

Install R package ctrdata

Ralf Herold

2023-05-07

Install package ctrdata on a R system

The R Project website (<https://www.r-project.org/>) provides installers for the R system. It can be used with software products and graphical user interfaces such as R Studio, or Microsoft R Open, or from Visual Studio Code.

General information on the `ctrdata` package is available here: <https://github.com/rfhb/ctrdata>.

In R, execute:

```
install.packages("ctrdata")
```

For using the development version of package `ctrdata`, install from GitHub:

```
# install preparatory package  
install.packages(c("devtools", "httr"))  
devtools::install_github("rfhb/ctrdata")
```

Either of the above should install package `ctrdata` into the user's library.

Internet access via proxy

Functions in package `ctrdata` that start with `ctr...` require access to trial registers over the internet via the `https` protocol. Many organisations use transparent proxies that may *not* require users to do any configurations. However, if necessary, package `ctrdata` can use proxy settings set by users in their R session such as follows:

```
Sys.setenv(https_proxy = "your_proxy.server.domain:8080") # if needed  
Sys.setenv(https_proxy_user = "your_userid:your_password") # if needed
```

Additional installation aspects

The command line tools `perl`, `sed` and `php` (5.2 or higher) are required for `ctrLoadQueryIntoDb()`, the main function of package `ctrdata` to work with registers “EUCTR”, “CTGOV” and “ISRCTN” (but not “CTIS”). No other function in this package has this requirement.

MS Windows

For this requirement on MS Windows, the Cygwin environment has to be installed, into the local directory `c:\cygwin` (or any folder corresponding to `c:\cygw*`). The installation of a minimal Cygwin environment on MS Windows can be done with the help of package `ctrdata` in R as follows:

```
ctrdata::installCygwinWindowsDoInstall()
```

To update or correct the Cygwin installation:

```
ctrdata::installCygwinWindowsDoInstall(force = TRUE)
```

If internet access requires to specify a proxy (see above):

```
ctrdata::installCygwinWindowsDoInstall(proxy = "proxy.server.domain:8080")
```

To install Cygwin manually, download the setup executable from [here](#). In a MS Windows command window or Powershell window, use the following command line. The parameters are explained [here](#).

```
setup-x86_64.exe --no-admin --quiet-mode --verbose --upgrade-also --root c:/cygwin  
--site http://www.mirrorservice.org/sites/sourceware.org/pub/cygwin/  
--packages perl,php-jsonc,php-simplexml
```

macOS, Linux and other operating systems

The command line tools `perl`, `sed` and `php` (5.2 or higher) may already be available by default in some Linux and macOS systems. They are checked when running `ctrLoadQueryIntoDb()`; alternatively, check with `ctrdata:::checkBinary()`.

- In macOS, `homebrew` can be used to install `php` which may be needed starting from macOS 12 Monterey: first install `homebrew` and then run in Terminal the command: `brew install php`.
- In Linux, tools for installation vary by distribution (e.g., `sudo apt install php php-xml php-json`)
- Install script to automatically copy user's query from web browser, see [here](#)

Databases to use

At this time, a PostgreSQL, DuckDB, an SQLite or a MongoDB (local or remote) database can be used with the package `ctrdata`. A full SQLite database is provided in the R package `RSQLite`. Suggested installation instructions for PostgreSQL are [here](#) and for a local MongoDB server are [here](#); a remote MongoDB database server is accessible [here](#). See [here](#) for a speed comparison of the databases; recommended: DuckDB, PostgreSQL or MongoDB local server.

Purpose	Function call
Create SQLite database connection	<pre>dbc <- nodbi::src_sqlite(dbname = "name_of_my_database", collection = "name_of_my_collection")</pre>

Purpose	Function call
Create MongoDB database connection	<code>dbc <- nodbi::src_mongo(db = "name_of_my_database", collection = "name_of_my_collection")</code>
Create PostgreSQL database connection	<code>dbc <- nodbi::src_postgres(dbname = "name_of_my_database"); dbc[["collection"]] <- "name_of_my_collection"</code>
Create DuckDB database connection	<code>dbc <- nodbi::src_duckdb(dbname = "name_of_my_database", collection = "name_of_my_collection")</code>
Use connection with <code>ctrdata</code> functions	<code>ctrdata::{ctrLoadQueryIntoDb, dbQueryHistory, dbFindIdsUniqueTrials, dbFindFields, dbGetFieldsIntoDf}(con = dbc, ...)</code>

Attach package `ctrdata`

After completing the installation, attach the package from the user's library. This will also check the availability of the additional installation requirements as mentioned above:

```
library(ctrdata)
```

Remember to respect the registers' terms and conditions (see `ctrOpenSearchPagesInBrowser(copyright = TRUE)`). Please cite this package in any publication as follows: Ralf Herold (2023). `ctrdata`: Retrieve and Analyze Clinical Trials in Public Registers. R package version 1.13.0. <https://cran.r-project.org/package=ctrdata>