R Applications for Multiple Regression Model: Estimation

Introductory Remarks

This file builds on the previous R Activities. If you have not yet gone through those files, please make sure to do so before tackling this R Activity.

Loading the Relevant Dataset

Before looking loading the relevant dataset, make sure to clear your work environment in R, install and/or load the relevant packages, and set your working directory. The accompanying R script does this step by step and a detailed description of how to do this was provided in the R introduction.

The dataset we will work with today is called "CARD.DTA" and is available on the course website. A ".dta" extension means that the dataset is saved in "Stata format." In case you don't know, Stata is another software often used for statistical analyses. To load the dataset into R, first make sure that you install and load the "foreign" package. Then, type the following command

```
data <-read.dta("CARD.dta") # loading the CARD data
```

The dataset should now appear in your environment under the name "data."

To get a sense of what this data tells us, let's explore it using three already familiar commands, namely

```
View(data) # look at data
head(data) # only display first few rows
summary(data) # summarize data
```

Notice that the data has 34 variables and 3010 observations. We will only care about a few of these, namely "wage," "educ," "IQ," "black," and "exper." "wage" denotes the hourly wage in cents in 1976 of any individual (individuals are indexed by the variable "id" in the dataset), "educ" denotes the years of schooling, "IQ" denotes the IQ score, "black" is a dummy variable (i.e. a variable that is either 0 or 1) indicating if an individual is black ("black=1") or not ("black=0"), and "exper" denotes the work experience. Note that work experience is calculated by taking the person's age (the variable "age" in the dataset), deducting "educ" (i.e. the years of schooling) and deducting 6. Thus, the measure of work experience we use here is an approximation of people's true work experience. We call this a proxy variable.

To make things clearer, let's get rid of all the variables in the dataset we don't need. We can do this as follows

```
data_for_analysis <- subset(data, select = -c(nearc2,nearc4, age, fatheduc, motheduc,
                                              weight, momdad14, sinmom14, step14,
                                              reg661, reg662, reg663, reg664,
                                              reg665, reg666, reg667, reg668,
                                              reg669, south66, smsa, south,
                                              smsa66,enroll, KWW, married,
                                              libcrd14, lwage,expersq))
```

The command we use for this is "subset," which is a command that tells R to subset "data" (our dataset) by selecting whatever variables we specify. If you write a minus like we did above, this tells R to delete these variables. We have thus created a new dataset with six variables (the five described above plus we keep the "id" variable to be able to identify individuals) called "data_for_analysis." You should see it in your environment. Note that the original dataset "data" is also still there. While we could theoretically work with both datasets now, we will only focus on the new one.

Bivariate Regressions

Taking logs. Suppose we are interested in the following relationship

$$log(wage)_i = \beta_0 + \beta_1 educ_i + \epsilon_i$$

In words, we want to regress log wages on education to inquire about the relationship between the two variables. Technically, this is a simple bivariate regression, which we already know how to implement in R, with one tiny twist. The outcome here is in logs, but the variable in the dataset is not. Hence, before running this OLS regression, we need to convert the "wage" variable into a log wage ("lwage") variable. We can do this as follows

```
data_for_analysis$lwage <- log(data_for_analysis$wage) # create log wages
```

The left-hand side of the command tells R to create a new variable called "lwage" in the dataset "data_for_analysis" and the right-hand side of the command then tells R what values to assign to this variable, i.e. the log of the wages. Some of you will notice that the original dataset "data" actually already included a variable called "lwage." In reality, we could of course use this variable directly for our analysis. The aim here was to show you how to create such a variable. You can check for yourself that the two "lwage" variables in the two datasets are the same.

Running this regression. We can now run the above regression using the same "lm" command we learned about in "R Activity 1". Specifically, we run

```
bivariate_OLS <- lm(lwage ~ educ, data=data_for_analysis) # run OLS regression
summary(bivariate_OLS) # summary of results
```

As we can see from the summary of the results, we get an estimated intercept of 5.571 and an estimated coefficient on education of 0.052. The interpretation of this coefficient is therefore that an additional year of schooling (i.e. a one unit increase in "educ") is associated with a 5.2 percent increase in wages. As an exercise, try to plot the residuals using the codes introduced in "R Activity 1".

R-squared. Something we haven't looked at so far in R is the R-squared. As you can see in the second to last line of the summary output, R directly provides you with this statistic. To display it you could also type

```
summary(bivariate_OLS)$r.squared # get R squared
summary(bivariate_OLS)$adj.r.squared # get adjusted R squared
```

The R-squared from this regression is 0.0987 and the adjusted R-squared is 0.0984.

Multivariate Regressions

Adding one control. Suppose now that a friend of ours argues that they are worried the coefficient we estimated above on education is biased. They argue that they are worried that a confounder exists that is correlated with both education and log wages. As an example, they suggest work experience. After considering their argument, we come to the conclusion that our friend is right: work experience almost certainly correlates with education and log wages, and therefore our coefficient from above is biased. We know that we can address this issue by including a control variable into the above regression.

While we don't have data on work experience per se, we do have a variable called "exper," which, we argue, approximates work experience reasonably well. As mentioned above, we call this a proxy variable. Hence, we now aim to run a multivariate regression

$$log(wage)_i = \gamma_0 + \gamma_1 educ_i + \gamma_2 exper_i + v_i$$

Doing this in R is a very straightforward extension of the bivariate regression we ran above. Specifically, we run the following command

```
multivar_OLS <- lm(lwage ~ educ + exper, data=data_for_analysis) # run OLS regression
summary(multivar_OLS) # summary of results
```

As you can see, we still use the "lm" command to run this regression. The only real difference to the bivariate case is that we just add the control after the regressor of interest.

Looking at the summary of the output, we find the following coefficients: $\hat{\gamma}_0 = 4.67$, $\hat{\gamma}_1 = 0.093$, and $\hat{\gamma}_2 = 0.041$. The one we care about is the coefficient on education. It tells us that an additional year of schooling is associated with a 9.3 percent inrease in wages, controlling for work experience. Note that the effect is larger here than it was in the bivariate regression. Why?

The fact that the coefficient on education changes when we include work experience as a control variable leads us to conclude that the estimate on education in the bivariate regression was (downward) biased. To see why this is, recall the omitted variable bias formula from the lecture

$$\hat{\beta}_1 = \hat{\gamma}_1 + \hat{\gamma}_2 \hat{\pi}$$

where $\hat{\pi}$ is the OLS estimate from a regression of *exper* on *educ*. The bivariate and multivariate regressions we ran provide us with three of these coefficients, i.e.

$$.052 = .093 + .041\hat{\pi}$$

Solving for $\hat{\pi}$ we get $\hat{\pi} = -1$. This implies, that in a regression of *exper* on *edu*, the coefficient on *educ* should be -1. We can test this by running the following auxilliary regression

```
aux_reg <- lm(exper ~ educ, data=data_for_analysis) # run OLS regression
summary(aux_reg)
```

As you can see from the output, the coefficient on education is indeed -1. This should make sense given the definition of work experience discussed above. Recall that work experience is calculated by taking the person's age (the variable "age" in the dataset), deducting "educ" (i.e. the years of schooling) and deducting 6. Looking at this formula, it should be clear that as we increase education by one unit, we reduce work experience also by one unit. While the relationship here is mechanical because of the way that work experience is defined, it does make intuitive sense that education and work experience are negatively correlated.

The fact that the estimate on education from the bivariate regression is downward biased compared to the estimate on education from the multivariate regression should now make sense. The positive correlation between work experience and log wages as well as the negative correlation between work experience and education explain the bias as shown by the omitted variable bias formula.

Adding a second control. Let us now add a second control into the regression. Specifically, let us add our proxy variable for ability, i.e. we add IQ and aim to run this multivariate regression

$$log(wage)_i = \gamma_0 + \gamma_1 educ_i + \gamma_2 IQ_i + \gamma_3 exper_i + v_i$$

Doing this in R is again very straightforward

```
multivariate_OLS <- lm(lwage ~ educ + IQ + exper, data=data_for_analysis,
                       na.action=na.exclude) # run OLS regression
summary(multivariate_OLS) # summary of results
```

As you can see, we still use the "lm" command to run this regression and just add the additional control. Note one other difference: we include the option of "na.action = na.exclude." We do this because our new control variable has missing observations. To see this, look at the IQ variable. You'll sometimes see a "NA" which indicates a missing observation. Notice, that if our "lwage," "educ," and/or "experi" variables had any missing observations, we would've added this option in the bivariate/multivariate regressions above as well. Hence, this missing values issue isn't specific to this multivariate regressions. Furthermore, be aware that R automatically drops all missing observations in a regression. To see this, try to run the same command as above but exclude the "na.action=na.exclude" option and you will get the same result. It is, however, good

practice to include option as it will force you to think about what observations you drop from your sample and possible selection issues.

Looking at the summary of the output, we find the following coefficients: $\hat{\gamma}_0 = 4.538$, $\hat{\gamma}_1 = 0.068$, $\hat{\gamma}_2 = 0.005$, and $\hat{\gamma}_3 = 0.045$. The one we care about is the coefficient on education. It tells us that an additional year of schooling is associated with a 6.8 percent inrease in wages, controlling for IQ and experience. Notice that this effect is somewhere in between the two estimates we found above.

R_squared. To find the R-squared and adjusted R-squared of this multivariate regression, we can type

```
summary(multivariate_OLS)$r.squared # get R squared
summary(multivariate_OLS)$adj.r.squared # get adjusted R squared
```

We find a R-squared of 0.166 and an adjusted R-squared of 0.164. Notice that the R-squared cannot decrease with the inclusion of new variables. Thus, the increase we observe in the R-squared compared to the one we found in the bivariate regression is to some extent at least mechanical. The adjusted R-squared doesn't suffer from this issue. Thus, the fact that we observe a similar increase in the adjusted R-squared as we do in the R-squared, is a promising sign that this multivariate regression is edging closer to explaining more variation of our outcome variable, i.e. log wages.

FWL Theorem. In the lecture slides, you learned about the FWL theorem. In short, this theorem, applied to our running example, says that whether we run the multivariate regression as above leads to the same coefficient on education as if we (i) run an OLS regression of education on our controls and (ii) run an OLS regression of log wages on the residual obtained from the regression in (i). We already ran the multivariate regression above, so we can here implement the two step procedure to see if we get the same coefficient on education. To implement the procedure, run the following code

```
step1 <- lm(educ ~ IQ + exper, data=data_for_analysis,na.action=na.exclude) # first step OLS
data_for_analysis$step1_residuals <- residuals(step1) # get residuals for step 1
step2 <- lm(lwage ~ step1_residuals, data=data_for_analysis,na.action=na.exclude) # second step OLS
summary(step2)
```

The first two lines of the code execute step (i) of the FWL procedure. First, we run an OLS regression of education on IQ and experience. Second, we obtain the residuals from this OLS regression and store them in our dataset "data_for_analysis." The third line of the code executes step (ii) of the procedure by running an OLS regression of log wages on the residuals obtained in step (i). Lastly, we summarize the final result.

Looking at the output, we find a coefficient on the residuals of step (i) of 0.068. The FWL theorem tells us that this coefficient should be identical to the one we found on education in the multivariate regression. As you can see, this is clearly the case. The interpretation is therefore also the same as above.

Multicollinearity

Consider extending the above model to the following

$$log(wage)_i = \gamma_0 + \gamma_1 educ_i + \gamma_2 IQ_i + \gamma_3 exper_i + \gamma_4 black_i + v_i$$

In words, we extend the model to include a dummy variable indicating whether an individual is black or not. Running this model in R is analogous to what we did above, i.e.

```
multivariate_OLS2 <- lm(lwage ~ educ + IQ + exper + black, data=data_for_analysis,
                        na.action=na.exclude) # run OLS regression
summary(multivariate_OLS2) # summary of results
```

You can try to interpret the coefficients yourself. As you can see, the coefficient on education increases by a little bit, but not by much.

Now, consider creating a new variable called "nblack" which is a dummy variable that equals to 1 if an individual is not black and equal to 0 if an individual is black. In other words, it is "the opposite" of the variable "black." To create this variable, type

```
data_for_analysis$nblack = 1 - data_for_analysis$black
```

Clearly, if we were to also add this variable into our multivariate regression we would run into an issue of multicollinearity. Intuitively, the variables "black" and "nblack" measure the same thing: they just indicate whether someone is black or not. When running a regression, it is important to think about whether two variables are perectly correlated (as in our example here) or highly correlated because this can lead to issues in our regressions. To see what R does in this extreme case, type

```
multivariate_OLS3 <- lm(lwage ~ educ + IQ + exper + black + nblack, data=data_for_analysis,
                        na.action=na.exclude) # run OLS regression
summary(multivariate_OLS3) # summary of results
```

As you can see, R automatically omits the new regressor. Recall that our regression contains Note that in this case R does this because "black" and "nblack" measure virtually the same thing (they are perfectly correlated). Hence, R drops them. If we were to include another variable that is very highly correlated with one of our right-hand side variables, then R would not drop the variable. Yet, we might still run into multicollinearity issues. It is therefore pertinent that you think very carefully about what variables you include on the right-hand side of your model.

Notice that in the above we have the issue of perfect collinearity only because our regression includes an intercept. If we were to drop the intercept from the regression, R will estimate a coefficient for both variables. To see this you can type

```
multivariate_OLS4 <- lm(lwage ~ educ + IQ + exper + black + nblack -1, data=data_for_analysis,
                        na.action=na.exclude) # run OLS regression
summary(multivariate_OLS4) # summary of results
```

Standard Errors and Statistical Significance

Most of you by now will have noticed that so far we have completely ignored most of the summary output except for the OLS estimates. Indeed, if you type

```
summary(multivariate_OLS) # summary of results
```

you will see that R provides you with the standard errors associated with each coefficient *under the assumption of homoskedasiticty*. Moreover, R even provides t-statistics and p-values for each of your estimates. We will discuss the meaning and interpretation of these in the next R application.

Causality

Lastly, pay attention not to give any causal interpretations to anything we have done so far. It should be straightforward to see that there are likely other confounders that are biasing our results. Hence, as an exercise, you could try to add more controls to our above regressions. This might not solve our confounders problem (since there may always be some unobservables that act as confounders), but it can reduce it at least. We will, in due time, learn other methods that can get us closer to a causal effect.