Importing and ‘tidying’ data

Lacey Hartigan

12/6/21

```
knitr::opts_chunk$set(echo = TRUE)

library(haven)
library(readxl)
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   2.1.1     v forcats 0.5.1

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

Expanding R skills to examine our data

Today, we’re going to continue our exploration of descriptive statistics… only this week, we’re not going to work solely in naloxone data, but instead we’re going to work in a dataset of our choosing. Think about a dataset that you know well and that you use in your EMT work. Read that in here (remember, you will need to adjust the code to read in whatever file type you have - I’m supplying sample code below for the most common types). Note: if you don’t have a dataset you actively work with, you can always use the naloxone data again.

```
#exceldata<-read_excel("FILEPATH/FILENAME.xlsx")
#csvdata<-read_csv("FILEPATH/FILENAME.csv")
#statadata<-read_dta("FILEPATH/FILENAME.dta")
#spssdata<-read_spss("FILEPATH/FILENAME.sav")
#sasdata<-read_sas("FILEPATH/FILENAME.sas7bdat")

#example - pulling response rate data from TN Together
respratedata<-read_dta("M:/334 - TN Together/000 - TN Together 2020-21 Administration/Analysis/output/respo

#note: you can also read in directly from websites, when the data is stored in a site. Instead of using a f
hsb_sas<-read_sas("https://stats.idre.ucla.edu/wp-content/uploads/2016/02/hsb2.sas7bdat")
```

Descriptives

First, let’s pull a summary of your entire dataset (throughout this file I’m going to use the example SAS dataset I read in above, named hsb_sas):

```
summary(hsb_sas)
```

```
##        id             female           race           ses            schtyp
##  Min.   :  1.00   Min.   :0.000   Min.   :1.00   Min.   :1.000   Min.   :1.00
##  1st Qu.: 50.75   1st Qu.:0.000   1st Qu.:3.00   1st Qu.:2.000   1st Qu.:1.00
##  Median :100.50   Median :1.000   Median :4.00   Median :2.000   Median :1.00
##  Mean   :100.50   Mean   :0.545   Mean   :3.43   Mean   :2.055   Mean   :1.16
##  3rd Qu.:150.25   3rd Qu.:1.000   3rd Qu.:4.00   3rd Qu.:3.000   3rd Qu.:1.00
##  Max.   :200.00   Max.   :1.000   Max.   :4.00   Max.   :3.000   Max.   :2.00
##       prog            read           write            math
##  Min.   :1.000   Min.   :28.00   Min.   :31.00   Min.   :33.00
##  1st Qu.:2.000   1st Qu.:44.00   1st Qu.:45.75   1st Qu.:45.00
##  Median :2.000   Median :50.00   Median :54.00   Median :52.00
##  Mean   :2.025   Mean   :52.23   Mean   :52.77   Mean   :52.65
##  3rd Qu.:2.250   3rd Qu.:60.00   3rd Qu.:60.00   3rd Qu.:59.00
##  Max.   :3.000   Max.   :76.00   Max.   :67.00   Max.   :75.00
##     science          socst
##  Min.   :26.00   Min.   :26.00
##  1st Qu.:44.00   1st Qu.:46.00
##  Median :53.00   Median :52.00
##  Mean   :51.85   Mean   :52.41
##  3rd Qu.:58.00   3rd Qu.:61.00
##  Max.   :74.00   Max.   :71.00
```

I also pulled a codebook for my dataset so that I know how categorical variables were coded. I’m pasting that here: *id = Student ID* gender = Student’s gender, with levels female and male *race = Student’s race, with levels african american, asian, hispanic, and white* ses = Socio economic status of student’s family, with levels low, middle, and high *schtyp=Type of school, with levels public and private* prog=Type of program, with levels general, academic, and vocational *read=Standardized reading score* write=Standardized writing score *math=Standardized math score* science=Standardized science score *socst=Standardized social studies score

Pull frequency tables for at least 2 categorical variables from your dataset.

```
table(hsb_sas$schtyp)
```

```
##
##   1   2
## 168  32
```

```
table(hsb_sas$race)
```

```
##
##   1   2   3   4
##  24  11  20 145
```

Limiting the data

Limit your dataset if you need to (e.g., maybe you just want to look at one time period). For my data today, I’m going to leave all observations in. But, I’m going to drop variables (columns) that I know I’m not going to use (id).

```
hsb_data<-hsb_sas%>%
    select(-id)
```

Conditional means

First, I want to examine an outcome variable of interest (academic performance in math) by some of my demographic groups to see if I have any differences. I'm going to do that by grouping by key demographic variables and then calculating means on my outcome measure. I'm not saving these metrics right now as I'm just examining them for potential differences.

```
#this would give me the mean reading score for every observation
mean(hsb_data$read)
```

```
## [1] 52.23
```

```
#First, looking at math, reading, writing, science, and social studies scores by gender
hsb_data%>%
    group_by(female)%>%
    summarize(meanmath=mean(math),
              meanread=mean(read),
              meanwrite=mean(write),
              meanscience=mean(science),
              meansoc=mean(socst))
```

```
## # A tibble: 2 x 6
##   female meanmath meanread meanwrite meanscience meansoc
##    <dbl>    <dbl>    <dbl>     <dbl>       <dbl>   <dbl>
## 1      0     52.9     52.8      50.1        53.2    51.8
## 2      1     52.4     51.7      55.0        50.7    52.9
```

It looks like there might be gender differences in writing and science.

Next, I'm going to look at each outcome by race.

```
hsb_data%>%
    group_by(race)%>%
    summarize(meanmath=mean(math),
              meanread=mean(read),
              meanwrite=mean(write),
              meanscience=mean(science),
              meansoc=mean(socst))
```

```
## # A tibble: 4 x 6
##    race meanmath meanread meanwrite meanscience meansoc
##   <dbl>    <dbl>    <dbl>     <dbl>       <dbl>   <dbl>
## 1     1     47.4     46.7      46.5        45.4    47.8
## 2     2     57.3     51.9      58          51.5    51
## 3     3     46.8     46.8      48.2        42.8    49.4
## 4     4     54.0     53.9      54.1        54.2    53.7
```

There do appear to be large variations across each subject assessment based on race.

Given that SES is categorical in this dataset (low, middle, high), I can also examine based on SES.

```
hsb_data%>%
    group_by(ses)%>%
    summarize(meanmath=mean(math),
              meanread=mean(read),
              meanwrite=mean(write),
              meanscience=mean(science),
              meansoc=mean(socst))
```

```
## # A tibble: 3 x 6
##     ses meanmath meanread meanwrite meanscience meansoc
##   <dbl>    <dbl>    <dbl>     <dbl>       <dbl>   <dbl>
## 1     1     49.2     48.3      50.6        47.7    47.3
## 2     2     52.2     51.6      51.9        51.7    52.0
## 3     3     56.2     56.5      55.9        55.4    57.1
```

Again, there do appear to be differences on scores based on SES.

Next I'm going to look at school type and then program type.

```
hsb_data%>%
    group_by(schtyp)%>%
    summarize(meanmath=mean(math),
              meanread=mean(read),
              meanwrite=mean(write),
              meanscience=mean(science),
              meansoc=mean(socst))
```

```
## # A tibble: 2 x 6
##   schtyp meanmath meanread meanwrite meanscience meansoc
##    <dbl>    <dbl>    <dbl>     <dbl>       <dbl>   <dbl>
## 1      1     52.2     51.8      52.2        51.6    52.0
## 2      2     54.8     54.2      55.5        53.3    54.8
```

```
hsb_data%>%
    group_by(prog)%>%
    summarize(meanmath=mean(math),
              meanread=mean(read),
              meanwrite=mean(write),
              meanscience=mean(science),
              meansoc=mean(socst))
```

```
## # A tibble: 3 x 6
##    prog meanmath meanread meanwrite meanscience meansoc
##   <dbl>    <dbl>    <dbl>     <dbl>       <dbl>   <dbl>
## 1     1     50.0     49.8      51.3        52.4    50.6
## 2     2     56.7     56.2      56.3        53.8    56.7
## 3     3     46.4     46.2      46.8        47.2    45.0
```

There appear to be differences by both school type and program type.

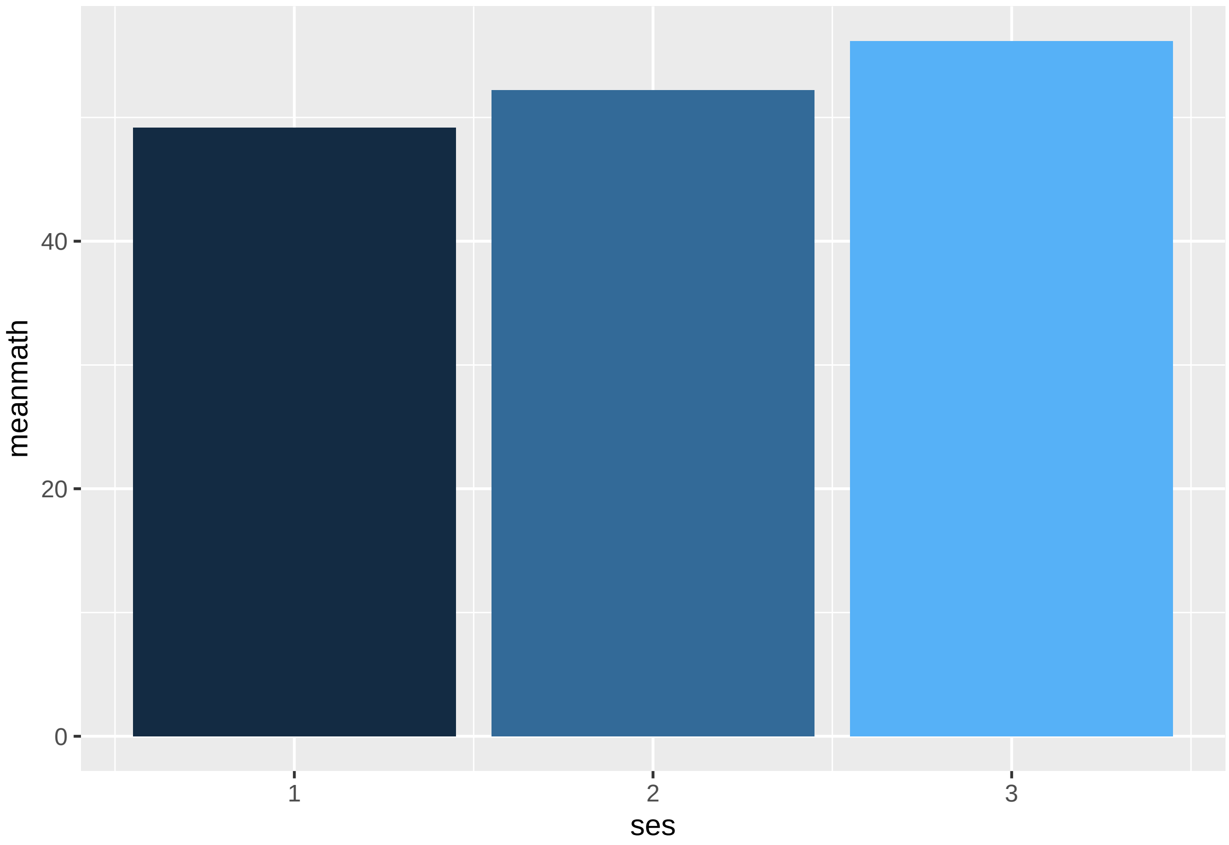
Plotting

Using your conditional means, go ahead and plot some of the interesting descriptive comparisons you found. Hint: if you didn't save any of your results above, you'll need to in order to plot them.

I'm going to plot mean math score by SES.

```
sesmathdat<-hsb_data%>%
    group_by(ses)%>%
    summarize(meanmath=mean(math))

sesmath<-ggplot(sesmathdat, aes(x=ses, y=meanmath, fill=ses))+
    geom_bar(stat="identity")+
    theme(legend.position="none")
sesmath
```


40
20
0
meanmath
1
2
3
ses