

Chapter 0

Introduction to R

This guidebook will serve as an introduction to R and RStudio. We begin by showing how to download and install the necessary programs. The guide is written so that it follows the organization of the text. The following online tutorials/readings are very useful if you want to read additional R guides:

- <https://www.rstudio.com/resources/cheatsheets/>
- <https://cran.r-project.org/doc/contrib/Owen-TheRGuide.pdf>
- <https://cran.r-project.org/web/packages/IPSUR/vignettes/IPSUR.pdf>

Installing R

To download and install R to your computer, go to the following R CRAN website:
<https://cran.r-project.org/mirrors.html>

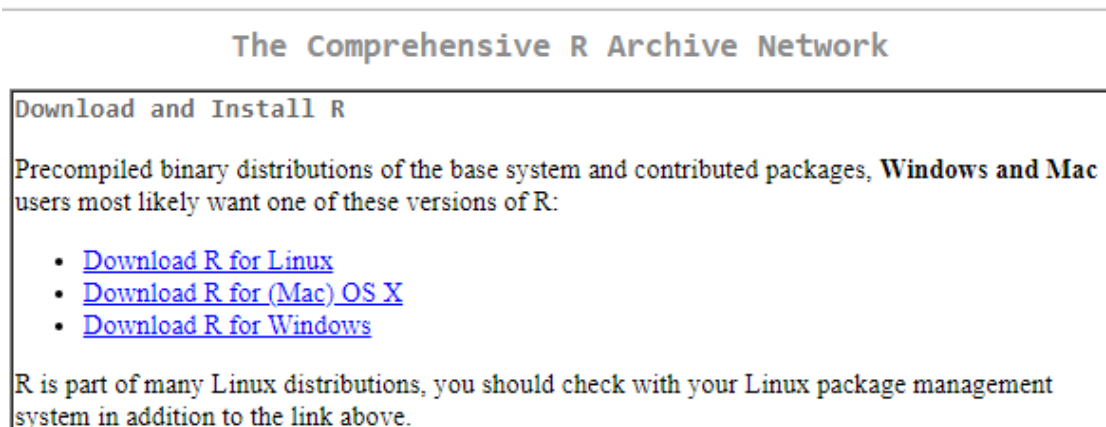
This webpage includes a list of R mirrors. See Figure 1. Each R mirror listed is a separate download for R. These are called mirrors, because each link is a replica of the rest. The reason there are so many mirrors is to allow faster download speeds and to ensure that network traffic is well maintained. All of these downloads are exactly the same, so scroll down the page. Click on the mirror that is in closest proximity to your location to ensure faster download speeds. If there is confusion about which mirror to choose, it does not matter. The only negative aspect of choosing the wrong mirror is slower download speeds.

Figure 1

https://cran.cnr.berkeley.edu/	University of California, Berkeley, CA
http://cran.cnr.berkeley.edu/	University of California, Berkeley, CA
http://cran.stat.ucla.edu/	University of California, Los Angeles, CA
https://mirror.las.iastate.edu/CRAN/	Iowa State University, Ames, IA
http://mirror.las.iastate.edu/CRAN/	Iowa State University, Ames, IA
https://ftp.ussg.iu.edu/CRAN/	Indiana University
http://ftp.ussg.iu.edu/CRAN/	Indiana University
https://rweb.crmda.ku.edu/cran/	University of Kansas, Lawrence, KS
http://rweb.crmda.ku.edu/cran/	University of Kansas, Lawrence, KS
https://cran.mtu.edu/	Michigan Technological University, Houghton, MI
http://cran.mtu.edu/	Michigan Technological University, Houghton, MI
http://cran.wustl.edu/	Washington University, St. Louis, MO
http://archive.linux.duke.edu/cran/	Duke University, Durham, NC

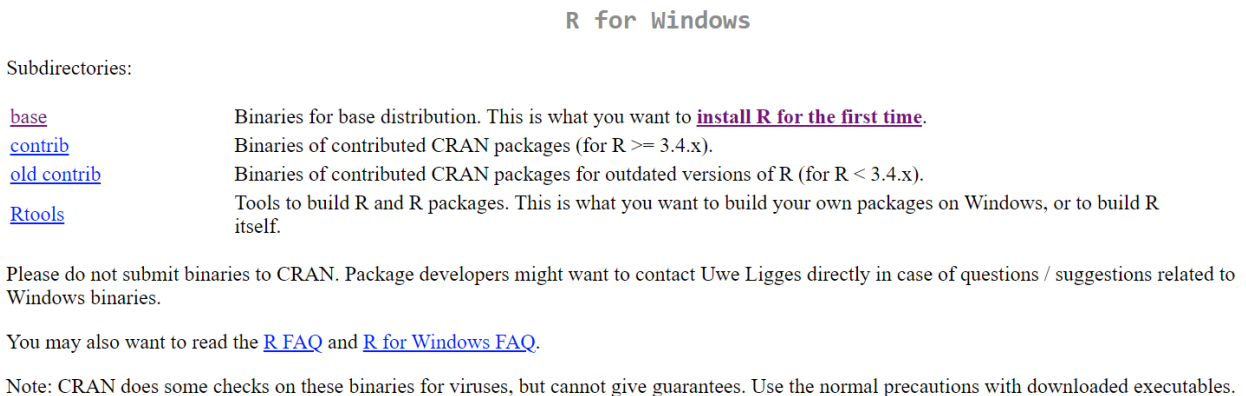
After clicking on the appropriate R mirror, a webpage like Figure 2 will appear. If you have a Windows computer, click “Download R for Windows.” If you have a Mac computer, click “Download R for (Mac) OS X.”

Figure 2



From there, Windows computer users will see Figure 3.

Figure 3



Click on the base subdirectory. This allows a download of all of the base packages for R.

Mac computer users will see Figure 4.

Figure 4

R for macOS

This directory contains binaries for the base distribution and of R and packages to run on macOS. R and package binaries for R versions older than 4.0.0 are only available from the [CRAN archive](https://cran-archive.r-project.org) so users of such versions should adjust the CRAN mirror setting (<https://cran-archive.r-project.org>) accordingly.

Note: Although we take precautions when assembling binaries, please use the normal precautions with downloaded executables.

R 4.3.1 "Beagle Scouts" released on 2023/06/16

Please check the integrity of the downloaded package by checking the signature:

```
pkgutil --check-signature R-4.3.1.pkg
```

in the *Terminal* application. If Apple tools are not available you can check the SHA1 checksum of the downloaded image:

```
openssl sha1 R-4.3.1.pkg
```

Latest release:

For Apple silicon (M1/M2) Macs:

[R-4.3.1-arm64.pkg](#)

SHA1-hash: 14c018ff54f75bb37c1d96b33207343b83e9345
(ca. 90MB, notarized and signed)

For older Intel Macs:

[R-4.3.1-x86_64.pkg](#)

SHA1-hash: 1af8f055a601d5de5dfefdb3956ecc8f745c2401
(ca. 92MB, notarized and signed)

R 4.3.1 binary for macOS 11 (**Big Sur**) and higher, signed and notarized packages.

Contains R 4.3.1 framework, R.app GUI 1.79, Tcl/Tk 8.6.12 X11 libraries and Texinfo 6.8. The latter two components are optional and can be omitted when choosing "custom install", they are only needed if you want to use the `tc1tk` R package or build package documentation from sources.

macOS Ventura users: there is a known bug in Ventura preventing installations from some locations without a prompt. If the installation fails, move the downloaded file away from the *Downloads* folder (e.g., to your home or Desktop)

Note: the use of X11 (including `tc1tk`) requires [XQuartz](#) (version 2.8.5 or later). Always re-install XQuartz when upgrading your macOS to a new major version.

This release uses Xcode 14.2/14.3 and GNU Fortran 12.2. If you wish to compile R packages which contain Fortran code, you may need to download the corresponding GNU Fortran compiler from <https://mac.R-project.org/tools>. Any external libraries and tools are expected to live in `/opt/R/arm64` (Apple silicon) or `/opt/R/x86_64` (Intel).

Click on the latest release package. Because R constantly updates and improves, new updates are released frequently. In Figure 4, the most recent update is "R-4.3.1 pkg." Your most recent update will likely be different! After clicking on the most recent update "R-3.X.X pkg," open the downloaded file and complete the prompted download instructions. And that is it. R is now officially downloaded on your computer! If issues arise, please refer to one of the guides linked at the beginning of this document.

Installing RStudio

The next step will be installing RStudio. RStudio is an open source license that is designed to increase productivity with R. It includes a code editor, as well as debugging and visualization tools. RStudio is not necessary to code in R, but it is highly recommended. The interface is much simpler than R and will make life easier. To download RStudio, go to the following website: <https://www.rstudio.com/products/RStudio/>

Click on the Download RStudio Desktop Open Source License and Figure 5 will appear.

Figure 5

Installers for Supported Platforms

Installers	Size	Date	MD5
RStudio 1.1.456 - Windows Vista/7/8/10	85.8 MB	2018-07-19	24ca3fe0dad8187aabd4bfb9dc2b5ad
RStudio 1.1.456 - Mac OS X 10.6+ (64-bit)	74.5 MB	2018-07-19	4fc4f4f70845b142bf96dc1a5b1dc556
RStudio 1.1.456 - Ubuntu 12.04-15.10/Debian 8 (32-bit)	89.3 MB	2018-07-19	3493f9d5839e3a3d697f40b7bb1ce961
RStudio 1.1.456 - Ubuntu 12.04-15.10/Debian 8 (64-bit)	97.4 MB	2018-07-19	863ae806120358fa0146e4d14cd75be4
RStudio 1.1.456 - Ubuntu 16.04+/Debian 9+ (64-bit)	64.9 MB	2018-07-19	d96e63548c2add890bac633bdb883f32
RStudio 1.1.456 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (32-bit)	88.1 MB	2018-07-19	1df56c7cd80e2634f8a9fdd11ca1fb2d
RStudio 1.1.456 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (64-bit)	90.6 MB	2018-07-19	5e77094a88fdbdddb0d35708752462

Windows computer users should click on the first download (Windows vista/7/8/10). Mac computer users should click on the second download (Mac OS C 10.6+ (64-bit)). After clicking on the download links, follow the prompted download instructions.

RStudio Cloud Rather than downloading the desktop version of RStudio, you might consider the RStudio Cloud at rstudio.cloud.

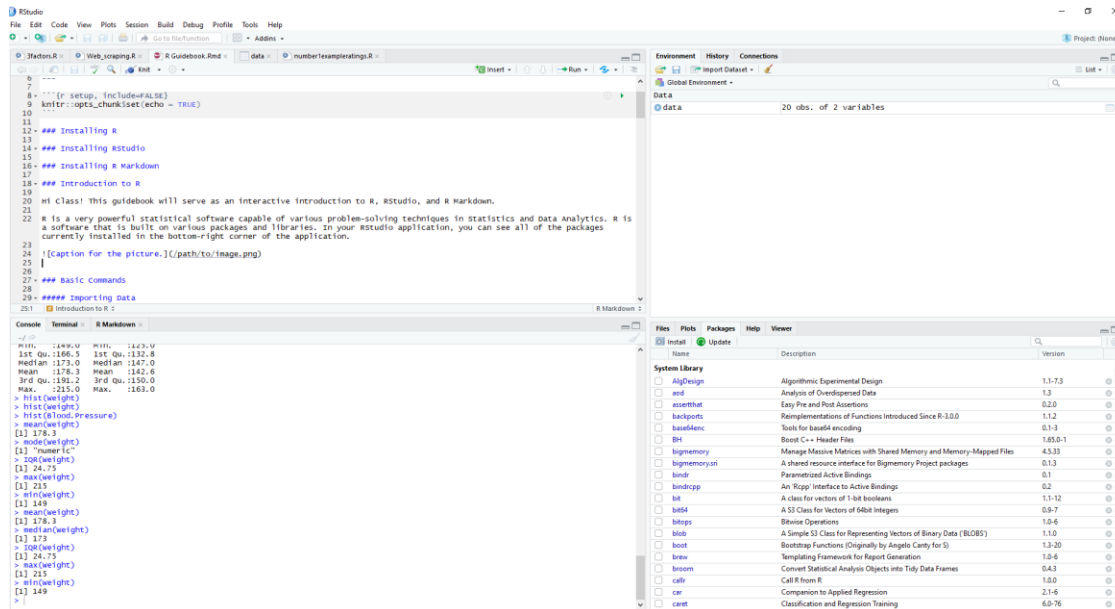
Introduction to R and RStudio

R is a very powerful statistical language capable of various problem-solving techniques in Probability, Statistics, and Data Analytics. For most Data Scientists, R is one of the two or three most commonly used programming languages.

R is a programming language built on various packages and libraries, ranging from simpler visualization/ graphing packages to more complex machine learning libraries. The R community is constantly developing new libraries to be used for data analysis.

RStudio was created to be a simple interface to perform statistical analysis using R. Figure 6 is a screenshot of the RStudio interface. The top-left corner is where the R commands are written (called R script). The user can type R commands in this area and save it without having to run code line by line. The bottom-left corner is known as the console. The user can write R commands in this area to try it line by line instead of writing an entire R program in the script window. The top-right corner is the data window. RStudio will display all imported datasets in this corner. In Figure 6, note the term “data” in the data window. This is a dataset that was imported into RStudio. As seen in Figure 6, this dataset has 20 observations of 2 variables. In order to view this dataset, the user simply clicks on the word “data” in the data environment area. A benefit of working in RStudio is the ability to review data sets. Finally, in the bottom-right corner, the user can view and search for different libraries and packages. This is also where plots/visualizations are displayed. All of these capabilities play a role in the user-friendly nature of RStudio.

Figure 6



R is valuable, because it is one of the few statistical softwares that can handle large amounts of data and process it quickly. Data science is a rapidly growing professional field, and data science is needed in nearly every industry. Annual demand for data scientists is expected to grow by nearly 364,000 listings, 28% growth a year.

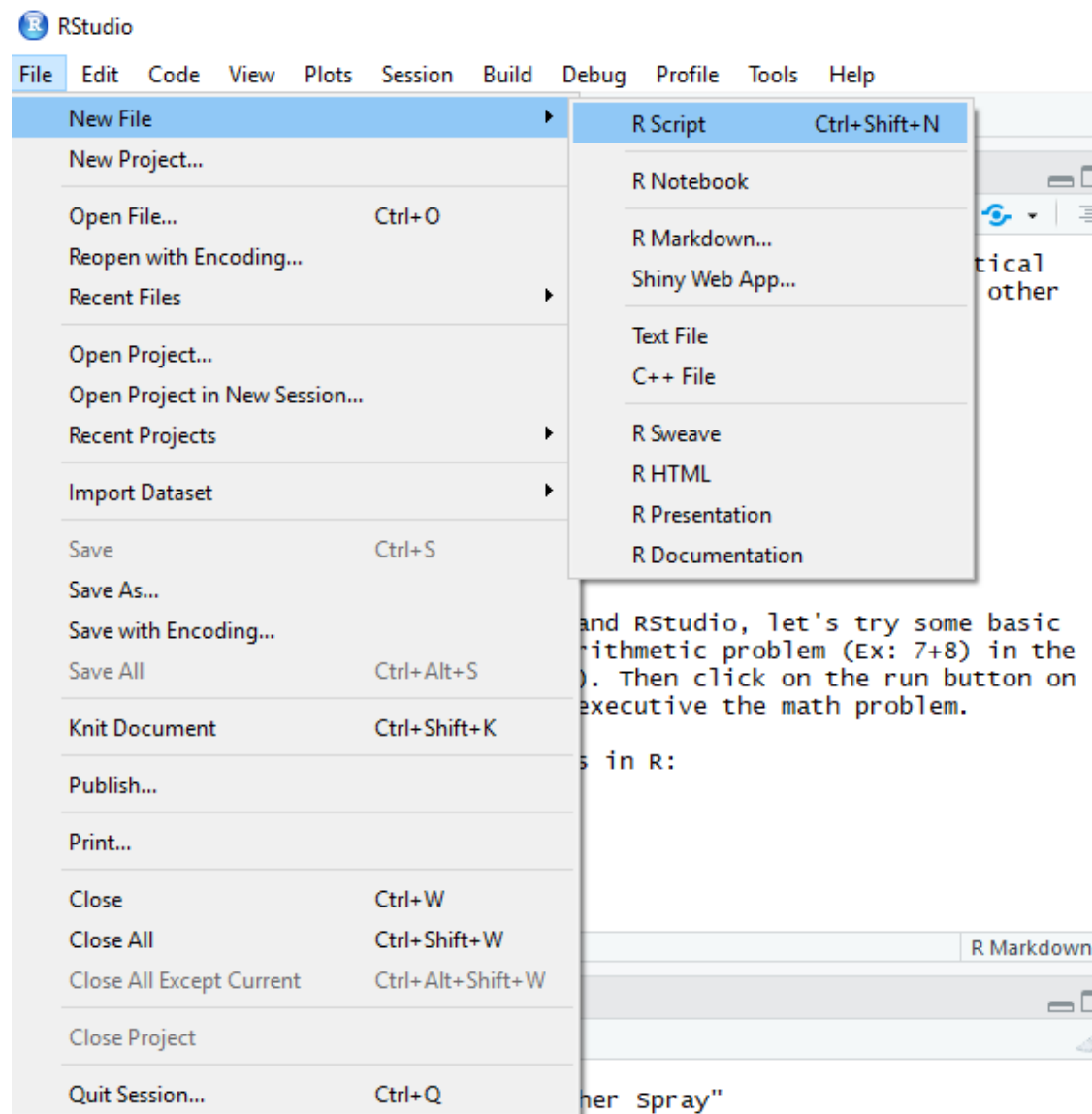
R can be used for a number of analyses, including mathematical calculations, statistical analysis, data visualization, or other advanced data science techniques.

Basic R Operations

Start a New Project

Now that you have installed RStudio, let's start a new project! Open RStudio on your computer or go to rstudio.cloud and select "New Project". Using the top menu, click on File -> New File -> R Script as shown in Figure 7. Save the untitled workbook in the desired folder on your computer and you are ready to get started.

Figure 7



The other way to open a new workbook is to click on the green plus sign right below “File”. Then click on R Script and you have opened a new workbook just like the previous method.

Arithmetic

Now that you know how to open a workbook in RStudio, let’s try some basic commands in R. Type a simple arithmetic expression in the R text editor (top-left corner). Then click on the run button on the toolbar. R will do the arithmetic.

Example: Type the following problems in R:

- $7 + 8$
- $999 - 876$
- $7 * 4 - 3$

$8/3+9$



The image shows a screenshot of an R IDE window titled "Untitled2* x". The window has a menu bar with "File", "Edit", "Code", "View", "Plots", "Session", "Build", "Debug", "Profile", "Tools", and "Help". Below the menu bar is a toolbar with icons for file operations and a search bar. The main editor area contains the following code:

```
1 7+8
2 999-876
3 7*4-3
4 8/3+9|
```

Overlaid on the editor is red text: "Type the expressions here. Highlight all and click Run." The status bar at the bottom of the editor shows "4:6 (Top Level) ↕" and "R Script ↕".

Below the editor is a console window titled "Console /cloud/project/ ↗". It shows the following output:

```
> 7+8
[1] 15
> 999-876
[1] 123
> 7*4-3
[1] 25
> 8/3+9
[1] 11.66667
> |
```

Overlaid on the console is red text: "The results appear down here."

Clearly, R can handle simple math and knows Order of Operations. What about more advanced calculations?

More Advanced Calculations

R knows the value of pi and how to do more advanced calculations like squareroot, log, and exponentials. Note that the `log()` command in R is the natural logarithm and the `exp()` command is the number e raised to a power. In RStudio, type the following directly in the console (bottom-left):

```
> pi
[1] 3.141593
> log(8)
[1] 2.079442
> sqrt(9)
[1] 3
> exp(3)
[1] 20.08554
> 6^2
[1] 36
> |
```

Variables

If you want, you can also create variables in R. For example, the command `x <- 3` will store the value 3 in the variable `x`. Variable names do not need to be a single character. For example, `Weight <- 130` will store the value 130 in the variable `Weight`. See below. Notice that R will not assume multiplication in the expression $4x - 12$.

```
> x <- 3
> x
[1] 3
> Weight <- 130
> Weight
[1] 130
> 4x - 12
Error: unexpected symbol in "4x"
> 4*x-12
[1] 0
> |
```


Importing Data

Data should always be imported into R. Most data sets are available as csv (comma separated values) files that are used with most spreadsheets (such as Excel or GoogleSheets). Data is going to either be stored on the Internet or on your computer. For data on the Internet, all you need is the URL that identifies the data. The data for the Sullivan texts are housed at GitHub. The URL for most data sets is

https://sullystats.github.io/Stats7/Data/Chx/x_s_p.csv

where

- x is the chapter number
- s is the section number (r is use for the chapter review problems and ct is used for the chapter test problems).
- p is the problem number

For example, to retrieve the data set for Problem 31 in Section 4.1, use the URL

https://sullystats.github.io/Stats7e/Data/Ch4/4_1_31.csv

To retrieve the data set for Problem 10 in the Chapter 2 Review, use the URL

https://sullystats.github.io/Stats7e/Data/Ch2/2_r_10.csv

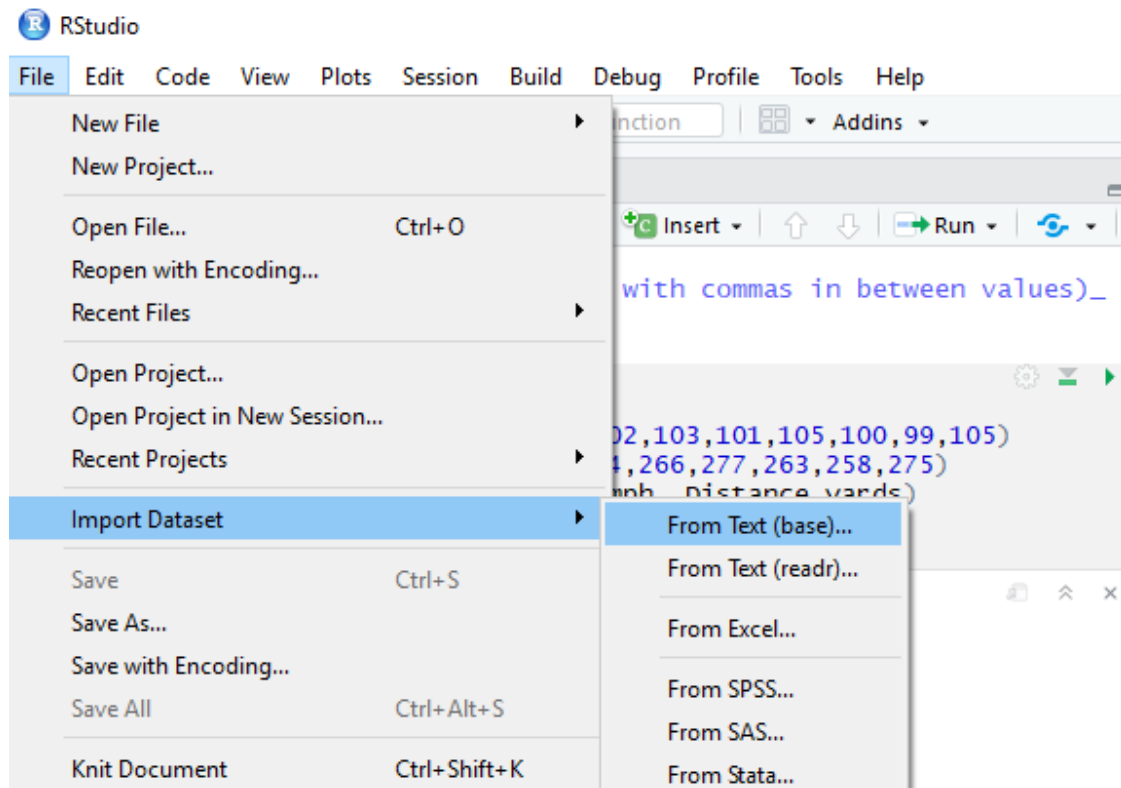
To retrieve data online from the text, go to www.pearsonhighered.com/sullivanstats. A new window will open. The first bullet item displayed is Data Sets. Click on the TXT.zip link. You will be prompted to save the zip file to your computer's hard-drive. Create a folder for this class and save the zip file. Once the file downloads (it is a large file so this may take a few minutes), click on the file in your system tray. An "unzipped" folder will appear in the same location as the zip file. Feel free to rename the folder (something like "StatsData"). Within this folder, there is a separate folder that contains data files for each chapter of the text. Complete the same process for the CSV.zip link.

Importing TXT Files

Now that you have all the data downloaded, you are ready to import a dataset.

For .txt files, from the toolbar at the top of RStudio, click File -> Import Dataset -> From Text(base)... See Figure 8.

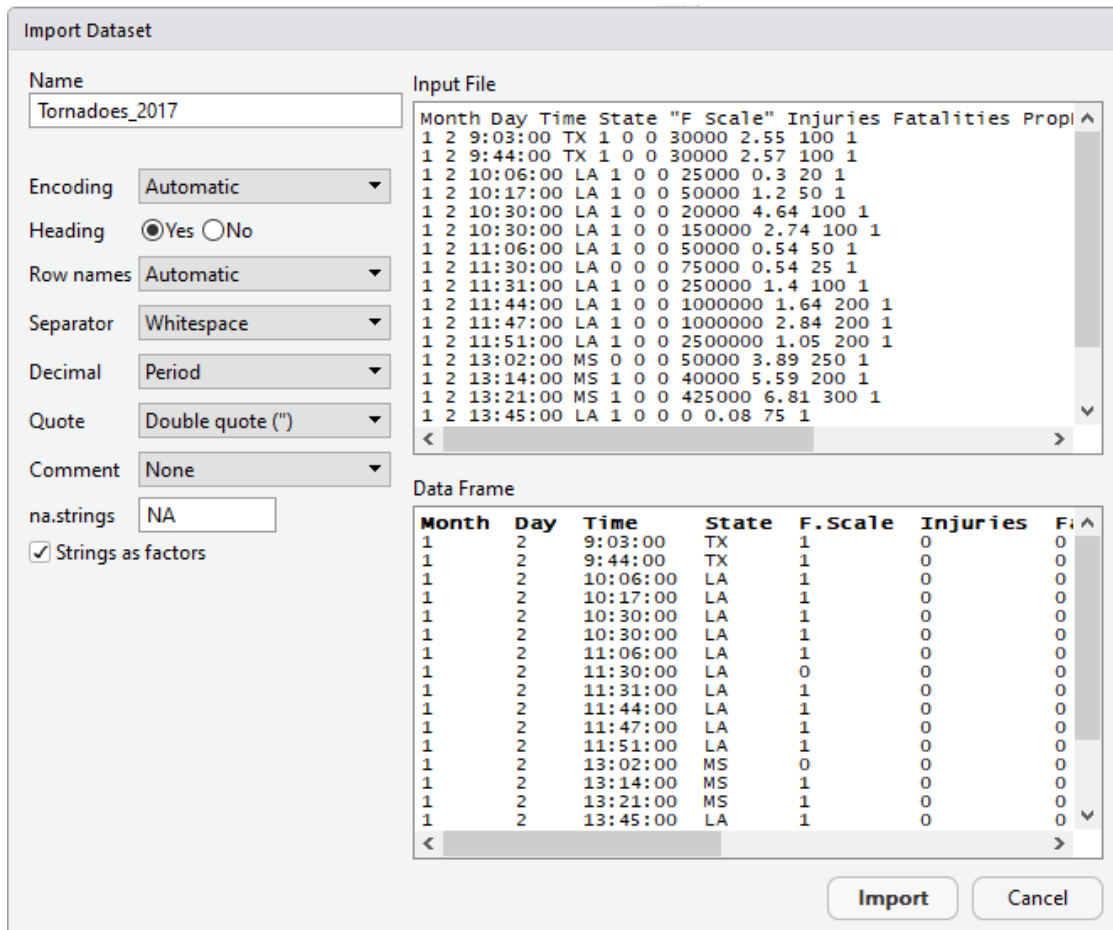
Figure 8



A popup will appear. Browse to the desired file and Click Open.

Then another popup will appear (See Figure 9). Click Import in the popup box and your txt file will appear both as a tab in you text editor area (top-left corner) and as a dataset stored in RStudio's global environment (top-right corner).

Figure 9

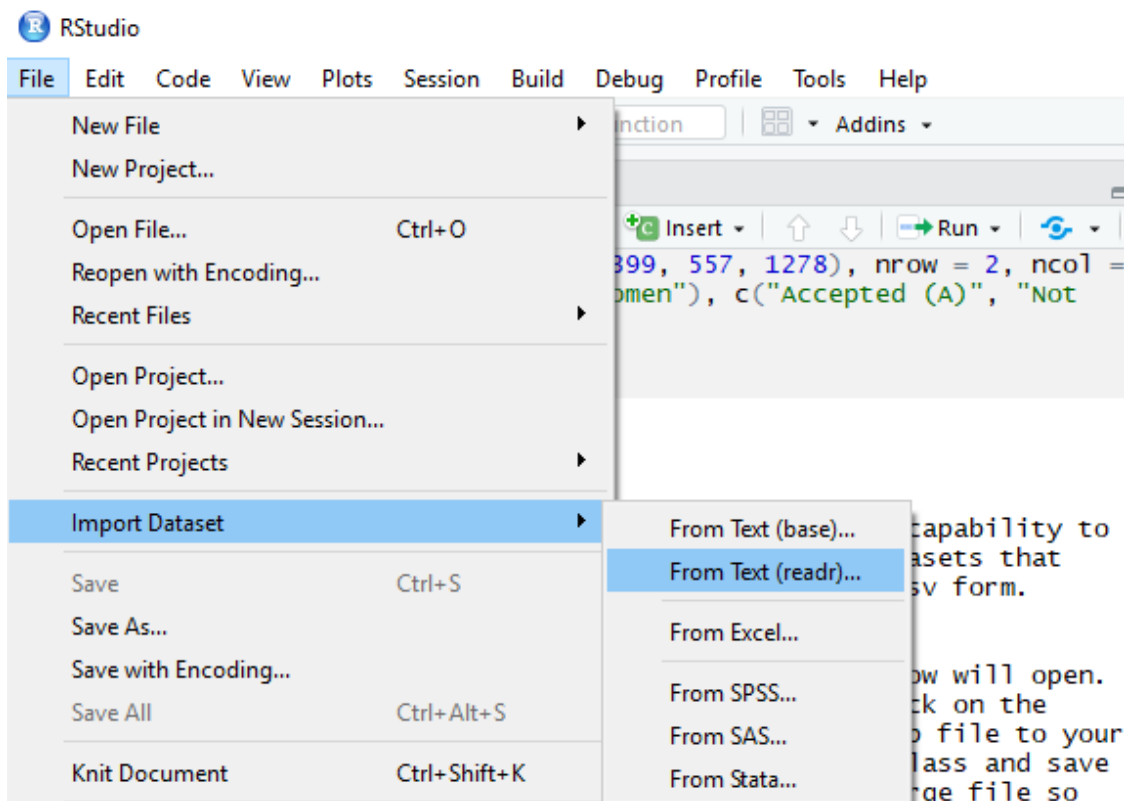


Importing CSV Files

Importing csv files follows a similar but slightly different process from txt files.

In the top toolbar, click File -> Import Dataset -> From Text(readr).... See Figure 10.

Figure 10



A popup will appear. Click Browse and find the desired dataset you want to import. Then click Import. The imported file will appear as a tab in the text editor area of RStudio and as a dataset in the Global Environment.

Creating New Data Files

Do not bother with entering data into R. Rather, use a spreadsheet such as Excel or GoogleSheets to create a csv file. Save it to GitHub or your computer, and then load the data set.

Final Note All of the data files we are working with are technically referred to as *data frames*. If you every want to see the contents of a data frame, use the following command:

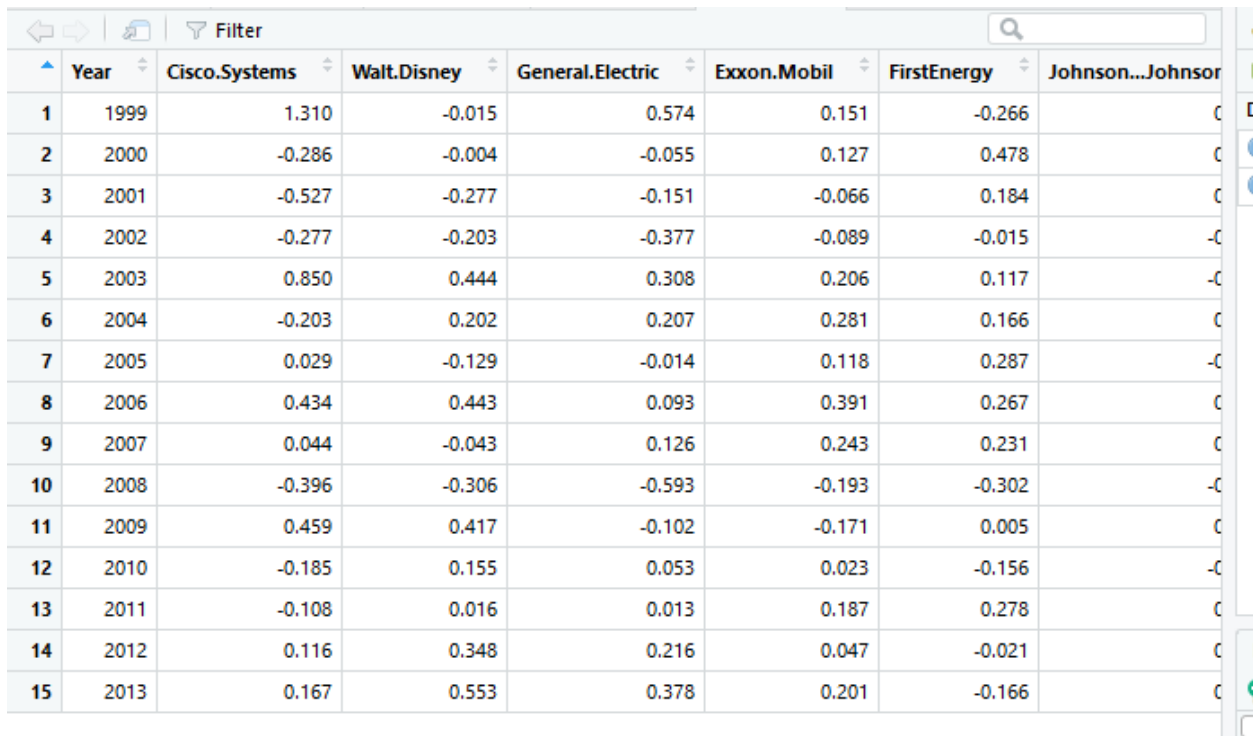
```
View(name)
```

For example, suppose we import the data frame “4_1_39.txt” and name it Stocks. We can view the contents of the data frame as follows:

```
> Stocks <- read.delim("C:/Users/Sullivan Texts Inc/iCloudDrive/Stats/Stats 5  
e/sst5e_data_txt/ch04/4_1_39.txt")  
> View(Stocks)
```

The data frame is shown in Figure 11.

Figure 11



	Year	Cisco.Systems	Walt.Disney	General.Electric	Exxon.Mobil	FirstEnergy	Johnson...Johnson
1	1999	1.310	-0.015	0.574	0.151	-0.266	
2	2000	-0.286	-0.004	-0.055	0.127	0.478	
3	2001	-0.527	-0.277	-0.151	-0.066	0.184	
4	2002	-0.277	-0.203	-0.377	-0.089	-0.015	
5	2003	0.850	0.444	0.308	0.206	0.117	
6	2004	-0.203	0.202	0.207	0.281	0.166	
7	2005	0.029	-0.129	-0.014	0.118	0.287	
8	2006	0.434	0.443	0.093	0.391	0.267	
9	2007	0.044	-0.043	0.126	0.243	0.231	
10	2008	-0.396	-0.306	-0.593	-0.193	-0.302	
11	2009	0.459	0.417	-0.102	-0.171	0.005	
12	2010	-0.185	0.155	0.053	0.023	-0.156	
13	2011	-0.108	0.016	0.013	0.187	0.278	
14	2012	0.116	0.348	0.216	0.047	-0.021	
15	2013	0.167	0.553	0.378	0.201	-0.166	

Packages in R

The R language is comprised of a variety of packages that have been written by members of the R community. These core packages (called **base R**) represent a collection of *functions* that allow us to perform a variety of statistical procedures. In addition, there are additional packages available to install. One such package is *MOSAIC*, which was written by Nicholas J. Horton, Randall Pruim, and Daniel T. Kaplan.

To use any package in R (beyond those in base R), it must be installed. This only needs to be done once (like installing any software on your computer). To install *MOSAIC* use the following command:

```
> install.packages("mosaic")
```

Please be patient when installing because R must communicate and upload to your computer. Once the package is installed, you will need to load it into your session. This must be done for each R session. Think of this as opening your software on your computer. The following command will load *MOSAIC*.

```
> library(mosaic)
```

Now, we are ready to work with R.