Economics 1016
Summer 2020

Gregory A. Bruich, Ph.D.
Harvard University

R Hints for Empirical Project #1

Notes and commands that may be useful to you but are not necessarily required to answer the questions.

Set-up:

- Start by installing R and RStudio on Your Computer

  Download and install R: https://cran.r-project.org

  Download and install "RStudio Desktop:"

  https://www.rstudio.com/products/rstudio/download/

- Open the "marcps_w.dta" data file. You can use the drop down menu: file -> import data set -> from Stata. The browse and find the file on your computer. Pressing import will read the data as a new data frame using the haven package. Code similar to that shown below will appear:

  ```
  > library(haven)
  > marcps_w <- read_dta("marcps_w.dta")
  > View(marcps_w)
  ```

- The variable year is the survey year, but the labor supply variables refer to the previous year.

  ```
  > marcps_w$year <- marcps_w$year - 1
  ```

- Limit your analysis to observations on people ages 21-39, for example by subsetting the data to a new data frame called df:

  ```
  > df <- subset(marcps_w, age>=21 & age<=39)
  ```

- Create a new variable that is log(weekly earnings) by running the commands:

  ```
  > #Define log weekly wage income worked
  > df$lnwkwage <- log(df$wsal_val/df$wkswork)

  > #eliminate log(0) entries which are -Infinity
  > df$lnwkwage[which(df$lnwkwage==-Inf)] <- NA
  ```

Question 1:

Part (a) and (b)

Use `ggplot2` with the `stat_summary` to replicate the graph. See this page for more details.

Question 2:

The following commands will create the variable `post`, which takes the value '1' for observations after the ADA was implemented and '0' otherwise and the variable `disabl1_post`, the interaction term:

```
> df$post <- 0
> df$post[which(df$year >= 92)] <- 1

> #Generate interation term
> df$post_disabl1 <- df$disabl1*df$post
```

Use the command "lm" to estimate the regression. When running the regressions, use the sandwich and lmtest packages to report heterskedasticity robust standard errors. For example, to regress a hypothetical variable y on two hypothetical regressors x1 and x2, the commands would be:

```
> reg2 <- lm(y ~ x1 + x2, data=df)
> coeftest(reg2, vcov = vcovHC(reg2, type="HC1"))
```

Question 3:

Part (a)

Inside the lm command, you can use the factor() function to generate indicator variables for years 1988-1996. For example:

```
> reg2 <- lm(y ~ x1 + x2 + factor(year), data=df)
```

You can generate interaction terms between disability status and year using the semicolon. An example is below:

```
> reg2 <- lm(y ~ x1 + factor(year) + factor(year):disabl1, data=df)
```

Part (b)

To produce the coefficient plot after you have run your regression, you have to create a new data frame that contains the coefficients and their standard errors.

This is how you can do this. I start by making a vector that contains the years 1988 to 1996.

```
> years <- 1988:1996
```

I next extract the relevant coefficients on the interaction terms. R stores a vector of coefficients inside the lm object. For example, if I previously wrote, reg3 <- lm(…. then I can find the coefficients inside the reg3$coef vector:

```
> beta <- reg3$coef[12:20]
```

I next extract the relevant standard errors by taking the square root of the diagonal elements of th e variance co-variance matrix:

```
> se1 <-  sqrt(diag(vcovHC(reg3, type="HC1")))
> se <- se1[12:20]
```

I combine the years, coefficients, and standard errors in a new data frame called dfgraph:

```
> dfgraph = data.frame(years, beta, se)
```

I also like to add a zero in 1987, to help the reader know that 1987 is the base year.

```
> years <- 1987
> beta <- 0
> se <- 0
> df1987 = data.frame(years, beta, se)
```

I create a forgraph data frame that adds this extra row to the dfgraph data frame using rbind:

```
> forgraph <- rbind(dfgraph,df1987)
```

Now I add the 95% confidence intervals to the data frame:

```
> forgraph$ub <- forgraph$beta + (1.96*forgraph$se)
> forgraph$lb <- forgraph$beta - (1.96*forgraph$se)
```

The forgraph data frame now has the coefficients and the upper bound and lower bound on the 95% confidence interval.

You can now use ggplot to draw the graph with geom_point, geom_line, and geom_errorbar.