Universidad de la República
Facultad de Ciencias Sociales
DEPARTAMENTO DE ECONOMIA

## Notas Docentes

# Tratamiento de la endogeneidad y métodos de correspondencia en Stata

**Mariana Gerstenblüth**
**Juan Pablo Pagano**

**Nota Docente No. 19**

Notas docentes

# Tratamiento de la endogeneidad y métodos de correspondencia en Stata

Mariana Gerstenblüth
Juan Pablo Pagano

Mayo, 2008

**Endogeneidad**

Introducción

La presencia de una variable explicativa potencialmente endógena, entendida como la existencia de correlación entre dicha variable y el término de error en la ecuación que determina el producto de interés bajo estudio, puede ser interpretada como un problema de simultaneidad o de existencia de variables omitidas o no observables que determinen ambos fenómenos. En dicho caso, las estimaciones que incluyen a la variable potencialmente endógena en la ecuación del producto de interés como una variable exógena más son sesgadas.

En Stata, dependiendo de la versión, existen varios comandos oficiales que permiten el abordaje de estos problemas, proponiendo pruebas de exogeneidad así como diferentes metodologías que arrojan resultados consistentes. Las diferencias entre los comandos radican principalmente en la naturaleza de la variable bajo estudio (binaria o continua) y en la metodología adoptada para contrastar la existencia de exogeneidad. Con respecto a esto, es importante recalcar que no existe en la literatura un consenso con respecto a la existencia de una única prueba válida.

A efectos ilustrativos se presenta un modelo genérico que servirá de base para la exposición.

Supongamos que pretendemos estudiar los determinantes de una determinada variable de interés (de aquí en adelante $y_1$) y que existe una sola variable "sospechosa" o potencialmente endógena ($y_2$).[1] En este sentido, el modelo podría presentarse de la siguiente manera:

$$y_{1i} = \delta_1' X_{1i} + \beta y_{2i} + u_{1i}$$

Donde el vector $X_{1i}$ denota los valores de las variables $x_1, x_2, x_3, \ldots\ldots, x_k$ correspondientes a la i-ésima observación, que se asumen exógenas, y $\delta_1$ el vector de coeficientes correspondiente. Siendo $u_1$ el respectivo término de error.

A su vez tenemos:

$$y_{2i} = \delta_2' X_{2i} + \alpha' Z_i + u_{2i}$$

[1] Por motivos de simplificación se presenta el caso de una sola variable potencialmente endógena. El análisis no cambia sustancialmente en el caso de existir más de una.

Donde $X_2$ es el vector de los determinantes exógenos de $y_2$, mientras que $Z$ es un vector de variables que determinan $y_2$ $(z_1, z_2 ..... z_l)$ y que se consideran posibles instrumentos, es decir, incorrelacionados con $y_1$ y por lo tanto excluibles de la primer ecuación.
Nota: de aquí en adelante el vector $X_2 = X_1$, aunque esta condición no es imprescindible.

En el caso de utilizar variables binarias, el investigador no observa directamente dichas variables (latentes) si no que observa una variable que toma el valor uno en caso de que las variables latentes ($*$) presenten un valor superior a cero (caso genérico) :

$$y_1 = \begin{cases} 1 & si \quad y_1^* \geq 0 \\ 0 & si \quad y_1^* < 0 \end{cases}$$

$$y_2 = \begin{cases} 1 & si \quad y_2^* \geq 0 \\ 0 & si \quad y_2^* < 0 \end{cases}$$

## Variables binarias

### probexog (tobexog)[2]

Este comando computa el test de exogeneidad para un modelo probit (tobit) propuesto por Smith y Blundell (1986). Se debe(n) especificar la(s) variable(s) que el investigador sospecha puedan presentar problemas de endogeneidad. Bajo la hipótesis nula, el modelo está correctamente especificado con todas las variables explicativas como exógenas. Bajo la hipótesis alternativa, se incluyen en la regresión los residuos de una regresión lineal de la variable potencialmente endógena sobre un conjunto de instrumentos que deben ser especificados. El rechazo de la nula implicaría que dichos residuos son significativos y por lo tanto existiría un problema de endogeneidad, ameritando otro tipo de estimación (no un probit (tobit)).

Siguiendo con nuestro modelo genérico, la sintaxis del comando sería:

**. probexog  y1 x1 x2 x3 ..... xk (y2=z1 z2 ...zl)**

El comando toma como conjunto de instrumentos a los ***Z*** propuestos más el resto de las variables exógenas (***X₁***). Por motivos de identificación, debe haber al menos tantos instrumentos como variables explicativas potencialmente endógenas.

Como resultado se presenta el valor del estadístico, los grados de libertad, y el *P-Value*. No arroja los coeficientes del modelo probit.

### Test de endogeneidad computado manualmente

Existe otra posibilidad de testear la presencia de un regresor endógeno, que puede computarse en dos pasos y que equivale a la realización de un test de Hausman.

En una primera etapa se realiza un *probit* de la variable binaria potencialmente endógena sobre todas las exógenas más los instrumentos. Luego se obtienen las predicciones ( $\hat{y}_2$ ) y se incluyen en un *probit* de la variable principal de estudio como una variable explicativa más, probando su significación.

**. probit  y2 x1 x2 ..... xk  z1 z2 .... zl**
**. predict py2**

**. probit y1 x1 x2 x3 ..... xk py2**
**. test py2**

[2] En todos los casos los comandos presentan una variada gama de opciones de estimación que pueden indicarse después de una coma. El propósito de este trabajo es brindar un panorama general acerca de los comandos disponibles para afrontar posibles problemas de endogeneidad, para una descripción más detallada de cada comando y sus opciones remitirse al menú de ayuda.

En este caso, el rechazo de la hipótesis nula (H:0 $py_2 = 0$) sería equivalente a rechazar la exogeneidad de $y_2$.

**ivprobit (ivtobit)**

Este comando oficial (Stata 9) ajusta un modelo probit con regresores endógenos, en donde se deben especificar los instrumentos. Por defecto la estimación es máximo verosímil. Con la opción *two-step* la estimación se realiza mediante el estimador propuesto por Newey (mínimo chi-2).

Sintaxis:

**. ivprobit y1 x1 x2 x3 ... xk (y2=z1 z2 ..... zl), first**

Con la opción *first* en la ventana de resultados aparecen los coeficientes de ambas ecuaciones ( $y_1$, $y_2$). En caso de omitirse dicha opción, se muestran sólo los resultados de la ecuación que determina $y_1$.

A su vez, la salida presenta el resultado de una prueba de Wald sobre la exogeneidad de $y_2$ cuya hipótesis nula es que dicha variable es exógena.

Luego de la estimación existe un comando disponible que permite probar la validez de las restricciones de exclusión. Computa una variante del test propuesto por Sargan (1958) y Basman (1960) para el caso de un sistema sobredeterminado, esto es, el caso en el que el número de instrumentos ($l$) exceda el número de variables instrumentadas. En nuestro caso, se podría computar si $l \geq 2$. En la sintaxis se debe especificar la variable dependiente principal:

**. overid, depvar(y1)**

La salida arroja el resultado de la prueba cuya hipótesis nula es que los instrumentos son válidos. Se reporta el valor del estadístico propuesto por Amemiya-Lee-Newey.

Es importante recalcar que este comando está diseñado originalmente para regresores endógenos continuos ( $y_2$ no binaria). De todos modos no queda del todo claro que tan importante es esta asunción, incluso se ven en el foro de stata.com preguntas acerca de esto que no tienen una respuesta definitiva. Como resumen a este tipo de preguntas se encuentran 3 opciones:

buscar la manera especificar la variable $y_2$ de manera continua (generalmente no es posible debido al trabajo con variables latentes no observadas)
utilizar un procedimiento en 2 etapas, es decir, primero un *probit* para $y_2$ y luego utilizar las predicciones ( $\hat{y}_2$ ) en el *ivprobit*. De esta manera se estaría creando una versión continua de $y_2$

proceder con la versión binaria de $y_2$ (los resultados no cambian sustancialmente, aunque no es técnicamente correcto)

**ivprob**

Este comando es muy similar al anterior pero se encuentra disponible en versiones anteriores a Stata 9 como un archivo .ado que puede descargarse mediante:

**. findit ivprob**

Los regresores endógenos son tomados como funciones lineales de los instrumentos y las otras variables exógenas. El método de estimación es *Amemiya Generalized Least Squares (AGLS)* cuyo detalle puede encontrarse en Maddala (1983).

La sintaxis del mismo difiere del anterior ligeramente. Siguiendo con nuestro ejemplo, sería:

**. ivprob y1, endog(y2) iv(z1 z2 ... zl) exog(x1 x2 x3 ... xk)**

Autor: Joe Harkness.

**probitiv**

La sintaxis de este comando es igual a la del anterior escribiendo probitv en lugar de ivprob. Al igual que el anterior, no es un comando oficial si no un archivo .ado que puede descargarse de la misma forma. La diferencia con el comando anterior es que además se puede especificar si la primer etapa es una modelo probit o una regresión lineal. Si la parte de *stage1 (<modelo>)* se deja en blanco o se especifica "linear",la primer etapa es un MCO. Como alternativa se puede especificar: "probit".

**. ivprob y1, endog(y2) iv(z1 .. zl) exog(x1 ... xk) stage1(probit)**

Autor: Jonah B. Gelbach

**biprobit**

La otra opción cuando se trabaja con variables binarias es utilizar un procedimiento de ecuaciones simultáneas, y analizar la correlación de los términos de error de las mismas. Existen 2 variantes de este modelo: *bivariate probit* y *seemingly unrelated bivariate probit.* La diferencia radica en la sintaxis.

*Bivariate probit:*

**. biprobit y1 y2 x1 x2 x3 ..... xk z1 z2 .... zl**

Este modelo no tiene instrumentos excluidos de la primer ecuación.

*Seemingly unrelated bivariate probit:*

**. biprobit (y1=x1 x2 x3 ... xk) (y2=x1 x2 x3 ...xk z1 z2 ..... zl)**

En este caso ambas ecuaciones se relacionan solamente a través del rho (coeficiente de correlación de los errores de ambas ecuaciones). Los posibles instrumentos son excluidos de la primer ecuación.

*También, seemingly unrelated probit con dummy endógena:*

**. biprobit (y1 = x1 x2 x3 ... xk y2) (y2 = x1 x2 .. xk z1 z2 .. zl)**

Para probar la existencia de endogeneidad o simultaneidad en la ventana de resultados se muestra el valor del estadístico de la prueba de Wald sobre rho. La aceptación de la hipótesis nula implicaría que *corr* $(u_1, u_2) = 0$, lo cual puede ser interpretado como que $y_2$ es exógena en la primer ecuación, es decir, que ambas ecuaciones pueden ser estimadas por separado. Esta prueba también es presentada con el comando anterior (ivprobit).

**Variables continuas**

**ivreg**

Este es el principal comando de Stata para trabajar variables continuas con problemas de endogeneidad. Implementa una regresión lineal de la variable dependiente sobre las exógenas, instrumentando la(s) variable(s) endógena(s) a través de los instrumentos indicados así como el resto de las exógenas mediante una regresión lineal también. La sintaxis ofrece dos posibilidades:

**. ivreg y1 (y2 = z1 z2 ... zl ) x1 x2 x3 ... xk**
**. ivreg y1 x1 x2 x3 ... xk (y2 = z1 z2 ... zl)**

**ivreg2**

Esta es una extensión al comando anterior disponible en la versión 9 de Stata pero que puede descargarse también con versiones anteriores. Las principales ventajas con respecto al ivreg estándar es que este comando permite computar las pruebas de sobreidentificación (subidentificación) de los instrumentos, de exogeneidad de un subconjunto de instrumentos (estadístico *C* de Sargan), de endogeneidad de un subconjunto de regresores y otros que pueden encontrarse en el "help" del comando.

La sintaxis del comando es la misma que la del anterior, y las opciones que mas interesan en este caso son:

**, orthog (z1 z2 .. zl )** prueba de sobre identificación de los instrumentos
**, endog (x2)** prueba de endogeneidad del regresor

También existe la posibilidad de computar un test de sobreidentificación mediante el comando **overid** después de la estimación ivreg e ivreg2.

**cdsimeq**

Este comando implementa el modelo de ecuaciones simultáneas que puede encontrarse en Maddala (1983): *two-stage probit least squares* (*2SPLS*). Está pensado para una variable continua y otra endógena binaria:

**. cdsimeq (y1 x1 x2 x3 ... xk) (y2 x1 x2 x3 ... xk z1 z2 ... zl)**

En este caso, la primer variable $y_1$ debe ser la continua mientras que $y_2$ es la binaria.

**Propensity Score**

Rosenbaum y Rubin (1983) propusieron la estimación a través del ***propensity score*** como forma de reducir el sesgo en la estimación del efecto de un tratamiento sobre un set de datos observados.

Dado que en estudios empíricos la asignación de sujetos a los grupos de tratamiento y de control no es aleatoria, la estimación de los efectos de un tratamiento va a ser sesgada. La correspondencia a través del *propensity score* es una forma de corregir la estimación de los efectos del tratamiento, basado en la idea de que el sesgo se reduce cuando la comparación de los resultados se hace utilizando sujetos tratados y de control que sean lo más parecidos posibles. Dado que la "combinación" de sujetos se hace en base a un vector n-dimensional de características, puede resultar bastante complicado para n grande, de lo que este método propone resumir las características previas al tratamiento de cada individuo en una única variable: el *propensity score*.

La medida en que el sesgo es reducido depende principalmente de la riqueza y calidad de las variables de control sobre las que el *propensity score* se computa, y cuan bien se haga el matcheo. Más precisamente, el sesgo se elimina totalmente sólo si la exposición al tratamiento puede ser considerada puramente aleatoria entre los individuos que tienen el mismo valor del *propensity score.*

El *propensity score* es definido por Rosenbaum y Rubin (1983) como la probabilidad condicional de recibir tratamiento dadas las características previas al mismo:

$$p(X) \equiv Pr\{D = 1|X\} = E\{D|X\}. \qquad (1)$$

donde D = {0, 1} es el indicador de exposición al tratamiento y X es el vector multidimensional de características pre-tratamiento. Rosenbaum y Rubin (1983) muestran que si la exposición al tratamiento es aleatoria al interior de las celdas definidas por X, también lo es al interior de las celdas definidas por los valores de la variable unidimensional p(X). Como resultado, dados i individuos, si el *propensity score* p(Xi) es conocido, el Average effect of Treatment on the Treated (ATT) puede estimarse como sigue:

$$\begin{aligned} \tau &\equiv E\{Y_{1i} - Y_{0i}|D_i = 1\} \qquad (2) \\ &= E\{E\{Y_{1i} - Y_{0i}|D_i = 1, p(X_i)\}\} \\ &= E\{E\{Y_{1i}|D_i = 1, p(X_i)\} - E\{Y_{0i}|D_i = 0, p(X_i)\}|D_i = 1\} \end{aligned}$$

donde $Y_{1i}|$ y $Y_{0i}$ son los resultados potenciales de las situaciones contrafactuales (tratamiento y no tratamiento) respectivamente.

Formalmente las dos siguientes hipótesis son necesarias para derivar (2) dado (1).

Lema 1. **Equilibrio de las variables pre-tratamiento dado el *propensity score*.**

Si p(X) es el propensity score, entonces

$$D \perp X \mid p(X). \qquad (3)$$

Lema 2**. La distribución de los resultados es independiente del tratamiento D condicionado a los valores que toma el conjunto de las variables pre-tratamiento (de control)**

$$Y_1, Y_0 \perp D \mid X. \qquad (4)$$

Esto es, la asignación al tratamiento es "*unconfounded*"[3] dado el *propensity score*.

$$Y_1, Y_0 \perp D \mid p(X). \qquad (5)$$

Si la hipótesis de equilibrio del lema 1 es satisfecha, observaciones con el mismo *propensity score* tendrán la misma distribución de características observables (e inobservables), independientemente de su estado respecto al tratamiento. En otras palabras, para un determinado *propensity score*, la exposición al tratamiento es aleatoria, de lo que las unidades tratadas y de control deberán, en promedio, ser observacionalmente idénticas. Cualquier modelo probabilístico estándar puede ser usado para estimar el *propensity score*.

El programa *pscore.ado* estima el *propensity score* y testea la hipótesis de equilibrio de acuerdo al siguiente algoritmo:

1. Estimación a través de un modelo probit (o logit):

$$Pr\{D_i = 1|X_i\} = \Phi(h(X_i)) \qquad (6)$$

donde Φ denota la f.d.a normal (logistica) y h(Xi) es la especificación inicial.

2. Se divide la muestra en k intervalos de igual tamaño de *propensity score*, con k determinado por el usuario. Por defecto en le programa el número de intervalos es 5.

3. Al interior de cada intervalo se testea que el promedio del *propensity score* entre los tratados y los de control no difiera.

4. Si el test da resultados negativos, se debe separar los intervalos en mitades y volver a hacer la prueba.

[3] También se la conoce como selección en observables ó supuesto de exogeneidad. Es decir, la asignación al grupo de tratamiento es aleatoria al interior de cada grupo de individuos con idénticas características observables. Esto es posible sólo si podemos usar una amplio set de información detallada.

5. Se continúa hasta que, en todos los intervalos, el promedio del *propensity score* no difiera entre los tratados y los de control.

6. En cada intervalo se testea que las medias de cada característica no difieran entre las unidades tratadas y de control. Esto es una condición necesaria de la hipótesis de equilibrio.

7. Si las medias de una o más características difieren, la propiedad de equilibrio no se satisface, de lo que debe probarse una especificación de h(Xi) más parsimoniosa.

Los pasos del 2 al 7 pueden restringirse a la región de soporte común[4] (common support). Esta restricción implica que el test de la propiedad de equilibrio se haga solamente en base a aquellas observaciones cuyo *propensity score* pertenece a la intersección de los soportes del *propensity score* de los tratados y los de control. Imponer la condición de *common support* en la estimación del *propensity score* puede mejorar la calidad de las correspondencias (*matching*) usados para estimar el ATT.

La estimación del *propensity score* no es suficiente para estimar el ATT a través de la ecuación (2). Esto es así porque la probabilidad de observar dos unidades con exactamente el mismo valor del *propensity score* es en principio cero dado que p(X) es una variable continua.

Varios métodos se han propuesto en la literatura para abordar este problema. Los más usados son: correspondencia a través del Vecino más Cercano, Radio, Kernel y Estratificación.

El Método de **Estratificación** cosiste en dividir el rango de variación del *propensity score* en intervalos tal que al interior de los mismos las unidades tratadas y de control tengan en promedio el mismo *propensity score*. En la práctica, los mismos intervalos que se definieron para estimar el *propensity score* pueden ser usados. Luego, al interior de cada intervalo donde tratados y de control están presentes, la diferencia entre los resultados promedio de tratados y los de control es computada. El ATT se obtiene finalmente como un promedio de los ATT de cada intervalo ponderados por la distribución de las unidades tratadas entre bloques. Una de las desventajas de éste método es que pueden quedar intervalos donde unidades de control o de tratamiento estén ausentes.

El Método del **Vecino más Cercano** toma cada unidad tratada y busca a aquella de control que tenga un *propensity score* más cercano. Aunque no sea necesario, este método usualmente se utiliza con reposición, en el sentido que una unidad de control puede ser el mejor "match" para más de una unidad tratada. Dado que todas las unidades tratadas tienen su correspondiente unidad de control, la diferencia entre el resultado de la unidad tratada y su correspondiente unidad de control se computa. El ATT se obtiene promediando todas estas diferencias.

[4] Esto es, aquellas unidades tratadas cuyo propensity score es más grande que el mayor propensity score de las de control no se usan ni se machean.

En el caso del Vecino más Cercano, todas las unidades tratadas encuentran su correspondiente de control. Sin embargo, es obvio que algunas de estas correspondencias son bastante pobres dado que el vecino más cercano en algunos casos tiene un *propensity score* muy diferente, pero contribuye, al igual que en otro caso, al ATT. El método de

**Radio** y el **Kernel Matching** ofrecen una solución a este problema. En el primer caso cada unidad tratada se machea con una única unidad de control cuyo *propensity score* cae en un vecindario predeterminado. Si la dimensión del vecindario (el radio) es demasiado pequeño, es posible que alguna unidad de tratamiento no encuentre su correspondiente. Aunque hay que tener en cuenta que cuanto menor el tamaño del vecindario, mejor es la calidad del matching. Con el Método Kernel todas las unidades tratadas se corresponden con un promedio ponderado de todas las unidades de control, con ponderadores inversamente proporcional a la distancia entre el *propensity score* de los tratados y los de control.

T es el conjunto de las unidades tratadas y C el de las de control, y Yit y Yic los resultados observados de las unidades tratadas y de control respectivamente. C(i) es el set de unidades de control macheadas con la unidad de tratamiento i, con un propensity score estimado pi. El Método del **Vecino más Cercano** establece que:

$$C(i) = \min_j \| p_i - p_j \| \qquad (7)$$

En la práctica el caso de múltiples vecinos más cercanos es raro, en particular si el set de características X contiene variables continuas.

En el Método **Radio**,

$$C(i) = \{p_j \mid \| p_i - p_j \| < r\}, \qquad (8)$$

Entonces todas las unidades de control con *propensity scores* estimados que caigan dentro de un radio r respecto a pi se machean con la unidad de tratamiento i.

En el caso del método radio y el vecino mas cercano el número de unidades de control macheadas con la observación i $\in$ T se denomina Nci, se definen los ponderadores como wij = 1 si j $\in$ C(i) y wij = 0 en otro caso. Entonces, la fórmula para ambos tipos de estimadores es la que sigue (donde M es cualquiera de los dos estimadores y el número de unidades en el grupo de tratamiento es Nt):

$$\begin{aligned} \tau^M &= \frac{1}{N^T} \sum_{i \in T} \left[ Y_i^T - \sum_{j \in C(i)} w_{ij} Y_j^C \right] \qquad (9) \\ &= \frac{1}{N^T} \left[ \sum_{i \in T} Y_i^T - \sum_{i \in T} \sum_{j \in C(i)} w_{ij} Y_j^C \right] \end{aligned}$$

$$= \frac{1}{N^T}\sum_{i\in T} Y_i^T - \frac{1}{N^T}\sum_{j\in C} w_j Y_j^C$$

con wj definido por wj = Σiwij.
El estimador **Kernel** está dado por:

$$\tau^K = \frac{1}{N^T}\sum_{i\in T}\left\{ Y_i^T - \frac{\sum_{j\in C} Y_j^C G(\frac{p_j - p_i}{h_n})}{\sum_{k\in C} G(\frac{p_k - p_i}{h_n})} \right\} \qquad (11)$$

con G(·) una función del tipo kernel y hn un parámetro acotado.

Método de **Estratificación**:

Se basa en la misma estratificación realizada para estimar el *propensity score*. Al interior de cada bloque, el programa computa:

$$\tau_q^S = \frac{\sum_{i\in I(q)} Y_i^T}{N_q^T} - \frac{\sum_{j\in I(q)} Y_j^C}{N_q^C}$$

con I(q) el set de unidades en el bloque q y NTq y NCq el número de unidades tratadas y de control en el bloque q.

Entonces, el estimador del ATT basado en el método de estratificación se computa de acuerdo a la siguiente fórmula:

$$\tau^S = \sum_{q=1}^{Q} \tau_q^S \frac{\sum_{i\in I(q)} D_i}{\sum_{\forall i} D_i}$$

con ponderaciones dadas por la correspondiente proporción de unidades tratadas y Q el número de bloques.

Sintaxis y opciones en Stata

**Pscore y att*** son comandos del tipo de los de regresión

Es importante limpiar la base de datos antes de correr el programa

En **pscore** la opción (**newvar**) es obligatoria

**Pscore** y **att*** están estrechamente relacionados. Primero se debe correr *pscore* para estimar el **propensity score** y testear si la propiedad de equilibrio se satisface y luego estimar el ATT mediante uno o más de los programas **att***

Sin embargo, debe tenerse en cuenta que los **att*** son programas que pueden correrse solos, es decir, si el usuario desea estimar el *propensity score* de otra forma, puede hacerlo. Luego debe especificar el nombre de su estimación como variable en el programa *att**.

Para **att***s*, además de estimar el *propensity score*, le usuario debe proveer una variable que contenga el identificador del bloques. En este caso es entonces más conveniente usar *pscore* porque ya genera este identificador.

Opciones para **pscore**

**pscore(newvar)** es una opción obligatoria que pide al usuario que especifique un nombre para el *propensity score* estimado.

**blockid(newvar)** permite al usuario especificar el nombre de la variable para el número de bloques del *propensity score*.

**detail** muestra una salida mas detallada, con los pasos para alcanzar el resultado.

**logit** usa un modelo logit para estimar en vez de un probit.

**comsup** restringe el análisis de la propiedad de equilibrio a todos aquellas unidades de control y las tratadas que pertenecen a la región de soporte común. Se crea una variable llamada it:comsup para identificar a las variables en esta región.

**level(#)** permite establecer el nivel de significación para la prueba de la propiedad de balanceo. Por defecto es 0.01

**numblo(#)** permite fijar el número de bloques utilizados para estimar la propiedad de balanceo. Por defecto es 5.

Opciones comunes a todos los comandos **att***

**comsup** restringe el ATT a la región *common support*.

**detail** muestra una salida mas detallada, mostrando los pasos para alcanzar el resultado.

**bootstrap** hace bootstrap para obtener el error estándar del efecto del tratamiento.

**reps(integer)** especifica el número de replicaciones. Por defecto es 50. Esta opción tiene sentido sólo si la anterior fue usada.

**noisily** muestra todas las replicaciones del bootstrap. Esta opción tiene sentido sólo si bootstrap fue usada.

Opciones para **attnd** y **attnw**[5]

**pscore(scorevar)** especifca el nombre de la variable provista por el usuaro que contiene el propensity score estimado. Si esta opción o se especifica, *attnd* y *attnw* estimará el *propensity score* con la especificación provista en la lista de variables por el usuario.

**logit** usa un modelo logit para estimar en vez de un probit.

Opciones para **attr**

**pscore(scorevar)** especifica el nombre de la variable provista por el usuario que contiene el *propensity score* estimado. Si esta opción o se especifica, *attr* estimará el *propensity score* con la especificación provista en la lista de variables por el usuario.

**logit** usa un modelo logit para estimar en vez de un probit.

**radius(#)** especifica el tamaño del radio. Por defecto es 0.1.

Opciones para **attk**

**pscore(scorevar)** especifica el nombre de la variable provista por el usuario que contiene el *propensity score* estimado. Si esta opción o se especifica, *attk* estimará el *propensity score* con la especificación provista en la lista de variables por el usuario.

**logit** usa un modelo logit para estimar en vez de un probit.

**epan** especifica que se use Epanechnikov kernel en vez de kernel Gaussiana.

**bwidth(#)** especifica el ancho de la banda a ser usado cuando se elige la opción **epan**. Por defecto es 0.06. Esta opcion tiene efecto sólo si se requirió el uso de Epanechnikov kernel.

Opciones para *atts*

**pscore(scorevar)** especifica el nombre de la variable provista por el usuario que contiene el *propensity score* estimado. Si esta opción no se especifica, *atts* estimará el *propensity score* con la especificación provista en la lista de variables por el usuario.

[5] Ambas hacen el att mediante el método del vecino mas cercano, pero en un caso si encuentra dos correspondencias, una hacia atrás y otra hacia delante, lo que hace es ponderarlas (*attnw*), mientras que en el otro caso toma indistintamente cualquiera de las dos. (*attnd*)

**blockid(blockvar)** es una opción obligatoria que especifica el nombre de la variable provisto por el usuario que contiene el identificador para los bloques del *propensity score* estimado.

Ejemplo

Los datos provienen de la *National Supported Work* (NSW) restringido a la NSW-PSID-1 sub muestra. Se usa esta base porque es ampliamente conocida en la aplicación de este método a la economía laboral, y porque está públicamente disponible el sitio Web de Rajeev Dehejia (http://www.columbia.edu/˜rd247/nswdata.html.).
Se trata de replicar los resultados obtenidos por Dehejia y Wahba (1999) pero no se ha podido replicar numéricamente sus resultados dada la falta de información acerca de aspectos tales como niveles de significación, número de bloques usados en la estratificación, etc. Sin embargo se obtienen resultados cualitativamente similares.
La variable de interés es RE78 (ganancias reales en 1978); el tratamiento T es la participación en el grupo de tratamiento NSW. Las variables de control son edad, educación, black (1 si negro, 0 en otro caso), hispanic (1 si hispano, 0 en otro caso), married (1 si casado, 0 en otro caso), nodegree (1 if no degree, 0 otherwise), RE75 (ganancias en 1975), y RE74 (ganancias en 1974). El grupo de tratamiento contiene 185 observaciones, el de control 2490 observaciones, de lo que en total son 2675 observaciones.

Resultado de **pscore**

```
. pscore T age age2 educ educ2 marr black hisp RE74 RE75 RE742 RE752 blackU74
> , pscore(mypscore) blockid(myblock) comsup numblo(5) level(0.005) logit;

**************************************************
Algorithm to estimate the propensity score
**************************************************

The treatment is T
```

| T | Freq. | Percent | Cum. |
|---|---|---|---|
| 0 | 2490 | 93.08 | 93.08 |
| 1 | 185 | 6.92 | 100.00 |
| Total | 2675 | 100.00 | |

```
Estimation of the propensity score
Iteration 0:   log likelihood = -672.64954
  (output omitted)
Iteration 9:   log likelihood = -204.97537
Logit estimates                                   Number of obs   =       2675
                                                  LR chi2(12)     =     935.35
                                                  Prob > chi2     =     0.0000
Log likelihood = -204.97537                       Pseudo R2       =     0.6953
```

| T | Coef. | Std. Err. | z | P>\|z\| | [95% Conf. | Interval] |
|---|---|---|---|---|---|---|
| age | .3316904 | .1203295 | 2.76 | 0.006 | .0958489 | .5675318 |
| age2 | -.0063668 | .0018554 | -3.43 | 0.001 | -.0100033 | -.0027303 |
| educ | .8492683 | .3477041 | 2.44 | 0.015 | .1677807 | 1.530756 |
| educ2 | -.0506202 | .0172492 | -2.93 | 0.003 | -.084428 | -.0168124 |
| marr | -1.885542 | .2993282 | -6.30 | 0.000 | -2.472214 | -1.298869 |
| black | 1.135973 | .3517793 | 3.23 | 0.001 | .446498 | 1.825447 |
| hisp | 1.96902 | .5668567 | 3.47 | 0.001 | .8580017 | 3.080039 |
| RE74 | -.0001059 | .0000353 | -3.00 | 0.003 | -.000175 | -.0000368 |
| RE75 | -.0002169 | .0000414 | -5.24 | 0.000 | -.000298 | -.0001357 |
| RE742 | 2.39e-09 | 6.43e-10 | 3.72 | 0.000 | 1.13e-09 | 3.65e-09 |
| RE752 | 1.36e-10 | 6.55e-10 | 0.21 | 0.836 | -1.15e-09 | 1.42e-09 |
| blackU74 | 2.144129 | .4268089 | 5.02 | 0.000 | 1.307599 | 2.980659 |
| _cons | -7.474742 | 2.443502 | -3.06 | 0.002 | -12.26392 | -2.685566 |

```
note: 22 failures and 0 successes completely determined.

Note: the common support option has been selected
The region of common support is [.00061067, .9752541]

Description of the estimated propensity score
in region of common support
```

```
                Estimated propensity score
-------------------------------------------------------------
      Percentiles      Smallest
 1%     .0006426       .0006107
 5%     .0008025       .0006149
10%     .0010932       .0006159       Obs                1342
25%     .0023546        .000618       Sum of Wgt.        1342
50%     .0106667                      Mean           .1377463
                       Largest        Std. Dev.      .2746627
75%     .0757115        .974804
90%     .6250823       .9749805       Variance       .0754396
95%      .949302       .9752244       Skewness       2.185182
99%      .970598       .9752541       Kurtosis       6.360726

*****************************************************
Step 1: Identification of the optimal number of blocks
Use option detail if you want more detailed output
*****************************************************

The final number of blocks is 7

This number of blocks ensures that the mean propensity score
is not different for treated and controls in each blocks
```

Siguiendo el algoritmo previamente explicado, los bloques para los cuales el *propensity score* promedio de los tratados y los de control difiere, se parten a la mitad. El algoritmo continúa hasta que la propiedad se cumple. En nuestro caso esto ocurre con 7 bloques. Luego **pscore** testea la propiedad para cada variable.

```
************************************************************
Step 2: Test of balancing property of the propensity score
Use option detail if you want more detailed output
************************************************************

The balancing property is satisfied
```

Este es el único resultado mostrado por Stata cuando la opción **detail** no se pide. Luego, si la propiedad se cumple, se tabula la distribución de tratados y de control al interior de cada grupo.

```
This table shows the inferior bound, the number of treated
and the number of controls for each block
  Inferior |
  of block |          T
of pscore  |         0          1 |     Total
-----------+----------------------+----------
  .0006107 |       924          7 |       931
       .05 |       102          4 |       106
```

| | | | |
|---|---|---|---|
| .1 | 56 | 7 | 63 |
| .2 | 41 | 28 | 69 |
| .4 | 14 | 21 | 35 |
| .6 | 13 | 20 | 33 |
| .8 | 7 | 98 | 105 |
| Total | 1157 | 185 | 1342 |

```
Note: the common support option has been selected

*****************************************
End of the algorithm to estimate the pscore
*****************************************
```

Luego se estima el ATT

Resultados con **attnd** y **attnw**

```
. attnd RE78 T age age2 educ educ2 marr black hisp RE74 RE75 RE742 RE752 blackU
> 74, comsup boot reps(100) dots logit;

 The program is searching the nearest neighbor of each treated unit.
 This operation may take a while.

ATT estimation with Nearest Neighbor Matching method
(random draw version)
Analytical standard errors

------------------------------------------------------------------------------
n. treat.   n. contr.         ATT   Std. Err.           t
------------------------------------------------------------------------------
      185          57    1667.644    2113.592       0.789
------------------------------------------------------------------------------

Note: the numbers of treated and controls refer to actual
nearest neighbour matches


Bootstrapping of standard errors
command:     attnd RE78 T age age2 educ educ2 marr black hisp RE74 RE75 RE742 R
> E752 blackU74     , pscore() logit  comsup
statistic:   r(attnd)
(obs=2675)
.....................................................................
> ..................

Bootstrap statistics
------------------------------------------------------------------------------
Variable |    Reps   Observed       Bias   Std. Err.   [95% Conf. Interval]
---------+--------------------------------------------------------------------
     bs1 |     100   1667.644  -85.68572   1211.026   -735.2937  4070.582  (N)
         |                                           -839.9554  3643.178  (P)
         |                                           -394.5013  4064.472 (BC)
------------------------------------------------------------------------------
                                N = normal, P = percentile, BC = bias-corrected

ATT estimation with Nearest Neighbor Matching method
(random draw version)
Bootstrapped standard errors

------------------------------------------------------------------------------
n. treat.   n. contr.         ATT   Std. Err.           t
------------------------------------------------------------------------------
      185          57    1667.644    1211.026       1.377
------------------------------------------------------------------------------

Note: the numbers of treated and controls refer to actual
nearest neighbour matches
```

Resultados con **attnr**

Para un radio r=0.001 se obtiene:

```
. attr RE78 T age age2 educ educ2 marr black hisp RE74 RE75 RE742 RE752 blackU7
> 4, comsup boot reps(100) dots logit radius(0.0001);

 The program is searching for matches of treated units within radius.
 This operation may take a while.

ATT estimation with the Radius Matching method
Analytical standard errors

n. treat.   n. contr.         ATT   Std. Err.           t
       23          66   -5546.140    2388.723      -2.322

Note: the numbers of treated and controls refer to actual
matches within radius


Bootstrapping of standard errors
command:     attr RE78 T age age2 educ educ2 marr black hisp RE74 RE75 RE742 RE
> 752 blackU74     , pscore() logit  comsup radius(.0001)
statistic:   r(attr)
(obs=2675)
.....................................................................
> ..................
Bootstrap statistics
Variable |    Reps   Observed       Bias   Std. Err.   [95% Conf. Interval]
---------+-------------------------------------------------------------------
       bs1 |     100   -5546.14    425.988   4657.081    -14786.8  3694.519  (N)
           |                                           -13867.87  5668.455  (P)
           |                                           -13867.87  5668.455 (BC)

                              N = normal, P = percentile, BC = bias-corrected

ATT estimation with the Radius Matching method
Bootstrapped standard errors

n. treat.   n. contr.         ATT   Std. Err.           t
       23          66   -5546.140    4657.081      -1.191

Note: the numbers of treated and controls refer to actual
matches within radius
```

Resultados con **attk**

```
. attk RE78 T age age2 educ educ2 marr black hisp RE74 RE75 RE742 RE752 blackU7
> 4, comsup boot reps(100) dots logit;

 The program is searching for matches of each treated unit.
 This operation may take a while.

ATT estimation with the Kernel Matching method

n. treat.   n. contr.         ATT   Std. Err.          t
      185        1157    1537.943           .          .

Note: Analytical standard errors cannot be computed. Use
the bootstrap option to get bootstrapped standard errors.


Bootstrapping of standard errors
command:     attk RE78 T age age2 educ educ2 marr black hisp RE74 RE75 RE742 RE
> 752 blackU74     , pscore() comsup logit   bwidth(.06)
statistic:   r(attk)

(obs=2675)
..................................................................................
> ..................
Bootstrap statistics
Variable |    Reps   Observed       Bias   Std. Err.   [95% Conf. Interval]
---------+--------------------------------------------------------------------
     bs1 |     100   1537.943  -51.50918   1016.874    -479.755  3555.642  (N)
         |                                             -439.9654  3601.629  (P)
         |                                             -343.8961  3826.322 (BC)

                           N = normal, P = percentile, BC = bias-corrected

ATT estimation with the Kernel Matching method
Bootstrapped standard errors

n. treat.   n. contr.         ATT   Std. Err.          t
      185        1157    1537.943    1016.874      1.512
```

Al ATT estimado es muy parecido al del vecino mas cercano.

Resultados con **atts**

Se realiza con el mismo número de bloques que el *propensity score.*

```
. atts RE78 T, pscore(mypscore) blockid(myblock) comsup boot reps(100) dots;

ATT estimation with the Stratification method
Analytical standard errors
```

| n. treat. | n. contr. | ATT | Std. Err. | t |
|---|---|---|---|---|
| 185 | 1157 | 2208.600 | 777.866 | 2.839 |

```
Bootstrapping of standard errors
command:     atts RE78 T  , pscore(mypscore) blockid(myblock) comsup
statistic:   r(atts)
(obs=2675)
.....................................................................
> ..................
```

Bootstrap statistics

| Variable | Reps | Observed | Bias | Std. Err. | [95% Conf. | Interval] | |
|---|---|---|---|---|---|---|---|
| bs1 | 100 | 2208.6 | 96.6845 | 850.9957 | 520.0395 | 3897.16 | (N) |
| | | | | | 570.5178 | 4012.478 | (P) |
| | | | | | 778.2358 | 4184.918 | (BC) |

N = normal, P = percentile, BC = bias-corrected

```
ATT estimation with the Stratification method
Bootstrapped standard errors
```

| n. treat. | n. contr. | ATT | Std. Err. | t |
|---|---|---|---|---|
| 185 | 1157 | 2208.600 | 850.996 | 2.595 |

Los resultados obtenidos por attnw, attk y atts son muy parecidos entre sí, y tomados en su conjunto dan evidencia de un positivo ATT en el rango de 1500-2200 $, lo cual se asemeja bastante a los resultados experimentales de $1700.

También se usa **psmatch**, **nnmatch** y **treateff** en la estimación del *propensity store*.

**Referencias bibliográficas**

Baum C.F., Schaffer M.E. y Stillman, S. (2003): “Instrumental Variables and GMM: Estimation and Testing”, Stata Journal.

Becker, S. y Andrea Ichino, (2002). "Estimation of average treatment effects based on propensity scores", Stata Journal.

Lee, L., Amemiya's Generalized Least Squares and Tests of Overidenfication in Simultaneous Equation Models with Qualitative or Limited Dependent Variables. (1992) Econometric Reviews, Vol. 11, No. 3, 1992, pp. 319-328.

Maddala, G. (1983). Limited-Dependent and Qualitative Variables in Econometrics. Cambridge University Press, Inglaterra.

Rosenbaum, P.R., y D.B. Rubin (1983) “The Central Role of the Propensity Score in Observational Studies for Causal Effects”, *Biometrika* 70(1), 41-55

Smith, Richard J. y Blundell, Richard W. (1986), An exogeneity test for a simultaneous equation Tobit model with an application to labor supply. Econometrica, 54:4, 679-686.