211
Data Analysis in R

Instructor: Ishita Sharma
Email: isharma3@uh.edu

TA: Aravind Kumar Reddy Pasunuri
Email: apasunur@cougarnet.uh.edu

UNIVERSITY of HOUSTON
DIVISION OF RESEARCH
HPE Data Science Institute

Objectives

- Introduction to programming in R for Data Science
- Environment/Packages & Intro to R Studio
- Data loading, cleaning, visualization
- Simple Data analysis, writing scripts & functions
- Debugging, improve workflow and performance
- Hands-On

Course Organization

- We will have 15 contact hours
- Grading:

  20% Attendance

  30% Homework

  50% Project

- We will utilize the Moodle course management system for course material, home work, grades, questions etc.

https://hpedsi.uh.edu/education/training

Course Resources

- Please use your laptops (you will need them for homework & project)
- Download your exercises & slides from Moodle
- Install R & RStudio
- Software and Packages for R: http://cran.r-project.org/
- Books:
  - R for Data Science http://r4ds.had.co.nz/
  - http://www.r-project.org/doc/bib/R-books.html
  - "R in Action" http://www.manning.com/kabacoff2/?a_aid=RiA2ed&a_bid=5c2b1e1d

What is R?

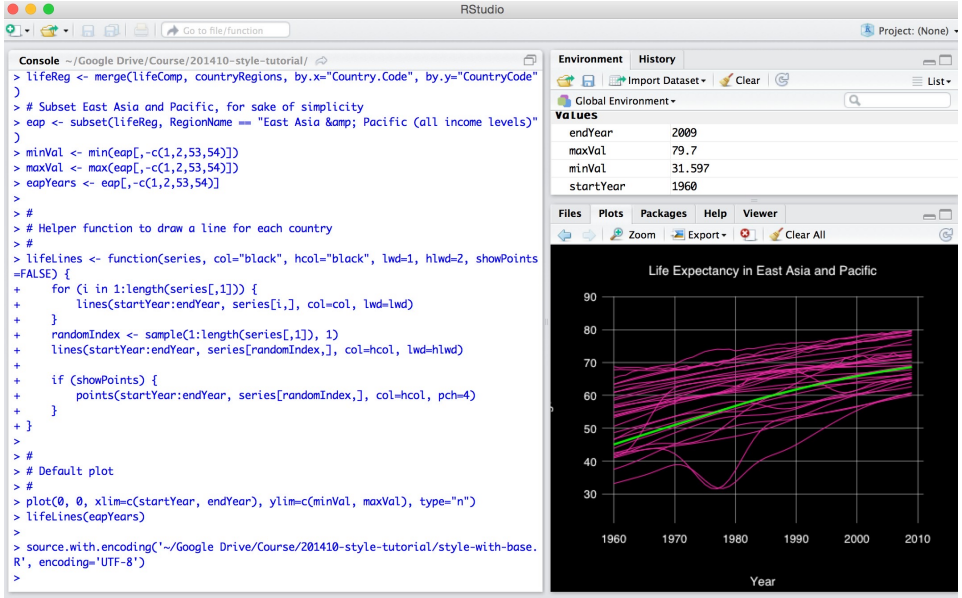
R is a language and environment for statistical computing and graphics

R is an environment for interactive data analysis

- **Data Manipulation** (connecting to data sources, slicing & dicing data)
- **Modeling & Computation** (statistical modeling, numerical simulation)
- **Data Visualization** (visualization fit of models, composing statistical graphics)
- R is freely distributed software and a GNU project
- R is a community

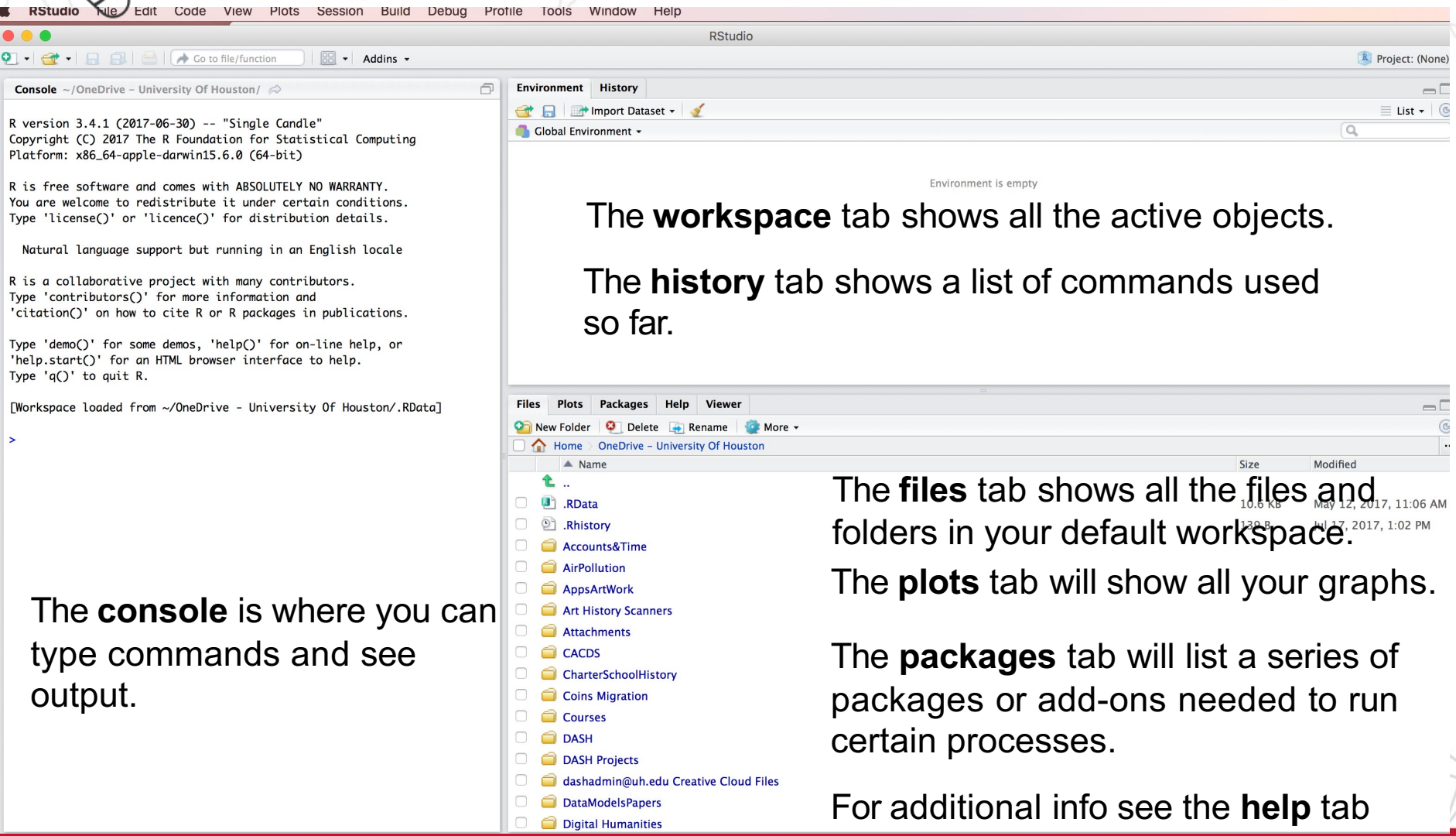
The R Environment

- Interaction with R can be done in different modes:
  - Interactive console
  - Scripts
  - Interactive GUI (e.g. R Studio)



- R can be extended (easily) via packages

R Studio



The **workspace** tab shows all the active objects.

The **history** tab shows a list of commands used so far.

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

The **files** tab shows all the files and folders in your default workspace.

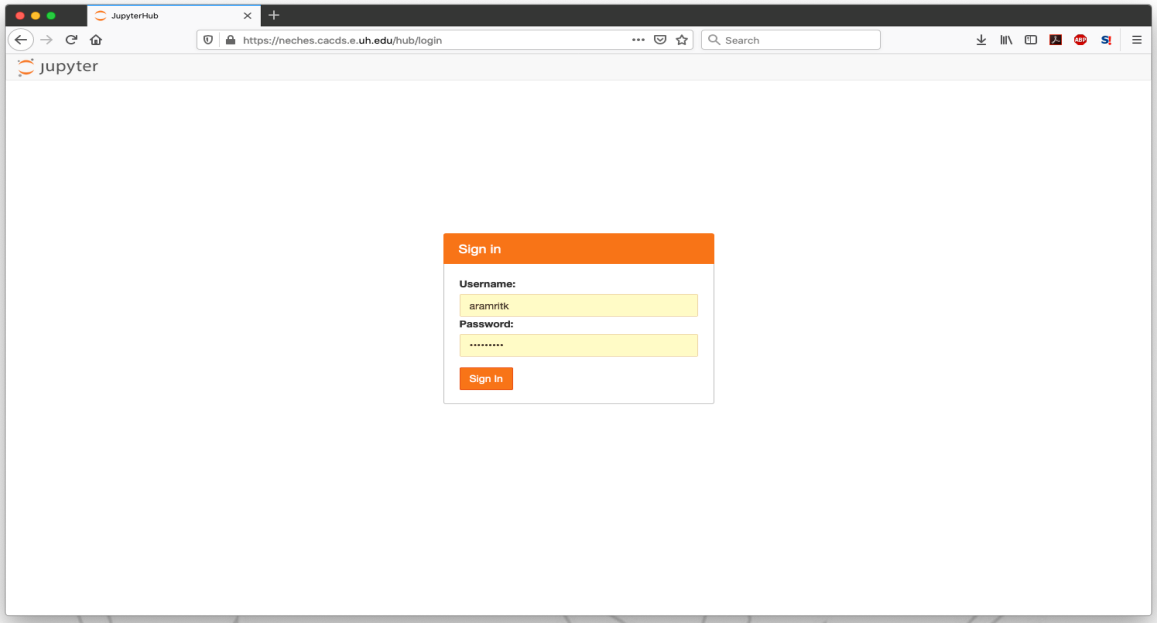
The **plots** tab will show all your graphs.

The **packages** tab will list a series of packages or add-ons needed to run certain processes.

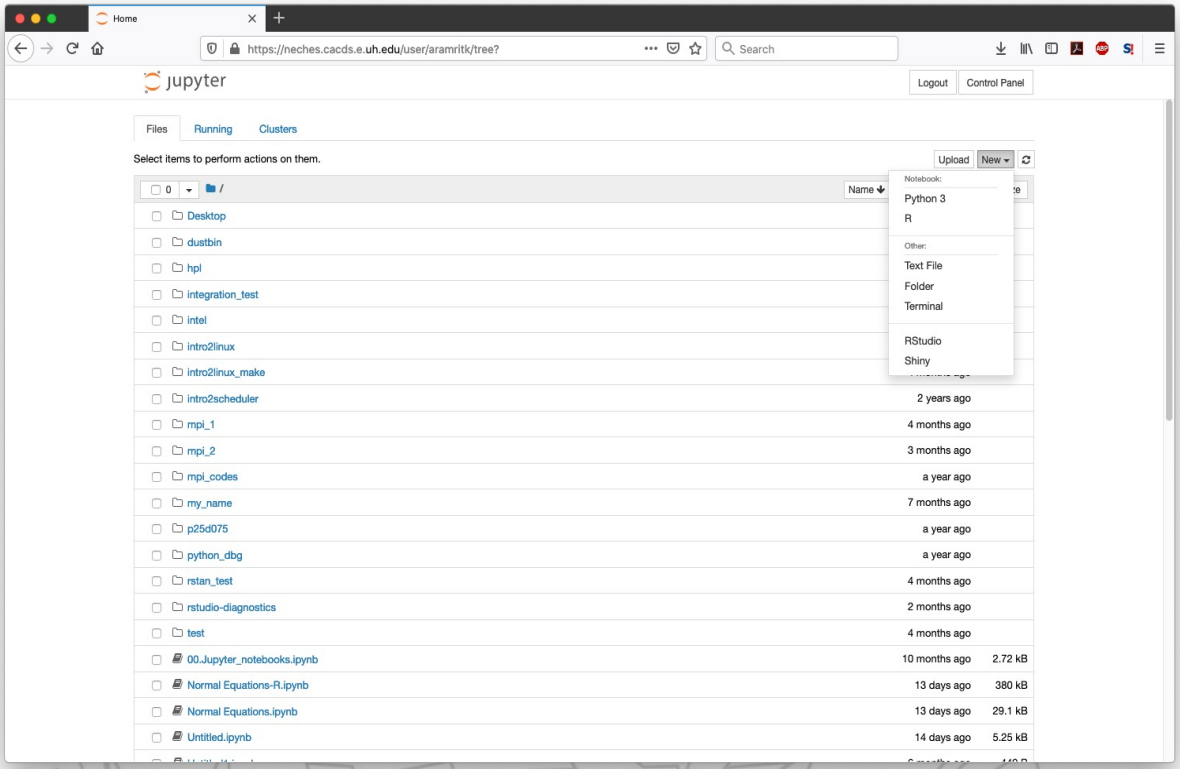
For additional info see the **help** tab

Jupyter Notebook - R

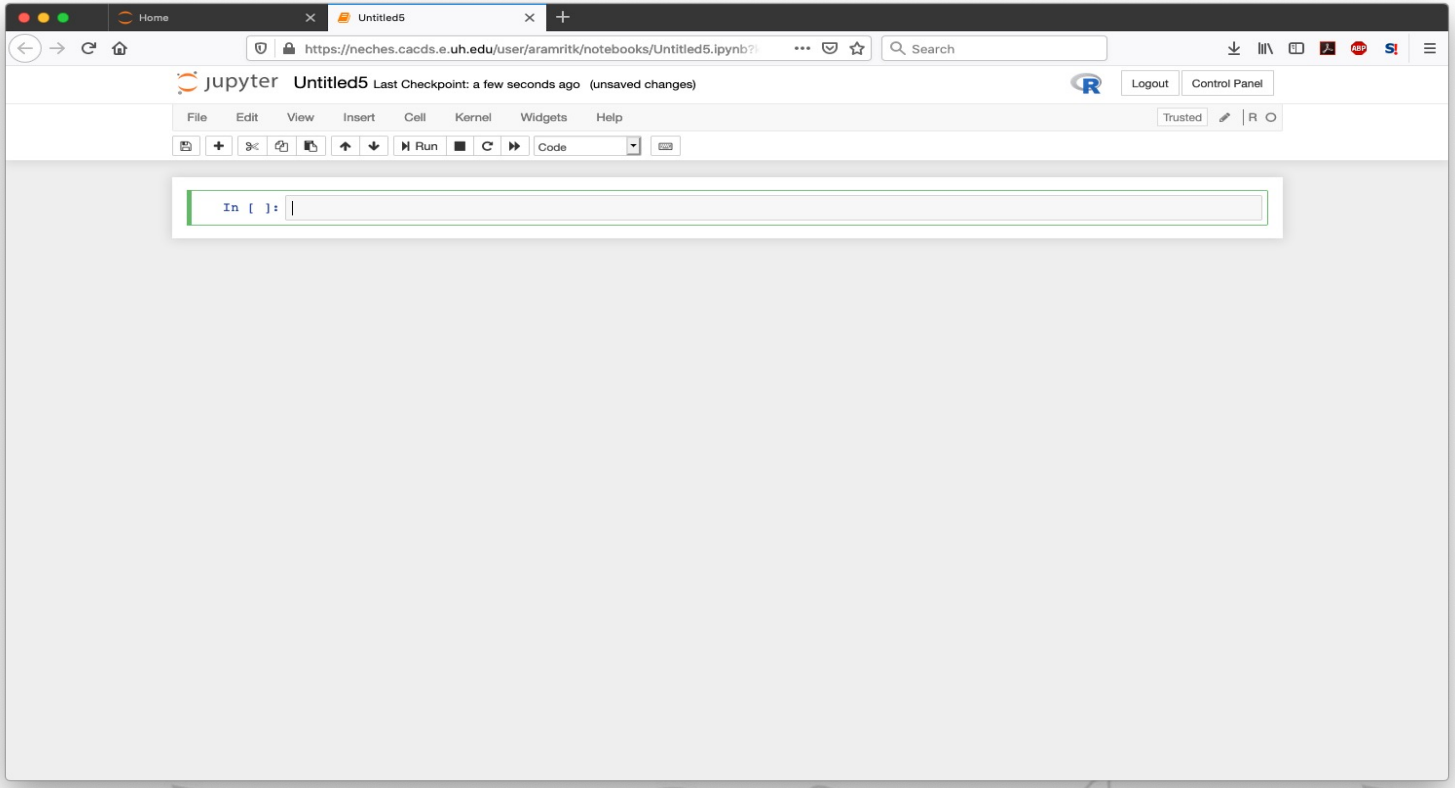
https://neches.rcdc.uh.edu/



Jupyter Notebook - R



Jupyter Notebook - R



Basics 1

- Use the console to run R and hit <ENTER>
  ```
  > 7 + 5
  ```

- Heuristics minimize output when you don't want to see it

"object name gets value"

- ```
  a <- 7 + 5 #assignment
  ```

Windows/Linux: "Alt" + "-"
Mac: "Option" + "-"

- several R commands in one line have to be separated by ";"
  ```
  > a <- 7 + 5; a - 7
  ```
  ```

Basics 2

- Names for objects (I recommend *snake convention*)

```
i_use_snake_case
otherPeopleUseCamelCase
some.people.use.periods
And_aFew.People_RENOUNCEconvention
```

- Calling functions (all R "commands" are functions)

```
> seq(1, 10)
>  [1]  1  2  3  4  5  6  7  8  9 10
> x <- "hello world"
> x <- "hello world
+
> y <- seq(1, 10, length.out = 5)
> y
> ( < seq(1, 10, length.out = 5))
  y -
```

```
function_name(arg1 = val1, arg2 = val2, ...)
```

Basics 3

Variable Name	Validity	Reason
var_name2.	valid	Has letters, numbers, dot and underscore
var_name%	Invalid	Has the character '%'. Only dot(.) and underscore allowed.
2var_name	invalid	Starts with a number
.var_name, var.name	valid	Can start with a dot(.) but the dot(.)should not be followed by a number.
.2var_name	invalid	The starting dot is followed by a number making it invalid.
_var_name	invalid	Starts with _ which is not valid

https://www.tutorialspoint.com/r/r_variables.htm

The R Language

Introduction

- R is a true object-oriented programming language, much like others such as C++, Python etc
- Objects are manipulated by functions, creating new objects which may then have more functions that can be applied to them.
- Objects can be just about anything: a single value, variable, datasets, lists of several types of objects etc.
- The object's class (e.g. numeric, factor, data frame, matrix etc.) determines how a generic function (like `summary` and `plot`) will treat the object.
- Typically there are often several ways to do the same thing depending on the objects and functions being used and the same function may do different things for different classes of objects.

Introduction

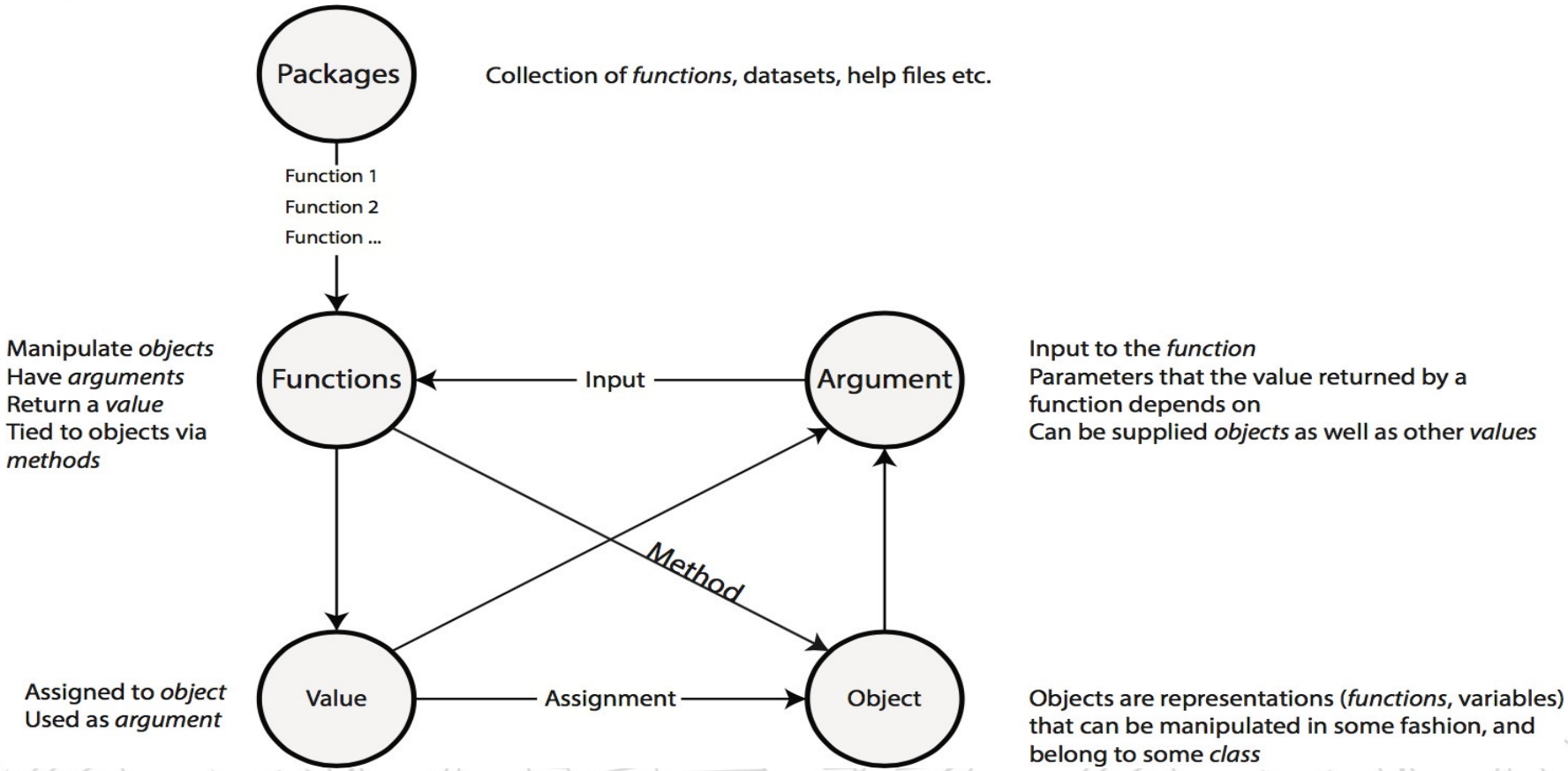


Image courtesy of:
MICHAEL CLARK, "Introduction to R", CENTER FOR SOCIAL RESEARCH UNIVERSITY OF NOTRE DAME

R Packages

- are collections of R functions, data, and compiled code in a well-defined format.

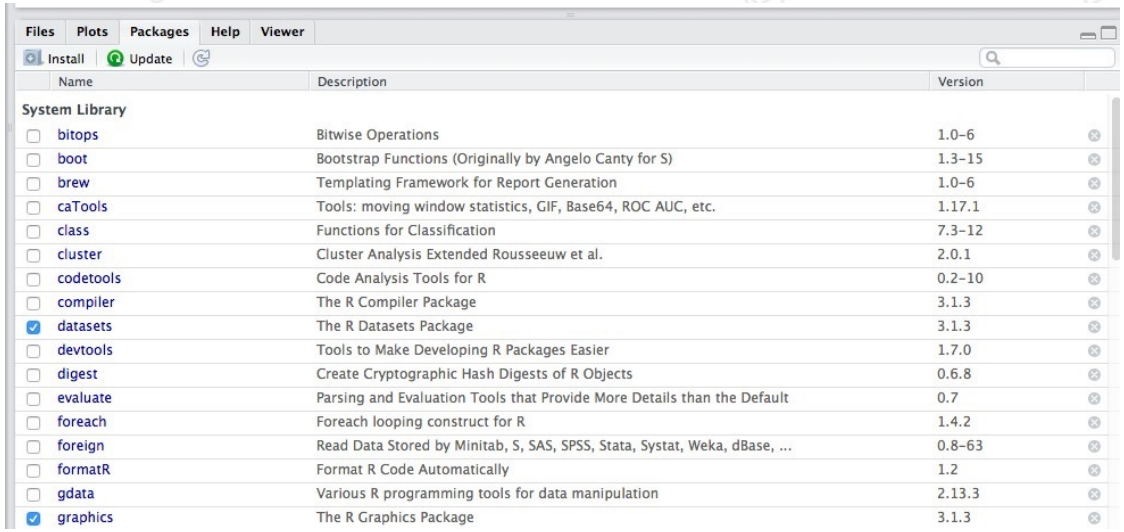
- The directory where packages are stored is called the *library*.

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

- 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.

R Packages

- R Studio:



Exercise: Install package *nycflights13*, load it and find documentation

- https://cran.r-project.org/web/packages/nycflights13/nycflights13.pdf

- Homework: Install package *tidyverse*