Science Communication: Research Productivity and Data Analytics using Open Source Software

Mohit Garg

Central Library

IIT Delhi

Week: 04

Lecture 16 : Packages and Help

Dear Learners, Welcome again. So, this is our last lecture of this week on working on R. In this lecture, I will be discussing the packages and Help, how we can install the packages and how we can call the packages, what are the different components of the packages and if we are stuck anywhere, how we can seek the help.

So, first we will discuss the packages and then I will discuss Help. So, the availability of various kinds of packages in R makes it a powerful tool for data analysis, visualization and other computation. So, a package is a collection of code that is written specifically for a given task.

There are different kinds of analysis there. So, for that R has a strong community and that community develops a different kind of package for that particular kind of task. So, say for example, you want to do it for a sentiment analysis or you want to use R for text mining. So, there are different packages that you can use for that particular analysis.

And the R community regularly develops these new packages and updates the existing one. Presently, it has more than 20,000 packages in R for different kinds of analysis and computation. And all these packages in R we can see on the CRAN page. This is the CRAN page and this is the same page where we have started our first lecture when we installed R. And if you see on the left hand side of this page, there are different details there.

These are mirrors and all those things. But under software you will find the packages. And if you click on packages, it has listed all the packages, like either by date of publication when the package was published or by the name. So, we will click on each of the links. So, very recently, these are some of the packages that have been published.

So, here you can see the packages by the name and very small description of what exactly that package is. So, for example, we have the package for ANOVA and there are like 44 results. And if I go, so this is a, this particular package is ANOVA Sign2, interactive document for working with analysis of variance. And there are like many different tasks

specific packages are there. There is another package WordCloud if you want to generate WordCloud.

So, this particular package is for WordCloud and there is another package for WordCloud. So, there are like different packages listed here and you can select each of the packages as per the requirement of your analysis and all those packages are free. So, sometimes we are worried that the software is free but packages are, we have to pay something. But these packages are free, you can install them and then you can use it for your analysis. All these packages come with various kinds of details.

So, let us see what are those different details of packages with the example. So, let us take the example of bibliometrics. So, this is a package for comprehensive science mapping analysis. If you see here, this particular package is for comprehensive science mapping analysis. Then the description is there of that tool for quantitative research in scientific and bibliometrics. It provides various routines for importing bibliographic data from Scopus, Web of Science, Dimension and then Lens, PubMed and you can do different kinds of analysis like co-citation, coupling and all.

So, it has various other analysis features also. This is very brief about this particular package. Then this version is 4.1.4. Then it depends on the R version which is either 3.3.0 or higher than this. So, whenever you are using this version of this package, you must have this version. Then it has these many imports.

So, whenever you are installing this package, these are also some packages of R. So, this is a single package which depends on multiple other packages. So, like for example, this Ggplot2 is a popular package for data visualization. and it depends on many other packages. Then we have a Stringy for string processing. Then we have this tdtext and it is for like other text mining and other tools.

Then we have this shiny. So, Shiny is a package in R which is used for creating the web-based interface. So, we will see all those things when we will be doing the analysis. If you see, all those packages are listed on CRAN and they have all these details. It also suggests that you can have these packages also.

Then this particular version was published on 28 November of 2023. Then these are the authors and they are like, these are the ORCID of this particular author. And among these authors, this particular author is the maintainer. So, the maintainer who maintains this package. If there are any bugs and those things, what are the bugs and all those things you can report on this page.

And also there are many issues already listed here. Then it has a license. So, what exactly is that license? So, a license becomes very important whenever you are using any of the

open source software or the package. So, it has the details of the license. So, this particular package has this particular GPL3 license.

Under this license, it has all the details of what exactly the copyright is and other those things. Then it has the URL. These are the URLs where you can get these packages. Then no compilation is required. Then citation, I have already discussed the citations of our software and the packages. So, it is very important whenever you are using this package or any other packages, you must cite it.

So, if I click on it, it will show how to cite this particular package. Then the materials and those CRAN checks. So, these are the CRAN checks. And then finally the reference manual, which is very important. How to use the package? How to call the particular function in that package? So, this is the whole reference manual.

So, each package has their own reference manual. How can you use the package? What are the different functions? How you have to call or how you have to do the analysis in that particular package? So, if you see here, like this particular histplot function is used to plot the historical co-citation network and there are many other functions there. So, similarly we see for the Shiny, we have this Shiny and we have also for TDtext, we can click here. Then we have Stringy also. So, this is the reference manual of Stringy and then Ggplot2.

So, each package has a comprehensive kind of reference manual, which you can use during the analysis. So, if you see here, like so for Shiny, all the details are there. Then for TDtext, the details are there. Then instead, there are multiple ways of installing the package. So, it provides you the file also.

We will see how to install the package. So, you can install this or you can install the binaries also. Binaries depending on your operating system, either Windows or Mac. If you want to see the old source, because this is the latest version, if you want to see the old source of this particular package, you can see here. and these are the older files of this particular package. So, each package in R comes with these many details.

So, whenever you are using any of the packages, you must go first to the page of where that package is, what the different dependencies are there, and exactly the manuals. Becauseș manuals will be very helpful during the analysis. So, packages in R can be installed by using multiple ways. So, the first way is using the command line and the second way is using the graphical user interface of R or RStudio. Then in the third way, like if you want to install from a local repository or if you have already downloaded the packages and you want to install it from a local repository, you can do that also. So, first we will see how we can install the package by using the command line and then we will see other options.

So, to install the package, you have to use this function install.packages and under this you have to mention the package name. So, say for example, I am installing some package and say for example, I am installing this ZFA. So, I have to mention ZFA. So, at this moment you will be required to use the internet because we are installing the package from the repository.

So, once this package is installed, you can use it for your analysis. Another way of installation is that from the local location. So, say for example, you have already downloaded this package. Say for example, this is the package. I am just downloading this package. Now, this package is installed. I have to call the location of this particular package. So, under downloads, it depends on where that file is downloaded. So, by default my download directory is Download. So, it will be in Download. So, I will just, this is the path of my package location and if I run this.

So, whenever you are installing the package locally, you make it sure that the other dependency of this package is already installed. For this particular case, if you see the page on CRAN, it says that it imports tweedie and VGAM. And the same error is showing here that this tweedie and VGAM are not available for package accessor. So, these are the main dependencies. So, whenever you are doing the installation by this way, you have to be very careful about the dependency of this particular package.

And the second way is mostly not preferred by our users. but you must know that we can install in this way also. And the third way is the easiest way and the most popular way to install the package is the GUI installation. So, how can we do that? By going to the tools in this menu bar we can click on the install packages and here you have to just return ‘assessor’. So, this was our package which we were not able to install using this way. So, and then we don't have anything like these. some of the different repositories are there and then we will click on the install dependency.

And now if we will install, this package will be installed. So, this is how you can install the package and this is the most convenient one to install the package. In the first case, we have installed the package by typing this command name, but whenever we are installing the package by using the GUI, we don't have to type it down this command. It will automatically type down and the package will be installed. So, this is how you can install the packages in R.

Then you can check the number of packages installed in your system by using this command. It is installed.packages if I run this. So, these many packages are installed in my system.

So, you can use this installed.packages command to list all the packages installed in your system. Now, after installation, you have to call this package. So, for each session in R, you have to call the package which is the package you want to use. So, to call the

package, we have to use the function library. So, what will we call the package? So, we will have to use it in this way. This is a function name and under this function name, we have to write the package name.

So, these are the two requirements whenever you are doing the analysis in R for a specific kind of package. First, you make it so that the package is installed in the R and in the second, you must call it the package before doing the analysis. Now, say for example, this particular package is already installed in my machine if I call it.

So, this particular package is called. Now, if I have to run this, it will show these many options are there. Now, take an example that if I take a new session. And now, if I do like this. you will see that no options are showing Biblioshiny and other things. Many times these errors have been reported that sometimes people ask that I am not able to see the function, that particular function. So, this is a very basic and fundamental step that whenever you are doing the analysis, you make it so that it is well installed, the correct version is installed and you have called that package.

Say for example, you are currently using this particular package and after one year you are doing another analysis or using this package. And the community which manages these packages are continuously updating the things. If there are any bugs there or if there are any other issues are there. So, it's another suggestion that whenever you are doing the analysis on a package which was installed long back ago, you must update that package. So, to update the package you can use the command update.packages.

You have to give the name of that particular package. So, after the installation of the package and after the calling of that package, another thing which is very important during the analysis is the path. So, the whole calling in the R depends on the path, where exactly that particular file is. So, path is basically the location of that particular file. So, by default in Windows the path will be your Documents directory.

So, what directory you are using it in. So, if I run this, it will give me the output that I am using this particular directory. Now, if you have this document directory and whatever the data set you have, you have to keep in the documents. If you have to call it here. Another way is that you can provide the whole path like what I have done here. When I was installing the package, I provided the whole path.

So, where this file is located. So, in this way you don't have to worry about the working directory. But say for example, if I do like this. So, here you will see the directory of my Documents only. You won't find the data which is located in downloads. So, to do that you must use the right directory. whenever you like after calling the package whenever you are doing the analysis you must use the right directory where that data is or whether where that package is located.

If you want to like, change the directory. So, if you want to like my default directory is document but if you want to change this directory to downloads. Now, what you can do for that is to use this function setwd (set working directory) and under setwd you have to give the path. So, like say for example, I just copied this here and now Downloads. Something is missing. So, this is another important thing if you notice here this '+' symbol is another good feature. like whenever you have missed anything in the command it will show that this '+' sign.

This '+' sign signifies that you missed something. So, what I missed I missed here double notes. Now, if I run this and now if I see the directory. it will say that this is the directory user del download. So, now if I run here now if I see the path of the location. So, now it's showing all those things. So, this is another important concept you must remember whenever you are working with R. Whether you have set the right path, whether you are on the right path or you are calling the right path.

So, these are some of the other common issues that have occurred among the users that they couldn't find that particular location. So, this is about the packages and path. The path becomes very important whenever you are doing the analysis. So, you must set the right path during the analysis. So, whenever you will be doing the analysis you will be facing many issues you will be seeing many errors during the analysis. So, the help feature in R is very beautiful and it gives a straightforward way to solve those particular errors.

So, there are different ways to get the help from its interface itself. So, one way is using the function called Help. So, under Help you just like say for example you want to know about the Help about mean. So, you can simply run it and it will give that. okay if you want to know about Help of mean and so this is arithmetic mean and this is how you can call this function. So, it gives one example also like say for example you run this example.

It is created. Now if I create now if I check the value of this mean it's 8.75. So, this is how you can use this Help function to know about the particular. if you face any kind of error if you want to know how to calculate mean or another computation in R. So, there is a shortcut for instead of using this Help function, you can use this '?' sign and then you can simply write like mean and if you do like this it will give the same result.

So, this is how to get Help on this particular concept. Now you want to see the example of that particular function. So, you can use that function Example and if you run like this example it will give you the whole example that this is how you have to process whenever you are using means. Now like say for example if you don't want to use this function and if you want it through this menu bar. So, there are different Help options I already discussed in the starting lecture when I was discussing the interface of RStudio

but you can search here. So, if you click here on search R Help it will automatically come here and like say for example you want maybe about correlation.

So, these are some of the different Help pages for this particular correlation. then say for example I am going with this. So, this is how you can check the Help in R by using either the function or by using the '?' symbol or using the interface itself. So, this is about the Help that you can take from this interface whenever you are stuck to any kind of analysis or you are starting you want to know how to go ahead with this particular example of this particular analysis. Okay, so this is the end of the fourth week. In this week we learned about the installation of R, then some of the shortcomings of using the R interface and then we moved ahead to install the RStudio and after that we have seen the different features in RStudio. and how RStudio is useful during the analysis.

Then we have discussed the different value types in R. We have discussed the numeric, we have discussed the characters, we have discussed the integers, logical, and we have discussed complex and rows. Then we have discussed the data structures in R like vector and data-frame, list, matrices, factor and array. And in the last we have discussed the packages, how we can install the packages and then the importance of setting up the path and in the end we have discussed how we can use the Help function or if we have to know about the different examples. So, I request you all to give a practice on all these concepts. So, these concepts will be used when we will be doing the analysis and if you have any of the issues or if you stuck to any of the points, please let us know through our discussion forum. Thank you.