

TUTORIAL DE LIMDEP/NLOGIT

```
/*PERSONALMENTE ORGANIZO LOS ARCHIVOS DE INSTRUCCIONES POR BLOQUES.  
ES DECIR, CADA BLOQUE DE OPERACIONES LO SEPARO CON UNA LINEA EN  
BLANCO.*//
```

```
? LA EMPIEZO CON UN TITULO EN LETRAS MAYUSCULAS, PRECEDIDO POR UN  
INTERROGANTE (LIMDEP NO EJECUTA ESAS LINEAS).
```

```
/* tambien se pueden poner comentarios entre "parentesis" del tipo /*  
para abrir y */ para cerrar,  
que pueden afectar a  
varias  
lineas, como se muestra aqui*/
```

```
/*LOS COMANDOS LOS ESCRIBO EN MINUSCULA, PARA QUE SE DISTINGA MEJOR  
DEL TITULO.  
PERO TODO ESO ES PERSONAL y a veces me lo salto. LIMDEP NO LO REQUIERE  
(LIMDEP LO PASA TODO A MAYUSCULAS)*/
```

```
?EJEMPLOS DE COMANDOS HABITUALES EN EJERCICIOS DE VALORACION  
CONTINGENTE
```

```
?PRIMERO, PUEDO LEER LOS DATOS DE LA BASE DE DATOS. SI ESTA EN FORMATO  
WKS (Lotus 123):
```

```
read; file=c:\subdirectorío\archivo.wks; format=wks; names$
```

```
/* LA OPCION NAMES PERMITE DAR EL NOMBRE DE LAS VARIABLES  
AUTOMATICAMENTE IGUAL AL DE LA PRIMERA FILA DE LA BASE DE DATOS.  
COMO SIEMPRE, LAS INSTRUCCIONES SE SEPARAN CON ;  
Y SE TERMINAN CON $. DA IGUAL QUE OCUPEN VARIAS LINEAS COMO SIEMPRE,  
EJECUTO LA INSTRUCCION CON LA SELECCION MARCADA.
```

```
LIMDEP LEE BASES CON VARIOS FORMATOS EXCEL (XLS), LOTUS, ASCII.  
PERO SI LA VERSION DE EXCEL ES MAS RECIENTE QUE LA DE LIMDEP, DA  
ERROR. POR ESO SUELO GUARDAR LOS ARCHIVOS XLS EN WKS.  
AL ESTAR LOTUS EN DESUSO, TENGA LA VERSION QUE TENGA DE LIMDEP PUEDO  
IMPORTAR LOS DATOS  
LIMDEP NO ADMITE LETRAS ENTRE LOS VALORES. SOLO NUMEROS.  
LOS TITULOS DE LAS VARIABLES LOS CORTA A LOS 8 CARACTERES. NO DEBEN  
CONTENER ESPACIOS EN BLANCO NI CARACTERES "INTERNACIONALES"*/
```

```
/*  
A LA DERECHA DEL $ TAMBIEN PUEDO ESCRIBIR COMENTARIOS, PORQUE NADA SE  
EJECUTA A LA DERECHA DE $
```

```
Otra forma de importar una base de datos es mediante menu de windows  
de LIMDEP:
```

```
PROJECT  
IMPORT  
VARIABLES
```

```
y seleccionar el archivo correspondiente. De forma similar se puede  
exportar a varios formatos.
```

```
*/
```

```
/*NO ES NECESARIO TENER YA UNA BASE DE DATOS PARA OPERAR CON LIMDEP.  
PUEDO GENERAR LOS DATOS INVENTADOS  
POR EJEMPLO CON NUMEROS ALEATORIOS (RANDOM NUMBERS). HAY MUCHAS FORMAS  
DE HACERLO*/
```

```

create; wtp=rnn(100,50)$ CREA UNA DISTRIBUCION NORMAL, POR DEFECTO DE
UN NUM. DETERMINADO DE OBSERVACIONES, CON MEDIA 100 Y ERROR ESTANDAR
50
create; income=rnd(200)$ CREA UNA DISTRIBUCION UNIFORME DISCRETA ENTRE
1 Y 200
create; educ=rnu(0,10)$
create; if(educ<=5)edummy=0;
        if(educ>5)edummy=1$ Convierte la variable educ en una dummy

/*Los numeros aleatorios los crea con una semilla que determina el
reloj del ordenador.
Pero pupedo elegir y fijar la semilla ****poner ejemplo****.
Una forma de crear dummies es con la funcion fix(variable), que
redondea al integro mas proximo.
Por ejemplo, la variable edummy se puede generar, entre otras maneras,
de la siguiente forma: */

create; edummy=rnu(0,1); edummy=fix(edummy)$

/* Otra posibilidad */

create; edummy=rnd(2)$
recode; edummy; 2=0$ La orden RECODE permite recodificar valores
concretos dentro de una variable

?ESTADÍSTICOS BÁSICOS
dstat; rhs=*$ rhs significa RIGHT HAND SIDE. El asterisco equivale
aqui a TODAS LAS VARIABLES

?REGRESION POR MINIMOS CUADRADOS ORDINARIOS
regress; lhs=wtp; rhs=one, income, edummy$ lhs es LEFT HAND SIDE. La
sintaxis para la constante (vector de unos) es ONE

/*Otra forma de proceder es utilizar los menus. En la ventana de
output aparece la instruccion que ejecuta.
La instruccion se puede entonces copiar en la ventana de
instrucciones*/

/*EJERCICIO: Utilizando los menus de LIMDEP, generar la regresion
anterior con distintas opciones de output,
asi como con distintas formas de estimacion. Luego, copiar las
instrucciones y realizar cambios directamente*/

/*A veces algunas observaciones de una variable estan en blanco
("missing"), sin ningun valor.
LIMDEP trata estos casos poniendole el valor -999.
Comprobamos que pasando algunos valores de una variable a -999 resulta
en observaciones en blanco en la base de datos*/

create; emissing=edummy$
recode; emissing;1=-999$
list; emissing$ En cambio, ir a la ventana de la base de datos y
observar la variable emissing

/* Otra forma de crear missing values: */

create; if (edummy=1&income>=100)emissing=-999; (else)emissing=edummy$

/*Para realizar una regresion que no tenga en cuenta los valores
missing de las variables que intervienen,

```

se puede utilizar la orden SKIP al principio y NOSKIP al final, y así saltarse las observaciones con missing values*/

```
skip$
regress; lhs=wtp; rhs=one, income, emissing$
noskip$
```

?OTRA FORMA MAS EFICIENTE DE GENERAR DATOS Y REALIZAR REGRESIONES...

```
create; X = Rnn(0,1) ; Y = X + 1 + Rnn(0,2)$
proc
    sample; first-last$
    regress; lhs=y; rhs=one,x$
endproc
calc; first=1; last=10$ PARA 10 OBSERVACIONES
execute
```

/*Excepcionalmente, las ordenes que implican solo una palabra (execute, proc, endproc, skip, unskip...) pueden prescindir del \$ final*/

/* EJERCICIO: Por cualquiera de los procedimientos anteriores, o cualquier otro, preparar una base de datos para una regresión hedónica y estimar el valor implícito de la variable de interés.*/

/* En LIMDEP se guardan por separado las instrucciones (extensión .lim) (se pueden tener abiertas varias simultáneamente), los outputs o resultados, y los datos. Estos últimos se guardan con el proyecto, bajo la extensión .lpj (LIMDEP PROJECT), de forma que al llamar a este archivo de nuevo, todos los datos, estadísticos, etc. se recuperan. */

?EL MANUAL DE LIMDEP ESTA REALMENTE MUY BIEN. NO DUDEIS EN CONSULTARLO.