14.03/14.003 Recitation 3
Regressions and Econometrics with R

Andrea Manera[1]

Spring 2020

[1]I am grateful to previous TAs for sharing their slides on hypotheses testing, all errors are my own.

Agenda

- Getting familiar with RStudio
- Testing differences in means
- Regression analysis

RStudio: Resources

- RStudio is free and open source;
- There are many wonderful cheatsheets at `https://www.rstudio.com/resources/cheatsheets/`;
- A fantastic website showing applications to econometrics with code snippets: `https://www.econometrics-with-r.org` (which features also the Card-Krueger example we will look into today).

Environment

- R works with packages. These have to be loaded each time into the workspace with calls of the kind:

```
library('')
```

- If you do not have the package on your laptop, you can install any package calling:

```
install.packages('')
```

  with the correct name between quotes;
- To get help, you can always type a question mark followed by the command that you need help on;
- Indexing is done with matrix convention (similar to Python, Matlab, etc.) or with a dollar sign for *dataframes*, special data structures that R uses to store data.

Preliminaries: Working Directory and Libraries

- Define directory with a string

```
path='/Users/andreamanera/Dropbox (MIT)/Spring 2019/14-03 Spring 2019
    TA Folder/Recitations Spring 2019/Recitation 3 - RStudio/
    CardKrueger'
outputPath = paste(path,'/out', sep='')
dataPath = paste(path,'/data', sep='')
```

- Set working directory

```
setwd(path)
getwd()
```

Manipulating Data with dplyr and foreign

- Import data:

```
data<- read.dta(paste(dataPath, '/fastfood.dta', sep=''))
```

This command creates a dataframe in R, a matrix with named columns. Observations (the interviewed restaurants) will be on the rows, while the named columns will denote the variables of interest.

- Add variables defined as in CK: for them full time employment is a sum of employees, managers and .5 the part time employees

```
data <- mutate(data, FTE =  nmgrs + empft + (0.5 * emppt), FTE2 =
    nmgrs2 + empft2 + (0.5 * emppt2))
```

`mutate()` generates new data columns using existing ones

- Can use `subset()` to create smaller datasets of interest (e.g. all restaurants in NJ):

```
data_NJ <- data %>% subset(state==1)
```

Here we used the pipe operator `%>%` that tells R that it has to use data as an input for subset, i.e. `x %>% f(y)` is equivalent to writing `f(x,y)`.

Summarize Data

- Use the command `summarize()` that prints a table where the required statistics are stored. e.g. take the means before and after grouping by state:

```
data %>% group_by(state) %>%
   summarize(mean(FTE, na.rm=T),
             mean(FTE2, na.rm=T))
```

this prints:

```
  state `mean(FTE, na.rm = T)` `mean(FTE2, na.rm = T)`
<int>                    <dbl>                     <dbl>
1     0                   23.3                      21.2
2     1                   20.4                      21.0
```

The data reported in the paper. 0 and 1 are here the values of the grouping variable state, and recall that 1 denotes *NJ*.

- We could have computed the standard deviations using `sd` instead of `mean`.

How to compare means of two groups?

- Estimate mean for the treatment group

$$\bar{Y}_{X_i=1} = \frac{1}{N_T} \sum_{i:X_i=1} Y_i \longrightarrow_{N_T \to \infty} E[\,Y_i | X_i = 1]$$

- Estimate mean for the control group

$$\bar{Y}_{X_i=0} = \frac{1}{N_C} \sum_{i:X_i=0} Y_i \longrightarrow_{N_C \to \infty} E[\,Y_i | X_i = 0]$$

- Take the difference

$$\hat{d} = \bar{Y}_{X=1} - \bar{Y}_{X=0} \to E[\,Y_i | X_i = 1] - E[\,Y_i | X_i = 0]$$

  - ♢ Law of Large Numbers ensures that the estimates are arbitrarily close to their population means

t-Statistic

- Define a new statistic, the **t-statistic** as:

$$t = \frac{\left(\widehat{d} - d(=0)\right)}{\mathrm{se}\left(\hat{d}\right)}$$

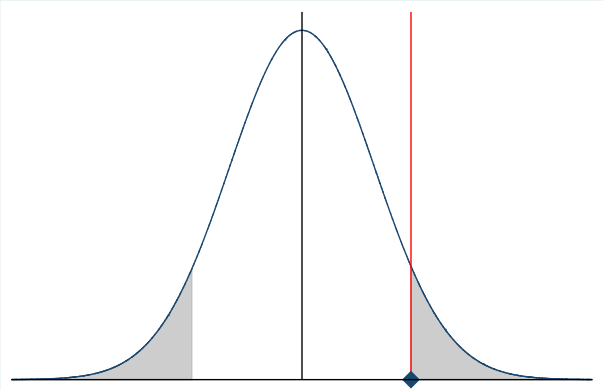
- This statistic is approximately normally distributed with mean 0 and standard deviation of 1
  - ♢ exactly normal when sample size is large

t-stat (cont...)

$$\textbf{t-stat: } t = \frac{\left(\widehat{d} - d(=0)\right)}{\text{se}\left(\widehat{d}\right)}$$

- tells us how many standard deviations far is our estimate from the null (zero)
- the further from the null it is, the less likely that we got a non-zero estimate just by random chance

t-stat and p-value



Hypothesis testing

- We want to test the **Null Hypothesis**: $H_0 : d = 0$ against the alternative that $H_a : d \neq 0$
  - The null means (in this case) that the mean of the treatment group is the same as the mean of the control group.
  - When treatment is randomly assigned, this is the causal impact of the treatment.
- Suppose that the true value is $d = 0$. But since we have a random sample, it could be that we estimate $\widehat{d} \neq 0$ because of statistical noise (or just by chance)
  - We want to minimize the chances that we falsely reject the null
- That is, we want to minimize **Type I Error**: reject the null when it is actually true
  - We want the probability of making a Type I Error to be low (how low?)

Statistical Significance

- To do so, typically adopt a decision rule where if a p-value falls below some threshold α we "reject" the null.
- Conventionally, **statistical significance** $\alpha = 0.05$ (5%). This ensures that we do not make Type-I error mistakes more than 5% of the time.
- Note the p-value is **not** the probability of a type-I error. That is, it is not the probability of *incorrectly rejecting* that the null is true.
- The true interpretation is: the probability that the value of the statistic we observe could be due to sampling error (assuming that Null is true). In formulas:

$$\text{p-value} = P(\text{Data}|\text{Null is True}) \neq P(\text{Null is True}|\text{Data})$$

- Interestingly, the probability that the Null is true given that the data gave a p-value of 5% can be quite high. See `https://faculty.washington.edu/jonno/SISG-2011/lectures/sellkeetal01.pdf` for more information.

Practically Testing for Difference in Means

- We will use a t-test to see whether the two mean are significantly different in a statistical sense. Our null hypothesis is that their difference is 0
- Carrying out a t-test on R is done through the command `t.test()`. Example: is actual mean number of employees in NJ before the change equal to 20?

```
1 t.test(data_NJ$FTE, mu=20)
```

Turns out it is a good hypothesis *we cannot reject*; The first naive test of the effect of the minimum wage is to see whether there was a significant change in NJ employment after the introduction. To do this we can just compute:

```
1 t.test(data_NJ$FTE,data_NJ$FTE2, mu=0)
```

where with `mu=0` we are now testing the hypothesis of no change. It turns out we cannot reject this hypothesis either (the p-value is .42!)

- But this does not measure anything... other confounding variables might have changed in the meanwhile!
- **Diff-in-diff is a potential solution!**

Diff-in-Diff through a simple t-test

- To see how it changed before and after let us build some new variables:

```
data %>% mutate(dFTE = FTE2-FTE)
t.test(data_NJ$dFTE, data_PA$dFTE, mu=0, var.equal = T)
```

- The value of t is very high (larger than the "magic" 2), so we can reject the hypothesis that the difference is 0, in favor of a positive effect of the minimum wage;
- `var.equal` tells R to use the all observations to compute the variance instead of treating the two sets of observations for PA and NJ as coming from different samples. (see commands in Slide with Regression)
- This replicates Table 3, column (iii) in CK (1994).

Means

- We can also accomplish similar objectives using regression
- Suppose you have data (y_i) that you are interested in
- You can always write this as

$$y_i = \alpha + \varepsilon_i$$

where $E[\varepsilon_i] = 0$, then

$$E[y_i] = \alpha$$

- Suppose you wanted to see the means by two groups $X_i = 1$ and $X_i = 0$
- Similarly you can write, for $X_i = 0$

$$y_i = \alpha + \varepsilon_i \text{ with } E[\varepsilon_i | X_i = 0] = 0$$

and for $X_i = 1$

$$y_i = \alpha + \beta + \eta_i \text{ with } E\big[\eta_i | X_i = 1\big] = 0$$

Comparing means in two group

- Combining, we get

$$y_i = \alpha + \beta X_i + \epsilon_i$$

  with $E[\epsilon_i | X_i] = 0$
- Since $E[\varepsilon_i | X_i = 0] = E\left[\eta_i | X_i = 1\right] = 0$, we have

$$\begin{aligned} E[y_i | X_i = 1] &= \alpha + \beta \\ E[y_i | X_i = 0] &= \alpha \end{aligned} \implies E[y_i | X_i = 1] - E[y_i | X_i = 0] = \beta$$

- How do we implement this?

Comparing means in multiple groups

- Can similar think of the situation where there are many groups. Let $X0_i, X1_i, X2_i, \cdots, Xk_i$ indicate the $k+1$ groups. Then can write

$$y_i = \alpha + \beta_1 X1_i + \beta_2 X2_i + \cdots + \beta_k Xk_i + \varepsilon_i$$

  with $E[\varepsilon_i | X_i] = 0$. (note the omission of $X0_i$)

- Then

$$\begin{aligned}
\beta_1 &= E[y_i | X1_i = 1] - E[y_i | X0_i = 1] \\
\beta_2 &= E[y_i | X2_i = 1] - E[y_i | X0_i = 1] \\
&\vdots \\
\beta_k &= E[y_i | Xk_i = 1] - E[y_i | X0_i = 1]
\end{aligned}$$

- How do we compare group j with group k?
- How do we estimate the βs?

Regression

- Regression can estimate the following relationships

$$y_i = \alpha + \beta X_i + \varepsilon_i$$

$$y_i = \alpha + \beta_1 X1_i + \beta_2 X2_i + \cdots + \beta_k Xk_i + \varepsilon_i$$

  with $E[\varepsilon_i | X_i] = 0$
- Ordinary Least Squares (OLS) is the best linear unbiased estimator of coefficients
- Can then conduct many hypothesis tests of interest

Testing Treatment

- One of the most common versions we're interested in is

$$y_i = \alpha + \beta X_i + \epsilon_i$$

  where X_i is an indicator variable for being treated.
- As noted this then gives us $\beta = E[Y|X=1] - E[Y|X=0]$ or the treatment effect.

Testing linear relationships

- We can also use OLS to test whether two variables of economic interest are related to each other
- We have our data in two variables

$$y = (y_1, y_2, ..., y_N)$$

$$x = (x_1, x_2, ..., x_N)$$

  - y is the dependent variable (say FTE employment)
  - x is the explanatory variable/ exogenous variable (say a dummy for "minimum wage introduced")

- Write the simplest relationship between the two variables

$$E[y|x] = a + bx$$

- Best linear approximation of $E[Y|X]$

Practically Using Regressions for CK (1994)

- Running Regressions can be done specifying a model using the command `lm` (which stands for "linear model").
- First reorganize the data to make it "long" from its current "wide" form

```
#Create dataframe with employment before the change
dataBefore <- data.frame(id = data$sheet,
chain = data$chain,
state = data$state,
empl = data$FTE) %>%
mutate(after = 0)
#Create dataframe with employment after the change
dataAfter <- data.frame(id = data$sheet,
chain = data$chain,
state = data$state,
empl = data$FTE2) %>%
mutate(after = 1)
#Concatenate vertically
dataReg <- data.frame(rbind(dataBefore, dataAfter))
```

Step 1: "Naive Regression"

- Run a regression on data from NJ only before and after the introduction of the minimum wage. Our model is:

$$Empl_{i,t} = \alpha + \beta \mathbf{1}(W_{min} \text{ introduced for } i)$$

(i denotes the restaurant).

- In R:

```
### Specify the naive model of interest
model <- lm(formula = empl ~ after,
data =dataReg,
subset =
(dataReg$state == 1))
```

- `formula` denotes our model, constant α is always included by default. `subset` allows us to run the regression on a subset.

Step 1.1: Testing Coefficients

- Testing in R takes as input a model of the kind we defined before. The function to get diagnostics is `coeftest` in the package `AER` (so we need to call `library(AER)` before). To see diagnostic for the naive regression:

```
coeftest(model, vcov. = vcovHC, type = "HC1")
```

  Options are technical (if you have heard of heteroskedasticity, these are used to get heteroskedasticity-robust standard errors).
- Output:

```
t test of coefficients:

Estimate Std. Error t value Pr(>|t|)
(Intercept) 20.43941    0.50826 40.2142   <2e-16 ***
after        0.58802    0.72736  0.8084   0.4191
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘’. 0.1 ‘’  1
```

  We see there is no statistical significance on “after”. What can we conclude?
- Actually, nothing at all!

Step 2: The Regression in Recitation

- Run a regression on data from both states. Our model becomes:

$$Empl_{i,t} = \alpha + \beta_1 \mathbf{1}_{i,t}(W_{min} \text{ introduced}) + \beta_2 \mathbf{1}_{i,t}(\text{State = NJ}) +$$
$$+ \beta_3 \mathbf{1}_{i,t}(W_{min} \text{ introduced}) \mathbf{1}_{i,t}(\text{State = NJ})$$

- β_3 is the diff-in-diff coefficient.

```
### Specify the CK model of interest and get diagnostics
modelCK <- lm(formula = empl ~ state*after,
data =dataReg)
coeftest(modelCK, vcov. = vcovHC, type = "HC1")
```

- `formula` denotes our model, constant α is always included by default. `subset` allows us to run the regression on a subset.

Step 3: The Card-Krueger Regression

- We can take differences over time so that the model

$$Empl_{i,t} = \alpha + \beta_1 \mathbf{1}_{i,t}(W_{min} \text{ introduced}) + \beta_2 \mathbf{1}_{i,t}(\text{State} = \text{NJ}) + \\ + \beta_3 \mathbf{1}_{i,t}(W_{min} \text{ introduced}) \mathbf{1}_{i,t}(\text{State} = \text{NJ})$$

  becomes (convince yourself taking the differences along the time dimension):

$$\Delta Empl_{i,1} = Empl_{i,1} - Empl_{i,0} = \beta_1 + \beta_3 \mathbf{1}_{i,1}(\text{State} = \text{NJ})$$

- β_3 is the diff-in-diff coefficient.

```
### Specify the CK model of interest and get diagnostics
modelDiff = lm(formula = dFTE ~ state, data = data)
coeftest(modelDiff)
```

- You will see that the t-tstat is the same as the one we obtained at the beginning (rerun the command in Slide on t-test)

Step 3: Replicating Table 4 in CK (1994)

- Note that CK do not get 2.75 as coefficient in Table 4, column 1, but 2.33. This is because they select the sample to include only observations with wages in both years
- To carry out the sample selection, we need to drop observations that are `NA`. This is achieved as follows:

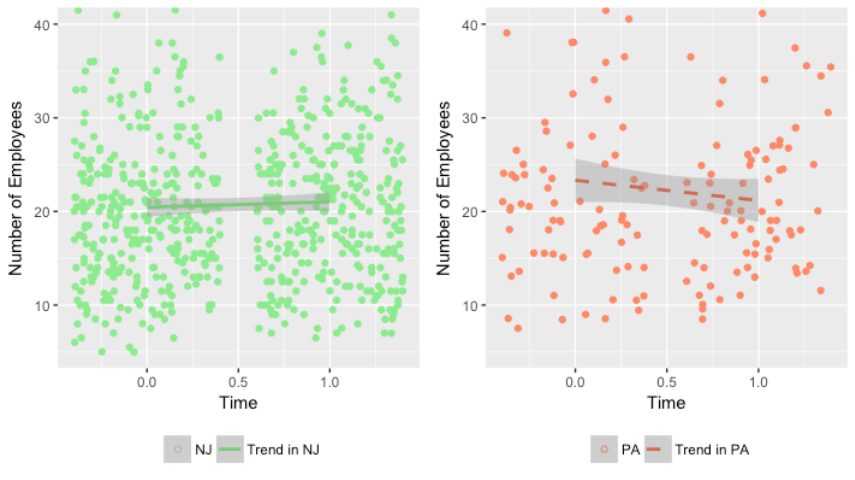
```
dataAllWages = data[complete.cases(
data.frame(data$wage_st,data$wage_st2, data$FTE, data$FTE2)),]
```

- `complete.cases` identifies the rows of the dataset in parentheses that contain *no missing values*. This produces logical indexes that are then used to select the rows of `data` (convince yourself this works).

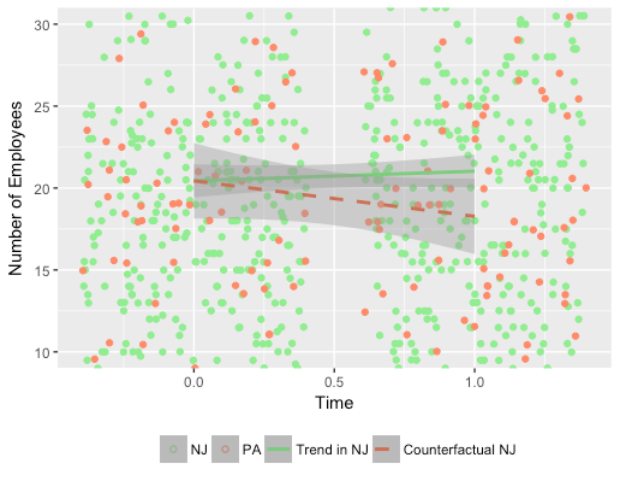
```
modelDiff = lm(formula = dFTE ~ state, data = dataAllWages)
coeftest(modelDiff)
```

  produces a coefficient of 2.28, very close to the 2.33 obtained by CK, and the same standard error. The difference is explained by the sample: if we follow their indications we get a sample of 359 observations, not 357 as they say...
- Many times economic papers do not exactly replicate!

Extra: Diff-in-Diff with graphs I



Extra: Diff-in-Diff with graphs II



Extra: Diff-in-Diff with graphs III