Introduction to R Studio

Ryan Godwin

1. Installation

1.1 Install R

For Windows: Download and run the .exe file from the following link: R for Windows

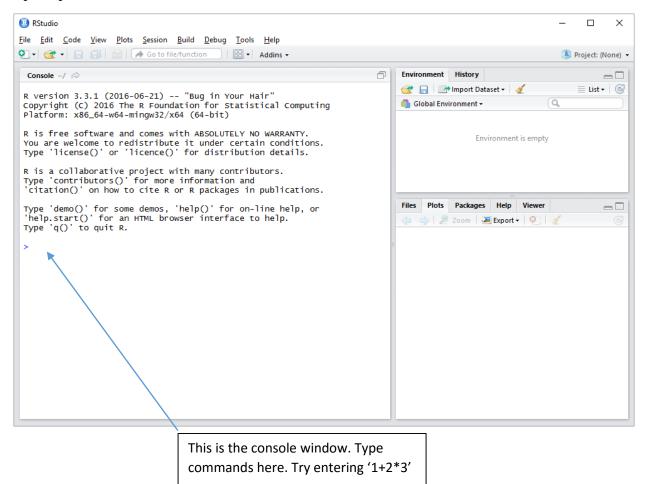
For Mac: <u>R for Mac</u>

1.2 Install R Studio

On the following link <u>Download R Studio</u> choose the download for RStudio Desktop (Free License)

2. Basic Tutorial

Open up RStudio. It should look like this:



8 RStudio	– 🗆 X
<u>File Edit Code View Plots Session Build Debug Tools H</u> elp	
🝳 🔹 🚽 🔒 📄 🗁 🕼 Go to file/function 🛛 🖾 🗸 Addins 🗸	🔳 Project: (None) 🝷
Console ~/ 🔗	Environment History
R version 3.3.1 (2016-06-21) "Bug in Your Hair" Copyright (C) 2016 The R Foundation for Statistical Computing Platform: x86_64-w64-mingw32/x64 (64-bit)	Import Dataset •
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.	Environment is empty
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.	Files Plots Packages Help Viewer Image: Comparison of the state
> 1+2*3 [1] 7 >	
Output is displayed in the cons	sole window.

To keep track of your commands, you can use a 'script' file. Go to 'File \rightarrow New File \rightarrow R Script'. Type a command in the 'R Script' window. Highlight the command. Click the 'Run' button. Finally, save your script by selecting 'File \rightarrow Save'.

File Edit Code View Plots Session Build Debug Tools Help <td <t<="" <td="" th=""><th>ne) 🔻</th></td>	<th>ne) 🔻</th>	ne) 🔻
	ne) 🔻	
Untitled1* x Environment History		
🗇 🗇 🗊 🔚 🗌 Source on Save 9. 🦉 + 🗐 🖶 Run 🤥 🕞 Source 🔹 🗟 🚭 🔒 📷 Import Dataset + 💆 📃 List +	C	
1 exp(1)		
Environment is empty		
Files Plots Packages Help Viewer		
→ → Zoom Zoom Zeport · ♀	æ	
	9	
1:1 (Top Level) ≎ R Script ≎		
Console / \approx 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 'bela for the formation balls		
'help.start()' for an HTML browser interface to help. Type 'q()' to quit R.		
> 1+2*3 [1] 7		
$> \exp(1)$		
[1] 2.718282		
1. Type your command here. Highlight it.		
2. Click the 'Run' button, or press 3. Output displayed here.		
'CTRL + R' on your keyboard.		

Next, download some data from the course website. It is in '.csv' format (comma separated values), one of the simplest and most common form of data file. To get the data into R Studio, enter the following command:

mydata <- read.csv("https://home.cc.umanitoba.ca/~godwinrt/3040/data/introdata.csv")</pre>

You have now created a data frame called 'mydata' (you could have named it anything). To view the data, you can click the object in the environment window.

								_	~
RStudio							_		×
File Edit Code View Plots Session Build Debug Tools									
Q. ▼ 🚰 ▼ 🔒 🔝 🚔 🌈 Go to file/function 🔡 ▼	Addins -						Pro Pro	oject: (N	lone) •
Untitled1* × mydata ×	-0	Enviro	onment	History					_
🗇 🔿 🗐 🙇 🛛 🖓 Filter	Q,	🚭 I	. 🗷	Himport Da	ataset 🕶 🖂	1		List 🕶	C
x		📕 GI	obal En	vironment -		9			
1 1 1	^	Data							
2 1 2		O my o	data	104	obs. o	f 2 vari	ables		
3 1 3		4							
4 1 4									
5 1 5									
6 2 4					=	1			
7 3 3		Files	Plots	_		Viewer		6	
8 4 5 9 4 4			$> \ge$	Zoom	🚈 Export 🖲	• 🤤 🍕			
9 4 4 10 4 3		н							
	~								
Showing 1 to 11 of 104 entries									
Console ~/ 🖘	-0								
<pre>> mydata <- read.csv("https://home.cc.umanitob trodata.csv")</pre>	a.ca/~godwinrt/3040/data/ir	1							
> View(mydata)									
>									
	F		1						
		1. CI	ick ł	nere to	view	the dat	ta.		
1	L	1.01						J	
2. Click have to go back to your corint	7								
2. Click here to go back to your script.									

To extract a variable from the data frame (the *x* variable for example), use:

mydata\$x

Alternatively, you can load all of the variables into memory using

attach(mydata)

and refer to each variable directly.

Try the following commands:

summary(mydata)	mean(x)	var(y)	cor(x,y)
?cor	myvar <- c(1,3)	myvar	plot(x,y)