R for Machine Learning
Introduction & Overview

CE 5331 Machine Learning for Civil Engineers

Venki Uddameri, Ph.D. , P.E.

Focus on the logic
Identify what steps are necessary
Then worry about the syntax

R Basics

Mastery in R means how comfortable are you searching and finding necessary syntax

The lecture here is not intended to give you formal training in R

R is too vast to be taught extensively along with machine learning concepts

I will expose you to syntax necessary for performing ML tasks of the course (Good starting point for your explorations)

I urge you to practice offline and become familiar with the syntax

Focus on the algorithm and see what steps are necessary

Then figure out what syntax is necessary

Google is your friend

R – What is it?

- R is a general purpose software for programming, statistical computing and visualization
  - Very similar in functionality to MATLAB
  - Free open-source software
  - Written mostly in Fortran, C, C++ and R
- R was created in **R**oss Ihaka and **R**obert Gentleman in early 1990s
  - First version made public in 1993
  - One school of thought says R is named because it is the first letter in the names of its creators
- Currently maintained by R Core Team
- Based on a software called S developed in Bell Labs by John Chambers in 1976
  - Other school of thought says R was named as a play on S

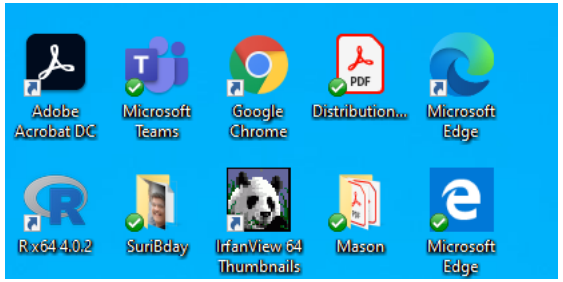
R - Philosophy

- Most data analysts will run software in an interactive mode and (slowly) transition into programming
  - User → Programmer model
- R is built on lean philosophy
  - No need to pack a lot of functionality that users will seldom use
  - A lot of functionality is in 'external packages'
  - Nearly 13,000 packages available today
- R is distributed under GNU General Public License
  - Free software to use, share and modify
  - Free software → freedom to change the software and not the price
  - R is free (pricewise) as well but commercial versions are also available
- R is actively maintained by R core-team
  - New version releases and bug-fixes

R has a very large User Community
A lot of help can be found on the web

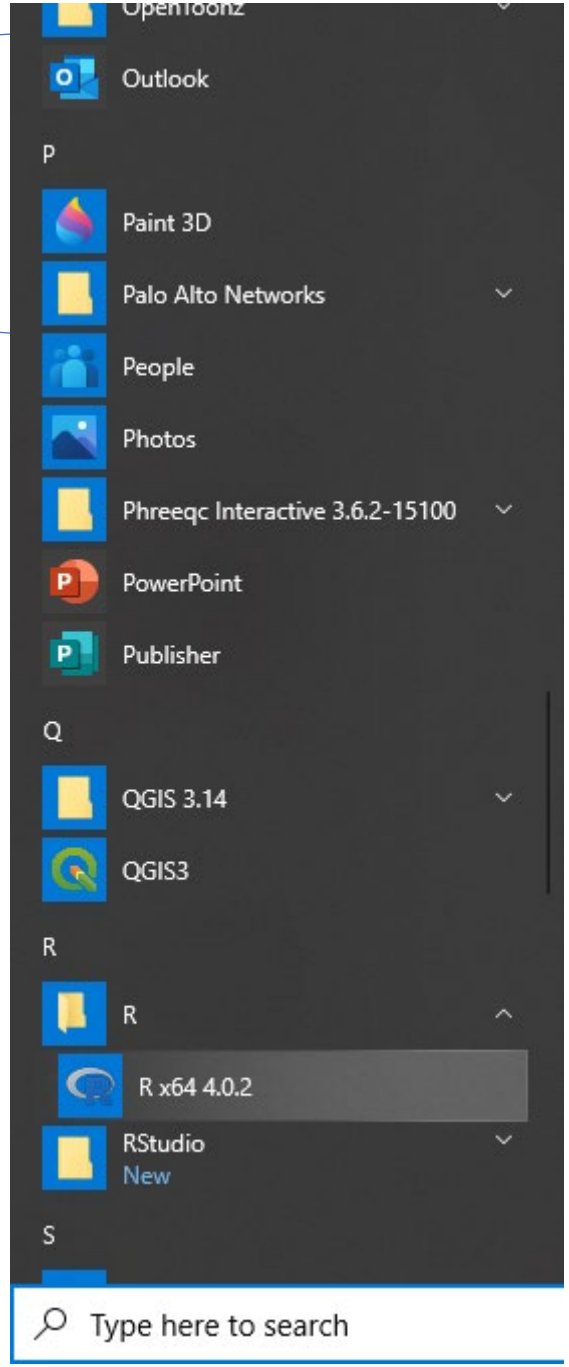
Accessing R

- Once installed R can be accessed from desktop icons or windows start menu

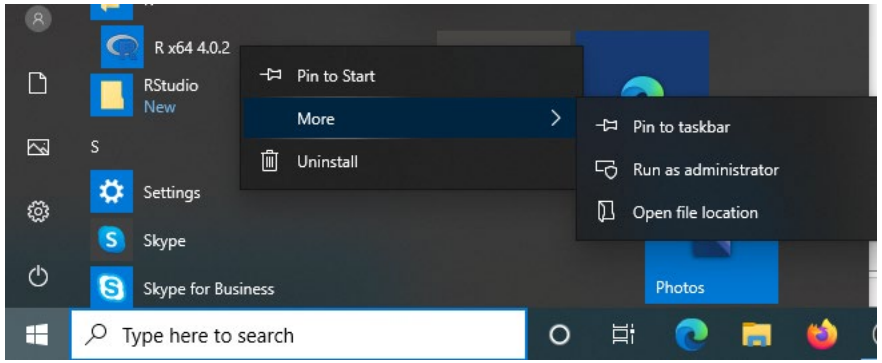


Desktop Icons

Windows Start Menu



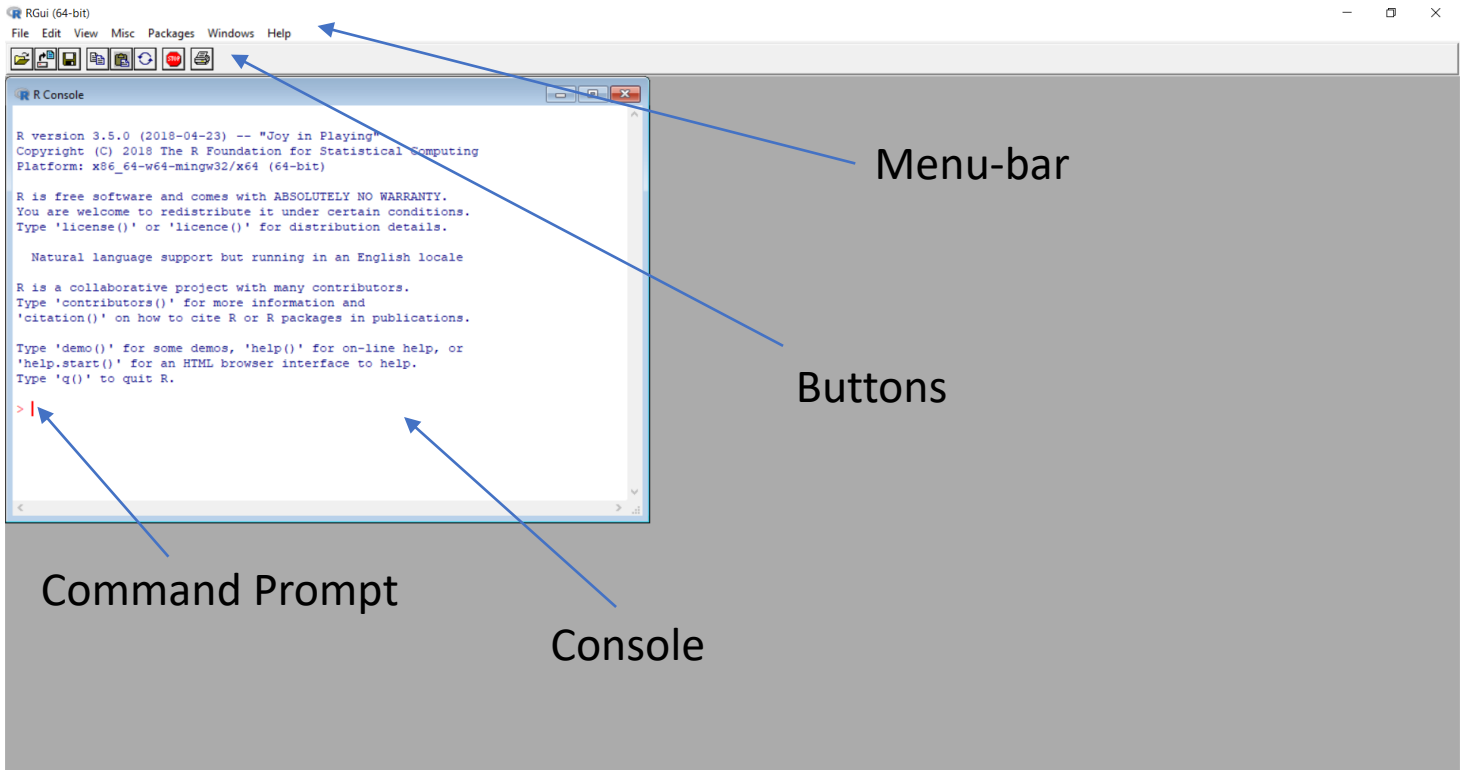
Running R



- Good practice to Run R as Administrator
  - Packages can be downloaded for all users
  - Need to have administrator privileges
- Usually run 64 bit version of R is your computer supports it
  - Runs a little faster and has more memory
  - 32 bit memory is limited to typically 2 GB
- 32-bit may be useful to run some legacy code
  - Not all packages may be 64-bit compatible, especially those not found on CRAN
- R can be run from command prompt but GUI is most commonly used

R GUI

- Basic R comes with a native GUI
- There are other GUI versions
  - Rstudio is very popular
  - Rcommander, JGR, etc.

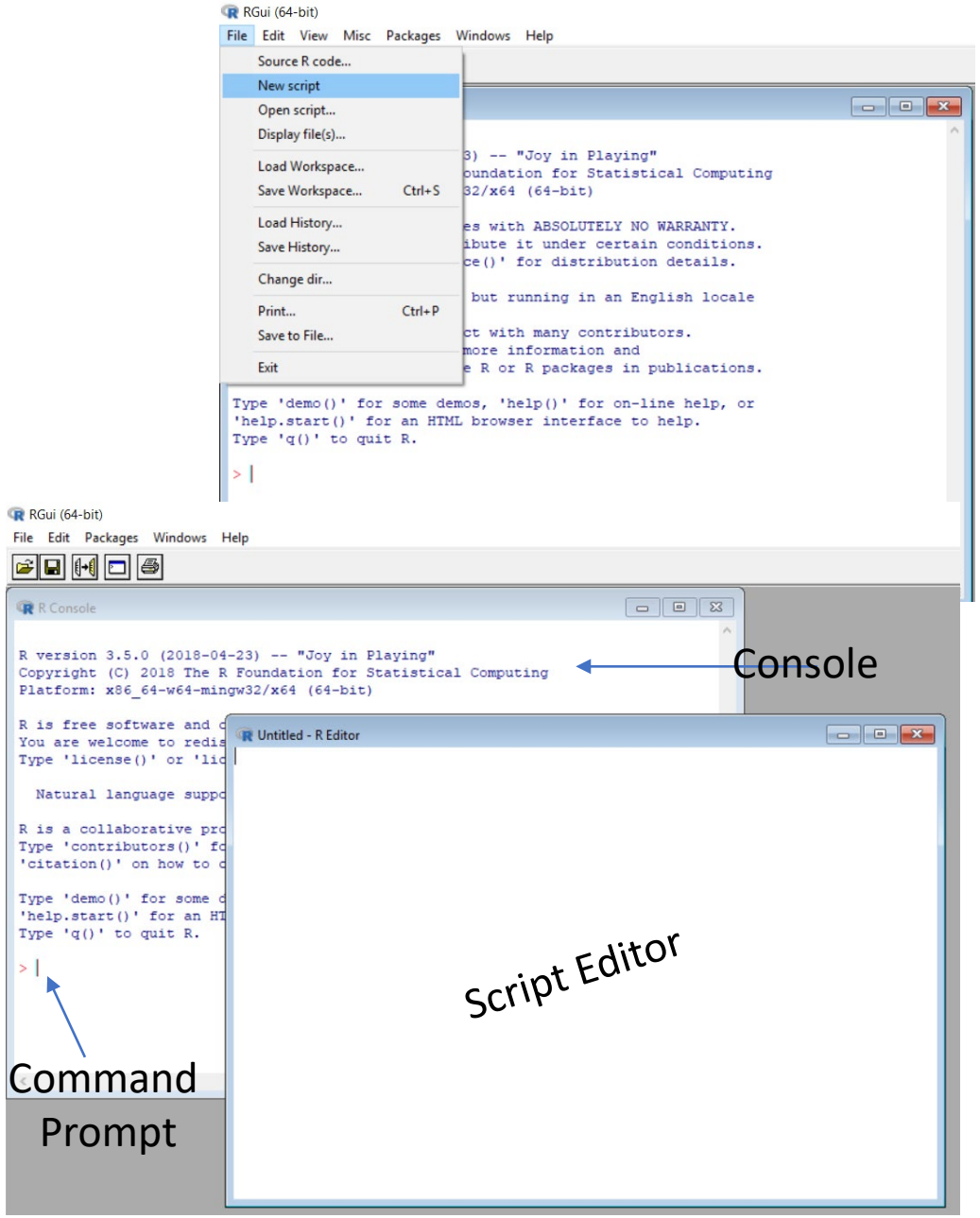


All R calculations are executed in the Console at the Command Prompt

R Script Editor

- R Script
  - A **File** containing a sequence of R commands
  - It is a textfile
  - Typically stored with .R extension
- R Script can be executed at the command line or within R GUI
  - Must adhere to R syntax
- R GUI provides a built-in script editor

Scripts help save your code and reuse it for other purposes



R Packages

- R is built on lean programming principles
  - Core package has very basic functionality
- Significant functionality added through external packages
  - You download and load packages as necessary
- R installs certain base packages with its installation
- Others must be manually installed
- Nearly 13000 packages enhance the functionality of R

R Packages Can be downloaded from CRAN Mirrors

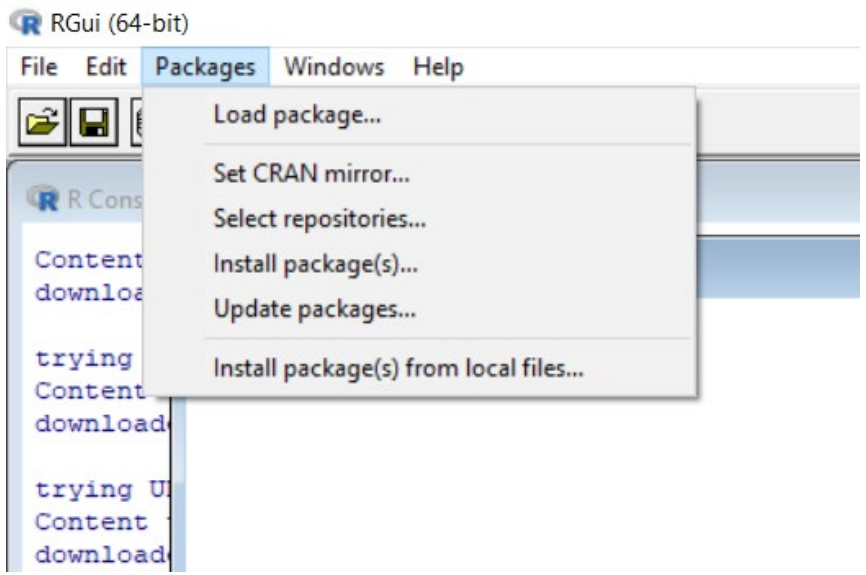
R GUI provides a Menu Item to do so easily

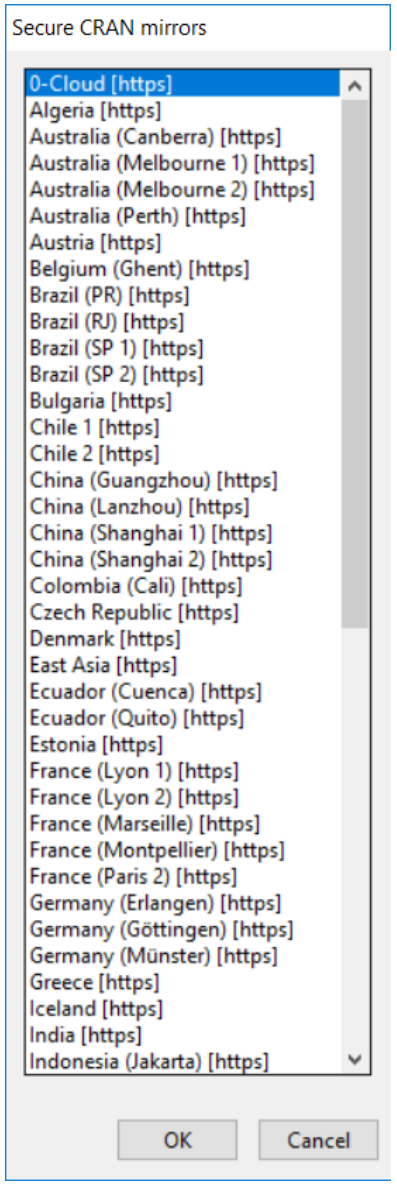
Can also use install.packages command

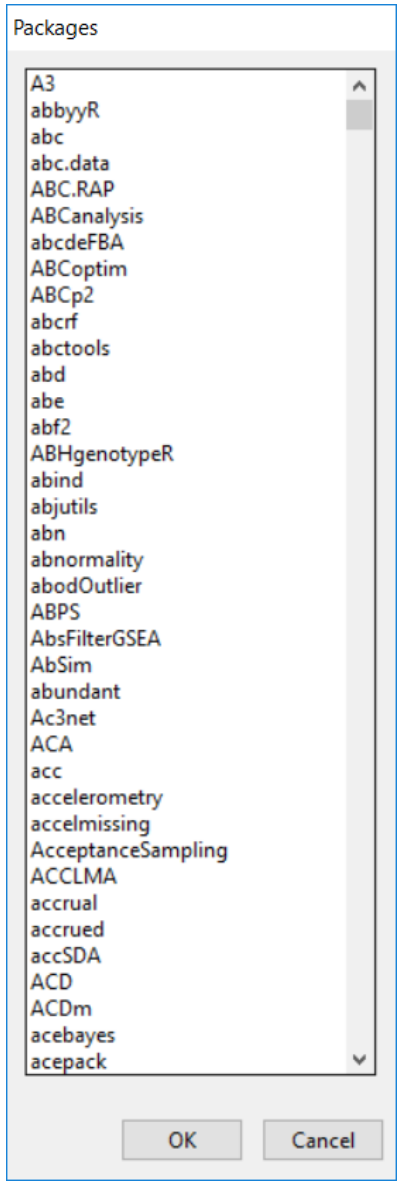
Pre-installed Packages

- base
- compiler
- datasets
- graphics
- grDevices
- grid
- methods
- parallel
- splines
- stats
- stats4
- tcltk
- tools
- translations
- utils.

R Packages



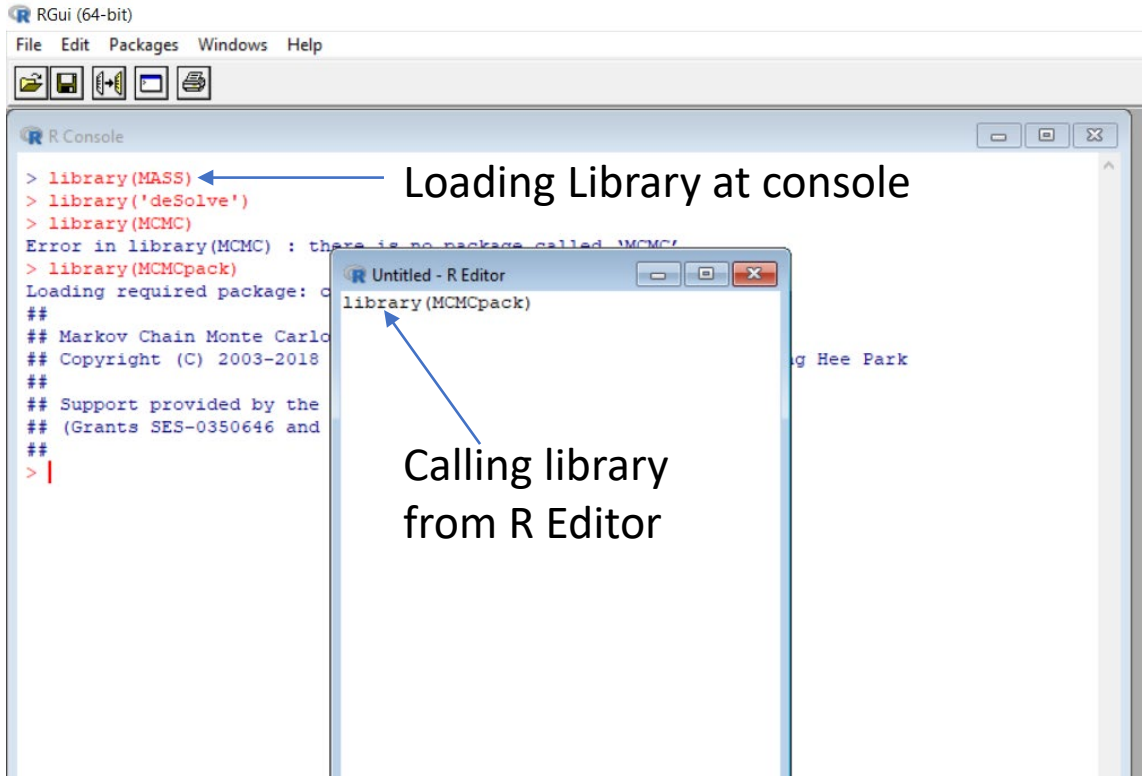




If You Run as Administrator Package is usually stored in C:\Program Files\R\R-3.5.0\library and available to all users. Otherwise it is stored in a personal library and only available to you

R Packages – Downloading versus Loading

- A downloaded package is stored on computer hard-drive
- By default it is not accessible to R unless you load it into RAM
- Use library('package name') to load a downloaded R package into memory
- You can put the library command at the command prompt or in a script
  - R Editor



It is a good idea to put the library command at the beginning of your script

Executing Commands from CONSOLE

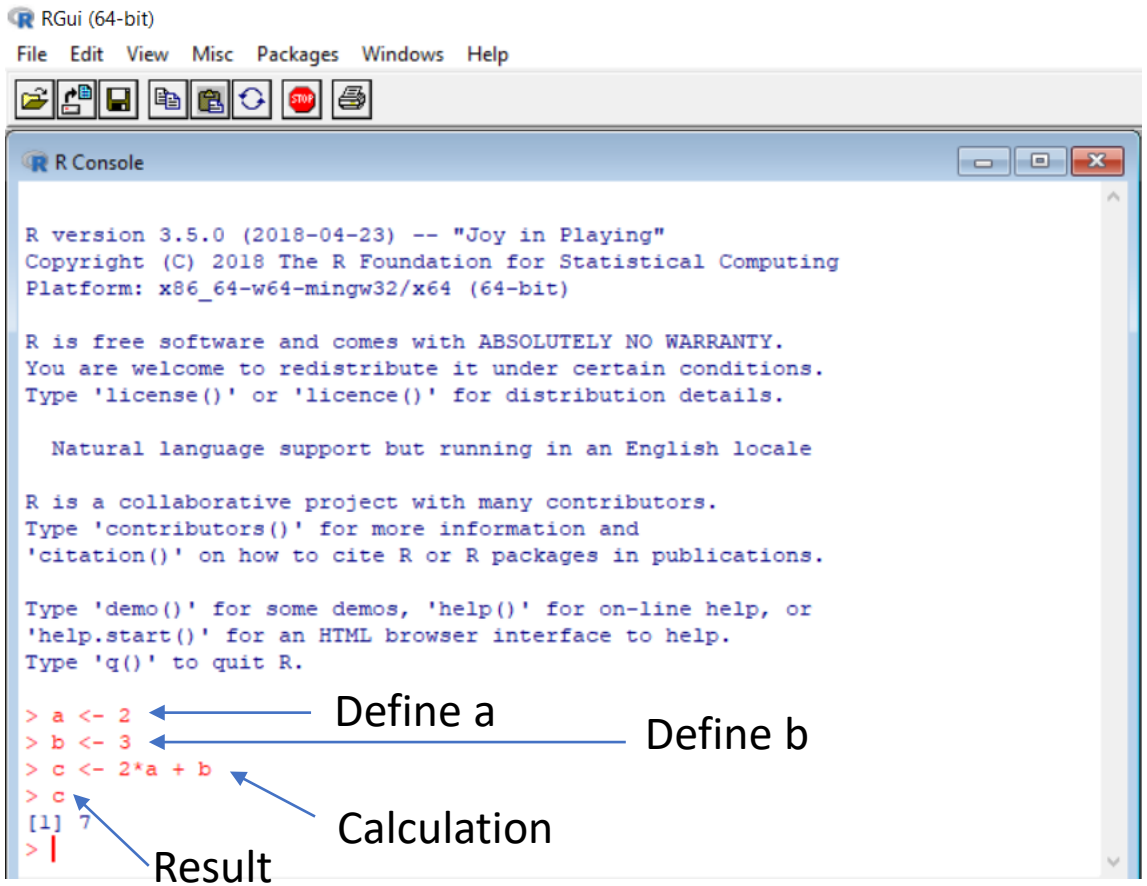
- You can execute commands sequentially from the console
- Type each command one at a time
- Hit enter after each command to execute it

<u>Problem Statement:</u>

Define two variables a and b and assign values 2 and 3

Define a third variable c = 2a + b

Find the value of c



< - is the left assignment operator (assigns value on the right to an object on the left)

Executing Commands from a Script

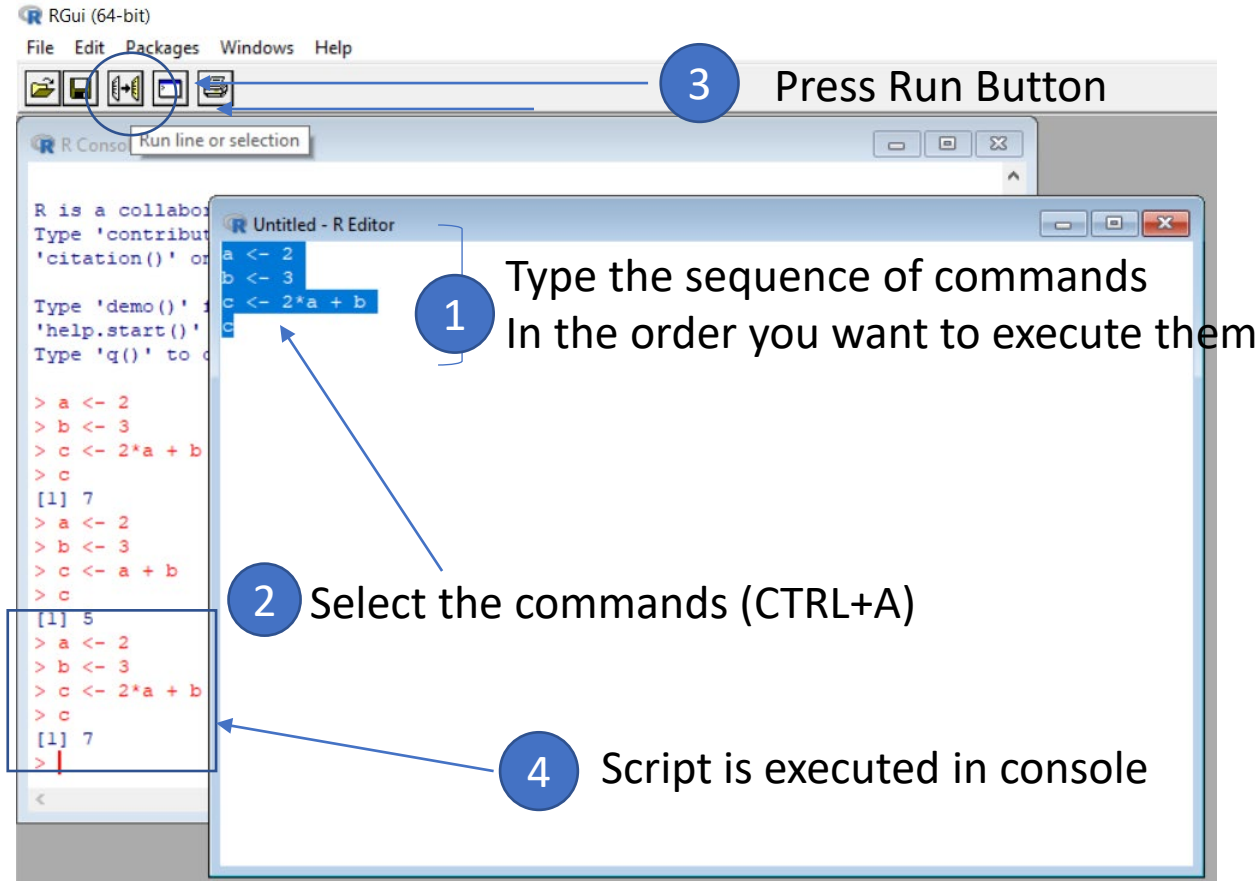
- Open a new script file
- Type up the necessary commands
  - Similar to what you would at a console
- Select the commands that you typed
- Run them
- R runs the commands from top to bottom

<u>Problem Statement:</u>

Define two variables a and b and assign values 2 and 3

Define a third variable c = 2a + b

Find the value of c



Save the Script for Future Use

Setting Working Directory

- Working directory is where R will look for and store your files
- Good practice to setup a working directory for each project
  - Store data, R code and results in one location
- Working directory is set using **setwd** command
  - getwd tells the location of working directory
- R uses the forwardslash (/) to specify the path
  - Unix type specification
- You can also use double backslash as well \\
  - Single backslash is an escape character
- Put your working directory on top your script

Untitled - R Editor

```
setwd('E:\\Dropbox\\Ethiopia-Courses\\RWorkshop\\Day1-AM\\Rcode')
a <- 2
b <- 3
c <- 2*a + b
c
```

Working Directory specification using setwd

The directory must be first created in Windows Explorer before using in setwd

You can also use mkdir command to create directories from within R

Comment Lines

- Comment lines are most important part of a script
- Comments help you make notes on what you did
- Helps remember what you did at a later date
- A line in a script starting with # is considered as a comment
- Include meta-data (what the code does who wrote it and when) in the beginning of the script using comment lines

Untitled - R Editor

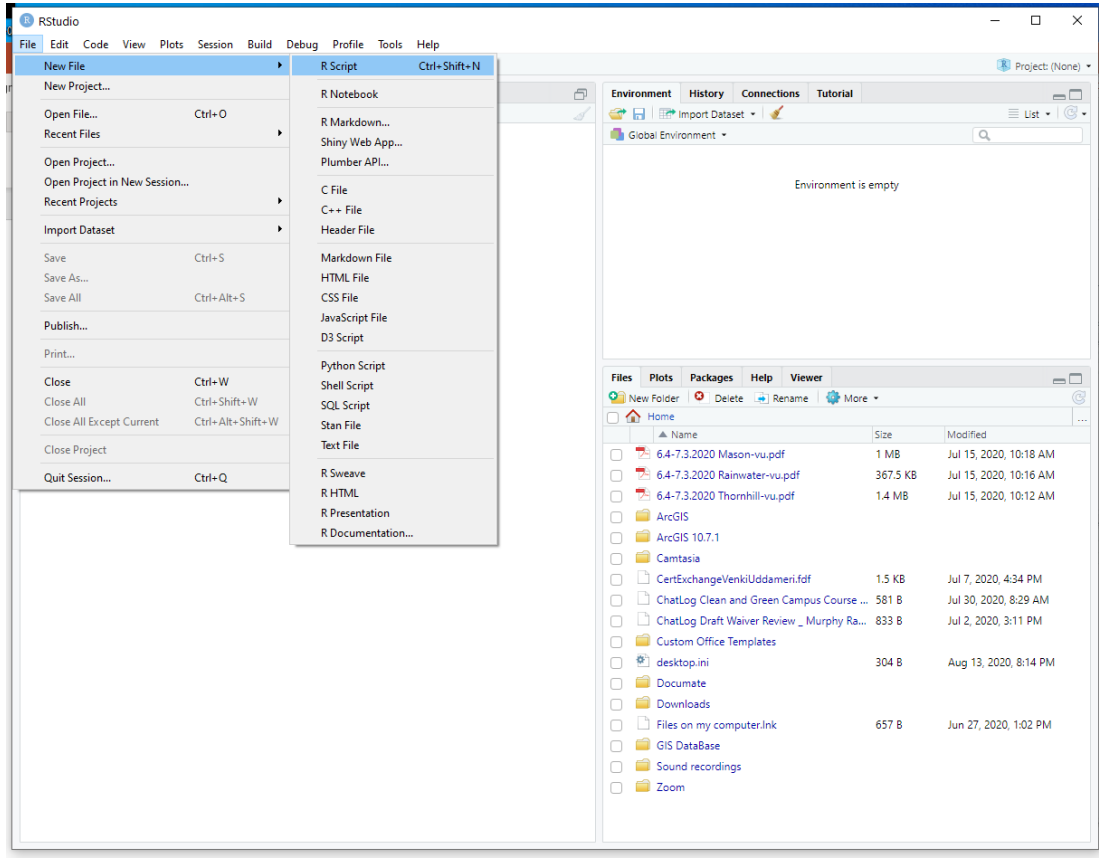
```
# Illustrative Code
# Written by Venki Uddameri 06/01/2018

# Step 1: Set Working Directory
setwd('E:\\Dropbox\\Ethiopia-Courses\\RWorkshop\\Day1-AM\\Rcode')

# Step 2: Write Input Data
a <- 2
b <- 3

# Step 3: Perform Calculations
c <- 2*a + b

# Step 4: Write the Result to Console
c
```

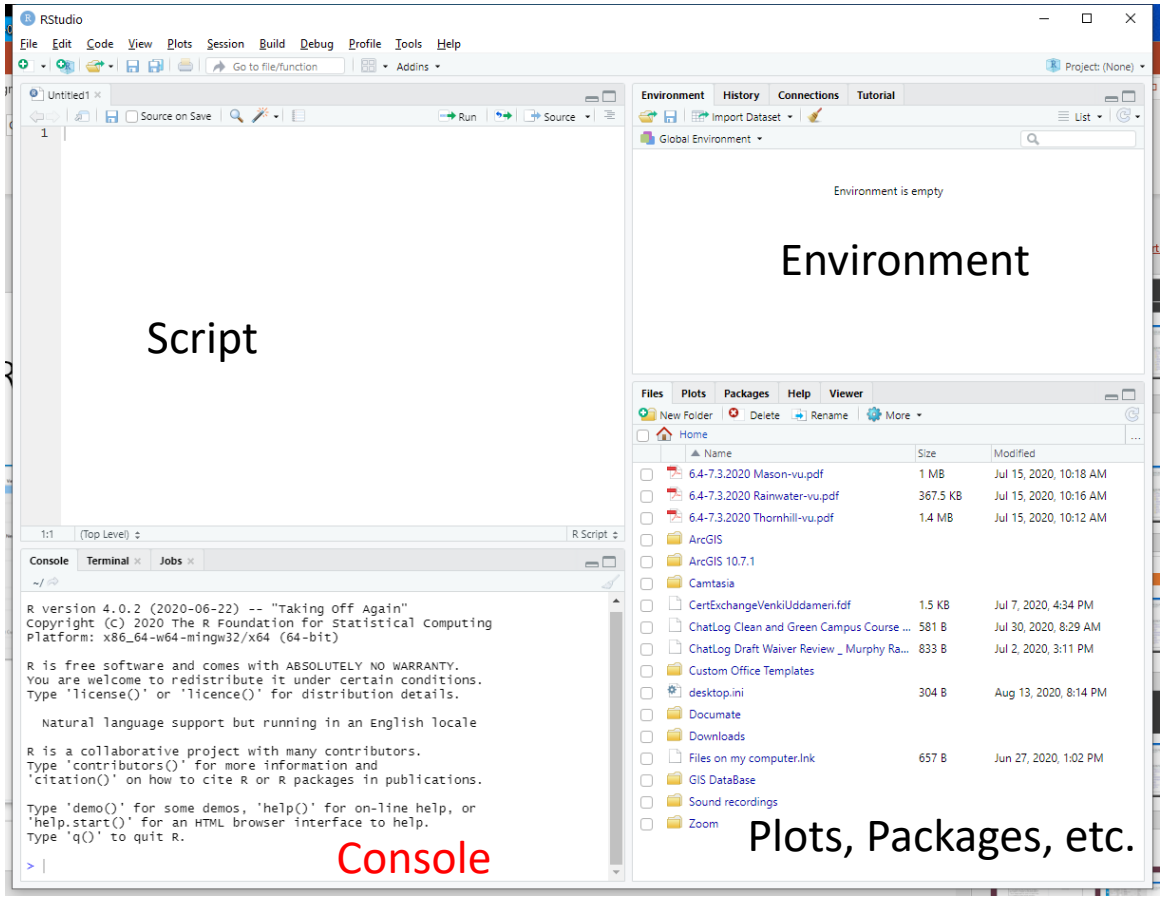
Script Meta Data using Comment Lines

Comment Lines

Comment lines are ignored when R executes the script but is still very important to have them

R Studio GUI





R Studio GUI can be used to open several types of files

Closing Remarks

- This module deals with downloading and setting up R
- Introduces console and scripts
- Setting up working directory
- Comment lines
- Some useful practices

You should know

- How to download and install R on windows
- How to download packages from CRAN mirrors
- Understand R Console and Script Windows
- How to run scripts (select the script contents and run)
- How to use **library** function to load R libraries
- How to use **setwd** function to set working directory
  - **Use windows explorer to create a folder first**

- Use of # to write comment lines
- Metadata of the script using comment lines
- What a Left assignment operator (<-) does
- Defining variables and assigning values to them
- Why it is useful to run R as administrator