Multivariate Linear Regression II

Manu Navjeevan

8/30/2021

Lecture's Objective

The goal for this lecture is to cover some additional topics concerning multivariate regression in R. We will start by learning how to create and use dummy variables in R, and then proceed to test hypotheses about multiple coefficients. We will be using some of the results and formulas that we covered in the main lecture of Econ 103 along the way. Much of our discussion is based on "*Principles of Econometrics with R" by Constantin Colonescu (available for free at https://bookdown.org/ccolonescu/RPoE4/)

Dummy Variables

It is often helpful to create dummy variables for our regressions. These are variables that only take values 0 and 1, with 1 indicating the observation of satisfies some property of interest. Before getting started, let's load the cps dataset again.

```
# Clear the workspace
rm(list = ls())

# Our data set is store as part of the PoEData library, so let's tell R we need access to it
library(PoEdata)

# Looking at the output, there are multiple datasets called CPS. We'll use cps.
data(cps)

# Remind ourselves what variables are contained in cps dataset
#attributes(cps)
```

Suppose that we want to estimate how expected log wages depend on education. We conjecture that college graduation is particularly important so we want to create a dummy variable such that

$$D = \begin{cases} 1 & \text{if edu } \geq 16 \\ 0 & \text{if edu } < 16 \end{cases}$$

Ideally, we would like to know whether individual graduated college, but we don't have that information. Instead, we are therefore proxying for college graduation with someone having 16 (or more) years of education. How do we create the variable D? A helpful way to do this is to use boolean logic. The idea behind this is that, in R, one times true = 1 and one times false = 0.

Let's start by generating a true false list of whether the education variable is greater than or equal to 16

```
print(cps$educ[1:10])
```

```
##  [1] 12 13  8 10 18 12 13 12 13 16
```

```
# (cps$educ >= 16) returns a vector that has a TRUE entry if education is larger than 16 and a FALSE en
print((cps$educ >= 16)[1:10])
```

```
##  [1] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE
```

We see this returns a vector that which has a “TRUE” entry if, for that particular, years of education is greater than 16 and “FALSE” entry otherwise. To generate

```
# If we multiply this by one, we will get the dummy variable that we want
D <- 1*(cps$educ >= 16)
```

Now that we have created the desired dummy variable, we can run the regression

$$\log(\text{wage}) = \beta_0 + \beta_1 \text{edu} + \beta_2 D + \epsilon$$

to get a sense of whether there is an extra return from graduating college. We can run such regression by using the `lm` command in the usual way.

```
#  Create a log wages variable
logwage <- log(cps$wage)

# Run the desired regression
regout <- lm(logwage ~ educ + D, data = cps)

# Obtain summary and print it out
regsum <- summary(regout)
regsum
```

```
##
## Call:
## lm(formula = logwage ~ educ + D, data = cps)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.37350 -0.34640 -0.00835  0.33644  2.24567
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.094558   0.064568  16.952  < 2e-16 ***
## educ        0.077073   0.005259  14.656  < 2e-16 ***
## D           0.184297   0.028479   6.471 1.07e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.489 on 4730 degrees of freedom
## Multiple R-squared:  0.2093, Adjusted R-squared:  0.209
## F-statistic:   626 on 2 and 4730 DF,  p-value: < 2.2e-16
```

The output indicates that all variables are significant – i.e., there seems to be something special about college graduation. Note also that the ouput also reports the details for an F-test that none of the variables (other than the constant) matter:

$$H_0 : \beta_1 = \beta_2 = 0 \qquad H_1 : \beta_1 \neq 0 \text{ or } \beta_2 \neq 0$$

In this case, the F-statistic $((\text{SSE}_\text{R} - \text{SSE}_\text{U})/J)/(\text{SSE}_\text{U}/(n-p-1))$ equals 626, $J = 2$, $n - p - 1 = 4730$, and the p-value≈ 0.

We might want to explore whether returns are different for men and women. To this end we would create a new dummy variable

$$\text{fem} = \begin{cases} 1 & \text{if female} \\ 0 & \text{if male} \end{cases}$$

This variable is actually already in data set under the name `cps$female`. We could then estimate the regression

$$\log(\text{wage}) = \beta_0 + \beta_1 \text{edu} + \beta_2 \text{fem} + \beta_3 \text{fem} \times \text{edu} + \epsilon$$

Notice that β_2 allows us to capture that men and women may not earn the same, while β_3 allows us to capture that wages may depend on education in a different way for men and women. Estimating this model is straightforward in R, we just need to adjust our call to `lm`.

```r
# Create our new dummy variable
intregout <- lm(logwage ~ educ + female + female*educ, data = cps)

# Obtain summary and print it out
intregsum <- summary(intregout)
intregsum
```

```
##
## Call:
## lm(formula = logwage ~ educ + female + female * educ, data = cps)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.29648 -0.32106 -0.01062  0.31657  2.17021
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.979826   0.052713  18.588  < 2e-16 ***
## educ         0.098205   0.003918  25.068  < 2e-16 ***
## female      -0.502260   0.079805  -6.294 3.38e-10 ***
## educ:female  0.019105   0.005904   3.236  0.00122 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4748 on 4729 degrees of freedom
## Multiple R-squared:  0.2547, Adjusted R-squared:  0.2542
## F-statistic: 538.7 on 3 and 4729 DF,  p-value: < 2.2e-16
```

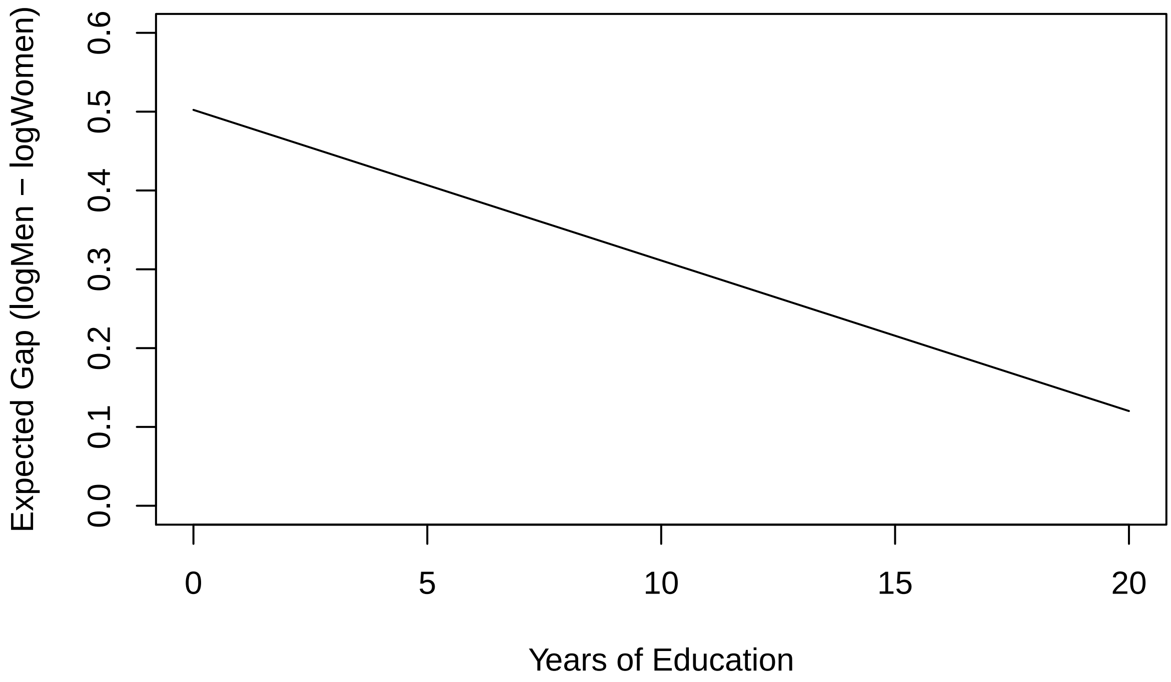
Interestingly, our estimates suggest that there is a gap in earnings between men and women but the gap decreases with education because women have higher returns to education than men. Mathematically note that

$$\begin{aligned} E[\log(\text{wage})|\text{male}, \text{edu}] &= \beta_1 + \beta_2 \text{edu} \\ E[\log(\text{wage})|\text{female}, \text{edu}] &= \beta_1 + \beta_2 \text{edu} + \beta_3 + \beta_4 \text{edu} \end{aligned}$$

Hence, the gap between men and women is given by $-\beta_3 - \beta_4 \text{edu}$. We can plot this function to see what it looks like

```r
# Extract the relevant coefficients
b2 <- intregsum$coefficients[3,1]
b3 <- intregsum$coefficients[4,1]

# Plot the curve for education between 0 and 20
curve(-b2 - b3*x , from = 0, to = 20, xlab = "Years of Education", ylab ="Expected Gap (logMen - logWome
```



The above picture would be more interesting if done for wages (instead of log wages). To test your understanding, you should try re-doing the above analysis on the model

$$\text{wage} = \beta_0 + \beta_1 \text{edu} + \beta_2 \text{fem} + \beta_3 \text{fem} \times \text{edu} + \epsilon$$

Joint Hypotheses

As we've seen, the code for testing hypotheses on a single coefficient is conceptually the same in both the single regression model or the multivariate regression model. We'll next examine how to test joint hypotheses in the model

$$Y = \beta_0 + \beta_1 X_2 + \ldots + \beta_p X_p + \epsilon.$$

A bit of a reminder from class, remember that a joint hypotheses is one where we are interested in testing whether multiple coefficients are zero against the alternative that at least one of them is not equal to zero. For example, let's return to the cps data set.

```r
# Clear the workspace
rm(list = ls())

# Our data set is store as part of the PoEData library, so let's tell R we need access to it
library(PoEdata)

# Looking at the output, there are multiple datasets called CPS. We'll use cps.
data(cps)
```

Suppose that we postulate a model that log wages depend linearly on education and quadratically in experience through the relation

$$\log(\text{wage}) = \beta_0 + \beta_1 \text{edu} + \beta_2 \text{exp} + \beta_3 \text{exp}^2 + \epsilon$$

We are interested in testing whether experience affects wages at all. In the above model this amounts to testing

$$H_0 : \beta_2 = \beta_3 = 0 \qquad H_1 : \beta_2 \neq 0 \text{ or } \beta_3 \neq 0$$

The test statistic that we saw in class for this purpose was the F-statistic, which was based on the fit of both the restricted model (i.e. the model under H_0) and the unrestricted model (i.e. the model under H_1). In this example, this amounts to the quantities

$$\mathrm{SSE_R} = \min_{b_0,b_1} \frac{1}{n}\sum_{i=1}^{n}(\log(\mathrm{wage}_i) - b_0 - b_1\mathrm{edu}_i)^2$$
$$\mathrm{SSE_U} = \min_{b_0,b_1,b_2,b_3} \frac{1}{n}\sum_{i=1}^{n}(\log(\mathrm{wage}_i) - b_0 - b_1\mathrm{edu}_i - b_2\mathrm{exp}_i - b_3\mathrm{exp}_i^2)^2;$$

note that we can think of $\mathrm{SSE_R}$ as imposing that $b_2 = b_3 = 0$ (which is true under the null hypothesis). The F statistic is the defined as

$$\frac{(\mathrm{SSE_R} - \mathrm{SSE_U}/J)}{\mathrm{SSE_U}/(n-p-1)}$$

which follows an F distribution with $(J, n-p-1)$ degrees of freedom. Here J is the number of restrictions tested for (in this example 2) and $p+1$ is the total number of regressors including the intercept (in this example 4). Now that we have the formulas, let's code this for our example

```r
# We'll begin by creating log wages for our regression
logwage <- log(cps$wage)

# Next we'll run the unrestricted regression
Uregout <- lm(logwage ~ educ + exper + I(exper^2), data = cps)

# And the restricted regression
Rregout <- lm(logwage ~ educ, data = cps)
```

Now that we have the outputs from the restricted and unrestricted regressions, we have what we need to compute our F statistic. Remember that there is more in the `lm` output that meets the eye, and it is helpful to use the `attributes` function to see what exactly is in there.

```r
# Examine the output of the `lm' function
attributes(Rregout)
```

```
## $names
##  [1] "coefficients"  "residuals"     "effects"       "rank"
##  [5] "fitted.values" "assign"        "qr"            "df.residual"
##  [9] "xlevels"       "call"          "terms"         "model"
##
## $class
## [1] "lm"
```

```r
# The residuals are being stored under Rregout$residuals, which lets us get the sum of squares
SSER <- sum(Rregout$residuals^2)  # Get the restricted sum of squares
SSEU <- sum(Uregout$residuals^2)  # Get the unrestricted sum of squares

# The next thing we need is J and n-p-1. J = 2, and n-p-1 is just the "degrees of freedom" of unrestric
df <- Uregout$df.residual
#df <- nrow(cps) - 4

# Now compute the F statistic
F <- ((SSER - SSEU)/2)/(SSEU/df)
```

We have computed the F statistic. To get a p-value, we need to compute the probability that a random variable distributed according to an F distribution with $(J, n-p-1)$ degrees of freedom. Recall that when working with distributions in R, functions that start with `p` are for computing the value of a cdf. The `f` refers to the F distribution, and hence the function we need is `pf`. Also recall that we can always rely on `?pf` to ask R for help on a function.

```r
# Recall we stored (n-p-1) in the value df
# Call pf to get the p-value for this null hypothesis
1-pf(F,2,df)
```

```
## [1] 0
```

As a second illustration of using an F statistic we can test the following null hypothesis

$$H_0 : \beta_1 = \beta_2 = \beta_3 = 0 \qquad H_1 : \beta_1 \neq 0 \text{ or } \beta_2 \neq 0 \text{ or } \beta_3 \neq 0.$$

The null hypothesis essentially states that none of our regressors are relevant against the alternative that at least one of them is relevant. The null hypothesis that none of the regressors matter is a standard one and hence reported in standard regression output. The `summary` function applied to regression output (in our example `Uregout`) in fact computes it for us.

```r
# Get the summary for the full unrestricted model
summary(Uregout)
```

```
## 
## Call:
## lm(formula = logwage ~ educ + exper + I(exper^2), data = cps)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -2.21566 -0.30288 -0.00392  0.30771  2.39575 
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  3.173e-01  4.290e-02   7.397 1.64e-13 ***
## educ         1.076e-01  2.919e-03  36.854  < 2e-16 ***
## exper        3.903e-02  2.041e-03  19.120  < 2e-16 ***
## I(exper^2)  -6.601e-04  4.778e-05 -13.817  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.4627 on 4729 degrees of freedom
## Multiple R-squared:  0.292,  Adjusted R-squared:  0.2916 
## F-statistic: 650.1 on 3 and 4729 DF,  p-value: < 2.2e-16
```

The reported F statistics is 650.1 with 3 (J) and 4729 ($n - p - 1$) degrees of freedom, which translates into a p-value of essentially zero. To test your understanding, you should try to replicate these numbers by modifying the code we used to test the null hypothesis that $\beta_2 = \beta_3 = 0$. (Hint: To run a regression on just a constant you can do `lm(logwage ~ 1, data = cps)`).

Linear Combinations

As a final illustration we will overview how to conduct inference on linear combinations by using R. To be concrete, let's return to the model

$$\log(\text{wage}) = \beta_0 + \beta_1 \text{edu} + \beta_2 \text{exp} + \beta_3 \text{exp}^2 + \epsilon.$$

Suppose we are interested in testing the null hypothesis that increasing experience from 29 to 30 has no effect against the alternative that it increases expected log wages Writing the difference as

$$\beta_2 30 + \beta_3 30^2 - \beta_2 29 - \beta_3 29^2 = \beta_2 + \beta_3 59$$

Writing $\theta_0 = \beta_2 + \beta_3 59$ as our parameter of interest, we can then write the hypothesis we are interested in testing as

$$H_0 : \theta_0 = 0 \qquad H_1 : \theta_0 > 0$$

As our estimator for θ_0 we will simply use the sample analogue $\hat{\theta}_n = \hat{\beta}_2 + 59\hat{\beta}_3$, and recall that the p-value for this statistic is given by

$$P\left(Z > \frac{\hat{\theta}}{\sqrt{\widehat{\text{Var}}(\hat{\theta})}}\right)$$

where Z follows a standard normal distribution. Also recall that using the formula for the variance of linear combinations of random variables we can motivate the following variance estimate:

$$\widehat{\text{Var}}(\hat{\theta}) = \widehat{\text{Var}}(\hat{\beta}_2 + 59\hat{\beta}_3) = \widehat{\text{Var}}(\hat{\beta}_3) + 59^2\widehat{\text{Var}}(\hat{\beta}_3) + 2 \times 59\widehat{\text{Cov}}(\hat{\beta}_2, \hat{\beta}_3)$$

Given the above formulas, let's figure out how to code this in R.

```r
# Let's begin by clearing the workspace and loading the cps data
# Clear the workspace
rm(list = ls())

# Our data set is store as part of the PoEData library, so let's tell R we need access to it
library(PoEdata)

# Looking at the output, there are multiple datasets called CPS. We'll use cps.
data(cps)

# Then run the regression we are interested in
# Create log wages
logwage <- cps$wage

# Run the regression we need
regout <- lm(logwage ~ educ + exper + I(exper^2), data = cps)

# Process out
regsum <- summary(regout)
```

Next we will extract the coefficients, which are stored in `regsum$coefficients` – remember that you can learn what is hiding in the variable by using `attributes(regsum)`. The other piece of data we need to construct our test is the variance and the covariance. For this end we can use the `vcov` function, which takes as input the regression output (in example it would be `regout`). Remember, you can always use `?vcov` to ask R for help on the function. We are ready to run our test:

```r
# First let's get the coefficients. If unsure it is helpful to print out output
# to make sure we are assigning the right variables
regsum$coefficients
```

```
##                 Estimate   Std. Error   t value      Pr(>|t|)
## (Intercept) -9.657019350 0.4959714203 -19.47092  2.738406e-81
## educ         1.191516012 0.0337524427  35.30162 1.623528e-242
## exper        0.360659473 0.0236026809  15.28045  1.722759e-51
## I(exper^2)  -0.005833394 0.0005523966 -10.56015  8.811048e-26
```

```r
# We see b3 and b4 are in the third and fourth columns
b2 <- regsum$coefficients[3,1]
b3 <- regsum$coefficients[4,1]

# Create out estimate theta
theta <- b2 + 59*b3

# Next to get the covariance matrix us vcov
```

```
covm <- vcov(regout)

# Print it out to see what it looks like
covm
```

```
##               (Intercept)          educ         exper    I(exper^2)
## (Intercept)  2.459876e-01 -1.501194e-02 -3.414392e-03  5.028559e-05
## educ        -1.501194e-02  1.139227e-03 -7.913868e-05  2.770856e-06
## exper       -3.414392e-03 -7.913868e-05  5.570865e-04 -1.246855e-05
## I(exper^2)   5.028559e-05  2.770856e-06 -1.246855e-05  3.051420e-07
```

```
# Now estimate the var of theta.
# Recall that the intercept is the first term, so that beta hat 2 is the *third* row/column
vartheta <- covm[3,3] + (59^2)*covm[4,4] + 2*59*covm[3,4]

# We are now ready to obtain the pvalue
pval <- (1-pnorm(theta/sqrt(vartheta)))

# Print it out
pval
```

```
## [1] 0.08764171
```

Should we reject the hypothesis that the marginal return to experience at thirty years is zero? To test your understanding you should try to adjust the code to test the hypothesis that the marginal return to experience at 10 years is zero against the alternative that it is positive.