Frequencies & Central Tendency

Note: The dataset used in this tutorial and the R Script are on Moodle:

Loading the 2016 CCES dataset

```
install.packages("foreign", dependencies=TRUE)
library(foreign)

dat <- read.dta(file.choose(), convert.factors=FALSE)
```

Understanding the Structure of Variables and Recoding

1. Basic Exploration of Variables

Variables are almost never in a readily useable format. Usually, there are missing values and people that did not answer questions. Therefore, the first step to any analysis is exploring the structure of the variables you want to use and making sure that they are accurately depicting the measurement scales intended.

EXAMPLE: Let us pretend that the thing I ultimately want to explore is how Americans view themselves ideologically (Very liberal - 0 to middle of the road - 4 to very conservative - 7). In the survey, the codebook indicates that the variable "CC16_340a" provides this information. I can view a summary of this information by telling R that I want a summary of the CC16_340a variable in the "dat" dataset that I loaded with the command below. Note: The dataset I am loading is titled "dat" and the variable in the dataset is "CC16_340a".

```
summary(dat$CC16_340a)
```

Discussion: The output indicates that the minimum value for the variable is 1, median is 4, mean is 4.267 and max is 8. Does this make sense? No. Remember, the ideology variable is measured from 0-7. Therefore, we need to continue our exploration.

Example: You could also make a table of the variable in order to get a better view at the categories of the variable.

```
table(dat$CC16_340a)
```

Discussion: As you can see, there is the category 8 that do not fit in our 1-7 scale. If we look in the codebook, 8 indicates "not sure." In order to successfully move on and perform an analysis, we will need to recode this variable in order to remove the categories.

2. Recoding Variables - Numerical

In most instances, we might want to recode data so that these people do not exist in our analysis or "Not Applicable (NA)". For recoding variables, the "car" package provides us tool to do this type of recoding.

```
install.packages("car", dependencies=TRUE)
library(car)
```

Example: As indicated before, it is necessary to recode the categories/number "8" as NA so that we can perform meaningful statistical tests on the data.

```
dat$ideology <- recode(dat$CC16_340a, "8=NA")
summary(dat$ideology)
table(dat$ideology)
```

Discussion: After recoding the variable, the summary of the variable indicates that we now have a minimum value of 1, max of 7, mean of 4.054, and 3,735 observations have been set to NA. As you can see, the new mean makes much more sense theoretically as it is closer to the middle.

3. Recoding Variables - Categorical (nominal)

Another situation that might arise is that R might treat a categorical variable as numeric in ways that we do not wish. For example, a variable that explores profession might have several categories that cannot be ranked in a meaningful way numerically. Therefore, the variable should be recoded as a factor.

Example: For simplicity sake, here we are recoding the gender variable ("gender") to be categorical instead of numeric. In addition, if respondents refused to answer, we could do as we did above and recode other values to NA. The codebook indicates that for the variable a 1 indicates male and a 2 indicates female.

```
table(dat$gender)
dat$gender1 <- recode(dat$gender, "1='Man'; 2='Woman'")
dat$gender1 <- as.factor(dat$gender1)
table(dat$gender1)
```

Or, we could code it as numeric in a more meaningful way.

```
dat$gender2 <- recode(dat$gender, "1='0'; 2='1'")
dat$gender2 <- as.factor(dat$gender2)
table(dat$gender2)
```

4. Recoding Variables - Categorical (ordinal level)

With ordinal level variables you can leave them as numeric or recode them into factor variables. What is important is that you check the variable in order to make sure it makes sense. Here, we use the variables that asks how religious a respondent sees themselves as being (i.e. variables

"pew_religimp"). The variable is coded as 1 = very important to 4 = not at all important. We want to make sure to order the variables in a more meaningful way (i.e. switch the number ordering) and make sure that people skipping the question are coded as NA.

```
table(dat$pew_religimp)

dat$religiosity <- recode(dat$pew_religimp, "4=0; 3=1; 2=2; 1=3; else=NA")
```

Very Basic Descriptives

1. Nominal Level variables

Example: Again, let us use gender as an example. We can table the variable and get the number of women and number of women. In order to calculate percentages or proportions you will need to do a bit of math. When we table the variable we receive 29,531 men, 35,069 women.

```
table(dat$gender1)

prop.table(table(dat$gender1))
```

2. Ordinal level variables

Example: Let us use our religiosity variable we created. Say we wanted to know specifically how many respondents indicate that religion is not at all important. We could calculate the proportion and percentage as shown below.

```
table(dat$religiosity)

prop.table(table(dat$religiosity))
```

3. Interval/ratio level variables

You could see the minimum and maximum of the variable by using the summary command. However, measures of central tendency are better for describing interval level/ration variables.

Graphically Displaying Variables

1. Bar Chart

Example: For categorical variables, the best mode of presentation is usually providing a simple bar chart. Here, we provide a bar chart that presents the frequencies for our newly created gender variable.

```
barchart(dat$gender1, col="black")
```

2. Density Plot

Example: For continuous numerical variables, a density plot usually provides a better snapshot. Here, we create two density plots. One density plot for political ideology and the other for political ideology while splitting up the sample by gender.

```
histogram(dat$ideology, col="black", pch=2, xlab="Political ideology")

install.packages("ggplot2", dependencies=TRUE)
library(ggplot2)
ggplot(dat, aes(x=ideology)) + geom_density(aes(group=gender1, colour=gender1))
```

Measures of central tendency

1. The mean and median can be calculated for you using the summary command.

```
summary(dat$ideology)
summary(dat$religiosity)
```

2. We can also estimate measure of central tendency for one variable by a categorical variable.

```
tapply(dat$ideology, dat$gender1, summary)
tapply(dat$religiosity, dat$gender1, summary)
```

Lab Activity

In the 2020 Finland European Social Survey dataset, you are to find the variable labels for the four variables provided below assessing the state of democracy in Finland. Then, explore the four variables and recode them so that they are in a usable format. Finally, calculate the mean for each variable. Explain what the mean conveys substantively for each variable.

1. In country the media are free to criticise the government.

2. In country the courts treat everyone the same.

3. In country the rights of minority groups are protected.

4. In country takes measures to reduce differences in income levels.