STAT 40001/STAT 50001 Statistical Computing

Lecture 1

Department of Mathematics and Statistics

Outline - Getting Started with R

What is R?

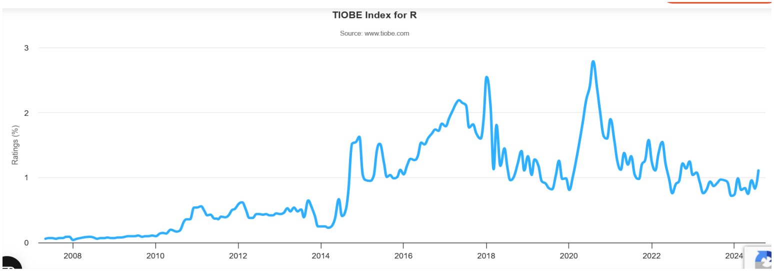
- R is an open-source environment for statistical computing and visualization. It is the product of an active movement among statisticians for a powerful, programmable, portable, and open computing environment, applicable to the most complex and sophisticated problems, as well as “routine” analysis, without any restrictions on access or use.
    - Performs a variety of simple and advanced statistical methods
    - Produces high quality graphics
    - R is a computer language so we can write new functions that extends R’s uses
- R was initially written by Ross Ihaka and Robert Gentleman at the University of Auckland in Auckland, New Zealand (hence the name).
- The official R home page is http://www.R-project.org

Install R

- R can be installed on Windows, MacOS, and Linux
- You will want to install the base system.
- In addition to the base system there are user contributed add-on packages. Packages are a collection of functions, examples and documentation that usually focus on a specific task.
- The base system contains some packages. To install an additional package, say survival, be connected to the Internet and type
  > install.packages("survival")
- You will be asked to select the mirror site nearest to you, after that everything is automatic.
- Before using the contents of the package we need to load it,
  > library(survival)
- See the R website for a complete list of contributed packages

R is Popular!!

- There are 21145 packages as of 08/15/2024 `https://cran.r-project.org/web/packages/`
- Many many R user groups
  - Chicago R User Group (CRUG) (3,987 members as of 08/15/2024
  - IndyUseR Group (637 members as of 08/15/2024)
- R-Ladies (`https://rladies.org/`)



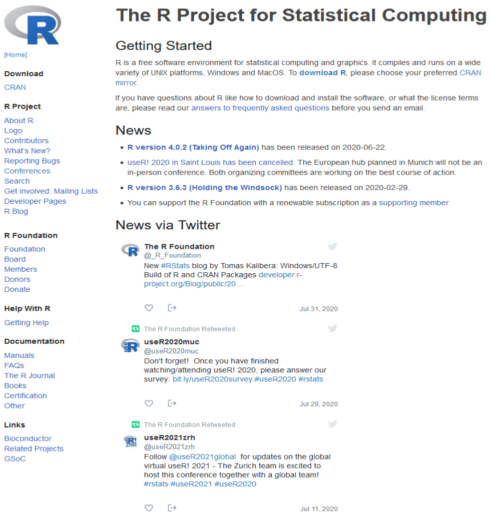
Useful links

- Quick-R at `http://www.statmethods.net/`
- R-Tutorial at `https://www.r-tutor.com/`
- R-bloggers at `https://www.r-bloggers.com/2015/12/how-to-learn-r-2/`
-

Installing R

Visit www.r-project.org

The homepage appears as below.



[Home]

Download

CRAN

R Project

About R
Logo
Contributors
What's New?
Reporting Bugs
Conferences
Search
Get Involved: Mailing Lists
Developer Pages
R Blog

R Foundation

Foundation
Board
Members
Donors
Donate

Help With R

Getting Help

Documentation

Manuals
FAQs
The R Journal
Books
Certification
Other

Links

Bioconductor
Related Projects
GSoC

The R Project for Statistical Computing

Getting Started

R is a free software environment for statistical computing and graphics. It compiles and runs on a wide variety of UNIX platforms, Windows and MacOS. To **download R**, please choose your preferred CRAN mirror.

If you have questions about R like how to download and install the software, or what the license terms are, please read our answers to frequently asked questions before you send an email.

News

- **R version 4.0.2 (Taking Off Again)** has been released on 2020-06-22.
- useR! 2020 in Saint Louis has been cancelled. The European hub planned in Munich will not be an in-person conference. Both organizing committees are working on the best course of action.
- **R version 3.6.3 (Holding the Windsock)** has been released on 2020-02-29.
- You can support the R Foundation with a renewable subscription as a supporting member

News via Twitter

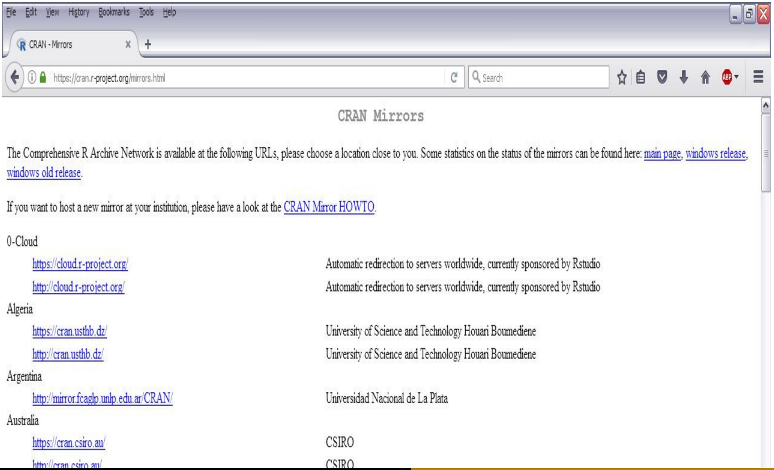
The R Foundation
@_R_Foundation
New #RStats blog by Tomas Kalibera: Windows/UTF-8 Build of R and CRAN Packages.developer.r-project.org/Blog/public/20…
Jul 31, 2020

The R Foundation Retweeted
useR2020muc
@useR2020muc
Don't forget! Once you have finished watching/attending useR! 2020, please answer our survey: bit.ly/useR2020survey.#useR2020 #rstats
Jul 29, 2020

The R Foundation Retweeted
useR2021zrh
@useR2021zrh
Follow @useR2021global for updates on the global virtual useR! 2021 - The Zurich team is excited to host this conference together with a global team! #rstats #useR2021 #useR2020
Jul 11, 2020

Installing R

The Comprehensive R Archive Network (CRAN) allows selection of a regional computer network from which you can download R. If you click on the CRAN link, you will be shown a list of network servers all over the world. Choose the mirror site closer to you.


File Edit View History Bookmarks Tools Help
CRAN - Mirrors
https://cran.r-project.org/mirrors.html
Search

CRAN Mirrors

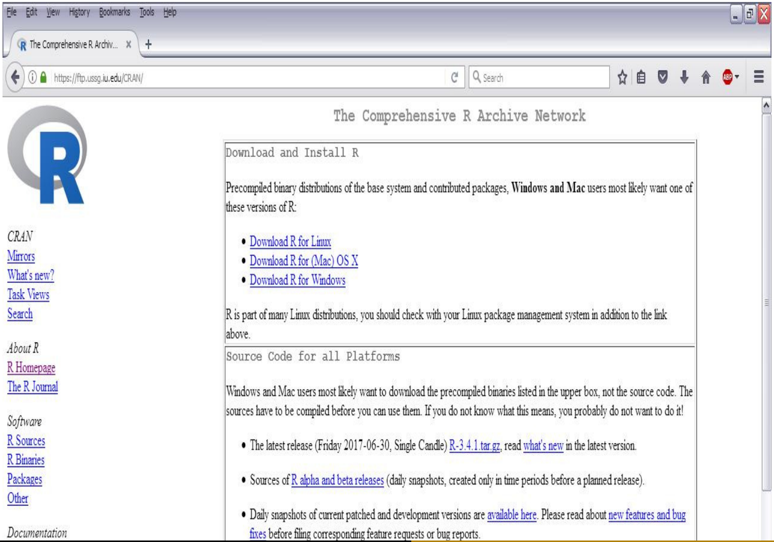
The Comprehensive R Archive Network is available at the following URLs, please choose a location close to you. Some statistics on the status of the mirrors can be found here: main page, windows release, windows old release.

If you want to host a new mirror at your institution, please have a look at the CRAN Mirror HOWTO.

0-Cloud	
https://cloud.r-project.org/	Automatic redirection to servers worldwide, currently sponsored by Rstudio
http://cloud.r-project.org/	Automatic redirection to servers worldwide, currently sponsored by Rstudio
Algeria	
https://cran.usthb.dz/	University of Science and Technology Houari Boumediene
http://cran.usthb.dz/	University of Science and Technology Houari Boumediene
Argentina	
http://mirror.fcaglp.unlp.edu.ar/CRAN/	Universidad Nacional de La Plata
Australia	
https://cran.csiro.au/	CSIRO
http://cran.csiro.au/	CSIRO

Installing R

If we choose Indiana University the webpage shown below will appear.


File Edit View History Bookmarks Tools Help
The Comprehensive R Archiv...
https://ftp.ussg.iu.edu/CRAN/
Search

The Comprehensive R Archive Network

CRAN
Mirrors
What's new?
Task Views
Search

About R
R Homepage
The R Journal

Software
R Sources
R Binaries
Packages
Other

Documentation

Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- Download R for Linux
- Download R for (Mac) OS X
- Download R for Windows

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

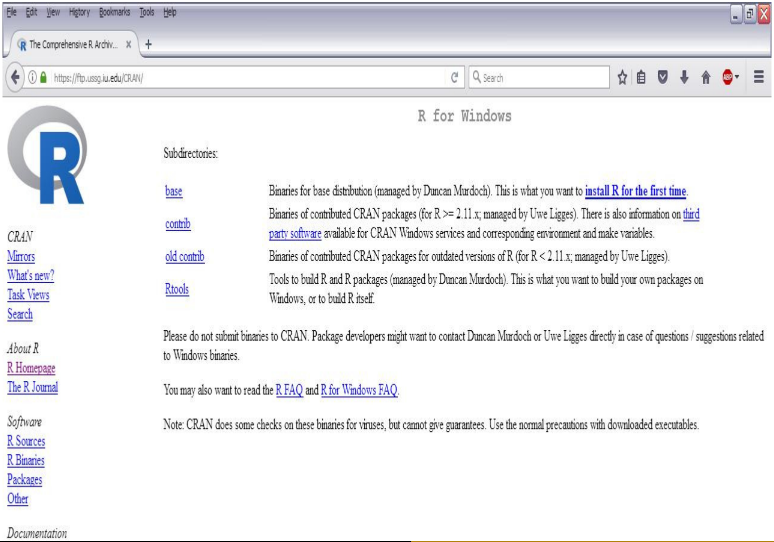
Source Code for all Platforms

Windows and Mac users most likely want to download the precompiled binaries listed in the upper box, not the source code. The sources have to be compiled before you can use them. If you do not know what this means, you probably do not want to do it!

- The latest release (Friday 2017-06-30, Single Candle) R-3.4.1.tar.gz, read what's new in the latest version.
- Sources of R alpha and beta releases (daily snapshots, created only in time periods before a planned release).
- Daily snapshots of current patched and development versions are available here. Please read about new features and bug fixes before filing corresponding feature requests or bug reports.

Installing R

If you click on Download R for Windows the page below will appear.


File Edit View History Bookmarks Tools Help
The Comprehensive R Archiv...
https://ftp.ussg.iu.edu/CRAN/
Search

R for Windows

Subdirectories:

base	Binaries for base distribution (managed by Duncan Murdoch). This is what you want to install R for the first time.
contrib	Binaries of contributed CRAN packages (for R >= 2.11.x; managed by Uwe Ligges). There is also information on third party software available for CRAN Windows services and corresponding environment and make variables.
old contrib	Binaries of contributed CRAN packages for outdated versions of R (for R < 2.11.x; managed by Uwe Ligges).
Rtools	Tools to build R and R packages (managed by Duncan Murdoch). This is what you want to build your own packages on Windows, or to build R itself.

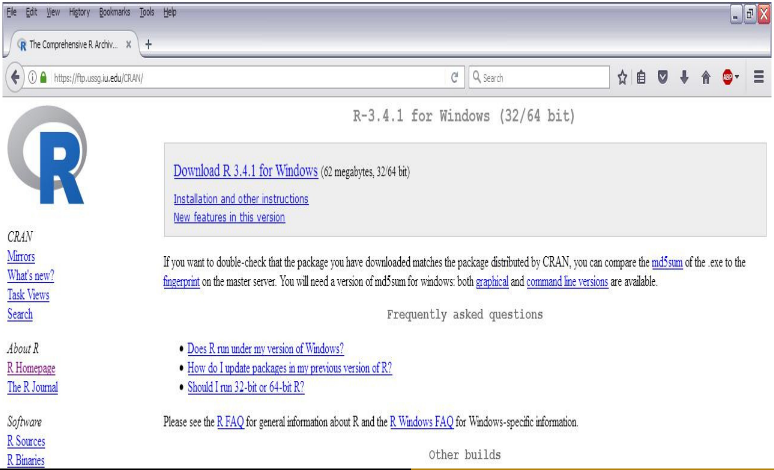
Please do not submit binaries to CRAN. Package developers might want to contact Duncan Murdoch or Uwe Ligges directly in case of questions / suggestions related to Windows binaries.

You may also want to read the R FAQ and R for Windows FAQ.

Note: CRAN does some checks on these binaries for viruses, but cannot give guarantees. Use the normal precautions with downloaded executables.

Installing R

If you click on base the page below will appear where you can download Download R 4.02 for Windows which is 84 megabytes, 32/64 bit. Note that this is the latest version at the time of writing this note, and you may see a more recent versions.



File Edit View History Bookmarks Tools Help
The Comprehensive R Archiv...
https://ftp.ussg.iu.edu/CRAN/
Search

R-3.4.1 for Windows (32/64 bit)

Download R 3.4.1 for Windows (62 megabytes, 32/64 bit)

Installation and other instructions
New features in this version

If you want to double-check that the package you have downloaded matches the package distributed by CRAN, you can compare the md5sum of the .exe to the fingerprint on the master server. You will need a version of md5sum for windows: both graphical and command line versions are available.

Frequently asked questions

- Does R run under my version of Windows?
- How do I update packages in my previous version of R?
- Should I run 32-bit or 64-bit R?

Please see the R FAQ for general information about R and the R Windows FAQ for Windows-specific information.

Other builds

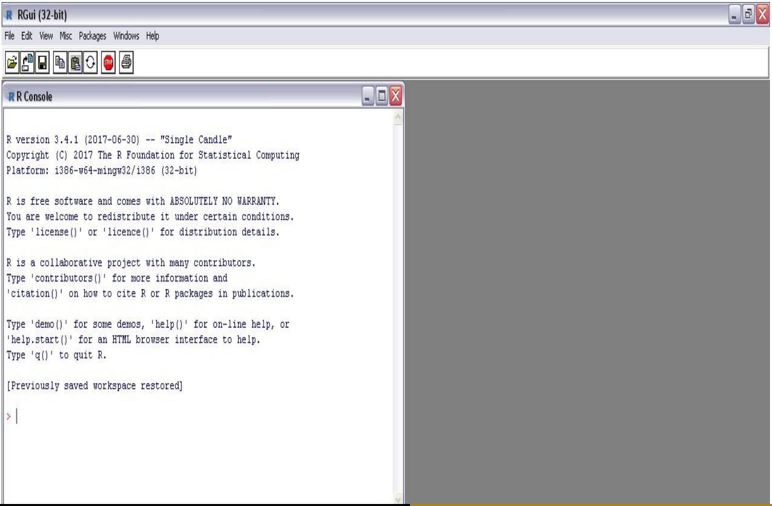
CRAN
Mirrors
What's new?
Task Views
Search

About R
R Homepage
The R Journal

Software
R Sources
R Binaries

Installing R

R can be started by double-clicking the desktop shortcut icon R or by going to Start$-$ $>$Program$-$ $>$ R.
The R startup window appears as below



```
R version 3.4.1 (2017-06-30) -- "Single Candle"
Copyright (C) 2017 The R Foundation for Statistical Computing
Platform: i386-w64-mingw32/i386 (32-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Previously saved workspace restored]

> |
```

R Console

The R Console is where computations are performed. After starting R, you will be looking at a console where you interact with R

- An expression is inputed into the console and the expression is evaluated. Depending on the expression, the system may respond by outputting results to the console or creating a graph in a new window. Then another expression is inputed and evaluated.
- An R session is this interaction between you and the system
- To get the last expression entered use the up arrow
- To get the value of the last evaluated expression type .Last.value
- Press Esc to stop evaluating the current expression

Screen Prompt

The screen prompt $>$ is an invitation to put R to work
We can use built-in function or do ordinary calculation
When continuation is expected the prompt changes to "$+$"
Note that the $+$ does not carry out the arithmetic plus.
Remark: If you made a mistake and want to get rid of the $+$ prompt and return to $>$ prompt, then either press the "Esc" key or use the Up arrow to edit and complete the last (incomplete) line.

R as a calculator

Enter a math expression and the result is displayed to the console

Binary Operators	$+$	$-$	$*$	$/$	$\wedge$	%%
Math Functions	abs	sqrt	log	exp	log10	factorial
Trig Functions	sin	cos	tan	asin	acos	atan
Rounding	round	ceiling	floor	trunc	signif	zapsmall

Some R functions

```
# write any comment after the hashtag
> 5+4
        [1] 9
> 6*7
        [1] 42
> 7^3
        [1] 343
> 20-15
        [1] 5
> 20%%7
        [1] 6
##%% is for modular arithmetic
> abs(5)
        [1] 5
> abs(-5)
        [1] 5
```

R functions

```
> factorial(4)
[1] 24
> sqrt(25)
[1] 5
> sort(c(2,5,1))
[1] 1 2 5
> rank(c(2,5,1))
[1] 2 3 1
> tail(c(10,20,30),1)
[1] 30
> tail(c(10,20,30),2)
[1] 20 30
> head(c(10,20,30),1)
[1] 10
> rev(c(10,20,30))
[1] 30 20 10
```

More Examples

```
> 30%%5
[1] 0
> 31%%5==1
[1] TRUE
> floor(5.7)
[1] 5
> ceiling(5.7)
[1] 6
> round(5.2)
[1] 5
> round(5.6)
[1] 6
> 3/0
[1] Inf
> is.finite(10)
[1] TRUE
> is.infinite(10)
[1] FALSE
> 0/0
```

More Examples

```
> LETTERS
 [1] "A" "B" "C" "D" "E" "F" "G" "H" "I" "J" "K" "L" "M"
[14] "N" "O" "P" "Q" "R" "S" "T" "U" "V" "W" "X" "Y" "Z"
> LETTERS[1:5]
[1] "A" "B" "C" "D" "E"
> letters
 [1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j" "k" "l" "m"
[14] "n" "o" "p" "q" "r" "s" "t" "u" "v" "w" "x" "y" "z"
> letters[1:5]
[1] "a" "b" "c" "d" "e"
```

Advance Math Functions

```
> optim()
[1]
> uniroot()
[1]
> integrate()
[1]
> deriv()
[1]
```

Syntex Note

- Functions can have optional arguments.
- Use help(read.table) for a complete description of the function and all the arguments.
- There is more than one way to do it.
- NA is the code for missing data.
- Use is.na() to see whether there is any missing data.

Variable Names

In R we need to assign a name to the variable

- Variable names in R are case sensitive so x is not the same as X
- Variable names should not begin with numbers or symbols and underscore (_)
- Variable names should not contain blank space: use *New.York* (not *New York*)
- Reserved words cannot be used as variables (TRUE, FALSE, NULL, if...)

R is CASE SENSITIVE

Objects and Functions

In R we normally create objects and then perform functions on those objects
For example, Assign an object a name "x" using either
$x < -$ object
$x =$ object
Call a function by
function name(list of arguments separated by commas) :
Example: mean, median, var, summary etc.

- Each function has a set of formal arguments some with default values
- A function call can include any subset of the complete argument list
- When specifying values for an argument use an $=$.

R is CASE SENSITIVE

Example

Suppose we want to calculate the mean of the scores 0, 5, 7, 9, 1, 2, 8. First we assign the vector of numbers a name say "x" and then call the function mean().

```
>  x = c(0,5,7,9,1,2,8)
> mean(x)
[1] 4.571429
> mean(X)
Error in mean(X) : object 'X' not found
> Mean(x)
Error: could not find function "Mean"
```

Example

Suppose we want to sort a vector y so that the numbers are descending. By default R will sort ascending so I need to change the formal argument decreasing to TRUE (the default value for decreasing is FALSE)

```
> y <- c(4,2,0,9,5,3,10)
> y
[1]  4  2  0  9  5  3 10
> sort(y)
[1]  0  2  3  4  5  9 10
> sort(y,decreasing=TRUE)
[1] 10  9  5  4  3  2  0
```

Character vectors

Scalars and vectors can be made up of strings of characters instead of numbers. All elements of a vector must be of the same type. For example

```
 colors <- c("red", "yellow", "blue")
 more.colors <- c(colors, "green","magenta", "cyan")
 mix <- c("red", "green", 1)
 > mix
[1] "red" "green""1" # 1 has been converted to the characte
```

There are two basic operations we might want to perform on character vectors. To take substrings, use substr() and to add additional information paste().

```
colors <- c("red", "yellow", "blue")
substr(colors, 1, 2)
paste(colors, "flower")
```

```
colors <- c("red", "yellow", "blue")
paste0(colors, "s") # paste0 reduces the space
```

Script Editor

The script editor is used for writing programs in R.

- To start a new script, click File > New Script
- The easiest way to run code is keyboard shortcuts
- To run part of a program, highlight the lines you want and hit Ctrl+R
- To run an entire program, select all the code then run, Ctrl+A then Ctrl+R
- To comment a line of code use a #

Comments are notes that help users understand portions of your code. They are also useful reminders to yourself of what was done and why it was done. Including meaningful comments in code is a major part of writing good programs, this semester we will practice commenting our programs.

Working Directory

When we load/save datasets, load source files or save graphs we will need to specify the file path. To avoid typing the path every time we can specify a working directory.
To set the working directory click File > Change dir... or type
>setwd(file path)

Package Documentations

Packages are collections of R functions, data, and compiled code in a well-defined format. The directory where packages are stored is called the library. R comes with a standard set of packages. Others are available for download and installation. Once installed, they have to be loaded into the session to be used.

.libPaths() # get library location
library() # see all packages installed
search() # see packages currently loaded

Adding Packages

One can expand the types of analyses by adding other packages. A complete list of contributed packages is available from CRAN.
on MS Windows

- Choose Install Packages from the Packages menu.
- Select a CRAN Mirror. (e.g. USA(IN))
- Select a package. (e.g. survival)
- Then use the library(package) function to load it for use. (e.g. library(survival))

Workspace

- The workspace is where all the objects you create during a session are located.
- When you close R you will be prompted to “Save workspace image?” If you say yes, the next time you open R from the same working directory the workspace will be restored. That is all of the objects you created during your last session will still be available. Also see save.image()
- Depending on what you are working on, it is usually wiser to write a script and then run the entire script with each new session. This way you know exactly what objects you have created; sometimes lingering objects can cause errors in programs. If a particular object is very important then save it to a file.

Managing the Workspace

- To list what variables you have created in the current session, ls()
- To see what libraries and dataframes are loaded, search()
- To see what libraries have been installed, library()
- To remove an object, rm(object names)
- To remove all objects, rm(list=ls())
- Another extremely useful feature in R is to type history(). This lists all the commands you have typed

RStudio

- RStudio makes R easier to use. It includes a code editor, debugging and visualization tools.
- The RStudio project currently provides most of the desired features for an IDE (Integrated Development Environment) in a novel way, making it easier and more productive to use R.
- What is IDE? R, like other programming languages, is extended (or developed) through user-written functions. An integrated development environment (IDE), such as RStudio, is designed to facilitate such work.

RStudio

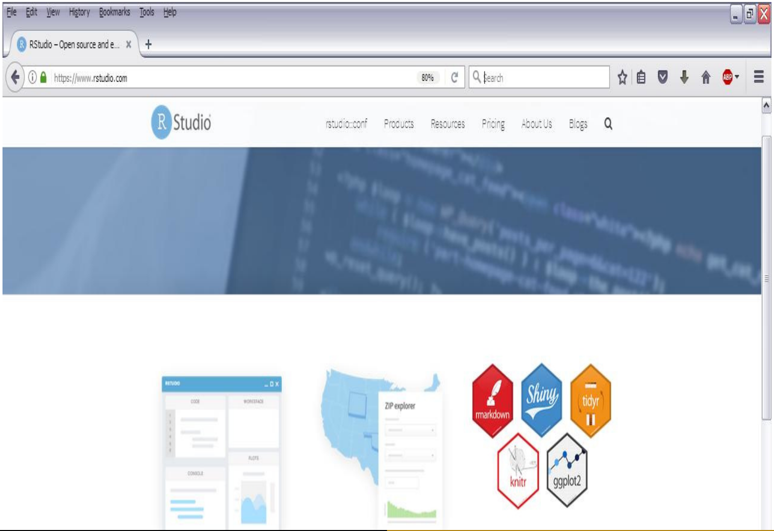
RStudio offers numerious helpful features:

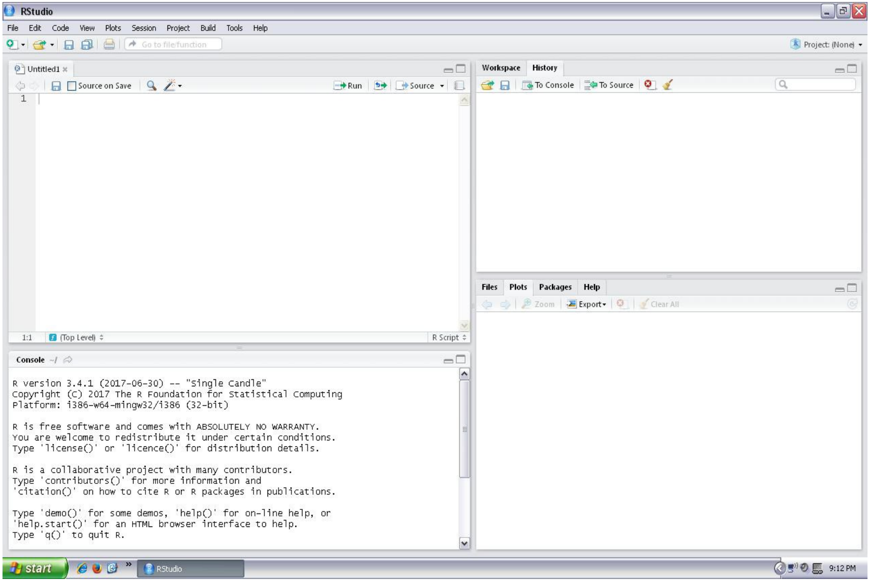
- A user-friendly interface
- Ability to write and save reusable scripts
- Easy access to all imported data and objects
- Exhaustive help on any object
- Code autocompletion
- The ability to create projects to organize and share your work
- Plot previewing
- Easy switching between terminal and console
- Operational history tracking

RStudio

Visit www.rstudio.com

The homepage appears as below.





```
R version 3.4.1 (2017-06-30) -- "Single Candle"
Copyright (C) 2017 The R Foundation for Statistical Computing
Platform: i386-w64-mingw32/i386 (32-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
```

RStudio-Components Layout

The RStudio interface consists of several main components sitting below a top-level toolbar and menu bar. Although this placement can be adjusted, the default layout utilizes four main panels or panes in the following positions:

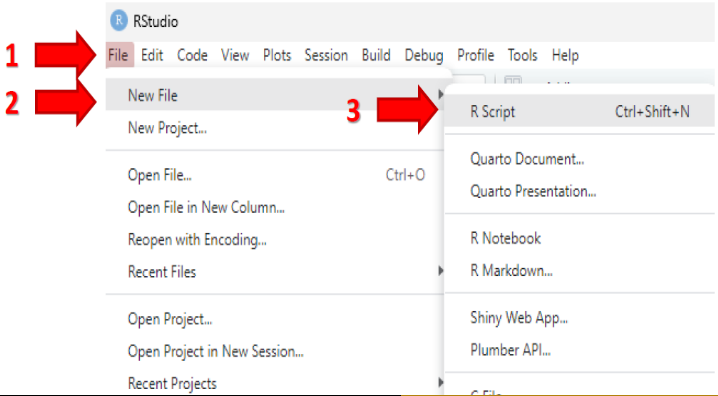
- In the upper left is a R script editor for editing files or viewing some data sets.
- In the lower left is a Console for interacting with an R process
- In the upper right is a notebook widget to hold a Workspace browser and History browser
- In the lower right is a notebook to hold tabs for interacting with the Files, Plots, Packages and Help system components

Editor and Console tab

- R codes are typed into the editor panel and submitted to the R console
- The console tab is for interacting with R
- It provides the details about the version of R
- Ctrl+L clears the R console.
- RStudio supports the automatic completion of the R code

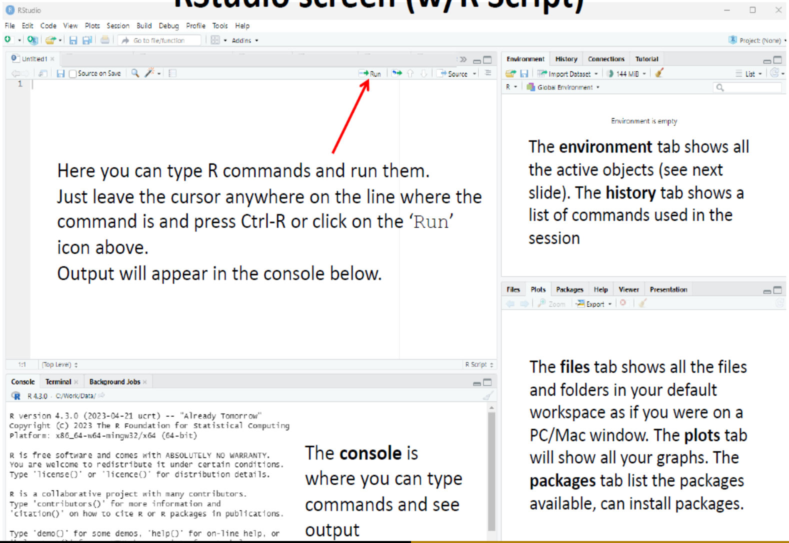
R Script

Open an R Script window



R Script

RStudio screen (w/R Script)



Here you can type R commands and run them. Just leave the cursor anywhere on the line where the command is and press Ctrl-R or click on the 'Run' icon above.
Output will appear in the console below.

The **environment** tab shows all the active objects (see next slide). The **history** tab shows a list of commands used in the session

The **files** tab shows all the files and folders in your default workspace as if you were on a PC/Mac window. The **plots** tab will show all your graphs. The **packages** tab list the packages available, can install packages.

The **console** is where you can type commands and see output

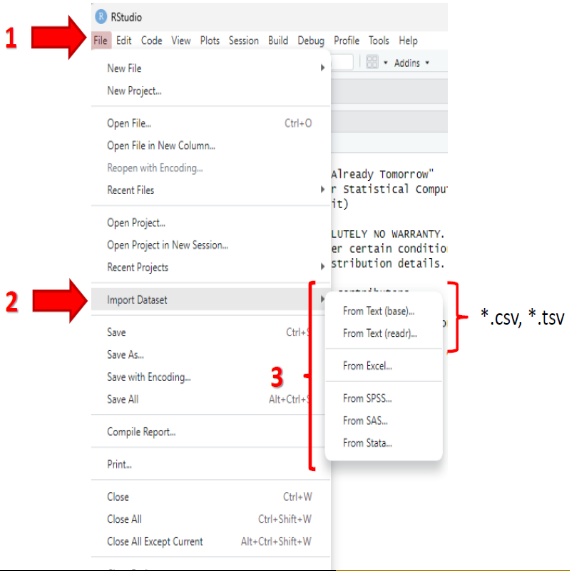
```
R version 4.3.0 (2023-04-21 ucrt) -- "Already Tomorrow"
Copyright (C) 2023 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
```

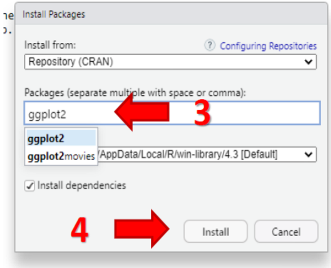
Importing Datsets



```
Already Tomorrow"
r Statistical Compu
it)

LUTELY NO WARRANTY.
er certain conditio
stribution details.
```

Installing packages


Install Packages

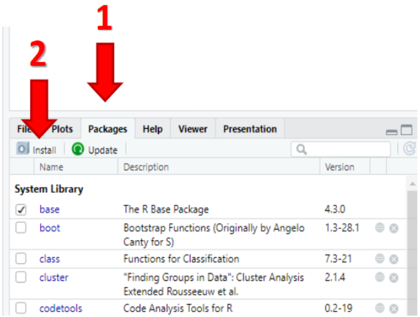
Install from: Repository (CRAN)

Packages (separate multiple with space or comma): ggplot2

ggplot2
ggplot2movies

/AppData/Local/R/win-library/4.3 [Default]

☑ Install dependencies



	Name	Description	Version
	System Library		
✓	base	The R Base Package	4.3.0
	boot	Bootstrap Functions (Originally by Angelo Canty for S)	1.3-28.1
	class	Functions for Classification	7.3-21
	cluster	"Finding Groups in Data": Cluster Analysis Extended Rousseeuw et al.	2.1.4
	codetools	Code Analysis Tools for R	0.2-19

Once a package is installed, no need to install it again until a new version of R is installed.
To activate a package type

```
library(name_of_package)
```

Environment and Workshop tab

- Many items listed under the Environment tab can be double-clicked to open them for viewing as a tab in the Script Editor
- The workspace is the session that is currently active in the R console
- The workspace tab stores any object, value, function or anything you create during your R session.
- It categorize data, values, functions.
- The history tab is the records of the script and command which has been executed.

Files, Plots, Packages, Help, Viewer tab

- The Plots tab will show the plots produced by commands submitted to the Console.
- One can cycle through the history of constructed plots with the arrows on the left side of the plot toolbar and plots can be saved to external files using the “Export” tab on the plot toolbar
- A list of all installed packages is seen by selecting the Packages tab
- New packages can also be installed through this tab
- Help for each package can be obtained by clicking on the name of the package. The help will then appear in the Help tab.

R Markdown

Markdown is a simple way to create text that rendered into an HTML webpage. In fact, using rmarkdown we can convert it to many different output types. The top three that you might be most likely to use are HTML, PDF, and Microsoft Word. In a nutshell, R Markdown stands on the shoulders of `knitr` and `Pandoc`. The former executes the computer code embedded in Markdown, and converts R Markdown to Markdown. The latter renders Markdown to the output format you want (such as PDF, HTML, Word, and so on).

How to convert to HTML, PDF, or Word?

a. Controlling the “knit” button

b. changing the YAML option

   title: "Document"
   output: html_document

   title: "Document"
   output: pdf_document

   title: "Document"
   output: word_document

`rmarkdown` provides an environment where you can write your complete analysis, and marries your text, and code together into a rich document. You write your code as code chunks, put your text around that, and then you have a document you can reproduce.

In order to ensure that RMarkdown works perfectly the following packages should be installed.

```
install.packages("rmarkdown")
install.packages("knitr")
```

R Markdown

The RStudio layout has the following features:

- On the upper left, the Rmarkdown script
- On the lower left, the R console
- On the lower right, the view for files, plots, packages, help, and viewer.
- On the upper right, the environment / history pane

The anatomy of an rmarkdown document

This is an rmarkdown document (demo). It has three parts:

- Metadata (YAML, Ain't Markup Language)
- Text (markdown formatting)
- Code (code formatting)

Metadata Section

The metadata of the document tells you how it is formed - what the title is, what date to put, and other control information.

```
---
title: "Example document"
author: "Gokarna Aryal"
output: html_document
---
```

It starts and ends with three dashes ---, and has fields like the following: title, author, and output.

Help

- help(fn) for help on fn
- help.search(”topic”) for help pages related to “topic”
- RSiteSearch(”missing”) for help pages related to “missing”
- apropos(”tab”) for functions whose names contain “tab”
- Search function on the http://www.r-project.org web site.
- Rseek website http://www.rseek.org

Good Resources to learn R

- Try R: http://tryr.codeschool.com/
- DataCamp: https://www.datacamp.com/
- Shiny App: https://www.showmeshiny.com/
- R bloggers: https://www.r-bloggers.com/how-to-learn-r-2/

Getting Started with R

THANK YOU