Package ‘deepdep’

March 6, 2020

Title  Visualise and Explore the Deep Dependencies of R Packages

Version  0.2.0

Description  Provides the tools for exploration of package dependencies.
    Main deepdep() function allows to acquire deep dependencies of any package
    and plot them in an elegant way.
    It also adds some popularity measures for the packages e.g. in the form of down-
    load count through the 'cranlogs' package.
    Other data acquire functions are: get_dependencies(), get_downloads() and get_description().
    The deepdep_shiny() function runs shiny application that helps to produce nice 'deepdep' plot.

License  GPL-3

Encoding  UTF-8

LazyData  true

RoxygenNote  7.0.2

Depends  R (>= 3.2.0)

Imports  cranlogs, ggforce, ggplot2, ggraph, graphlayouts, httr,
        igraph, jsonlite, scales

Suggests  BiocManager, shiny, shinyCSSLoaders, covr, devtools, knitr,
         miniCRAN, plyr, rmarkdown, spelling, stringi, testthat (>=
         2.1.0)

VignetteBuilder  knitr

URL  https://github.com/DominikRafacz/deepdep

BugReports  https://github.com/DominikRafacz/deepdep/issues

Language  en-GB

NeedsCompilation  no

Author  Dominik Rafacz [aut, cre],
        Hubert Baniecki [aut],
        Szymon Maksymiuk [aut],
        Mateusz Bakala [aut]
Maintainer Dominik Rafacz <dominikrafacz@gmail.com>

Repository CRAN

Date/Publication 2020-03-06 13:20:05 UTC

R topics documented:

deepdep ................................................................. 2
deepdep_shiny .......................................................... 3
get_available_packages .............................................. 4
get_dependencies ...................................................... 5
get_description ........................................................ 6
get_downloads .......................................................... 7
plot_dependencies ..................................................... 7
plot_downloads .......................................................... 9
print.available_packages ........................................... 10
print.deepdep ............................................................ 11
print.package_dependencies ......................................... 12
print.package_description ........................................... 12
print.package_downloads ............................................ 13

Index 14

____________________

deepdep  Acquire the dependencies of the package on any depth level

Description

This function is an ultimate wrapper for get_dependencies. It inherits all of the arguments and allows to recursively search for the dependencies at the higher level of depth.

Usage

deepdep(
  package,
  depth = 1,
  downloads = FALSE,
  bioc = FALSE,
  local = FALSE,
  dependency_type = c("Depends", "Imports")
)


Arguments

package A character. Name of the package that is on CRAN, Bioconductor repository or locally installed. See bioc and local arguments.

depth An integer. Maximum depth level of the dependency. By default it's 1.

downloads A logical. If TRUE add dependency downloads data. By default it's FALSE.

bioc A logical value. If TRUE the Bioconductor dependencies data will be taken from the Bioconductor repository. For this option to work properly, BiocManager package needs to be installed.

local A logical value. If TRUE only data of locally installed packages will be used (without API usage).

dependency_type A character vector. Types of the dependencies that should be sought. Possibilities are: "Imports", "Depends", "Suggests", "Enhances", "LinkingTo". By default it's "Depends", "Imports".

Value

An object of deepdep class.

See Also

get_dependencies

Examples

library(deepdep)

dd_downloads <- deepdep("ggplot2")
head(dd_downloads)

dd_2 <- deepdep("ggplot2", depth = 2, downloads = TRUE)
plot_dependencies(dd_2, "circular")

dd_local <- deepdep("deepdep", local = TRUE)
plot_dependencies(dd_local)

---

deepdep_shiny Run Shiny app

Description

This function runs shiny app that helps to produce nice deepdep plot.
get_available_packages

Usage

depdep_shiny()

Description

Get the list of available packages

Usage

get_available_packages(bioc = FALSE, local = FALSE, reset_cache = FALSE)

Arguments

bioc  A logical value. If TRUE the Bioconductor dependencies data will be taken from the Bioconductor repository. For this option to work properly, BiocManager package needs to be installed.

local  A logical value. If TRUE only data of locally installed packages will be used (without API usage).

reset_cache  A logical value. If TRUE the cache will be cleared before obtaining the list of packages.

Details

Function uses caching - only the first usage scraps information from servers. Those objects are then saved locally in temporary file and further usages loads needed data from the file.

Arguments bioc and local cannot be TRUE simultaneously. If neither local nor bioc are TRUE, vector contains all packages available currently on CRAN. If bioc is TRUE, vector contains all packages available currently on CRAN and via Bioconductor. If local is TRUE, vector contains all of the packages that are currently installed.

Value

A character vector.

Examples

library(deepdep)

av <- get_available_packages()
head(av)
get_dependencies

Description

This function uses get_description and get_downloads to acquire the dependencies of the package (with their downloads).

Usage

get_dependencies(
  package,
  downloads = TRUE,
  bioc = FALSE,
  local = FALSE,
  dependency_type = c("Depends", "Imports")
)

Arguments

package A character. Name of the package that is on CRAN, Bioconductor repository or locally installed. See bioc and local arguments.
downloads A logical. If TRUE add package downloads data. By default it's TRUE.
bioc A logical value. If TRUE the Bioconductor dependencies data will be taken from the Bioconductor repository. For this option to work properly, BiocManager package needs to be installed.
local A logical value. If TRUE only data of locally installed packages will be used (without API usage).
dependency_type A character vector. Types of the dependencies that should be sought. Possibilities are: "Imports", "Depends", "Suggests", "Enhances", "LinkingTo". By default it's "Depends", "Imports".

Value

An object of package_dependencies class.

See Also

get_description get_downloads

Examples

library(deepdep)

dependencies <- get_dependencies("htmltools", downloads = FALSE)
dependencies
dependencies_local <- get_dependencies("deepdep", downloads = FALSE, local = TRUE)
dependencies_local

get_description	Scrap the DESCRIPTION file and CRAN metadata of the package

Description
This function uses api of CRAN Data Base to scrap the DESCRIPTION file and CRAN metadata of the package. It caches the results to speed the computation process.

Usage
get_description(package, bioc = FALSE, local = FALSE, reset_cache = FALSE)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>package</td>
<td>A character. Name of the package that is on CRAN, Bioconductor repository or locally installed. See bioc and local arguments.</td>
</tr>
<tr>
<td>bioc</td>
<td>A logical value. If TRUE the Bioconductor dependencies data will be taken from the Bioconductor repository. For this option to work properly, BiocManager package needs to be installed.</td>
</tr>
<tr>
<td>local</td>
<td>A logical value. If TRUE only data of locally installed packages will be used (without API usage).</td>
</tr>
<tr>
<td>reset_cache</td>
<td>A logical value. If TRUE the cache will be cleared before obtaining the list of packages.</td>
</tr>
</tbody>
</table>

Value
An object of package_description class.

Examples

library(deepdep)

description <- get_description("ggplot2")
description

description_local <- get_description("deepdep", local = TRUE)
description_local
**get_downloads**

*Scrap the download data of the package*

**Description**

This function uses **API** of **CRAN Logs** to scrap the download logs of the package.

**Usage**

```r
get_downloads(package)
```

**Arguments**

- `package` A character. Name of the package that is on CRAN.

**Value**

An object of `package_downloads` class.

**Examples**

```r
library(deepdep)

downloads <- get_downloads("ggplot2")
downloads
```

**plot_dependencies**

*Main plot function for a deepdep object*

**Description**

Visualize dependency data from a `deepdep` object using `ggplot2` and `ggraph` packages. Several tree-like layouts are available.

**Usage**

```r
plot_dependencies(
  x,
  type = "circular",
  same_level = FALSE,
  reverse = FALSE,
  label_percentage = 1,
  show_version = FALSE,
  show_downloads = FALSE,
```
plot_dependencies

## Default S3 method:
plot_dependencies(
  x,
  type = "circular",
  same_level = FALSE,
  reverse = FALSE,
  label_percentage = 1,
  show_version = FALSE,
  show_downloads = FALSE,
  ...
)

## S3 method for class 'character'
plot_dependencies(
  x,
  type = "circular",
  same_level = FALSE,
  reverse = FALSE,
  label_percentage = 1,
  show_version = FALSE,
  show_downloads = FALSE,
  ...
)

## S3 method for class 'deepdep'
plot_dependencies(
  x,
  type = "circular",
  same_level = FALSE,
  reverse = FALSE,
  label_percentage = 1,
  show_version = FALSE,
  show_downloads = FALSE,
  ...
)

Arguments

x A `deepdep` object or a character package name.
type A character. Possible values are circular and tree.
same_level A logical. If TRUE links between dependencies on the same level will be added. By default it's FALSE.
reverse A logical. If TRUE links between dependencies pointing from deeper level to more shallow level will be added. By default it's FALSE.
plot_downloads

label_percentage

A numeric value between 0 and 1. A fraction of labels to be displayed. By default it’s 1 (all labels displayed).

show_version

A logical. If TRUE required version of package will be displayed below package name. Defaults to FALSE.

show_downloads

A logical. If TRUE total number of downloads of packages will be displayed below package names. Defaults to FALSE.

... Other arguments passed to the deepdep function.

Value

A ggplot2, gg, ggraph, deepdep_plot class object.

Examples

library(deepdep)

dd <- deepdep("ggplot2")
plot_dependencies(dd, "tree")

dd2 <- deepdep("ggplot2", depth = 2)
plot_dependencies(dd2, "circular")

plot_dependencies("deepdep", label_percentage = 0.5, depth = 2, local = TRUE)

plot_downloads x, ...

Plot download count of CRAN packages.

Description

This function uses API of CRAN Logs to scrap the download logs of the packages and then plots the data. It works on objects of class character (vector), deepdep, package_dependencies and package_downloads.

Usage

plot_downloads(x, ...)

## Default S3 method:
plot_downloads(x, ...)

## S3 method for class 'deepdep'
plot_downloads(x, from = Sys.Date() - 365, to = Sys.Date(), ...)

## S3 method for class 'package_dependencies'
plot_downloads(x, from = Sys.Date() - 365, to = Sys.Date(), ...)  
## S3 method for class 'package_downloads'
plot_downloads(x, from = Sys.Date() - 365, to = Sys.Date(), ...)  
## S3 method for class 'character'
plot_downloads(x, from = Sys.Date() - 365, to = Sys.Date(), ...)  

Arguments

x  A character vector. Names of the packages that are on CRAN.
...
from  A Date class object. From which date plot the data. By default it’s one year back.
to  A Date class object. To which date plot the data. By default it’s now.

Value

A ggplot2 class object.

Examples

library(deepdep)

plot_downloads("htmltools")

dd <- deepdep("ggplot2")
plot_downloads(dd)

print.available_packages

Print function for an object of available_packages class

Description

Print function for an object of available_packages class

Usage

## S3 method for class 'available_packages'
print(x, ...)
print.deepdep

Arguments

x  An object of available_packages class.

...  other

Examples

library(deepdep)

av <- get_available_packages()
head(av)

print.deepdep  Print function for an object of deepdep class

Description

Print function for an object of deepdep class

Usage

## S3 method for class 'deepdep'
print(x, ...)

Arguments

x  An object of deepdep class.

...  other

Examples

library(deepdep)

dd <- deepdep("stringr")

dd
print.package_dependencies

Print function for an object of package_dependencies class

Description

Print function for an object of package_dependencies class

Usage

## S3 method for class 'package_dependencies'
print(x, ...)

Arguments

x An object of package_dependencies class.
...
other

Examples

library(deepdep)

get_dependencies("htmltools", downloads = TRUE)

print.package_description

Print function for an object of package_description class

Description

Print function for an object of package_description class

Usage

## S3 method for class 'package_description'
print(x, ...)

Arguments

x An object of package_description class.
...
other
Examples

library(deepdep)

description <- get_description("ggplot2")
description

print.package_downloads

Print function for an object of package_downloads class

Description

Print function for an object of package_downloads class

Usage

## S3 method for class 'package_downloads'
print(x, ...)

Arguments

x An object of package_downloads class.
... other

Examples

library(deepdep)

desc <- get_downloads("stringr")
desc
Index

depdep, 2, 8
deepdep_shiny, 3

get_available_packages, 4
get_dependencies, 2, 3, 5
get_description, 5, 6
get_downloads, 5, 7

plot_dependencies, 7
plot_downloads, 9
print.available_packages, 10
print.deepdep, 11
print.package_dependencies, 12
print.package_description, 12
print.package_downloads, 13