R Script for the Midterm (and beyond!)

TA: Melissa Gordon Wolf

214A: Fall 2019

Packages to load

Important: You have to INSTALL a package before you can load it. Once you've installed it, it will always be installed, but you'll need to LOAD it every time you open R. To install a package, type install.packages("package"). *For example:* install.packages("tidyverse") *will install tidyverse (note the quotations!). Again, you only need to do this once. However, you'll have to load the package every time you open R (see the commands below).*

```
library(tidyverse)
library(Hmisc)
library(psych)
library(knitr)
library(kableExtra)
library(sjmisc)
library(haven)
library(dplyr)
library(gmodels)
library(stargazer)
library(broom)
```

How to read in data

On the top right of RStudio, you will see three tabs: Environment, History, and Connections. Under Environment, you should see "Import Dataset". If you click on this and select "From SPSS", you'll be able to follow through the point and click menu to import the dataset you want to use. The code will pop up in the bottom right (I've copied and pasted it here) although you can just press okay. By default, R uses the "haven" package to import data from SPSS, which has some nice commands and options. To learn more, type "?haven" into your console of your r-script, and the package info will pop up in the bottom right.

```
library(haven)
GSS_health2010 <- read_sav("C:/Users/Melissa/Documents/UCSB/214/GSS_health2010.sav")

df<-GSS_health2010
```

How to get frequencies

```
#sjmisc package
frq(df$SEX)
```

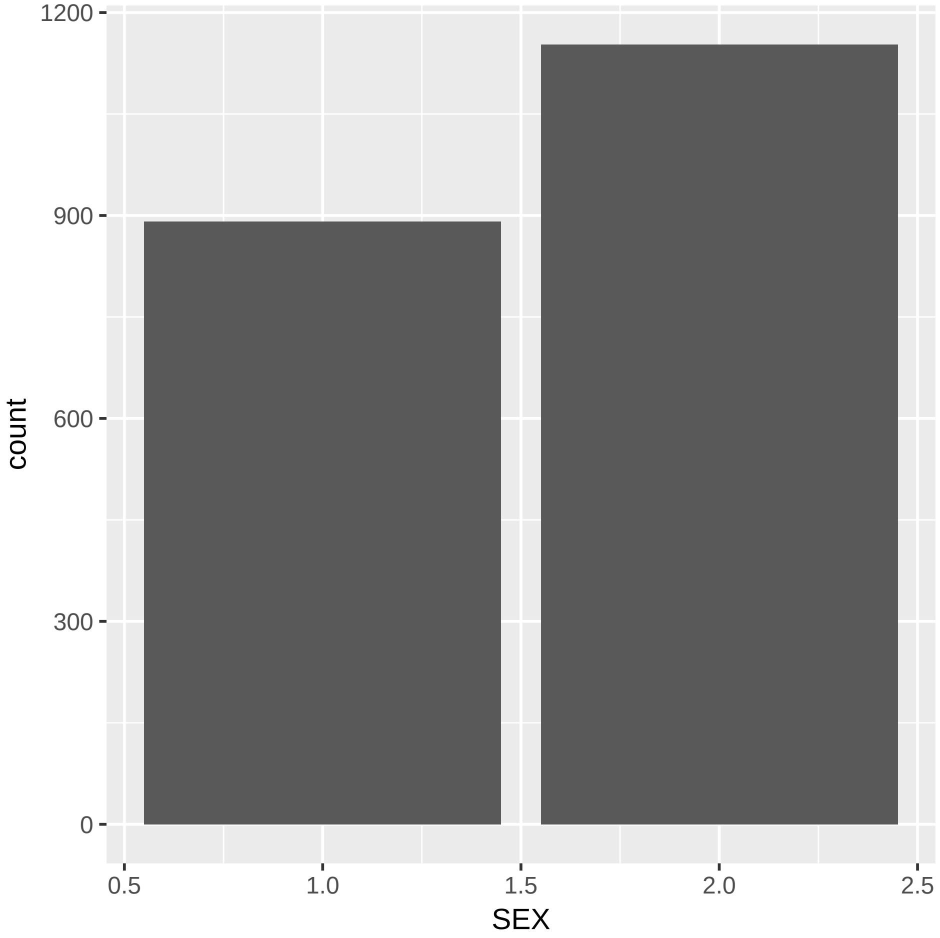
```
##
## RESPONDENTS SEX (x) <numeric>
## # total N=2044  valid N=2044  mean=1.56  sd=0.50
##
##  val  label  frq raw.prc valid.prc cum.prc
##    1   MALE  891   43.59     43.59   43.59
##    2 FEMALE 1153   56.41     56.41  100.00
##   NA   <NA>    0    0.00        NA      NA
```

```
#from dplyr package
df%>%
  count(SEX)%>%
  mutate(prop=prop.table(n))
```

```
## # A tibble: 2 x 3
##           SEX     n  prop
##     <dbl+lbl> <int> <dbl>
## 1 1 [MALE]     891 0.436
## 2 2 [FEMALE]  1153 0.564
```

How to create a bar graph of SEX

```
#using ggplot, tidyverse, and dplyr
df%>%
  ggplot(aes(x=SEX))+
  geom_bar()
```



```
#There are a lot of ways to make these graphs REALLY cool
#Check out: https://rstudio.com/wp-content/uploads/2015/03/ggplot2-cheatsheet.pdf
```

How to recode a variable

```
#Base R
summary(df$AGE)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   18.00   33.00   47.00   47.97   61.00   89.00       3
```

```
#sjmisc package
frq(df$AGE_R<-rec(df$AGE,rec="min:33=Young;
                 33:61=Middle;61:max=Gold;else=copy"))
```

```
##
## AGE OF RESPONDENT (x) <character>
## # total N=2044  valid N=2041  mean=1.99  sd=0.70
##
##     val  frq raw.prc valid.prc cum.prc
##    Gold  513   25.10     25.13   25.13
##  Middle 1038   50.78     50.86   75.99
##   Young  490   23.97     24.01  100.00
##    <NA>    3    0.15        NA      NA
```

How to summarize a continuous variable

```
#Base R
summary(df$AGE)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   18.00   33.00   47.00   47.97   61.00   89.00       3
```

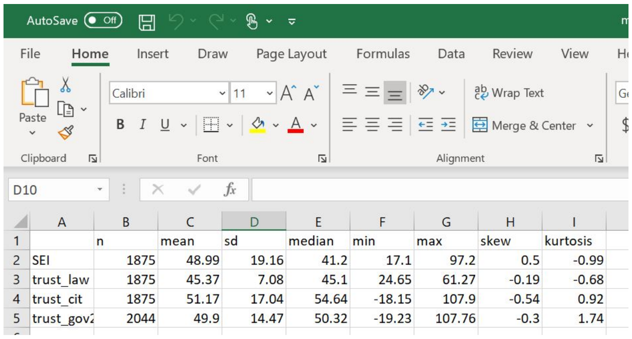
```
#psych package
describe(df$AGE)
```

```
##    vars    n  mean    sd median trimmed   mad min max range skew kurtosis
## X1    1 2041 47.97 17.68     47   47.18 20.76  18  89    71 0.29    -0.77
##      se
## X1 0.39
```

```
#make it look pretty!
#you'll need to do this to publish
dd<-describe(df$AGE)
class(dd)<-"data.frame" #need to change this to a dataframe to format it nicely

rdd<-round(dd, digits=2) #round to two digits

#your's won't look as pretty as mine becacuse I'm using latex, but...
kable(rdd, booktabs=T)%>%       #kable package
  kable_styling()               #kableExtra package
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
X1	1	2041	47.97	17.68	47	47.18	20.76	18	89	71	0.29	-0.77	0.39

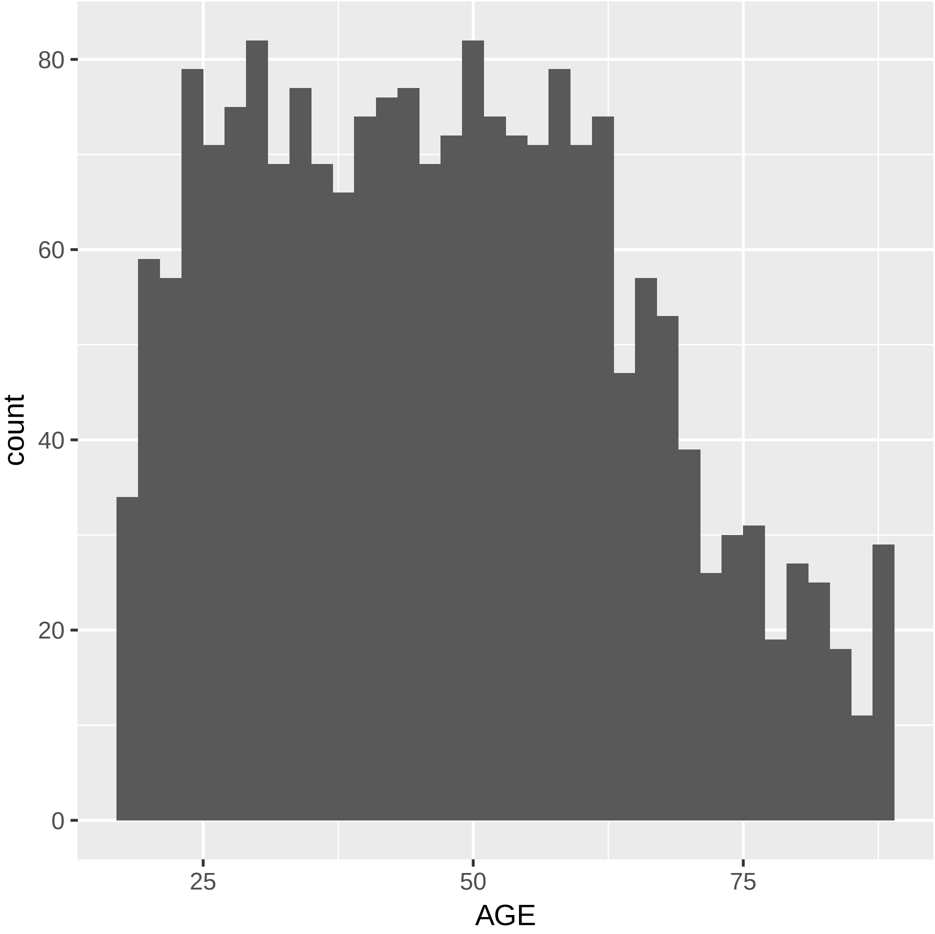
```
#alternatively, you can (should) export your table to a csv file.
#open it in excel, and make it pretty there, instead!
#you will find the file wherever your r script is saved

#have to comment out so that the pdf will print, but here it is:
#write.csv(rdd,file="myfile.csv")

#and, a screenshot!
```



	A	B	C	D	E	F	G	H	I
1		n	mean	sd	median	min	max	skew	kurtosis
2	SEI	1875	48.99	19.16	41.2	17.1	97.2	0.5	-0.99
3	trust_law	1875	45.37	7.08	45.1	24.65	61.27	-0.19	-0.68
4	trust_cit	1875	51.17	17.04	54.64	-18.15	107.9	-0.54	0.92
5	trust_gov2	2044	49.9	14.47	50.32	-19.23	107.76	-0.3	1.74

How to create a histogram of AGE

```
#using ggplot, tidyverse, and dplyr
df%>%
  ggplot(aes(x=AGE))+
  geom_histogram(binwidth=2)
```



How to create cross tabs or a table for two categorical variables

```
#Base R
table(df$SEX, df$RACE)
```

```
##
##          1    2    3
##   1 689 118  84
##   2 861 193  99
```

```
#gmodels
CrossTable(df$SEX,df$RACE, prop.chisq=FALSE)
```

```
## 
## 
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |         N / Table Total |
## |-------------------------|
## 
## 
## Total Observations in Table:  2044 
## 
## 
##              | df$RACE 
##       df$SEX |         1 |         2 |         3 | Row Total | 
## -------------|-----------|-----------|-----------|-----------|
##            1 |       689 |       118 |        84 |       891 | 
##              |     0.773 |     0.132 |     0.094 |     0.436 | 
##              |     0.445 |     0.379 |     0.459 |           | 
##              |     0.337 |     0.058 |     0.041 |           | 
## -------------|-----------|-----------|-----------|-----------|
##            2 |       861 |       193 |        99 |      1153 | 
##              |     0.747 |     0.167 |     0.086 |     0.564 | 
##              |     0.555 |     0.621 |     0.541 |           | 
##              |     0.421 |     0.094 |     0.048 |           | 
## -------------|-----------|-----------|-----------|-----------|
## Column Total |      1550 |       311 |       183 |      2044 | 
##              |     0.758 |     0.152 |     0.090 |           | 
## -------------|-----------|-----------|-----------|-----------|
## 
## 
```

```
#haven
print_labels(df$SEX)
```

```
## 
## Labels:
##  value  label
##      1   MALE
##      2 FEMALE
```

```
print_labels(df$RACE)
```

```
## 
## Labels:
##  value label
##      0   IAP
##      1 WHITE
##      2 BLACK
##      3 OTHER
```

```r
#Base R
xt<-xtabs(~df$SEX+df$RACE)
addmargins(xt)
```

```
##        df$RACE
## df$SEX    1    2    3  Sum
##    1    689  118   84  891
##    2    861  193   99 1153
##    Sum 1550  311  183 2044
```

```r
#Base R
prop.table(xt,1)
```

```
##        df$RACE
## df$SEX          1          2          3
##      1 0.77328844 0.13243547 0.09427609
##      2 0.74674761 0.16738942 0.08586297
```

```r
prop.table(xt,2)
```

```
##        df$RACE
## df$SEX         1         2         3
##      1 0.4445161 0.3794212 0.4590164
##      2 0.5554839 0.6205788 0.5409836
```

```r
#There are many more ways to create tables!
#You can export all of these to an excel file and make it pretty there
```

How to create cross tabs for a continuous variable and a categorical variable

```r
#Let's get the mean of age by race
#dplyr
#need to remove NA values or it won't work
#again, could export to excel to format nicely

df%>%
  group_by(RACE)%>%
  summarise(mean=(mean(AGE,na.rm=TRUE)))
```

```
## # A tibble: 3 x 2
##         RACE  mean
##   <dbl+lbl> <dbl>
## 1 1 [WHITE]  49.6
## 2 2 [BLACK]  44.5
## 3 3 [OTHER]  40.4
```

How to summarize the relationship between two continuous variables

Correlation

```r
#using Hmisc package
#gives you sample size and p-values as well!
rcorr(df$AGE,df$SEI)
```

```
##      x    y
## x 1.00 0.08
## y 0.08 1.00
##
```

```
## n
##      x    y
## x 2041 1873
## y 1873 1875
##
## P
##   x     y
## x       7e-04
## y 7e-04
```

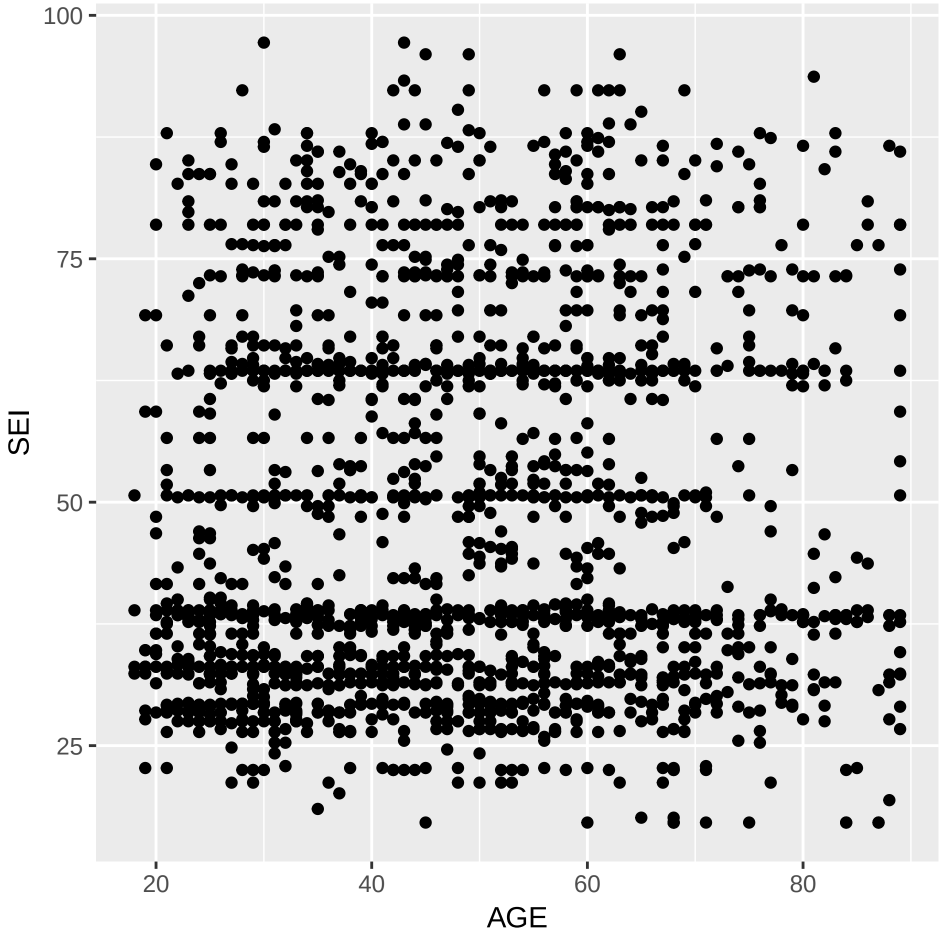
```
#let's see what this object looks like
rc<-rcorr(df$AGE,df$SEI)
str(rc)
```

```
## List of 3
##  $ r: num [1:2, 1:2] 1 0.0786 0.0786 1
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:2] "x" "y"
##   .. ..$ : chr [1:2] "x" "y"
##  $ n: int [1:2, 1:2] 2041 1873 1873 1875
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:2] "x" "y"
##   .. ..$ : chr [1:2] "x" "y"
##  $ P: num [1:2, 1:2] NA 0.000667 0.000667 NA
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:2] "x" "y"
##   .. ..$ : chr [1:2] "x" "y"
##  - attr(*, "class")= chr "rcorr"
```

```
#it has 3 components: r-correlation, sample size (n), and p-values
```

Scatterplot

```
#ggplot and dplyr
df%>%
  ggplot(aes(AGE,SEI))+
  geom_point()
```



Simple Linear Regression

```
#Base R
lm(AGE~SEI,data=df)
```

```
##
## Call:
## lm(formula = AGE ~ SEI, data = df)
##
## Coefficients:
## (Intercept)          SEI
##    45.02064      0.07053
```

```
#This gives us a LOT more info!
model<-lm(AGE~SEI,data=df)
summary(model)
```

```
##
## Call:
## lm(formula = AGE ~ SEI, data = df)
##
## Residuals:
## <Labelled double>: AGE OF RESPONDENT
##     Min      1Q  Median      3Q     Max
## -30.994 -14.306  -0.960  12.117  42.096
```

```
##
## Labels:
##  value       label
##     89 89 OR OLDER
##     98          DK
##     99          NA
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 45.02064    1.08888  41.346  < 2e-16 ***
## SEI          0.07053    0.02069   3.408 0.000667 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.15 on 1871 degrees of freedom
##   (171 observations deleted due to missingness)
## Multiple R-squared:  0.006171,   Adjusted R-squared:  0.005639
## F-statistic: 11.62 on 1 and 1871 DF,  p-value: 0.0006675
```

```r
#unfortunately, there is no easy way to write this out to excel
#this is the best we can do....
#broom package
write.csv(tidy(model),"model.csv")
write.csv(glance(model),"model1.csv")
```

```r
#this won't work for you, but look how pretty!
#if you're interested in this, you need to learn how to use latex and RMarkdown
stargazer(model, header=FALSE)
```

Table 1:

	Dependent variable:
	AGE
SEI	0.071***
	(0.021)
Constant	45.021***
	(1.089)
Observations	1,873
R^2	0.006
Adjusted R^2	0.006
Residual Std. Error	17.153 (df = 1871)
F Statistic	11.617*** (df = 1; 1871)
Note:	*p<0.1; **p<0.05; ***p<0.01