Trabajo Práctico-Evalución de Impacto de Políticas Públicas 2014

Víctor Funes Leal

Contents

Detalles de la simulación

Modelo utilizado

El presente trabajo práctico busca replicar los resultados de Heckman, Lalonde y Smith (1999)(Heckman, Lalonde, and Smith 1999), quienes afirman que los modelos de evaluación de impacto de políticas públicas poseen dos elementos:

1. El modelo de la variable de resultado
2. El modelo para la participación en el programa

La estructura que se utiliza en este trabajo (y que utilizan los autores) puede describirse por medio de las siguientes ecuaciones:

$$Y_{it} = \beta + \alpha_i D_i + \theta_i + U_{it}$$

si $t > k$.

$$Y_{it} = \beta + \theta_i + U_{it}$$

si $t < k$.

$$U_{it} = \rho U_{it-1} + \epsilon_{it}$$

En donde Y_{it} es el ingreso de los individuos, β es una forma de ingreso permanente, α_i es el efecto del programa para quienes acceden a él, θ_i es un efecto fijo individual no observable. Por su parte el componente no observable posee una parte que evoluciona como un proceso estcástico autoregresivo de orden 1 cuyo coeficiente de autocorrelación está dado por el parámetro ρ, a su vez éste componente no observable posee un shock aleatorio contemporáneo ϵ_{it}.

El entrenamiento tiene lugar en el año k y los individuos que deciden participar de éste los que la variable dummy D_i es igual a uno. La decisión de participar depende del valor actual esperado de los beneficios futuros (α_i/r, dónde r es la tasa de descuento), del ingreso que se renuncia en el año que accede al programa de entrenamiento Y_{ik} y de un costo idiosincrático específico a cada individuo c_i que se interpreta como la matrícula que debe pagar (en el caso que sea positivo) o el subsidio que recibe para participar (en el caso que sea negativo) y que se describe por medio de la fórmula: $c_i = \Phi Z_i + V_i$, dónde Z_i y V_i son variables aleatorias independientes entre sí y de las demás.

$$D_i = \begin{cases} 1 & \alpha_i/r - Y_{ik} - c_i > 0 \text{ y } t > k \\ 0 & \text{caso contrario} \end{cases}$$

Proceso generador de datos

Para el ejercicio de simulación se utilizan las siguientes variables y distribuciones:

- Existen 1000 individuos para los que se simulan 10 años de datos de ingresos y a su vez, se replican 100 muestras.
- Los 10 años se distribuyen de la sigueinte forma: hay 5 años pre programa ($k-5$ a $k-1$), un año de implementación ($k = 6$) y cuatro años post programa ($k+1$ a $k+4$).
- El ingreso permanente es igual para todos los individuos ($\beta = 1000$).
- El efecto tratamiento (α_i) se distribuye como normal con media 100 y desv. estándar igual a 300: $\alpha_i \sim N(100, 300)$.
- El efecto fijo también se distribuye de la misma forma con media 0 y desv. estándar igual a 300: $\alpha_i \sim N(100, 300)$.

- El componente idiosincrático de error se distribuye también como normal con media igual a 0 y desv. estándar igual a 280: $\epsilon_{it} \sim N(100, 280)$.
- Se supone que $U_{ik-5} = \epsilon_{ik-5}$ y que $\rho = 0,78$.
- Por último los componentes de la función de costos se distribuyen como normales con $\Phi = 1$, $V_i \sim N(100, 200)$ y $Z_i \sim N(\mu_Z, 200)$, dónde μ_Z se escoge de manera tal que para todas las muestras el 10% de la población participe del programa y la tasa de interés (r) es igual a 0,10.

Carga de los parámetros

En primer lugar, el programa requiere que se fijen valores para todos los parámetros, según lo indicado en el punto anterior:

```
n<-1000
beta<-rep(1000,n)
r<-0.1
rho<- 0.78
phi<-1
mean_alpha<-100
sd_alpha<-300 #(=0 para el modelo de coeficientes comunes, =300 para coef. aleatorios)
mean_theta<-0
sd_theta<-300
mean_epsilon<-0
sd_epsilon<-280
mean_v<-0
sd_v<-200
sd_z<-300
```

El comando `rep` replica el valor de β (1000) n veces, en esta caso también igual a 1000 con el objeto de agregarlo posteriormente a los vectores de ingresos.

Luego, se generan las variables aletorias también detalladas previamente:

```
#Generar variables aleatorias
alpha_i<-rnorm(n, mean_alpha, sd_alpha)
theta_i<-rnorm(n, mean_theta, sd_theta)
V_i<-rnorm(n, mean_v, sd_v)
Epsilon_it<-replicate(10, rnorm(n, mean_epsilon, sd_epsilon))
```

Las variables aleatorias α_i, θ_i y V_i se crean con una funcion `rnorm` que genera n números aletorios normalmente distribuidos cuya media es el valor que corresponde al segundo argumaneto y su desviación estándar es igual al tercero. Los valores de ϵ_{it} deben ser distintos apra cada uno de los 10 años, por lo que se generó una matriz de 1000×10 (1000 individuos en 10 años) de numeros aleatorios normales con media y desviación estándar detalladas, para ello se utilizó el comando *replicate* que, justamente, replica el vector de 1000 número aleatorios 10 veces, generando cada vez una realización distinta.

Para los valores del término aleatorio U_{it} y para los ingresos Y_{it} se optó por la siguiente estrategia: crear dos matrices de 1000×10 dónde cada columna es uno de los 10 años en los que se simula la intervención y cada fila es uno de los 1000 individuos que forman parte del estudio.

```
#Generar data frame para guardar los datos
dat<-data.frame(matrix(0, ncol=10, nrow=n))

#Nombres de las 10 columnas de la matriz de ingresos
```

```
y<-rep(0,10)
for(i in 1:10){
  y[i]<-paste("y_",i, sep="")
}
colnames(dat)<-y
```

En primer lugar se genera una matriz "vacía" de dimensiones 1000×10 (en realidad con todos sus elementos iguales a cero) y luego se colocan los nombres de las columnas con un bucle, de manera tla que sean iguales a Y_i con $i = 1 \ldots, 10$.

De idéntica manera se crea la matriz `Ui_t`, la cual posee la particularidad que sus columnas dependen de los valores de la matriz `Epsilon_it`, la primer columna de ambas matrices es igual, pero de la columna 2 en adelante se crean según la fórmula del proceso AR(1).

```
U_it<-matrix(0, nrow=n, ncol=10)
U_it[,1]=Epsilon_it[,1]

for(k in 2:10){
  U_it[,k]=rho*U_it[,k-1]+Epsilon_it[,k]
}
```

Luego se procede a rellenar los valores de la matriz de ingresos (`dat`), para ello se requieren valores para D_i pero, dado que para los períodos 1 a 6 nadie participa porque todavía no se implementó el programa, son iguales a cero para todos los individuos.

```
#Período incial
#Nadie participa antes de la implementación del programa
D_i<-rep(0, n)
dat[,1]<-beta+alpha_i*D_i+theta_i+U_it[,1]

#Períodos 2 a 5: se generan los ingresos según la ley de movimiento de U_it
for(k in 2:5){
  dat[,k]<-beta+alpha_i*D_i+theta_i+U_it[,k]
}
```

En el período 6 se implementa el programa y, como primera medida, debe individualizarse a quienes participan de los que no por medio de la variable D_i, éste valor dependerá de la media de la variable aleatoria Z_i, la cual debe fijarse de manera tal que para cada muestra participe el 10% de los individuos. Ésto se logró por medio de un bucle que itera hasta que se cumple la condición especificada, ahora bien, la velocidad de convergencia depende del valor inicial, elcual, tras varias pruebas se fijó en 700 para el modelo de coeficientes coumes ($\alpha_i = \alpha \, \forall i$) y en 4000 para el modelo de coeficientes aleatorios.

```
#mu<-700 Coeficientes comunes
mu<-4000 #Coeficientes aleatorios
ntreated<-0
y<-beta+alpha_i*D_i+theta_i+U_it[,6]
while(ntreated!=100){
  Z_i<-rnorm(n, mu, sd_z)
  c_i<-Z_i*phi+V_i
  D_i<-as.numeric(I((alpha_i/r-y-c_i)>0))
  ntreated<-as.numeric(table(D_i)[2])
  #si participantes < 1000 reduce la media de u, caso contrario la aumenta
  if(ntreated<100){
```

```
    mu=mu-1
  }else{
    mu=mu+1
  }
}
  print(paste("Media de u: ", mu))
```

```
## [1] "Media de u:  3995"
```

```
  print(paste("Número de participantes: ", ntreated))
```

```
## [1] "Número de participantes:  100"
```

El algoritmo converge rápidamente al valor de 100 individuos, a partir de los cuales se genera el vector D_i que indica cuáles de ellos participan del programa y cuales no.

Luego se rellenan los valores del período 6 haciendo que $Y_{ik} = 0$ si $D_i = 1$ (quienes participan no obtienen ingresos en el período), mientras que los que no participan continuán generando sus ingreos según la misma ley de movimiento. Por último, para los períodos 7 a 10 se vuelve al esquema anterior, pero partiendo de los nuevos valores de D_i iguales a 1 para 100 perosnas y cero apra los 900 remanentes.

```
#Reemplazo valores para el período 6 (0 para los que participan)
dat[,6]<-(1-D_i)*(beta+alpha_i+theta_i+U_it[,6])


#Períodos 7 a 10: se generan los ingresos según la ley de movimiento
for(k in 7:10){
  dat[,k]<-beta+alpha_i*D_i+theta_i+U_it[,k]
}
```

Una vez creada la matriz de valores puede observarse su composición por medio de los valores de los primeros 6 individuos a través del comando `head`:

```
head(dat)
```

```
##       y_1    y_2    y_3    y_4    y_5     y_6     y_7     y_8    y_9
## 1   36.84  809.4  912.6  768.3  198.6  -10.32   26.92  -91.67  320.2
## 2  812.00  905.0 1154.8  815.5  729.8  722.68  949.64  556.57  722.2
## 3  799.64 1081.2  857.9  964.0 1330.5 1082.96 1236.82  854.50 1196.7
## 4  910.28 1248.7  800.7  972.0 1018.4 1449.28 1881.24 1148.25 1041.8
## 5 1694.78 1676.8 1821.8 1976.5 2094.2 2599.45 1991.43 1971.42 1644.3
## 6  808.16  625.5  263.9   85.9  179.7  455.76  939.24  555.12  293.6
##     y_10
## 1  194.6
## 2  626.1
## 3  711.3
## 4  822.9
## 5 1614.4
## 6  586.2
```

Y luego, se agregan las columnas de valores de α_i y D_i a la matriz de datos, puesto que se los requerirá posteriormente.

```r
dat<-cbind(dat, alpha_i, D_i)
```

Matching

Ahora se utiliza un estimador de "matching" con el objeto de aparear observaciones entre los 100 individuos del grupo de tratamiento y un subconjunto de individuos del grupo de contol. En este caso el apareamiento se realiza por el criterio del "vecino más cercano" (nearest neighbour) con reemplazo.

Para realizar el mencionado análisis se utilizó la librería "MatchIt"(Ho et al. 2011), la cual posee un comando para aparear datos (`matchit`), utilizando diversos métodos de matching (exacto, genético, vecino más cercano, óptimo, etc.) y permitiendo también escoger la función de distancia a utilizar, en este caso se optó por la función logística, además, se incluye la opción `replace=TRUE` para que el apareamiento sea con reemplazo.

```r
library(MatchIt)
```

```
## Loading required package: MASS
```

```r
match<-matchit(D_i~y_4, method="nearest", distance="logit", data=dat, replace=TRUE)
summary(match)
```

```
##
## Call:
## matchit(formula = D_i ~ y_4, data = dat, method = "nearest",
##     distance = "logit", replace = TRUE)
##
## Summary of balance for all data:
##          Means Treated Means Control SD Control Mean Diff eQQ Med eQQ Mean
## distance         0.105         0.099      0.023     0.006   0.005    0.006
## y_4            894.368      1025.800    518.958  -131.432 118.304  130.101
##          eQQ Max
## distance   0.015
## y_4      472.383
##
##
## Summary of balance for matched data:
##          Means Treated Means Control SD Control Mean Diff eQQ Med eQQ Mean
## distance         0.105         0.105      0.024     0.000    0.00    0.001
## y_4            894.368       894.691    514.607    -0.322   16.79   23.524
##          eQQ Max
## distance   0.013
## y_4      181.280
##
## Percent Balance Improvement:
##          Mean Diff. eQQ Med eQQ Mean eQQ Max
## distance      99.27   98.46    89.05   16.81
## y_4           99.75   85.80    81.92   61.62
##
## Sample sizes:
##           Control Treated
## All           900     100
## Matched        96     100
```

```
## Unmatched     804       0
## Discarded       0       0
```

Una vez realizado el matching, se deben calcular las medias de los α_i para el total de la muestra y para los individuos del grupo de tratamiento ($D_i = 1$).

```
#Conjunto de datos apareados (weights=1)
m.data<-match.data(match)

#Medias de los alfas para los grupos apareados y no apareados
mean_alpha_i<-mean(dat$alpha_i)       #E(alpha_i)
w1<-subset(m.data, weights==1)
mean_alpha_i_Tr<-mean(w1$alpha_i)     #E(alpha_i/D_i=1)
rm(w1)

#Resultados
mean_alpha_i
```

```
## [1] 93.05
```

```
mean_alpha_i_Tr
```

```
## [1] 597.5
```

El valor de $E(\alpha_i/D_i = 1)$ es muy superior al de $E(\alpha_i)$, reflejando el efecto del tratamiento medio sobre los tratados (ATT), mientras que la segunda es el efecto medio del tratamiento (ATE). Estos resultados son importantes para, luego, calcular el sesgo de los estimadores tras realizar la simulación de montecarlo.

Gráficos

En esta sección se replicarán los gráficos de la sección 8.3.4 de Heckman et Al. Con el obejto de mostrar la existencia (o no) del llamado "Ashenfelter's dip", fenómeno que ocurre cuando no se cumple el supuesto de identificación del estimador de diferencias en diferencias.

El mencionado supuesto de identificación del estimador DiD afirma que, en ausencia del programa de entrenamineto, el cambio en los ingresos entre dos períodos de tiempo t y t' debería haber sido el mismo para los que participan como para los que no, esto es que se cumpla:

$$E(Y_{0t} - Y_{0t'}/D = 1) = E(Y_{0t} - Y_{0t'}/D = 0)$$

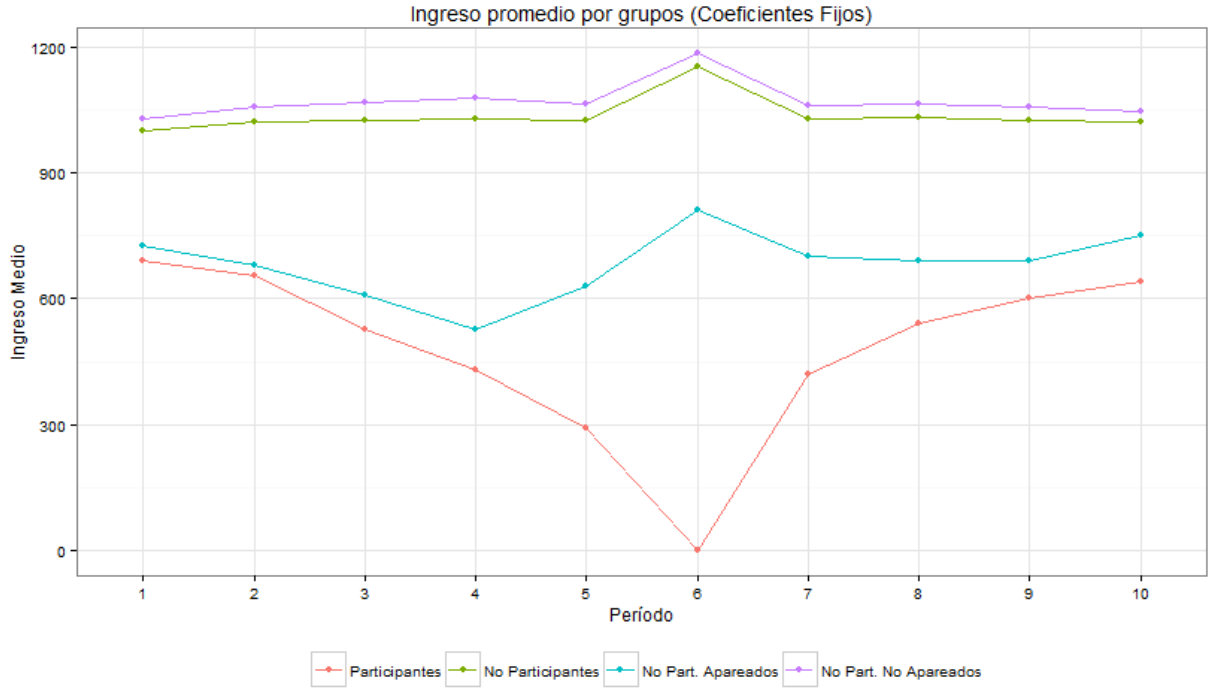
Ashenfelter (1978)(Ashenfelter 1978) observó un hecho estilizado, el cual consiste en que, previo a inscribirse en un programa de entrenamiento los participantes experimentan una caída en sus ingresos, tanto en términos absolutos como relativo a los del grupo de control. Éste fenómeno sugiere que al menos una parte del incremento de los ingresos posterior a la implementación del programa se debe a una reversión del ingreso permanente que fuera interrumpido temporalmente por un shock adverso.

El supuesto de identificación del estimador de diferencias en diferencias puede no cumplirse en la medida que el momento base t' coincida con el moemnto del "dip" transitorio y, si los no participantes no experimentan la mencionada caída, el sendero temporal de los ingresos será diferente entre participantes y no participantes entre los momentos t y t', en este caso el estimador DiD sobreestimará el efecto del entrenamiento en los participantes.

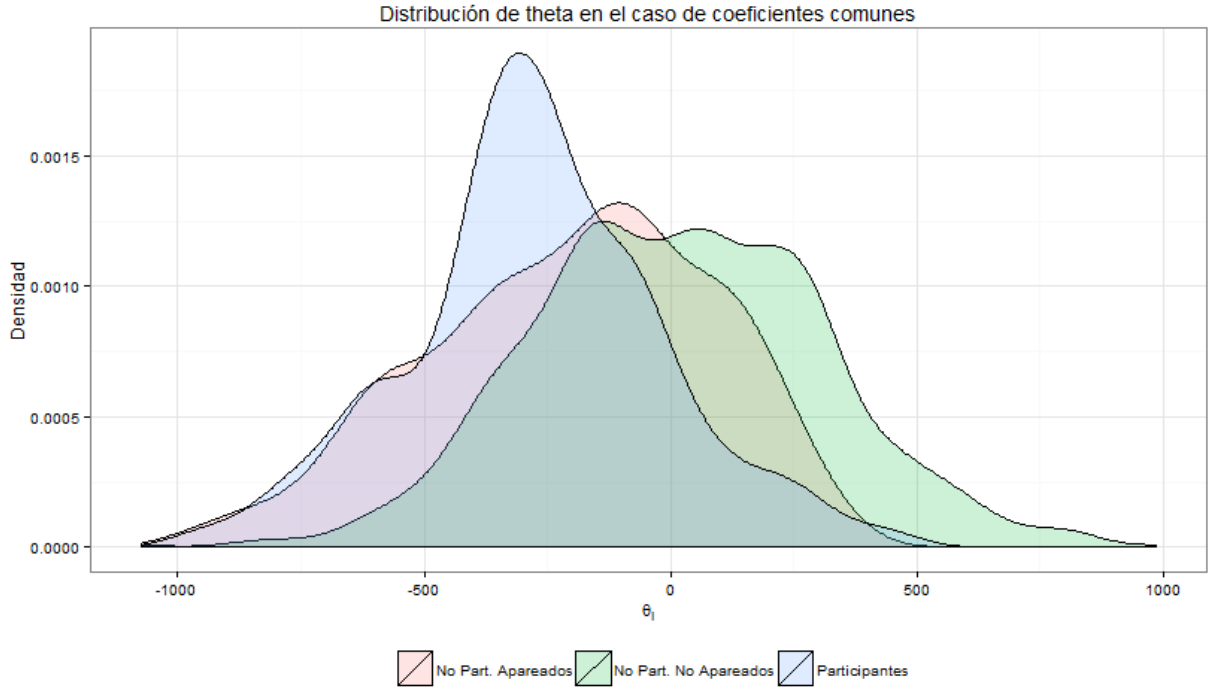
Para evaluar la existencia del "dip" es necesario contar priemro con los datos del ingreso promedio por cada período de cuatro grupos, donde w es la ponderación de la observación apareada.

- Participantes del programa (individuos con $D_i = 1$)
- No participantes del programa (individuos con $D_i = 0$)
- No participantes apareados (individuos con $D_i = 0$ y $w! = 0$)
- No participantes apareados (individuos con $D_i = 0$ y $w = 0$)

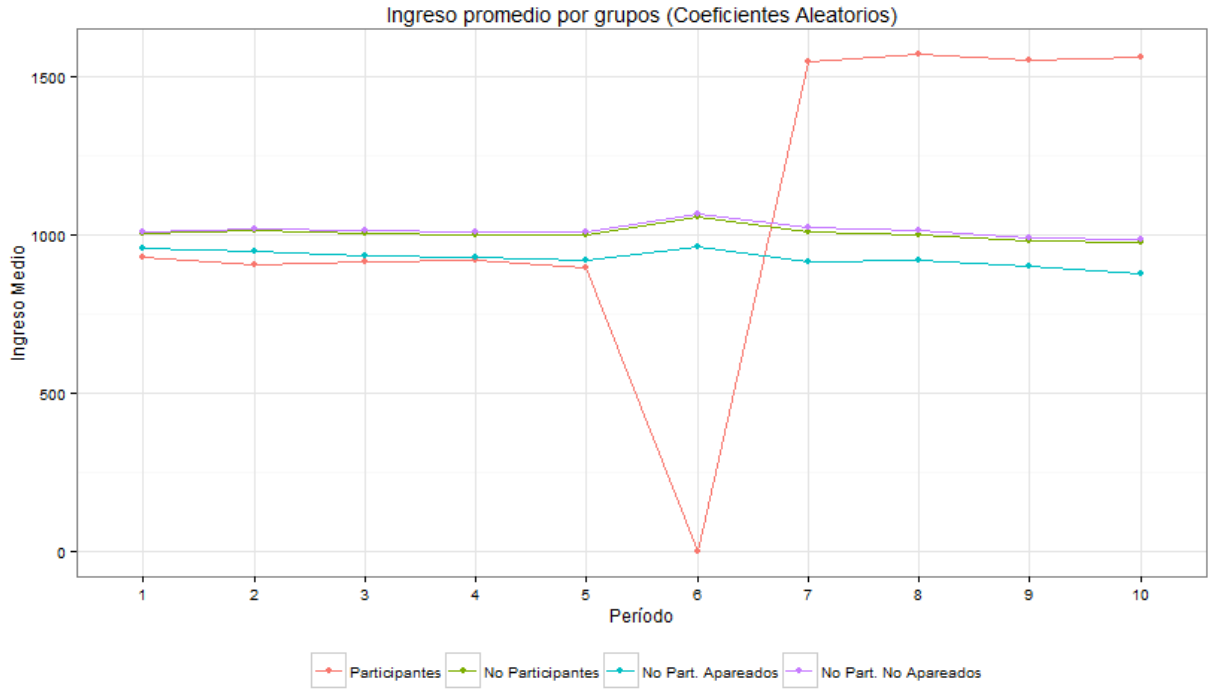
En el primer gráfico se muestra el "Ashenfelter's dip" para el caso de coeficientes fijos, dónde claramente hay una caída en el ingreso de los individuos que participan del programa en el año previo a iniciarlo. Cabe señalara que la participación en el programa de éstos, si bien aumenta sus ingresos, no llega a igualar a los de los no participantes, debido a que el obtener un beneficio positivo por participar implica que los individuos poseen una productividad baja y por lo tanto inferior a la media de la población no participante.



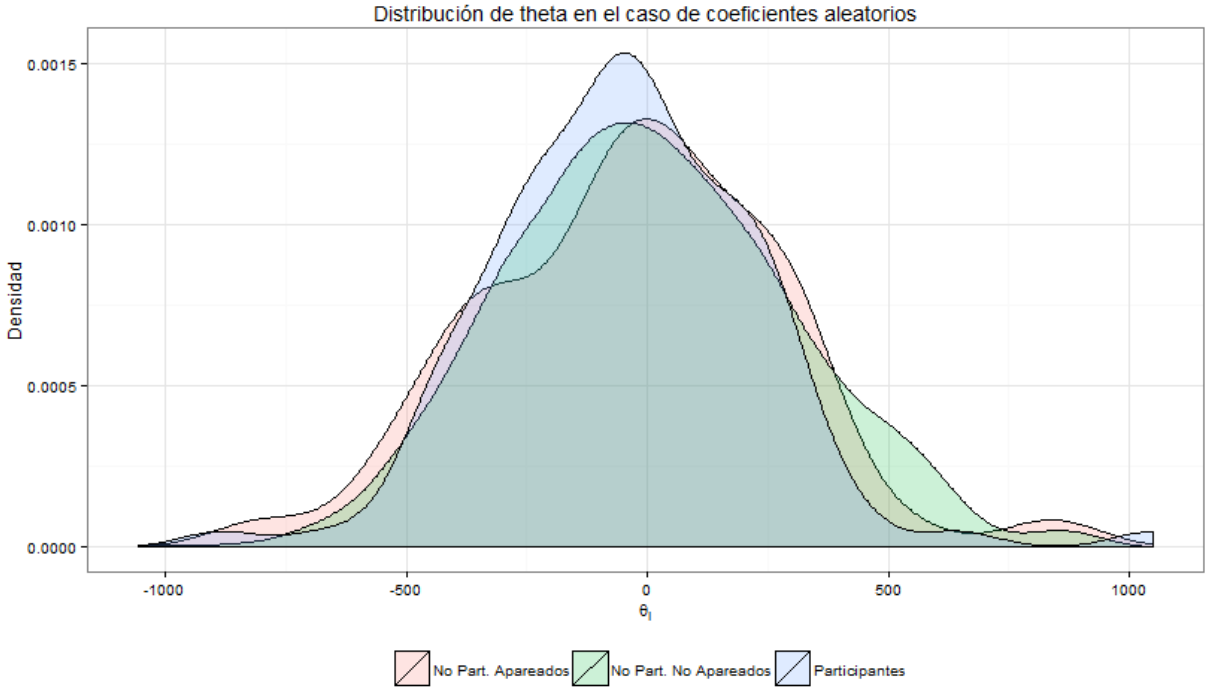
El gráfico de las distribuciones de los θ_i muestra que éstas difieren bastante entre participantes y no participantes en el caso de los coeficientes comunes.



Para el caso de coeficientes aleatorios se observa que los resultados son muy diferentes:



La figura del ingreso promedio para cada período difiere notablemente de la obtenida por Heckman et Al. (Fig. 15, pág. 2023) porque se observa un salto de gran magnitud en los ingresos del grupo tratado a partir del período 7, debido a la magnitud de $alpha_i$.



Como contracara de lo anterior, las distribuciones de los distintos subgrupos en el caso de los coeficientes aleatorios son muy similares, reflejando con mucha exactitud los resultados de Heckman.

Resumiendo, la diferencia entre el caso de coeficientes fijos y el de coeficientes aleatorios reside en el mecanismo de seleccíon, porque en el segundo sólo dependerá de θ_i y de U_{it} ya que la ganancia de participar es igual para todos ya que α es idéntico para todos los individuos, y es ésta autoselección la que se refleja en el "dip" mas pronunciado para éste caso.

References

Ashenfelter, Orley. 1978. "Estimating the Effect of Training Programs on Earnings." *Review of Economics and Statistics* 6 (1): 47–57.

Heckman, James, Robert Lalonde, and Jeffrey Smith. 1999. "The Economics and Econometrics of Active Labor Market Programs." In *Handbook of Labor Economics*, edited by Orley Ashenfelter and David Card, 3A:1865–2097. Elsevier Science Publishers.

Ho, Daniel, Kosuke Imai, Gary King, and Elizabeth Stuart. 2011. "MatchIt: Nonparametric Processing for Parametric Causal Inference." *Journal of Statistical Software* 42 (8): 1–28.