Package: BiocPkgStats (via r-universe)

August 27, 2024

Title Use templates to create package stats reports

Version 0.0.15

Description Create package reports as RMarkdown documents. Rendering
RMarkdown documents will provide a brief summary of package
statistics. A starting date must be specified.

License Artistic-2.0

Imports BiocBaseUtils, BiocPkgTools, biocViews, graphics, lubridate,
stats, tools, utils, whisker

Suggests BiocStyle, knitr

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

Date 2024-05-15

Repository https://link-ny.r-universe.dev

RemoteUrl https://github.com/Bioconductor/BiocPkgStats

RemoteRef HEAD

RemoteSha da6323c0f866da309c6ec323405aa2d4322c11c3

Contents

---

downloadTrend *Plot the download activity for a package*

Description

Plot the download activity for a package

Usage

```
downloadTrend(package, since_date, include_recent = FALSE)
```

Arguments

package	`character(1)` The package to download activity for
since_date	`character()` The date from when to start looking at commit and issue history. This should be specified in the year, month, and day format, 'YYYY-MM-DD'. It can be a vector of dates that match the length packages.
include_recent	`logical(1)` Whether to include the latest month of download data. Usually, the data for the most recent month has incomplete numbers and is excluded (default is `FALSE`).

Examples

```
if (interactive()) {
downloadTrend("MultiAssayExperiment", "2017-05-01")
downloadTrend("cBioPortalData", "2017-05-01")
downloadTrend("RaggedExperiment", "2017-05-01")


downloadTrend("SingleCellMultiModal", "2017-05-01")
downloadTrend("curatedTCGAData", "2017-05-01")

# low downloads
downloadTrend("TENxIO", "2017-05-01")
downloadTrend("terraTCGAdata", "2017-05-01")
}
```

---

generateReport *Create a report for a given set of packages*

Description

The function compiles statistics for Bioconductor packages that have GitHub repositories. It uses functionality in `BiocPkgTools` to extract commit and issue history. A token is required to access the GitHub commit and issue history.

Usage

```
generateReport(packages, gh_org, since_date, outdir = ".", overwrite = FALSE)
```

Arguments

packages   character() A vector of valid package names that are installed

gh_org   character() The GitHub organization from which to read issue and commit
              data from. It can be either a scalar character to be recycled to the length of
              packages or a vector the length of packages for instances where packages are
              hosted under different GitHub organizations.

since_date   character() The date from when to start looking at commit and issue history.
              This should be specified in the year, month, and day format, 'YYYY-MM-DD'.
              It can be a vector of dates that match the length packages.

outdir   character(1) The directory in which to place rendered RMarkdown docu-
              ments, by default they will be placed in the current working directory.

overwrite   logical(1) Whether to overwrite an existing rendered product, i.e., a runnable
              RMarkdown document.

Details

Note that packages must be installed in order to determine the package type via the `biocViews`
field.

Authentication

The package uses the gh package calls from `BiocPkgTools`. Users must authenticate with a GitHub
Fine Grained Token and add the token using `gitcreds::gitcreds_set()`.

Examples

```
if (interactive()) {

generateReport(
    c(
        "MultiAssayExperiment", "cBioPortalData", "SingleCellMultiModal"
    ),
    gh_org = "waldronlab",
    since_date = "2017-05-01",
    overwrite = TRUE
)

}
```

---

generateTable   *Create a table of package metrics*

Description

The function compiles statistics for Bioconductor packages that have GitHub repositories. It uses functionality in `BiocPkgTools` to extract commit and issue history. A token is required to access the GitHub commit and issue history. See the `.token` argument in `?gh::gh` for details on its use.

Usage

```
generateTable(packages, gh_org, since_date)
```

Arguments

packages	`character()` A vector of valid package names that are installed
gh_org	`character()` The GitHub organization from which to read issue and commit data from. It can be either a scalar character to be recycled to the length of packages or a vector the length of `packages` for instances where packages are hosted under different GitHub organizations.
since_date	`character()` The date from when to start looking at commit and issue history. This should be specified in the year, month, and day format, 'YYYY-MM-DD'. It can be a vector of dates that match the length `packages`.

Details

Note that pull requests are a type of "issue" on GitHub; therefore, issues and pull requests are combined when reporting the number of closed issues since a specific date. For more information on how the queries are preformed, see the GitHub API documentation for the /repos/{owner}/{repo}/issues endpoint at <https://docs.github.com/en/rest/issues> and the gh R package. The following list describes each column in the returned `data.frame`:

- download.rank: The percentile rank of the package in its respective repository, e.g., software, data-experiment etc.
- avg.downloads: The average number of monthly downloads from distinct IP addresses since the given date
- med.downloads: The median number of montly downloads from distinct IP addresses since the given date
- num.revdeps: The number of "all" reverse dependencies including "Depends", "Imports", "LinkingTo", "Suggests", and "Enhances"
- issues.since: The number of closed issues (including pull requests) since the date given
- commits.since: The number of commits since the given date

Value

A `data.frame` of metrics, see the `details` section for specifics

Examples

```
if (interactive()) {

generateTable(
    packages = c(
        "MultiAssayExperiment", "cBioPortalData", "SingleCellMultiModal"
    ),
    gh_org = "waldronlab",
    since_date = "2019-05-01"
)

}
```

---

supportSiteStats	*Gather support site statistics for a given period*

Description

Gather support site statistics for a given period

Usage

```
supportSiteStats(from = "2021-01-01", to = "2021-12-31")

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

Arguments

from	character(1) date in ISO 8601 format ("YYYY-MM-DD")
to	character(1) date in ISO 8601 format ("YYYY-MM-DD")
x	bioc_support_stats The class object that has a print method
...	Additional arguments to the print method. Not used.

Value

a `bioc_support_stats` list class with components userdiff, toplevdiff, questdiff, and respdiff

Author(s)

Vincent J. Carey

Examples

```
if (interactive()) {
supportSiteStats()
}
```

Index