Estimating and Testing IV models in R

Elad Guttman

Outline

A Reminder

The Model

- The general model you discussed in class:

$$y = X\beta + u = X_1\beta_1 + X_2\beta_2 + u$$

- where:
  - x_1 are k_1 exogenous variables
  - x_2 are k_2 endogenous variables
  - $K = k_1 + k_2$
- z is a $1 \times L$ vector of:
  - k_1 exogenous variables
  - m instruments
  - $L = k_1 + m$

The Identification Assumptions

1. **2SLS.1:** $E(z'u) = 0$ - all the variables in z are uncorrelated with u
2. **2SLS.2:**

   a. $rank(E(z'z)) = L$ - no multicollinearity

   b. $rank(E(z'x)) = K$ - The instruments are correlated with X, and the model is at least just-identified ($L \geq K$)

The 2SLS Estimator

1. **First stage:** for each endogenous variable j we estimate:

$$x_j = Z\delta + \epsilon$$

   using OLS.

2. **Second stage:** then we replace x_2 with $\hat{x}_2$ (i.e., the predicted values from the first stage) and estimate:

$$y = X_1\beta_1 + \hat{X}_2\beta_2 + u$$

   using OLS (again). Under assumptions **2SLS.1** and **2SLS.2**, this is a valid procedure for obtaining a consistent estimate for β.

One endogenous variable

Data

For this exercise, we'll use the following dataset:

```r
library(wooldridge)
data("card")
#function to print the content from an R help file.
#You don't need that, just use ?card instead
help_console(card, "text", 5:8)
```

```
##      Wooldridge Source: D. Card (1995), Using Geographic Variation in
##      College Proximity to Estimate the Return to Schooling, in Aspects
##      of Labour Market Behavior: Essays in Honour of John Vanderkamp.
##      Ed. L.N. Christophides, E.K. Grant, and R. Swidinsky, 201-222.
```

Data

We'll pay special attention to the following variables:

```
help_console(card, "text", c(22:27, 30:33, 70:71))
```

```
##          • *nearc2:* =1 if near 2 yr college, 1966
##
##          • *nearc4:* =1 if near 4 yr college, 1966
##
##          • *educ:* years of schooling, 1976
##
##          • *fatheduc:* father's schooling
##
##          • *motheduc:* mother's schooling
##
##          • *wage:* hourly wage in cents, 1976
```

Estimation

The familiar `lfe` package provides flexiable syntax for estimating IV models. Here are some examples:

```
library(lfe)
library(stargazer)
library(tidyverse)

card = card %>%
  select(lwage, nearc2, nearc4, fatheduc, motheduc, educ) %>%
  drop_na() #caution: in general, this is a bad practice!

#one endogenous variable, two exogenous variables, and one instrument
twosls_with_exog = felm(lwage ~ fatheduc + motheduc | 0 | (educ ~ nearc4), data = card)

#one endogenous variable and one instrument, no exogenous variables
twosls_no_exog = felm(lwage ~ 1 | 0 | (educ ~ nearc4), data = card)

#we can easily look at the first stage results:
stage1_with_exog = twosls_with_exog$stage1
stage1_no_exog = twosls_no_exog$stage1
```

First Stage

```r
stargazer(stage1_no_exog, stage1_with_exog, dep.var.labels = "educ",
          type = "text", keep.stat = c("n"))
```

```
##
## =========================================
##                   Dependent variable:
##              ----------------------------
##                          educ
##                   (1)            (2)
## -----------------------------------------
## fatheduc                      0.216***
##                                (0.017)
##
## motheduc                      0.203***
##                                (0.020)
##
## nearc4          0.703***      0.364***
##                 (0.118)        (0.103)
##
## Constant       13.143***      9.043***
##                 (0.098)        (0.183)
##
## -----------------------------------------
## Observations     2,220          2,220
## =========================================
## Note:         *p<0.1; **p<0.05; ***p<0.01
```

Second Stage

```
stargazer(twosls_no_exog, twosls_with_exog, type = "text", keep.stat = c("n"))
```

```
##
## =========================================
##                  Dependent variable:
##              ----------------------------
##                         lwage
##                   (1)            (2)
## -----------------------------------------
## fatheduc                      -0.051**
##                                (0.020)
##
## motheduc                      -0.039**
##                                (0.019)
##
## `educ(fit)`     0.179***      0.287***
##                 (0.035)        (0.089)
##
## Constant        3.851***      3.311***
##                 (0.483)        (0.820)
##
## -----------------------------------------
## Observations     2,220          2,220
## =========================================
## Note:         *p<0.1; **p<0.05; ***p<0.01
```

Standard Errors

```
#As usual, the felm function also calculates robust SE:
stage1_no_exog$robustvcv
```

```
##               (Intercept)      nearc4
## (Intercept)  0.01014738 -0.01014738
## nearc4      -0.01014738  0.01433573
```

```
twosls_no_exog$robustvcv
```

```
##               (Intercept)  `educ(fit)`
## (Intercept)  0.22874653 -0.016784276
## `educ(fit)` -0.01678428  0.001232261
```

```
summary(twosls_no_exog, robust = T)$coefficients
```

```
##              Estimate Robust s.e  t value     Pr(>|t|)
## (Intercept) 3.8514036 0.47827453 8.052705 1.307759e-15
## `educ(fit)` 0.1786111 0.03510358 5.088118 3.920201e-07
```

Manual Estimation

Alternatively, we can estimate IV models manually by estimating two OLS models:

```
manual_stage1_with_exog = lm(educ ~ fatheduc + motheduc + nearc4, data = card)

card$`educ(fit)` = predict(manual_stage1_with_exog)
manual_twosls_with_exog = lm(lwage ~ fatheduc + motheduc + `educ(fit)`, data = card)
```

First Stage

```
stargazer(manual_stage1_with_exog, stage1_with_exog ,dep.var.labels.include = F, dep.var.caption = "Model:",
          type = "text", keep.stat = c("n"))
```

```
##
## =========================================
##                          Model:
##                 ---------------------------
##                     OLS             felm
##                     (1)              (2)
## -----------------------------------------
## fatheduc          0.216***       0.216***
##                   (0.017)         (0.017)
##
## motheduc          0.203***       0.203***
##                   (0.020)         (0.020)
##
## nearc4            0.364***       0.364***
##                   (0.103)         (0.103)
##
## Constant          9.043***       9.043***
##                   (0.183)         (0.183)
##
## -----------------------------------------
## Observations       2,220           2,220
## =========================================
## Note:         *p<0.1; **p<0.05; ***p<0.01
```

Second Stage

```
##
## ==========================================
##                         Model:
##              ----------------------------
##                    OLS            felm
##                    (1)             (2)
## ------------------------------------------
## fatheduc        -0.051***      -0.051**
##                  (0.013)        (0.020)
##
## motheduc        -0.039***      -0.039**
##                  (0.012)        (0.019)
##
## `educ(fit)`      0.287***       0.287***
##                  (0.054)        (0.089)
##
## Constant         3.311***       3.311***
##                  (0.503)        (0.820)
##
## ------------------------------------------
## Observations      2,220          2,220
## ==========================================
## Note:          *p<0.1; **p<0.05; ***p<0.01
```

Keep in mind that the right standard errors
aren't the standard errors of the second stage!

Testing for weak instruments

- Remember that weak instrument can cause the 2SLS estimator to be biased towards OLS.
- We can test that by testing the null that the coefficients on the excluded instruments are 0.
- This is a simple Wald / F-test.

```
r = 0
R = matrix(c(0, 0, 0, 1), nrow = 1)
waldtest(stage1_with_exog, R, r)
```

```
##            p         chi2          df1          p.F            F          df2
## 4.262525e-04 1.241345e+01 1.000000e+00 4.348831e-04 1.241345e+01 2.216000e+03
## attr(,"formula")
## ~nearc4
## <environment: 0x7fb681918400>
```

Testing for weak instruments

`felm` calculates this F-statistic by default (and also the robust version):

```
stage1_with_exog$iv1fstat
```

```
## $educ
##            p         chi2          df1          p.F            F          df2
## 4.262525e-04 1.241345e+01 1.000000e+00 4.348831e-04 1.241345e+01 2.216000e+03
## attr(,"formula")
## ~nearc4
## <environment: 0x7fb67d985e88>
```

```
stage1_with_exog$rob.iv1fstat
```

```
## $educ
##            p         chi2          df1          p.F            F          df2
## 4.313838e-04 1.239111e+01 1.000000e+00 4.400888e-04 1.239111e+01 2.216000e+03
## attr(,"formula")
## ~nearc4
## <environment: 0x7fb67d9a9280>
```

Multiple endogenous variables

Estimation

Very easy with the `felm` function:

```
#genetate squared education,
#so that we'll have two endogenous variables:
card = card %>%
  mutate(educ2 = educ^2)

twosls = felm(lwage ~ fatheduc + motheduc | 0 | (educ | educ2 ~ nearc4 + nearc2), data = card)

stage1 = twosls$stage1
```

First Stage

```r
summary(stage1, lhs = "educ")$coefficients
```

```
##                Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)  9.05423913 0.18370566 49.2866641 0.000000e+00
## fatheduc     0.21694207 0.01665175 13.0281834 1.937660e-37
## motheduc     0.20322762 0.02002206 10.1501861 1.081443e-23
## nearc4       0.37209458 0.10417818  3.5717133 3.622168e-04
## nearc2      -0.05817805 0.09690856 -0.6003396 5.483413e-01
```

```r
summary(stage1, lhs = "educ2")$coefficients
```

```
##               Estimate Std. Error    t value     Pr(>|t|)
## (Intercept) 72.955840  4.9895297 14.6217869 2.726431e-46
## fatheduc     5.655023  0.4522692 12.5036656 1.028820e-34
## motheduc     5.304095  0.5438083  9.7536117 4.907601e-22
## nearc4       9.669448  2.8295269  3.4173373 6.437919e-04
## nearc2      -1.090832  2.6320809 -0.4144371 6.785941e-01
```

Second Stage

```
##
## ========================================
##                    Dependent variable:
##              ---------------------------
##                           lwage
## ----------------------------------------
## fatheduc                 -0.048
##                          (0.063)
##
## motheduc                 -0.037
##                          (0.059)
##
## `educ(fit)`              -5.055
##                          (5.777)
##
## `educ2(fit)`              0.204
##                          (0.222)
##
## Constant                 36.758
##                         (36.087)
##
## ----------------------------------------
## Observations              2,220
## ========================================
## Note:        *p<0.1; **p<0.05; ***p<0.01
```

In the following slides we'll see that this estimation suffers from a weak IV problem, so do not try to interpret these results

Testing for weak instruments

```
stage1$iv1fstat
```

```
## $educ
##            p         chi2          df1          p.F            F          df2
## 1.686436e-03 1.277028e+01 2.000000e+00 1.717644e-03 6.385138e+00 2.215000e+03
## attr(,"formula")
## ~nearc4 | nearc2
## <environment: 0x7fb681f6a278>
##
## $educ2
##            p         chi2          df1          p.F            F          df2
## 2.907569e-03 1.168088e+01 2.000000e+00 2.952532e-03 5.840438e+00 2.215000e+03
## attr(,"formula")
## ~nearc4 | nearc2
## <environment: 0x7fb682019478>
```

What is this test? Is it enough?

Testing for weak instruments

- This is an equation-by-equation F-test
- For each first stage equation, we test $H_0 : \beta_{nearc2} = \beta_{nearc4} = 0$
- Remember the alternative: $H_1 : \beta_{nearc2} \neq 0$ or $\beta_{nearc4} \neq 0$
- So a single instrument could be correlated with all the endogenous regressors, while the rest of the instruments are not
- This is not enough

Conditional F-test

The **general** idea proposed by Angrist and Pischke (2009):

- Consider a model with two endogenous variables, x_1 and x_2 and a valid vector of instruments z
- Let's look at the following model: $y = \beta_1 x_1 + \beta_2 \hat{x}_2 + u*$
  - $\hat{x}_2$ is exogenous since $\hat{x}_2 = P_z x_2$
  - x_1 is endogenous and (potentially) correlated with $\hat{x}_2$

Conditional F-test

- Now let's regress x_1 on $\hat{x_2}$ and keep the residuals $M_{\hat{x_2}} x_1$
- Let's look at the following model: $y = \beta_1 M_{\hat{x_2}} x_1 + \epsilon$
  - $M_{\hat{x_2}} x_1$ is endogenous, but not correlated with $\hat{x_2}$
  - This is the short version of the previous equation
  - Applying the 2SLS procedure yields a consistent estimate for β_1
- So now we have only one endogenous variable, and we can use the usual F-test in the first stage: $M_{\hat{x_2}} x_1 = \kappa Z + \epsilon *$
- Note: This is just the general idea. See Sanderson and Windmeijer (2016) for the formal test

Conditional F-test

`condfstat` from the `lfe` package performs that test:

```
condfstat(twosls)
```

```
##            educ    educ2
## iid F 1.01738 1.009886
## attr(,"df1")
## [1] 1
```