

Package ‘packageRank’

October 21, 2019

Type Package

Title Computation and Visualization of Package Download Counts and Percentiles

Version 0.3.0

Date 2019-10-21

Maintainer Peter Li <lindbrook@gmail.com>

Description Compute and visualize the cross-sectional and longitudinal number and rank percentile of package downloads from RStudio's CRAN mirror.

URL <https://github.com/lindbrook/packageRank>

BugReports <https://github.com/lindbrook/packageRank/issues>

Depends R (>= 3.4)

License GPL (>= 2)

Encoding UTF-8

Language en-US

LazyData true

RoxygenNote 6.1.1

Imports cranlogs, data.table (>= 1.12.2), ggplot2, grDevices, memoise, RCurl, rversions, stats

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Peter Li [aut, cre]

Repository CRAN

Date/Publication 2019-10-21 19:50:02 UTC

R topics documented:

bioconductorDownloads	2
bioconductorRank	3
cranDownloads	4
fetchLog	5
packageRank	5
packageRankTime	6
plot.bioconductor	6
plot.bioconductor_rank	7
plot.cran_downloads	8
plot.package_rank	9
plot.package_rank_time	9
print.bioconductor	10
print.bioconductor_rank	11
print.cran_downloads	11
print.package_rank	12
print.package_rank_time	12
summary.bioconductor	13
summary.bioconductor_rank	13
summary.cran_downloads	14
summary.package_rank	14
summary.package_rank_time	15

Index	16
--------------	-----------

bioconductorDownloads *Annual/monthly package downloads from Bioconductor (beta).*

Description

Annual/monthly package downloads from Bioconductor (beta).

Usage

```
bioconductorDownloads(packages = NULL, from = NULL, to = NULL,
  when = NULL, observation = "month")
```

Arguments

packages	Character. Vector of package names.
from	Start date as yyyy-mm or yyyy.
to	End date as yyyy-mm or yyyy.
when	"last-year", or "year-to-date" or "ytd".
observation	"year" or "month".

Examples

```

# all packages
bioconductorDownloads()

# entire history
bioconductorDownloads(packages = "clusterProfiler")

# year-to-date
bioconductorDownloads(packages = "clusterProfiler", when = "ytd")
bioconductorDownloads(packages = "clusterProfiler", when = "year-to-date")

# last 12 months
bioconductorDownloads(packages = "clusterProfiler", when = "last-year")

# from 2015 to current year
bioconductorDownloads(packages = "clusterProfiler", from = 2015)

# 2015 through 2018 (yearly)
bioconductorDownloads(packages = "clusterProfiler", from = 2010, to = 2015, observation = "year")

# selected year (yearly)
bioconductorDownloads(packages = "clusterProfiler", from = 2015, to = 2015)

# selected year (monthly)
bioconductorDownloads(packages = "clusterProfiler", from = "2015-01", to = "2015-12")

# June 2014 through March 2018
bioconductorDownloads(packages = "clusterProfiler", from = "2014-06", to = "2015-03")

```

bioconductorRank	<i>Package download counts and rank percentiles (cross-sectional) (prototype).</i>
------------------	--

Description

From bioconductor

Usage

```

bioconductorRank(packages = "monocle", date = "2019-01",
  count = "download")

```

Arguments

packages	Character. Vector of package name(s).
date	Character. Date. yyyy-mm
count	Character. "ip" or "download".

Value

An R data frame.

Examples

```
bioconductorRank(packages = "cicero", date = "2019-09")
```

cranDownloads

Daily package downloads from the RStudio CRAN mirror.

Description

S3 implementation of `cranlogs::cran_downloads()`.

Usage

```
cranDownloads(packages = NULL, when = NULL, from = NULL, to = NULL)
```

Arguments

packages	A character vector, the packages to query, or NULL for a sum of downloads for all packages. Alternatively, it can also be "R", to query downloads of R itself. "R" cannot be mixed with packages.
when	last-day, last-week or last-month. If this is given, then from and to are ignored.
from	Start date as yyyy-mm-dd, yyyy-mm or yyyy.
to	End date as yyyy-mm-dd, yyyy-mm or yyyy.

Examples

```
cranDownloads(packages = "HistData")
cranDownloads(packages = "HistData", when = "last-week")
cranDownloads(packages = "HistData", when = "last-month")

# January 7 - 31, 2019
cranDownloads(packages = "HistData", from = "2019-01-07", to = "2019-01-31")

# February to March 2019
cranDownloads(packages = "HistData", from = "2019-02", to = "2019-03")

# 2019 year-to-date
cranDownloads(packages = "HistData", from = 2019)
```

fetchLog	<i>Fetch Package Logs.</i>
----------	----------------------------

Description

Fetch Package Logs.

Usage

```
fetchLog(x)
```

Arguments

x Character. URL

Note

mFetchLog() is memoized version.

packageRank	<i>Package download counts and rank percentiles (cross-sectional).</i>
-------------	--

Description

From RStudio's CRAN Mirror <http://cran-logs.rstudio.com/>

Usage

```
packageRank(packages = "HistData", date = Sys.Date() - 1,  
            memoization = TRUE)
```

Arguments

packages Character. Vector of package name(s).
date Character. Date. yyyy-mm-dd
memoization Logical. Use memoization when downloading logs.

Value

An R data frame.

Examples

```
packageRank(packages = "HistData", date = "2019-01-01")  
packageRank(packages = c("h2o", "Rcpp", "rstan"), date = "2019-01-01")
```

packageRankTime	<i>Package download counts and rank percentiles (longitudinal).</i>
-----------------	---

Description

Temporal pattern over last week or month.

Usage

```
packageRankTime(packages = "HistData", when = "last-month",
  sample.pct = 5, multi.core = TRUE)
```

Arguments

packages	Character. Vector of package name(s).
when	Character. "last-month" or "last-week".
sample.pct	Numeric. Percent of packages to sample.
multi.core	Logical or Numeric. TRUE uses <code>parallel::detectCores()</code> . FALSE uses one, single core. You can also specify the number logical cores to use. Note that due to performance considerations, the number of cores defaults to one on Windows.

Note

Most useful with `plot()` method.

Examples

```
plot(packageRankTime(packages = "HistData", when = "last-week"))
plot(packageRankTime(packages = c("Rcpp", "rlang", "data.table"), when = "last-month"))
```

plot.bioconductor	<i>Plot method for bioconductor_downloads().</i>
-------------------	--

Description

Plot method for `bioconductor_downloads()`.

Usage

```
## S3 method for class 'bioconductor'
plot(x, graphics = NULL, count = "download",
  points = "auto", smooth = FALSE, smooth.f = 2/3, se = FALSE,
  log_count = FALSE, ...)
```

Arguments

x	object.
graphics	Character. NULL, "base" or "ggplot2".
count	Character. "download" or "ip".
points	Character or Logical. Plot points. "auto", TRUE, FALSE. "auto" for bioconductorDownloads(observation = "month") with 24 or fewer months, points are plotted.
smooth	Logical. Add stats::lowess smoother.
smooth.f	Numeric. smoother span.
se	Logical. Works only with graphics = "ggplot2".
log_count	Logical. Logarithm of package downloads.
...	Additional plotting parameters.

Examples

```
plot(bioconductorDownloads())
plot(bioconductorDownloads/packages = "graph")
plot(bioconductorDownloads/packages = "graph", from = 2010, to = 2015)
plot(bioconductorDownloads/packages = "graph", from = "2014-06", to = "2015-03")
plot(bioconductorDownloads/packages = c("graph", "IRanges", "S4Vectors"), from = 2018)
```

plot.bioconductor_rank

Plot method for bioconductorRank().

Description

Plot method for bioconductorRank().

Usage

```
## S3 method for class 'bioconductor_rank'
plot(x, graphics = "base",
     log_count = TRUE, ...)
```

Arguments

x	An object of class "bioconductor_rank" created by bioconductorRank().
graphics	Character. "base" or "ggplot2".
log_count	Logical. Logarithm of package downloads.
...	Additional plotting parameters.

Value

A base R or ggplot2 plot.

plot.cran_downloads *Plot method for cranDownloads().*

Description

Plot method for cranDownloads().

Usage

```
## S3 method for class 'cran_downloads'  
plot(x, graphics = NULL, points = "auto",  
     log_count = FALSE, smooth = FALSE, se = FALSE, f = 1/3,  
     r.version = FALSE, ...)
```

Arguments

x	object.
graphics	Character. NULL, "base" or "ggplot2".
points	Character or Logical. Plot points. "auto", TRUE, FALSE.
log_count	Logical. Logarithm of package downloads.
smooth	Logical. Add smoother.
se	Logical. Works only with graphics = "ggplot2".
f	Numeric. stats::lowess() smoother window. For use with graphics = "base" only.
r.version	Logical. Add R release dates.
...	Additional plotting parameters.

Value

A base R or ggplot2 plot.

Examples

```
plot(cranDownloads/packages = c("Rcpp", "rlang", "data.table"))  
plot(cranDownloads/packages = c("Rcpp", "rlang", "data.table"), when = "last-month")  
plot(cranDownloads/packages = "R", from = "2019-05-01", to = "2019-05-01")  
plot(cranDownloads/packages = "R", from = 2019)
```

plot.package_rank *Plot method for packageRank().*

Description

Plot method for packageRank().

Usage

```
## S3 method for class 'package_rank'  
plot(x, graphics = NULL, log_count = TRUE, ...)
```

Arguments

x	An object of class "package_rank" created by packageRank().
graphics	Character. "base" or "ggplot2".
log_count	Logical. Logarithm of package downloads.
...	Additional plotting parameters.

Value

A base R or ggplot2 plot.

Examples

```
plot(packageRank(packages = "HistData", date = "2019-01-01"))  
plot(packageRank(packages = c("h2o", "Rcpp", "rstan"), date = "2019-01-01"))
```

plot.package_rank_time *Plot method for timeSeriesRank().*

Description

Plot method for timeSeriesRank().

Usage

```
## S3 method for class 'package_rank_time'  
plot(x, graphics = NULL, log_count = TRUE,  
      smooth = TRUE, sample_smooth = TRUE, f = 1/3, ...)
```

Arguments

<code>x</code>	Object. An object of class "time_series" created by <code>packageRankTime()</code> .
<code>graphics</code>	Character. "base" or "ggplot2".
<code>log_count</code>	Logical. Logarithm of package downloads.
<code>smooth</code>	Logical. Add smoother for selected package.
<code>sample_smooth</code>	Logical. lowess background.
<code>f</code>	Numeric. <code>stats::lowess()</code> smoother window. For use with <code>graphics = "base"</code> only.
<code>...</code>	Additional plotting parameters.

Value

A base R or ggplot2 plot.

Examples

```
plot(packageRankTime(packages = "HistData", when = "last-week"))
plot(packageRankTime(packages = c("Rcpp", "rlang", "data.table"), when = "last-month"))
```

`print.bioconductor` *Print method for bioconductorDownloads().*

Description

Print method for `bioconductorDownloads()`.

Usage

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

Arguments

<code>x</code>	object.
<code>...</code>	Additional parameters.

```
print.bioconductor_rank
```

Print method for bioconductorRank().

Description

Print method for bioconductorRank().

Usage

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

Arguments

x	An object of class "bioconductor_rank" created by bioconductorRank()
...	Additional parameters.

```
print.cran_downloads
```

Print method for packageRank().

Description

Print method for packageRank().

Usage

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

Arguments

x	object.
...	Additional parameters.

print.package_rank *Print method for packageRank().*

Description

Print method for packageRank().

Usage

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

Arguments

x An object of class "package_rank" created by packageRank()
... Additional parameters.

print.package_rank_time
 Print method for timeSeriesRank().

Description

Print method for timeSeriesRank().

Usage

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

Arguments

x An object of class "time_series" created by packageRankTime().
... Additional parameters.

summary.bioconductor *Summary method for bioconductorDownloads().*

Description

Summary method for bioconductorDownloads().

Usage

```
## S3 method for class 'bioconductor'  
summary(object, ...)
```

Arguments

object	Object.
...	Additional parameters.

summary.bioconductor_rank
Summary method for bioconductorRank().

Description

Summary method for bioconductorRank().

Usage

```
## S3 method for class 'bioconductor_rank'  
summary(object, ...)
```

Arguments

object	Object. An object of class "bioconductor_rank" created by bioconductorRank()
...	Additional parameters.

Note

This is useful for directly accessing the data frame.

summary.cran_downloads

Summary method for packageRank().

Description

Summary method for packageRank().

Usage

```
## S3 method for class 'cran_downloads'  
summary(object, ...)
```

Arguments

object Object.
... Additional parameters.

Note

This is useful for directly accessing the data frame.

summary.package_rank *Summary method for packageRank().*

Description

Summary method for packageRank().

Usage

```
## S3 method for class 'package_rank'  
summary(object, ...)
```

Arguments

object Object. An object of class "package_rank" created by packageRank()
... Additional parameters.

Note

This is useful for directly accessing the data frame.

summary.package_rank_time

Summary method for timeSeriesRank().

Description

Summary method for timeSeriesRank().

Usage

```
## S3 method for class 'package_rank_time'  
summary(object, ...)
```

Arguments

object Object. An object of class "time_series" created by packageRankTime().
... Additional parameters.

Note

This is useful for directly accessing the data frame.

Index

bioconductorDownloads, [2](#)
bioconductorRank, [3](#)

cranDownloads, [4](#)

fetchLog, [5](#)

packageRank, [5](#)
packageRankTime, [6](#)
plot.bioconductor, [6](#)
plot.bioconductor_rank, [7](#)
plot.cran_downloads, [8](#)
plot.package_rank, [9](#)
plot.package_rank_time, [9](#)
print.bioconductor, [10](#)
print.bioconductor_rank, [11](#)
print.cran_downloads, [11](#)
print.package_rank, [12](#)
print.package_rank_time, [12](#)

summary.bioconductor, [13](#)
summary.bioconductor_rank, [13](#)
summary.cran_downloads, [14](#)
summary.package_rank, [14](#)
summary.package_rank_time, [15](#)