Set default download repo

Practice notebook for DSC 205 Spring 2022

1 README

In this notebook, you're learning and practicing how to set the repository for R package uploads and upgrades using a self-written function, and use the function in the `.Rprofile` init file for R.

2 Standard notebook instructions

- `[ ]` Work through this notebook at your own pace. When you're done, check a task off by typing `C-c C-c` on the line with the bullet point. Check this task off for practice!
- `[ ]` You can close bullet points with the `<TAB>` key on your keyboard to reduce the clutter on the screen. Close this long paragraph now by going to the line with the bullet point `*` in it, then check the task off.
- `[ ]`

  Make sure that you can run R in this buffer by executing the code block below (`C-c C-c` with the cursor ON the block) and then save the file (`C-x C-s`).

  ```
  print("hello world")
  ```

- `[ ]` Sometimes, you want to check what happened in the R session buffer. Make sure that you find that buffer (e.g. `C-x C-b` for `list-buffers`).
- `[ ]`

  Practice: go to the buffer `*R*` now, and run the command shown below, then come back here again.

  ```
  search()
  ```

  Did you know what this command would do? Do you recognize the output in the session buffer? If not, type `?search` for help, then return here again to continue.

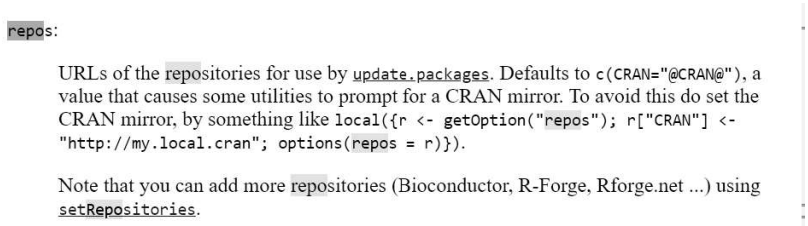
- `[ ]` In case you have multiple buffers open: remember that `C-x 1` closes all but the current buffer, and `C-x o` cycles through them.
- `[ ]` If you have any trouble with executing the `hello world` program, analyze the problem by yourself first. Typical sources of errors are:
  - Can you write to the current directory? (Windows may forbid it)
  - Is this file an Org-file? (Syntax highlighting/modeline check)
  - Did you use the correct key sequence? (Check caps lock e.g.)
  - Do you have the right code block header arguments? (You only need `:session :results output` for now.
  - Are you working with an old `*R*` session and writing to some far-flung directory? (check with `getwd()` and reset with `setwd()`).

3 Change package installation repository

- `[X]` The utility function `options` allows you to set and see the default download repository.

- [X]

  Open the (local) help for `options` (you need to go to the `*R*` session buffer to do this, or open a new buffer with `M-x R`) - or google the documentation.



repos:

URLs of the repositories for use by update.packages. Defaults to c(CRAN="@CRAN@"), a value that causes some utilities to prompt for a CRAN mirror. To avoid this do set the CRAN mirror, by something like local({r <- getOption("repos"); r["CRAN"] <- "http://my.local.cran"; options(repos = r)}).

Note that you can add more repositories (Bioconductor, R-Forge, Rforge.net ...) using setRepositories.

Figure 1: Part of the options help revealing the attribute repos

- [X]

  Why is googling the help for this function only second best compared to looking up the help using R on your PC?

  > ANSWER: because the local documentation is aligned with the version of R that you're actually using right now, while Google will serve any documentation using their (unknown) algorithm.

- [X]

  Check current setting of the download repo with the R function `getOption`.

```
getOption("repos")
```

```
[1] "https://cloud.r-project.org/"
```

- [X]

  Write a function `frepos` that changes the installation repository to the URL `"https://cloud.r-project.org/"`. The function should take one argument only - a `character` vector with the repo URL as its only element.

```
frepos <- function(x) options(repos=x)
r <- c("https://mirrors.nics.utk.edu/cran/")
frepos(r)
```

- [X] The chosen URL automatically picks the nearest CRAN mirror. Though the output from 1 is set to `silent`, you should see the mirror site URL in the echo area at the bottom.
- [X]

  Re-run the `getOption` function on the attribute of `options` that you just changed in `frepos` to check the download repo.

```
getOption("repos")
```

```
[1] "https://mirrors.nics.utk.edu/cran/"
```

- To test the function, you need to install a package. Let's use `ggplot2` for that, since we're going to need it soon.

4 Install and test a package from the new repo location

- `[X]`

  Split the screen horizontally with `C-x 2` to see both this file and the R session in one window (see figure 2 - to open the figure, use the toggling command `C-x C-c C-v`).

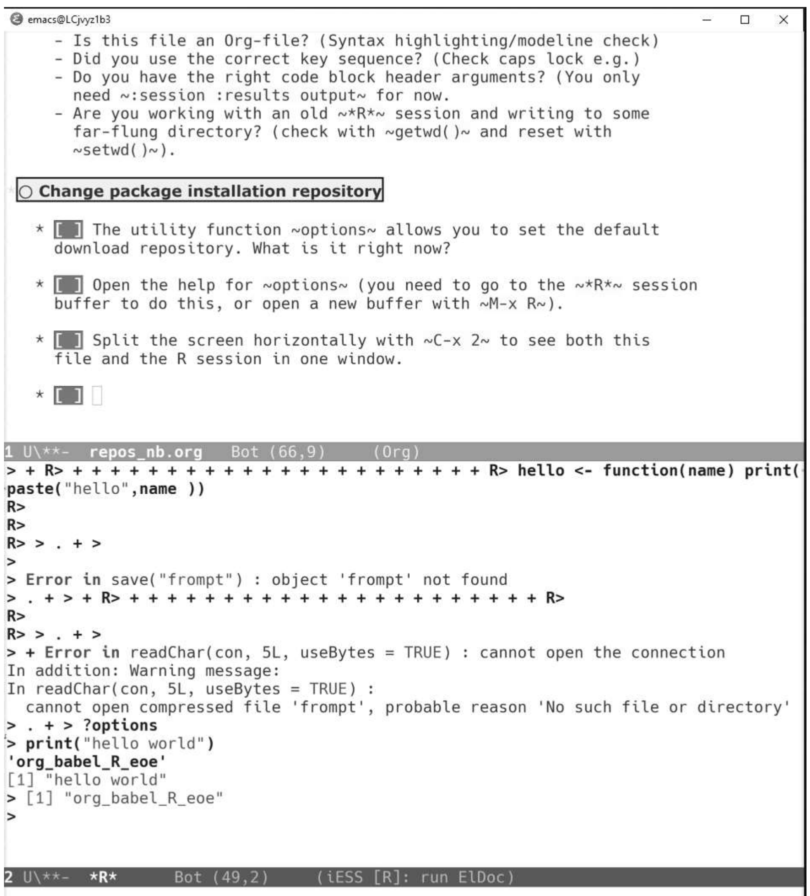


Figure 2: Split window with Org (top) and **R** (bottom)

- `[X]`

  Install and load `ggplot2` (if you already have the package, re-install it - this won't hurt unless your work relies on an older version of the package)[1].

```
install.packages("ggplot2")
library(ggplot2)
```

The result will be a list of currently loaded packages.

- [ ]

  Test the installation with the command `qplot(rnorm(100))`

```
qplot(rnorm(100))
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```


```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```


- [ ]

  Since the output is graphics, move the output into this Org-file by adding the following arguments to the code block header line of 1, and then re-run the command.

```
:results output graphics file :file plot.png
```

```
qplot(rnorm(100))
```

- [X]

  Note that your file will be saved whereever your current working directory is - find it with `getwd()` and open your file, or change the working directory with `setwd()`, change the path in the header argument and re-run the code block 1.

```
getwd()
setwd("c:/Users/birkenkrahe/Documents/GitHub/")
getwd()
```

```
[1] "c:/Users/birkenkrahe"
[1] "c:/Users/birkenkrahe/Documents/GitHub"
```

5 Save your function and use it in your R init file

- [X]

  Save the function `frepos` to your home directory. Using the absolute path (e.g. `"c:\Users\birkenkrahe\frepos"`) to the file is safer here.

```
save(frepos, file='~/frepos')
```

- [X] Check that the (binary) function file is there.

6 Use function in your R init file

- [X] Load the function to your `.Rprofile` file. Create one if necessary. This file contains R code and is executed by `Rscript`. Put the code here and test it.

```
load('~/frepos')
frepos(c("https://cloud.r-project.org/"))
```

- To test the `.Rprofile` setup, start another R session and test the download repo location with `getOption` as before.

```
getOption("repos")
```

```
[1] "https://cloud.r-project.org/"
```

7 Summary

- [ ] Display options can be changed with `options`
- [ ] The function `getOption("x")` fetches the value of the option x
- [ ] When R starts, it loads a file `.Rprofile` from the current working directory
- [ ] To display graphics, a device needs to be opened
- [ ] You can look at the devices with `dev.list()` and open a new device with `dev.new()`.
- [ ] To find out about the function arguments, enter the function name without arguments or enter `args(functionName)`
- [ ] To find a function in currently installed packages, use `??`
- [ ] Whenever you find yourself doing the same thing several times, e.g. run a function with different values, write your own
- [ ] You can check the path where packages are downloaded to with `.Library`

Footnotes:

[1] If your OS reports that it cannot write to the library location (e.g. because it is in the system partition on Windows), change the location by running `.libPaths(new)` and/or adding this command to your `.Rprofile` file.

Author: Marcus Birkenkrahe
Created: 2022-03-11 Fri 13:16
Validate