SOCI 1230, Week One, Class One

Matt Lawrence

January 10, 2022

Getting Started

To begin, copy all the text on this screen. Then open RStudio, and in the "File" menu select "New File" then "R Markdown". Add a title like "SOCI 1230, Week One, Class One", include your name in the "Author" box, select "PDF" as the Default Output Format, and click Ok. A default markdown file will open. Delete all the text in the default file below the header (everything after line 7) and paste in the text you copied to your clipboard. Make sure that "R Markdown" is selected in the file type drop down menu in the bottom right corner of this pane. Save this file in the folder you created during the morning's session for this course.

Welcome to R Studio!

Let's start with a recap of some navigation before we begin:

- The upper left pane is the *text editor* where you keep all of your commands.
- The lower left pane is the *console* where the commands will be executed.
- The upper right pane's `Environment` window will show the data that you load.
- The lower right pane will show previews of your R Markdown documents, plots, help documents, packages, and other features.

This file is an R Markdown Notebook which is an R Markdown file that allows you to integrate text, R code, and R output into a single document. This is the recommended file type for our section, but you may want to use other file types in the future. All the possible file types are found when you click File>New File. For our purposes - and for producing transparent and reproducible research - the advantage of an R Notebook is that you can easily include text, notes, and comments that are not code (like this paragraph).

How does R know what is code and what is regular text? In an R Notebook, code goes in a fenced code block. To open the fence, type three backticks and an r in curly brackets on a single line; to close the fence, type three backticks on a single line like this:

Everything between those fences will be run as code (unless it is preceded by a hashtag).

Want a shortcut? On a mac, Command-Option-I will open a blank code block. For Windows, use Control-Alt-I.

Let's start with a reminder of how to save objects in R. Since even the following calculation is considered code we need to place it within a code block. After the r, in the line where you open your fence, you can provide a short description of what the code in that chunk does. Those short descriptions are collected as bookmarks in the drop down menu on the bottom of this pane and can help you find pieces of your file.

Note that multiple chunks cannot have the same description. If they do, the file will not knit.

In the chunk below, what do you need to change to save our course number as an object named `coursenumber`?

```r
600 + 630
```

```
## [1] 1230
```

After writing the code to complete the object assignment, press the right-facing green arrow to run the code in the current chunk. The icon in the center runs all the code in chunks above the current chunk. The gear icon offers some option (like naming the chunk).

The Mac keyboard shortcut for running the individual line where your cursor blinks is: Command-Return. The Mac keyboard shortcut for running the full chunk in which your cursor blinks is: Shift-Command-Return.

Introducing R Markdown

For our section, we will be using R Notebooks with R Markdown to write and run our code. R Markdown is a plain text-based form of writing with lots of neat tricks. For example, look at the heading for this paragraph. The two hashtags are how we start a new section in R Markdown. If we used one hashtag, the section heading would be bigger in the output. If we used three (or four) hashtags, the section headings would be smaller (or even smaller) in the output, which would be helpful if you have lots of sub- (or sub-sub-) headings.

There's an R Markdown cheat sheet in the pages folder on Canvas that has much more info on how to use R Markdown. Here are a few first day features to get you started:

Wrapping a word in `single backticks` will highlight it when you print.

Wrapping text in one asterisk will italicize it when you print.

Wrapping text in two asterisks will bold it when you print.

Wrapping text in three asterisks will bold and italicize it when you print.

And three (or more) dashes on a single line will create a horizontal rule:

Click the "Knit" button in the toolbar to see how this R Markdown file prints as a PDF.

Getting Started With Opportunity Insights Data

In this morning's session, we installed the `ggplot2` package and saw how to load it with the `library()` function. That package is part of the suite of packages called the `tidyverse` that you installed at the start of this class. To use the packages, we need to load them with the `library()` function.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.0     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Next we'll get the data we want from the Opportunity Insights webpage and save it as a data frame called `mobility`.

```
mobility <- read.csv("https://opportunityinsights.org/wp-content/uploads/2018/04/mrc_table2.csv", string
```

How many observations are there in this data frame?

Types of Variables

We have different types of variables in our data frame. Some have the `int` tag identifying them as interval variables, some have the `Factor` tag identifying them as categorical variables, and some have the `numeric` tag.

What is the difference between an interval variable and a numeric variable?

Let's open the spreadsheet view of the data frame. What are the rows? What are the columns?

To see how many observations have each value of a variable, we will use the `table()` function. In R, we have to reference all variables by their data frame. So if we want a table of the `tier_name` variable, we first have to tell R that the variable is in the `mobility` data frame. We separate the data frame and the variable name by a dollar sign:

```
table(mobility$tier_name)
```

```
##
##      Attending college with insufficient data
##                                             1
##                            Four-year for-profit
##                                            78
##                         Highly selective private
##                                            71
##                          Highly selective public
##                                            26
```

```
##                                     Ivy Plus
##                                           12
##       Less than two-year schools of any type
##                                           46
## Nonselective four-year private not-for-profit
##                                           79
##                Nonselective four-year public
##                                           74
##      Not in college between the ages of 19-22
##                                            2
##      Other elite schools (public and private)
##                                           64
##                              Selective private
##                                          584
##                               Selective public
##                                          375
##  Two-year (public and private not-for-profit)
##                                          718
##                             Two-year for-profit
##                                           72
```

When you run that line of code you should see a table with every possible value of the `tier_name` variable and the number of observations with that value. Remember our observations are colleges, so the table tells us there are 71 colleges in the "Highly selective private" tier, for example. How many colleges are in the selective public tier? Which tier name is most common?

By default, R will sort factor variables alphabetically. An easier way to see which factor has the most observations - in this case, which tier has the most colleges - is to use the `sort()` function to order the names by the number of observations in each. Note that functions "wrap" code in parentheses:

```
sort(table(mobility$tier_name))
```

```
##
##      Attending college with insufficient data
##                                             1
##      Not in college between the ages of 19-22
##                                             2
##                                     Ivy Plus
##                                           12
##                       Highly selective public
##                                           26
##       Less than two-year schools of any type
##                                           46
##      Other elite schools (public and private)
##                                           64
##                      Highly selective private
##                                           71
##                           Two-year for-profit
##                                           72
##                Nonselective four-year public
##                                           74
##                          Four-year for-profit
##                                           78
## Nonselective four-year private not-for-profit
```

```
##                                          79
##                              Selective public
##                                         375
##                             Selective private
##                                         584
##  Two-year (public and private not-for-profit)
##                                         718
```

```
sort(table(mobility$tier_name), decreasing=TRUE)
```

```
##
##  Two-year (public and private not-for-profit)
##                                         718
##                             Selective private
##                                         584
##                              Selective public
##                                         375
## Nonselective four-year private not-for-profit
##                                          79
##                           Four-year for-profit
##                                          78
##                 Nonselective four-year public
##                                          74
##                            Two-year for-profit
##                                          72
##                       Highly selective private
##                                          71
##      Other elite schools (public and private)
##                                          64
##        Less than two-year schools of any type
##                                          46
##                        Highly selective public
##                                          26
##                                     Ivy Plus
##                                          12
##      Not in college between the ages of 19-22
##                                           2
##      Attending college with insufficient data
##                                           1
```

To find the tier that Middlebury is in, recall the value of `mobility$tier_name` as an object and use brackets to index for the observation where the value of `mobility$name` is equal to "Middlebury College".

```
mobility$tier_name[mobility$name=="Middlebury College"]
```

```
## [1] Other elite schools (public and private)
## 14 Levels: Attending college with insufficient data ... Two-year for-profit
```

Now let's look at a numeric variable - `par_q1` - which captures the proportion of a college's students coming from the bottom quintile of the distribution of parent's income. Chetty et al call this measure the ***access rate***. For numeric variables, use the `summary()` function rather than the `table()` function to see the distribution. For example, run the following code chunk:

```
summary(mobility$par_q1)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.01119 0.05918 0.10161 0.12509 0.16414 0.60977
```

Find this value for Middlebury.

REPLACE THIS LINE WITH YOUR CODE CHUNK

```
mobility$par_q1[mobility$name=="Middlebury College"]
```

```
## [1] 0.02321795
```

And find Middlebury's value of `par_q5` (the proportion of students who grew up in the top quintile).

REPLACE THIS LINE WITH YOUR CODE CHUNK

```
mobility$par_q5[mobility$name=="Middlebury College"]
```

```
## [1] 0.7580932
```

Here are the cutoffs for the distribution:

- `par_q1` = Less than 20,000
- `par_q2` = 20,000 - 37,300
- `par_q3` = 37,301 - 65,300
- `par_q4` = 65,301 - 110,200
- `par_q5` = 110,201 or more

We can get even more precise at the top of the income distribution:

- `par_top10pc` = 157,700 or more
- `par_top1pc` = 630,500 or more
- `par_toppt1pc` = 14,760,800 or more

What is Middlebury's proportion of students from the top 1 percent?

REPLACE THIS LINE WITH YOUR CODE

```
mobility$par_top1pc[mobility$name=="Middlebury College"]
```

```
## [1] 0.2110714
```

We can compare these proportions in ratios. Here is Middlebury's ratio of the top 1 percent to bottom 20 percent.

```
.2110714 / .02321795
```

```
## [1] 9.090872
```

How would you describe that value in words?

Outcome Data

Now let's turn from students' backgrounds to their economic outcomes when they are in their early 30's. Here are the cutoffs for the student's ("kid's") quintiles:

- `k_q1` = Less than 1,000
- `k_q2` = 1,001 - 19,000
- `k_q3` = 19,001 - 35,000
- `k_q4` = 35,001 - 56,000
- `k_q5` = 56,001 or more
- `k_top10pc` = 77,100 or more
- `k_top1pc` = 182,500 or more

What proportion of students from Middlebury College end up in the top quintile?

REPLACE THIS LINE WITH YOUR CODE

```
mobility$k_q5[mobility$name=="Middlebury College"]
```

```
## [1] 0.5536269
```

That measure is for all Middlebury students. To capture economic mobility, we need to know the proportion of students who grew up in the bottom quintile of the parent's income distribution *and* who end up in the top quintile of the kid's income distribution. That variable is `kq5_cond_parq1`, corresponding to *kid's quintile 5 conditional on parent's quintile 1.* Chetty et al call that measure the ***success rate***. What is this proportion at Middlebury?

REPLACE THIS LINE WITH YOUR CODE CHUNK

```
mobility$kq5_cond_parq1[mobility$name=="Middlebury College"]
```

```
## [1] 0.5460001
```

How do you think you find the proportion of Middlebury students from the top quintile who stay in the top quintile?

REPLACE THIS LINE WITH YOUR CODE CHUNK

```
mobility$kq5_cond_parq5[mobility$name=="Middlebury College"]
```

```
## [1] 0.5784013
```

Mobility Table Function

So far we have been pulling values of individual mobility *origins* and *destinations*. It would be nice to look at the full distribution of parent's quintiles and the full distribution of kid's quintiles. That is what a ***mobility table*** would do. This is a bit advanced for the first day, so I have given you the code below to write a function to pull the mobility table for any college in the dataset. Run the following chunk to save the function as an object called `mobility_table`.

```
mobility_table <- function(x) {
  df <- mobility |>
    filter(name == x)

  pq1 <- c(df$kq1_cond_parq1,
           df$kq2_cond_parq1,
           df$kq3_cond_parq1,
           df$kq4_cond_parq1,
           df$kq5_cond_parq1)

  pq2 <- c(df$kq1_cond_parq2,
           df$kq2_cond_parq2,
           df$kq3_cond_parq2,
           df$kq4_cond_parq2,
           df$kq5_cond_parq2)

  pq3 <- c(df$kq1_cond_parq3,
           df$kq2_cond_parq3,
           df$kq3_cond_parq3,
           df$kq4_cond_parq3,
           df$kq5_cond_parq3)

  pq4 <- c(df$kq1_cond_parq4,
           df$kq2_cond_parq4,
           df$kq3_cond_parq4,
           df$kq4_cond_parq4,
           df$kq5_cond_parq4)
```

```
  pq5 <- c(df$kq1_cond_parq5,
           df$kq2_cond_parq5,
           df$kq3_cond_parq5,
           df$kq4_cond_parq5,
           df$kq5_cond_parq5)

  college_table <- rbind(pq1, pq2, pq3, pq4, pq5)

  colnames(college_table) <- c("kq1", "kq2", "kq3", "kq4", "kq5")

  round(college_table,3)
}
```

With the function set up, all we have to do is recall the name of the function and enter a specific value of the `name` variable.

```
mobility_table("Middlebury College")
```

```
##       kq1   kq2   kq3   kq4   kq5
## pq1 0.098 0.103 0.077 0.175 0.546
## pq2 0.121 0.084 0.224 0.114 0.457
## pq3 0.094 0.150 0.081 0.208 0.468
## pq4 0.098 0.090 0.099 0.239 0.474
## pq5 0.075 0.100 0.097 0.150 0.578
```

The key for interpreting mobility tables is to compare the four corner cells to the cells along the diagonal (top left to bottom right).

To find the names of other colleges, open the spreadsheet view of the `mobility` data frame and use the search box.

What do the mobility tables look like for some of the other colleges discussed in the Tough chapter?

REPLACE THIS LINE WITH YOUR CODE CHUNK