General Perspective

Variables Instrumentales (IV)

VARIABLES INSTRUMENTALES

- Las variables instrumentales nor permiten resolver el problema de **endogeneidad** y **sesgo por variables omitidas**.

- Recordemos que si queremos ver el efecto de una variable Y en X, podemos tener que tanto Y esta correlacionada con X como X con Y.

- Además, podemos estar omitiendo variables relevantes (que no podemos controlas, por ser no observables, por ejemplo).

- Por lo tanto, recurrimos a un instrumento que este correlacionado fuertemente con la variable independiente endógena y que no este correlacionada con el error.

variables Instrumentales: 2SLS

- Conceptualmente, la estimación por variables instrumentales se puede interpretar como dos etapas independientes de mínimos cuadrados ordinarios:

  1. Primer etapa
  $$X \sim Z\delta + \epsilon,$$
  donde X son los predictores endógenos, Z los potenciales instrumentos y ϵ el vector de errores. Dado la estimación de $\hat{\delta}$, podemos obtener $\hat{X} = Z\hat{\delta}$.
  2. Segunda etapa
  $$Y \sim \hat{X}\beta_{IV} + \mu$$
  donde Y es la variable dependiente de interés y μ es el error.

2SLS en R

- Para efectos ilustrativos en R, generamos datos de 3 variables: salarios, educación y educación de los padres.
- Obsérvese que la educación de los padres es función de la variable educación.

```
# Creamos los datos aleatoriamente
set.seed(32)
salarios <- c(10, 10, 11, 13, 15, 18, 20, 20, 19, 21)
educacion <- c(1,1,1,2,1,2,3,3,3,3)
educacion_padres <- educacion - sample(c(0,1,2), size = 10,
            replace = T, prob = c(0.7,0.2,0.1))
datos_iv <- cbind.data.frame(salarios, educacion, educacion_padres)
```

- Podemos fácilmente estimar el coeficiente de IV utilizando felm() y 2SLS.

2SLS en R

```r
fs <- felm(educacion ~ educacion_padres, data = datos_iv) # first stage

datos_iv$y_hat <- fs$fitted.values # valores ajustados

ss <- felm(salarios ~ y_hat, data = datos_iv) # second stage

ss$coefficients

             salarios
(Intercept) 7.642857
y_hat       4.028571
```

- El coeficiente que obtenemos es el correcto, sin embargo los errores estandar estan mal.

IV: Forma Reducida

- Noten que en la práctica no podemos tomar los *predicted values* e incorporarlos en el *second stage* ya que los errores estandar estarían mal.
- Otro camino que podemos tomar, que de igual forma nos proporciona intuición muy valiosa, es estimar la **forma reducida** (reduced form).
- La ecuación es la siguiente:

$$Y \sim Z\gamma + \epsilon,$$

IV: Forma Reducida

- Y podemos recuperar el coeficiente de IV con el siguiente cociente:

$$\beta_{IV} = \frac{\text{Est. Reduced form}}{\text{Est. First stage}}.$$

- En R:

```
fs <- felm(educacion ~ educacion_padres, data = datos_iv) # first stage

rf <- rf <- felm(salarios ~ educacion_padres) # reduced form

b_iv <- rf$coefficients[2]/fs$coefficients[2]

### [1] 4.028571
```

IV: FELM()

- Aunque pudimos obtener el coeficiente de IV correcto de dos formas diferentes, seguimos sin poder obtener los errores estandar correctos.
- En el caso del cociente, sacar el error estandar de una división de coeficientes se complica demasiado.
- Por lo tanto, lo que se hace siempre es utilizar paquetes y funciones ya establecidas que realizan la corrección a los errores estandar.
- Nosotros tenemos la suerte que felm() nos da la opción de estimar el IV de la siguiente forma:

```
reg_iv = felm(salarios ~ 1 | 0 | (educacion ~ educacion_padres),
                data =datos_iv )
```

IV: felm()

- Observemos en una tabla todo lo que hemos hecho:

```r
tableiv <- stargazer(fs, rf, ss, reg_iv,
                     header = FALSE,
                     font.size = "scriptsize",
                     dep.var.labels.include = FALSE,
                     table.placement = "H",
                     omit = c("Constant"),
                     column.labels = c("First Stage", "Reduced Form",
                                       "2SLS", 'IV (felm)'),
                     covariate.labels = c("Instrument", "IV Estimate",
                                          "IV Estimate"),
                     omit.stat = c("f", "ser","adj.rsq"),
                     title = "Estimación de IV",
                     type = "latex")
```

Estimación por IV

	Dependent variable: Salarios			
	First Stage	Reduced Form	2SLS	IV (felm)
	(1)	(2)	(3)	(4)
Instrument	0.673***	2.712**		
	(0.199)	(1.094)		
IV Estimate			4.029**	
			(1.626)	
IV Estimate				4.029***
				(0.921)
Observations	10	10	10	10
R^2	0.589	0.434	0.434	0.818
Note:		*p<0.1; **p<0.05; ***p<0.01		

Relevancia del Instrumento

- Uno de los supuestos para obtener resultados causales de la estimación por variables instrumentales es que el instrumento debe ser relevante.
- Debe estar correlacionado fuertemente con la variable endógena.
- Para ver que el supuesto se cumple, necesitamos observar un F-Stat mayor a 10.
- Noten que este no es el F de la regresión, sino más bien, el F del test de significancia del instrumento en el First Stage.
- En R:

```r
reg_iv[["stage1"]][["iv1fstat"]][["educacion"]][["F"]]
### [1] 11.46199
```

Efectos Fijos en IV

- La función felm() nos proporciona con los errores estandar correctos al estimar IV.
- Noten que la diferencia no es pequeña, no utilizar felm() nos puede llevar a concluir erróneamente (principalmente respecto a la significancia de nuestros efectos).
- Otra gran ventaja de usar felm(), en lugar de otros paquetes como ivreg(), es que nos permite agregar efectos fijos

```
reg_iv_felm = felm(salarios ~ 1 | unidades + tiempo |
                   (educacion ~ educacion_padres) | unidades,
                   data = datos_iv)

# Y podemos acceder a los resultados usando:
summary(reg_iv_felm, diagnostics = T)
```

Doble Instrumento y Doble Variable Endógena

- En algunos proyectos, necesitaran utilizar dos instrumentos para dos variables endógenas.
- La función felm igual nos permite hacer esto.
- Supongamos que quisiéramos instrumentar Q y W con Z_3 y Z_4. Lo que haríamos entonces sería:

```
reg_iv_felm = felm(salarios ~ X1 + ... + Xn | unidades + tiempo |
                (Q|W ~ Z3 + Z4) | unidades,
                   data = datos_iv)
```

- Donde $X_1, ..., X_n$ son controles.