Installing R

Introduction to R

R is an "object based language" meaning all the commands and code are centered on the creation, manipulation, alteration, & analysis of "objects", we'll get into a bit more about what this means later on but for now just understand that when we open R there isn't anything "in" it, we as programmers and researchers must populate our ENVIRONMENT (see jargon doc) with the objects necessary for our analytical tasks. Today we installed R & R studio; tested out different working directories; and installed, loaded, & utilized a package.

Installing R & R Studio

So a quick aside, although often used interchangeablely R & R studio are two distinct things. R is the programming language & R studio is one (of many) user interface systems (more formally, an Integrated Devlopment Environment or IDE). Consider the example of words and paper, words can be used without paper or the act of writing but through the use of writing on paper we are able to organize our choice of words, style, and more easily identify errors. Likewise R code can be used without an IDE but there are substantial advantages to using an IDE. However, we are are Political Scientists not Computer Scientists and so we don't need to delve too deep into the weeds. Onto to R & R Studio installation.

This is the link to the website to install R, just follow the prompts https://cran.r-project.org/mirrors.html

This is the link to the website to install R Studio, likewise just follow the prompts https://posit.co/download/rstudio-desktop/

Setting Up Directories and Choosing a Working Directory—- The command setwd() is used to set your working directory, a working directory is just a designated location in your computer's file system that R has permission to read and write files to. This enables you to use certain short-hands as well as keep all your work organized.

setwd() takes a single ARGUMENT, that is a file path which corresponds to a real location in your computer's file directory I'm going to set my working directory to my Teaching folder

which is located in my "2023 or Earlier" folder which is turn located in my "R Stuff" folder which is on my Desktop. R needs to know all these steps to be able to find the folder, this is called the PATH. Pathes will be more or less unique to your machine.

```
setwd("C:/Users/19107/Desktop/R Stuff/2023 or Earlier/Teaching")
```

When we highlight line 41 and hit Run (or ctrl+enter) we should see the line appear in our console in blue text but nothing else should happen which is good!

Now I'll use a more specific directory so that I can illustrate some advantages

```
setwd("C:/Users/19107/Desktop/R Stuff/2023 or Earlier/Research/Is it the
↪  Economy")
```

One advantage of utilizing a working directory is it allows calling files by short-hand instead of the full path. See below

```
setwd("C:/Users/19107/Desktop/R Stuff/2023 or Earlier/Research/Is it the
↪  Economy")

head(read.csv("usa_gdp.csv"), n=20L)
```

```
   gdp_growth year
1   2.3000000 1961
2   6.1000000 1962
3   4.4000000 1963
4   5.8000000 1964
5   6.4000000 1965
6   6.5000000 1966
7   2.5000000 1967
8   4.8000000 1968
9   3.1000000 1969
10 -0.2834913 1970
11  3.2933516 1971
12  5.2589074 1972
13  5.6457226 1973
14 -0.5405517 1974
15 -0.2054561 1975
16  5.3881393 1976
17  4.6241531 1977
18  5.5353050 1978
19  3.1661453 1979
```

```
20 -0.2567497 1980
```

vs without the correct directory specification

```
head(read.csv("C:/Users/19107/Desktop/R Stuff/2023 or
 ↪  Earlier/Research/Is it the Economy/usa_gdp.csv"), 20)
```

```
   gdp_growth year
1   2.3000000 1961
2   6.1000000 1962
3   4.4000000 1963
4   5.8000000 1964
5   6.4000000 1965
6   6.5000000 1966
7   2.5000000 1967
8   4.8000000 1968
9   3.1000000 1969
10 -0.2834913 1970
11  3.2933516 1971
12  5.2589074 1972
13  5.6457226 1973
14 -0.5405517 1974
15 -0.2054561 1975
16  5.3881393 1976
17  4.6241531 1977
18  5.5353050 1978
19  3.1661453 1979
20 -0.2567497 1980
```

Understanding the Role of Packages in R

A quick explanation of packages and their role in R: R by itself contains a huge set of commands for all sorts of tasks but naturally it cannot contain every possible utility or operation that every possible user could need. However, R enable and empowers users to make their own commands or suites of commands and they can be shared with others users, these are called packages. Packages are essential to R programming especially for researchers; certain models, important basic commands, and processes for dealing with data types are all made accessible through the use of packages.

Installing & Loading Packages—-

Let's start with a simple use of a package. Anyone familiar with Stata will be familiar with it's proprietary file format: the .dta. DTA files are ubiquitous and have some attractive features, including a useful labeling system for variables unfortunately BASE R does not come with a command for reading DTA files, but, a kind soul named Hadley Wickham took it upon themself to creation such a command and we can access this command through the haven package.

the first time you want to use a package it must be INSTALLED before it can be LOADED and used. This is done using the install.packages() command. See below

```
install.packages("haven")
```

Now that is it installed we can load it, with the exception of updates we will never need to install the package again.

the command for loading packages is library(), note that now we don't need to call the package's name in quotes.

```
library(haven)
```

Now we can read dta files like the ANES 2020 Timeseries Survey that we retrieved from the election study website.

```
read_dta("anes_2020.dta")
```