

packageTitle, 19
packageTitle(), *8–11*, *15*, *18*, *20*
parse_ratio, 19
pload, 20
pload(), *8–11*, *15*, *18, 19*
prune_cache_item *(*initial_cache*)*, 12

quiet, 21

rbindlist, 21
read.cb, 22
read.cb(), *18*, *24*, *29*
rm_cache *(*initial_cache*)*, 12
rm_cache_item *(*initial_cache*)*, 12

scale_range, 22
scihub_dl, 23
set_PCRE, 23
set_regexpr_style *(*set_PCRE*)*, 23
set_TRE *(*set_PCRE*)*, 23
show_in_excel, 24
show_in_excel(), *18*, *22*, *29*
str_detect, 25
str_detect(), *26, 27*
str_ends *(*str_starts*)*, 26
str_extract, 25
str_extract(), *25–27*
str_starts, 26
str_starts(), *25–27*
str_wrap, 27
str_wrap(), *25, 26*

update_cache_item *(*initial_cache*)*, 12
use_perl *(*set_PCRE*)*, 23
user_dir, 27
user_dir(), *9*, *11*

with_cache *(*initial_cache*)*, 12

yread *(*yread_tsv*)*, 28
yread_tsv, 28
yread_tsv(), *18*, *22*, *24*
yulab_abort, 29
yulab_inform *(*yulab_abort*)*, 29
yulab_msg, 29
yulab_warn *(*yulab_abort*)*, 29