

Lab 2 Installing R Packages (Free Server)

B. Sosnovski

10/12/2021

Installing R packages from the source

In this lab, you will install the R package `gapminder`. The name of the package you will need to install for different labs may change.

General Goals

In this lab, you will learn the following:

- How to install R packages in Cocalc with a free server.

Installing R packages

Do the following steps:

1. Download the package file to your computer (usually found in CRAN).
2. Go to your `Home` directory in your project in Cocalc.
3. Upload the package file to your project.
4. Double-click on the file and then extract all.
5. Create a Linux Terminal file in your project or open one that is already in your project.
6. When the prompt appears (`~$`), type an upper-case letter `R` and press the Return Key. This will start up R.
7. Type `install.packages("gapminder")`

The following will be displayed:

```
Installing package into '/usr/local/lib/R/site-library'
(as 'lib' is unspecified)
Warning in install.packages("gapminder") :
  'lib = "/usr/local/lib/R/site-library"' is not writable
Would you like to use a personal library instead? (yes/No/cancel)
```

8. Type `yes`.

```
The following will be displayed:
Would you like to create a personal library
'~/R/x86_64-pc-linux-gnu-library/4.1'
to install packages into? (yes/No/cancel)
```

9. Type `yes` again.

Then something similar to the following will be displayed:

```
Installing package into '/home/user/R/x86_64-pc-linux-gnu-library/4.1'
(as 'lib' is unspecified)
Warning: unable to access index for repository https://cloud.r-project.org/src/contrib:
cannot open URL 'https://cloud.r-project.org/src/contrib/PACKAGES'
Warning message:
package 'gapminder' is not available for this version of R

A version of this package for your version of R might be available elsewhere,
see the ideas at
https://cran.r-project.org/doc/manuals/r-patched/R-admin.html#Installing-packages
```

All that means is that we need to adjust the code to specify the path to the newly created library directory and the type of file installed.

First, let's confirm that the library directory was created and its path.

10. (Optional) Type the command: `.libPaths()`

Then something similar to the following will be displayed:

```
[1] "/home/user/R/x86_64-pc-linux-gnu-library/4.1"
[2] "/usr/local/lib/R/site-library"
[3] "/usr/lib/R/site-library"
[4] "/usr/lib/R/library"
>
```

Note: You can see that the library directory was created and appears in line [1].

11. Then enter the following

command: `install.packages("gapminder", "/home/user/R/x86_64-pc-linux-gnu-library/4.1", repos = NULL, type="source")`

The following should be displayed:

```
* installing *binary* package 'gapminder' ...
* DONE (gapminder)
>
```

This means that the package `gapminder` is now installed in your project.

Note: The code syntax you have above has the following

form: `install.packages("package_name", "your_personal_library_directory", repos = NULL, type="source")`

You will have to replace the `package_name` and library directory `your_personal_library_directory` in the code according to the package you are installing and the library path in your project.

Next time you install another package in your project's `Home` directory, you will not need to create the library because it has already been created (SKIP steps 8 and 9).