

Package ‘rdataretriever’

May 25, 2019

Title R Interface to the Data Retriever

Description Provides an R interface to the Data Retriever <<http://data-retriever.org/>> via the Data Retriever's command line interface. The Data Retriever automates the tasks of finding, downloading, and cleaning public datasets, and then stores them in a local database.

Version 2.0.0

Date 2019-5-2

BugReports <https://github.com/ropensci/rdataretriever/issues>

URL <https://github.com/ropensci/rdataretriever/>

Depends R (>= 3.4.0)

Imports reticulate (>= 1.8)

Suggests testthat (>= 1.0.0), DBI, devtools, RSQLite, RPostgreSQL

SystemRequirements Python (>= 3.0) and Python headers and libraries

License MIT + file LICENSE

LazyData true

RoxygenNote 6.1.1

NeedsCompilation no

Author Daniel McGlinn [aut] (<<https://orcid.org/0000-0003-2359-3526>>),
Pranita Sharma [aut] (<<https://orcid.org/0000-0002-5871-380X>>),
David J Harris [aut] (<<https://orcid.org/0000-0003-3332-9307>>),
Henry Senyondo [aut, cre] (<<https://orcid.org/0000-0001-7105-5808>>),
Shawn Taylor [aut] (<<https://orcid.org/0000-0002-6178-6903>>),
Harshit Bansal [aut] (<<https://orcid.org/0000-0002-3285-812X>>),
Max Pohlman [aut],
Ethan White [aut] (<<https://orcid.org/0000-0001-6728-7745>>)

Maintainer Henry Senyondo <henrykironde@gmail.com>

Repository CRAN

Date/Publication 2019-05-25 21:10:03 UTC

R topics documented:

datasets	2
download	3
fetch	3
get_citation	4
get_updates	5
install	5
install_csv	6
install_json	7
install_msaccess	7
install_mysql	8
install_postgres	9
install_sqlite	10
install_xml	10
reload_scripts	11
reset	11
use_RetrievePath	12
Index	13

datasets *Name all available dataset scripts.*

Description

Additional information on the available datasets can be found at url <https://retriever.readthedocs.io/en/latest/datasets.html>

Usage

```
datasets(keywords = "", licenses = "")
```

Arguments

keywords	Search all datasets by keywords
licenses	Search all datasets by licenses

Value

returns a character vector with the available datasets for download

Examples

```
rdataretriever::datasets()
```

download	<i>Download datasets via the Data Retriever.</i>
----------	--

Description

Directly downloads data files with no processing, allowing downloading of non-tabular data.

Usage

```
download(dataset, path = "./", quiet = FALSE, sub_dir = "",
         debug = FALSE, use_cache = TRUE)
```

Arguments

dataset	the name of the dataset that you wish to download
path	the path where the data should be downloaded to
quiet	logical, if true retriever runs in quiet mode
sub_dir	downloaded dataset is stored into a custom subdirectory.
debug	Setting TRUE helps in debugging in case of errors
use_cache	Setting FALSE reinstalls scripts even if they are already installed

Examples

```
rdataretriever::download('plant-comp-ok')
# downloaded files will be copied to your working directory
# when no path is specified
dir()
```

fetch	<i>Fetch a dataset via the Data Retriever</i>
-------	---

Description

Each datafile in a given dataset is downloaded to a temporary directory and then imported as a `data.frame` as a member of a named list.

Usage

```
fetch(dataset, quiet = TRUE, data_names = NULL)
```

Arguments

dataset	the names of the dataset that you wish to download
quiet	logical, if true retriever runs in quiet mode
data_names	the names you wish to assign to cells of the list which stores the fetched dataframes. This is only relevant if you are downloading more than one dataset.

Examples

```
## fetch the portal Database
portal = rdataretriever::fetch('portal')
class(portal)
names(portal)
## preview the data in the portal species datafile
head(portal$species)
vegdata = rdataretriever::fetch(c('plant-comp-ok', 'plant-occur-oosting'))
names(vegdata)
names(vegdata$plant_comp_ok)
```

get_citation

Get dataset citation information and a description

Description

Get dataset citation information and a description

Usage

```
get_citation(dataset)
```

Arguments

dataset	name of the dataset
---------	---------------------

Value

returns a string with the citation information

Examples

```
rdataretriever::get_citation('plant-comp-us')
```

get_updates	<i>Update the retriever's dataset scripts to the most recent versions.</i>
-------------	--

Description

This function will check if the version of the retriever's scripts in your local directory '~/retriever/scripts/' is up-to-date with the most recent official retriever release. Note it is possible that even more updated scripts exist at the retriever repository <https://github.com/weecology/retriever/tree/master/scripts> that have not yet been incorporated into an official release, and you should consider checking that page if you have any concerns.

Usage

```
get_updates()
```

Examples

```
rdataretriever::get_updates()
```

install	<i>Install datasets via the Data Retriever (deprecated).</i>
---------	--

Description

Data is stored in either CSV files or one of the following database management systems: MySQL, PostgreSQL, SQLite, or Microsoft Access.

Usage

```
install(dataset, connection, db_file = NULL, conn_file = NULL,
        data_dir = ".", log_dir = NULL)
```

Arguments

dataset	the name of the dataset that you wish to download
connection	what type of database connection should be used. The options include: mysql, postgres, sqlite, msaccess, or csv'
db_file	the name of the database file the dataset should be loaded into
conn_file	the path to the .conn file that contains the connection configuration options for mysql and postgres databases. This defaults to mysql.conn or postgres.conn respectively. The connection file is a file that is formatted in the following way:

```
host      my_server@my_host.com
port      my_port_number
user      my_user_name
password  my_password
```

data_dir	the location where the dataset should be installed. Only relevant for csv connection types. Defaults to current working directory
log_dir	the location where the retriever log should be stored if the progress is not printed to the console

Examples

```
rdataretriever::install('iris', 'csv')
```

install_csv	<i>Install datasets via the Data Retriever.</i>
-------------	---

Description

Data is stored in CSV files

Usage

```
install_csv(dataset, table_name = "{db}_{table}.csv",
            data_dir = getwd(), debug = FALSE, use_cache = TRUE)
```

Arguments

dataset	the name of the dataset that you wish to install
table_name	the name of the database file to store data
data_dir	the dir path to store data, defaults to working dir
debug	Setting TRUE helps in debugging in case of errors
use_cache	Setting FALSE reinstalls scripts even if they are already installed

Examples

```
rdataretriever::install_csv('iris')
```

install_json *Install datasets via the Data Retriever.*

Description

Data is stored in JSON files

Usage

```
install_json(dataset, table_name = "{db}_{table}.json",  
             data_dir = getwd(), debug = FALSE, use_cache = TRUE)
```

Arguments

dataset	the name of the dataset that you wish to install
table_name	the name of the database file to store data
data_dir	the dir path to store data, defaults to working dir
debug	Setting TRUE helps in debugging in case of errors
use_cache	Setting FALSE reinstalls scripts even if they are already installed

Examples

```
rdataretriever::install_json('iris')
```

install_msaccess *Install datasets via the Data Retriever.*

Description

Data is stored in MSAccess database

Usage

```
install_msaccess(dataset, file = "access.mdb",  
                 table_name = "[{db} {table}]", debug = FALSE, use_cache = TRUE)
```

Arguments

dataset	the name of the dataset that you wish to install
file	file name for database
table_name	table name for installing of dataset
debug	Setting TRUE helps in debugging in case of errors
use_cache	Setting FALSE reinstalls scripts even if they are already installed

Examples

```
rdataretriever::install_msaccess(dataset='iris', file='sqlite.db', debug=FALSE, use_cache=TRUE)
```

install_mysql

Install datasets via the Data Retriever.

Description

Data is stored in MySQL database

Usage

```
install_mysql(dataset, user = "root", password = "",
  host = "localhost", port = 3306, database_name = "{db}",
  table_name = "{db}.{table}", debug = FALSE, use_cache = TRUE)
```

Arguments

dataset	the name of the dataset that you wish to install
user	username for database connection
password	password for database connection
host	hostname for connection
port	port number for connection
database_name	database name in which dataset will be installed
table_name	table name specified especially for datasets containing one file
debug	Setting TRUE helps in debugging in case of errors
use_cache	Setting FALSE reinstalls scripts even if they are already installed

Examples

```
rdataretriever :: install_mysql(dataset='portal', user='postgres', password='abcdef')
```

install_postgres	<i>Install datasets via the Data Retriever.</i>
------------------	---

Description

Data is stored in PostgreSQL database

Usage

```
install_postgres(dataset, user = "postgres", password = "",  
  host = "localhost", port = 5432, database = "postgres",  
  database_name = "{db}", table_name = "{db}.{table}", bbox = list(),  
  debug = FALSE, use_cache = TRUE)
```

Arguments

dataset	the name of the dataset that you wish to install
user	username for database connection
password	password for database connection
host	hostname for connection
port	port number for connection
database	the database name default is postgres
database_name	database schema name in which dataset will be installed
table_name	table name specified especially for datasets containing one file
bbox	Optional extent values used to fetch data from the spatial dataset
debug	Setting TRUE helps in debugging in case of errors
use_cache	Setting FALSE reinstalls scripts even if they are already installed

Examples

```
rdataretriever::install_postgres(dataset='portal', user='postgres', password='abcdef')
```

install_sqlite *Install datasets via the Data Retriever.*

Description

Data is stored in SQLite database

Usage

```
install_sqlite(dataset, file = "sqlite.db",
               table_name = "{db}_{table}", data_dir = getwd(), debug = FALSE,
               use_cache = TRUE)
```

Arguments

dataset	the name of the dataset that you wish to install
file	Sqlite database file name or path
table_name	table name for installing of dataset
data_dir	the dir path to store the db, defaults to working dir
debug	Setting TRUE helps in debugging in case of errors
use_cache	Setting FALSE reinstalls scripts even if they are already installed

Examples

```
rdataretriever::install_sqlite(dataset='iris', file='sqlite.db', debug=FALSE, use_cache=TRUE)
```

install_xml *Install datasets via the Data Retriever.*

Description

Data is stored in XML files

Usage

```
install_xml(dataset, table_name = "{db}_{table}.xml",
            data_dir = getwd(), debug = FALSE, use_cache = TRUE)
```

Arguments

dataset	the name of the dataset that you wish to install
table_name	the name of the database file to store data
data_dir	the dir path to store data, defaults to working dir
debug	Setting TRUE helps in debugging in case of errors
use_cache	Setting FALSE reinstalls scripts even if they are already installed

Examples

```
rdataretriever::install_xml('iris')
```

reload_scripts	<i>Update the retriever's global_script_list with the scripts present in the ~/.retriever directory.</i>
----------------	--

Description

Update the retriever's global_script_list with the scripts present in the ~/.retriever directory.

Usage

```
reload_scripts()
```

Examples

```
rdataretriever::reload_scripts()
```

reset	<i>Reset the scripts or data(raw_data) directory or both</i>
-------	--

Description

Reset the scripts or data(raw_data) directory or both

Usage

```
reset(scope = "all")
```

Arguments

scope	All resets both scripst and data directory
-------	--

Examples

```
rdataretriever::reset('iris')
```

use_RetrieverPath	<i>Setting path of retriever</i>
-------------------	----------------------------------

Description

Setting path of retriever

Usage

```
use_RetrieverPath(path)
```

Arguments

path	location of retriever in the system
------	-------------------------------------

Examples

```
rdataretriever::use_RetrieverPath("/home/<system_name>/anaconda2/envs/py27/bin/")
```

Index

*Topic **utilities**

get_updates, 5

datasets, 2

download, 3

fetch, 3

get_citation, 4

get_updates, 5

install, 5

install_csv, 6

install_json, 7

install_msaccess, 7

install_mysql, 8

install_postgres, 9

install_sqlite, 10

install_xml, 10

reload_scripts, 11

reset, 11

use_RetrieverPath, 12