exercises_3.Rmd

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
library(AER)
```

```
## Loading required package: car

## Loading required package: carData

## Loading required package: lmtest

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: sandwich

## Loading required package: survival
```

```
library(gmm)
library(stargazer)
```

```
##
## Please cite as:

##  Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables.

##  R package version 5.2.2. https://CRAN.R-project.org/package=stargazer
```

```
data("PSID1976")
df <- subset(PSID1976, participation=="yes")
```

Estimate the relationship between:

$$\log(wage_i) = \beta_0 + \beta_1 educ_i + \beta_2 exper_i + \beta_3 exper_i^2 + \varepsilon_i$$

First we ignore the endogeneity of "education"

```
iv_results1 <- lm(log(wage) ~ education + experience + I(experience^2), data = df)

exog_ols <- cbind(df$education, df$experience, I(df$experience^2))
gmm_results1 <- gmm(log(wage) ~ education + experience + I(experience^2),
              x = exog_ols, data = df)

summary(iv_results1)
```

```
##
## Call:
## lm(formula = log(wage) ~ education + experience + I(experience^2),
##     data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.08404 -0.30627  0.04952  0.37498  2.37115
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -0.5220406  0.1986321  -2.628  0.00890 **
## education        0.1074896  0.0141465   7.598 1.94e-13 ***
## experience       0.0415665  0.0131752   3.155  0.00172 **
## I(experience^2) -0.0008112  0.0003932  -2.063  0.03974 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6664 on 424 degrees of freedom
## Multiple R-squared:  0.1568, Adjusted R-squared:  0.1509
## F-statistic: 26.29 on 3 and 424 DF,  p-value: 1.302e-15
```

```r
summary(gmm_results1)
```

```
##
## Call:
## gmm(g = log(wage) ~ education + experience + I(experience^2),
##     x = exog_ols, data = df)
##
##
## Method:  twoStep
##
## Kernel:  Quadratic Spectral
##
## Coefficients:
##                  Estimate     Std. Error   t value      Pr(>|t|)
## (Intercept)      -5.2204e-01   2.0342e-01  -2.5663e+00   1.0279e-02
## education         1.0749e-01   1.3689e-02   7.8525e+00   4.0781e-15
## experience        4.1567e-02   1.4317e-02   2.9033e+00   3.6923e-03
## I(experience^2)  -8.1119e-04   3.9241e-04  -2.0672e+00   3.8713e-02
##
## J-Test: degrees of freedom is 0
##                 J-test                 P-value
## Test E(g)=0:    2.98455924400836e-24   *******
```

1. How come the point estimates $\widehat{\beta}$ are the same but the standard errors are different?

Where we have mother's and father's education as instruments for the endogenous variable (education):

```r
iv_results <- ivreg(log(wage) ~ education + experience + I(experience^2) |
                   .-education + feducation + meducation, data = df)

exog <- cbind(df$feducation, df$meducation, df$experience, I(df$experience^2))
gmm_results <- gmm(log(wage) ~ education + experience + I(experience^2),
```

```r
             x = exog, data = df)

summary(iv_results)
```

```
##
## Call:
## ivreg(formula = log(wage) ~ education + experience + I(experience^2) |
##     . - education + feducation + meducation, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.0986 -0.3196  0.0551  0.3689  2.3493
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.0481003  0.4003281   0.120  0.90442
## education        0.0613966  0.0314367   1.953  0.05147 .
## experience       0.0441704  0.0134325   3.288  0.00109 **
## I(experience^2) -0.0008990  0.0004017  -2.238  0.02574 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6747 on 424 degrees of freedom
## Multiple R-Squared: 0.1357,  Adjusted R-squared: 0.1296
## Wald test: 8.141 on 3 and 424 DF,  p-value: 2.787e-05
```

```r
summary(gmm_results)
```

```
##
## Call:
## gmm(g = log(wage) ~ education + experience + I(experience^2),
##     x = exog, data = df)
##
##
## Method:  twoStep
##
## Kernel:  Quadratic Spectral(with bw =  0.28778 )
##
## Coefficients:
##                  Estimate     Std. Error   t value      Pr(>|t|)
## (Intercept)       0.00310758   0.46562511   0.00667400   0.99467496
## education         0.06430017   0.03689420   1.74282594   0.08136402
## experience        0.04549283   0.01436735   3.16640345   0.00154337
## I(experience^2)  -0.00093366   0.00039618  -2.35663420   0.01844140
##
## J-Test: degrees of freedom is 1
##                 J-test    P-value
## Test E(g)=0:    0.37641   0.53953
##
## Initial values of the coefficients
##     (Intercept)       education      experience I(experience^2)
##    0.0481003046    0.0613966279    0.0441703943   -0.0008989696
```

2. Why do both the point estimates and the standard errors differ now?

3. Let's write our own linear IV GMM estimator

a. a function that recovers $\widehat{\beta}$
b. a function that returns the GMM objective function $Q(\theta)$
c. a function that returns the sandwich standard errors $SE(\widehat{\beta})$
d. a function that returns an updated weighting matrix $\hat{W}$.

```
gmm_estimates<- function(Y, X, Z, W){
  return
}

gmm_obj<- function(Y, X, Z, W, beta){
  return
}

gmm_se<- function(Y, X, Z, W, beta){
  return
}

gmm_W<- function(Y, X, Z, W, beta){
  return
}
```

4. Put your GMM estimates in a table with the following:

a. OLS estimates
b. OLS (GMM) estimates
c. IV estimates
d. IV (GMM) estimates
e. Your estimates of one-step GMM using Identity weights
f. Your estimates of two-step GMM starting at Identity weights
g. Your estimates of one-step GMM using 2SLS weights
h. Your estimates of two-step GMM starting at 2SLS weights
i. Use the (GMM) package to estimate continuously updating GMM (type='cue')