



# Universidad Nacional del Callo

## Guía de Usuario de Stata 11

### **Autores:**

Juan Manuel Rivas Castillo

[juanmanuel263@gmail.com](mailto:juanmanuel263@gmail.com)

David Joel Esparta Polanco

[david.esparta@gmail.com](mailto:david.esparta@gmail.com)

23 de mayo de 2013

# Índice general

Apendice de tablas	VII
Apendice de figuras	VIII
<b>1 Introducción</b>	<b>1</b>
<b>I Introducción al STATA</b>	<b>3</b>
<b>2 Aspectos Generales del STATA</b>	<b>5</b>
2.1 Entorno de STATA . . . . .	5
2.2 La Barra de Herramientas . . . . .	7
2.3 Tipos de Archivo . . . . .	8
2.4 Sintáxis de los Comandos del STATA . . . . .	8
2.5 Expresiones Lógicas del STATA . . . . .	9
2.6 Organizando un Proyecto de Trabajo . . . . .	10
2.7 Recursos del STATA . . . . .	11
2.8 Comandos de Ayuda . . . . .	11
2.9 Instalación de Nuevos Comandos . . . . .	12
2.10 Ejercicio Propuesto . . . . .	15
<b>3 Gestión de Base de Datos</b>	<b>17</b>
3.1 El Do-File . . . . .	17
3.1.1 Comentarios en el Do-File . . . . .	18
3.2 Iniciando la Estructura de un Do-File . . . . .	19
3.3 Asignando Memoria . . . . .	20
3.4 Manejo de Directorios . . . . .	20
3.5 Guardar Resultados en Bitácoras . . . . .	21
3.6 Creando Base de Datos . . . . .	24
3.7 Cargando Base de Datos . . . . .	25
3.7.1 Abriendo base de datos del STATA. . . . .	25

3.7.2	Importando Base de Datos . . . . .	27
3.7.3	Convertir Base de Datos . . . . .	30
3.8	Guardar Base de Datos . . . . .	34
3.9	Inspección Base de Datos . . . . .	34
3.10	Generando y Transformando Variables . . . . .	39
3.11	Nombrando y Etiquetando Variables . . . . .	41
3.12	Tipo y Formato de Variables . . . . .	43
3.12.1	Tipo de Variables . . . . .	43
3.12.2	Formato de Variables . . . . .	44
3.13	Conversión de Variables . . . . .	46
3.13.1	De una Variable String Numérica a una Variable Numérica . . . . .	46
3.13.2	De una Variable Numérica a una Variable String . . . . .	46
3.13.3	De una Variable String No-Numérica a una Variable Numérica . . . . .	47
3.14	Selección de Muestra y Variables . . . . .	47
3.15	Manipulación de Base de Datos . . . . .	49
3.15.1	Ordenar Observaciones y Variables . . . . .	49
3.16	Preservar y Restaurar Base de Datos . . . . .	51
3.17	Tablas y Tabulaciones . . . . .	52
3.17.1	Tabulate . . . . .	52
3.17.2	Table . . . . .	55
3.17.3	Tabstat . . . . .	57
3.18	Formas de Base de Datos . . . . .	59
3.18.1	Formas Long y Wide . . . . .	59
3.19	Colapsar Base de Datos . . . . .	60
3.20	Fusión de Base de Datos . . . . .	61
3.21	Ejercicio Propuesto . . . . .	64
<b>4</b>	<b>Gráficos en STATA</b> . . . . .	<b>67</b>
4.1	Introducción a STATA GRAPH . . . . .	67
4.2	Tipos de Gráficos . . . . .	68
4.2.1	Histograma . . . . .	68
4.2.2	Graph Toway . . . . .	73
4.2.3	Gráfico de Caja y Bigote (Box Plot) . . . . .	87
4.2.4	Gráfico de Pastel (Pie) . . . . .	89
4.2.5	Gráfico de Barras (Bar) . . . . .	92
4.2.6	Gráfico de Puntos (Dot Plot) . . . . .	96
4.3	Añadiendo Textos a los Gráficos . . . . .	98
4.4	Múltiples Ploteos . . . . .	100
4.5	Guardar, Combinar y Exportar Gráficos . . . . .	105
4.6	Ejercicio Propuesto . . . . .	106

<b>5</b>	<b>Programación en STATA</b>	<b>109</b>
5.1	Generando Números Seudo-Aleatorios . . . . .	109
5.2	Macros Local y Global . . . . .	111
5.2.1	Macro Global . . . . .	112
5.2.2	Macro Local . . . . .	112
5.3	Comandos para Bucles . . . . .	113
5.3.1	El comando foreach . . . . .	113
5.3.2	El comando forvalues . . . . .	115
5.3.3	El comando while . . . . .	115
5.4	Escalares y Matrices . . . . .	116
5.4.1	Escalar . . . . .	116
5.4.2	Matrices . . . . .	117
5.5	Usando los Resultados de los Comandos de STATA . . . . .	119
5.5.1	Usando los Resultados con el Comando r-class . . . . .	119
5.5.2	Usando los Resultados con el Comando e-class . . . . .	121
5.6	Ejercicio Propuesto . . . . .	123
<b>6</b>	<b>Diseño Muestral</b>	<b>125</b>
6.1	Muestra vs Censo . . . . .	125
6.2	Diseño Muestral . . . . .	126
6.3	Técnicas de Muestreo . . . . .	128
6.4	La Encuesta Nacional de Hogares (ENAH) . . . . .	132
6.5	Aplicación - ENAH . . . . .	136
6.6	Ejercicio Propuesto . . . . .	150
<b>II</b>	<b>Modelos de Regresión Lineal</b>	<b>151</b>
<b>7</b>	<b>Modelo de Regresión Lineal General</b>	<b>153</b>
7.1	Especificación y Supuestos del Modelo General . . . . .	153
7.2	Formas Funcionales . . . . .	154
7.3	Bondad de Ajuste . . . . .	155
7.3.1	Coeficiente de Determinación . . . . .	155
7.3.2	Coeficiente de Determinación Ajustado . . . . .	155
7.4	Prueba de Hipótesis e Intervalo de Confianza . . . . .	156
7.5	Criterios para elección de modelos . . . . .	156
7.5.1	Criterio de Información de AKAIKE (AIC) . . . . .	156
7.5.2	Criterio de Información de SCHWARZ (BIC) . . . . .	157
7.6	Pruebas de Hipotesis y Estimacion MCO con Variables Dummy . . . . .	167
7.7	Ejercicio Propuesto . . . . .	171

<b>8</b>	<b>Heteroscedasticidad</b>	<b>175</b>
8.1	Problema de Heteroscedasticidad . . . . .	175
8.2	Test de Heteroscedasticidad . . . . .	178
8.2.1	Método Informal (Método Gráfico) . . . . .	179
8.2.2	Método Formal . . . . .	181
8.3	Medidas Correctivas . . . . .	189
8.4	Ejercicio Propuesto . . . . .	193
<b>9</b>	<b>Autocorrelación</b>	<b>195</b>
9.1	Problema de Autocorrelación . . . . .	195
9.2	Test de Autocorrelación . . . . .	197
9.2.1	Método Informal (Método Gráfico) . . . . .	197
9.2.2	Método Formal . . . . .	201
9.3	Medidas Correctivas . . . . .	204
9.3.1	Método de Estimación Prais-Winsten . . . . .	205
9.3.2	Método de Estimación Cochrane-Orcutt . . . . .	206
9.3.3	Estimación de Modelos Dinámicos . . . . .	208
9.3.4	Estimación de Modelos Dinámicos . . . . .	210
9.4	Ejercicio Propuesto . . . . .	211
<b>10</b>	<b>Multicolinealidad</b>	<b>213</b>
10.1	Problema de Multicolinealidad . . . . .	213
10.2	Detección de Multicolinealidad . . . . .	215
10.3	Medidas Correctivas . . . . .	218
10.4	Ejercicio Propuesto . . . . .	220
<b>III</b>	<b>Modelos de Elección Discreta</b>	<b>221</b>
<b>11</b>	<b>Modelo de Elección Discreta Binaria</b>	<b>223</b>
11.1	Tipos de Variables de Elección Discreta . . . . .	223
11.2	Modelos de Elección Discreta para Variables Dicotómicas . . . . .	224
11.2.1	Modelo Lineal de Probabilidad (MLP) . . . . .	224
11.2.2	Modelo Logístico (Logit) . . . . .	226
11.2.3	Modelo Probabilístico (Probit) . . . . .	227
11.2.4	Relaciones entre Modelos Logit y Probit . . . . .	227
11.3	Ejercicio Propuesto . . . . .	242

<b>IV</b>	<b>Econometría de Series de Tiempo</b>	<b>243</b>
<b>12</b>	<b>Introducción a Series de Tiempo en STATA</b>	<b>245</b>
12.1	Análisis de Serie Temporal Univariado en STATA . . . . .	245
12.2	Operadores de Serie de Tiempo . . . . .	248
12.2.1	Operador de Rezagos . . . . .	249
12.2.2	Operador de Adelanto . . . . .	250
12.2.3	Operador de Diferencia . . . . .	251
12.2.4	Operador de Diferencia Estacional . . . . .	252
12.2.5	Combinando Operadores de Serie Temporales . . . . .	253
12.2.6	Expresiones con Operadores . . . . .	254
12.2.7	Cambios Porcentuales . . . . .	256
12.3	Ejercicio Propuesto . . . . .	257
<b>13</b>	<b>Series de Tiempo Estacionarios</b>	<b>259</b>
13.1	La Naturaleza de Series de Tiempo . . . . .	259
13.2	Estacionariedad . . . . .	261
13.3	Procesos Autoregresivos y de Media Móvil . . . . .	263
13.3.1	Procesos de Media Móvil (MA) . . . . .	264
13.3.2	Procesos Autoregresivos (AR) . . . . .	268
13.3.3	Procesos Autoregresivos y Medias Móviles (ARMA) . . . . .	275
13.4	Función de Autocorrelación Muestral (FAS) y Parcial (FAP) . . . . .	279
13.4.1	Función de Autocorrelación Muestral (FAS) . . . . .	279
13.4.2	Función de Autocorrelación Parcial (FAP) . . . . .	283
13.5	Ejercicio Propuesto . . . . .	286
<b>14</b>	<b>Procesos Estocásticos No Estacionarios</b>	<b>287</b>
14.1	Serie No Estacionaria en Media . . . . .	287
14.1.1	Proceso Estacionario de Tendencia Determinística . . . . .	288
14.1.2	Proceso Estacionario de Tendencia Estocástica . . . . .	289
14.2	Proceso de Raíz Unitaria . . . . .	293
14.2.1	Pruebas de Raíz Unitaria . . . . .	294
14.2.2	Transformación de Series No estacionarias . . . . .	300
14.3	Ejercicio Propuesto . . . . .	303
<b>15</b>	<b>Modelos de Vectores Autoregresivos</b>	<b>305</b>
15.1	Ejercicio Propuesto . . . . .	315
<b>16</b>	<b>Modelos de Corrección de Errores</b>	<b>317</b>
16.1	Ejercicio Propuesto . . . . .	330

<b>V</b>	<b>Modelos de Panel de Datos</b>	<b>333</b>
<b>17</b>	<b>Modelos de Datos de Panel Estáticos</b>	<b>335</b>
17.1	Modelo Agrupado (Pooled)	336
17.2	Modelos con efectos individuales (One-Way)	336
17.3	Modelo de Efectos Fijos (FE)	337
17.4	Modelo de Efectos Aleatorios (RE)	338
17.5	Comparación de Modelos	339
17.5.1	Modelo Pooled vs. Modelo de Efectos Fijos: Prueba F	339
17.5.2	Modelo Pooled vs. Modelo de Efectos Aleatorios: Prueba LM	339
17.5.3	Modelo de Efecto Fijo vs. Modelo de Efecto Aleatorio: Prueba Hausman	340
17.6	Ejercicio Propuesto	354

# Apéndice de Tablas

3.1	Tipo de Variable Numérico . . . . .	43
3.2	Tipo de Variable No Numérico . . . . .	43
3.3	Formato de Variable Numérico . . . . .	44
3.4	Formato de Variable con Fechas . . . . .	44
4.1	Opciones de <b>mysymbol()</b> . . . . .	75
4.2	Opciones - <b>legend()</b> . . . . .	81
4.3	Opciones - <b>connect()</b> . . . . .	82
4.4	Opciones - <b>clpattern()</b> . . . . .	83
5.1	Funciones de Variables Aleatorias . . . . .	110
6.1	Muestra vs. Censo . . . . .	126



# Índice de figuras

2.1	Entorno del STATA 11	5
2.2	Viewer Windows, Do-File, Data Windows	6
2.3	La Barra de Herramientas	7
2.4	Expresiones Lógicas	9
2.5	Esquema de un Proyecto de Trabajo	10
3.1	Manejo de Directorio	21
3.2	Editando Base de Datos	25
3.3	Cargando Base de Datos	26
3.4	Cargando Base de Datos	27
3.5	STAT TRANSFER	30
3.6	Fomas de Base de Datos Long y Wide	59
4.1	Histograma (1)	69
4.2	Histograma (2)	70
4.3	Histograma (3)	71
4.4	Histograma (4)	72
4.5	Histograma (5)	73
4.6	Scatter Plot (1)	74
4.7	Scatter Plot (2)	76
4.8	Scatter Plot (3)	76
4.9	Scatter Plot (4)	77
4.10	Scatter Plot (5)	78
4.11	Multiples Scatter Plot	79
4.12	Line Plot (1)	80
4.13	Line Plot (2)	80
4.14	Line Plot (3)	81
4.15	Line Plot (4)	82
4.16	Line Plot (5)	83
4.17	Line Connected Plot	84
4.18	Otros Plot (1)	85
4.19	Otros Plot (2)	86

4.20 Otros Plot (3)	86
4.21 Otros Plot (4)	87
4.22 Box Plot (1)	88
4.23 Box Plot (2)	89
4.24 Pie Graph (1)	91
4.25 Pie Graph (2)	92
4.26 Bar Graph (1)	93
4.27 Bar Graph (2)	94
4.28 Bar Graph (3)	95
4.29 Bar Graph (4)	96
4.30 Otros Plot	97
4.31 Dot Plot	98
4.32 Texto en Gráficos (1)	99
4.33 Texto en Gráficos (2)	100
4.34 Ploteos Múltiples (1)	101
4.35 Ploteos Múltiples (2)	102
4.36 Ploteos Múltiples (3)	103
4.37 Ploteos Múltiples (4)	104
4.38 Ploteos Múltiples (5)	105
4.39 Graficos Combinados	106
5.1 Gráficos - Variables Aletarias	111
8.1 Método Gráfico (1) - Heteroscedasticidad	179
8.2 Método Gráfico (2) - Heteroscedasticidad	180
9.1 Método Gráfico (1) - Autocorrelación	198
9.2 Método Gráfico (2) - Autocorrelación	199
12.1 Comando tsline	249
13.1 Ruido Blanco	263
13.2 Proceso Media Móvil - MA	267
13.3 Proceso Autoregresivo - AR	273
13.4 Proceso Autoregresivo de Media Móvil - ARMA	278
13.5 FAS para un Proceso MA(1)	280
13.6 FAS para un Proceso AR(1)	281
13.7 FAS para un Proceso AR(1) con $\Phi < 0$	282
13.8 FAS para un Proceso ARMA(1,1)	283
13.9 FAP para un Proceso AR	285
15.1 Proyección	313

15.2 Función de Impulso Respuesta . . . . .	315
16.1 Función de Impulso Respuesta en un MCE . . . . .	329
16.2 Proyección en un MCE . . . . .	330
17.1 Datos de Panel balanceado y No balanceado . . . . .	336
17.2 Heterogeneidad entre Individuos . . . . .	342
17.3 Caja y Bigote de la Heterogeindad entre Individuos . . . . .	342

# Capítulo 1

## Introducción

El presente manual de *Stata 11* es una recopilación de clases dictadas en la Facultad de Ciencias Económicas de la Universidad Nacional del Callado (UNAC) y en el Departamento de Economía de la Universidad Nacional Agraria la Molina (UNALM) en los cursos de Econometría I, II e Intermedia. El orden está organizado de tal forma que cualquier estudiante pueda empezar a familiarizarse y utilizar modelos micro y macroeconómicos desde la parte básica hasta una introducción a los temas avanzados de econometría usando el software Stata. Este manual no tiene el interés de reemplazar un libro teórico de econometría, por tanto, se requiere que el estudiante previamente haya revisado autores tales como Gujarati & Porter (2010), Wooldridge (2006), Greene (2010) y Cameron & Trivedi (2009).

Durante el contenido de los temas, el estudiante podrá contar con ejercicios resueltos y propuestos en cada capítulo y cuya información proviene de fuentes económicas peruanas tales como *BCRP*, *INEI*, *MINEM*, *MINAG*, etc. Lo anterior se considera relevante pues el estudio de la teoría y la resolución de ejercicios debe completarse con el análisis de problemas reales donde el estudiante compruebe por sí mismo lo que aporta la teoría estudiada.

El objetivo final que se pretende con su publicación es facilitar la difusión a los estudiantes e interesados en la práctica de Econometría usando *Stata*. Por tal motivo, usted puede descargar la base de datos de los ejercicios resueltos y propuestos de cada capítulo en la siguiente pagina web **xxxxxx** para su práctica.

Queremos dar las gracias a Juan Pichihua Serna (UNALM) y a Juan Manuel

Rivas Castillo (UNAC) por su apoyo y motivación en el estudio de la econometría. Sin embargo, como es de rigor, todos los errores son de nuestra entera responsabilidad y agradeceríamos comentarios para su mejora de esta primera versión.

# Parte I

## Introducción al STATA



# Capítulo 2

## Aspectos Generales del STATA

### 2.1. Entorno de STATA

Al momento de iniciar la sesión en STATA, esta mostrará cuatro ventanas importantes:

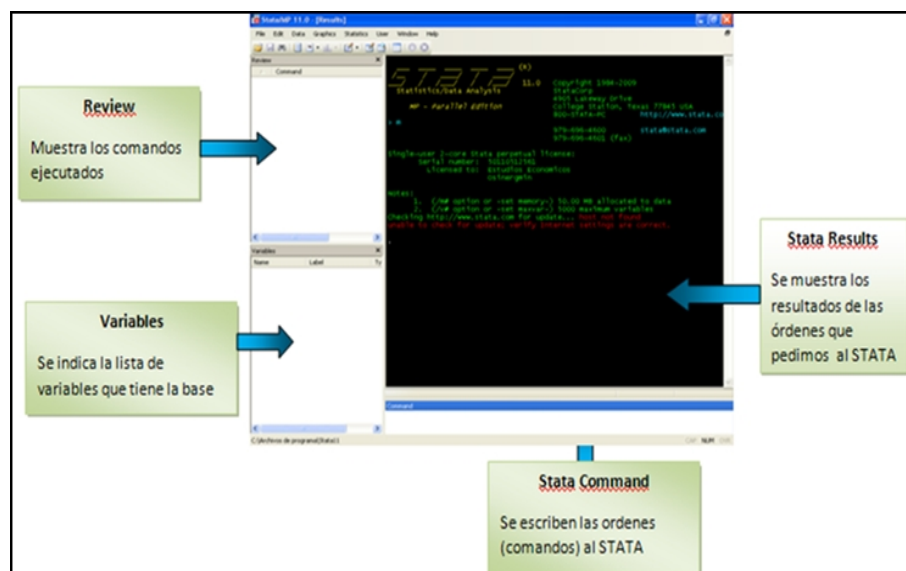


Figura 2.1: Entorno del STATA 11



Otras ventanas a tomar en consideración son:

1. **STATA Viewer:** Podemos acceder a la información online y a las ayudas que nos otorga el programa.
2. **STATA Do-File Editor:** Es una ventana que funciona como editor de texto para poder guardar y ejecutar una lista de comandos programados.
3. **STATA Data Editor:** Nos permite digitar y modificar los datos de la misma forma que una hoja de Excel.
4. **STATA Browser:** Accedemos a la ventana de datos sin poder modificar su contenido.
5. **STATA Graphs:** Nos muestra una ventana con el gráfico que hemos ejecutado.

En el siguiente capítulo enseñaremos como acceder a ellas a través de la barra de herramientas

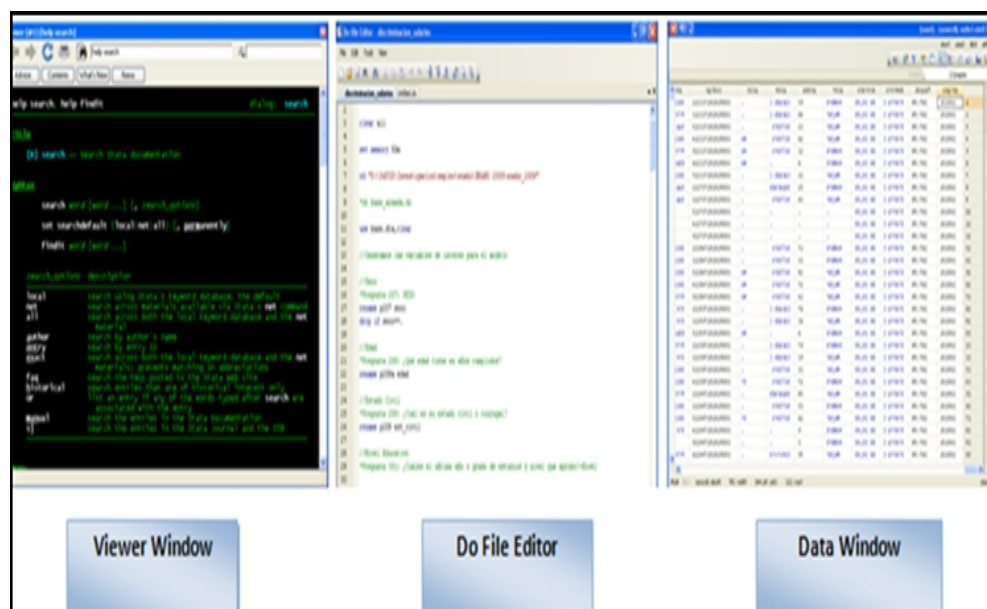


Figura 2.2: Viewer Windows, Do-File, Data Windows

## 2.2. La Barra de Herramientas

La barra de herramientas nos permite realizar operaciones rutinarias como abrir, guardar, imprimir algún archivo, además de otras particularidades.

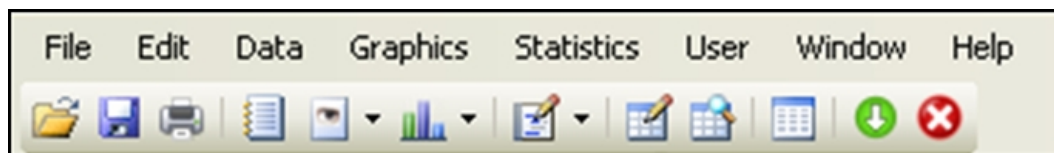


Figura 2.3: La Barra de Herramientas

A continuación se explicará la función de cada uno de los elementos de la Barra de Herramientas:



Nos permite abrir una base de datos con extensión **\*.dta**.



Nos permite guardar una base de datos que está siendo utilizada.



Nos permite imprimir el contenido registrado en la ventana del *Stata Result*.



Nos permite iniciar, cerrar, suspender o resumir una bitácora (la cual se guardan con extensión **\*.log** o **\*.smcl**). Es útil para guardar los resultados mostrados por el *Stata Result*.



Nos muestra la ventana del *Stata Viewer* oculta.



Nos indica la ventana del *Stata Graphic*.



Nos permite iniciar el uso del *Stata Do-File*.



Nos permite abrir la ventana del *Stata Editor* que está oculta.



Nos permite abrir la ventana del *Stata Browser* que está oculta.



Ordena al Stata continuar la ejecución de un comando que fue detenido.



Ordena al Stata detener la ejecución de un comando.

## 2.3. Tipos de Archivo

Stata reconoce 4 tipos de archivos:

1. **Archivo \*.dta** : Lee base de datos del entorno de STATA.
2. **Archivo \*.do** : Lee el Do-File, la cual contiene una serie de comandos y/o funciones.
3. **Archivo \*.log** : Guarda los resultados que arroja el STATA, también llamado bitácora.
4. **Archivo \*.gph** : Guarda los gráficos creados en el STATA.

## 2.4. Sintáxis de los Comandos del STATA

Describe la estructura básica de los comandos del lenguaje de programación de Stata.

$[prefix :]$ <b>command</b> $[varlist]$ $[if\ expr]$ $[in]$ $[weight]$ <b>using</b> $filename$ $[, options]$
--

Donde:

- *prefix*: Permite repetir las ejecuciones de un determinado comando o modificar el input y/o output de la base de datos.
- *command*: Indica el comando del STATA.
- *varlist*: Indica la lista de nombres de variables.
- *weight*: Indica la variable de ponderación.
- *if*: Indica una expresión lógica condicional.
- *exp*: Indica la expresión matemática utilizada para la condicional.

- *in*: Señala el rango de observaciones que queremos analizar.
- *filename*: Señala el nombre del archivo.
- *options*: Señala una o más opciones que aplica el comando.

Utilizando el comando **help language** podemos obtener mayor información de cada uno de sus componentes.

```
. help language
```

## 2.5. Expresiones Lógicas del STATA

Las siguientes expresiones nos servirán para la programación en STATA.

Expresiones Lógicas	Descripción
<code>==,!=</code>	<i>Igual a, Diferente a</i>
<code>&lt;,&gt;,&lt;=,&gt;=</code>	<i>Menor que, Mayor que, Menor igual que, Mayor igual que</i>
<code>&amp;, </code>	<i>Y-AND,O-OR</i>
Operadores Matemáticos	Descripción
<code>+, -, *, /, ^</code>	<i>Suma, Resta, Multiplicación, División, Potencia</i>
Funciones Matemáticas	Descripción
<code>abs(x)</code>	<i>Retorna el valor absoluto de una variable</i>
<code>exp(x)</code>	<i>Retorna el exponencial de una variable</i>
<code>log10(x)</code>	<i>Retorna el logaritmo en base 10 de una variable</i>
<code>ln(x)</code>	<i>Retorna el logaritmo neperiano de una variable</i>
<code>mean(x)</code>	<i>Retorna la media de una variable</i>
<code>median(x)</code>	<i>Retorna la mediana de una variable</i>
<code>mode(x)</code>	<i>Retorna la moda de una variable</i>
<code>sqrt(x)</code>	<i>Retorna la raíz cuadrada de una variable</i>
<code>sum(x)</code>	<i>Retorna la suma de elementos de una variable</i>
<code>uniform(a,b)</code>	<i>Retorna una variable aleatoria uniforme comprendido entre a y b</i>
<code>rnormal(a,b)</code>	<i>Retorna una variable aleatoria normal con media a y desviación estándar b</i>

Figura 2.4: Expresiones Lógicas

## 2.6. Organizando un Proyecto de Trabajo

Al momento de trabajar con STATA (específicamente en un archivo **Do-file**) es recomendable mantener el siguiente esquema de trabajo:

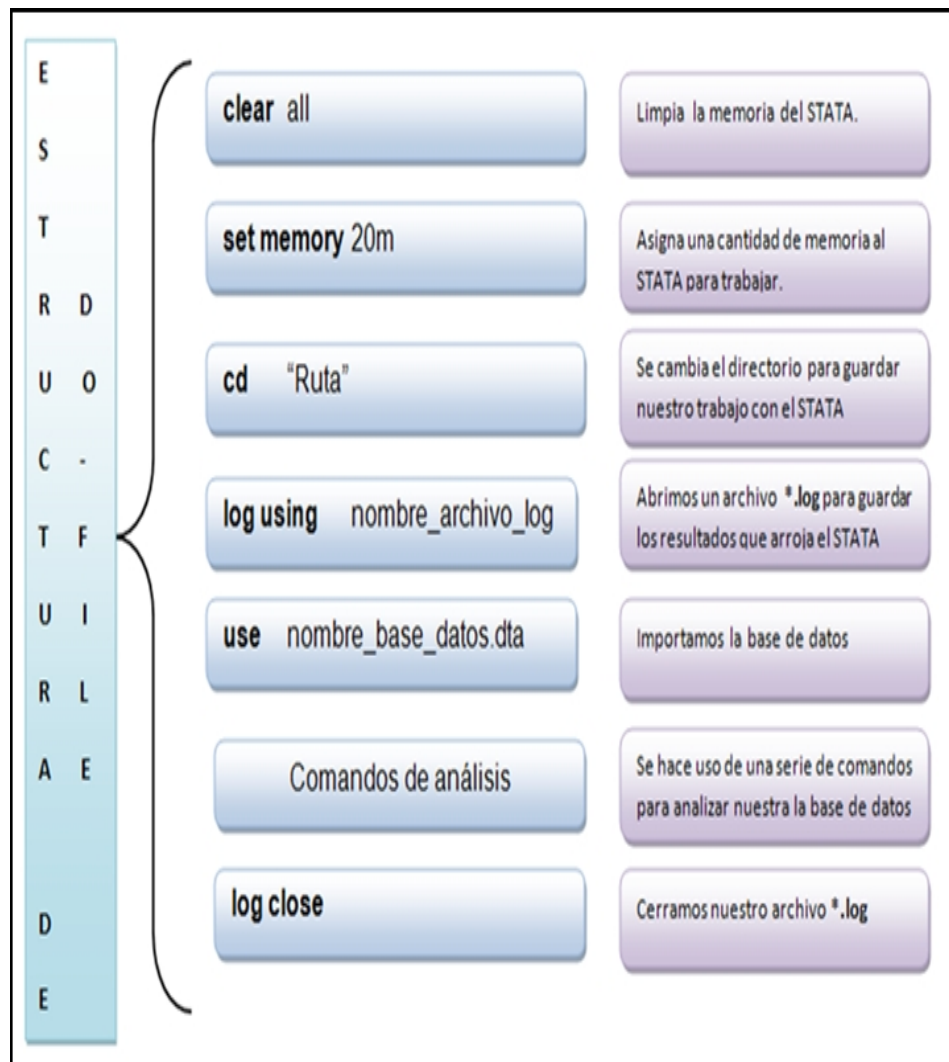


Figura 2.5: Esquema de un Proyecto de Trabajo

## 2.7. Recursos del STATA

STATA cuenta con una documentación extensa la cual puede encontrarse en el mismo software así como también en la web.

1. **Guide's User STATA:** La guía de usuario se accede en la barra de herramientas a través de la siguiente ruta: *Help* —> *PDF Documentation*. Esta guía es muy importante para los usuarios que comienzan a trabajar con el STATA.
2. **STATA Journal (SJ) y STATA Technical Bulletin (STB):** Presentan documentación detallada acerca de nuevos comandos que no están incluidos en el software. El SJ pueden ser descargados por la web siempre y cuando presenten más de 3 años de antigüedad, mientras que el STB siempre está disponible online.
3. **Otras Fuentes:**

<http://www.stata.com/support>

Incluye un resumen de lo que hace el STATA. En particular se recomienda ver la parte de respuestas: FREQUENTLY ASKED QUESTION (FAQs).

<http://www.ats.ucla.edu/stat/stata/>

Provee diversos tutoriales y videos para aprender STATA.

## 2.8. Comandos de Ayuda

Existen diversos comandos que sirven como ayuda para el manejo de STATA, entre ellas tenemos:

- **help** : Es muy útil si se conoce el nombre comando para la cual se necesita ayuda.

```
. help regress
```

- **search** : Busca una palabra clave “keyword” en los archivos oficiales de ayuda, FAQs, examples, the SJ y el STB, pero no del internet.

```
. search ols
```

- **net search** : Busca en Internet paquetes instalables, incluyendo códigos del SJ y el STB.

```
. net search random effect
```

- **hsearch** : Busca el “keyword” en todos los archivos de ayuda (con extensión **\*.sthlp** o **\*.hlp**). El inconveniente es que se necesita el “keyword” completo.

```
. hsearch weak instrument
```

- **findit** : Provee la más amplia búsqueda del “keyword” con información relacionado al STATA. Es útil ya que no se necesita especificar el “keyword” en su forma completa.

```
. findit weak inst
```

## 2.9. Instalación de Nuevos Comandos

Durante el desarrollo de los temas estudiados en este manual, se hará necesario emplear diversos comandos que el software no cuenta en un inicio y que son programados por usuarios libres el cual deben ser descargado a través de la web. Estos comandos se guardan en archivos con extensión **\*.ado**.

Una manera sencilla de realizar este procedimientos es a través del comando **update all**, el cual permite actualizar una lista de archivos *ado*. Los archivos descargados se guardan en la carpeta que se ubica el software. para acudir a este comando escribimos la siguiente sintáxis:

```
. update all
```

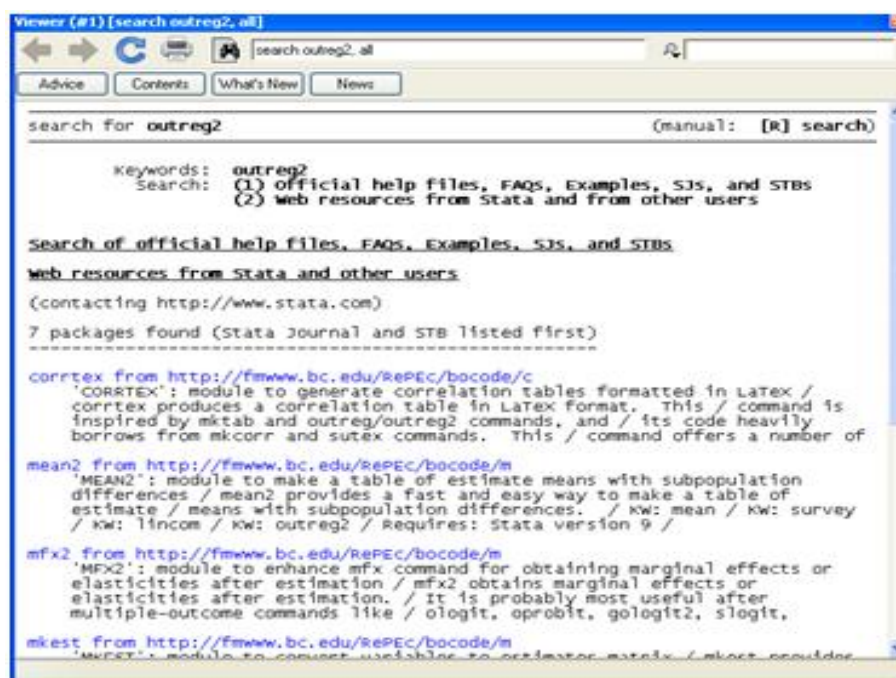
Hay que esperar unos minutos que se descargar todos los archivos de programación.

Otra de las formas más comunes para realizar este procedimiento es utilizar el comando **findit** siempre y cuando se conozca el nombre del comando que se busca. Por ejemplo, supongamos que queremos instalar el comando **outreg2.ado** para elaborar una mejor presentación de los resultados de nuestras estimaciones.

Entonces, deberíamos escribir en la ventana de comandos la siguiente sintaxis:

```
. findit outreg2
```

Nos saldrá la siguiente ventana:

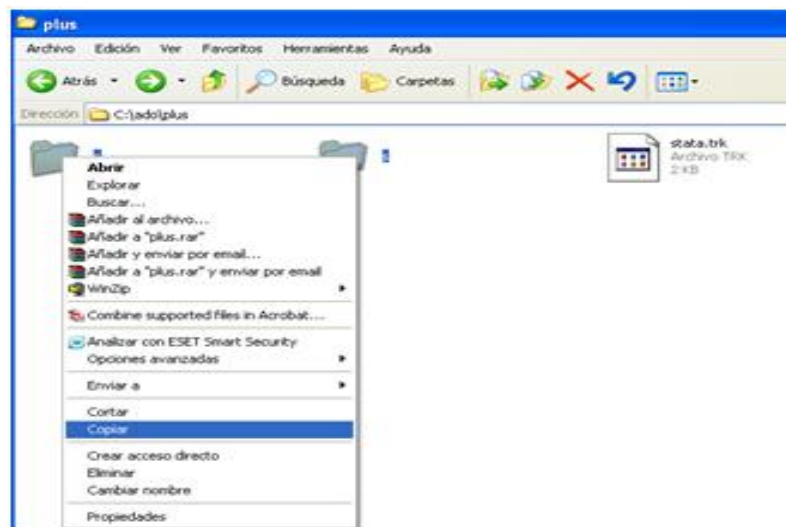


A continuación, hacemos click en [outreg2 from http://fmwww.bc.edu/RePEc/bocode/o](http://fmwww.bc.edu/RePEc/bocode/o) y se observará lo siguiente:

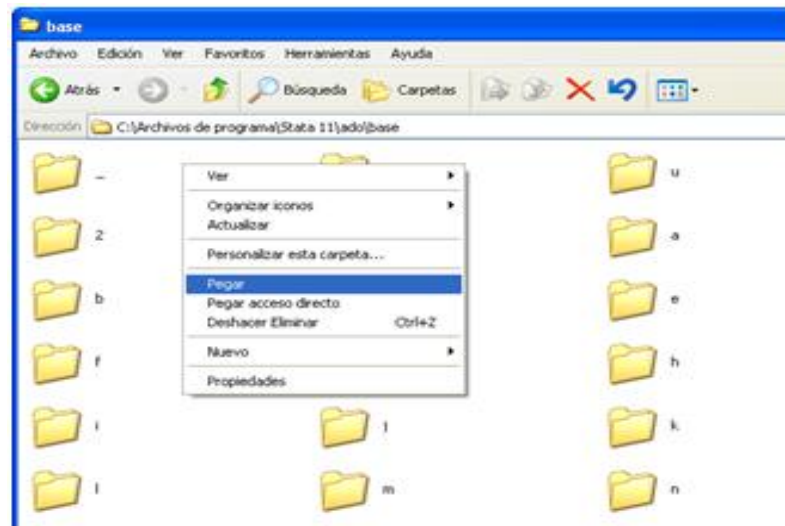




Damos click en la opción *click here to install* y esperamos unos segundos a que se instale el comando. El comando instalado se ubica por default en la ruta *C:\ado\plus* como se puede apreciar en la siguiente figura:



Finalmente, hay que copiar las carpetas con *nombres de letras*, según como inicia el nombre de cada uno de los comandos descargados, y pegarlos en el interior de la carpeta *base* que se ubica dentro del software en la siguiente ruta *C:\Archivos de programa\Stata 11\ado\base*.



Otras herramientas que permiten instalar paquetes de comandos desde la web es el **ssc install** y el **net install**, el cual es necesario tener conocimiento del nombre del paquete que se quiere instalar. En ambos casos, los nuevos comandos se descargan en la ruta por defecto de *C:\ado\plus*.

```
. net install outreg2  
. ssc install outreg2
```

## 2.10. Ejercicio Propuesto

Por medio de los comandos de ayuda, descarge y explique en que consiste la siguiente lista de comandos:

- `usespss`
- `lmhgl`
- `fgtest`
- `xttest3`
- `xtcsd`
- `sim_arma`



# Capítulo 3

## Gestión de Base de Datos


Aprenderemos en qué consiste una sesión de trabajo en STATA y exploraremos algunos comandos que nos permitirán realizar un análisis de base de datos haciendo uso del Do-file. Para dicho fin, explicaremos el funcionamiento de esta herramienta.

### 3.1. El Do-File

STATA cuenta con una ventana que nos permite trabajar con una serie de comandos y almacenarlas. Estos archivos son muy importantes por los siguientes motivos:

- Permite registrar una serie de comandos, la cual representa todo el procedimiento de nuestro trabajo.
- Permite ir corrigiendo posibles errores que se pueden generar en la elaboración y ejecución de nuestro trabajo.
- Permite replicar los procedimientos en sesiones posteriores sin necesidad de crearlo nuevamente.
- Además, sirve como un mecanismo de seguridad que permite regresar a la base de datos original después de haberle hecho diversas transformaciones.

Para acceder al Do-file hacemos clic al ícono correspondiente en la barra de herramientas o simplemente presionamos la siguiente sucesión de teclas **Ctrl+8**. Recuerde que el archivo Do-file se guarda con extensión **\*.do**.

Con respecto a las formas de poder ejecutar los comandos, se puede hacer a través del ícono  (*execute do*) ubicado en la parte superior derecha de la barra de herramientas del archivo Do-file o presionando los teclados **Ctrl+D** una vez que sombreemos el comando queramos correr. Una vez realizada esta acción, se reflejará los resultados en la ventana Result View del STATA.

### 3.1.1. Comentarios en el Do-File

El Do-file puede incluir comentarios incrementando el entendimiento de un programa o archivo de trabajo. Existen diferentes formas de incluir un comentario:

- Una simple línea de comentario empieza con un asterisco (\*); donde STATA ignorará tales líneas.
- Para colocar un comentario en la misma línea donde fue escrito el comando utilizamos dos slash (//).
- Para líneas con múltiples comentarios, colocamos el texto entre los símbolos (/\*) al inicio y (\*/) al final.
- En el caso de que se haga uso de un comando la cual presenta una expresión muy larga podemos utilizar triple slash (///) en medio de la expresión y así continuar en la siguiente línea la parte faltante. STATA entenderá como si fuera una única línea de comando.
- Por último, también se utilizan los símbolos de comentarios con fines decorativo.

```
*****
** MI PRIMER DO FILE **
*****

*En este capítulo elaboraremos nuestro primer Do-File

/*
```

```
CURSO: ECONOMETRIA
```

```
FACULTAD: ECONOMÍA
```

```
*/
```

Como se podrá apreciar en el Do-file elaborado , los comentarios se registran con color **verde**.

## 3.2. Iniciando la Estructura de un Do-File

Como se explicó en el esquema usual de un do-file, esta empieza con el comando **clear**. Este comando nos permite limpiar por inercia una base de datos y etiquetas existentes en la memoria del STATA.

```
. clear //Limpiamos alguna base de dato que estuviese cargada.
```

Es importante saber que este comando presenta algunas opciones que se mostrarán a continuación:

- Para remover funciones del MATA<sup>1</sup>, además de matrices, programas y archivos \*.ado, se puede usar la siguiente sintaxis:

```
clear [, mata | results | matrix | programs | ado]
```

- Si se desea borrar todo de una sola vez, se usa la siguiente sintaxis:

```
clear all
```

---

<sup>1</sup>**MATA** es un lenguaje de programación matricial que puede ser usado por quienes desean calcular iteraciones en un entorno de matrices.

### 3.3. Asignando Memoria

Generalmente se suele trabajar con una capacidad de memoria de 20m (megabyte), pero en este caso utilizaremos 100m. Para realizar esta operación escribimos lo siguiente<sup>2</sup>:

```
. set memory 100m //Establecemos una memoria de 100 megabyte
```

#### Current memory allocation

settable	current value	description	memory usage (1M = 1024k)
set maxvar	5000	max. variables allowed	1.909M
set memory	100M	max. data space	100.000M
set matsize	400	max. RHS vars in models	1.254M
			103.163M

### 3.4. Manejo de Directorios

Cuando se inicia una sesión en STATA, por defecto se trabaja en la carpeta en donde se encuentra ubicado el software. Si por ejemplo, el software se ubicase en la ruta *C:\Archivos de Programas*, entonces la carpeta de trabajo o directorio se encontrará en la siguiente ruta *C:\Archivos de Programas\Stata 11*. Para saber en qué directorio se está trabajando actualmente utilizamos el siguiente comando **pwd**:

```
. pwd //Este es el directorio actual donde se está trabajando.
C:\Archivos de Programas\Stata 11
```

---

<sup>2</sup>Para saber cuáles son las diversas opciones que presenta el STATA para trabajar con diferentes tamaños de base de datos recurra al comando **help set** eligiendo la opción memory o simplemente escriba **help memory**. Sin embargo, para no establecer cualquier cifra para la memoria, Cameron & Trivedi recomiendan asignar una cantidad de memoria igual a 1.5 veces el peso de la base de datos, con el fin de que el STATA no elimine variable u observaciones, no disminuya el rendimiento de la computadora y pueda generar nuevas variables, estimaciones, guardar gráficos, etc. Es decir, si la base de datos pesa 50m, entonces, deberá asignarse una memoria de 75m (1.5x(50m)).

También es posible saber lo anterior viendo la parte inferior izquierda del entorno del STATA.

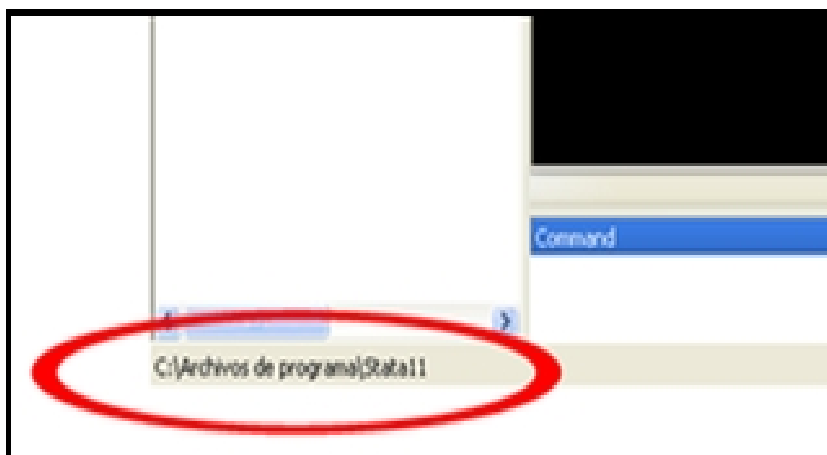


Figura 3.1: Manejo de Directorio

Supongamos que en el disco *D:* creamos una carpeta con el nombre **Econometría-Stata** la cual nos va a servir para guardar nuestros trabajos, entonces, el nuevo directorio se encontraría en la siguiente ruta *D:\Econometria*. Ahora, utilizamos el comando **cd** para cambiarnos al nuevo directorio creado, indicando la nueva ruta entre comillas:

```
. cd "D:\Econometria-Stata" //Nos cambiamos al nuevo directorio de trabajo.  
D:\Econometria-Stata
```

### 3.5. Guardar Resultados en Bitácoras

Los resultados que arroja STATA en la ventana de resultados puede ser almacenados en una bitácora, el cual se guarda en archivos con extensión **\*.log**, **\*.smcl** o **\*.text**.

Para realizar este procedimiento, se emplea el comando **log**<sup>3</sup> el cual presenta la siguiente sintáxis:

---

<sup>3</sup>Para una descripción completa del funcionamiento de este comando puede escribir en la ventana de comando **help log**.



- Si desea crear una bitácora:

```
log using nombre_bitacora [, [text | smcl ]]
```

- Para dejar de registrar momentáneamente los resultados:

```
log off
```

- Para volver a registrar los resultados :

```
log on
```

- Para cerrar la bitácora:

```
log close
```

- Para reanudar la bitácora:

```
log using nombre_bitacora, append
```

- Para sobrescribir en la bitácora<sup>4</sup>:

```
log using nombre_bitacora, replace
```

- Para observar una bitácora ya elaborada en el Result View:

```
type nombre_bitacora
```

Para el ejemplo que estamos siguiendo, se puede estructurar el Do-file de la siguiente manera:

```
*Creamos nuestra primera bitácora con extensión *.smcl
log using primera.bitacora, replace smcl

*Este comentario se grabará en la bitacora

log off //Dejamos de registrar momentáneamente los resultados

*Este comentario no se guardará en la bitácora

log on //Volvemos a registrar los resultados

*Este comentario se volverá a grabar en la bitacora

log close //Cerramos la bitácora
```

---

<sup>4</sup>Es importante usar siempre esta opción cuando se crea una bitácora para poder ejecutar el Do-File sin problemas en posteriores sesiones. Si no se usa esta opción es probable que salga el siguiente error: **log file already open**.

```
*Este comentario ya no se grabará en la bitácora

log using primera.bitacora,append //Reanudamos a grabar en la bitácora

*Este comentario se grabará en la bitácora reanudada

log close

*Vemos lo que grabó la bitácora
type primera_bitacora.smcl
```

---

```
name: <unnamed>
log: D:\Econometria-Stata\primera_bitacora.smcl
log type: smcl
opened on: 14 Feb 2012, 00:15:43
```

```
. *Este comentario se grabará en la bitacora

. log off //Dejamos de registrar momentáneamente los resultados
name: <unnamed>
log: D:\Econometria-Stata\primera_bitacora.smcl
log type: smcl
paused on: 14 Feb 2012, 00:15:43
```

---

```
name: <unnamed>
log: D:\Econometria-Stata\primera_bitacora.smcl
log type: smcl
resumed on: 14 Feb 2012, 00:15:43
```

```
. *Este comentario se volverá a grabar en la bitacora

. log close //Cerramos la bitácora
name: <unnamed>
log: D:\Econometria-Stata\primera_bitacora.smcl
log type: smcl
closed on: 14 Feb 2012, 00:15:43
```

---


```
name: <unnamed>
log: D:\Econometria-Stata\primera_bitacora.smcl
log type: smcl
opened on: 14 Feb 2012, 00:15:43
```

```
. *Este comentario se grabará en la bitácora reanudada

. log close
name: <unnamed>
log: D:\Econometria-Stata\primera_bitacora.smcl
log type: smcl
closed on: 14 Feb 2012, 00:15:43
```

---

Si revisamos nuestra carpeta de trabajo, se observa que se creó un nuevo archivo con el nombre *primera\_bitacora*. Para ver el contenido de este archivo,

se debe ir a la ventana principal del STATA y hacer clic en el ícono  (*Log Begin/Close/Suspend/Resume*) y buscar el archivo correspondiente.

**Importante:** Se sugiere tener en cuenta dos cosas al momento de usar este comando:


- Al momento de crear una bitácora con el comando **log using**, siempre se recomienda usar la opción **replace**. Esto porque al ejecutar el Do-file más de una vez, STATA puede arrojar un mensaje de error diciendo que la bitácora ya está creada.
- Siempre que se crea una bitácora, no se debe olvidar colocar al final de la grabación el comando **log close**.

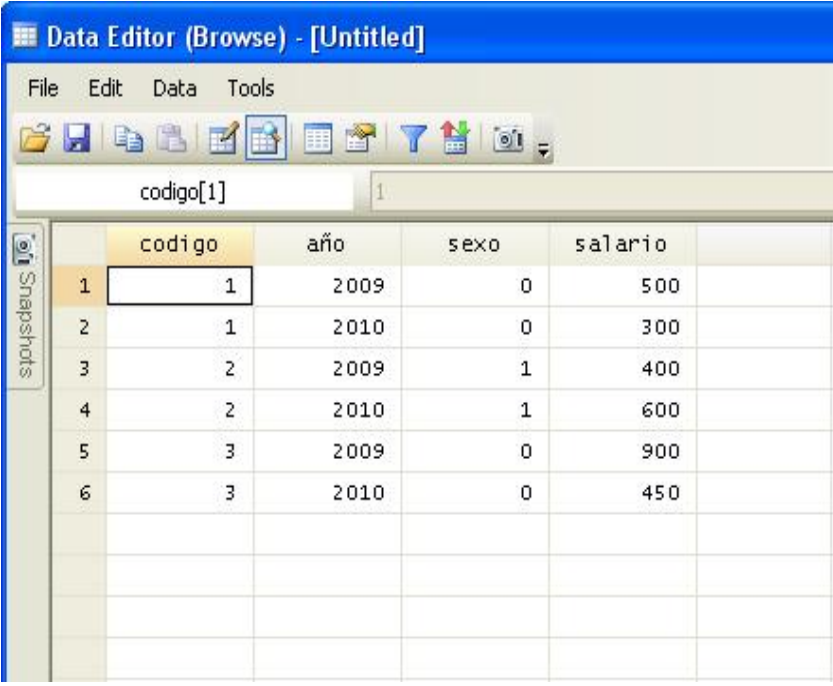
## 3.6. Creando Base de Datos

La manera de editar manualmente una base de datos a través del Do-file es a través del comando **input**.

```
*Creamos una base de datos
. input codigo año sexo salario
      codigo      año      sexo      salario
1      2009      0      500
1      2010      0      300
2      2009      1      400
2      2010      1      600
3      2009      0      900
3      2010      0      450
end

. browse
```

Para observar la base de datos elaborada se debe hacer clic al ícono  (*Data Editor Browse*) de la ventana principal del STATA o en su defecto escribir en la ventana de comando **browse**.



The screenshot shows the STATA Data Editor window titled "Data Editor (Browse) - [Untitled]". It has a menu bar with "File", "Edit", "Data", and "Tools". Below the menu is a toolbar with various icons. A text box at the top shows "codigo[1]" and "1". The main area is a table with 6 rows and 5 columns. The columns are labeled "codigo", "año", "sexo", "salario", and an empty column. The first row is highlighted in orange.

	codigo	año	sexo	salario	
1	1	2009	0	500	
2	1	2010	0	300	
3	2	2009	1	400	
4	2	2010	1	600	
5	3	2009	0	900	
6	3	2010	0	450	

Figura 3.2: Editando Base de Datos

## 3.7. Cargando Base de Datos

Existen diferentes formas de cargar una base de datos, ya sea en formato **.dta** o en otros formatos (**\*.txt** , **\*.xls** , **\*.sav**, etc).

### 3.7.1. Abriendo base de datos del STATA.

Para abrir una base de datos desde la ventana principal del STATA debemos acceder a la siguiente ruta: *File* —> *Open*. Luego aparecerá un cuadro de diálogo para buscar y elegir la base de datos que deseamos trabajar.

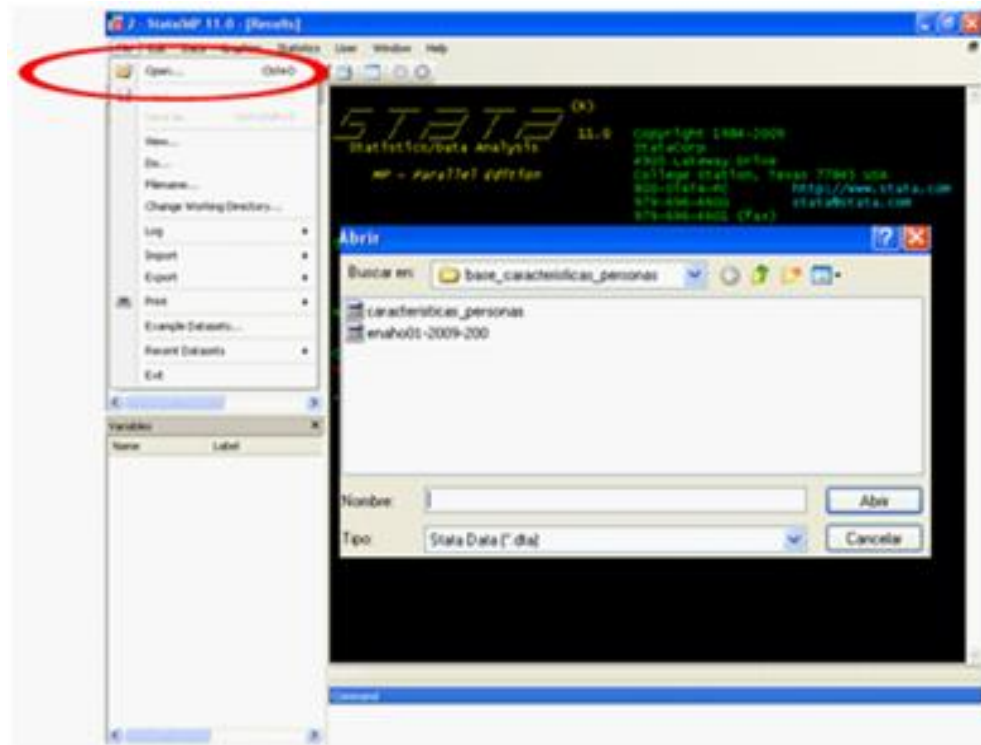


Figura 3.3: Cargando Base de Datos

STATA cuenta con bases de datos dentro de su sistema como ejemplos aplicativos, para cargarlos se utiliza el comando **sysuse**.

```
*Cargamos una base de datos del sistema del STATA
. clear all

. sysuse auto.dta
(1978 Automobile Data)
```

Si deseamos cargar una base de datos propia, basta con guardarlo en el directorio actual que se está trabajando y cargarlo usando el comando **use**. En este caso, cargaremos la base de datos denominada *enaho01-2010-100.dta* de la siguiente manera:

```
. *Cargamos una base de datos de la carpeta de trabajo
. clear all

. use enaho01-2010-100.dta
```



### 3.7.2. Importando Base de Datos

Como caso aplicativo, se ha descargado de la base de información del Banco Central de Reservas del Perú (BCRP) correspondiente al Índice General de la Bolsa de Valores de Lima desde enero de 1992 hasta Junio de 2012, donde el archivo que se descarga por defecto es en excel y se ha guardado con el nombre de *“iqbvl\_mensual.xls”*.

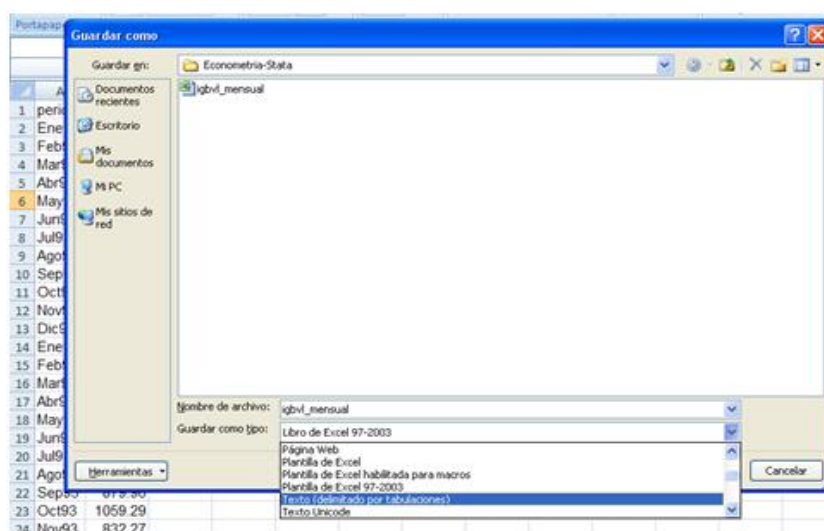
	A	B	C	D	E	F	G
1	Mes/Año	IGB (dic. 1991=100)					
2	Ene92	108.55					
3	Feb92	170.57					
4	Mar92	158.81					
5	Abr92	144.83					
6	May92	138.55					
7	Jun92	121.21					
8	Jul92	121.9					
9	Ago92	122.01					
10	Sep92	148.82					
11	Oct92	207.76					
12	Nov92	231.37					
13	Dic92	372.95					
14	Ene93	330.61					
15	Feb93	380.76					
16	Mar93	487.86					

Para importar esta base de datos es importante mencionar que la primera fila de la hoja de cálculo se registre el nombre de las variables de la forma más sencilla <sup>5</sup>. En nuestro caso, el nombre de las variables ubicada en la primera fila será *periodo* e *igbv*, y a partir de la segunda fila se comienzan a registrar los datos.

	A	B	C	D	E	F	G
1	periodo igbv						
2	Ene92	108.55					
3	Feb92	170.57					
4	Mar92	158.81					
5	Abr92	144.83					
6	May92	138.55					
7	Jun92	121.21					
8	Jul92	121.9					
9	Ago92	122.01					
10	Sep92	148.82					
11	Oct92	207.76					
12	Nov92	231.37					
13	Dic92	372.95					
14	Ene93	330.61					
15	Feb93	380.76					
16	Mar93	487.86					
17	Abr93	496.67					

Luego, dicha base lo guardamos en nuestra carpeta de trabajo “D:\Econometria I” con formato **Texto (delimitado por tabulaciones)** o **csv (delimitado por comas)**.

<sup>5</sup>Se recomienda designar un nombre corto y sin dejar espacios entre palabras. Además, la base de datos a importar debe de comenzar desde la celda A1



Finalmente utilizamos el comando **insheet** para importar la base de datos como se indica a continuación:

```
. Importamos una base de datos desde excel

. //Si fue guardado como delimitado por comas
. clear all
. insheet using igbvl_mensual.csv, delimiter(", ;")

. // Si fue guardado como delimitado por tabulaciones
. clear all
. insheet using igbvl_mensual.txt, tab
```

Por último, esta guía trabajará en parte con bases de datos proveniente de la Encuesta Nacional de Hogares (ENAH) que pueden ser descargados del Instituto Nacional de Estadística e Informática (INEI), el cual están guardados con formatos del SPSS (\*.sav). Para poder cargar una base de datos con este tipo de formato directamente en el STATA usamos el comando **usespss**.

```
. *Cargamos una base de datos de la carpeta de trabajo con formato *.sav
. usespss using " Enaho01-2010-100.sav ",clear
```



### 3.7.3. Convertir Base de Datos

STATA cuenta con una herramienta que permite convertir base de datos de SPSS, Matlab, Gauss, SAS, Excel, etc. al formato **\*.dta** a través del software **STAT/TRANSFER**.



Figura 3.5: STAT TRANSFER

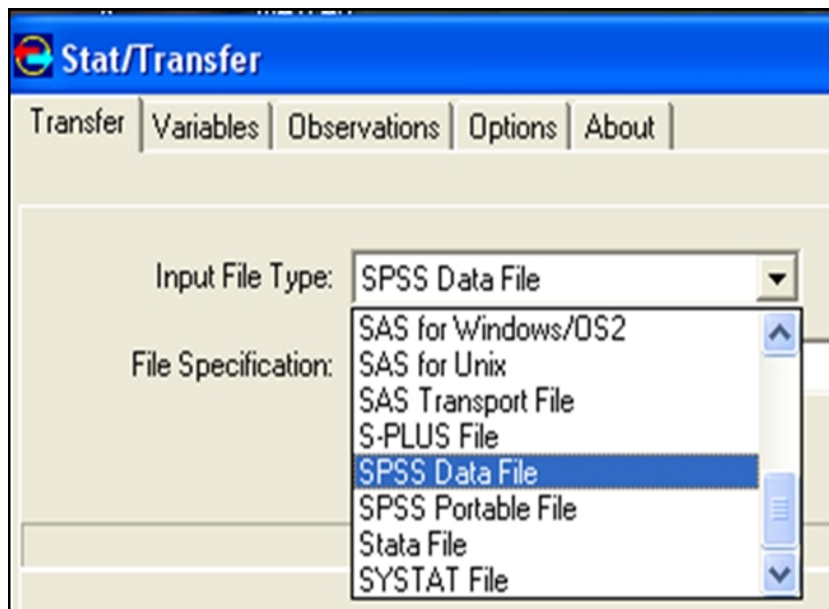
Para acceder a este software basta con hacerle clic y posteriormente nos saldrá una ventana de dialogo solicitándonos la siguiente información:

- *Input File Type* : Indicamos el tipo de archivo en la cual se encuentra nuestra base de datos original.
- *File Specification* : Indicamos la ruta donde se encuentra nuestra base de datos original haciendo uso del botón Browse.
- *Output File Type* : Indicamos el tipo de archivo al cual deseamos que la base de datos se convierta.
- *File Specification* : Indicamos la ruta donde queremos colocar la base de datos convertida haciendo uso del botón Browse.

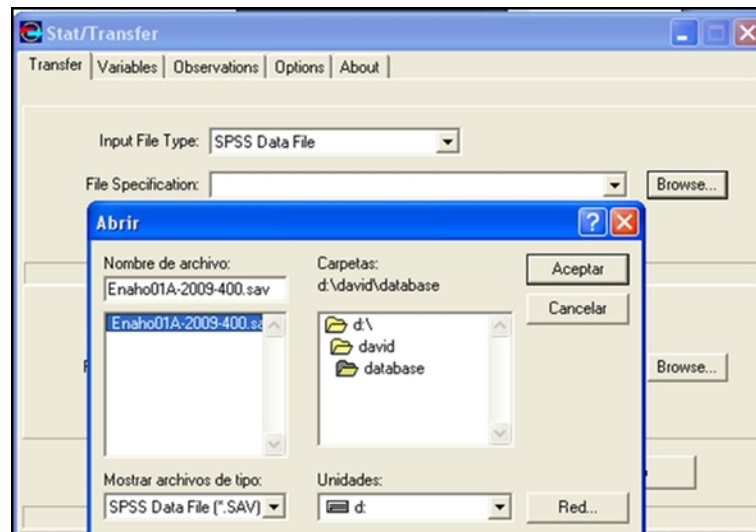
Para nuestro caso ilustrativo, contamos con la base de datos de la Enaho en formato de SPSS llamada “SUMARIA-2010.sav” que se encuentra en nuestra carpeta de trabajo, el cual queremos convertirlo a un archivo de base de datos del STATA con el mismo nombre y que se guarde en la misma carpeta de trabajo.

Para desarrollar esta aplicación realizamos los siguientes pasos:

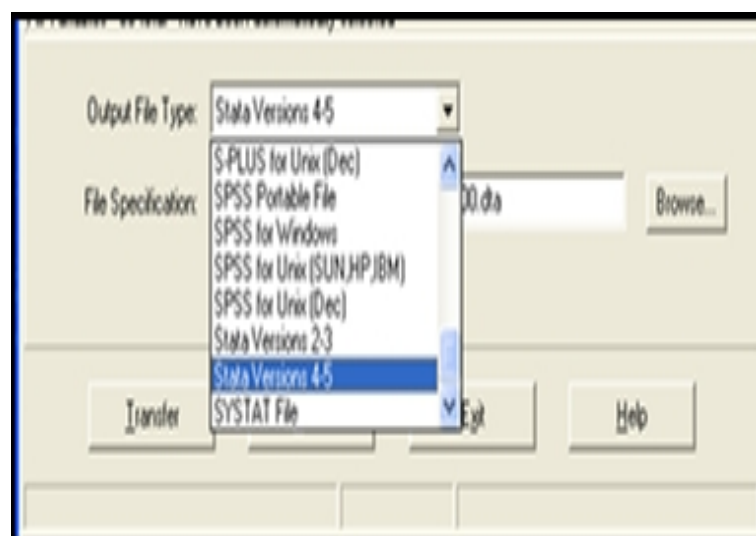
1. Abrimos la ventana de diálogo del *STAT/TRANSFER*.
2. En la sección *Input File Type* hacemos clic a la barra desplegable y elegimos el formato **SPSS Data File** ya que se debe un archivo de base de datos del *SPSS*.



3. En la sección *File Specification* hacemos clic en el botón Browse para definir la ruta donde se encuentra nuestra base original. Observe que en la barra “Unidades” (ubicado en la parte inferior derecha) escogemos el disco **D**, En la barra “Mostrar Archivos Tipos” (ubicado en la parte inferior izquierda) por default se muestra **SPSS Data File (\*.sav)** En el cuadro “Carpetas” hacemos clic en las carpetas según como se señala la ruta del archivo de origen **D:\Econometria-Stata**. Finalmente hacemos clic en el archivo original llamado *SUMARIA-2010.sav*.

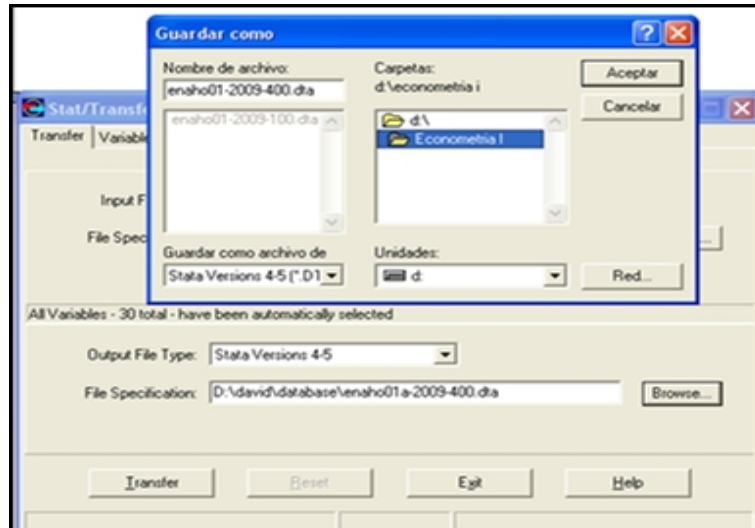


4. En la sección *Output File Type* hacemos clic a la barra desplegable y elegimos el formato **Stata Version 4-5** la cual es el formato de la base de datos que queremos obtener.

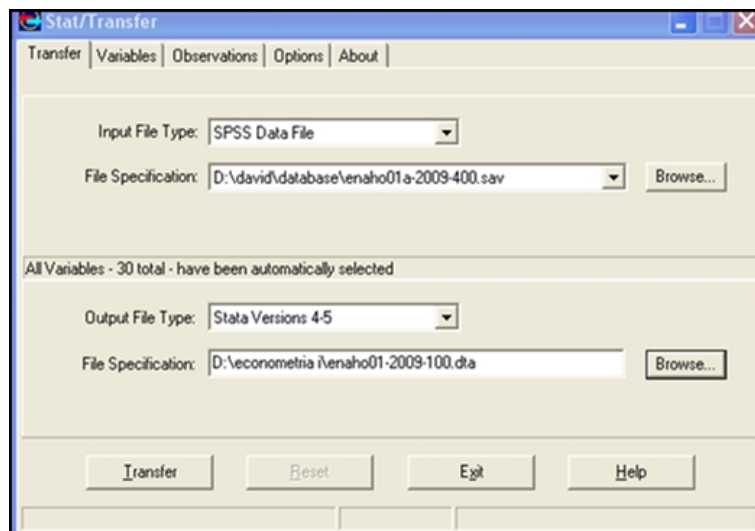


5. En la sección *File Specification* hacemos clic en el boton Browse para definir la ruta donde queremos que se encuentre nuestra base convertida. Observe que en la barra “Unidades” (ubicado en la parte inferior derecha) escogemos el disco **D**, En la barra “Mostrar Archivos Tipos” (ubicado en la parte inferior izquierda) por default se muestra **Stata version 4-5 (\*.dta)**. En el

cuadro “Carpetas” hacemos clic en las carpetas según como se señala la ruta donde se guardará la base convertida **D:\Econometria-Stata**. Finalmente escribimos el nombre de nuestra nueva base, este caso *sumaria-2010.dta*.



6. Finalmente hacemos clic en el botón **Transfer**.



Esperamos unos segundos mientras el programa está convirtiendo la base de datos.

## 3.8. Guardar Base de Datos

Una vez trabajado y modificado la base de datos es posible guardarlo con el comando **save**.

```
. *Cargamos una base de datos de la carpeta de trabajo con formato *.sav
. usespss using " Enaho01-2010-100.sav ",clear

. *Guardamos la base de datos con formato *.dta
. save modulo100-2010.dta, replace
file modulo100-2010.dta saved
```

**Importante:** Al guardar una base con el comando **save**, siempre se recomienda usar la opción **replace**. Esto porque al ejecutar el Do-file más de una vez, STATA puede arrojar un mensaje de error diciendo que ya existe el archivo.

## 3.9. Inspección Base de Datos

En esta sección aprenderemos comandos que nos permitan dar una revisada a la base de datos, es decir, saber con qué esquema de datos y tipos de variables que estamos trabajando.

Usualmente después de abrir una base de datos, recurrimos a la siguiente rutina de inspección de una base de datos:

- Descripción de la base de datos con el comando **describe**.
- Observar la base de a través del comando **browse** o **edit**. El comando *browse* nos permite ver la base de datos sin poder modificarla y el comando *edit* nos permite ver la base de datos pudiendo modificarlo.
- Inspeccionar las variables de la base de datos usando el comando **inspect**.
- Generamos un diccionario de variables con el comando **codebook**.
- A veces podría ser de ayuda hacer una lista de los valores de algunas variable de interés para un determinado rango de observaciones a través del comando **list**.

- Realizar un cuadro estadístico resumen de diferentes variables numéricas con el comando **summarize**.

```
. *Cargamos la base "enaho01-2010-100.dta"
. use enaho01-2010-100.dta, clear
```

```
. *Describimos la base modulo100-2010.dta
. d p1141 p1142 p1143 p1144 // describir algunas variables
```

variable name	storage type	display format	value label	variable label
p1141	byte	%8.0g	p1141	su hogar tiene : teléfono (fijo)
p1142	byte	%8.0g	p1142	su hogar tiene : celular
p1143	byte	%8.0g	p1143	su hogar tiene : tv. cable
p1144	byte	%8.0g	p1144	su hogar tiene : internet

```
. *Vemos la ventana de la base de datos
. browse // para observar el Data Window sin modificar
. br p1141 p1142 p1143 p1144 // observar algunas variables
```

```
. *Inspeccionamos las variables
. ins p1141 p1142 p1143 p1144 // inspeccionar algunas variables
```

p1141: su hogar tiene : teléfono (fijo)		Number of Observations		
		Total	Integers	Nonintegers
#	Negative	-	-	-
#	Zero	17059	17059	-
#	Positive	4437	4437	-
#				
#	Total	21496	21496	-
#	Missing	5680		
0		27176		
1				
(2 unique values)				

p1141 is labeled and all values are documented in the label.

p1142: su hogar tiene : celular			Number of Observations		
			Total	Integers	Nonintegers
#		Negative	-	-	-
#		Zero	6783	6783	-
#		Positive	14713	14713	-
#					
#		Total	21496	21496	-
#	#	Missing	5680		
<hr/>			<hr/>		
0			27176		
(2 unique values)					

p1142 is labeled and all values are documented in the label.

p1143: su hogar tiene : tv. cable			Number of Observations		
			Total	Integers	Nonintegers
#	Negative		-	-	-
#	Zero		17425	17425	-
#	Positive		4071	4071	-
#	Total		21496	21496	-
# #	Missing		5680		
0	1		27176		

(2 unique values)

p1143 is labeled and all values are documented in the label.

p1144: su hogar tiene : internet			Number of Observations		
			Total	Integers	Nonintegers
#	Negative		-	-	-
#	Zero		19702	19702	-
#	Positive		1794	1794	-
#	Total		21496	21496	-
# .	Missing		5680		
0	1		27176		

(2 unique values)

p1144 is labeled and all values are documented in the label.

```
. *Creamos un diccionario de variables
. codebook p1141 p1142 p1143 p1144 // diccionario de algunas variables
```

p1141			su hogar tiene : teléfono (fijo)		
type:	numeric (byte)				
label:	p1141				
range:	[0,1]		units:	1	
unique values:	2		missing .:	5680/27176	
tabulation:	Freq.	Numeric	Label		
	17059	0	pase		
	4437	1	telefono		
	5680	.			

p1142			su hogar tiene : celular		
type:	numeric (byte)				
label:	p1142				
range:	[0,1]		units:	1	
unique values:	2		missing .:	5680/27176	
tabulation:	Freq.	Numeric	Label		
	6783	0	pase		
	14713	1	celular		
	5680	.			

---

p1143 su hogar tiene : tv. cable

---

```

      type: numeric (byte)
      label: p1143
      range: [0,1]
unique values: 2
      units: 1
      missing .: 5680/27176
      tabulation: Freq.   Numeric   Label
                  17425      0   pase
                  4071      1   tv. cable
                  5680      .

```

---

p1144 su hogar tiene : internet

---

```

      type: numeric (byte)
      label: p1144
      range: [0,1]
unique values: 2
      units: 1
      missing .: 5680/27176
      tabulation: Freq.   Numeric   Label
                  19702      0   pase
                  1794      1   internet
                  5680      .

```

```

. *Realizamos una lista de valores de algunas variables
. list p1141 p1142 p1143 p1144 in 10/20 // listado de valores entre la observación 10 y 20

```

	p1141	p1142	p1143	p1144
10.	pase	celular	pase	pase
11.	telefono	celular	pase	pase
12.	pase	celular	pase	pase
13.	telefono	celular	tv. cabl	internet
14.	telefono	celular	tv. cabl	internet
15.	telefono	celular	pase	pase
16.	pase	celular	pase	pase
17.	pase	celular	tv. cabl	pase
18.	pase	celular	tv. cabl	pase
19.	telefono	celular	pase	pase
20.	pase	celular	tv. cabl	pase

```

. *Realizamos un cuadro estadístico resumen de algunas variables
. summarize p1141 p1142 p1143 p1144 // resumen estadístico de algunas variabl
> es

```

Variable	Obs	Mean	Std. Dev.	Min	Max
p1141	21496	.2064105	.404738	0	1
p1142	21496	.6844529	.4647442	0	1
p1143	21496	.1893841	.3918225	0	1
p1144	21496	.0834574	.2765788	0	1



```
. *Realizamos un cuadro resumen detallado de algunas variables
. sum p1141 p1142 p1143 p1144,detail // resumen estadístico detallado de algun
> as variables
```

su hogar tiene : teléfono (fijo)					
	Percentiles	Smallest			
1%	0	0			
5%	0	0			
10%	0	0	Obs		21496
25%	0	0	Sum of Wgt.		21496
50%	0		Mean		.2064105
		Largest	Std. Dev.		.404738
75%	0	1			
90%	1	1	Variance		.1638128
95%	1	1	Skewness		1.450797
99%	1	1	Kurtosis		3.104812
su hogar tiene : celular					
	Percentiles	Smallest			
1%	0	0			
5%	0	0			
10%	0	0	Obs		21496
25%	0	0	Sum of Wgt.		21496
50%	1		Mean		.6844529
		Largest	Std. Dev.		.4647442
75%	1	1			
90%	1	1	Variance		.2159872
95%	1	1	Skewness		-.793801
99%	1	1	Kurtosis		1.63012
su hogar tiene : tv. cable					
	Percentiles	Smallest			
1%	0	0			
5%	0	0			
10%	0	0	Obs		21496
25%	0	0	Sum of Wgt.		21496
50%	0		Mean		.1893841
		Largest	Std. Dev.		.3918225
75%	0	1			
90%	1	1	Variance		.1535249
95%	1	1	Skewness		1.58553
99%	1	1	Kurtosis		3.513905
su hogar tiene : internet					
	Percentiles	Smallest			
1%	0	0			
5%	0	0			
10%	0	0	Obs		21496
25%	0	0	Sum of Wgt.		21496
50%	0		Mean		.0834574
		Largest	Std. Dev.		.2765788
75%	0	1			
90%	0	1	Variance		.0764958
95%	1	1	Skewness		3.012179
99%	1	1	Kurtosis		10.07322

## 3.10. Generando y Transformando Variables

Para la creación de nuevas variables, STATA cuenta con el comando **generate** y **egen**, la diferencia entre ambos se explica a continuación:

- El comando *generate* nos permite generar variables haciendo uso de expresiones matemáticas, lógicas, numéricas. Si bien es cierto que STATA solamente reconoce los comandos con letras minúsculas, también es importante decir que hace una diferenciación de los nombres de las variables entre si son minúsculas o mayúsculas. Por ejemplo, generar una variable llamada *HoGaR* es diferente a que si lo denominamos *hogar* u *HOGAR*.
- El comando *egen* es una extensión del anterior, que permite utilizar expresiones que incluyan funciones más complejas del STATA, como es el caso de: medias, máximos, mínimos, desviación estándar, promedios móviles, variables estandarizadas, etc.

```
. *Generación de variables

. //Creación manual de variables

. generate alqp1= i105b^2 //gasto de alquiler pagado al cuadrado
(25304 missing values generated)

. generate alqp2=i105b/2    //la mitad del gasto de alquiler pagado
(25304 missing values generated)

. generate alqp3=i105b*2    //el doble del gasto de alquiler pagado
(25304 missing values generated)

. generate id=_n    // variable cuyo valor va de 1 hasta la última
.                  // observacion en saltos de 1 unidad

. generate N=_N    // variable cuyo valor indica el número total de
.                  // observaciones en la muestra

. //Creación de variables con "generate" usando funciones

. gen ln_alq=ln(i105b)    //logaritmo natural del gasto en alquiler pagado
(25304 missing values generated)

. gen sq_alq=sqrt(i105b) //raíz cuadrada del gasto en alquiler pagado
(25304 missing values generated)

. gen exp_alq=exp(i105b) //exponencial del gasto en alquiler pagado
(26825 missing values generated)
```

```
. //Generando variables dicótomicas

. gen luz =(p1121==1)      // 1 si el hogar posee luz; 0 si no posee

. gen agua =(p111==1 | p111==2 | p111==3) // 1 si el hogar posee agua; 0 si no posee

. sum luz agua
```

Variable	Obs	Mean	Std. Dev.	Min	Max
luz	27176	.6522667	.4762596	0	1
agua	27176	.5457021	.4979161	0	1

```
. browse p1121 luz p111 agua

. //Creación de variables con "egen" usando funciones

. egen medan_alq=mean(i105b)      //promedio del gasto en alquiler pagado

. egen median_alq=median(i105b)   //mediana del gasto en alquiler pagado

. egen moda_alq=mode(i105b)       //moda del gasto en alquiler pagado

. egen min_alq=min(i105b) //valor mínimo gasto en alquiler pagado

. egen max_alq=max(i105b) //valor máximo del gasto en alquiler pagado
```

En caso que se quiera cambiar los valores de una variable una vez creadas o de las ya existentes, podemos hacer uso de los comandos **replace** y **recode**.

- El comando *replace* permite reemplazar o modificar una variable o sus respectivos valores. Si se trata de reemplazar algunos valores de una variable, generalmente tendrá que cumplir algunas condiciones, por lo que se debe usar la condicional **if** . Este comando se digita después del comando *generate*.
- El comando *recode* permite modificar valores específicos de una variable.

```
. *Transformación de variables

. //Usando el comando "replace" para cambios en la variable

. gen telefono=1 if p1141==1      // 1 si el hogar posee teléfono fijo; 0 si no posee
(22739 missing values generated)

. replace telefono=0 if telefono==.
(22739 real changes made)

. gen movil=0 if p1142==1 // 1 si el hogar posee celular fijo; 0 si no posee
(12463 missing values generated)
```

```

. replace movil=1 if p1142==0
(6783 real changes made)

. gen cable =1 if p1143==1          // 1 si el hogar posee servicio de cable; 0 s
> i no posee
(23105 missing values generated)

. replace cable=0 if p1143!=1
(23105 real changes made)

. gen internet=0 if p1144!=1       // 1 si el hogar posee servicio de internet; 0 si no posee
(1794 missing values generated)

. replace internet=1 if p1144==1
(1794 real changes made)

. sum telefono movil cable internet

```

Variable	Obs	Mean	Std. Dev.	Min	Max
telefono	27176	.1632691	.3696178	0	1
movil	21496	.3155471	.4647442	0	1
cable	27176	.1498013	.3568831	0	1
internet	27176	.0660141	.2483114	0	1

```

. br p1141 telefono p1142 movil p1143 cable p1144 internet

. //Usando el comando "recode" para recodificar valores específicos de las variables

. recode p105a (2/4=2) (5/7=0),gen(viv_alq) //1 si es alquila; 2 si es propia y 0 otro
(4629 differences between p105a and viv_alq)

. recode p101 (5=0) (6=0) (7=0) (8=0),gen(tipo_viv) //0 si es otro tipo de vivienda
(970 differences between p101 and tipo_viv)

. sum tipo_viv viv_alq

```

Variable	Obs	Mean	Std. Dev.	Min	Max
tipo_viv	21064	1.162885	.7527236	0	4
viv_alq	21496	1.597693	.7496249	0	2

### 3.11. Nombrando y Etiquetando Variables

Si se desea cambiar de nombre a una variable se hace uso del comando **rename**.

**\*Nombrando y Etiquetando Variables**

```

. //De las últimas variables creadas cambiamos de nombre a la variable id y N
rename id ident_obs
ren N ident_total

```

Si deseamos darle el significado a la variable, podemos etiquetarlo con el comando **label variable**.

```
. //De las últimas variables renombradas lo etiquetamos de la
. // siguiente forma:
label variable ident_obs "Identificador de Observaciones"
la var ident_total "Identificador Total"
```

En el caso que tengamos variables categóricas, es útil explicar el significado de cada uno de los valores discretos, para este proceso usamos los comandos **label define** y **label value**.

```
. //De la variable categórica que creamos "tipo_viv" podemos etiquetar sus valores de la siguiente forma

. //Primero definimos una etiqueta llamada "tipo_vivienda" y luego etiquetamos los valores

. label define vivienda_alquilada 0 "Otro" 1 "Alquilada" 2 "Propia"
. label value viv_alq vivienda_alquilada
. label list vivienda_alquilada

vivienda_alquilada:
    0 Otro
    1 Alquilada
    2 Propia

. br p105a viv_alq

. label list p101

p101:
    1 casa independiente
    2 departamento en edificio
    3 vivienda en quinta
    4 vivienda en casa de vecindad (callejón, solar o corralón)
    5 choza o cabaña
    6 vivienda improvisada
    7 local no destinado para habitación humana
    8 otro

. label value tipo_viv p101
. label define p101 0 "Otro" ,add
. label list p101

p101:
    0 Otro
    1 casa independiente
    2 departamento en edificio
    3 vivienda en quinta
    4 vivienda en casa de vecindad (callejón, solar o corralón)
    5 choza o cabaña
    6 vivienda improvisada
    7 local no destinado para habitación humana
    8 otro

. label value p101 tipo_vivienda

. br p101 tipo_viv
```

## 3.12. Tipo y Formato de Variables

### 3.12.1. Tipo de Variables

En STATA existen dos tipos de formatos:

- *Tipo Numérico*: Se puede encontrar la siguiente clasificación <sup>6</sup>:

Tipo	Byte	Mínimo	Máximo
<i>byte</i>	1	-127	10
<i>int</i>	2	-32,767	32,740
<i>long</i>	4	-2,147,483,647	2,147,483,620
<i>float</i>	4	$-1,70141173319 * 10^{38}$	$1,70141173319 * 10^{38}$
<i>double</i>	8	$-8,9884656743 * 10^{307}$	$8,9884656743 * 10^{307}$

Tabla 3.1: Tipo de Variable Numérico

- *Tipo No Numérico*: Este tipo es reconocido como *cadena de texto* o *string*. Generalmente se encierran entre comillas y presenta la siguiente clasificación:

Tipo	Byte	Descripción
<i>str1</i>	1	Hasta 1 carácter
<i>str2</i>	2	Hasta 2 carácter
⋮	⋮	⋮
<i>str20</i>	20	Hasta 20 carácter

Tabla 3.2: Tipo de Variable No Numérico

---

<sup>6</sup>Cuando se genera una variable con datos numéricos, STATA por default le asigna un formato *float*.

### 3.12.2. Formato de Variables

La forma cómo podemos especificar el formato de las variables es de la siguiente manera:

- *Formato Numérico:*

Esquema	Símbolo	Descripción
Primero	%	indica el comienzo del formato
luego ( <i>opcional</i> )	-	si se quiere alinear el resultado a la izquierda
luego ( <i>opcional</i> )	0	si se quiere conservar los <i>ceros principales</i>
luego	#	cifra que indique el tamaño del resultado
luego	.	se coloca un punto
luego	#	número de dígitos después del punto decimal
luego ( <i>cualquiera</i> )	<i>e</i>	para notación científica. ejm: $10e + 04$
	<i>f</i>	para formato fijo. ejm: 5000.0
	<i>g</i>	para formato general (STATA muestra acorde al número elegido)
luego ( <i>opcional</i> )	<i>c</i>	para el formato de <i>coma</i> (no se permite para notación científica)

Tabla 3.3: Formato de Variable Numérico

- *Formato de Series de Tiempo (Fechas)*

Esquema	Símbolo	Descripción
Primero	%	indica el comienzo del formato
luego ( <i>opcional</i> )	-	si se quiere alinear el resultado a la izquierda
luego	<b>t</b>	se coloca <i>t</i> para indicar formato fecha
luego ( <i>cualquiera</i> )	<b>h</b>	para horas. <i>ex: 1972h2</i>
	<b>d</b>	para días. <i>ex: 05jul1972</i>
	<b>w</b>	para semanas. <i>ex: 1972w27</i>
	<b>m</b>	para meses. <i>ex: 1972m7</i>
	<b>q</b>	para trimestres. <i>ex: 1972q3</i>
	<b>y</b>	para años. <i>ex: 1972</i>

Tabla 3.4: Formato de Variable con Fechas

```
. *Formatos Numéricos
```

```
. describe ln_alq
```

variable name	storage type	display format	value label	variable label
ln_alq	float	%9.0g		

```
. list ln_alq if ln_alq!=. in 1/100 //lista las primeras 20 observaciones
```

	ln_alq
26.	7.78239
30.	7.78239
69.	6.73578
80.	7.4313
86.	7.785305

```
. format%5.2f ln_alq //nueve dígitos y dos decimales
```

```
. describe ln_alq
```

variable name	storage type	display format	value label	variable label
ln_alq	float	%5.2f		

```
. list ln_alq if ln_alq!=. in 1/100 //lista las primeras 20 observaciones
```

	ln_alq
26.	7.78
30.	7.78
69.	6.74
80.	7.43
86.	7.79

```
. format%-5.2f ln_alq //doce dígitos y un decimal alineado a la izquierda
```

```
. describe ln_alq
```

variable name	storage type	display format	value label	variable label
ln_alq	float	%-5.2f		

```
. list ln_alq if ln_alq!=. in 1/100 //lista las primeras 20 observaciones
```

	ln_alq
26.	7.78
30.	7.78
69.	6.74
80.	7.43
86.	7.79



## 3.13. Conversión de Variables

En STATA es posible generar una variable *numérica* a partir de una variable *string* y viceversa.

### 3.13.1. De una Variable String Numérica a una Variable Numérica

Para poder realizar esta conversión se recurre a la función **real()** después del comando **generate**. También es posible realizar la misma operación con el comando **destring** donde la variable generada se coloca como opción en la misma línea de comando.

```
. *Conversión de Variables

. //De una Variable String Numérica a una Variable Numérica
. gen year= real(año)

. br año year

. destring mes, gen(month)
mes has all characters numeric; month generated as byte

. br mes month
```

### 3.13.2. De una Variable Numérica a una Variable String

A través del comando **tostring** podemos convertir una variable numérica a string. Aquí también la variable generada se coloca en la misma línea de comando como una opción.

```
. //De una Variable Numérica a una Variable String
. tostring result, gen(resultado)
resultado generated as str1

. br result*

. decode result, gen(resultado1)

. br result*
```

### 3.13.3. De una Variable String No-Numérica a una Variable Numérica

Para poder realizar esta conversión se recurre al comando **encode**. Este comando codifica una variable string a una numérica. Aquí también la variable generada se coloca en la misma línea de comando como una opción. Después de ejecutar esta operación es recomendable utilizar el comando **label list** para observar las etiquetas que fueron asignadas a los valores de la nueva variable.

```
. //De una Variable String No-Numérica a una Variable Numérica

. gen encuesta="aceptado" if result==1 | result==2
(5680 missing values generated)

. replace encuesta="rechazado" if result>=3
encuesta was str8 now str9
(5680 real changes made)

. br result encuesta

. encode encuesta, gen(encuesta1) label(encuesta)

. label list encuesta
encuesta:
      1 aceptado
      2 rechazado

. br result encuesta*
```

## 3.14. Selección de Muestra y Variables

Existen ocasiones que no deseamos trabajar con todas las variables u observaciones de la base de datos, por lo tanto, STATA cuenta con los comandos **drop** y **keep** para la selección particular de las mismas con la finalidad de obtener más memoria para trabajar.

- El comando *keep* permite mantener observaciones o variables en la memoria del STATA.
- El comando *drop* permite eliminar observaciones o variables de la memoria del STATA.

```

. * Selección de Muestra y Variables

. //Guardamos la base modifciada con el nombre "base_modif.dta"

. save base_modif.dta,replace
file base_modif.dta saved

. //Usando el comando "keep" para guardar algunas variables
. keep año mes encuesta1 telefono movil cable internet luz agua

. browse

. //Usando el comando "keep" para seleccionar aquellas observaciones que tienen agua y luz
. keep if agua==1
(12346 observations deleted)

. sum agua

```

Variable	Obs	Mean	Std. Dev.	Min	Max
agua	14830	1	0	1	1

```

. browse agua

. //Si de esta nueva muestra seleccionamos las primeras 1500 observaciones

. count //contamos el número de observaciones
14830

. keep in 1/1500
(13330 observations deleted)

. count //contamos el nuevo número de observaciones
1500

. //Usando el comando "drop" para eliminar algunas variables

. //Volvemos a cargar la base modificada.

. use base_modif.dta,clear

. //Eliminación las variables
. drop encuesta resultado1

. browse

. //Eliminamos una muestra
. drop if agua!=1
(12346 observations deleted)

. sum agua

```

Variable	Obs	Mean	Std. Dev.	Min	Max
agua	14830	1	0	1	1

```

. browse agua

. //Eliminamos las últimas 1500 observaciones

```

```
. count //contamos el número de observaciones
14830

. drop in -1500/1
(1500 observations deleted)

. count //contamos el nuevo número de observaciones
13330
```

## 3.15. Manipulación de Base de Datos

La utilidad de manipular base de datos incluye reordenar las observaciones y/o variables, realizarle cambios temporales y guardarlos para luego acceder a otra base de datos y así combinarlos, obteniendo una nueva base fusionada.

### 3.15.1. Ordenar Observaciones y Variables

El comando **sort** ordenar observaciones de manera ascendente acorde a la(s) variable(s) señalada(s). En cambio el comando **gsort** nos permite ordenarlo de manera ascendente como descendente.

```
. *Ordenar observaciones

. use base_modif.dta,clear

. //Podemos ordenar de forma ascendente la variable mes
. sort mes

. br mes

. //O en forma descendente
. gsort -mes

. br mes

. //También podemos ordenar de forma ascendente variables consecutivamente

. sort año mes conglome vivienda hogar

. br año mes conglome vivienda hogar
```

También se puede ordenar las variables usando el comando **order**. Este puede ser útil, si por ejemplo uno desea distribuir las variables de una base de datos a

otras bases.

```
. *Ordenar variables

. //Podemos order en el siguiente orden las variables
. order año mes conglome ubigeo vivienda hogar

. //También podemos ordenar las variable de forma alfabética
. order _all, alphabetic
```

Estos comandos son importantes al usar el prefijo **by()**, que nos permite realizar algunas operaciones por grupo de observaciones.

```
. *Usando el prefijo by()
.
. //Calculamos un cuadro resumen estadísitico del monto de alquiler anual por tipo de vivienda

. sort tipo_viv

. by tipo_viv : sum i105b if p105a==1
```

-> tipo\_viv = Otro

Variable	Obs	Mean	Std. Dev.	Min	Max
i105b	1	120	.	120	120

-> tipo\_viv = casa independiente

Variable	Obs	Mean	Std. Dev.	Min	Max
i105b	1102	2301.61	2411.309	60	30460

-> tipo\_viv = departamento en edificio

Variable	Obs	Mean	Std. Dev.	Min	Max
i105b	213	4623.033	3517.066	603	21524

-> tipo\_viv = vivienda en quinta

Variable	Obs	Mean	Std. Dev.	Min	Max
i105b	58	2682.603	2031.373	241	15004

-> tipo\_viv = vivienda en casa de vecindad (callejón, solar o corralón)

Variable	Obs	Mean	Std. Dev.	Min	Max
i105b	328	1190.805	761.9599	120	4595

-> tipo\_viv = .

Variable	Obs	Mean	Std. Dev.	Min	Max
i105b	36	1358.028	723.0779	294	3648

```
. //También podemos escribir del siguiente modo:
. bysort tipo_viv : sum i105b if p105a==1
```

```
-> tipo_viv = Otro
```

Variable	Obs	Mean	Std. Dev.	Min	Max
i105b	1	120	.	120	120

```
-> tipo_viv = casa independiente
```

Variable	Obs	Mean	Std. Dev.	Min	Max
i105b	1102	2301.61	2411.309	60	30460

```
-> tipo_viv = departamento en edificio
```

Variable	Obs	Mean	Std. Dev.	Min	Max
i105b	213	4623.033	3517.066	603	21524

```
-> tipo_viv = vivienda en quinta
```

Variable	Obs	Mean	Std. Dev.	Min	Max
i105b	58	2682.603	2031.373	241	15004

```
-> tipo_viv = vivienda en casa de vecindad (callejón, solar o corralón)
```

Variable	Obs	Mean	Std. Dev.	Min	Max
i105b	328	1190.805	761.9599	120	4595

```
-> tipo_viv = .
```

Variable	Obs	Mean	Std. Dev.	Min	Max
i105b	36	1358.028	723.0779	294	3648

### 3.16. Preservar y Restaurar Base de Datos

En algunos casos, es necesario realizar cambios temporales a una base de datos, desarrollar algunos cálculos y entonces retornar a la base original. El comando **preserve** nos permite retener la base de datos y el comando **restore** nos permite regresar a la base de datos original. El comando *restore* se usa inmediatamente después del comando *preserve*.

```

. *Preservar y Restaurar base de datos

. //Si calculamos temporalmente el alquiler mensual

. sum i105b

```

Variable	Obs	Mean	Std. Dev.	Min	Max
i105b	1872	2752.722	3835.5	60	77541

```

. preserve

. replace i105b=i105b/12
i105b was long now double
(1872 real changes made)

. sum i105b

```

Variable	Obs	Mean	Std. Dev.	Min	Max
i105b	1872	229.3935	319.625	5	6461.75

```

. restore

. sum i105b

```

Variable	Obs	Mean	Std. Dev.	Min	Max
i105b	1872	2752.722	3835.5	60	77541

## 3.17. Tablas y Tabulaciones

En esta sección veremos diversas formas de presentar tablas de estadísticas descriptivas, entre estas tenemos:

### 3.17.1. Tabulate

El comando **tabulate** muestra una tabla la cual señala una lista de los distintos valores que tiene una variable con su frecuencia absoluta, porcentual y acumulada. Es recomendable usar este comando para aquellas variables con pocos valores diversos. También es útil para crear variables dummy con ayuda del comando *generate* y además tabular por tipo de individuo con el prefijo *by*. El comando *tabulate* puede mostrarnos tablas tanto de un solo sentido como de doble sentido

```
. *Tabulaciones y Tablas
```

```
. // TABULATE
```

```
. //Tabulación de un solo sentido
```

```
. tabulate tipo_viv
```

RECODE of p101 (tipo de vivienda)	Freq.	Percent	Cum.
Otro	970	4.61	4.61
casa independiente	18,133	86.09	90.69
departamento en edificio	642	3.05	93.74
vivienda en quinta	198	0.94	94.68
vivienda en casa de vecindad (callejón,	1,121	5.32	100.00
Total	21,064	100.00	

```
. tab tipo_viv if i105b>300
```

RECODE of p101 (tipo de vivienda)	Freq.	Percent	Cum.
Otro	969	4.62	4.62
casa independiente	18,053	86.12	90.74
departamento en edificio	642	3.06	93.80
vivienda en quinta	197	0.94	94.74
vivienda en casa de vecindad (callejón,	1,102	5.26	100.00
Total	20,963	100.00	

```
. //generamos variablesdummy con tabulate
```

```
. tabulate tipo_viv, gen(dum_viv)
```

RECODE of p101 (tipo de vivienda)	Freq.	Percent	Cum.
Otro	970	4.61	4.61
casa independiente	18,133	86.09	90.69
departamento en edificio	642	3.05	93.74
vivienda en quinta	198	0.94	94.68
vivienda en casa de vecindad (callejón,	1,121	5.32	100.00
Total	21,064	100.00	

```
. br tipo_viv dum_viv*
```

```
. //Tabulación por grupos
```

```
. by tipo_viv: tab agua
```

```
-> tipo_viv = Otro
```

agua	Freq.	Percent	Cum.
0	747	77.01	77.01
1	223	22.99	100.00
Total	970	100.00	



-> tipo\_viv = casa independiente

agua	Freq.	Percent	Cum.
0	5,782	31.89	31.89
1	12,351	68.11	100.00
Total	18,133	100.00	

-> tipo\_viv = departamento en edificio

agua	Freq.	Percent	Cum.
1	642	100.00	100.00
Total	642	100.00	

-> tipo\_viv = vivienda en quinta

agua	Freq.	Percent	Cum.
0	1	0.51	0.51
1	197	99.49	100.00
Total	198	100.00	

-> tipo\_viv = vivienda en casa de vecindad (callejón, solar o corralón)

agua	Freq.	Percent	Cum.
0	67	5.98	5.98
1	1,054	94.02	100.00
Total	1,121	100.00	

-> tipo\_viv = .

agua	Freq.	Percent	Cum.
0	5,749	94.06	94.06
1	363	5.94	100.00
Total	6,112	100.00	

. //Tabulación de doble sentido

. tab agua luz

agua	luz		Total
	0	1	
0	8,115	4,231	12,346
1	1,335	13,495	14,830
Total	9,450	17,726	27,176

```
. tab agua luz, row col
```

Key
<i>frequency</i>
<i>row percentage</i>
<i>column percentage</i>

agua	luz		Total
	0	1	
0	8,115	4,231	12,346
	65.73	34.27	100.00
	85.87	23.87	45.43
1	1,335	13,495	14,830
	9.00	91.00	100.00
	14.13	76.13	54.57
Total	9,450	17,726	27,176
	34.77	65.23	100.00
	100.00	100.00	100.00

```
. tab agua luz, row col nofreq
```

Key
<i>row percentage</i>
<i>column percentage</i>

agua	luz		Total
	0	1	
0	65.73	34.27	100.00
	85.87	23.87	45.43
1	9.00	91.00	100.00
	14.13	76.13	54.57
Total	34.77	65.23	100.00
	100.00	100.00	100.00

### 3.17.2. Table

El comando **table** nos permite crear tablas de doble y triple sentido, mostrando las frecuencias absolutas o porcentuales visto de forma horizontal o vertical.

```
. //TABLE
```

```
. //Tabla de doble entrada
```

```
. table agua luz
```

agua	luz	
	0	1
0	8,115	4,231
1	1,335	13,495

```
. table agua luz, row col
```

agua	luz		Total
	0	1	
0	8,115	4,231	12,346
1	1,335	13,495	14,830
Total	9,450	17,726	27,176

```
. table agua luz, row col scol
```

agua	luz		Total
	0	1	
0	8,115	4,231	12,346
1	1,335	13,495	14,830
Total	9,450	17,726	27,176

```
. //Tabla de tres entrada
```

```
. table agua luz tipo_viv
```

agua	RECODE of p101 (tipo de vivienda) and luz									
	Otro		casa indep		departamen		vivienda e		vivienda e	
	0	1	0	1	0	1	0	1	0	1
0	622	125	1,793	3,989				1	9	58
1	153	70	1,172	11,179		642	1	196	8	1,046

```
. table agua luz tipo_viv,scol
```

agua	RECODE of p101 (tipo de vivienda) and luz									
	Otro		casa indep		departamen		vivienda e		vivienda e	
	0	1	0	1	0	1	0	1	0	1
0	622	125	1,793	3,989				1	9	58
1	153	70	1,172	11,179		642	1	196	8	1,046

	RECODE of p101 (tipo de vivienda) and luz	
	—— Total ——	
agua	0	1
0	2,424	4,173
1	1,334	13,133

### 3.17.3. Tabstat

El comando **tabstat** provee un resumen estadísticos que permite más flexibilidad que el *summarize*.

```
. //TABSTAT

. //Tabla con estadísticos descriptivas

. tabstat i105b agua luz internet telefono,by(tipo_viv) ///
. stat(mean median min max sum sd va cv sk k)
```

Summary statistics: mean, p50, min, max, sum, sd, variance, cv, skewness, kurtosis  
by categories of: tipo\_viv (RECODE of p101 (tipo de vivienda))

tipo_viv	i105b	agua	luz	internet	telefono
Otro	12150.5	.2298969	.2010309	.0010309	.0103093
	12150.5	0	0	0	0
	120	0	0	0	0
	24181	1	1	1	1
	24301	223	195	1	10
	17013.7	.4209834	.4009779	.0321081	.101062
	2.89e+08	.177227	.1607832	.0010309	.0102135
	1.400247	1.831183	1.994608	31.14482	9.803013
	0	1.283863	1.491969	31.09664	9.695897
	1	2.648303	3.225972	968.001	95.01042
casa independien	2659.605	.6811338	.836486	.0779794	.208184
	1801	1	1	0	0
	60	0	0	0	0
	60527	1	1	1	1
	3167590	12351	15168	1414	3775
	3563.231	.4660499	.3698442	.2681465	.4060203
	1.27e+07	.2172025	.1367847	.0719026	.1648525
	1.339759	.6842266	.4421404	3.438685	1.950296
	6.409775	-.7773367	-1.819659	3.147774	1.437485
	75.69523	1.604252	4.31116	10.90848	3.066363

departamento en	5324.809	1	1	.3909657	.5436137
	3642	1	1	0	1
	603	1	1	0	0
	77541	1	1	1	1
	1363151	642	642	251	349
	5918.226	0	0	.4883472	.4984826
	3.50e+07	0	0	.238483	.2484849
	1.111444	0	0	1.249079	.9169794
	7.57581	.	.	.4468922	-.1751223
	88.43645	.	.	1.199713	1.030668
vivienda en quin	2682.603	.9949495	.9949495	.2373737	.4949495
	2187	1	1	0	0
	241	0	0	0	0
	15004	1	1	1	1
	155591	197	197	47	98
	2031.373	.0710669	.0710669	.4265517	.5012419
	4126476	.0050505	.0050505	.1819464	.2512434
	.7572393	.0714277	.0714277	1.796963	1.012713
	3.955407	-13.96442	-13.96442	1.234514	.0202031
	24.5146	196.0051	196.0051	2.524024	1.000408
vivienda en casa	1196.274	.9402319	.984835	.0481713	.1364853
	966	1	1	0	0
	120	0	0	0	0
	4595	1	1	1	1
	393574	1054	1104	54	153
	767.2367	.2371624	.1222636	.2142236	.3434564
	588652.1	.056246	.0149484	.0458917	.1179623
	.6413555	.2522382	.1241463	4.447123	2.516435
	1.579983	-3.71415	-7.934518	4.220174	2.117748
	6.370189	14.79491	63.95658	18.80987	5.484855
Total	2780.069	.6868116	.8215913	.0838872	.2081751
	1798	1	1	0	0
	60	0	0	0	0
	77541	1	1	1	1
	5104207	14467	17306	1767	4385
	3866.625	.4638013	.3828655	.2772252	.4060124
	1.50e+07	.2151116	.146586	.0768538	.164846
	1.390838	.6752962	.4660048	3.304737	1.950341
	7.742873	-.8055866	-1.679958	3.002056	1.437551
	113.5848	1.64897	3.822259	10.01234	3.066554

## 3.18. Formas de Base de Datos

### 3.18.1. Formas Long y Wide

El comando **reshape** nos permite transformar una base de datos de forma larga (*long*) a una de forma ancha (*wide*) y viceversa. Como se muestra a continuación:

<i>Forma Long</i>					<i>Forma Wide</i>				
	id	year	sex	salary		id	salary2009	salary2010	sex
1	1	2009	0	550	1	1	550	800	0
2	1	2010	0	800	2	2	1200	1000	1
3	2	2009	1	1100	3	3	900	1400	1
4	2	2010	1	1000					
5	3	2009	1	900					
6	3	2010	1	1400					

Figura 3.6: Fomas de Base de Datos Long y Wide

En general para efectos de estimación es necesario que la base de datos este en formato long, esta distinción es importante para análisis de panel data.

```
. *Formas de Base de Datos
. //Reshape
. clear all
. input codigo año genero ingreso
      codigo    año    genero    ingreso
1. 1      2009      0      500
2. 1      2010      0      300
3. 1      2011      0      400
4. 2      2009      1      600
5. 2      2010      1      900
6. 2      2011      1      450
7. 3      2009      0      500
8. 3      2010      0      300
9. 3      2011      0      400
10. end
. browse
```

```
. save base_long, replace
file base_long.dta saved

. *long -> wide
. reshape wide ingreso,i(codigo) j(año)
(note: j = 2009 2010 2011)
```

Data	long	->	wide
Number of obs.	9	->	3
Number of variables	4	->	5
j variable (3 values)	año	->	(dropped)
xij variables:	ingreso	->	ingreso2009 ingreso2010 ingreso2011

```
> 011

. browse

. save base_wide, replace
file base_wide.dta saved

. *wide -> long
. reshape long ingreso,i(codigo) j(año)
(note: j = 2009 2010 2011)
```

Data	wide	->	long
Number of obs.	3	->	9
Number of variables	5	->	4
j variable (3 values)		->	año
xij variables:	ingreso2009 ingreso2010 ingreso2011	->	ingreso

```
. browse
```

## 3.19. Colapsar Base de Datos

Hay ocasiones en que la base de datos con forma *long* puede requerirse para colapsarlo tal que cada grupo de individuos este representada por una observación en particular, ya sea por el *promedio*, la *mediana*, *desviación estándar*, *máximo*, *mínimo*, la *suma*, *etc.*, de alguna variable en particular. Para hacer esta operación recurrimos al comando **collapse**.

```
. //Collapse

. *buscando cuanto ganó en total cada persona en el periodo

. preserve
```

```
. collapse (sum) ingreso,by(codigo)
. save collapse_saltot,replace
file collapse_saltot.dta saved
. list
```

	codigo	ingreso
1.	1	1200
2.	2	1950
3.	3	1200

```
. restore
```

```
. *buscando cuanto ganó en promedio cada persona en el periodo
```

```
. preserve
. collapse (mean) ingreso,by(codigo)
. save collapse_salprom,replace
file collapse_salprom.dta saved
. list
```

	codigo	ingreso
1.	1	400
2.	2	650
3.	3	400

```
. restore
```

```
. *buscando cuanto ganó en total y en promedio cada persona en el periodo a la vez
```

```
. preserve
. collapse (sum) sum_ingreso=ingreso (mean) mean_ingreso=ingreso,by(codigo)
. save collapse_otro,replace
file collapse_otro.dta saved
. list
```

	codigo	sum_in-o	mean_i-o
1.	1	1200	400
2.	2	1950	650
3.	3	1200	400

```
. restore
```

## 3.20. Fusión de Base de Datos

Es común la combinación de varias bases de datos. Se va a mostrar dos operaciones básicas: añadir variables y añadir observaciones. Los comandos asociados a estas operaciones son **merge**, **append** y **joinby**.



- El comando *merge* se utiliza para añadir variables, es decir, una o dos bases de manera horizontal. Los ficheros de datos deben de tener una variable de identificación y además deben de estar ordenados por dicha variable. Este comando requiere de dos bases de datos, uno se va a denominar *base master* al cual se le van añadir las variables y una *base using* la cual contiene las variables que se van a añadir a la base master. Al realizar el *merge* se crea una variable *\_merge* de manera automática que toma valores dependiendo si el registro de los datos está presente en una de las bases o en ambos. Cuando el valor de la variable *\_merge* es 1 quiere decir que el dato solo aparece en la *base master*, 2 cuando aparece en la *base using* y 3 cuando aparece en ambos.

```
. *Fusión de Base de Datos

. //MERGE

. *Base Master
. clear all
. input codigo año genero ingreso
      codigo    año    genero    ingreso
1. 1      2009      0      500
2. 1      2010      0      300
3. 1      2011      0      400
4. 2      2009      1      600
5. 2      2010      1      900
6. 2      2011      1      450
7. 3      2009      0      500
8. 3      2010      0      300
9. 3      2011      0      400
10. end

. sort codigo año
. save base_master, replace
file base_master.dta saved

. *Base Using
. clear all
. input codigo año exper casado
      codigo    año    exper    casado
1. 1      2009      18          0
2. 1      2010      19          1
3. 1      2011      19.75      1
4. 2      2009      10          1
5. 2      2010      11          0
6. 2      2011      12          0
7. 3      2009      5           1
8. 3      2010      5.5         1
9. 3      2011      6.5         1
10. end

. sort codigo año
. save base_using, replace
file base_using.dta saved

. *merge
```

```
. use base_master, clear
. browse
. merge codigo año using base_using
(note: you are using old merge syntax; see [R] merge for new syntax)
. browse

. tabulate _merge
      _merge |      Freq.   Percent   Cum.
-----+-----+-----+-----
          3 |          9   100.00   100.00
-----+-----+-----+-----
      Total |          9   100.00
. drop _merge

. save base_full, replace
file base_full.dta saved
```

- El comando *append* se utiliza para añadir observaciones, es decir, une a dos bases de manera vertical. Aquí también será necesario una *base master* y una *base using*, además la base originada tendrá una forma *long*.

```
. //APPEND

. *base using
. clear
. input codigo año experiencia casado genero ingreso
      codigo      año  experie-a      casado      genero      ingreso
1. 1      2008    17.8      0      0      400
2. 2      2008      9      1      1      300
3. 3      2008    4.5      0      0      540
4. 4      2008      3      0      1      800
5. end

. sort codigo año
. save base_append, replace
file base_append.dta saved

. *base master
. use base_full, clear
. sort codigo año
. browse
. append using base_append
. browse

. save base_full, replace
file base_full.dta saved
```

- El comando *joinby* forma todo los pares de combinaciones dentro de cada grupo a partir de dos bases de datos, una *master* y otra *using*. La base *master*

contiene variables para cada individuo clasificado por grupos y la base *using* contiene variables a nivel de grupos, entonces, el comando *joinby* colocará los valores de las variables según al grupo que pertenece cada individuo.

```
. //JOINBY

. *base using
. clear
. input año tc
      año      tc
1. 2009      3.01
2. 2010      2.89
3. 2011      2.65
4. end

. sort año
. save base_joinby, replace
file base_joinby.dta saved

. *base master
. use base_full, clear
. sort codigo año
. browse
. joinby using base_joinby, unmatched(both)
join on common variables: año
. browse

. tabulate _merge
```

_merge	Freq.	Percent	Cum.
only in master data	4	30.77	30.77
both in master and using data	9	69.23	100.00
Total	13	100.00	

```
. drop _merge

. save base_full, replace
file base_full.dta saved
```

## 3.21. Ejercicio Propuesto

Se muestra la base de datos concerniente a la demanda de electricidad por departamentos para los periodos 2010-2011, en el un archivo de Excel con el nombre de “demanda\_electricidad”.

Las variables a considerar en esta base son los siguientes:

- **Venta:** Es la demanda de electricidad medida en Mega Watt -hora (MW-h).
- **Facturación:** Es la venta de electricidad medido en Miles de dólares
- **Cliente:** Son los clientes de energía eléctrica beneficiados de este servicio.

A partir de este archivo se le pide lo siguiente:

1. Crear en el disco *D*: una carpeta de trabajo llamado Ejercicio1, luego guarde el archivo “demanda\_electricidad” en dicha carpeta.
2. En un archivo Do-File escriba una plantilla de inicio. Es decir, escriba como comentario sus datos personales (nombre y apellidos), limpie la base de datos, establezca una memoria de 50 megabytes, cambie la ruta de trabajo y cree una bitácora con el nombre de *Solución1* y formato de texto.
3. Se pide que a partir del *STATA TRANSFER* convierta el archivo del Excel al archivo en “.dta” con el nombre “Dda\_Elect”.
4. Importar la base de datos al STATA.
5. Hacer una descripción general de la base de datos siguiendo los criterios que se desarrollaron en la clase (*tratamiento de variables*).
6. Se desea crear una nueva variable llamada **precio**, la cual se origine de la división entre la variable *facturación* y *ventas*. Y establecer esta nueva variable en formato con solo dos decimales.
7. Se pide codificar la variable “departamento” de manera manual creando una variable llamada **dep1**, respetando un orden alfabético de los nombres de los departamentos. Además, desarrollar una etiqueta a los valores para esta variable con el nombre **label\_dep1**.
8. ¿Cómo hubiese sido si codificase la variable “departamento” de manera directa creando una nueva variable llamada **dep2** y una nueva etiqueta de valores llamada **label\_dep2**?
9. A continuación elimine la variable “dep1” y renombre la variable “dep1” por **dep**.

10. Genere nuevas variables que sean el logaritmo natural de la variable *venta*, *facturación*, *pbi*, *cliente* y *precio* que se llamen **ln\_vta**, **ln\_fact**, **ln\_pbi**, **ln\_cte** y **ln\_precio**. Y a continuación etiqueta las variables con las siguientes descripciones: *Logaritmo de Ventas*, *Logaritmo de Facturacion*, *Logaritmo del PBI*, *Logaritmo de Clientes* y *Logaritmo de Precio*.
11. Luego guarde la base de datos modificada con el nombre “Dda\_Elect\_modif.dta”.
12. Realizar un análisis descriptivo de las variables *ln\_vta*, *ln\_fact*, *ln\_pbi*, *ln\_cte* y *ln\_precio* por año y departamento.
13. Ahora se pide un cuadro de estadísticos (como la media, mediana, mínimo, máximo, desviación estándar, varianza, curtosis, asimetría) para las variables *ln\_vta*, *ln\_fact*, *ln\_pbi*, *ln\_cte* y *ln\_precio* por departamento y en forma global.

Ahora se requiere convertir la unidad de la variable *facturación* a miles de soles. Para ello, recurra a la página del BCRP y descargue la serie del Tipo de Cambio (TC) Bancario Nuevo Sol/Dólar-Venta (S/. por US\$) mensual, para el mismo periodo de análisis (2010-2011). Luego realice los siguientes procedimientos:

1. Transforme la abse de datos del Tipo de Cambio de la manera adecuada para ser importada al STATA. *Sugerencia: Genere una columna de variable numerica para los años y otra para los meses de forma independiente.*
2. Calcule el Tipo de Cambio Promedio trimestral usando el comando **collapse**. *Sugerencia: Genere una variable que permita identificar a que trimestre pertenece cada mes, por ejemplo: el mes 1,2 y 3 serían igual a 1 por ser el primer trimestre; los meses 4,5 y 6 serían 2 por ser el segundo trimestre y así sucesivamente. Luego, crear una variable identificadora con la variable año y trimestre.*
3. Fusione la base de datos de la demanda de electricidad y del tipo de cambio, usando dos métodos diferentes.
4. Cree una nueva variable de facturación en miles de soles.

# Capítulo 4

## Gráficos en STATA

### 4.1. Introducción a STATA GRAPH

STATA presenta una amplia variedad de gráficos, la cual abarca figuras como: matrices de ploteos, histogramas, áreas, líneas, caja y bigote, etc. Comenzaremos demostrando siete tipos de gráficos:

- **histogram** : Histogramas
- **graph twoway** : Scatterplot, líneas, y otros entre dos variables.
- **graph matrix** : Matrices de Scatterplots.
- **graph box** : Gráficas de caja y bigotes.
- **graph bar** : Gráficas de barras
- **graph dot** : Gráficas de puntos.
- **graph pie** : Gráficas de pastel o pie.

Para cada uno de estos gráficos existen muchas opciones <sup>1</sup>.

---

<sup>1</sup>Para observar otros tipos de gráficos y comandos relacionados a estos, se recomienda tipear en la ventana de comandos **help graph\_other**.

## 4.2. Tipos de Gráficos

Los comandos del STATA GRAPH empiezan con la palabra **graph** (aunque en algunos casos esto es opcional) seguido por la palabra que indica el tipo de gráfico. A continuación describiremos los diferentes tipos de gráficos que se puede elaborar en el ambiente del STATA.

### 4.2.1. Histograma

La función de densidad de una variable puede ser estimada usando un histograma a través del comando **histogram**. Para ilustrar este comando, utilizaremos la base de la Encuesta Permanente de Empleo (EPE) correspondiente al trimestre móvil *Diciembre-Enero-febrero del año 2010*, la cual contiene información referente a la situación de empleo que tiene un individuo en el mercado laboral.

La figura 4.1 muestra un histograma simple del ingreso total del individuo **ingtot** convertido en logaritmo, donde dicha variable lo denominamos **lningt**. Esto se genera de la siguiente forma:

```
. clear all

. set mem 200m

. set more off

. cd "D:\Econometria-Stata\graficos"

. use trim_dic08-ene-feb09.dta,clear

. *HISTOGRAMA
. *-----

. *generamos el logaritmo del ingreso total
gen lning=ln(ingtot)

. histogram lningtot, frequency title("Histograma del Ln. del Ingreso Total")
(bin=49, start=0, width=2)
```

La figura 4.1 presenta dos opciones: **frequency** (en vez de la función de densidad que aparece por defecto) el cual se muestra en el eje vertical; y el **title()** que aparece en la parte superior del gráfico. Este figura revela que mayor parte de la población presenta un ingreso total expresado en logaritmo no mayor a 10.



Figura 4.1: Histograma (1)

La figura 4.2 contiene una versión con mayores mejoras (basado en algunos experimentos para encontrar los valores correctos):

1. El eje  $x$  está etiquetada desde 0 hasta 10, con incrementos de 2.5 unidades.
2. El eje  $y$  está etiquetada desde 0 hasta 800, con incrementos de 250 unidades.
3. Los marcadores sobre el eje  $y$  desde 1 hasta 800, con incrementos de 125 unidades.
4. La primera barra del histograma comienza en 0.
5. El ancho de cada barra (o bin) es 0.25.

```
. histogram lningtot, frequency title("Histograma del Ln. del Ingreso Total") ///
xlabel(0(2.5)10) ylabel(0(250)800) ytick(0(125)800) start(0) width(.25)
(bin=49, start=0, width=2)
```



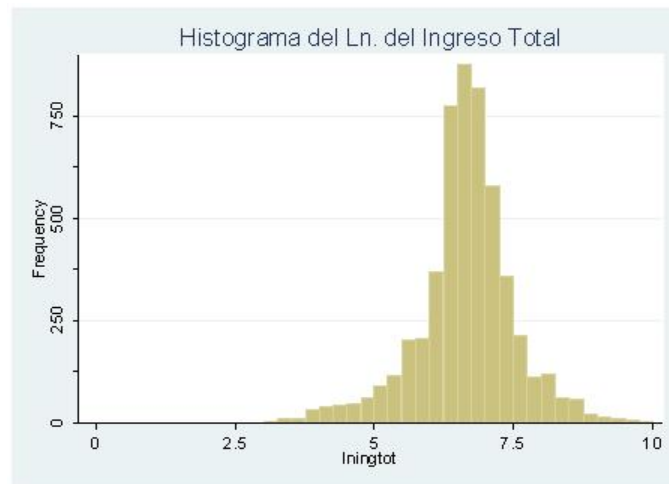


Figura 4.2: Histograma (2)

Otras útiles opciones son los siguientes:

- **bin** : Muestra un histograma con  $\#$  de bins (o barras). Podemos especificar `bin( $\#$ )` o `start( $\#$ )` con `width( $\#$ )`, pero no ambos.
- **percent** : Muestra los porcentajes en el eje vertical. Otras posibilidades son las opciones `fraction` que muestra la fracción de la data y `frequency` especificado en la Figura 4.1, el histograma por default muestra la densidad (`density`) lo que quiere decir que las barras están escaladas de tal forma que el área bajo la gráfica sume la unidad.
- **gap( $\#$ )** : Indica el espacio entre las barras, el número  $\#$  se especifica entre  $0 < \# < 100$ .
- **addlabel** : Etiqueta la parte superior de las barras del histograma con la frecuencia de datos.
- **discrete** : Especifica que la data es discreta, requiriendo una barra para cada valor de la variable.
- **norm** : Sobrepone una curva normal sobre el histograma, basado sobre la media muestral y desviación estándar.

- **kdensity** : Sobrepone un estimador de densidad de kernel sobre el histograma<sup>2</sup>.

El número de intervalos por default es  $\min(\sqrt{N}, 10\ln N/\ln 10)$ . Con los histogramas también podemos especificar nuestro propios títulos en el eje de las abscisas con **xtitle()** y en el eje de la ordenada con **ytitle()**.

En la figura 4.3 ilustra un ejemplo con algunas otras opciones de comando del *histogram*. Note el cambio de construcción de gráficos desde la figura 4.1 hasta más elaborada figura 4.3.

Este es un patrón normal para la construcción de gráficos en STATA: iniciamos por lo más simple, entonces experimentamos la suma de opciones para obtener una figura que se muestre claramente.

```
. histogram lningtot, frequency title("Histograma del Ln. del Ingreso Total") ///
xlabel(0(2.5)10) ylabel(0(250)800) ytick(0(125)800) start(0) width(.25) ///
norm gap(5)
(bin=49, start=0, width=2)
```

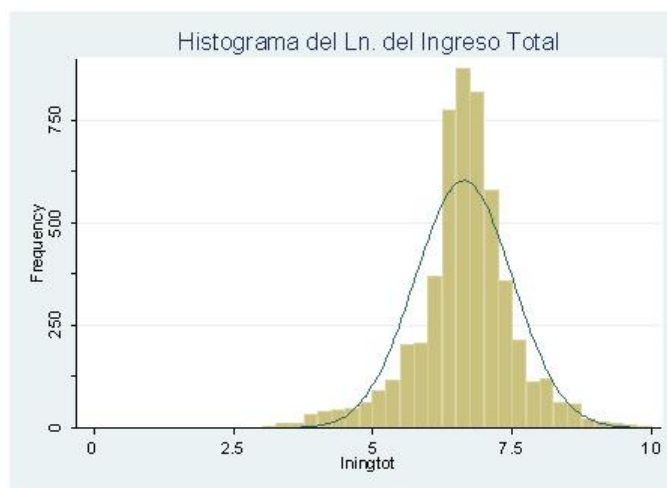


Figura 4.3: Histograma (3)

<sup>2</sup>Ver **help kdensity** para más detalle.

Supongamos que queremos saber como se distribuye el logaritmo del ingreso total según el sexo del individuo (representado por la variable **p107**). La figura 4.4 muestra un ejemplo en la cual expresamos en porcentajes sobre el eje de la ordenada y los datos agrupados en 8 bins.

```
. histogram lningtot, by(p107) percent bin(10)
```

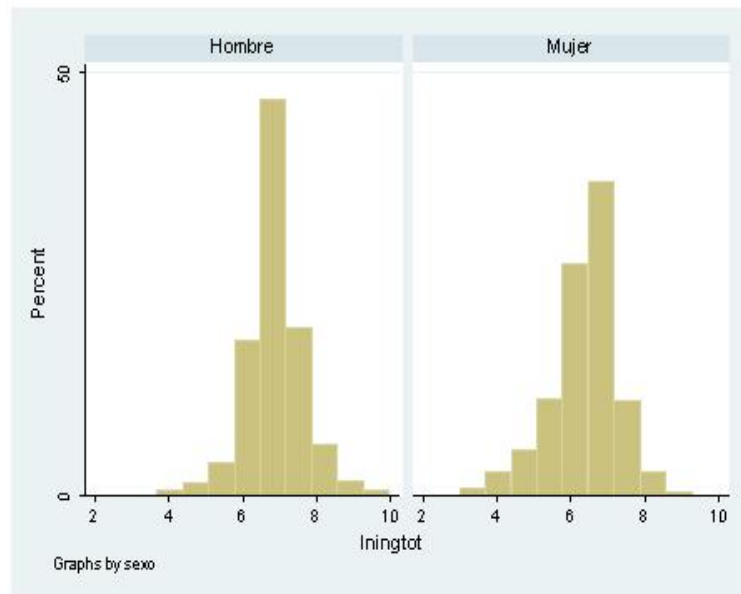


Figura 4.4: Histograma (4)

La siguiente figura 4.5 contiene un gráfico similar por el grupo de género, pero esta vez incluye un tercer elemento que señala la distribución para todos los individuos en su totalidad.

```
. histogram lningtot, by(p107,total) percent bin(10)
```

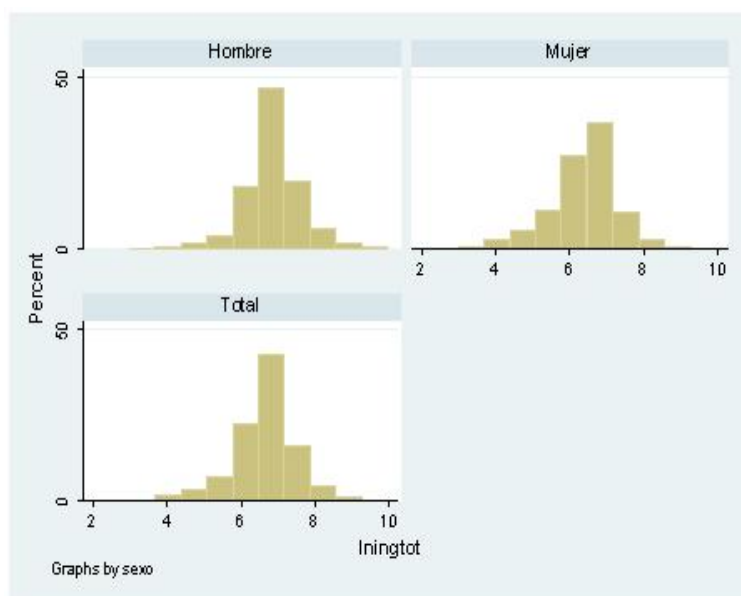


Figura 4.5: Histograma (5)

### 4.2.2. Graph Toway

#### Scatterplot

Los diagramas de dispersión de puntos (*scatterplot*) se accede a través del comando **graph twoway scatter**, cuya sintáxis general es:

```
graph twoway scatter x y
```

donde **y** es la variable que se muestra en el eje vertical y **x** en el eje horizontal.

Para ilustrar este tipo de gráfico, haremos un ploteo entre el logaritmo del ingreso total y los años de educación del individuo representado por la variable **p108**.

```
. graph twoway scatter lningtot p108
```

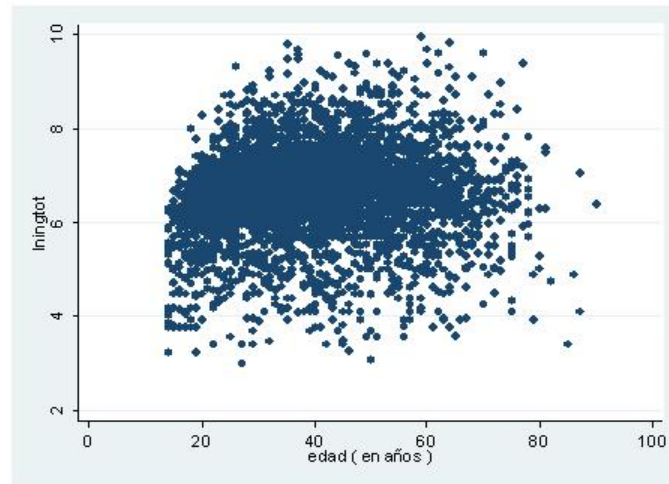


Figura 4.6: Scatter Plot (1)

De la misma manera que en el histograma, podemos usar **xlabel()**, **xtick()**, **xtitle()** para controlar las etiquetas de los ejes, los marcadores de los ejes, o títulos, respectivamente. El *scatterplot* también permite controlar las formas, colores, tamaños y otros atributos.

La figura 4.6 emplea marcadores por defecto, la cual son círculos sólidos. El mismo efecto podríamos obtener si incluimos la opción *msymbol(circle)* o escribimos esta opción de manera abreviada como *msymbol(O)*. La Tabla 4.1 muestra las diversas formas de marcadores para la dispersión de puntos.

<b>msymbol()</b>	<b>Abreviación</b>	<b>Descripción</b>
circle	O	circulo sólido
diamond	D	diamante sólido
triangle	T	triangulo sólido
square	S	cuadrado sólido
plus	+	signo +
x	X	letra "x"
smcircle	o	pequeño círculo sólido
smdiamond	d	pequeño diamante sólido
smsquare	s	pequeño cuadrado sólido
smtriangle	t	pequeño diamante sólido
smplus	smplus	pequeño signo +
smx	x	pequeña pequeño
circle_hollow	Oh	circulo con vacio
diamond	Dh	diamante con vacio
triangle_hollow	Th	triangulo con vacio
square_hollow	Sh	cuadrado con vacio
smcircle_hollow	oh	pequeño círculo con vacio
smdiamond_hollow	dh	pequeño diamante con vacio
smsquare_hollow	sh	pequeño cuadrado con vacio
smtriangle_hollow	th	pequeño diamante con vacio
point	p	punto pequeño
none	i	invisible

Tabla 4.1: Opciones de **mysymbol()**

Un uso interesante de este tipo de gráfico es hacer que el tamaño de los símbolos sean proporcionales a una tercera variable. De este modo, los ploteos se diferenciarán visualmente por medio de un ponderador **weight**. Si modificamos el *scatterplot* entre la variable **lningtot** y **p108**, haciendo que el tamaño de los símbolos se pondere por un factor de expansión poblacional **fa\_d8ef9**, como se muestra la figura 4.7. Dado que son muchas observaciones, puede causar mucha confusión y desorden, así que nos concentraremos solamente en aquellos individuos que no tienen un nivel educativo (representado por la variable **p109a** y cuyo valor es 1).

Para esto usaremos el ponderador de frecuencia **weight[ ]** y la opción de círculos vacíos, *msymbol(Oh)* <sup>3</sup>.

<sup>3</sup>El ponderador de frecuencia suele ser útil en otros gráficos, pero a la vez es un tópico complejo, porque los ponderadores **weight** vienen de diferentes formas y tienen diferentes significados para

```
. graph twoway scatter lningtot p108 [weight= fa_d8ef9] if p109a==1, msymbol(Oh)
```

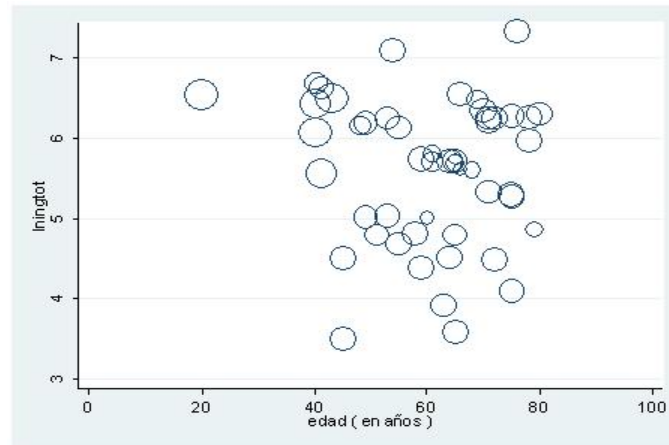


Figura 4.7: Scatter Plot (2)

El ejemplo de la figura 4.8 incluye una regresión lineal simple derivado del comando **twoway lfit** que ha sido añadido al grafico 4.6 especificando el siguiente símbolo ( || ).

```
. graph twoway scatter lningtot p108 if p109a==1, msymbol(S) mcolor(green) ///  
  || lfit lningtot p108)
```

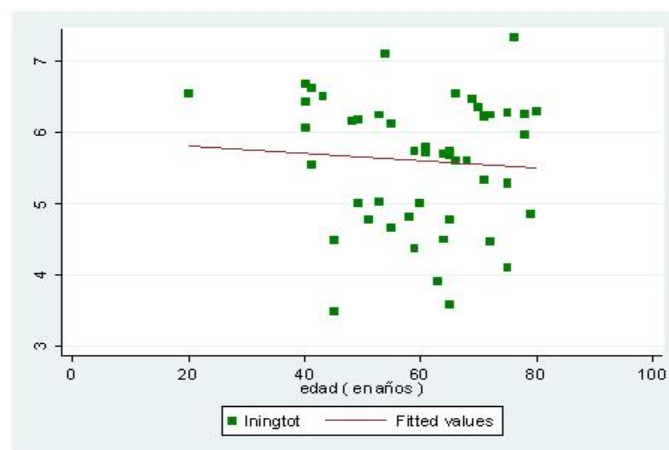


Figura 4.8: Scatter Plot (3)

diversos contextos. Para una información general de este tema en STATA, tipear **help weight**

Los marcadores de un *scatterplot* pueden identificarse con etiquetas. Por ejemplo, podemos desear observar el sexo de las personas en la figura 4.9.

```
. graph twoway scatter lningtot p108 if p109a==1, mlabel(p107) ///
msymbol(S) mcolor(purple) | lfit lningtot p108
```

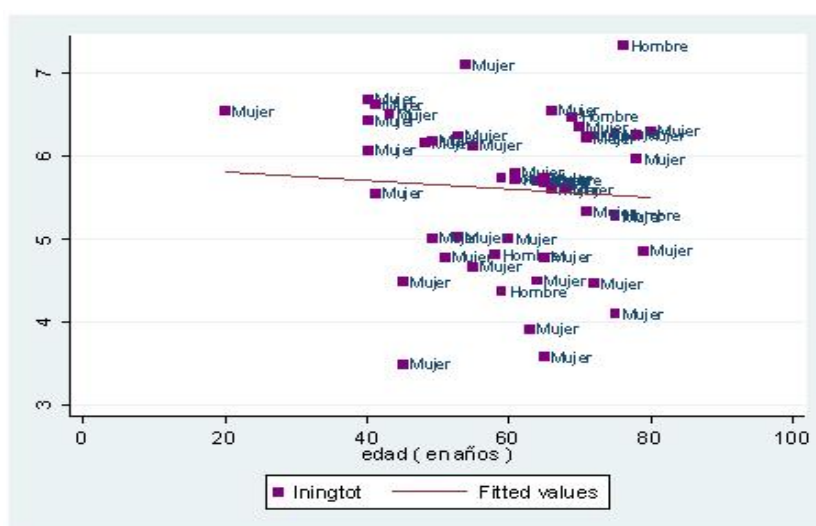


Figura 4.9: Scatter Plot (4)

La figura 4.10 muestra un *scatterplot* entre **lningtot** y **p108** para género. La relación entre estas dos grupos aparece una mayor pendientes en los hombres. La opción **xlabel()** e **ylabel()** en este ejemplo da las etiquetas para los ejes *x* e *y* de tres dígitos como máximo sin decimales, haciendo fácil de leer para pequeños sub-ploteos.

```
. graph twoway scatter lningtot p108, by(p107) ///
xlabel(,format(%3.0f)) ylabel(,format(%3.0f)) | lfit lningtot p108
```



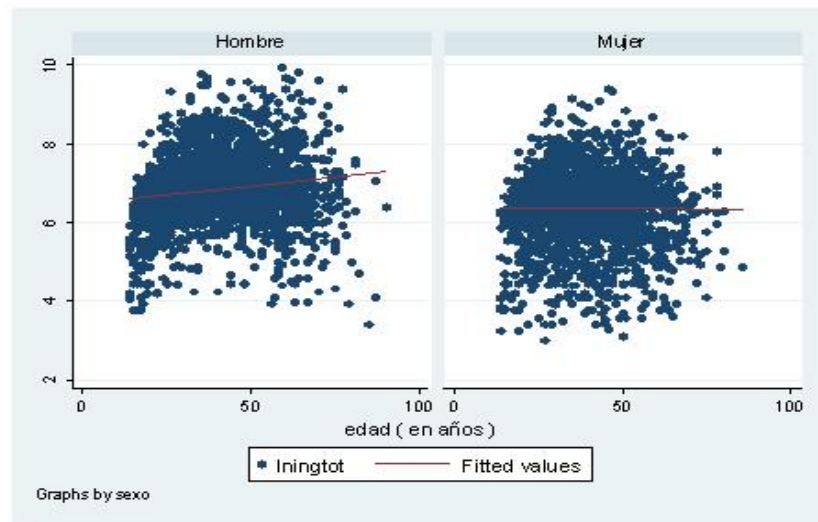


Figura 4.10: Scatter Plot (5)

### Multiples Scatterplots

El comando **graph matrix** nos muestra un útil análisis multivariado. Este comando otorga una gráfica compacta de la relación entre un número de variables por pareja, permitiéndole al analista observar los signos de no linealidad, outliers o clueter que puedan afectar al modelamiento estadístico. Este tipo de gráfico es útil si se quiere observar la influencia de una lista de variables explicativas a una variable dependiente.

La figura 4.11 muestra una matriz de scatterplot que implica la relación entre el logaritmo del ingreso total (**lningt**), la edad (**p108**), los años de estudios (**p109b**) y el total de horas trabajadas (**p209t**).

La opción **half** especificado en la figura 3.11 hace que se muestre solo la parte triangular inferior de la matriz ya que la parte superior es simétrica y redundante.

```
. graph matrix lningt p108 p109b p209t , half msymbol(0h)
```

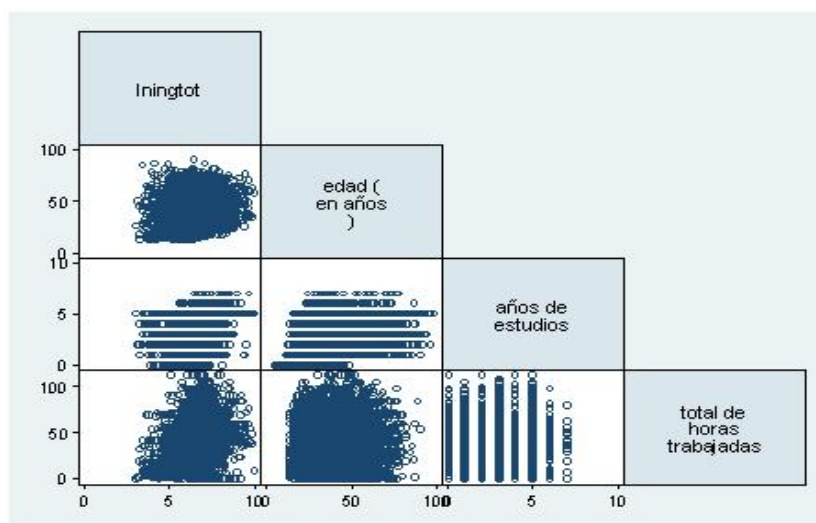


Figura 4.11: Multiples Scatter Plot

### Ploteo de Lineas (Line Plot)

Si usamos la base de datos *data\_trim.dta*, el cual contiene la serie de las variables del Producto Bruto Interno (**pbi**), las Importaciones (**m**) y los componentes de la demanda agregada (consumo privado (**c**), inversión (**i**), gasto público (**g**) y exportaciones (**x**)), todas medidas en millones de nuevos soles de 1994, desde el primer trimestre del 2003 hasta el tercer trimestre del 2011 y teniendo como fuente de información al BCRP.

Un simple ploteo para los componentes de la Oferta Agregada (Producto Bruto Interno y las Importaciones) pueden ser construidos señalando una grafica lineal de ambas variables a través del tiempo (**time**).

La figura 4.12 muestra una caída en el año 2009 producto de la crisis internacional, sin embargo, se nota la pronta recuperación para el siguiente año.

```
. graph twoway line m pbi time
```

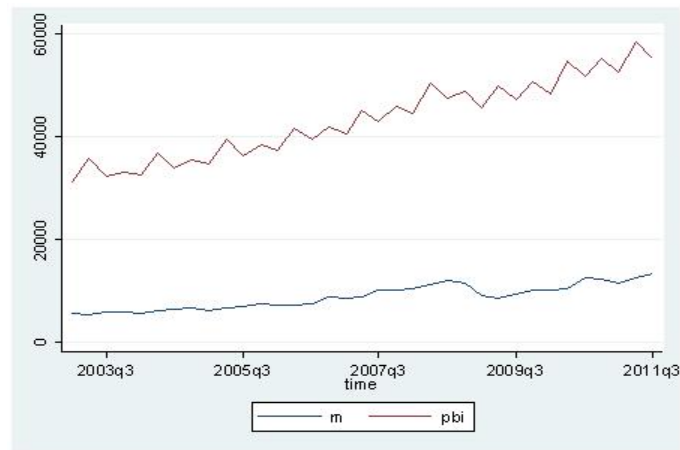


Figura 4.12: Line Plot (1)

En la figura 4.12, STATA elige por defecto una línea sólida azul para la primera variable **pbi**, y una línea sólida roja para la segunda variable **m**. Además de una leyenda en la parte inferior que muestra el significado de las variables. Si se mejora este gráfico a través de un arreglo en la leyenda, suprimiendo el título redundante en el eje x y colocando un título al gráfico, como se ve en la figura 4.13.

```
. twoway line m pbi time, legend(label(1 "Importaciones") ///
label (2 "PBI") position(3) ring(0) rows(2)) xtitle("") ///
title("Evolución del PBI e Importaciones" "2003-I - 2011-III")
```

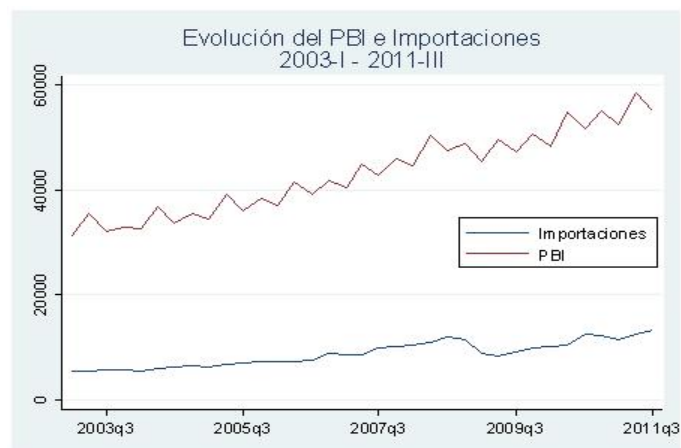


Figura 4.13: Line Plot (2)

Se tiene sub-opciones para la opción **legend()** la cual se colocan dentro de los paréntesis y se señalan en la Tabla 4.2 como sigue:

<b>legend()</b>	<b>Descripción</b>
<i>label (1 "Importaciones ")</i>	La etiqueta para la primera variable del eje <b>y</b>
<i>label (2 "PBI")</i>	La etiqueta para la segunda variable del eje <b>y</b>
<i>position(3)</i>	Establecer la legenda a las 3 de la hora del reloj (superior derecha)
<i>ring(0)</i>	Establecer la legenda entre los espacio del ploteo
<i>rows(2)</i>	Dice que la legenda tenga dos filas

Tabla 4.2: Opciones - **legend()**

La figura 4.12 y 4.13 conecta de una manera simple cada punto de la data con un segmento de recta. Otras estilos de conexiones son posibles, usando la opción **connect()**. Por ejemplo, *connect(stairstep)* o equivalentemente *connect(J)* generaría puntos para ser conectados en forma de escalera. La figura 4.14 ilustra el ploteo en forma de escalera para la variable del consumo privado (**c**).

```
. graph twoway line c time, connect(stairstep)
```

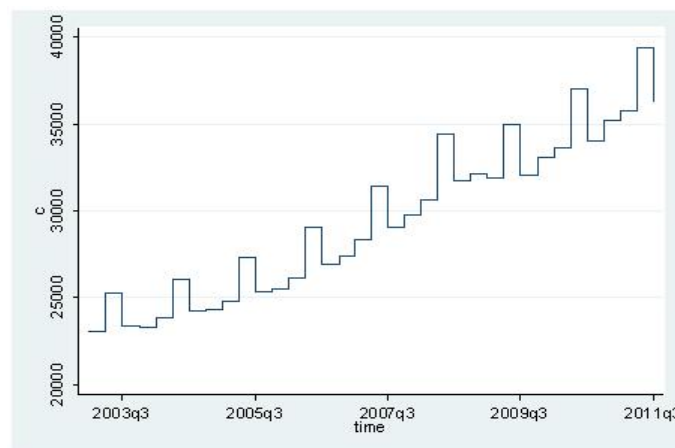


Figura 4.14: Line Plot (3)

Otras formas de conexión se muestran en la Tabla 4.3. Por defecto, el segmento de línea recta corresponde a `connect(direct)` o `connect(l)`<sup>4</sup>.

<code>connect()</code>	Abreviación	Descripción
<code>none</code>	<code>i</code>	no conecta puntos
<code>direct</code>	<code>l</code>	conecta con líneas rectas
<code>ascending</code>	<code>L</code>	es similar a <code>direct</code> solo si $x(i+1) > x(i)$
<code>stairstep</code>	<code>J</code>	recta constante, luego vertical
<code>stepstairs</code>		vertical, luego se mantiene constante

Tabla 4.3: Opciones - `connect()`

La figura 4.15 repite este ploteo escalonado del consumo privado, pero con algunas modificaciones de las etiquetas de los ejes y títulos. La opción `xtitle("")` no presenta ningún título en el eje  $x$ . la opción `angle()` permite definir en este caso la alineación de los valores en el eje  $y$ .

```
. graph twoway line c time, connect(stairstep) xtitle("") ///
yttitle("Millones de Nuevos Soles de 1994") ///
ylabel(, angle(horizontal)) clpattern(dash) ///
title("Evolución Consumo Privado" "2003-I - 2011-III")
```

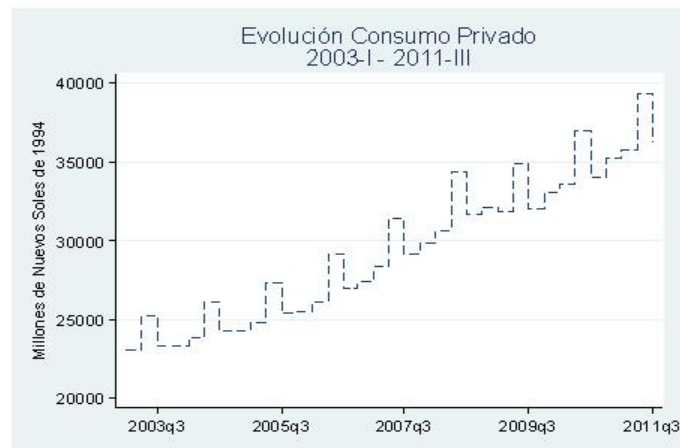


Figura 4.15: Line Plot (4)

Otro modo de especificar el tipo de línea que se desea usar es a través de la opción `clpattern()`, que nos permite elegir un patrón de línea y se muestra en la Tabla 4.4:

<sup>4</sup>Para más detalle, ver `help connectstyle`

<b>clpattern()</b>	<b>Descripción</b>
<i>solid</i>	línea sólida
<i>dash</i>	guiones
<i>dot</i>	puntos
<i>dot_dash</i>	puntos y guiones
<i>shortdash_dot</i>	guiones pequeños con puntos
<i>longdash</i>	guiones grandes
<i>longdash_dot</i>	guiones grandes con puntos
<i>blank</i>	línea invisible
<i>formula</i>	por ejemplo: <b>clpattern(-.); clpattern(-..)</b>

Tabla 4.4: Opciones - **clpattern()**

Para la siguiente figura 4.16 se grafica la evolución trimestral del producto bruto interno, el consumo privado y las importaciones. Note que las opciones *connect()*, *clpattern()* y *legend()* son utilizados en este ejemplo.

```
. graph twoway line pbi c m time, connect(line line staircase) ///
title("Evolución del PBI, Importaciones y Consumo Privado" "2003-I - 2011-III") ///
xtitle("") ytitle("Millones de Nuevos Soles de 1994") ///
clpattern(solid longdash dash) ylabel(, angle(horizontal)) ///
legend( label (1 "PBI") label (2 "Consumo") label(3 "Importaciones") ///
position(10) ring(0) rows(3))
```

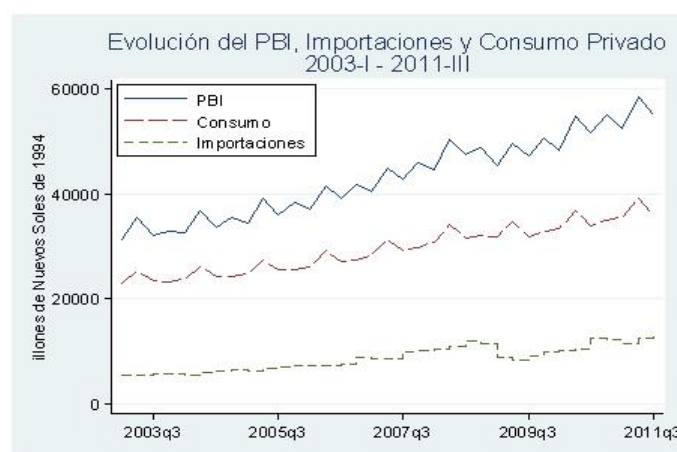


Figura 4.16: Line Plot (5)

### Ploteo de Líneas Conectadas (Connected-Line)

En el ploteo de líneas de la subsección anterior, los puntos de los datos son invisibles y vemos solo la conexión de las líneas. El comando **graph twoway connected** crea ploteo una conexión de puntos en la cual acomodamos la imagen mostrando un control de los marcadores de símbolos, patrón de líneas, ejes y leyenda. La figura 4.17 nos muestra un ejemplo de un ploteo de líneas conectadas a través del tiempo de las variables **pbi** y **c**.

```
. graph twoway connected pbi m time , msymbol(T oh) clpattern(dash solid) ///
yttitle("Miles de Tonelada") xtitle("") ///
title("Evolución del PBI y Consumo Privado" "2003-I - 2011-III") ///
ylabel(, angle(horizontal)) ///
legend(label(1 "PBI") label(2 "Consumo") ///
position(3) rows(2) ring(0))
```

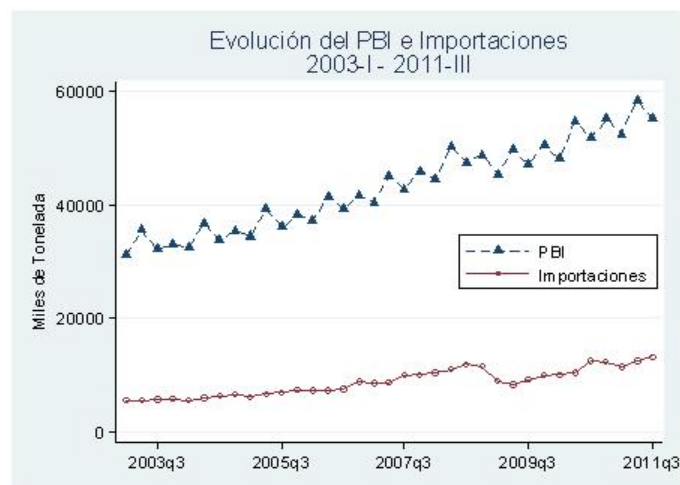


Figura 4.17: Line Connected Plot

### Otros Tipos de Scatter Plot

Además de los ploteos con líneas y scatterplot, el comando **graph twoway** presenta una amplia variedad de otros tipos <sup>5</sup>

<sup>5</sup>Para ver toda la lista de posibles tipos de gráficos con el comando **graph twoway** tipear **help twoway**.

Una observación que se puede hacer es que existen comandos como **graph twoway bar** y **graph twoway dot** que son muy distintos a los tipos de gráficos de barras (**bar**) y puntos (**dot**) respectivamente. Las versiones del *twoway* provee varios métodos para plotear una variables *y* contra otra variable *x*; además tienen la ventaja de sobreponer otros gráficos del *twoway* para formar gráficos más complejos. Por otro lado, las versiones que no son del *twoway* proveen modos de ploteos usando resúmenes estadísticos (tal como media o mediana) de las variables *y* contra las categorías de otras variables *x*.

Mucho de estos tipos de ploteos son útiles en la composición del gráfico final, que se construye por superposición de dos o más ploteos simples. El gráfico 4.18 muestra un ploteo de áreas de las variables **pbi** y **c**.

```
. graph twoway area pbi m time
```

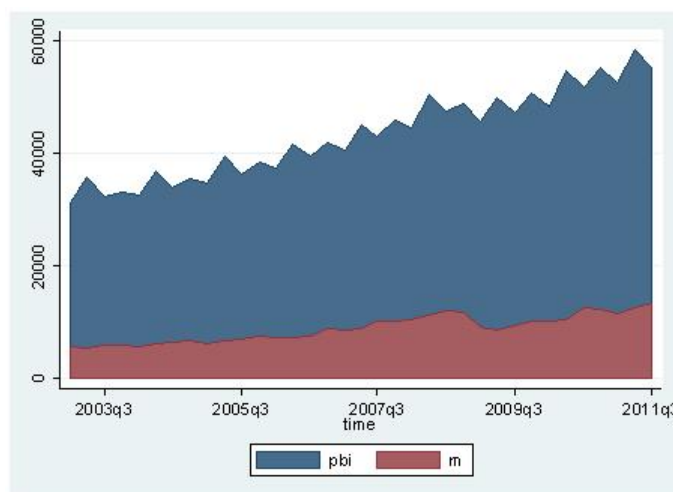


Figura 4.18: Otros Plot (1)

El color de las áreas pueden ser controlados por la opción **bcolor**<sup>6</sup>. Por ejemplo, el gris oscuro (*gs0*) es actualmente el color negro. Por ejemplo, la escala en grises se encuentra entre el valor 0 y 16. El color gris más ligero (*gs16*) es blanco. En la figura 4.19 muestra un ligero gris para este gráfico.

```
. graph twoway area pbi m time
```

<sup>6</sup>Tippear **help colorstyle** para ver la lista de colores.



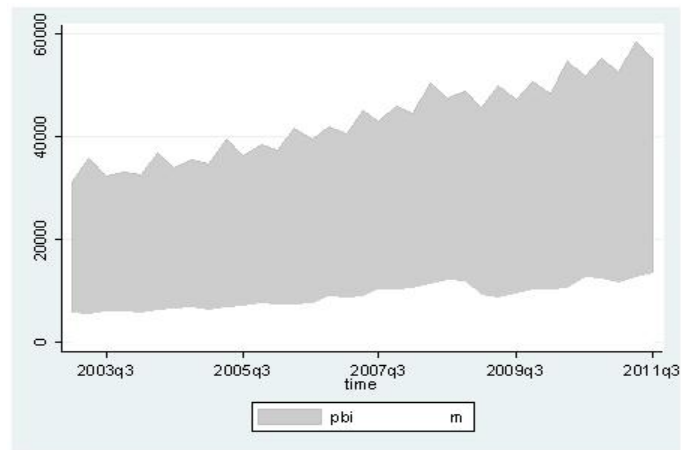


Figura 4.19: Otros Plot (2)

La figura 4.20 usa esta media de consumo privado (29607.66 millones de soles de 1994) como la base de un plot de líneas punteadas (**spike**), en la cual sobresalen líneas hacia arriba y hacia abajo a partir de esta media referencial. La opción **ylines(29607.66)** traza una línea horizontal en 29607.66.

```
. sum c
. graph twoway spike c time, ///
base(29607.66) yline(29607.66) ylabel(, angle(horizontal))
```

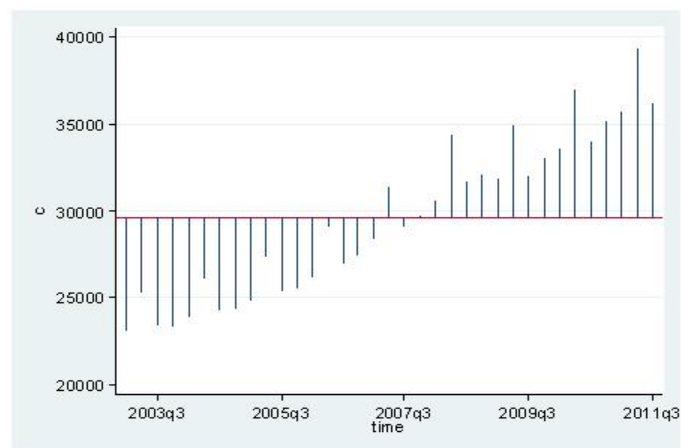


Figura 4.20: Otros Plot (3)

Una diferente vista de la misma data se muestra en la figura 4.21, donde se emplea la regresión mínima para suavizar la serie de tiempo con **graph twoway lowess**. La opción de ancho de banda, *bwidth(.4)*, especifica una curva basada en el suavizamiento de los datos que son derivados de la regresión ponderador entre una banda que cubre el 40 % de la muestra. El ancho de la banda pequeño sea tal como *bwidth(.2)*, o 20 % de la data, debería darnos un mayor ajuste. Una curva suavizada que sea más semejante a la data original. Altos anchos de bandas como *bwidth(.8)*, por defecto tendría un suavizamiento más radical.

```
. graph twoway lowess c time, ///
bwidth(.4) yline(29607.66) ylabel(, angle(horizontal))
```

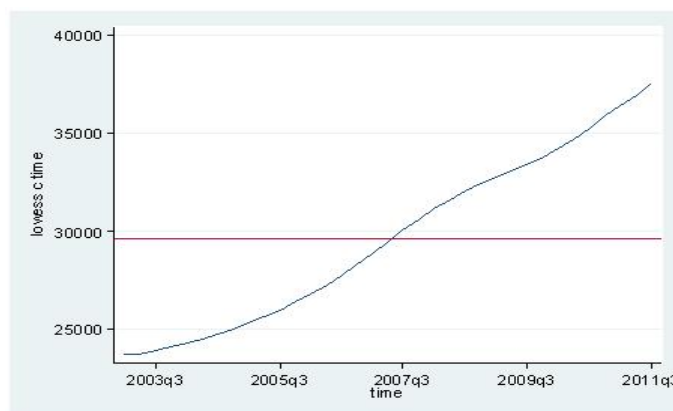


Figura 4.21: Otros Plot (4)

### 4.2.3. Gráfico de Caja y Bigote (Box Plot)

La gráfica de caja y bigote brinda información acerca del centro, amplitud, simetría y outliers con solo un vistazo. Para obtener este gráfico, se debe tipear el comando de la siguiente forma:

```
graph box x
```

Si diversas variables tienen escalas similares, podemos comparar sus distribución con la siguiente sintaxis:

```
graph box x y z
```

Para esta ocasión, volveremos a utilizar la base *trim\_dic08-ene-feb09.dta* de la Encuesta Permanente de Empleo (EPE). La figura 4.22 compara la distribución del logaritmo del ingreso total de la persona por género.

```
. use trim_dic08-ene-feb09.dta,clear

. *generamos el logaritmo del ingreso total
gen lningtot=ln(ingtot)
sum lningtot, detail //copiamos el valor de la mediana

graph box lningtot, over(p107) yline(6.684612)
```

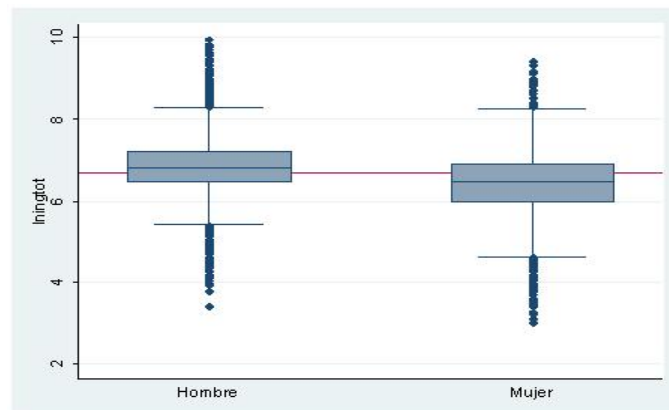


Figura 4.22: Box Plot (1)

El mediana del logaritmo del ingreso total de los hombres suele ser mayor que el de las mujeres. Por otro lado, el ingreso de los hombres presenta mayor variabilidad. La mediana por género (la línea entre las cajas) en la figura 4.22 puede ser comparado con la mediana considerando todas las personas por la opción `ylines(6.68)`.

Las cajas en estos gráficos se extienden desde el primer hasta el tercer cuartil, una distancia denominada *rango intercuartil (IQR)*. Esta además contiene aproximadamente la mitad, el 50 %, de la data. Los outliers, definidos como observaciones mayores a 1.5 IQR del primer o tercer cuartil, la cual se plotean separadamente de la caja. La caja y bogote en STATA define los cuartiles de la misma manera que el comando *summarize, detail*.

Numerosas opciones controlan la apariencia, forma y detalles de las cajas en este diagrama <sup>7</sup>. La figura 4.23 demuestra alguna de estas opciones, además del arreglo horizontal de *graph box*, usando el logaritmo del ingreso total (**lningt**). La opción *over(p107,sort(1))* hacen que las cajas se ordenan de forma ascendente acorde a la primera variable (en este caso ordena según sus medianas y la única variable que existe). La opción *intensity(30)* controla la intensidad de la sombra de las cajas, estableciéndole algo menos oscuro que el default (ver figura 4.22). La línea vertical marca la mediana total (6.684612), la cual se crea con la opción *yline()*, en vez de *xline()*.

```
. graph hbox lningt, over(p107,sort(1)) yline(6.684612) intensity(80)
```

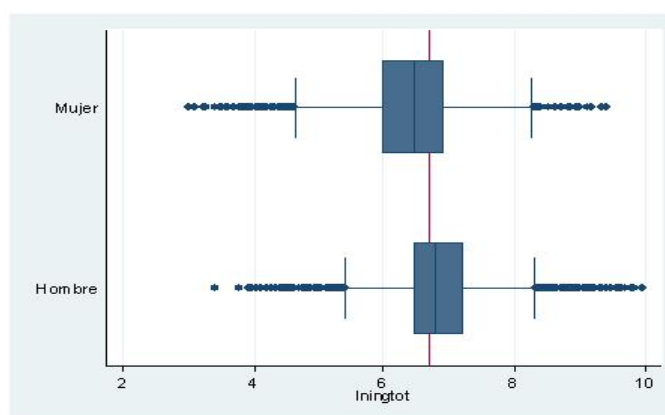


Figura 4.23: Box Plot (2)

La gráfica de caja y bigote para los años de educación en la figura 3.23 no solo la diferencia entre las medianas, sino también la presencia de outliers, principalmente en el caso de los hombres.

#### 4.2.4. Gráfico de Pastel (Pie)

Este estilo es muy popular en las presentaciones de graficas, siempre y cuando tengan pocos valores para trabajar. El comando básico del gráfico de pie en el STATA tiene la forma:

---

<sup>7</sup>Ver **help graph box**.

```
graph pie x y w z
```

donde  $x$ ,  $y$ ,  $w$ ,  $z$  son variables que miden cantidades de algunas cosas en las mismas unidades (por ejemplo, pueden estar medidos en dinero, horas, personas, etc.).

En la base de la EPE, mostraremos la proporción de la población según su nivel educativo. Para esto crearemos variables ficticias para cada nivel educativo derivado de la variable **p107**, y luego agruparemos los niveles educativos en 5 grupos: Sin Nivel, Primaria, Secundaria, Superior No Universitario y Superior universitario.

La mayoría de la población presenta un nivel educativo de Secundaria, como se puede ver claramente en la Figura 4.24. La opción **pie(3,explode)** provoca el llamado de la tercera variable, *secundaria*, para ser puesta en énfasis al gráfico. La cuarta variable, *SNU*, es sombreado con un ligero color gris, **pie(4,color(gs13))**, para compararlos con los grupos de nivel educativo (es importante mencionar que existen otros colores que se pueden utilizar como `color(blue)` o `color(chranberry)`<sup>8</sup>).

```
. *generamos variables ficticias por nivel educativo

tab p109a, gen(p109a)

. *creamos variables para cada nivel educativo

gen sinivel=p109a1+p109a2
gen primaria=p109a3+p109a4
gen secundaria=p109a5+p109a6
gen snu=p109a7+p109a8
gen su=p109a9+p109a10

. *colapsamos la base de datos

collapse (sum) sinivel primaria secundaria snu su, by(p107)

. *Etiquetamos las variables colapsadas

label variable sinivel "Sin Nivel"
label variable primaria "Primaria"
label variable secundaria "Secundaria"
label variable snu "SNU"
label variable su "SU"

. *graficamos el pie

graph pie sinivel primaria secundaria snu su, pie(3, explode) ///
pie(4, color(gs13)) plabel(3 percent , gap(10)) ///
legend( position(6) rows(2) ring(1))
```

---

<sup>8</sup>Tippear `help colorstyle` para observar la lista de colores.

La opción **plabel(3 percent, gap(20))** genera una etiqueta de porcentaje que se señala en el pedazo (slide) correspondiente a la tercera variable, *secundaria*, con una brecha (gap) de 10 unidades separados del centro. Podemos ver que cerca del 46.22 % de la población a alcanzado un nivel de educación de secundaria. La opción **legend()** señala las cuatros variables localizado en la posición de las 6 en punto del reloj.

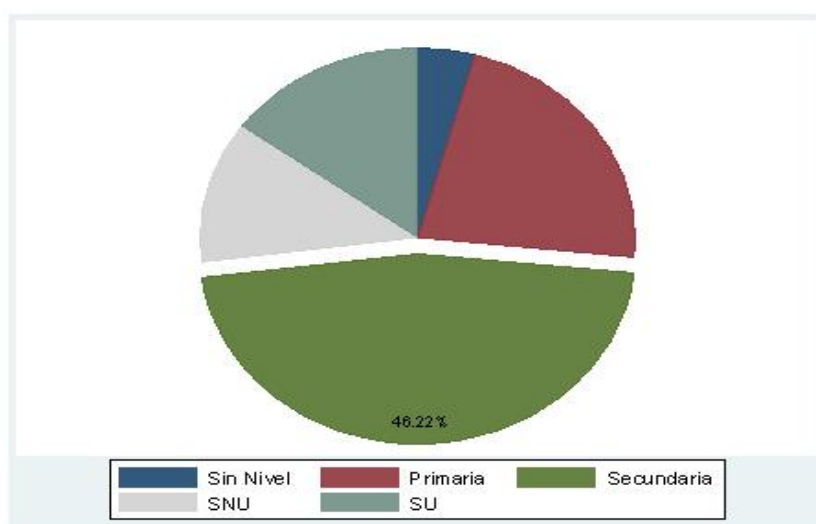


Figura 4.24: Pie Graph (1)

Las personas con nivel de educación secundaria son el grupo dominante en la figura 4.24, pero si mostramos el pastel separado por género con la opción **by(p107)**, emerge similares detalles mostrados en la figura 4.25. La opción **angle0()** especifica el ángulo del primer slide del pie. Estableciendo este primer slide un ángulo en cero (horizontal), orienta los slides de tal forma que las etiquetas son más fáciles de leer. La figura muestra que la mayoría de las mujeres y hombres alcanzan un nivel de educación secundaria.

```
. graph pie sinivel primaria secundaria snu su , pie(5, explode) ///
pie(4, color(gs13)) plabel(5 percent , gap(10)) ///
legend( position(11) rows(4) ring(1)) by(p107) angle0(0)
```

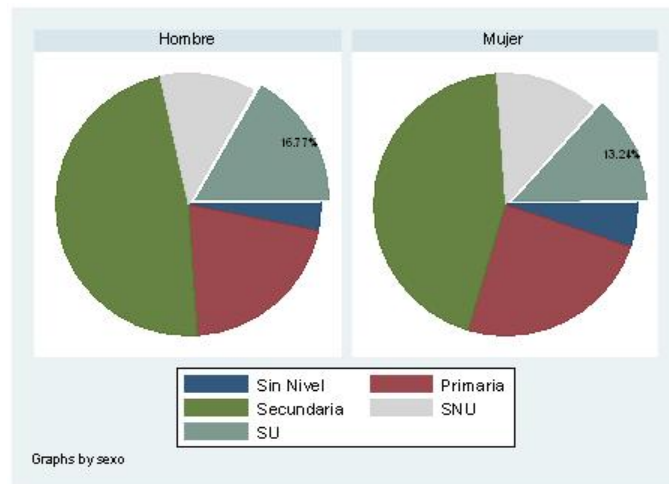


Figura 4.25: Pie Graph (2)

#### 4.2.5. Gráfico de Barras (Bar)

El gráfico de barras provee una simple y versátil exhibición conjunto de resúmenes estadísticos como media, mediana, suma o conteo. Para obtener barras verticales mostrando la media de la variable  $y$  frente a las categorías de  $x$ , tipeamos:

```
graph bar (mean) y, over(x)
```

Para barras horizontales mostrando la media de  $y$  frente a las categorías de  $x_1$ , por cada una de las categorías de  $x_2$ , tipeamos:

```
graph hbar (mean) y, over(x1) over(x2)
```

Este tipo de gráfico puede calcular los siguientes estadísticos:

- **mean** : Media, se calcula por defecto si no se especifica el estadístico.
- **sd** : Desviación estándar.

- **sum** : Suma.
- **rawsum** : Suma ignorando los ponderados especificados como opción.
- **count** : Cuenta el número de observaciones sin considerar los missing values.
- **max** : Máximo.
- **min** : Mínimo.
- **median** : Mediana.
- **p1** : Primer percentil.
- **p2** : Segundo percentil (y así hasta p99).
- **iqr** : Rangos intercuartiles.

La figura 4.26 indica la mediana del ingreso total en logaritmos por género. Vemos una diferencia a favor de los hombres de 0.33. Note que el eje vertical ha sido automáticamente etiquetado como “p50 of lningtot”, que significa el 50th percentil o mediana. La opción **blabel(bar)** etiqueta la parte superior de la barra con el valor de las medianas. **bar(1,bcolor(gs10))** especifica el color de las barras a un color gris ligero.

```
. graph bar (median) lningtot, over(p107) blabel(bar) bar(1,bcolor(gs10))
```

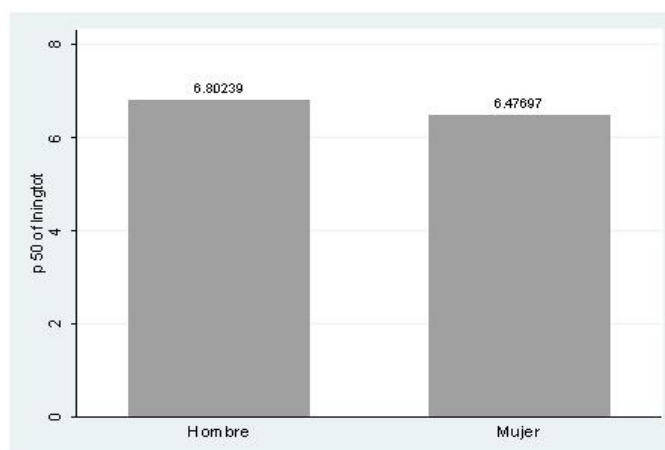


Figura 4.26: Bar Graph (1)



La figura 4.27 elabora la anterior idea añadiendo otra variable, la edad en años **p108**, y el color de la barra es gris oscuro. La etiqueta de la barra son *size(medium)*, haciéndoles más grande que el tamaño por defecto *size(small)*. Otras posibilidades para **size()** son las subopciones *tiny*, *medsmall*, *medlarge* o *large* <sup>9</sup>.

```
. graph bar (median) lningtot p108, over(p107) ///
  blabel(bar, size(medium)) bar(1,bcolor(gs10)) bar(2,bcolor(gs7))
```

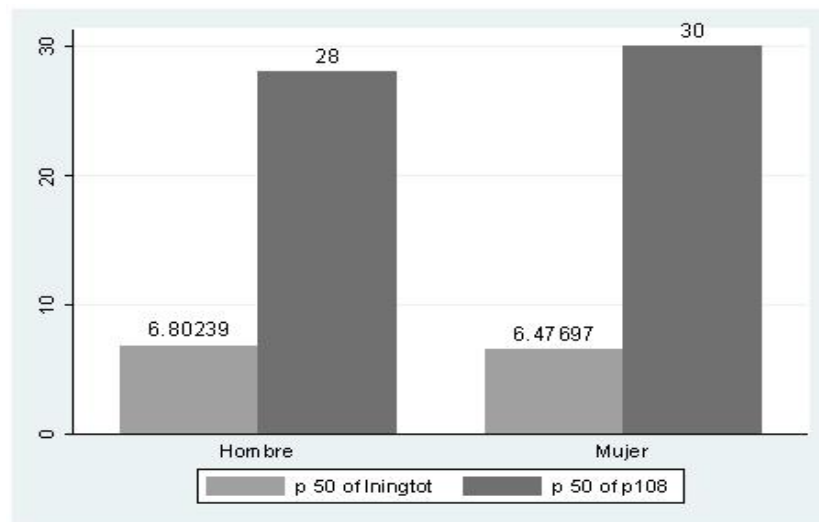


Figura 4.27: Bar Graph (2)

La figura 4.28 muestra las diferencias del ingreso con respecto a la edad donde el valor de la mediana del ingreso es mayor en el caso de los hombres y la edad en el caso de las mujeres.

```
. graph hbar (mean) lningtot, over(p109b) over(p107) yline(6.684612) ///
  title("Ingreso Promedio (logaritmos)" "según género y años de educación")
```

<sup>9</sup>Puedes ver una lista más detallada con el comando **help textsizestyle**.

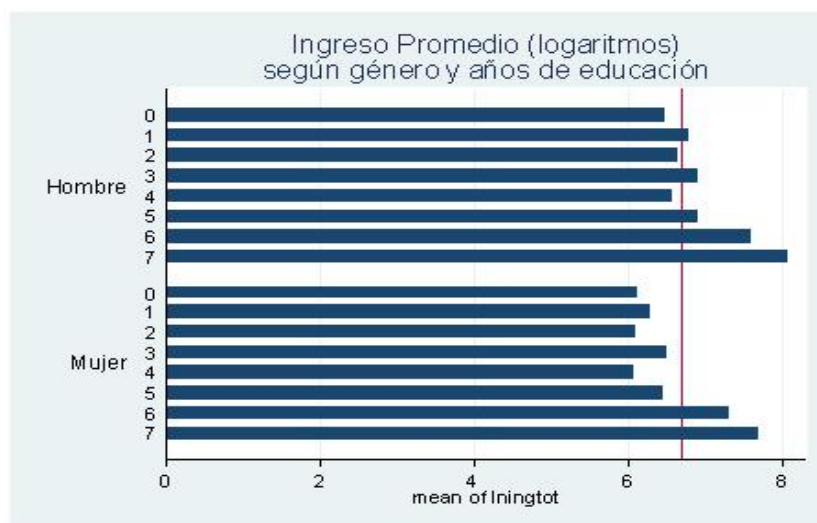


Figura 4.28: Bar Graph (3)

El orden de dos opciones **over()** controlan el orden en la organización del gráfico. Para este ejemplo utilizamos las barras horizontales (**hbar**), donde las opciones **ytitle()** y **yline()** se refieren al eje horizontal. En este caso, colocamos una línea horizontal que indica el valor de la mediana total de 6.684612 , *yline(6.684612)*, y será mostrado de forma vertical.

Las barras también pueden estar montadas o apiladas entre sí, como se muestra en la figura 4.29. Este ploteo, se basa en la generación de nuevas variables del nivel educativo, donde se emplea todas las opciones por defecto para graficar la composición de la población con respecto a su nivel de educación por género.

```
. *generamos variables ficticias por nivel educativo
tab p109a, gen(p109a)

. *creamos variables para cada nivel educativo

gen sinivel=p109a1+p109a2
gen primaria=p109a3+p109a4
gen secundaria=p109a5+p109a6
gen snu=p109a7+p109a8
gen su=p109a9+p109a10

graph bar (sum) sinivel primaria secundaria snu su, over(p107) stack
```

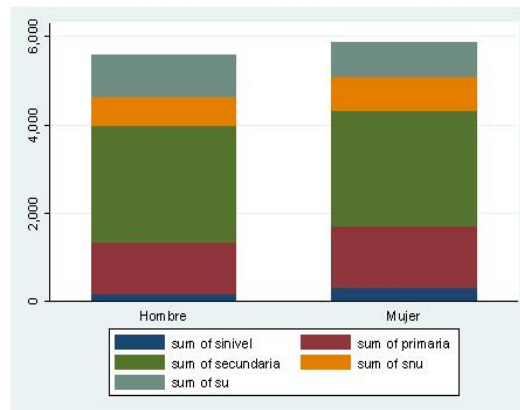


Figura 4.29: Bar Graph (4)

La figura 4.30 se vuelve a graficar este último ploteo con una mejor *leyenda* y *etiqueta los ejes*. La opción `over()` ahora incluye subopciones que reetiquetan los tipos de comunidad en el eje de la abscisa para dar mayor información. La opción `legend()` especifica tres filas en el mismo orden vertical. También se mejora la etiqueta de las legendas con `ytitle()` y `ylabel()` como opciones del formato del eje vertical.

```
. graph bar (sum) sinivel primaria secundaria snu su, ///
over(p107, relabel(1 "Varones" 2 "Feminas" )) ///
legend(rows(3) order(5 4 3 2 1) position(6) ring(1) ///
label(1 "Sin Nivel") label(2 "Primaria") ///
label(3 "Secundaria") label(4 "SNU") ///
label(5 "SU")) stack ///
yttitle("Personas") ylabel(0 (1000) 6000)
```

Mientras el pie de la figura 4.29 muestra el tamaño relativo (porcentajes) de los grupos según nivel educativo por género, esta última barra muestra sus tamaños absolutos. Consecuentemente, esta figura te dice algo más que el anterior: la mayoría de la población con un nivel superior universitario son hombres.

#### 4.2.6. Gráfico de Puntos (Dot Plot)

Los ploteos con puntos son igual de útiles que las gráficas con barras: comparando visualmente resúmenes estadísticos de una o más variables. Las opciones que usa el STATA para ambos gráficos son ampliamente similares, incluyendo la

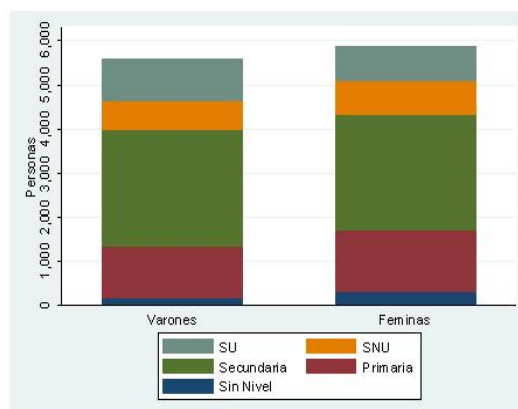


Figura 4.30: Otros Plot

elección de los estadísticos. Para ver este diagrama comparando las medianas de las variables  $x, y, w$  y  $z$ , debemos tipear:

```
graph dot (median) x y w z
```

Y para ver la comparación de promedios de la variable  $y$  según las categorías de  $x$ , escribimos:

```
graph dot (mean) y , over(x)
```

La figura 4.31 muestra un plot de puntos del ingreso total promedio en logaritmos y la edad promedio por nivel educativo creada (**niveduc**). La opción **over()** incluye una subopción, *sort(lningtot)*, la cual ordena la media del ingreso promedio para cada una de los niveles educativos, esto es desde el más bajo hasta el más alto ingreso total. También podemos especificar un triángulo sólido como marcador de símbolo para **lningtot** y círculos con un centro vacío para **p108**.

```
. *generamos una variable categorica de nivel educativo

gen niveduc=1 if sinivel==1
replace niveduc=2 if primaria==1
replace niveduc=3 if secundaria==1
replace niveduc=4 if snu==1
replace niveduc=5 if su==1
```

```

label define educa 1 "Sin Nivel" 2 "Primaria" 3 "Secundaria" 4 "SNU" 5 "SU"
label value niveduc educa

tab niveduc

graph dot (mean) lningtot p108, over(niveduc, sort(lningtot)) ///
marker(1, msymbol(T)) marker(2, msymbol(Oh))

```

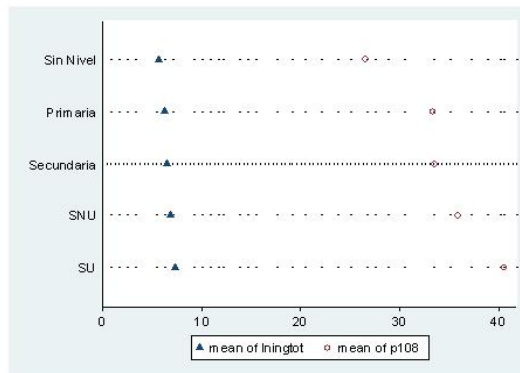


Figura 4.31: Dot Plot

Además, la figura 4.31 calcula solo 8 promedios, esto hace que sea fácil las comparaciones. Vemos que el ingreso total en logaritmos es mayor para las personas que tienen un nivel de educación superior universitario, así como también tienen una mayor edad en promedio. La gráfica en barras podría darnos la misma información, pero una ventaja de estos gráficos es la forma de compactar los datos. Los ploteos de puntos (particularmente cuando se quiere ordenar por estadísticos de interés) es fácil de entender incluso con varias filas.

### 4.3. Añadiendo Textos a los Gráficos

Los títulos, los nombres de gráficos y las notas pueden ser añadidos al gráfico para que sea más explicativo. Los títulos y subtítulos aparecen encima del área del ploteo; las opciones *note* (la cual puede documentar la fuente de los datos) y *caption* aparecen en la parte inferior <sup>10</sup>.

<sup>10</sup>Tippear **help text** para más información acerca de la especificación de los títulos en gráficos.

La figura 4.32 muestra el uso de estas opciones en un scatterplot sobre el ingreso total en logaritmos y la edad de cada uno de los individuos. La figura 4.42 también incluye títulos para ambos lados (derecha e izquierda) del eje  $y$ ,  $yaxis(1\ 2)$  y la parte superior e inferior del eje  $x$ ,  $xaxis(1\ 2)$ . Luego las opciones  $xtitle()$  y  $ytitle()$  se refieren al segundo eje específicamente, al incluirse la subopción  $axis(2)$ .

```
. graph twoway scatter lningtot p108 , yaxis(1 2) xaxis(1 2) ///
title("Es es un Título") subtitle("Este es un Subtítulo") ///
caption("Este es un caption") note("Esta es una Nota") ///
ytitle("Este es el Porcentaje de adultos fumadores") ///
ytitle("Este es el Eje Y 2", axis(2)) ///
xtitle("Porcentaje de adultos con Grado Superior") ///
xtitle("Este es el Eje X 2", axis(2))
```

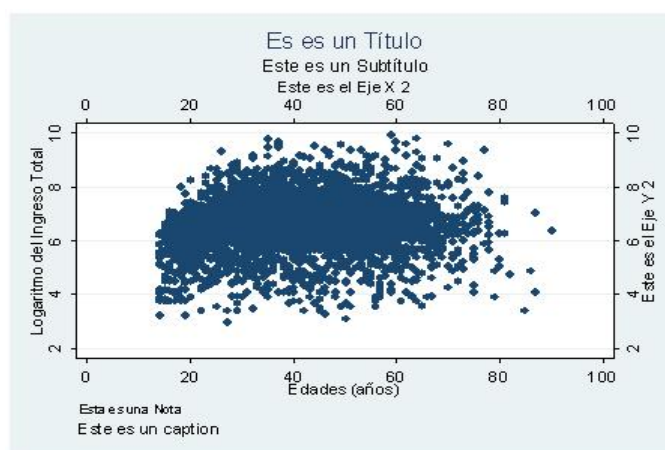


Figura 4.32: Texto en Gráficos (1)

El título añade el texto fuera del espacio de ploteo. También podemos añadir cajas de texto en coordenadas específicas en el espacio de ploteo. Diversos outliers se observan en este ploteo en la parte inferior derecha.

Los cuadros de texto son de instrumentos para identificar dichas observaciones en nuestro gráfico, como se señala en la figura 4.33. La opción `text(3 80 "Outliers")` establece la palabra Outliers en la posición  $x=80$  e  $y=3$  del scatterplot. De una forma similar podemos establecer la palabra "Agglomeración" en  $x=10$  e  $y=9$  y ubicarlo en un cuadro pequeño (con pequeños márgenes<sup>11</sup>) alrededor del nombre del estado.

Las cinco líneas de textos justificados hacia la izquierda son colocados al lado

<sup>11</sup>Ver `help marginstyle`.

inferior izquierdo (cada línea se especifica separadamente entre comillas), donde sus coordenadas son  $x=6.2$  e  $y=3.5$ . Algunos cuadros de texto o títulos pueden tener múltiples líneas, así que podemos escribir una parte del título en líneas diferentes escribiéndolo entre comillas diferentes, para luego definir el tipo de justificación. El cuadro de Aglomeración utiliza un formato de fondo por defecto, mientras que, el cuadro de la relación entre ambas variables se eligió un color de fondo blanco <sup>12</sup>.

```
. graph twoway scatter lningtot p108, yaxis(1 2) xaxis(1 2) ///
title("Es es un Título") subtitle("Este es un Subtítulo") ///
caption("Este es un caption") note("Esta es una Nota") ///
ytitle("Logaritmo del Ingreso Total") ///
ytlabel("Este es el Eje Y 2", axis(2)) ///
xtlabel("Edades (años)") ///
xttitle("Este es el Eje X 2", axis(2)) ///
text(3 80 "Outliers") ///
text(9 10 "Aglomeración", box margin(small)) ///
text(3.5 6.2 "Relación" "Directa" "entre" "Ingreso" "y Edad", ///
justification(left) box margin(small) bfcolor(white))
```

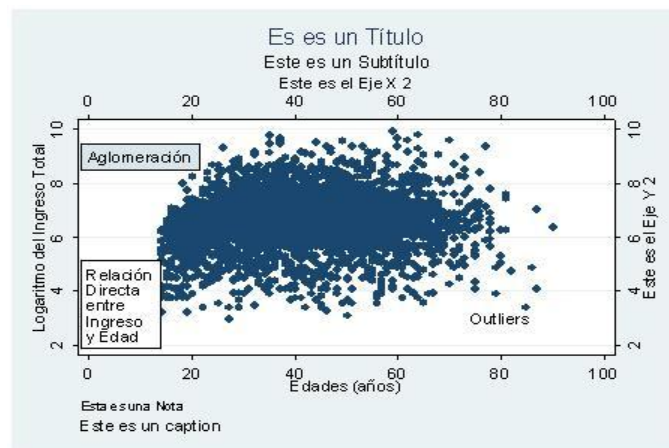


Figura 4.33: Texto en Gráficos (2)

## 4.4. Múltiples Ploteos

Dos o más graficos de la familia **graph twoway** pueden ser sobrepuestos en un único gráfico. La familia **twoway** incluye diversos modelos de ploteos tales como **lfit** (recta de regresión lineal), **qfit** (curva de regresión cuadrática) y más. Por ello, tales ploteos brindan información al mínimo.

<sup>12</sup>Ver **help textbox\_option** y **help colorstyle**.

Por ejemplo, la figura 4.34 describe la recta de regresión lineal, teniendo bandas al 95 % de nivel de confianza para la media condicional, de la regresión que surge entre *lningt* sobre *p109b*.

```
. graph twoway lfitci lningt p109b
```

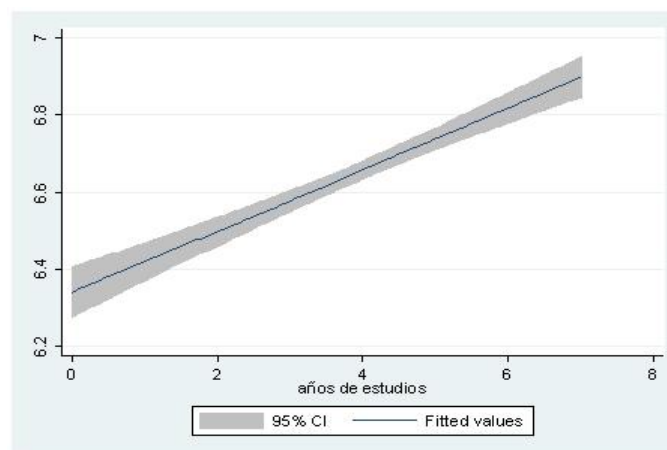


Figura 4.34: Ploteos Múltiples (1)

Un gráfico con mayor información cuando sobreponemos un scatterplot sobre la recta de regresión lineal, se puede ver en la figura 4.35. Para hacer esto, damos dos distintas indicaciones de comandos de gráficos, separado por `||`.

```
. graph twoway lfitci lningt p109b || scatter lningt p109b
```

El segundo ploteo (*scatterplot*) se coloca encima del primer ploteo en la figura 4.35. Este orden tiene consecuencia para el estilo de línea usado por defecto (solid, dashed, etc) y también para el marcado de símbolos (square, circle, etc) usado por cada subploteo. Lo más importante es tratar que los ploteos sean los más visibles posibles.

La figura 4.36 desarrolla la idea anterior, mejorando la imagen usando las opciones de etiquetas de ejes y leyenda. Por que dichas opciones se aplican al gráfico como un todo y no por separado, estas opciones son establecidas después del segundo separador `||`, seguido por una coma. La mayoría de estas opciones se asemeja a los ejemplos realizados anteriormente. La opción *order(2 1)* en este caso hace una nueva función: omite una de los tres ítems de la leyenda, tal que solo dos de ellos



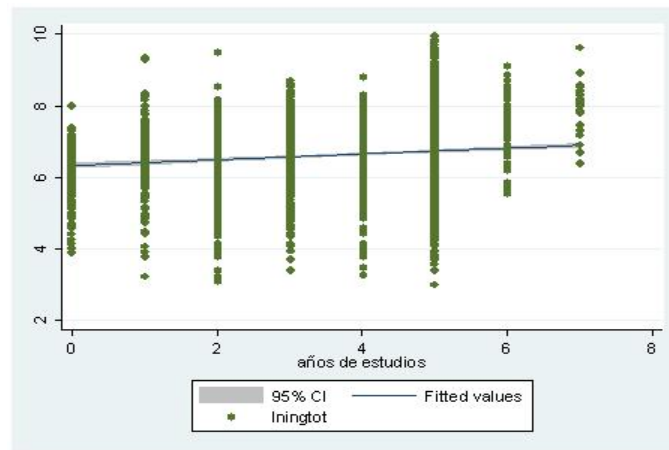


Figura 4.35: Ploteos Múltiples (2)

(2 de la regresión lineal, seguido por 1 del intervalo de confianza) aparezcan en la figura. Comparando esta leyenda con la figura 4.35 vemos la diferencia. Aunque listemos solo dos items en la leyenda, aun es necesario especificar tres filas en el formato de la leyenda (`rows(3)`) como si cada uno de los items estan retenidos.

```
. graph twoway lfitci lningtot p109b || scatter lningtot p109b, ///
ylab(2 (1) 10, angle(horizontal)) ///
xtitle("Años de Educación") ///
ytitle("Ingreso Total (Logaritmos)") ///
note("Encuesta Permanente de Empleo - EPE") ///
legend(order(2 1) label(1 "95% c.i") label(2 "Regresión Lineal") ///
rows(3) position(5) ring(0))
```

Ambos scatterplot (**lfitci** y **scatter**) en la figura 4.36 presentan la misma escala de los ejes  $x$  e  $y$ , pero cuando ambas variables de interés tienen distintas escalas, nosotros necesitaríamos escalas independientes.

La figura 4.37 ilustra este caso juntando dos ploteos con líneas basado sobre la data de las series del PBI y sus componentes, *data\_trim.dta*. Estas figuras combinan series de tiempo del gasto público e inversión privada, ambos expresados en millones de soles de 1994.

El ploteo de **line** hace uso de la opción *yaxis(1)*, lo cual por defecto es el lado izquierdo y será usado para mostrar la variable consumo privado. El ploteo de la inversión privada usa el *yaxis(2)*, la cual por defecto es el lado derecho. Las opciones

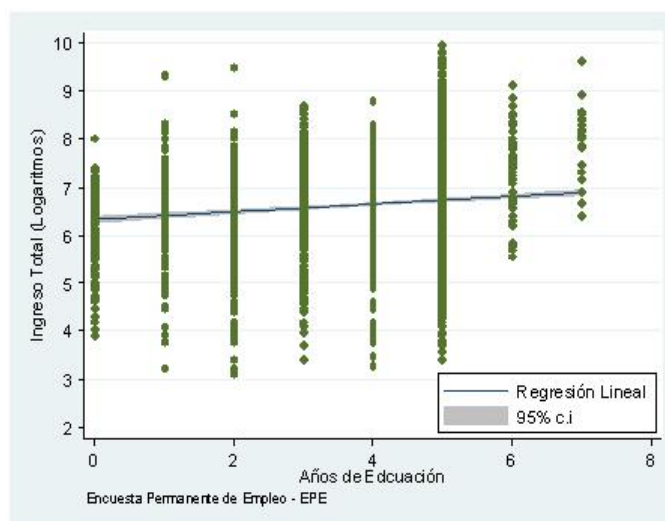


Figura 4.36: Ploteos Múltiples (3)

**ytitle()** y **yline()** se incluyen con la subopción *axis(1)* o *axis(2)*, declarando cual de los ejes de la ordenada se refieren.

```
. graph twoway line g time, ///
yaxis(1) ytitle("Gasto Público ",axis(1)) yline(29607.66,axis(1)) ///
|| line i time, ///
yaxis(2) ytitle("Inversión Privada",axis(2)) ///
yline(10145.6, axis(2) lpattern(dot)) ///
|| , ///
xtitle("") ///
legend(position(11) ring(0) rows(2) order(2 1) ///
label(1 "Gasto Gobierno") label(2 "Inversión")) ///
note("Fuente: Banco Central de Reservas del Perú - BCRP")
```

Para localizar el gasto público, la inversión privada y las exportaciones, necesitamos tres escalas verticales independientes. La figura 4.38 envuelve tres ploteos superpuestas, la cual todos están en el lado izquierdo del eje *y* por defecto. La forma básica de estos tres ploteos es como sigue:

**connected x time**; plotea una línea conectada en la variable de exportaciones a través del tiempo, usando **yaxis(3)** la cual debería estar ubicado en la parte superior izquierda del eje *y*. Los rangos de escala en el eje *y* va desde 4000 a 12000, sin líneas horizontales como malla. Su título es *Exportaciones*. Este título es localizado en la posición noroeste, **placement(nw)**

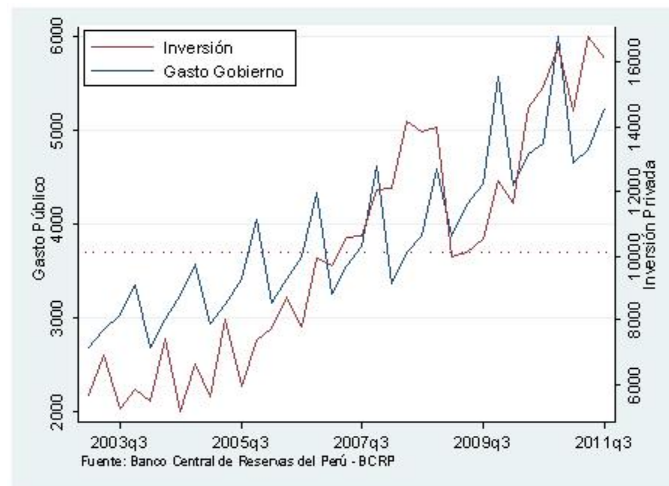


Figura 4.37: Ploteos Múltiples (4)

**line i time;** plotea una línea conectada en la variable de inversión privada a través del tiempo, usando `yaxis(2)` y rangos de escala entre 6000 hasta 16000, con las etiquetas por defecto.

**connected g time;** plotea una línea conectada en la variable del gasto del gobierno a través del tiempo, usando `yaxis(1)`. El título se localiza en la parte suroeste.

Brindando estos tres componentes del ploteo de forma conjunta, el comando para elaborar la figura 3.38 aparece a continuación:

```
. graph twoway connected x time, yaxis(3) yscale(range(4000,12000) axis(3)) ///
yttitle("Exportaciones",axis(3) placement(nw)) ///
clpattern(dash) ///
|| line i time, yaxis(2) yscale(range(6000,16000) axis(2)) ///
ylabel(, nogrid axis(2)) ///
yttitle("Inversión Privada",axis(2)) ///
clpattern(solid) ///
|| connected g time, yaxis(1) yscale(range(2000,6000) axis(1)) ///
ylabel(, nogrid axis(1)) ///
yttitle("Gasto Público",axis(1) placement(sw)) ///
|| , ///
legend(position(5) ring(0) rows(3) label(1 "Gasto del Gobierno") ///
label(2 "Inversión Privada") label(3 "Exportaciones")) ///
xtitle("")
```

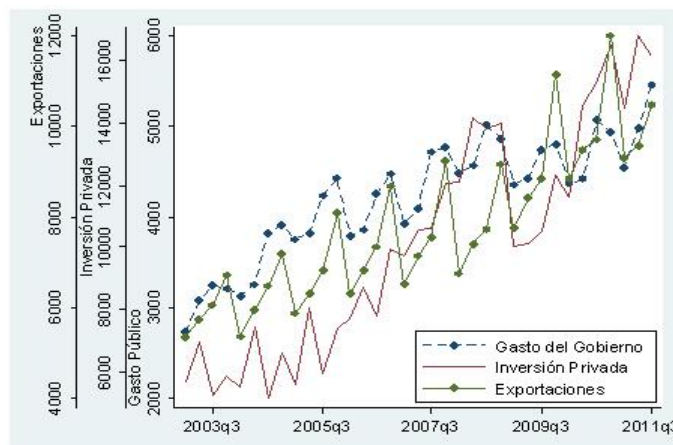


Figura 4.38: Ploteos Múltiples (5)

## 4.5. Guardar, Combinar y Exportar Gráficos

Una vez que el gráfico fue creado, este puede ser guardado. Para esto STATA usa el comando **save** principalmente para guardar los gráficos en STATA con extensión **\*.gph**. El procedimiento puede realizarse usando la opción *saving()* en la misma línea de comando del gráfico o a través del comando **graph save** después que el gráfico haya sido creado. Cuando guardamos en esta última manera, se puede volver a acceder a los gráficos para ser manipulados a gusto personal a través del Editor de Gráficos.

```
. *I Forma
graph twoway scatter g time,saving(g1, replace)

. *II Forma
tw (sc g time, msize(small)) (lfit g time, lwidth(medthick)), ///
title("Dispersión de Puntos" "y Línea OLS Ajustada")

graph save g2.gph,replace
```

Dos o más gráficos pueden combinarse en uno solo utilizando el comando **graph combine**.

```
. *Combinemos el grafico g1.gph y g2.gph
graph combine g1.gph g2.gph
```

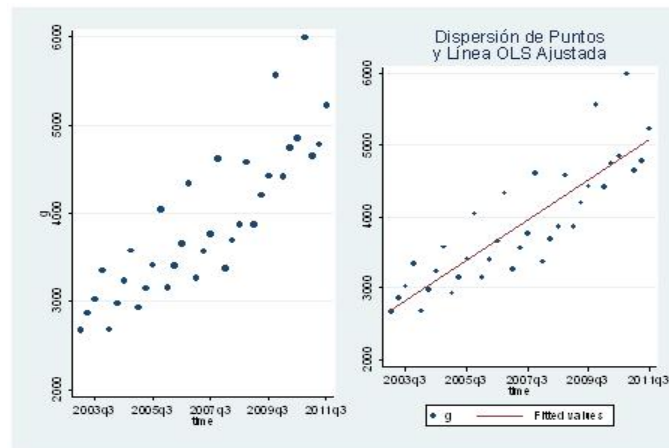


Figura 4.39: Graficos Combinados

```
graph save g12.gph,replace
```

Dado que la extensión del grafico por default del STATA es \*.gph, no es reconocido por otros programas, tal como procesadores de texto. Para guardar un gráfico en otro formato, se debería usar el comando `graph export`. Varios formatos están disponibles, incluyendo PostScript (.ps), Encapsulated PostScript (.eps), Window Metafile (.wmf), PDF (.pdf) y Portable Network Graphics (.png). La mejor selección del formato depende en parte de cual procesador de texto se usa, por eso es necesario una prueba de ensayo y error.

```
. *Exportar Gráficos

. //Guardamos el gráfico combinado como:

graph export g12.wmf, replace //Window meta-file
graph export g12.ps , replace //PostScript
graph export g12.eps, replace //Encapsulated PostScript
graph export g12.png, replace //Portable Network Graphics
```

## 4.6. Ejercicio Propuesto

A través de la base **EPE-abr-may-jun12.dta** proveniente de la Encuesta Permanente de Empleo (EPE) para el periodo trimestral Abril-Mayo-Junio 2012, in-

tente replicar algunos gráficos del *Informe Técnico: Situación del Mercado Laboral en Lima Metropolitana, trimestre móvil: Abril-mayo-Junio 2012*, el cual se mencionará a continuación. *Nota: En cada enunciado no se olvide filtrar para las personas mayores o iguales a los 14 años de edad (p108), usar la variable del factor de expansión (fa\_amj12) y personalizar los formatos según lo enseñado en este capítulo.:*

- El Gráfico Nro. 1 *Composición de la población en edad de trabajar, según condición de actividad*: Para esto, calcule dos nuevas variables dicótomicas, una para la población que pertenece a la *PEA* (engloba a ocupado, desocupado abierto y oculto) y otra a la *PEI* (equivalente a no pea), usando la variable **ocu200**. Luego use el comando **bar** con las opciones *stack* y *percentages* para obtener el gráfico en el periodo de análisis.
- El Gráfico Nro. 8 *Lima Metropolitana: Población ocupada por rango de horas trabajadas por semana*: Se debe generar una variable auxiliar donde todos sus valores sea igual a la unidad y otra variable categórica que englobe todos los rangos de las horas trabajadas a la semana (no olvidar etiquetar los valores de esta variable categórica con el nombre de los rango descritos en el gráfico a replicar). Luego aplicar el comando **graph bar** con la preopción (**sum**) para obtener los valores totales de esta variable auxiliar. Finalmente se sugiere usar las opciones **over()** donde se agrupe según la variable categórica y **blabel()** para mostrar la etiqueta de los valores totales.
- El Gráfico Nro. 15 *Lima Metropolitana: Nivel de educación de la PEA desempleada con experiencia laboral (cesantes)*: Es necesario calcular una variable ficticia para cada nivel educativo (representado en la variable **p109a**), como por ejemplo: **primaria** tendrá el valor de 1 cuando el individuo no tenga nivel de educación o alcanzó el nivel inicial, primaria incompleta e completa; **secundaria** cuando tenga secundaria incompleta y completa; **snu** cuando tenga superior no universitario incompleto y completo; y por último **su** con un nivel superior universitario incompleto y completo. Luego use el comando **graph pie** con la opción *plabel(\_all percent)* para obtener los valores en porcentajes.



# Capítulo 5

## Programación en STATA

### 5.1. Generando Números Seudo-Aleatorios

STATA incluye un conjunto de funciones para generar números seudo-aleatorios, el cual pueden seguir diversas funciones de distribución . Estas funciones comienzan con la letra **r** (de *random*).

Los generadores de números seudo-aleatorios usa determinandos mecanismos para producir largas cadenas de numeros que imitan las realizaciones de alguna función de distribución objetivo<sup>1</sup>.

Para mostrar como se generan números seudo-aleatorios, previamente indicaremos al STATA que solo trabajaremos con 1000 observaciones, por lo cual usaremos el comando **set obs**. Luego, cuando estos números seudo-aleatorios son generados, debemos establecer un valor específico como semilla con el comando **set seed**, tal que al correr varias veces el programa o Do-file obtengamos los mismos valores seudo-aleatorios. En este caso, generamos una variable seudo-aleatoria con distribución *uniforme*, *npormal*, *chi-cuadrado* y *t-student*.

A continuación mostramos la descripción de algunas funciones para generar

---

<sup>1</sup>Para una pequeña ilustración como STATA genera números seudo-aleatorios, ver a Cameron y Trivedi, Microeconomic usign STATA ( Capítulo 4)



números pseudo-aleatorios:<sup>2</sup>

Función de Distribución	function
Uniforme	<b>runiform()</b>
Normal	
media 0 y desviación estándar 1	<b>rnormal()</b>
media $m$ y desv. estándar 1	<b>rnormal(<math>k</math>)</b>
media $m$ y desv. estándar $s$	<b>rnormal(<math>m,s</math>)</b>
Normal Inversa	<b>invnormal()</b>
t-student	<b>rt(<math>gl</math>)</b>
Chi-Cuadrado	<b>rchi2(<math>gl</math>)</b>
Poisson	<b>rpoisson(<math>m</math>)</b>

Tabla 5.1: Funciones de Variables Aleatorias

```
. clear all

. set mem 200m

. set more off

. cd "D:\Econometria-Stata\programacion"

. *GENERANDO NÚMEROS PSEUDO-ALEATORIOS
. *-----

. set obs 1000
obs was 0, now 1000

. set seed 101010

. gen w=runiform()

. histogram w,saving(g1, replace) title("Distribución Uniforme")
(bin=29, start=.00182341, width=.03439541)
(file g1.gph saved)

. gen x=rnormal()

. histogram x, normal saving(g2, replace) title("Distribución Normal(0,1)")
(bin=29, start=-3.8545389, width=.22864615)
(file g2.gph saved)

. gen y=rchi2(5) //gl=5

. histogram y,normal saving(g3, replace) title("Distribución Chi-Cuadrado(5)")
(bin=29, start=.141712, width=.67029993)
(file g3.gph saved)
```

<sup>2</sup>Las funciones para generar números pseudo-aleatorios se pueden ver ejecutando el comando **help function** y elegir la opción Random-number functions *random-number functions*.

```
. gen z=rt(10)    //g1=10

. histogram z,normal saving(g4, replace) title("Distribución t-student(10)")
(bin=29, start=-4.4616423, width=.36922016)
(file g4.gph saved)

. graph combine g1.gph g2.gph g3.gph g4.gph
```

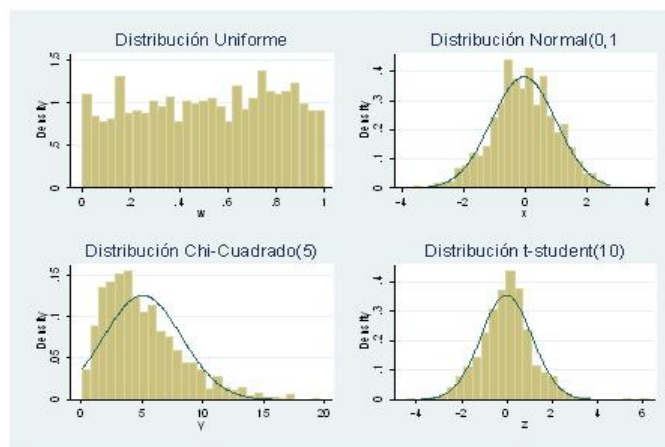


Figura 5.1: Gráficos - Variables Aletarias

## 5.2. Macros Local y Global

Una macro en STATA es un string que tiene un valor y un nombre, la cual sirve para reemplazar otros string. Esta macro puede contener cualquier combinación de caracteres alfanuméricos y caracteres. Existen dos tipos de macros en STATA conocidas como *local* y *global*. El contenido de la primera se define con el comando **local** y el segundo con el comando **global**.

Una *macro global* es accesible en los Do-file o a lo largo de una sesión en STATA. Una *macro local* puede ser accedida solo entre una sesión interactiva o un Do-file dado.

### 5.2.1. Macro Global

Estos son los más simples macros y son adecuados para muchos propósitos. Para acceder a este tipo de macro, ponemos el símbolo \$ inmediatamente antes del nombre de la macro. Por ejemplo, si queremos describir, generar un *codebook* y resumir los estadísticos de una lista de variables sin la necesidad de escribirlos en todas las líneas de comandos, entonces, usaremos una macro global **glist** que reemplace esta lista de variables.

```
. *MACROS
. *-----

. //Macro Global
. *-----

. global glist x y z

. sum $glist
```

Variable	Obs	Mean	Std. Dev.	Min	Max
x	1000	-.0366944	1.048498	-3.854539	2.776199
y	1000	5.061062	3.191537	.141712	19.58041
z	1000	-.0206937	1.128538	-4.461642	6.245742

### 5.2.2. Macro Local

Para acceder a esta macro, encerramos al nombre de la macro entre estas comillas especiales ( ‘ ’ )<sup>3</sup>. Como un ejemplo de macro local, consideramos una regresión de la variable mpg sobre diferentes regresores. Consideremos una macro local **llist**.

```
. //Macro Local
. *-----

. local llist x y z

. sum `llist`
```

Variable	Obs	Mean	Std. Dev.	Min	Max
x	1000	-.0366944	1.048498	-3.854539	2.776199
y	1000	5.061062	3.191537	.141712	19.58041
z	1000	-.0206937	1.128538	-4.461642	6.245742

<sup>3</sup>Estos símbolos donde se ubican en la parte superior de su derecha del teclado (el otro se ubica al costado del botón de la letra “p” y el primero símbolo se ubica en el botón del signo de interrogación “?”)

### 5.3. Comandos para Bucles

Los bucles proveen una forma de repetir el mismo comando muchas veces. STATA tiene tres constructores de bucles: **foreach**, **forvalues** y **while**.

Para la ilustración de estos comandos, lo usaremos para crear la suma de cuatro variables, donde cada variable se crea de una distribución uniforme. Existe muchas variaciones en la forma como uno puede realizar estos bucles.

```
. *BUCLES
. *-----

. gen x1var=runiform()
. gen x2var=runiform()
. gen x3var=runiform()
. gen x4var=runiform()

. *Nosotros deseamos la suma de las cuatro variables.
. gen suma= x1var+ x2var+x3var+ x4var

. summarize suma
```

Variable	Obs	Mean	Std. Dev.	Min	Max
suma	1000	2.026775	.5819928	.3077757	3.789982

A partir de esto presentamos diferentes formas para usar los bucles para calcular una suma progresiva de estas variables.

#### 5.3.1. El comando foreach

El comando **foreach** construye bucles para cada uno de los item de una lista, donde la lista puede ser nombre de variables (posiblemente dados en una macro) o una lista de números.

Comencemos por usar una lista de nombre de variables. En este caso la lista es *x1var*, *x2var*, *x3var* y *x4var*. Como vimos, la última variable creada fue *suma*, ahora crearemos otra que sea *suma1=0*, tal que los valores de esta sean todos iguales a cero. Usaremos esta misma idea para generar la suma de estas cuatro variable usando el comando **foreach**.

```

. // foreach
. *-----

. *I Forma

. gen suma1=0
. foreach var of varlist x1var x2var x3var x4var{
.     replace suma1=suma1 + `var'
. }
. //

(1000 real changes made)
(1000 real changes made)
(1000 real changes made)
(1000 real changes made)

. *II Forma

. replace suma1=0
(1000 real changes made)

. global xvar x1var x2var x3var x4var

. foreach var of varlist $xvar {
.     replace suma1=suma1 + `var'
. }

(1000 real changes made)
(1000 real changes made)
(1000 real changes made)
(1000 real changes made)

. *III Forma

. //Para esto se necesita que exista un orden entre
. //desde variables x1var hasta x4var

. replace suma1=0
(1000 real changes made)

. foreach var of varlist x1var-x4var {
.     replace suma1=suma1 + `var'
. }

(1000 real changes made)
(1000 real changes made)
(1000 real changes made)
(1000 real changes made)

```

El resultado es el mismo obtenido manualmente. La codificación en este bucle es un ejemplo de una programación, donde se coloca un corchete de apertura { al final de la primera línea de comando y un corchete de cierre } al final del programa. En este bucle, nos referimos a cada variable en la lista de variable llamada varlist a través de la macro local llamada **var**, así que es necesario el uso de las comillas

especiales para invocar a esta macro local. El nombre de la macro es opcional, pero la palabra *varlist* si es necesaria para indicarle al STATA que está trabajando con una lista de variable. Otros posibles listas que se podrían usar es *numlist*, *newlist*, *global* o *local* <sup>4</sup>.

### 5.3.2. El comando `forvalues`

El comando **forvalues** iter sobre valores consecutivos, En el siguiente código, nosotros usamos un índice **i** para que se una macro local **'i'**.

```
. // forvalues
. *-----

. gen suma2=0

. forvalues i=1/4 {
.     replace suma2= suma2 + x`i`var
. }
(1000 real changes made)
(1000 real changes made)
(1000 real changes made)
(1000 real changes made)

. summarize suma2
```

Variable	Obs	Mean	Std. Dev.	Min	Max
suma2	1000	2.026775	.5819928	.3077757	3.789982

Como vemos produce el mismo resultado. la elección del nombre **i** para la macro local fue arbitrario. En este caso específico donde el incremento es una unidad, uno puede usar otros incrementos. Por ejemplo, podemos escribir *forvalues i=1(2)11*, entonces el índice va de 1 hasta 11 en incrementos de 2 unidades.

### 5.3.3. El comando `while`

El comando **while** continua ejerciendo la operación ordenada hasta que una condición ya no sea cumplida. Este comando es utilizado cuando los comandos

---

<sup>4</sup>Para más detalle se recomienda ver `help foreach`.

foreach y forvalues no puedan ser utilizados. En el siguiente código, la macro local **i** se hace que inicie en el valor de 1 y luego aumenta hasta que  $i \leq 4$ .

```
. //while
. *-----

. gen suma3=0

. local i 1

. while `i' <=4 {
.     replace suma3= suma3 + x`i'var
.     local i= `i' + 1
. }

(1000 real changes made)
(1000 real changes made)
(1000 real changes made)
(1000 real changes made)

. summ suma3
```

Variable	Obs	Mean	Std. Dev.	Min	Max
suma3	1000	2.026775	.5819928	.3077757	3.789982

## 5.4. Escalares y Matrices

Los escalares pueden almacenar números o string, y las matrices pueden almacenar diferentes números o string como un vector.

### 5.4.1. Escalar

Los comandos que nos permite analizar variable (*describe*, *summarize*, *etc*) dan resultados como escalares numéricos. Podemos ver los contenido de un escalar usando el comando **display**, también podemos ver la lista de todos los escalares creados a través del comando **scalar list**.

```
. * Escalares y Matrices
. *-----

. //Escalar
. *-----
```

```
. scalar a = 2*3

. scalar list
      a =           6

. display "2 veces 3= " a
2 veces 3= 6
```

### 5.4.2. Matrices

STATA provee dos formas distintas para usar matrices, ambas almacenan tanto números o string en vectores. Una manera es a través de los comandos de STATA que tiene el prefijo **matrix**. El otro modo es usando el lenguaje de programación que incluye el STATA en esta versión llamada **MATA**. El siguiente código ilustra la definición de una matriz de tamaño  $2 \times 3$  (con el comando **matrix define**), la lista de la matriz (**matrix list**) y la extracción como un escalar de un elemento específico del elemento de una matriz.

```
. //Matrices
. *-----

. matrix define A = (1,2,3\4,5,6\7,8,9)
. matrix list A

A[2,3]
      c1  c2  c3
r1     1   2   3
r2     4   5   6
r2     7   8   9

. scalar a= A[2,3]
. display a
6
```

También es posible convertir las variables de una base de datos a una matriz agrupandolas, a través del comando **mkmat**. Es recomendable usar la opción **matrix()** donde se coloca el nombre de la matriz que se va a generar. Para una ilustración, agruparemos las primeras 100 observaciones de las variables aleatorias  $w$ ,  $x$ ,  $y$ ,  $z$  en una matriz llamada  $X$ , pero previamente podemos establecer una memoria máxima para crear una matriz, en nuestro caso estableceremos que la matriz puede ser de orden 1000 como máximo.

```
. set matsize 1000
```



Current memory allocation

settable	current value	description	memory usage (1M = 1024k)
set maxvar	5000	max. variables allowed	1.909M
set memory	200M	max. data space	200.000M
set matsize	1000	max. RHS vars in models	7.713M
			209.622M

```
. mkmat w x y z in 1/100, matrix(X)

. matrix list X
```

Lo interesante del entorno matricial del STATA, es que tiene implementado diversas funciones que son de gran utilidad como la Transpuesta, el Determinante, la Inversa, los Autovalores y Autovectores de una matriz, entre otros. A continuación se muestra un ejemplo de estas funciones:

```
. //Operaciones con Matrices

. scalar detA=det(A) //Determinante de A
. display "detA =" detA
detA =6.661e-16

. scalar trazaA=trace(A) //Traza de A
. display "trazaA =" trazaA
trazaA =15

. matrix define At=A' //Transpuesta de A
. matrix list At

At[3,3]
      r1  r2  r3
c1   1   4   7
c2   2   5   8
c3   3   6   9

. matrix define Ainv=inv(A) //Inversa de A
. matrix list Ainv

symmetric Ainv[3,3]
      r1          r2          r3
c1 -4.504e+15
c2  9.007e+15 -1.801e+16
c3 -4.504e+15  9.007e+15 -4.504e+15

. matrix define I5=I(5) //Identidad(5)
. matrix list I5
```

```

symmetric I5[5,5]
      c1 c2 c3 c4 c5
r1    1
r2    0 1
r3    0 0 1
r4    0 0 0 1
r5    0 0 0 0 1

. matrix define B=J(2,3,0)      //Matriz B de 2 filas y 3 columnas lleno de ceros
. matrix list B

B[2,3]
      c1 c2 c3
r1    0 0 0
r2    0 0 0

. matrix define d=vecdiag(A)    //Vector compuesto por los elementos de la diagonal de A
. matrix list d

d[1,3]
      c1 c2 c3
r1    1 5 9

. matrix define D=diag(d) //Matriz columna cuya diagonal principal es el vector d
. matrix list D

symmetric D[3,3]
      c1 c2 c3
c1    1
c2    0 5
c3    0 0 9

```

## 5.5. Usando los Resultados de los Comandos de STATA

### 5.5.1. Usando los Resultados con el Comando `r-class`

Los comandos del STATA que analizan pero que no estiman parámetros son comandos **r-class**. Todos los comandos *r-class* guardan su resultado en `r()`. Los contenidos de `r()` varían según el comando y se pueden observar tipeando **return list**. Como ejemplo, listamos los resultados almacenados después de usar un *summarize*:

```
. //Comando r-class
. *-----

. summ suma
```

Variable	Obs	Mean	Std. Dev.	Min	Max
suma	1000	2.026775	.5819928	.3077757	3.789982

```

. return list
scalars:
      r(N) = 1000
    r(sum_w) = 1000
    r(mean) = 2.026775246024132
    r(Var) = .3387156196380708
    r(sd) = .5819928003318176
    r(min) = .3077757060527802
    r(max) = 3.789982318878174
    r(sum) = 2026.775246024132

```

Hay ocho resultados almacenados separadamente escalares en el STATA con los nombres  $r(n)$ ,  $r(sum\_w)$ , ...,  $r(sum)$ . Otros resultados adicionales se mostrarán si usamos la opción **detail**.

El siguiente código calcula y muestra el rango de la data.

```
. summ suma
```

Variable	Obs	Mean	Std. Dev.	Min	Max
suma	1000	2.026775	.5819928	.3077757	3.789982

```

. scalar rango= r(max) - r(min)

. display "Sample range =" rango
Sample range =3.4822066

. scalar media=r(mean)

. scalar list
      media = 2.0267752
      rango = 3.4822066
      a = 6

```

Los resultados en  $r()$  desaparecen cuando otro comando **r-class** o **e-class** es ejecutado. Podemos también guardar el valor como un escalar.

### 5.5.2. Usando los Resultados con el Comando e-class

Los comandos de estimación se guardan como **e-class** (o clase de comando de estimación), tal como **regress**. Los resultados son guardados en **e()**, los contenidos se pueden ver tipeando **ereturn list**.

```
. //Comando e-class
. *-----

. regress y x z
```

Source	SS	df	MS		Number of obs =	1000
Model	27.9413623	2	13.9706811		F( 2, 997) =	1.37
Residual	10147.7801	997	10.1783151		Prob > F =	0.2539
					R-squared =	0.0027
					Adj R-squared =	0.0007
Total	10175.7215	999	10.1859074		Root MSE =	3.1903

	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
y					
x	-.1517576	.0962705	-1.58	0.115	-.3406737 .0371585
z	-.0463528	.0894426	-0.52	0.604	-.2218702 .1291646
_cons	5.054534	.1009668	50.06	0.000	4.856402 5.252666

```
. ereturn list

scalars:
      e(N) = 1000
    e(df_m) = 2
    e(df_r) = 997
      e(F) = 1.372592716977994
    e(r2) = .0027458851254827
    e(rmse) = 3.190347171151676
    e(mss) = 27.94136227917443
    e(rss) = 10147.78012725807
    e(r2_a) = .0007453753664566
    e(ll) = -2577.566020651634
    e(ll_0) = -2578.940851643387
    e(rank) = 3

macros:
    e(cmdline) : "regress y x z"
    e(title) : "Linear regression"
    e(marginsok) : "XB default"
    e(vce) : "ols"
    e(depvar) : "y"
    e(cmd) : "regress"
    e(properties) : "b V"
    e(predict) : "regres_p"
    e(model) : "ols"
    e(estat_cmd) : "regress_estat"

matrices:
      e(b) : 1 x 3
      e(V) : 3 x 3

functions:
      e(sample)
```

Los resultados numéricos en el análisis de regresión se han guardado como escalares. Por ejemplo, podemos usar los resultados para calcular el valor de  $R^2$ . La *suma de cuadrados del modelo* está guardado en **e(mss)** y la *suma de cuadrados de los residuos* en **e(rss)**.

```
. *Calculando el R-squared
. scalar r2=e(mss)/(e(rss)+e(mss))
. display "r-squared = " r2
r-squared = .00274589
```

El resultado es el mismo que arrojo la regresión original (0.00274589).

Los resultados de los parámetros y varianza están guardados como matrices. Aquí presentamos métodos para extraer escalares desde matrices y manipularlos. Específicamente, nosotros obtenemos el coeficiente MCO del precio desde la matriz  $e(b)$  de  $1 \times 3$ , el estimador de varianza desde la matriz  $e(V)$  de  $3 \times 3$ , y entonces formamos el estadístico de t-student para probar la significancia individual del modelo para la variable price.

```
. *Calculando el t-student para la variable price
. matrix b_est=e(b)
. scalar b_x=b_est[1,1]
. matrix V_est=e(V)
. scalar V_x=V_est[1,1]
. scalar t_x=b_x/sqrt(V_x)
. display " t-student para Ho: b_rpice=0 es " t_x
t-student para Ho: b_rpice=0 es -1.5763665
```

El resultado es el mismo que arrojo la regresión original, -1.5763665. Los resultados en  $e()$  desaparecen cuando otro comando **r-class** o **e-class** es ejecutado.

## 5.6. Ejercicio Propuesto

Resuelva los siguientes enunciados:

1. Cree una base de datos con 1000 observaciones y establezca la siguiente semilla 123456789 para generar números aleatorios.
2. Genere 3 variables aleatorias con cada una de las siguientes distribuciones: Uniforme, Beta con parámetros  $(a, b) = \{(1, 2), (4, 3), (\sqrt{3}, \sqrt{7})\}$ , Binomial con parámetros  $(n, p) = \{(3, 0.75), (1, 0.2), (1.5, 0.45)\}$ , Chi Cuadrado con 3, 6 y 10 grados de libertad, Gamma con parámetros  $(a, b) = \{(1, \sqrt{2}), (4, 8), (\sqrt{3}, 7)\}$ , Binomial Negativa con parámetros  $(n, p) = \{(5, 0.55), (11, 0.22), (2.5, 0.75)\}$ , Normal con parámetros  $(m, s) = \{(0, 1), (2, 1), (1.5, \sqrt{3})\}$ , Poisson con media 0.5, 0.75 y 1, T-Student con 10, 6 y 3 grados de libertad. *Nota: Se recomienda crear variables cuyos nombres presenten un componente en común, para luego usar comandos de bucles para hacer eficiente la programación más adelante.*
3. Guarde una lista de variables en una macro global para tipo de distribución. por ejemplo: *global macro normal var1 var2 var3*. Luego guarde en una nueva macro llamado *distribucion* todas las macros creadas anteriormente.
4. Realice una gráfico por tipo de distribución comparando las tres de variables. *Nota: Se recomienda usar comando de bucles.*
5. Con todas las variables creadas generar una matriz, con nombre **X**, que englobe a todas estas. *Nota: Puede usar el comando **mkmat** con las macros creadas.*
6. Se define la Matriz de Varianzas y Covarianzas en su forma matricial como:

$$VC = \left(\frac{1}{n}\right)X'M_0X$$

donde:

$$M_0 = I - \left(\frac{1}{n}\right)ii'$$

siendo  $I$  la matriz identidad de orden 100 (por el número de observaciones o filas de la matriz  $X$ ) y el vector  $i$  de orden  $100 \times 1$  cuyos valores son todos iguales a la unidad.

7. Calcule los Autovalores y Autovectores de esta Matriz de Varianzas y Covarianzas. ¿Qué relación guarda los autovalores con la traza y el determinante de la Matriz de Varianzas y Covarianzas?

# Capítulo 6

## Diseño Muestral

### 6.1. Muestra vs Censo

En la práctica, las investigaciones que podemos realizar se basan en análisis de datos a nivel muestral, y es muy difícil realizar con datos a nivel de poblacional. Antes de mencionar las razones , hay que tener en claro la diferencia entre ambos conceptos:

- **Censo** : Comprende un recuento completo de los elementos de la población.
- **Muestra** : Comprende un subgrupo de elementos de la población.

Es importante mencionar que las inferencias que unen las características de la *muestra* con los parámetros de la *población* se llaman procedimientos de estimación.

La diferencia entre ambos conceptos se muestra en el siguiente cuadro:



	<b>Muestra</b>	<b>Censo</b>
Presupuesto	<i>Reducido</i>	<i>Reducido</i>
Tiempo Disponible	<i>Breve</i>	<i>Prolongado</i>
Tamaño de la Población	<i>Pequeña</i>	<i>Numerosa</i>
Varianza en la Característica	<i>Baja</i>	<i>Alta</i>
Costos de los Errores de Muestreo	<i>Bajo</i>	<i>Alto</i>
Costos de los Errores de falta de Muestreo	<i>Alto</i>	<i>Bajo</i>
Naturaleza de la Medición	<i>Destruyiva</i>	<i>No Destruyiva</i>
Atención a Casos Individuales	<i>Si</i>	<i>No</i>

Tabla 6.1: Muestra vs. Censo

## 6.2. Diseño Muestral

El proceso de diseño de la muestra incluye cinco pasos, estos están estrechamente interrelacionados y son relevantes para todos los aspectos del proyecto de investigación de mercados, desde la definición del problema hasta la presentación de los resultados.

- Definir la población meta.
- Determinar el marco de la muestra.
- Seleccionar las técnicas de muestreo.
- Determinar el tamaño de la muestra.
- Ejecutar el proceso de muestreo.

A continuación se explicará brevemente cada punto:

### 1. Definición de la población meta.

La *población meta* es el conjunto de elementos u objetos que poseen la información que busca el investigador y sobre los que debe hacerse la inferencia, es decir, quien debe incluirse en la muestra y quien no. Las poblaciones meta deben definirse con precisión, ya que una definición inexacta de la población meta dará como resultado una investigación ineficaz en el mejor de los casos y engañosa en el peor.

La población meta debe definirse en términos de los elementos, las unidades de muestra, la extensión y el tiempo. Un *elemento* es el objeto sobre el cual o del cual se desea información. En la investigación con encuestas, por lo regular el elemento es el entrevistado. Una *unidad de muestra* es un elemento, o unidad que contiene el elemento, que esta disponible para su selección en alguna etapa del proceso de muestreo.

## 2. Determinación del marco de la muestra.

El *marco muestral* es una *representación de los elementos de la población meta*. Consiste en una lista o grupo de indicaciones para identificar la población meta. *Ejemplos:* directorio telefónico, el directorio de una asociación que lista las empresas en una industria, una lista de correo comprada a una organización comercial, el directorio de una ciudad o mapa. Si no se puede compilarse una lista, debe especificarse, por lo menos, algunas indicaciones para identificar la población meta como los procedimientos de marcar dígitos aleatorios en las encuestas por teléfono.

Con frecuencia a la hora de listar los elementos incluimos elementos que no pertenecen a la población u omitimos uno que sí pertenece a la población. En algunos casos este error es pequeño y se ignora. Pero otras veces es necesario solucionar este error y hay diversos caminos como redefinir la población en función del marco muestral, corregir los errores en el proceso de entrevistas o ponderar los datos del marco dándole mayor probabilidad a los que pertenecen a la muestra.

## 3. Selección de una técnica de muestreo.

La *selección de una técnica de muestreo* comprende varias decisiones de naturaleza amplia. El investigador debe decidir si utilizar una estrategia de muestra bayesiana o tradicional, realizar la muestra con o sin reemplazo y si emplea una muestra de probabilidad o no probabilidad.

Este punto será tratado con mas detalle en la siguiente sección.

## 4. Determinación del tamaño de la muestra.

El *tamaño de la muestra* se refiere al *número de elementos que se incluyen en el estudio*. La determinación del tamaño de la muestra es compleja y comprende varias consideraciones:

- La importancia de la muestra (mientras más importante se necesitará mayor precisión y una muestra más grande).
- La naturaleza de la investigación.
- El número de variables.
- La naturaleza del análisis.
- Los tamaños de la muestra utilizada en estudios anteriores.
- Las limitaciones de recursos.

### 5. Ejecución del proceso de muestreo.

La *ejecución del proceso de muestreo* requiere de una especificación detallada de la forma en que se pone en práctica las decisiones del diseño de la muestra respecto a la población, el marco de la muestra, la unidad de muestra, la técnica de muestreo y el tamaño de la muestra. Deben proporcionarse información detallada para todas las decisiones sobre el diseño de la muestra.

## 6.3. Técnicas de Muestreo

Estadísticamente se conoce como **muestreo** a la técnica de seleccionar una muestra a partir de una población. Se espera que las propiedades de dicha muestra sean extrapolables a la población, es decir, debe ser una *muestra representativa* de la población objetivo. Este proceso permite ahorrar recursos, y a la vez obtener resultados parecidos a los que se alcanzarían si se realizase un estudio de toda la población.

Existen dos métodos para seleccionar muestras de poblaciones:

- **El muestreo aleatorio:** Incorpora el azar como recurso en el proceso de selección. Cuando se cumple con la condición de que todos los elementos de la población tienen *alguna oportunidad de ser escogidos en la muestra*, si la probabilidad correspondiente a cada sujeto de la población es conocida de antemano, recibe el nombre de *muestreo probabilístico*.

- **El muestreo no aleatorio:** Una muestra seleccionada por muestreo de juicio puede basarse en la experiencia de alguien con la población. Algunas veces una muestra de juicio se usa como guía o muestra tentativa para decidir cómo tomar una muestra aleatoria más adelante.

A continuación se explicará brevemente cada punto:

### 1. Muestreo Probabilístico

Forman parte de este tipo de muestreo todos aquellos métodos para los que puede calcular la probabilidad de extracción de cualquiera de las muestras posibles. Este conjunto de técnicas de muestreo es el más aconsejable, aunque en ocasiones no es posible optar por él. En este caso se habla de muestras probabilísticas, pues no es en rigor correcto hablar de muestras representativas dado que, al no conocer las características de la población, no es posible tener certeza de que tal característica se haya conseguido.

#### a. Muestreo Aleatorio Simple (SRS)

En esta técnica, cada elemento de la población tiene una probabilidad de selección idéntica y conocida, se elige independientemente de cualquier otro. Lo mismo ocurre con cualquier muestra de tamaño  $n$  que se formule por medio de un proceso aleatorio.

Características positivas:

- Fácil de comprender.
- Resultados pueden proyectarse a la población meta.
- La mayoría de planteamientos de inferencia suponen que la muestra ha sido recopilada por este procedimiento.

Limitaciones

- Difícil construir un marco del cual se pueda extraer una muestra por muestreo aleatorio simple.

- Pueden resultar muestras muy grandes.
- Baja precisión (con respecto a las demás técnicas).
- Existe incertidumbre acerca de la representatividad de la muestra.

#### **b. Muestreo sistemático**

En este caso, primero se elige aleatoriamente, un punto inicial. Luego, en base a ese punto inicial se eligen en sucesión cada  $i$ -ésimo elemento. El intervalo  $i$  de la muestra se determina dividiendo el tamaño de la población por el de la muestra que se desea. Por ejemplo, si aleatoriamente se elige el número 33 y sabemos que la población consta de 10000 individuos y se requiere una muestra de 100; los elementos siguientes serán 133 ( $33+100$ ), 233 ( $133+100$ ), etc.

Cada elemento de la muestra tiene probabilidad idéntica y conocida pero sólo las muestras de tamaño  $n$  tienen esa propiedad. Muestras de un tamaño distinto tienen una probabilidad de cero de ser elegidas. Una nota importante es que este tipo de muestreo es útil y representativo cuando los elementos presentan un orden que se relaciona con la característica de interés. Además, resulta ser menos costoso pues la selección aleatoria se realiza solo una vez (al principio).

#### **c. Muestreo Estratificado**

Una población se divide en subgrupos (estratos) y se selecciona una muestra de cada estrato. Hay que notar que los estratos deben ser lo más excluyentes posibles entre ellos; no obstante, dentro de un estrato, se requiere que la población sea bastante homogénea. Las variables que se utilizan para dividir a la población se llaman variables de estratificación, deben estar bastante relacionadas con la característica de interés y normalmente se emplea solo una.

Dentro de este tipo de muestreo tenemos otras dos categorías: Muestreo proporcionado: el tamaño de la muestra de cada estrato es proporcional al tamaño relativo de ese estrato en la población. Muestreo desproporcionado: el tamaño del estrato es proporcional al tamaño relativo del estrato y a la desviación estándar entre todos los elementos del mismo. Para utilizarlo se requiere que se tenga alguna información sobre la distribución de la característica de interés. Esta resulta una técnica de empleo bastante usada pues la muestra resulta ser representativa y además, el procedimiento es sencillo.

#### **d. Muestreo por Conglomerados**

Para utilizar esta técnica se siguen dos pasos. En primer lugar, se divide a la población objetivo en subpoblaciones mutuamente excluyentes y colectivamente exhaustivas (de modo que los elementos de las subpoblaciones sean homogéneos) que se denominarán grupos. En segundo lugar, se escogen aleatoriamente algunos grupos de forma aleatoria y se concentran los esfuerzos en estos, descartándose los no elegidos.

Una muestra de grupo también se puede realizar en más de dos etapas (muestra de etapas múltiples). La diferencia con el muestreo estratificado reside que en este caso se extrae una muestra de grupos para la selección posterior y no se seleccionan todas las subpoblaciones. Una forma particular del muestreo de grupos es el muestreo de áreas. En esta técnica, los grupos se refieren a áreas geográficas, la lógica es la misma que el muestreo de grupos y también puede realizarse en dos o más etapas.

### **2. Muestreo No Probabilístico**

#### **a. Muestreo de Juicio**

Aquél para el que no puede calcularse la probabilidad de extracción de una determinada muestra. Se busca seleccionar a individuos que se juzga de antemano tienen un conocimiento profundo del tema bajo estudio, por lo tanto, se considera que la información aportada por esas personas es vital para la toma de decisiones.

#### **b. Muestreo por cuotas**

Es la técnica más difundida sobre todo en estudios de mercado y sondeos de opinión. En primer lugar es necesario dividir la población de referencia en varios estratos definidos por algunas variables de distribución conocida (como el género o la edad). Posteriormente se calcula el peso proporcional de cada estrato, es decir, la parte proporcional de población que representan. Finalmente se multiplica cada peso por el tamaño de  $n$  de la muestra para determinar la cuota precisa en cada estrato. Se diferencia del muestreo estratificado en que una vez determinada la cuota, el investigador es libre de elegir a los sujetos de la muestra dentro de cada estrato.

#### **c. Muestreo de bola de nieve**

Indicado para estudios de poblaciones clandestinas, minoritarias o muy dispersas pero en contacto entre sí. Consiste en identificar sujetos que se incluirán en la muestra a partir de los propios entrevistados. Partiendo de una pequeña cantidad de individuos que cumplen los requisitos necesarios estos sirven como localizadores de otros con características análogas.

#### **d. Muestreo subjetivo por decisión razonada**

En este caso las unidades de la muestra se eligen en función de algunas de sus características de manera racional y no casual. Una variante de esta técnica es el muestreo compensado o equilibrado, en el que se seleccionan las unidades de tal forma que la media de la muestra para determinadas variables se acerque a la media de la población.

## **6.4. La Encuesta Nacional de Hogares (ENAHO)**

La **Encuesta Nacional de Hogares (ENAHO)**, es la investigación que permite al Instituto Nacional de Estadística e Informática (INEI) desde el año 1995, efectuar el seguimiento de los indicadores sobre las condiciones de vida.

#### **a. Objetivos**

- Generar indicadores mensuales, que permitan conocer la evolución de la pobreza, del bienestar y de las condiciones de vida de los hogares.
- Efectuar diagnósticos (mensuales) sobre las condiciones de vida y pobreza de la población.
- Medir el alcance de los programas sociales en la mejora de las condiciones de vida de la población.
- Servir de fuente de información a instituciones públicas y privadas, así como a investigadores.
- Permitir la comparabilidad con investigaciones similares, en relación a las variables investigadas.

### **b. Cobertura**

La encuesta se realiza en el ámbito nacional, en el área urbana y rural, en los 24 departamentos del país y en la Provincia Constitucional del Callao.

### **c. Población en Estudio**

La población de estudio está definida como el conjunto de todas las viviendas particulares y sus ocupantes residentes del área urbana y rural del país.

Por no ser parte de la población de estudio, se excluye a los miembros de las fuerzas armadas que viven en cuarteles, campamentos, barcos, y otros. También se excluye a las personas que residen en viviendas colectivas (hoteles, hospitales, asilos y claustros religiosos, cárceles, etc.).

### **d. Diseño y Marco Muestral**

#### **Marco Muestral**

El marco muestral para la selección de la muestra lo constituye la información estadística proveniente de los Censos de Población y Vivienda y material cartográfico actualizado para tal fin.

#### **Unidad de Muestreo**

##### **En el Área Urbana**

- La *Unidad Primaria de Muestreo (UPM)* es el centro poblado urbano con 2 mil y más habitantes.
- La *Unidad Secundaria de Muestreo (USM)* es el conglomerado que tiene en promedio 120 viviendas particulares.
- La Unidad Terciaria de Muestreo (UTM) es la vivienda particular.

##### **En el Área Rural**

- La Unidad Primaria de Muestreo (UPM) es de 2 tipos:



- El centro poblado urbano con 500 a menos de 2 mil habitantes.
- El Área de Empadronamiento Rural (AER) el cual tiene en promedio 100 viviendas particulares.
- La Unidad Secundaria de Muestreo (USM) es de 2 tipos:
  - El conglomerado que tiene en promedio 120 viviendas particulares.
  - La vivienda particular
- La Unidad Terciaria de Muestreo (UTM) es la vivienda particular.

### Tipo de Muestra

La muestra es del tipo probabilística, de áreas, estratificada, multietápica e independiente en cada departamento de estudio.

A fin de medir los cambios en el comportamiento de algunas características de la población, se ha implementado desde la ENAHO 2008 una muestra de viviendas tipo panel, en la cual viviendas encuestadas son nuevamente investigadas cada año.

En la muestra no panel se visitan cada año los mismos conglomerados en el mismo mes de encuesta pero se seleccionan distintas viviendas.

El nivel de confianza de los resultados muestrales, es del 95

### Características de la Encuesta

- **Método de Entrevista:** Directa.
- **Tipo de Encuesta:** Encuesta de Derecho.
- **Personal de Campo:** Coordinadores Departamentales, Supervisores y Encuestadoras.
- **Carga de Trabajo por día:** 1.5 viviendas.

### Factores de Expansión

En las encuestas por muestreo, las observaciones son seleccionadas mediante un proceso aleatorio, donde cada observación puede tener una probabilidad de selección diferente. La ponderación (o peso) de una observación (hogar, por ejemplo) es igual a la inversa de la probabilidad de pertenecer a la muestra.

La metodología de estimación para procesar los datos de la ENAHO, involucra el uso de un peso o factor de expansión para cada registro que será multiplicado por todos los datos que conforman el registro correspondiente.

El factor final para cada registro tiene dos componentes:

- El factor básico de expansión y
- Los factores de ajuste por la no entrevista.

El factor básico de expansión para cada hogar muestral es determinado por el diseño de la muestra. Equivale al inverso de su probabilidad final de selección, el mismo que es el producto de las probabilidades de selección en cada etapa.

El diseño de la muestra de la ENAHO, involucra hasta 3 etapas de muestreo donde las unidades son seleccionadas con probabilidades proporcionales al tamaño (ppt) excepto la última etapa. En la última etapa se selecciona un número de viviendas para cada conglomerado teniendo en cuenta un intervalo de selección.

Por consiguiente, los factores de expansión básicos para la ENAHO 2010 serán ajustados teniendo en cuenta las proyecciones de población por grupos de edad y sexo para cada mes de encuesta y niveles de inferencia propuestos en el diseño de la muestra.

Cabe mencionar que se podrán obtener estimaciones para otros niveles de desagregación y su precisión o confiabilidad estadística dependerá fundamentalmente del número de casos u observaciones contenidas en la base de datos.

## 6.5. Aplicación - ENAHO

Una de las ventajas que ofrece el Stata para el análisis de Encuestas como la ENAHO, con Diseño muestral complejo, es que permite calcular los estimadores teniendo en cuenta el diseño muestral de la misma (diferente al muestreo simple al azar). Además, Stata proporciona **estadísticos** con los cuales se puede **evaluar la confiabilidad** del resultado en forma simultánea a su estimación. De esta manera el usuario está en la capacidad de interpretar y utilizar adecuadamente cada estimación proveniente de la encuesta.

Los principales elementos que se deben tener en cuenta en el trabajo con datos de encuestas por muestreo son:

- **Ponderación:** En las encuestas por muestreo, las observaciones son seleccionadas mediante un proceso aleatorio, donde cada observación puede tener una probabilidad de selección diferente. La ponderación (o peso) de una observación (hogar, por ejemplo) es igual a la inversa de la probabilidad de pertenecer a la muestra. Es usual que luego del trabajo de campo se realicen ajustes sobre esta ponderación, debido, por ejemplo, al efecto de la “No-Respuesta”. Un peso  $w_j$  de una observación  $j$  significa que la observación  $j$  representa a  $w_j$  elementos de la población. Si no se toman en cuenta las ponderaciones, las estimaciones que se obtengan estarán sesgadas.
- **Conglomerados o cluster:** Algunas veces se utiliza el muestreo por conglomerados, es decir las observaciones son muestreadas en grupos o “clusters”, por ejemplo, provincias dentro de departamentos, distritos dentro de provincias y finalmente viviendas dentro de los distritos seleccionados, que son el objetivo final del muestreo. Todas las observaciones de un mismo cluster no son independientes entre si, si no se toma en cuenta este hecho, *los errores estándar que se obtengan serán menores a los verdaderos*.
- **Estratos:** En algunos casos, también se emplea el muestreo estratificado, donde diferentes grupos de observaciones o estratos, son muestreados en forma independiente. Al igual que el caso anterior, si no se toma en cuenta este hecho, se obtendrán *sub estimaciones de los errores estándar verdaderos*.

Para ilustrar este caso, trataremos de modelar dos ecuaciones importantes en el contexto del mercado laboral: La ecuación de participación laboral y la ecuación de

salarios. Para este fin, utilizaremos la base de la ENAHO del año 2010 correspondiente a los **modulos 100** (características de la vivienda); **modulo 200** (miembros del hogar); **modulo 300** (Educación) y **modulo 500** (Empleo).

En primer lugar, cargaremos las bases de cada uno de los modulos y seleccionaremos las variables de nuestro interés. Además de las variables que nos permitan identificar a cada una de las observaciones (como: el año (**aÑo**), el mes (**mes**), conglomerado (**conglome**), vivienda (**vivienda**), hogar (**hogar**), ubigeo (**ubigeo**), dominio (**dominio**), estrato (**estrato**) y código de la persona (**codperso**)), el cual nos van a servir como llaves para poder fusionar todas las bases de datos con que estamos trabajando. El código de la vivienda (**codviv**) estará compuesto por la concatenación de las variables año, mes, conglomerado, vivienda, hogar, ubigeo, dominio y estrato; mientras que, el código de la persona estará determinado por las mismas variables del código de la vivienda más el código de la persona.

Del modulo 100 consideraremos las variables que indicas los servicios básicos que cuenta la vivienda (como: telefono (**p1141**), celular (**p1142**) e internet (**p1144**)).

```
. *****
. * APLICACIÓN - ENAHO 2010 *
. *****

. clear all
. set mem 300m
Current memory allocation

```

settable	current value	description	memory usage (1M = 1024k)
set maxvar	5000	max. variables allowed	1.909M
set memory	300M	max. data space	300.000M
set matsize	400	max. RHS vars in models	1.254M
			303.163M

```
. set more off

. cd "D:\Econometria-Stata\aplicacion-enafo"
D:\Econometria-Stata\aplicacion-enafo

. * MODULO 100 - CARACTERÍSTICAS DE LA VIVIENDA
. *****

. use enafo01-2010-100.dta, clear

. *mantenemos las variables de nuestro interes
. keep año mes conglome vivienda hogar ubigeo dominio estrato ///
> p1141 p1142 p1144

. *creamos las variables respectivas
. rename p1141 telefono
. rename p1142 celular
```

```
. rename p1144 internet

. *creamos el identificador de vivienda
. egen codviv= concat(año mes conglome vivienda hogar ubigeo dominio estrato)
. sort codviv

. *guardamos la base modificada del modulo 100
. save modulo100.dta, replace
file modulo100.dta saved
```

Del modulo 200 consideraremos las variables que reflejan algunas características de los individuos importantes, como: el número de miembros en el núcleo familiar (**p203a**), el parentesco con el jefe del hogar (**p203b**), el sexo (**p207**), la edad en años (**p208a**) y el estado civil y conyugal (**p209**). De la variable **p209** crearemos una dummy cuyo valor 1 será si el individuo está casado o es conviviente y 0 en otro caso; de la variable **p207** una dummy donde 1 será si el encuestado es hombre y 0 si es mujer; de la variable **p203b** una dummy donde 1 si es jefe del hogar y 0 si no lo es. De la variable **p203a** calcularemos el número de miembros de la familia, conjuntamente con la variable **p203b** y **p208a** identificaremos a los miembros de la familia que son hijos en la familia y que presentan edades entre 0 a 5 años y 6 a 12 años, para este propósito se creó una variable que identifique los miembros del núcleo familiar con la variable **codnucfam** que es la concatenación de las variables año, mes conglomerado, vivienda, hogar y p203a.

```
. * MODULO 200 - CARACTERÍSTICAS DE LOS MIEMBROS DEL HOGAR
. *****

. use enaho01-2010-200.dta, clear

. *mantenemos las variables de nuestro interes
. keep año mes conglome vivienda hogar ubigeo dominio estrato codperso ///
p203a p203b p207 p208a p209

. *creamos las variables respctivas

. //jefe del hogar
. gen jhog=1 if p203b==1
(62720 missing values generated)
. replace jhog=0 if p203b!=1
(62720 real changes made)

. //genero
. gen genero=1 if p207==1
(50333 missing values generated)
. replace genero=0 if p207==2
(45965 real changes made)

. //Número de hijos

. *generamos el código del núcleo familiar
. egen codnucfam=concat(año mes conglome vivienda hogar p203a)
```

```

.      *-Número de hijos por núcleo familiar
.      gen aux4=1 if p203b==3
(53714 missing values generated)
.      replace aux4=0 if aux4==.
(53714 real changes made)
.      bysort codnucfam : egen numhij_nf=total(aux4)
.      la var numhij_nf "Número Total de hijos por Nucleo Familiar"

.      *-Número de hijos de 0 a 5 años
.      gen aux5=1 if p208a<=5
(85044 missing values generated)
.      replace aux5=0 if aux5==.
(85044 real changes made)
.      bysort codnucfam: egen numhij05_nf=total(aux5)
.      la var numhij05_nf "Numero de Hijos entre 0-5 años por Nucleo Familiar"

.      *-Número de hijos de 6 a 12 años
.      gen aux6=1 if p208a>=6 & p208a<=12
(82293 missing values generated)
.      replace aux6=0 if aux6==.
(82293 real changes made)
.      bysort codnucfam: egen numhij612_nf=total(aux6)
.      la var numhij612_nf "Numero de Hijos entre 6-12 años por Nucleo Famil
> iar"

. //Tamaño del Núcleo Familiar
. gen aux7=1
. bysort codnucfam: egen tamnf=total(aux7)
. la var tamnf "Tamaño del Núcleo Familiar"

. //edad
. rename p208a edad

. //estado civil (casado o soltero)
. gen casado=1 if (p209==1 | p209==2)
(60473 missing values generated)
. replace casado=0 if casado==.
(60473 real changes made)

. *creamos el identificador de vivienda
. egen codviv= concat(año mes conglome vivienda hogar ubigeo dominio estrato)
. sort codviv

. *creamos el identificador de persona
. egen codper= concat(año mes conglome vivienda hogar ubigeo dominio estrato co
> dperso)
. sort codper

. *mantenemos las nuevas variables de nuestro interes
. keep año mes conglome vivienda hogar ubigeo dominio estrato codperso ///
codviv codper jhog genero edad casado numhij_nf numhij05_nf numhij612_nf tamnf

. *guardamos la base modificada del modulo 200
. save modulo200.dta, replace
file modulo200.dta saved

```

Del modulo 300 seleccionaremos variables relacionado a la educación del encuestado, como: el nivel educativo aprobado (p301a), el último año de estudio que aprobó (p301b) y último grado de estudios que aprobó. De estas variables calcularemos una variable proxy de los años de educación donde el criterio de partida será: Si *no tiene nivel educativo* (0 años de educación), si tiene *educación inicial y nivel primaria* (se cuenta los años que ha estudiado o aprobado), si tiene *nivel secundaria completa o incompleta* (se cuenta los años que ha estudiado o aprobado más 6 años del nivel de primaria), si tiene *nivel de educación superior universitaria o no universitaria* (se cuenta los años estudiados o aprobados más 11 años entre educación primaria y secundaria) y por último, si tiene *nivel post-grado* (se cuenta los años estudiados o aprobados más 16 años entre educación primaria, secundaria y superior universitario).

```
. * MODULO 300 - EDUCACIÓN
. *****

. use enaho01a-2010-300.dta, clear

. *mantenemos las variables de nuestro interes
. keep año mes conglome vivienda hogar ubigeo dominio estrato codperso ///
  p301a p301b p301c

. *creamos las variables respctivas

. //Años de Escolaridad
. gen yeareduca=0 if p301a==1 //sin nivel
(75789 missing values generated)
. replace yeareduca=p301b+0 if p301a==2 //educación inicial
(3732 real changes made)
. replace yeareduca=p301b+0 if p301a==3 //primaria incompleta
(19752 real changes made)
. replace yeareduca=p301b+0 if p301a==4 //primaria completa
(9927 real changes made)
. replace yeareduca=p301c+0 if p301a==3 & p301b==0 //primaria incompleta
(13136 real changes made)
. replace yeareduca=p301c+0 if p301a==4 & p301b==0 //primaria completa
(5420 real changes made)
. replace yeareduca=p301b+6 if p301a==5 //secundaria incompleta
(14428 real changes made)
. replace yeareduca=p301b+6 if p301a==6 //secundaria completa
(13675 real changes made)
. replace yeareduca=p301b+11 if p301a==7 //Superior No Universitaria Incompleta
(2536 real changes made)
. replace yeareduca=p301b+11 if p301a==8 //Superior No Universitaria Completa
(4542 real changes made)
. replace yeareduca=p301b+11 if p301a==9 //Superior Universitaria Incompleta
(3393 real changes made)
. replace yeareduca=p301b+11 if p301a==10 //Superior Universitaria Completa
(3109 real changes made)
. replace yeareduca=p301b+16 if p301a==11 //Postgrado
(606 real changes made)
```

```

. *creamos el identificador de vivienda
. egen codviv= concat(año mes conglome vivienda hogar ubigeo dominio estrato)
. sort codviv

. *creamos el identificador de persona
. egen codper= concat(año mes conglome vivienda hogar ubigeo dominio estrato codperso)
. sort codper

. *mantenemos las nuevas variables de nuestro interes
. keep año mes conglome vivienda hogar ubigeo dominio estrato codperso ///
  codviv codper yeareduca

. *guardamos la base modificada del modulo 300
. save modulo300.dta, replace
file modulo300.dta saved

```

Del modulo 500 seleccionaremos variables relacionado a la situación laboral del encuestado, como: el número de horas trabajadas durante la semana de referencia de la encuesta (**i513t**), condición laboral del encuestado (**ocu500**), el ingreso anual imputado obtenido en su ocupación principal (i524a1) y secundaria (i538a1), por último, el factor de expansión para el modulo de empleo **fac500a7**. De la variable **i513t** calcularemos las horas trabajadas en el mes multiplicandola por 4. Crearemos una variable dummy a partir de **ocu500** cuyo valor 1 será si está laboran y 0 sino lo está. Además, calcularemos el ingreso laboral mensual que se deriva de la suma del ingreso principal y secundaria entre 12.

```

. * MODULO 500 - EMPLEO
. *****

. use enaho01a-2010-500.dta, clear

. *mantenemos las variables de nuestro interes
. keep año mes conglome vivienda hogar ubigeo dominio estrato codperso ///
  i513t ocu500 i524a1 i538a1 fac500a7

. *creamos las variables respctivas

. //condición laboral
. gen ocupado=1 if ocu500==1
(17826 missing values generated)
. replace ocupado=0 if ocu500!=1
(17826 real changes made)

. //horas trabajadas mensuales
. gen hrtrab=4*i513t
(14218 missing values generated)

. //ingreso laboral total mensual
. replace i524a1=0 if i524a1==.
(46167 real changes made)
. replace i538a1=0 if i538a1==.
(61507 real changes made)

```



```
. gen inglab=(i524a1+i538a1)/12
. replace inglab=ln(inglab)
(63810 real changes made, 44582 to missing)

. *creamos el identificador de vivienda
. egen codviv= concat(año mes conglome vivienda hogar ubigeo dominio estrato)
. sort codviv

. *creamos el identificador de persona
. egen codper= concat(año mes conglome vivienda hogar ubigeo dominio estrato codperso)
. sort codper

. *mantenemos las nuevas variables de nuestro interes
. keep año mes conglome vivienda hogar ubigeo dominio estrato codperso ///
  codviv codper ocupado hrtrab inglab fac500a7

. *guardamos la base modificada del modulo 500
. save modulo500.dta, replace
file modulo500.dta saved
```

Una vez que seleccionamos las variable de nuestro interés en cada uno de los modulos, procedemos a fusionar toda la información en una única base de datos al cual denominaremos *base2010.dta*.

```
. * FUSIÓN DE BASE DE DATOS
. *****

. //fusión base 500 y 300
. count
63810

. merge codper using modulo300.dta
(note: you are using old merge syntax; see [R] merge for new syntax)
(label estrato already defined)
(label dominio already defined)
. tab _merge
```

_merge	Freq.	Percent	Cum.
2	19,563	23.46	23.46
3	63,810	76.54	100.00
Total	83,373	100.00	

```
. keep if _merge==3
(19563 observations deleted)
. drop _merge
. sort codper

. save modulo2010.dta, replace
file modulo2010.dta saved

. //fusión base 500, 300 y 200
. count
63810

. merge codper using modulo200.dta
(note: you are using old merge syntax; see [R] merge for new syntax)
(label estrato already defined)
```

```
(label dominio already defined)
. tab _merge
      _merge |      Freq.   Percent   Cum.
-----+-----+-----+-----
           2 |    31,339    32.94    32.94
           3 |    63,810    67.06   100.00
-----+-----+-----+-----
        Total |    95,149   100.00
. keep if _merge==3
(31339 observations deleted)
. drop _merge
. sort codviv

. save modulo2010.dta, replace
file modulo2010.dta saved

. //fusión base 500, 300, 200 y 100
. count
63810
. joinby codviv using modulo100.dta, unmatched(both)
. tab _merge
      _merge |      Freq.   Percent   Cum.
-----+-----+-----+-----
    only in using data |     5,680     8.17     8.17
both in master and using data |    63,810    91.83   100.00
-----+-----+-----+-----
        Total |    69,490   100.00
. keep if _merge==3
(5680 observations deleted)
. drop _merge
. sort codviv

. save modulo2010.dta, replace
file modulo2010.dta saved
```

Luego, calcularemos algunas variables que podrían ser de ayuda, como es la experiencia laboral potencial (**exper**) y su cuadrado (**exper2**), el cual se define como el valor mínimo de la experiencia obtenida entre la diferencia de la edad actual y los años de educación menos 5 años, y la experiencia obtenida entre la diferencia entre la edad actual y 14 años que es la edad que se considera a una persona apta para participar en el mercado laboral. Además, de las variables geográficas como el ámbito geográfico (Lima Metropolitano, Resto Urbano y Rural), el área de residencia (Urbano y Rural) y las regiones del país (Costa, Sierra y Selva) que se derivan de las variables estrato y dominio.

```
. *generamos más variables de nuestro interes

. //EXPERIENCIA POTENCIAL

. *Experiencia Laboral Potencial

. *Aproximada por la Edad y Educación (Exp1)
```

```

. gen exper_a=edad-yeareduca-5
(64 missing values generated)
. replace exper_a=0 if exper<0
(47 real changes made)

. *Aproximada por la Edad a trabajar (Exp2)
. gen exper_b=edad-14

. *Experiencia Potencial= min(Exp1,Exp2)
. gen exper = min(exper_a,exper_b)
. gen exper2=exper^2

. la var exper "Experiencia Potencial"
. la var exper2 "Experiencia Potencial al Cuadrado"

. drop exper_a exper_b

. //CREACIÓN DE VARIABLES GEOGRÁFICAS

. *a. VARIABLE ÁMBITO GEOGRÁFICO
. gen ambito_geografico=1 if dominio==8
(55754 missing values generated)
. replace ambito_geografico=2 if (dominio>=1 & dominio<=7) & (estrato>=1 & estrato<=5)
(31290 real changes made)
. replace ambito_geografico=3 if (dominio>=1 & dominio<=7) & (estrato>=6 & estrato<=8)
(24464 real changes made)

. la var ambito_geografico "Ambito Geografico"
. la de amb_geo 1 "Lima Metropolitana" 2 "Resto Urbano" 3 "Rural"
. la val ambito_geografico amb_geo

. *b. VARIABLE ÁREA DE RESIDENCIA
. recode ambito_geografico (1=0) (2=0) (3=1) , gen(rural)
(63810 differences between ambito_geografico and rural)

. la var rural "=1 si es Rural,=0 si es Urbano"
. la de amb_res 0 "Urbana" 1 "Rural"
. la val rural amb_res

. tab rural

```

	Freq.	Percent	Cum.
Urbana	39,346	61.66	61.66
Rural	24,464	38.34	100.00
Total	63,810	100.00	

```

. *c. REGIONES: Costa, Sierra y Selva
. gen region=1 if dominio==1 | dominio==2 | dominio==3 | dominio==8
(38952 missing values generated)
. replace region=2 if dominio==4 | dominio==5 | dominio==6
(25059 real changes made)
. replace region=3 if dominio==7
(13893 real changes made)

```

```
. la var region "Región: Costa, Sierra y Selva"
. la de region 1 "Costa" 2 "Sierra" 3 "Selva"
. la val region region
```

```
. tab region
```

Región: Costa, Sierra y Selva	Freq.	Percent	Cum.
Costa	24,858	38.96	38.96
Sierra	25,059	39.27	78.23
Selva	13,893	21.77	100.00
Total	63,810	100.00	

Después de armar la base de datos, tenemos que especificar al STATA para que incorpore el Diseño Muestral (ponderaciones, conglomerados y estratos) antes de ejecutar las estimaciones. Es decir, Stata utiliza las fórmulas de estimación de estadísticos propias de cada tipo de muestreo. Todos los comandos para el análisis de datos provenientes de encuestas comienzan con las letras **svy**.

En el caso de la Enaho es necesario especificar las variables que contienen las ponderaciones (**fac500a7**), los conglomerados (**conglome**) y los estratos (**estrato**), antes de obtener cualquier estimación.

```
. *COMANDO SVY
. *****

. // Comando: svyset

. svyset [pweight= fac500a7], strata(estrato) psu(conglome)

    pweight: fac500a7
      VCE: linearized
Single unit: missing
  Strata 1: estrato
    SU 1: conglome
    FPC 1: <zero>

. // Comando: svydes

. svydes

Survey: Describing stage 1 sampling units
    pweight: fac500a7
      VCE: linearized
Single unit: missing
  Strata 1: estrato
    SU 1: conglome
    FPC 1: <zero>
```

Stratum	#Units	#Obs	#Obs per Unit		
			min	mean	max
1	633	10349	2	16.3	36
2	706	12197	3	17.3	34
3	284	4744	2	16.7	33
4	316	5322	5	16.8	43
5	366	5879	6	16.1	32
6	181	3698	9	20.4	34
7	691	15032	3	21.8	42
8	229	4884	9	21.3	36
8	3406	62105	2	18.2	43
			1705 = #Obs with missing values in the survey characteristics		
			63810		

### Estimación de Promedios: SVYMEAN

- Se utiliza para calcular promedio de variables cuantitativas.
- Por defecto presenta el promedio estimado, el error estándar, el intervalo de 95 % de confianza y el efecto de diseño de esta estimación.
- Se pueden utilizar las opciones *if* y *over*.

```
. // Comando: svymean
. svy: mean inglab
(running mean on estimation sample)
Survey: Mean estimation
Number of strata =      8      Number of obs   =   18577
Number of PSUs   =   3320      Population size =  7332557
                                   Design df      =    3312
```

	Linearized			
	Mean	Std. Err.	[95% Conf. Interval]	
inglab	6.379518	.0129515	6.354124	6.404912

```
. svy: mean inglab if genero==0
(running mean on estimation sample)
Survey: Mean estimation
Number of strata =      8      Number of obs   =    6732
Number of PSUs   =   2678      Population size =  2751055
                                   Design df      =    2670
```

	Linearized			
	Mean	Std. Err.	[95% Conf. Interval]	
inglab	6.174006	.0194611	6.135846	6.212167



```
_prop_1: ocupado = 0
_prop_2: ocupado = 1
0: genero = 0
1: genero = 1
```

Over	Linearized			
	Proportion	Std. Err.	[95% Conf. Interval]	
_prop_1	0	.3742463	.0037674	.3668598 .3816329
	1	.2031456	.0032725	.1967293 .209562
_prop_2	0	.6257537	.0037674	.6183671 .6331402
	1	.7968544	.0032725	.790438 .8032707

```
. svy: proportion ocupado if casad==1, over(genero)
(running proportion on estimation sample)
Survey: Proportion estimation
Number of strata =      8      Number of obs   =    33228
Number of PSUs   =   3392      Population size = 10881895
                                   Design df      =    3384

_prop_1: ocupado = 0
_prop_2: ocupado = 1
0: genero = 0
1: genero = 1
```

Over	Linearized			
	Proportion	Std. Err.	[95% Conf. Interval]	
_prop_1	0	.3386658	.0051795	.3285105 .3488211
	1	.0845736	.0028768	.0789331 .090214
_prop_2	0	.6613342	.0051795	.6511789 .6714895
	1	.9154264	.0028768	.909786 .9210669

### Cruce de dos variables : SVYTAB

- Produce una tabla de dos entradas con la proporción de la muestra que pertenece a cada celda (cruce de variables), respecto al total de la muestra.
- Para modificar el contenido de la tabla se deben especificar los estadísticos después de una coma.
- En caso de que se desee estimar las proporciones respecto a filas o columnas, basta con indicar *row* o *column* después de la coma.
- Se puede utilizar la opción *if*.

```
. // Comando: svytab
```

```
. svy: tab ocupado  
(running tabulate on estimation sample)
```

```
Number of strata =      8  
Number of PSUs   =     3406
```

```
Number of obs    =     62105  
Population size  =    21223493  
Design df       =       3398
```

ocupado	proportions
0	.289
1	.711
Total	1

Key: proportions = cell proportions

```
. svy: tab estrato ocupado  
(running tabulate on estimation sample)
```

```
Number of strata =      8  
Number of PSUs   =     3406
```

```
Number of obs    =     62105  
Population size  =    21223493  
Design df       =       3398
```

estrato	ocupado		
	0	1	Total
mayor de 20,00	.1284	.2567	.3852
de 10,00	.0514	.1113	.1627
de 4,001	.014	.0291	.0431
de 4,001 a 4,	.0194	.0428	.0622
401 a 4,	.0317	.0775	.1092
menos de	.0074	.0254	.0328
Área de	.0286	.1311	.1598
Área de	.0079	.0371	.045
Total	.289	.711	1

Key: cell proportions

Pearson:

Uncorrected chi2(7) = 1087.2036

Design-based F(6.40, 21745.66)= 104.1371 P = 0.0000

## Modelo de Regresión

```
// Comando: Modelos de regresión svy
```

```
. *Ecuación de participación laboral
```

```
. svy: logit ocupado genero edad yeareduca tamnf numhij* rural telefono celular internet  
(running logit on estimation sample)
```

Survey: Logistic regression

```
Number of strata =      8  
Number of PSUs   =     3406
```

```
Number of obs    =     62095  
Population size  =    21217764  
Design df       =       3398  
F( 11, 3388)    =     246.04  
Prob > F        =     0.0000
```



ocupado	Linearized		t	P> t	[95% Conf. Interval]	
	Coef.	Std. Err.				
genero	.8337505	.0258112	32.30	0.000	.7831434	.8843576
edad	.0208958	.0010715	19.50	0.000	.018795	.0229965
yeareduca	.0814506	.0038069	21.40	0.000	.0739866	.0889146
tamnf	-.0835613	.02635	-3.17	0.002	-.1352247	-.0318979
numhij_nf	.0474246	.0282622	1.68	0.093	-.007988	.1028372
numhij05_nf	.1543872	.0230538	6.70	0.000	.1091864	.199588
numhij612_nf	.2657892	.021297	12.48	0.000	.224033	.3075453
rural	.8338696	.0375555	22.20	0.000	.760236	.9075032
telefono	-.4613688	.0333934	-13.82	0.000	-.5268419	-.3958957
celular	.2017487	.0375183	5.38	0.000	.1281881	.2753094
internet	-.2034448	.0432895	-4.70	0.000	-.2883209	-.1185687
_cons	-1.061984	.0847552	-12.53	0.000	-1.22816	-.8958078

```
. *Ecuación de Salarios
. svy: regress inglab genero exper exper2 hrtrab numhij* rural
(running regress on estimation sample)
```

Survey: Linear regression

Number of strata	=	8	Number of obs	=	18577
Number of PSUs	=	3320	Population size	=	7332556.8
			Design df	=	3312
			F( 8, 3305)	=	471.70
			Prob > F	=	0.0000
			R-squared	=	0.2705

inglab	Linearized		t	P> t	[95% Conf. Interval]	
	Coef.	Std. Err.				
genero	.3667385	.0173177	21.18	0.000	.332784	.400693
exper	.0586678	.0021068	27.85	0.000	.054537	.0627987
exper2	-.0011507	.0000429	-26.85	0.000	-.0012347	-.0010666
hrtrab	.0031544	.0001323	23.84	0.000	.0028949	.0034138
numhij_nf	-.024092	.0077672	-3.10	0.002	-.0393209	-.008863
numhij05_nf	-.0060642	.0159811	-0.38	0.704	-.0373981	.0252697
numhij612_nf	-.0784575	.0165132	-4.75	0.000	-.1108346	-.0460804
rural	-.8493649	.028212	-30.11	0.000	-.9046796	-.7940503
_cons	5.348051	.0357812	149.47	0.000	5.277896	5.418207

## 6.6. Ejercicio Propuesto

Aplice los mismos procedimientos mostrados en este capítulo utilizando la base de datos de la ENAHO para el año 2011.

## Parte II

# Modelos de Regresión Lineal



## Capítulo 7

# Modelo de Regresión Lineal General

### 7.1. Especificación y Supuestos del Modelo General

En los modelos de regresión lineal se requiere explicar el comportamiento de una variable (dependiente) a partir de otras (regresoras ó explicativas). Utilizamos el concepto de distribución de probabilidad condicionada, pues se busca conocer los estimadores de los parámetros de regresión con la finalidad de estimar finalmente el  $E(Y|X = x)$ , es decir buscamos conocer el valor esperado de  $Y$ , dado que  $X = x$  (la variable regresora toma un determinado valor  $x$ ).

$$y_i = \alpha + \beta x_i + \mu_i \quad (7.1)$$

La variable  $y_i$  es la variable dependiente, las variables  $x_i$  son las variables explicativas o regresoras, y  $\mu_i$  es la perturbación aleatoria o comúnmente llamado termino de error. Los  $\beta$  son los parámetros asociados a cada una de las variables explicativas, también llamados coeficientes de regresión y miden el impacto de cada variable independiente en relación al comportamiento de la variable endógena.

Los parámetros  $\alpha$  y  $\beta$  son no conocidos. Sin embargo, utilizando información muestral se pueden obtener estimadores de los parámetros (o coeficientes).

Dado la expresión anterior, se considera que se cumplen las siguientes hipótesis clásicas básicas:

- Linealidad en los parámetros.
- No existen relaciones lineales entre las variables explicativas o regresores y estos no son variables aleatorias (no multicolinealidad).
- La esperanza del vector de la variable aleatoria es cero:  $E(\mu) = 0$ .
- La matriz de varianzas y covarianzas del vector de variables aleatorias es:  $E(\mu\mu') = \sigma_\mu^2 I$ . Es decir, todos los componentes del vector  $\mu$  tienen varianza idéntica (homoscedasticidad), y además las covarianzas son 0, es decir, los elementos del vector  $\mu$  no están correlacionados (no autocorrelación).
- La distribución de probabilidad del vector de perturbaciones aleatorias es:  $\mu \sim N(0, \sigma_\mu^2 I)$ , es decir es un vector normal esférico.
- Por tanto, las perturbaciones son variables aleatorias independientes e igualmente distribuidas, normales con media cero y varianza  $\sigma_\mu^2 I$ . Dado que  $X$  no es aleatoria, la distribución de probabilidad del vector  $Y$  se deriva a partir del vector de perturbaciones:  $Y \sim N(X\beta, \sigma_\mu^2 I)$ .

## 7.2. Formas Funcionales

Las principales formas funcionales a estimar se muestran a continuación:

La interpretación de los parámetros para cada forma funcional se explicará en los ejercicios aplicativos.

### 7.3. Bondad de Ajuste

La *Bondad de Ajuste* es entendida –en términos sencillos– lo *bien* que los datos se ajustan a la regresión. Par ello, se plantean distintos *indicadores* que permiten seleccionar las variables que deben ser explicativas en un modelo econométrico. Entre los principales (de fácil aplicación), se incluyen:

#### 7.3.1. Coeficiente de Determinación

El *Coeficiente de Determinación*  $R^2$  es el que mide el nivel de ajuste del modelo que se ha estimado, es decir, evalúa si la(s) variable(s) regresora(s) explica adecuadamente la variable dependiente. Si por ejemplo, el coeficiente de determinación fuera 0.90 significa que la variación de la variable dependiente es explicada por la(s) variable(s) regresora(s) en un 90 %, el 10 % restante es explicado por el residuo. Nótese que en la formula mostrada a continuación, el  $R^2$  depende de la **Suma de cuadrados del Residuo (SCR)** y la **Suma de cuadrados Totales (SCT)**.

$$R^2 = 1 - SCR/SCT$$

#### 7.3.2. Coeficiente de Determinación Ajustado

En general, se refiere a la proporción de la variación en  $Y$ , que es explicada por la(s) variable(s) explicativa(s). Se define de tal modo que penaliza la inclusión de nuevas variables explicativas en el modelo (si bien al aumentar el número de regresores aumenta también la Suma de Cuadrados Explicados, la inclusión de nuevas variables explicativas reduce los grados de libertad del modelo, por lo que no siempre resultará adecuado incorporar nuevas variables al mismo). El *Coeficiente de Determinación Ajustado* ( $\bar{R}^2$ ) es explicado mediante la siguiente fórmula:

$$\bar{R}^2 = 1 - \frac{n-1}{n-k} \frac{SCR}{SCT}$$

## 7.4. Prueba de Hipótesis e Intervalo de Confianza

Luego de estimar los parámetros del modelo de regresión lineal, es útil evaluarlos a través de la contrastación de hipótesis en el contexto del análisis de regresión lineal. Supongamos que se estime el siguiente modelo:  $Y = a + \beta_1 X$ . Por ejemplo, si se desea evaluar si  $\beta_1$  es significativo, es decir si  $X$  ayuda a explicar la variable dependiente, se realiza la siguiente hipótesis:

$$H_0 : \beta_1 = 0 \text{ (hipótesis nula)}$$

$$H_a : \beta_1 \neq 0 \text{ (hipótesis alterna)}$$

- Se acepta  $H_0$  si:  $t_{calculado} = t_{tabla} - (n - k)gl$ . Si  $pvalue = 0,05$ .
- Se rechaza  $H_0$  si:  $t_{calculado} > t_{tabla} - (n - k)gl$ . Si  $pvalue < 0,05$ .

## 7.5. Criterios para elección de modelos

### 7.5.1. Criterio de Información de AKAIKE (AIC)

Estadístico que mide el buen ajuste de la data a la regresión estimada, permitiendo la selección entre dos modelos de ajuste alternativos. También penaliza la inclusión de nuevos regresores en el modelo, seleccionando como modelo más adecuado aquel que presenta un menor valor de dicho coeficiente. Su fórmula de cálculo responde a la siguiente expresión:

$$AIC = \ln(SCR/n) + 2k/n$$

### 7.5.2. Criterio de Información de SCHWARZ (BIC)

Este criterio es una alternativa más restrictiva al criterio  $AIC$ , ya que permite la selección de variables que deben ser incluidas en el modelo. Este criterio penaliza en un grado mayor la inclusión de nuevos regresores en el modelo. Al igual que en el caso anterior, se considera mejor modelo aquel que presente un menor valor del coeficiente.

$$BIC = \ln(n)k/n + \ln(SCR/n)$$

**Nota:** Para poder comparar modelos según los criterios  $AIC$  y  $BIC$ , es requisito obligatorio que las estimaciones a comparar tengan la misma variable dependiente.

#### Ejercicio

Utilice los datos de archivo *carnes.xlsx* donde encontrará información de consumo de carnes de pollo, ovino y res (qpól, qovi y qres, respectivamente), también encontrará los precios de las carnes (ppól, povi y pres), y el ingreso (ing).

- Realice gráficos descriptivos de las variables explicativas y dependiente.
- Estime por MCO EN LA FORMA LIN-LIN según lo mencionado arriba. Guarde dicha estimación.
- Si encuentra alguna variable no significativa, pruebe otra estimación alternativa. Guarde dicha estimación.
- Compare los resultados de b) y c). ¿Cuál es el mejor modelo?
- Dado el mejor modelo escogido, evalúe bajo el Test de Ramsey si existe alguna señal de no linealidad u omisión de alguna variable relevante en el modelo.
- Estime como en b) pero usando una forma funcional LOG-LOG para la demanda de carne de ovino.
- Pruebe si se cumple la condición de homogeneidad en la demanda de carne de ovino.



- Estime por MCR suponiendo que en la función de demanda de carne de ovino se cumple la condición de homogeneidad.
- Encuentre el estimador del parámetro restringido, su varianza, su error estándar y su estadístico t-student.

**NOTA:** Acuérdesse que cuando se tiene que trabajar con datos que se encuentran en archivo de excel y se desea importar dicha información al Stata, se requiere *transformar* el formato de excel (.xls o .xlsx) a un formato (.csv) de tal forma de poder importarlo sin problemas. El formato *.csv* significa *delimitado por comas*.

### Solución

En el programa de Stata se realizará los siguientes pasos:

**Paso 1:** Especificamos la ruta donde se encuentra el archivo usando el siguiente comando:

```
. * MODELO DE REGRESIÓN LINEAL GENERAL
. *****

. *Limpiamos la memoria
. clear

. *Paso 1: Especificamos la ruta donde se encuentra el archivo usando el siguiente comando:
. cd "D:\Econometria-Stata\modelo-regresion-lineal"
D:\Econometria-Stata\modelo-regresion-lineal
```

**Paso 2:** Importamos la base de datos a usar al Stata:

```
. *Paso 2: Importamos la base de datos a usar al Stata:
. insheet using carnes.csv
(8 vars, 30 obs)

. browse

. describe

Contains data
   obs:                30
   vars:                 8
   size:              1,140 (99.5% of memory free)
```

---

variable name	storage type	display format	value label	variable label
obs	int	%8.0g		
ing	float	%9.0g		ING
povi	float	%9.0g		POVI
ppol	float	%9.0g		PPOL
pres	float	%9.0g		PRES
qovi	float	%9.0g		QOVI
qres	float	%9.0g		QRES
qpol	float	%9.0g		QPOL

Sorted by:

Note: dataset has changed since last saved

. summarize

Variable	Obs	Mean	Std. Dev.	Min	Max
obs	30	1974.5	8.803408	1960	1989
ing	30	409.5016	127.9457	184.8151	760.3892
povi	30	7.3559	3.214927	1.996	15.533
ppol	30	7.930467	3.9555	2.471	15.459
pres	30	8.373533	2.841277	4.016	14.219
qovi	30	8.363833	4.69644	1.36	19.623
qres	30	10.89157	6.671015	2.903	33.908
qpol	30	44.62747	32.25129	13.251	137.269

. summarize, detail

obs					
Percentiles		Smallest			
1%	1960	1960			
5%	1961	1961			
10%	1962.5	1962	Obs	30	
25%	1967	1963	Sum of Wgt.	30	
50%	1974.5		Mean	1974.5	
		Largest	Std. Dev.	8.803408	
75%	1982	1986			
90%	1986.5	1987	Variance	77.5	
95%	1988	1988	Skewness	0	
99%	1989	1989	Kurtosis	1.79733	
ING					
Percentiles		Smallest			
1%	184.8151	184.8151			
5%	192.061	192.061			
10%	275.1244	256.1177	Obs	30	
25%	312.654	294.1311	Sum of Wgt.	30	
50%	385.9284		Mean	409.5016	
		Largest	Std. Dev.	127.9457	
75%	487.6345	564.2039			
90%	571.2145	578.2252	Variance	16370.11	
95%	629.3932	629.3932	Skewness	.6043204	
99%	760.3892	760.3892	Kurtosis	3.405305	
POVI					
Percentiles		Smallest			
1%	1.996	1.996			
5%	3.369	3.369			

10%	4.165	4.142	Obs	30
25%	5.034	4.188	Sum of Wgt.	30
50%	6.9585		Mean	7.3559
		Largest	Std. Dev.	3.214927
75%	9.145	11.417		
90%	12.366	13.315	Variance	10.33575
95%	14.01	14.01	Skewness	.8562475
99%	15.533	15.533	Kurtosis	3.24582
PPOL				

	Percentiles	Smallest		
1%	2.471	2.471		
5%	2.708	2.708		
10%	3.7375	3.707	Obs	30
25%	4.157	3.768	Sum of Wgt.	30
50%	6.8755		Mean	7.930467
		Largest	Std. Dev.	3.9555
75%	11.115	13.774		
90%	14.046	14.318	Variance	15.64598
95%	14.922	14.922	Skewness	.4459752
99%	15.459	15.459	Kurtosis	1.942058

PRES				
	Percentiles	Smallest		
1%	4.016	4.016		
5%	4.605	4.605		
10%	4.851	4.759	Obs	30
25%	6.535	4.943	Sum of Wgt.	30
50%	7.7375		Mean	8.373533
		Largest	Std. Dev.	2.841277
75%	10.763	13.033		
90%	13.039	13.045	Variance	8.072857
95%	13.511	13.511	Skewness	.5208476
99%	14.219	14.219	Kurtosis	2.284577

QOVI				
	Percentiles	Smallest		
1%	1.36	1.36		
5%	2.181	2.181		
10%	3.011	2.682	Obs	30
25%	4.695	3.34	Sum of Wgt.	30
50%	7.505		Mean	8.363833
		Largest	Std. Dev.	4.69644
75%	10.963	13.194		
90%	15.1195	17.045	Variance	22.05655
95%	17.141	17.141	Skewness	.6309666
99%	19.623	19.623	Kurtosis	2.690953

QRES				
	Percentiles	Smallest		
1%	2.903	2.903		
5%	3.041	3.041		
10%	3.56	3.138	Obs	30
25%	6.231	3.982	Sum of Wgt.	30
50%	9.813		Mean	10.89157
		Largest	Std. Dev.	6.671015
75%	14.096	17.594		
90%	18.218	18.842	Variance	44.50245
95%	22.24	22.24	Skewness	1.426756
99%	33.908	33.908	Kurtosis	5.854005
QPOL				

Percentiles		Smallest		
1%	13.251	13.251		
5%	13.393	13.393		
10%	17.084	15.914	Obs	30
25%	23.667	18.254	Sum of Wgt.	30
50%	34.196	Largest	Mean	44.62747
75%	45.838		Std. Dev.	32.25129
90%	105.918	107.017	Variance	1040.146
95%	123.026	123.026	Skewness	1.635324
99%	137.269	137.269	Kurtosis	4.764288

```
. *Se puede etiquetar las variables con el comando: label var
. *Se puede renombrar las variables con el comando: rename
. rename obs year
```

### Paso 3: Gráficos descriptivos:

```
. *Paso 3: Gráficos descriptivos
. line ing year

. twoway (scatter ing year) (lfit ing year)

. scatter qres qpol

. h graphs
```

### Paso 4: Estimación por MCO –Función LIN-LIN.

```
. *Paso 4: Estimación por MCO --Función LIN-LIN
. reg qovi ppol pres povi ing
```

Source	SS	df	MS			
Model	337.526915	4	84.3817288	Number of obs =	30	
Residual	302.113028	25	12.0845211	F( 4, 25) =	6.98	
Total	639.639943	29	22.0565498	Prob > F	= 0.0006	
				R-squared	= 0.5277	
				Adj R-squared	= 0.4521	
				Root MSE	= 3.4763	

qovi	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
ppol	.057105	.1633502	0.35	0.730	-.279321	.393531
pres	.5198569	.230288	2.26	0.033	.0455698	.994144
povi	-.5957813	.2039296	-2.92	0.007	-1.015782	-.1757805
ing	.0182565	.0050538	3.61	0.001	.007848	.0286649
_cons	.4643743	3.595589	0.13	0.898	-6.94088	7.869629

Como observamos, el precio del pollo es no significativo ya que tiene una probabilidad mayor a 0.05, por lo tanto planteamos otra estimación sin considerar dicha variable. Tenga en cuenta también, que la constante no es significativa (nocons).

```
. * Guardando la ecuación anterior
```

```
. estimates store eq01
```

```
. *Veamos una prueba de hipótesis
```

```
. reg qovi ppol pres povi ing
```

Source	SS	df	MS	Number of obs = 30		
Model	337.526915	4	84.3817288	F( 4, 25) = 6.98		
Residual	302.113028	25	12.0845211	Prob > F = 0.0006		
				R-squared = 0.5277		
				Adj R-squared = 0.4521		
Total	639.639943	29	22.0565498	Root MSE = 3.4763		

qovi	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
ppol	.057105	.1633502	0.35	0.730	-.279321	.393531
pres	.5198569	.230288	2.26	0.033	.0455698	.994144
povi	-.5957813	.2039296	-2.92	0.007	-1.015782	-.1757805
ing	.0182565	.0050538	3.61	0.001	.007848	.0286649
_cons	.4643743	3.595589	0.13	0.898	-6.94088	7.869629

```
. test ppol=0
```

```
( 1) ppol = 0
```

```
F( 1, 25) = 0.12
Prob > F = 0.7296
```

```
. *Veamos una prueba de hipótesis
```

```
. test pres povi ing
```

```
( 1) pres = 0
```

```
( 2) povi = 0
```

```
( 3) ing = 0
```

```
F( 3, 25) = 9.30
Prob > F = 0.0003
```

```
. *Estimamos el modelo alternativo
```

```
. reg qovi pres povi ing, nocons
```

Source	SS	df	MS	Number of obs = 30		
Model	2433.75506	3	811.251686	F( 3, 27) = 71.93		
Residual	304.496143	27	11.2776349	Prob > F = 0.0000		
				R-squared = 0.8888		
				Adj R-squared = 0.8764		
Total	2738.2512	30	91.27504	Root MSE = 3.3582		

qovi	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
pres	.55915	.1694383	3.30	0.003	.2114914	.9068086
povi	-.5651093	.1698825	-3.33	0.003	-.9136794	-.2165392
ing	.0190616	.0039329	4.85	0.000	.010992	.0271312

```
. * Guardando la ecuación anterior
```

```
. estimates store eq02
```

```
. * Comparación de modelos
. estimates table eq01 eq02,star stats(N r2 r2_a F aic bic)
```

Variable	eq01	eq02
ppol	.05710497	
pres	.51985689*	.55915001**
povi	-.59578134**	-.56510929**
ing	.01825648**	.01906161***
_cons	.46437427	
N	30	30
r2	.52768267	.88879905
r2_a	.4521119	.87644339
F	6.9826291	71.934558
aic	164.42443	160.66014
bic	171.43041	164.86374

legend: \* p<0.05; \*\* p<0.01; \*\*\* p<0.001

El mejor modelo es aquel que tiene menor AIC y BIC. Por lo tanto el mejor modelo es el segundo. Acuérdesse que para comparar modelos se debe tener la misma variable dependiente. Por otro lado, si se tuvieran AIC y BIC negativos, primero se tiene que multiplicar por menos 1 y luego recién compararlo.

```
. * Dado el mejor modelo (eq02) se procede a evaluar bajo el test de Ramsey si
. * existe señal de no linealidad u omisión de alguna variable relevante en el modelo:
. reg qovi pres pov i ing
```

Source	SS	df	MS	Number of obs = 30		
Model	336.050059	3	112.016686	F( 3, 26) =	9.59	
Residual	303.589884	26	11.676534	Prob > F =	0.0002	
				R-squared =	0.5254	
				Adj R-squared =	0.4706	
Total	639.639943	29	22.0565498	Root MSE =	3.4171	

qovi	Coef.	Std. Err.	t	P> t	[95 % Conf. Interval]	
pres	.518302	.2263251	2.29	0.030	.0530842	.9835198
povi	-.5933208	.2003382	-2.96	0.006	-1.005122	-.1815198
ing	.0182417	.0049676	3.67	0.001	.0080307	.0284526
_cons	.9182247	3.295942	0.28	0.783	-5.856682	7.693132

```
. ovtest
```

```
Ramsey RESET test using powers of the fitted values of qovi
Ho: model has no omitted variables
F(3, 23) = 0.90
Prob > F = 0.4560
```

```
. *Por lo tanto, no existe señal de no linealidad u omisión de alguna variable
. *independiente relevante.
. *Nota: Fíjese que para aplicar dicho test, debe estimarse con intercepto siempre.
```

```
. * Realizamos una tabla igual que la anterior solo que más formal para usar
. esttab eq01 eq02, b(%9.3f) star stats(N r2 r2_a F aic bic) ///
  mtitles("Eq01" "Eq02") title("Comparaciones de Modelos")
```

Comparaciones de Modelos

	(1) Eq01	(2) Eq02
ppol	0.057 (0.35)	
pres	0.520* (2.26)	0.559** (3.30)
povi	-0.596** (-2.92)	-0.565** (-3.33)
ing	0.018** (3.61)	0.019*** (4.85)
_cons	0.464 (0.13)	
N	30.000	30.000
r2	0.528	0.889
r2_a	0.452	0.876
F	6.983	71.935
aic	164.424	160.660
bic	171.430	164.864

t statistics in parentheses  
 \* p<0.05, \*\* p<0.01, \*\*\* p<0.001

```
. ** Usamos outreg para tener la estimación anterior en un Excel
```

```
. outreg2 [eq01 eq02] using tabla1, replace see
```

Hit Enter to continue.

dir : seeout

```
. outreg2 using tabla1,excel
```

tabla1.xml

dir : seeout

### Paso 5: Generamos las variables en logaritmos.

```
. *Paso 5: Generamos las variables en logaritmos:
```

```
. generate logqovi=log(qovi)
```

```
. generate logppol=log(ppol)
```

```
. generate logpres=log(pres)
```

```
. generate logpovi=log(povi)
```

```
. generate loging=log(ing)
```

**Paso 6:** Estimamos por MCO – FUNCION LOGARITMICA (LOG-LOG).

```
. *Paso 6: Estimamos por MCO -- FUNCION LOGARITMICA (LOG-LOG)
```

```
. reg logqovi logppol logpres logpovi logging
```

Source	SS	df	MS	Number of obs = 30		
Model	6.71971035	4	1.67992759	F( 4, 25) =	8.03	
Residual	5.23042267	25	.209216907	Prob > F =	0.0003	
				R-squared =	0.5623	
				Adj R-squared =	0.4923	
Total	11.950133	29	.412073552	Root MSE =	.4574	

logqovi	Coef.	Std. Err.	t	P> t	[95 % Conf. Interval]	
logppol	.135395	.1611994	0.84	0.409	-.1966014	.4673914
logpres	.1909937	.2565774	0.74	0.464	-.3374374	.7194248
logpovi	-.6324893	.1962859	-3.22	0.004	-1.036748	-.2282309
logging	1.140894	.2653558	4.30	0.000	.5943833	1.687404
_cons	-4.313317	1.755359	-2.46	0.021	-7.928548	-.6980869

Prueba de hipótesis si cumple la condición de homogeneidad:

```
. *Prueba de hipótesis si cumple la condición de homogeneidad
```

```
. test logppol + logpres+ logpovi+ logging=0
```

```
( 1) logppol + logpres + logpovi + logging = 0
```

```
F( 1, 25) = 3.17
Prob > F = 0.0870
```

Estimación por MCR con la condición de homogeneidad donde:  $c(5)=-c(2)-c(3)-c(4)$  (si quieres ver los coeficientes del MCR, agregue al comando test la opción *coef*):

```
. *Estimación por MCR con la condición de homogeneidad donde:
. *c(5)=-c(2)-c(3)-c(4) Si quieres ver los coeficientes del MCR,
. *agregue al comando test la opción coef:
```

```
. test logging= -logppol -logpres- logpovi, coef
```

```
( 1) logppol + logpres + logpovi + logging = 0
```

```
F( 1, 25) = 3.17
Prob > F = 0.0870
```

**Otra forma:**

Definimos la restricción:

```
. *Definimos la restricción:
```



```
. constraint define 1 logging= -logppol -logpres- logpovi
```

```
. *Estimamos la regresión por MCR:
```

```
. cnsreg logqovi logppol logpres logpovi logging, constraint(1)
```

Constrained linear regression

Number of obs	=	30
F( 3, 26)	=	8.90
Prob > F	=	0.0003
Root MSE	=	0.4761

```
( 1) logppol + logpres + logpovi + logging = 0
```

	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
logppol	.0332251	.156823	0.21	0.834	-.2891291 .3555793
logpres	-.1001946	.2058752	-0.49	0.631	-.5233771 .3229879
logpovi	-.8120008	.1753465	-4.63	0.000	-1.172431 -.4515708
logging	.8789703	.2299482	3.82	0.001	.4063051 1.351635
_cons	-1.607849	.9163453	-1.75	0.091	-3.491423 .2757262

### Interpretación de los coeficientes según los tipos de modelos a estimar:

Sea por ejemplo, la variable dependiente producción (toneladas) y una de las variables explicativas es la temperatura mínima ( $C^o$ ). Si el beta asociado a la variable regresora temperatura mínima es 0.83. *¿Cómo se interpretaría este beta en un modelo LIN-LIN, LOG-LOG, LOG-LIN y LIN-LONG?*

#### MODELO LIN - LIN:

$\beta$  = Si la temperatura mínima se incrementara en 1  $C^o$ , la producción se incrementa en 0.83 toneladas.

#### MODELO LOG-LOG:

$\beta$  = La elasticidad de la producción respecto a la temperatura mínima es de 0.83. Lo que sugiere que si la temperatura mínima se incrementa en 1 %, en promedio, la producción se incrementa en 0.83 %.

#### MODELO LOG-LIN:

$\beta$  = La producción se incrementa a una tasa (anual) de  $0.83 \times 100 = 83\%$  dado el incremento en un grado centígrado de la temperatura mínima.

## MODELO LIN-LONG:

$\beta$  = Un incremento en la temperatura mínima de 1 % en promedio, propicia un incremento en la producción de 0.83/100 toneladas.

## 7.6. Pruebas de Hipotesis y Estimación MCO con Variables Dummy

En el archivo **dummy\_africa.csv** se encuentra información de 27 países árabes sobre el PBI per cápita *PCGDP*, origen colonial *COLONIAL* (británico, francés, etc.), ubicación geográfica *GEO* (norte, centro, sur y oeste) y porcentaje de tierras arables (*P\_ARABLE*).

```
. *Pruebas de Hipotesis y Estimación MCO usando Variables Dummy
. *****

. *Paso 1: Buscamos la ruta donde se encuentra el archivo
. cd "D:\Econometria-Stata\modelo-regresion-lineal"
D:\Econometria-Stata\modelo-regresion-lineal

. *Paso 2: Importación de datos

. insheet using dummy_africa.csv ,clear
(5 vars, 27 obs)

. *Paso 3: Etiquetando las variables que e importado

. label var country "países"
. label var pcgdp "PBI per cápita de los países"
. label var colonial "origen colonial, britanico, frances, etc."
. label var geo "ubicación geogr\U{e1}fica: norte, centro, sur, oeste"

. br

. *Paso 4: Generando variables dummy

. g france =colonial=="France"
. g britain = colonial=="Britain"
. g other= 1-france-britain
. g central=geo=="Central"
. g north=geo=="North"
. g south=geo=="South"
. g west=geo=="West"

. br
```

```
. *Paso 5: Estimando regresiones OLS usando variables dummy y guardándolas
. *Usando todas las dummies geográficas y por tanto no incluyo la constante
. xi: reg pcgdp central north south west p_arable, nocons.
```

Source	SS	df	MS	Number of obs = 27		
Model	22826040.8	5	4565208.15	F( 5, 22) = 2.49		
Residual	40323163.2	22	1832871.06	Prob > F = 0.0623		
				R-squared = 0.3615		
				Adj R-squared = 0.2163		
Total	63149204	27	2338859.41	Root MSE = 1353.8		

pcgdp	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
central	834.4267	885.9427	0.94	0.356	-1002.906	2671.759
north	1561.227	515.6808	3.03	0.006	491.7707	2630.684
south	934.278	668.2436	1.40	0.176	-451.5744	2320.13
west	671.0288	668.7515	1.00	0.327	-715.8769	2057.934
p_arable	-19.03867	32.66036	-0.58	0.566	-86.77212	48.69478

```
. * Guardando la ecuación anterior
. estimates store eq01

. * Igual que la ecuación anterior, incorporando la constante o intercepto
. xi: reg pcgdp central north south p_arable
```

Source	SS	df	MS	Number of obs = 27		
Model	5956375.43	4	1489093.86	F( 4, 22) = 0.81		
Residual	40323163.2	22	1832871.06	Prob > F = 0.5307		
				R-squared = 0.1287		
				Adj R-squared = -0.0297		
Total	46279538.7	26	1779982.26	Root MSE = 1353.8		

pcgdp	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
central	163.3979	817.2867	0.20	0.843	-1531.551	1858.347
north	890.1984	723.7104	1.23	0.232	-610.6852	2391.082
south	263.2492	723.2156	0.36	0.719	-1236.608	1763.107
p_arable	-19.03867	32.66036	-0.58	0.566	-86.77212	48.69478
_cons	671.0288	668.7515	1.00	0.327	-715.8769	2057.934

```
. * Guardando la ecuación anterior
. estimates store eq02

. *Incorpo todas las dummies y excluyo la constante
. xi: reg pcgdp france britain other central north south west p_arable, nocons
note: other omitted because of collinearity
```

Source	SS	df	MS	Number of obs = 27		
Model	26279794.4	7	3754256.35	F( 7, 20) = 2.04		
Residual	36869409.6	20	1843470.48	Prob > F = 0.1005		
				R-squared = 0.4162		
				Adj R-squared = 0.2118		
Total	63149204	27	2338859.41	Root MSE = 1357.7		

pcgdp	Coef.	Std. Err.	t	P> t	[95 % Conf. Interval]	
france	-1053.707	772.8657	-1.36	0.188	-2665.876	558.4629
britain	-512.9431	723.6893	-0.71	0.487	-2022.533	996.6463
other	(omitted)					
central	1243.614	947.5785	1.31	0.204	-732.9995	3220.229
north	2203.467	698.4497	3.15	0.005	746.5269	3660.408
south	1462.087	872.0647	1.68	0.109	-357.0082	3281.182
west	1559.272	933.4624	1.67	0.110	-387.8967	3506.44
p_arable	-27.76531	34.77613	-0.80	0.434	-100.3071	44.77644

```
. * Guardando la ecuación anterior
```

```
. estimates store eq03
```

```
. * Igual que la anterior, excluyendo una variable dummy geográfica
```

```
. xi: reg pcgdp france britain other central north south p_arable, nocons
```

Source	SS	df	MS	Number of obs = 27		
Model	26279794.4	7	3754256.35	F( 7, 20) = 2.04		
Residual	36869409.6	20	1843470.48	Prob > F = 0.1005		
				R-squared = 0.4162		
				Adj R-squared = 0.2118		
Total	63149204	27	2338859.41	Root MSE = 1357.7		

pcgdp	Coef.	Std. Err.	t	P> t	[95 % Conf. Interval]	
france	505.565	687.2577	0.74	0.470	-928.0295	1939.159
britain	1046.329	939.936	1.11	0.279	-914.3437	3007.001
other	1559.272	933.4624	1.67	0.110	-387.8967	3506.44
central	-315.6572	899.0002	-0.35	0.729	-2190.939	1559.624
north	644.1958	748.5282	0.86	0.400	-917.2067	2205.598
south	-97.18468	861.4306	-0.11	0.911	-1894.098	1699.728
p_arable	-27.76531	34.77613	-0.80	0.434	-100.3071	44.77644

```
. * Guardando la ecuación anterior
```

```
. estimates store eq04
```

```
. * Igual que la anterior, excluyendo ahora una variable dummy - origen colonial
```

```
. xi: reg pcgdp france other central north south west p_arable, nocons
```

Source	SS	df	MS	Number of obs = 27		
Model	26279794.4	7	3754256.35	F( 7, 20) = 2.04		
Residual	36869409.6	20	1843470.48	Prob > F = 0.1005		
				R-squared = 0.4162		
				Adj R-squared = 0.2118		
Total	63149204	27	2338859.41	Root MSE = 1357.7		

pcgdp	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
france	-540.7636	793.0095	-0.68	0.503	-2194.952	1113.425
other	512.9431	723.6893	0.71	0.487	-996.6463	2022.533
central	730.6713	1013.133	0.72	0.479	-1382.687	2844.03
north	1690.524	823.2165	2.05	0.053	-26.6752	3407.724
south	949.1439	704.1384	1.35	0.193	-519.6632	2417.951
west	1046.329	939.936	1.11	0.279	-914.3437	3007.001
p_arable	-27.76531	34.77613	-0.80	0.434	-100.3071	44.77644

```
. * Guardando la ecuación anterior
. estimates store eq05

. *Paso 6: Comparación de modelos
. estimates table eq01 eq02 eq03 eq04 eq05 ,star stats(N r2 r2_a F aic bic)
```

Variable	eq01	eq02	eq03	eq04	eq05
central	834.42672	163.39793	1243.6145	-315.65721	730.67135
north	1561.2272**	890.1984	2203.4675**	644.19576	1690.5243
south	934.27803	263.24925	1462.087	-97.184685	949.14387
west	671.02878		1559.2717		1046.3286
p_arable	-19.03867	-19.03867	-27.76531	-27.76531	-27.76531
france			-1053.7067	505.56495	-540.7636
britain			-512.94314	1046.3286	
other			(omitted)	1559.2717	512.94314
_cons		671.02878			

N	27	27	27	27	27
r2	.36146205	.1287043	.41615401	.41615401	.41615401
r2_a	.21633979	-.02971311	.21180791	.21180791	.21180791
F	2.4907416	.81243787	2.0365156	2.0365156	2.0365156
aic	470.47087	470.47087	472.05319	472.05319	472.05319
bic	476.95006	476.95006	481.12405	481.12405	481.12405

legend: \* p<0.05; \*\* p<0.01; \*\*\* p<0.001

```
. *Realizamos una tabla igual que la anterior solo que más formal para usar
. esttab eq01 eq02 eq03 eq04 eq05, b(%9.3f) star stats(N r2 r2_a F aic bic) ///
mtitles("Eq01" "Eq02" " Eq03" "Eq04" "Eq05") title("Comparaciones de Modelos")
```

Comparaciones de Modelos

	(1) Eq01	(2) Eq02	(3) Eq03	(4) Eq04	(5) Eq05
central	834.427 (0.94)	163.398 (0.20)	1243.614 (1.31)	-315.657 (-0.35)	730.671 (0.72)
north	1561.227** (3.03)	890.198 (1.23)	2203.467** (3.15)	644.196 (0.86)	1690.524 (2.05)
south	934.278 (1.40)	263.249 (0.36)	1462.087 (1.68)	-97.185 (-0.11)	949.144 (1.35)
west	671.029 (1.00)		1559.272 (1.67)		1046.329 (1.11)
p_arable	-19.039 (-0.58)	-19.039 (-0.58)	-27.765 (-0.80)	-27.765 (-0.80)	-27.765 (-0.80)
france			-1053.707 (-1.36)	505.565 (0.74)	-540.764 (-0.68)
britain			-512.943 (-0.71)	1046.329 (1.11)	
o.other			0.000 (.)	1559.272 (1.67)	512.943 (0.71)
_cons		671.029 (1.00)			
N	27.000	27.000	27.000	27.000	27.000
r2	0.361	0.129	0.416	0.416	0.416
r2_a	0.216	-0.030	0.212	0.212	0.212
F	2.491	0.812	2.037	2.037	2.037
aic	470.471	470.471	472.053	472.053	472.053
bic	476.950	476.950	481.124	481.124	481.124

t statistics in parentheses

\* p<0.05, \*\* p<0.01, \*\*\* p<0.001

. \*\*Usamos outreg para tener la estimación anterior en un Excel

. outreg2 [eq01 eq02 eq03 eq04 eq05] using tabla1, replace see  
Hit Enter to continue.  
dir : seeout

. outreg2 using tabla1,excel  
tabla1.xml  
dir : seeout

## 7.7. Ejercicio Propuesto

Se cree que el precio que un país paga por los medicamentos depende de su capacidad de pago y de las restricciones legales que el país ha impuesto para controlar el precio de los medicamentos. Para comprender este problema se ha obtenido

información de 32 países **pharma.dta** para las siguientes ocho variables:

$P$  =precios del medicamentos (índice)

$GDPN$  =Ingreso per cápita (índice)

$CV$  =Volumen de consumo (índice)

$N$  =Población (índice)

$CVN$  =Volumen de consumo per cápita (índice)

$PP$  =Existencia de protección de patentes (dummy)

$IPC$  =Existencia de controles indirectos de precios (dummy)

$DPC$  =Existencia de controles directos de precios (dummy).

Se espera que GDP afecte positivamente los precios, porque la demanda sea más inelástica y que el consumo per cápita de medicamentos afecte negativamente los precios como reflejo de la Ley de la Demanda. Por ahora se ignorará las variables de política pública PP, IPC y DPC.

- Si se supone que los costos marginales son constantes en cada país y que la demanda individual es lineal, estime por MCO la siguiente función de demanda:

$$P_i = \beta_1 + \beta_2 GDPN_i + \beta_3 CVN_i + u_i$$

- Pruebe la hipótesis nula que  $\beta_3 = 0$  al 10 % de nivel de significación. Cuál es el significado económico de esta hipótesis nula. Diga cuál es el significado de la hipótesis alternativa.

Las variables *dummy* PP, IPC y DPC incluidas en el set de datos tienen valores 1 si la política pública está presente y 0 en caso contrario.  $PP = 1$  implica que el país ha suscrito acuerdos internacionales que lo obligan a respetar las patentes. Se

espera que estos países tengan precios más altos y la demanda menor. Si  $IPC = 1$  si el gobierno es el principal comprador de medicamentos para todos los otros consumidores, generándose un monopolio bilateral, por lo que se espera que su efecto sea tener precios más bajos que el caso de un monopolio puro, por lo que, ceteris paribus, la demanda se desplaza hacia abajo. Finalmente, si  $DPC = 1$  el gobierno establece controles de precios a los medicamentos para abaratar los precios y aumentar la demanda.

- Estime la siguiente función de demanda:

$$P_i = \beta_1 + \beta_2 GDPN_i + \beta_3 CVN_i + \beta_4 PP_i + \beta_5 IPC_i + \beta_6 DPC_i + v_i$$

- Pruebe la significancia conjunta de las variables del modelo al 10
- Pruebe la hipótesis nula para los coeficientes asociadas a las variables dummy al 10 % de nivel de significación. Es decir:  $H_0 : \beta_1 = \beta_2 = \beta_3 = \beta_4 = \beta_5 = \beta_6 = 0$ .
- ¿Los signos de las variables de política tienen los signos esperados?.
- Construya un intervalo de confianza del 90 % para el coeficiente asociado a DPC (control directo de precios). Interprete los resultados.





# Capítulo 8

## Heteroscedasticidad

### 8.1. Problema de Heteroscedasticidad

Antes de empezar con los problemas de las perturbaciones no esféricas, primero es primordial ver las cuatro condiciones de Gauss-Markov para el planteamiento del modelo lineal general. Suponiendo una ecuación de regresión de la forma:

$$y_i = \alpha + \beta x_i + u_i$$

Estas condiciones (que son parte de los supuestos del modelo lineal general) asumen que el término de error o perturbación puede resumirse en:

1.  $E(u_i) = 0$  para todo  $i$ .
2.  $Var(u_i) = \sigma^2$  (constante) para todo  $i$ .
3.  $Cov(u_i u_j) = 0$  para todo  $i \neq j$ .
4.  $Cov(x_i u_j) = 0$  la cual implica que las variables explicativas son no estocásticas.

El segundo y tercer supuesto se pueden resumir en términos matriciales (para el modelo lineal general) a través de la siguiente expresión:

$$E(uu') = \sigma_u^2 I_n$$

Cuando se cumplen estas dos condiciones se dice que los errores son esféricos. La violación del segundo supuesto da origen al problema de **heteroscedasticidad** e implica que la varianza del término de error no es constante para cada observación. Por otro lado, si los elementos fuera de la diagonal de la matriz de varianzas y covarianzas de los errores son distintos de cero, se viola el tercer supuesto y como resultado tendremos el problema de **autocorrelación**, el cual se tratará en el siguiente capítulo. Este problema significa que los términos de error no son independientes, es decir, el tamaño del error para un periodo determinado afecta el valor del periodo u observación siguiente.

En general, ante problemas de heteroscedasticidad o autocorrelación los estimadores serán lineales (porque es una función lineal de los valores de  $x$ ), insesgados (por que el valor esperado de  $\hat{\beta}$  es igual al verdadero  $\beta$ ) y consistentes (porque se aproxima al verdadero valor  $\beta$  conforme el tamaño de muestra se hace más grande), pero no serán los mejores estimadores linealmente insesgados (MELI) pues no es eficiente (no poseen la mínima varianza).

La segunda condición de Gauss - Markov implica que la varianza de la perturbación debe ser constante para cada observación. Si este supuesto se verifica para toda la muestra se puede concluir que los errores son homocedásticos. En este sentido, el supuesto de homocedasticidad implica que la distribución relevante para cada observación es la misma. En algunos casos, sin embargo, puede ser más razonable pensar que la distribución del término de error es diferente para cada observación en cuyo caso su varianza también diferiría.

Por ejemplo, el hecho de que la varianza de la perturbación muestre un comportamiento creciente para cada observación no significa que el error deba necesariamente registrar un valor muy alto en las últimas observaciones pero sí implica que la probabilidad de tener un valor errático sea mayor. Este es un ejemplo de heterocedasticidad la cual, en términos generales, significa que el error muestra diferentes dispersiones para cada observación o, lo que es lo mismo, que la probabilidad de que el término de error tome un determinado valor es diferente para cada observación. A manera de resumen, sea el modelo lineal general en términos matriciales:

$$Y_i = \alpha + \beta X_i + u_i$$

La matriz de varianzas y covarianzas en presencia de heteroscedasticidad está dado por:

$$E(uu') = \sigma_u^2 \Sigma, \text{ donde } \Sigma \neq I_n$$

El problema de la heteroscedasticidad se da frecuentemente por los siguientes casos:

- Relación entre las variables explicativas y la varianza del error.
- Datos agregados.
- Errores de especificación.

### Ejercicio.

Se tiene información del modulo 500 (Empleo e Ingreso) de la ENAHO para el año 2009, la cual se trabajará sobre las siguientes variables: *GASTO* (fracción gasta en alimentos) y *LINGPC* (logaritmo del ingreso per capita en la familia), dichos datos se obtuvieron de las características del jefe de hogar. Usando los datos del archivo **engel.dta** se le pide estimar por MCO la "Curva de Engel":

$$GASTO_i = \alpha + \beta LINGGPC_i + u_i$$

Dado el modelo a estimar, a continuación se procederá a evaluar la existencia de heteroscedasticidad y su posible corrección a dicho problema:

```

. *Heteroscedasticidad
. *****

. *Limpiamos la memoria
. clear

. *Seleccionamos la ruta donde se encuentra el archivo

. cd "D:\Econometria-Stata\heteroscedasticidad"
D:\Econometria-Stata\heteroscedasticidad

. *Abrimos un archivo en Stata (.dta)
. use engel.dta

. *En primer lugar estimamos el modelo de regresion planteado arriba:
. quietly reg gasto lingpc
. estimates store engel

. *mostramos los resultados en una tabla
. estimates table engel, b(%7.2f) se(%7.2f) p(%7.2f) stats(N r2_a aic bic)

```

Variable	engel
lingpc	1568.72 123.17 0.00
_cons	-3159.17 500.11 0.00
N	1247
r2_a	0.11
aic	21839.67
bic	21849.92

legend: b/se/p

```

. *Comando para .tex del output
. outreg2 using myfile, tex
myfile.tex
dir : seeout

```

## 8.2. Test de Heteroscedasticidad

Para evaluar la existencia de heteroscedasticidad (varianza no constante en toda la muestra) se realizan dos pruebas:

1. Prueba informal y que consta en analizar gráficamente el residuo con la(s) variables regresora(s) y la dependiente.

2. La prueba formal, la cual consiste en realizar pruebas estadísticas y que su eficiencia o uso son muchas veces diferenciadas por su carácter de tipo muestral (algunas tienen mayor eficiencia si el tamaño de muestra es bajo u alto), esto será explicado brevemente.

### 8.2.1. Método Informal (Método Gráfico)

```
. *Método Gráfico
. *****

. *Encontrando el residuo de la regresion anterior y se le llama "residuo"
. predict residuo, residual

. *residuo vs la v.regresora
. twoway (scatter residuo lingpc)

. *Se puede graficar lo mismo usando el siguiente comando
. rvpplot lingpc
```

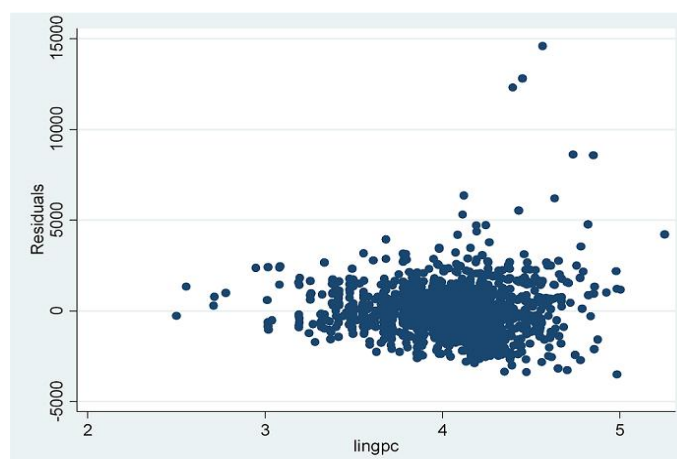


Figura 8.1: Método Gráfico (1) - Heteroscedasticidad

En el gráfico anterior se muestra la relación del residuo y la variable independiente o regresora. Es importante siempre hacer gráfico con una o mas regresoras que fueron estimadas previamente en el modelo de regresión e intentar, de manera visual, tener una idea de la variable que estaría generando la presencia de heteroscedasticidad. Como se observa en el gráfico anterior, posiblemente exista de heteroscedasticidad aunque simplemente proporcionan una sospecha inicial. Una vez realizado esto, se procede a realizar el gráfico de los errores en función de la

variable dependiente. Este se obtiene así:

```
. *residuo vs. la v. dependiente
. twoway (scatter residuo gasto)
```

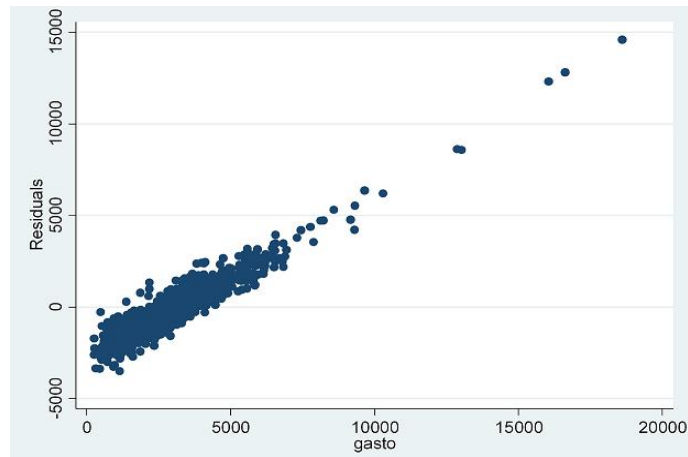


Figura 8.2: Método Gráfico (2) - Heteroscedasticidad

Lo que se observa en el gráfico anterior es que existe una relación positiva muy marcada entre la variable dependiente. Si no existiría heteroscedasticidad, se esperaría que el gráfico anterior sean constantes los residuos para cualquier observación o dato de la variable dependiente. En conclusión, los gráficos anteriores nos dan indicios de la existencia de heteroscedasticidad. Sin embargo, las pruebas gráficas serán insuficientes en la medida en que muestren la presencia de heteroscedasticidad en una variable en particular, ya que no detectan si esta se origina por la combinación lineal de todas o de algunas de las variables incluidas en el modelo (en este caso solo existe una regresora). Del mismo modo anterior, ustedes pueden analizar usando los siguientes comandos los patrones de la heteroscedasticidad si se gráfica los residuos estimados al cuadrado con la variable regresora y la variable dependiente, así:

```
. *Se genera el residuo al cuadrado
. g sqresiduo=residuo*residuo

. *residuo al cuadrado vs. la v.regresora
. twoway (scatter sqresiduo lingpc)

. *residuo al cuadrado vs. la v.dependiente
. twoway (scatter sqresiduo gasto)
```

```
. *Tambien se puede plotiar los residuos vs. los valores predecidos
. *de la regresión e incluyendo un linea en los valores 0:
. rvfplot, yline(0)
```

### 8.2.2. Método Formal

A continuación se realizan las principales pruebas formales, de las cuales es relevante mencionar que las pruebas de Breusch-Pagan-Godfrey (BPG) y White se suele utilizar cuando la muestra es grande (30 observaciones o más).

#### Prueba de Glejser

Para realizar esta prueba, es necesario instalar previamente el comando **lmhgl**. Dicho comando realiza en primer lugar la estimación por MCO y luego procede a realizar el test del multiplicador Langragiano de Glejser. A continuación se muestran los comandos y los resultados obtenidos:

```
. *GLESJER TEST
. *****

. *Buscando el comando lmhgl, que sirve para realizar la prueba de Glesjer
. findit lmhgl

. *Realizando la prueba de Glesjer
. lmhgl gasto lingpc

. *Realizando la prueba de Glesjer
. lmhgl gasto lingpc
```

Source	SS	df	MS	Number of obs = 1247		
Model	382840746	1	382840746	F( 1, 1245) = 162.20		
Residual	2.9385e+09	1245	2360270.42	Prob > F = 0.0000		
				R-squared = 0.1153		
				Adj R-squared = 0.1146		
Total	3.3214e+09	1246	2665631.95	Root MSE = 1536.3		

gasto	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
lingpc	1568.721	123.1736	12.74	0.000	1327.07	1810.372
_cons	-3159.17	500.1065	-6.32	0.000	-4140.314	-2178.025



```

=====
* OLS Glejser Lagrange Multiplier Heteroscedasticity Test
=====
Ho: No Heteroscedasticity - Ha: Heteroscedasticity
    Glejser LM Test          =    78.32800
    Degrees of Freedom       =         1.0
    P-Value > Chi2(1)        =     0.00000

```

En este caso se rechaza la hipótesis nula de homoscedasticidad ya que la probabilidad es menor a 0.05 y por lo tanto concluimos que existe Heteroscedasticidad.

### Prueba de Breusch-Pagan-Godfrey

La hipótesis nula se refiere a homoscedasticidad en los datos mientras que la alternativa se refiere a que los datos son heteroscedásticos. Teóricamente, la prueba de Breusch-Pagan-Godfrey se desarrolla de la siguiente manera:

$$\Theta = (1/2)(SCE) \sim X_{(m-1)}, (m-1) \text{ grados de libertad}$$

Aquí es importante aclarar que la *SCE* fue obtenido de la regresión la varianza del residuo (ajustado por la suma de residuos al cuadrado y el tamaño de la muestra) y la variable independiente. El residuo fue obtenido previamente de la regresión original. A continuación se programa la ecuación (5) y posteriormente se obtendrá el mismo resultado de una manera más fácil:

```

. *BPG - PROGRAMACION 1
. *****

. *Estimo la ecuación original
. reg gasto lingpc

```

Source	SS	df	MS
Model	382840746	1	382840746
Residual	2.9385e+09	1245	2360270.42
Total	3.3214e+09	1246	2665631.95

Number of obs = 1247  
 F( 1, 1245) = 162.20  
 Prob > F = 0.0000  
 R-squared = 0.1153  
 Adj R-squared = 0.1146  
 Root MSE = 1536.3

gasto	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
lingpc	1568.721	123.1736	12.74	0.000	1327.07 1810.372
_cons	-3159.17	500.1065	-6.32	0.000	-4140.314 -2178.025

```

. *Genero los residuos de la ecuación anterior
. predict e , resid

```

```
. *Genero la varianza del residuo ajustado por la suma
. *del residuo al cuadrado y el tamaño de la muestra
```

```
. g e2=e^2/(e(rss)/e(N))
```

```
. *Regresionar la varianza del residuo vs la variable independiente
```

```
. reg e2 lingpc
```

Source	SS	df	MS	Number of obs = 1247		
Model	394.671988	1	394.671988	F( 1, 1245) = 23.98		
Residual	20490.3016	1245	16.4580736	Prob > F = 0.0000		
				R-squared = 0.0189		
				Adj R-squared = 0.0181		
Total	20884.9736	1246	16.7616161	Root MSE = 4.0569		
e2	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
lingpc	1.592776	.3252566	4.90	0.000	.9546647	2.230888
_cons	-5.442438	1.3206	-4.12	0.000	-8.033284	-2.851591

```
. *Se computa el estadístico Chi2 a través de la suma de cuadrados
. *explicados de la regresión anterior
```

```
. display "Chi square(1)=" e(mss)/2
Chi square(1)=197.33599
```

```
. *Obtengo la probabilidad del estadístico Chi2
```

```
. display "prob<chi2=" chi2tail(1,e(mss)/2)
prob<chi2=7.965e-45
```

```
. *Lo anterior se puede obtener usando la prueba de BPG/COOK-WEISBERG
```

```
. reg gasto lingpc
```

Source	SS	df	MS	Number of obs = 1247		
Model	382840746	1	382840746	F( 1, 1245) = 162.20		
Residual	2.9385e+09	1245	2360270.42	Prob > F = 0.0000		
				R-squared = 0.1153		
				Adj R-squared = 0.1146		
Total	3.3214e+09	1246	2665631.95	Root MSE = 1536.3		
gasto	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
lingpc	1568.721	123.1736	12.74	0.000	1327.07	1810.372
_cons	-3159.17	500.1065	-6.32	0.000	-4140.314	-2178.025

```
. estat hettest
```

Breusch-Pagan / Cook-Weisberg test for heteroskedasticity

Ho: Constant variance

Variables: fitted values of gasto

chi2(1) = 197.34

Prob > chi2 = 0.0000

```
. * 0 de otra manera

. estat hettest lingpc

Breusch-Pagan / Cook-Weisberg test for heteroskedasticity
Ho: Constant variance
Variables: lingpc
chi2(1)      =   197.34
Prob > chi2   =   0.0000
```

Muchas veces, la ecuación anterior se puede plantiar apartir del número de observaciones y el  $R^2$  de la regresión entre el residuo al cuadrado y la variable regresora, así tenemos:

```
. *BPG - PROGRAMACION 2
. *****

. *Estimo por MCO el error al cuadrado y la v.explicativa

. reg sqresiduo lingpc
```

Source	SS	df	MS			
Model	2.1916e+15	1	2.1916e+15	Number of obs = 1247		
Residual	1.1378e+17	1245	9.1392e+13	F( 1, 1245) = 23.98		
Total	1.1597e+17	1246	9.3078e+13	Prob > F = 0.0000		
				R-squared = 0.0189		
				Adj R-squared = 0.0181		
				Root MSE = 9.6e+06		

sqresiduo	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
lingpc	3753353	766462.3	4.90	0.000	2249653	5257054
_cons	-1.28e+07	3111973	-4.12	0.000	-1.89e+07	-6719732

```
. *genero un scalar que es la multiplicacion de las observaciones
. *por el r2 de la regresion anterior

. scalar Nr2=e(N)*e(r2)

. *construyo el pvalue que se distribuye con una Chi2
. *con un grado de libertad (no se considera el intercepto) y el valor Nr2

. scalar pvalue=chi2tail(1,Nr2)

. *Aqui se muestra el Nr2 y su probabilidad

. scalar list Nr2 pvalue
      Nr2 = 23.565075
      pvalue = 1.208e-06

. *La probabilidad es menor a 0.05 y existe heteroscedasticidad

. *Al igual que lo anterior

. reg gasto lingpc
```

Source	SS	df	MS	Number of obs = 1247		
Model	382840746	1	382840746	F( 1, 1245) = 162.20		
Residual	2.9385e+09	1245	2360270.42	Prob > F = 0.0000		
				R-squared = 0.1153		
				Adj R-squared = 0.1146		
Total	3.3214e+09	1246	2665631.95	Root MSE = 1536.3		

gasto	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
lingpc	1568.721	123.1736	12.74	0.000	1327.07	1810.372
_cons	-3159.17	500.1065	-6.32	0.000	-4140.314	-2178.025

```
. estat hettest, iid
```

```
Breusch-Pagan / Cook-Weisberg test for heteroskedasticity
Ho: Constant variance
Variables: fitted values of gasto
chi2(1)      =    23.57
Prob > chi2   =    0.0000
```

En este último caso, la opción iid sirve para estimar la ecuación (6) al igual que lo desarrollado en la programación anterior. Se concluye en todos los casos que existe heteroscedasticidad pues la probabilidad es menor a 0.05 (se rechaza la hipótesis nula). Opcionalmente, se puede plantiar la prueba de BPG usando el test F:

```
. *Utilizando la prueba F
. reg gasto lingpc
```

Source	SS	df	MS	Number of obs = 1247		
Model	382840746	1	382840746	F( 1, 1245) = 162.20		
Residual	2.9385e+09	1245	2360270.42	Prob > F = 0.0000		
				R-squared = 0.1153		
				Adj R-squared = 0.1146		
Total	3.3214e+09	1246	2665631.95	Root MSE = 1536.3		

gasto	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
lingpc	1568.721	123.1736	12.74	0.000	1327.07	1810.372
_cons	-3159.17	500.1065	-6.32	0.000	-4140.314	-2178.025

```
. estat hettest,fstat
```

```
Breusch-Pagan / Cook-Weisberg test for heteroskedasticity
Ho: Constant variance
Variables: fitted values of gasto
F(1, 1245)   =    23.98
Prob > F      =    0.0000
```

### Prueba de Goldfeld-Quandt

Este test es una prueba eficaz cuando se sospecha la presencia de heteroscedasticidad en una variable específica. Esta prueba permite determinar claramente si el problema existe o no en los datos con los que se está trabajando. Las hipótesis con las que trabaja esta prueba son:

$$H_0 : \sigma_i^2 = \sigma^2, H_a : \sigma_i^2 \neq \sigma^2$$

Una vez que se detecta la variable que causa heteroscedasticidad, se deben ordenar las observaciones de tal manera que se pueda a continuación eliminar las  $c$  observaciones centrales de modo que representen  $1/3$  del total. Se realizan entonces dos regresiones con las observaciones de los extremos, considerando un estadístico  $F$  tal que:

$$F = \frac{SCE2}{SCE1}$$

Donde  $SCE1$  representa la suma de cuadrados del error de la primera regresión que se realizó con las observaciones de valores bajos, y  $SCE2$  la suma de cuadrados del error de la segunda regresión realizada con los valores altos. Este estadístico tiene  $(n - c - 2k)/2$  grados de libertad. Dicho lo anterior, se plantea la solución en STATA:

```
. **PRUEBA GOLDFELD-QUANDT
. *****

. *Ordenamos la variable que esta generando heteroscedasticidad

. sort lingpc

. *Dado el orden anterior, se genera una variable llamada index la
. *cual es igual enumera de 1 a 1 todas las observaciones

. gen index=_n

. *Regresiono las primeras 416 observaciones

. reg gasto lingpc if index < 417
```

Source	SS	df	MS	Number of obs = 416		
Model	28822268.6	1	28822268.6	F( 1, 414) = 24.08		
Residual	495563884	414	1197014.21	Prob > F = 0.0000		
				R-squared = 0.0550		
				Adj R-squared = 0.0527		
Total	524386153	415	1263581.09	Root MSE = 1094.1		

gasto	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
lingpc	1094.652	223.0805	4.91	0.000	656.1403	1533.164
_cons	-1338.643	816.0745	-1.64	0.102	-2942.809	265.5232

```
. *Calculo de la suma de cuadrados del error de la primer regresion
```

```
. scalar sce1=e(rmse)
```

```
. scalar list sce1
      sce1 = 1094.0814
```

```
. *Regresion las ultimas 831 observaciones
```

```
. reg gasto lingpc if index > 830
```

Source	SS	df	MS	Number of obs = 417		
Model	197054192	1	197054192	F( 1, 415) = 51.37		
Residual	1.5918e+09	415	3835607.21	Prob > F = 0.0000		
				R-squared = 0.1102		
				Adj R-squared = 0.1080		
Total	1.7888e+09	416	4300074.96	Root MSE = 1958.5		

gasto	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
lingpc	3939.434	549.6143	7.17	0.000	2859.059	5019.809
_cons	-13549.91	2416.601	-5.61	0.000	-18300.21	-8799.601

```
. *Calculo de la suma de cuadrados del error de la segunda regresion
```

```
. scalar sce2=e(rmse)
```

```
. scalar list sce2
      sce2 = 1958.4706
```

```
. *Se calcula el F calculado
```

```
. scalar r=rss2/rss1
```

```
. *Se calcula el F critico
```

```
. scalar f=invfprob(416,416, .05)
```

```
. scalar list f
      f = 1.1752439
```

```
. *Dado que r>f se rechaza la hipotesis nula de homoscedasticidad
```

## Prueba de White

La prueba White es la prueba más general comparada con las anteriores. Esta prueba es parecida a la de Breush –Pagan y su ecuación (6). En efecto, se procede a programarlo en STATA y algunas formas opcionales para el cálculo:

```
. *TEST DE WHITE
. *****

. *genero el cuadrado de la variable regresora
. gen lingpc2=lingpc^2

. *Regresiono el residuo al cuadrado y la variable regresora y el cuadratico de la misma
. *Aqui se estima un modelo de regresion con terminos cruzados
. reg sqresiduo lingpc lingpc2
```

Source	SS	df	MS	Number of obs = 1247		
Model	3.3778e+15	2	1.6889e+15	F( 2, 1244) = 18.66		
Residual	1.1260e+17	1244	9.0512e+13	Prob > F = 0.0000		
				R-squared = 0.0291		
				Adj R-squared = 0.0276		
Total	1.1597e+17	1246	9.3078e+13	Root MSE = 9.5e+06		

sqresiduo	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
lingpc	-3.45e+07	1.06e+07	-3.26	0.001	-5.53e+07	-1.37e+07
lingpc2	4836129	1335905	3.62	0.000	2215254	7457004
_cons	6.22e+07	2.09e+07	2.97	0.003	2.11e+07	1.03e+08

```
. *Genero el estadistico de White
. scalar white=e(N)*e(r2)

. *Genero la probabilidad del estadistico de White
. scalar pvalue =chi2tail(2,white)

. scalar list white pvalue
      white = 36.319299
      pvalue = 1.298e-08

. *Se rechaza la hipotesis nula y por tanto existe heteroscedasticidad

. *Otra forma usando el test de Cameron & Trivedi
. quietly reg gasto lingpc

. estat imtest, white

White's test for Ho: homoskedasticity
      against Ha: unrestricted heteroskedasticity
      chi2(2)      =      36.32
      Prob > chi2   =      0.0000

Cameron & Trivedi's decomposition of IM-test
```

Source	chi2	df	p
Heteroskedasticity	36.32	2	0.0000
Skewness	13.82	1	0.0002
Kurtosis	3.60	1	0.0579
Total	53.74	4	0.0000

. \*Otra forma es usar el comando Whitetst

. reg gasto lingpc

Source	SS	df	MS	Number of obs =	1247
Model	382840746	1	382840746	F( 1, 1245) =	162.20
Residual	2.9385e+09	1245	2360270.42	Prob > F =	0.0000
Total	3.3214e+09	1246	2665631.95	R-squared =	0.1153
				Adj R-squared =	0.1146
				Root MSE =	1536.3

gasto	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
lingpc	1568.721	123.1736	12.74	0.000	1327.07 1810.372
_cons	-3159.17	500.1065	-6.32	0.000	-4140.314 -2178.025

. whitetst

White's general test statistic : 36.31932 Chi-sq( 2) P-value = 1.3e-08

. \*Nota: Si no tienes instalado algun comando, buscalo con el comando findit e instalalo

### 8.3. Medidas Correctivas

Dado que en las pruebas formales e informales se demuestra la existencia de heteroscedasticidad, se procede ahora a utilizar diferentes métodos para la corrección de la misma. En Stata se hará énfasis en lo que realiza cada uno de los procedimientos y posteriormente se comparará las ecuaciones para su análisis:

```
. ** MEDIDAS CORRECTIVAS
. *****

. *Stata estima por MCO y corrige la heteroscedasticidad
. *usando el estimador robusto de varianzas y covarianzas

. reg gasto lingpc ,vce(robust)
```



Linear regression

Number of obs = 1247  
 F( 1, 1245) = 117.38  
 Prob > F = 0.0000  
 R-squared = 0.1153  
 Root MSE = 1536.3

gasto	Coef.	Robust Std. Err.	t	P> t	[95% Conf. Interval]	
lingpc	1568.721	144.7908	10.83	0.000	1284.66	1852.782
_cons	-3159.17	566.1164	-5.58	0.000	-4269.817	-2048.522

. \*Guardo la ecuación anterior en el sistema de Stata  
 . estimates store eq01

. \*Alternativamente se puede estimar heteroscedasticity robust covariance  
 . \*using a nonparametric bootstrap.

. reg gasto lingpc, vce(bootstrap, rep(100))

(running regress on estimation sample)

Bootstrap replications (100)

```

_____ 1 _____ 2 _____ 3 _____ 4 _____ 5
..... 50
..... 100

```

Linear regression

Number of obs = 1247  
 Replications = 100  
 Wald chi2(1) = 114.69  
 Prob > chi2 = 0.0000  
 R-squared = 0.1153  
 Adj R-squared = 0.1146  
 Root MSE = 1536.3172

gasto	Observed Coef.	Bootstrap Std. Err.	z	P> z	Normal-based [95% Conf. Interval]	
lingpc	1568.721	146.482	10.71	0.000	1281.622	1855.82
_cons	-3159.17	577.9106	-5.47	0.000	-4291.854	-2026.486

. \*Guardo la ecuación anterior en el sistema de Stata  
 . estimates store eq02

. \*Realizamos una regresión por MCP utilizando como variable ponderadora a 1/lingpc

. reg gasto lingpc [aweight=1/lingpc]  
 (sum of wgt is 3.1084e+02)

Source	SS	df	MS			
Model	367558061	1	367558061	Number of obs =	1247	
Residual	2.7994e+09	1245	2248520.07	F( 1, 1245) =	163.47	
Total	3.1670e+09	1246	2541705.9	Prob > F =	0.0000	
				R-squared =	0.1161	
				Adj R-squared =	0.1154	
				Root MSE =	1499.5	
gasto	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
lingpc	1491.048	116.6211	12.79	0.000	1262.253	1719.844
_cons	-2845	469.7762	-6.06	0.000	-3766.64	-1923.359

```
. *Guardo la ecuación anterior en el sistema de Stata
. estimates store eq03
```

```
. *otra opcion es estimar la varianza del error y reestimar el modelo por
. *Mínimos Cuadrados Generalizados
```

```
. reg gasto lingpc
```

Source	SS	df	MS	Number of obs = 1247		
Model	382840746	1	382840746	F( 1, 1245) = 162.20		
Residual	2.9385e+09	1245	2360270.42	Prob > F = 0.0000		
				R-squared = 0.1153		
				Adj R-squared = 0.1146		
Total	3.3214e+09	1246	2665631.95	Root MSE = 1536.3		

gasto	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
lingpc	1568.721	123.1736	12.74	0.000	1327.07	1810.372
_cons	-3159.17	500.1065	-6.32	0.000	-4140.314	-2178.025

```
. *captura el residuo
```

```
. predict resid, residual
```

```
. *genero el logaritmo del residuo al cuadrado
```

```
. g logresid2=log(resid^2)
```

```
. *Estimamos el siguiente modelo
```

```
. reg logresid2 lingpc
```

Source	SS	df	MS	Number of obs = 1247		
Model	198.860412	1	198.860412	F( 1, 1245) = 40.34		
Residual	6137.87126	1245	4.93001708	Prob > F = 0.0000		
				R-squared = 0.0314		
				Adj R-squared = 0.0306		
Total	6336.73168	1246	5.08565945	Root MSE = 2.2204		

logresid2	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
lingpc	1.130605	.1780167	6.35	0.000	.7813588	1.479851
_cons	8.525507	.7227796	11.80	0.000	7.107506	9.943507

```
. *Predecimos la varianza del error llamada zd
```

```
. predict zd
(option xb assumed; fitted values)
```

```
. *Como esta en logaritmos la linealizamos y se genera w
```

```
. g w=exp(zd)
```

```
. *Utilizando la variable 1/w como ponderadora
```

```
. reg gasto lingpc [aweight=1/w]
(sum of wgt is 2.7838e-03)
```

Source	SS	df	MS			
Model	329137090	1	329137090			
Residual	2.3482e+09	1245	1886110.05			
Total	2.6773e+09	1246	2148751.29			

Number of obs =	1247
F( 1, 1245) =	174.51
Prob > F	= 0.0000
R-squared	= 0.1229
Adj R-squared	= 0.1222
Root MSE	= 1373.4

gasto	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
lingpc	1286.329	97.37492	13.21	0.000	1095.292	1477.366
_cons	-2036.256	380.3819	-5.35	0.000	-2782.517	-1289.996

```
. *Guardo la ecuación anterior en el sistema de Stata
. estimates store eq04
```

Luego comparamos las ecuaciones anteriores y procedemos al análisis:

```
. *Regresionando el modelo original con heteroscedasticidad
. reg gasto lingpc
```

Source	SS	df	MS			
Model	382840746	1	382840746			
Residual	2.9385e+09	1245	2360270.42			
Total	3.3214e+09	1246	2665631.95			

Number of obs =	1247
F( 1, 1245) =	162.20
Prob > F	= 0.0000
R-squared	= 0.1153
Adj R-squared	= 0.1146
Root MSE	= 1536.3

gasto	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
lingpc	1568.721	123.1736	12.74	0.000	1327.07	1810.372
_cons	-3159.17	500.1065	-6.32	0.000	-4140.314	-2178.025

```
. *Guardo la ecuación anterior en el sistema de Stata
. estimates store original
```

```
. *Resumen
. estimates table original eq01 eq02 eq03 eq04, b(%7.2f) se(%7.2f) p(%7.2f) stats(N r2_a aic bic)
```

Variable	origi-1	eq01	eq02	eq03	eq04
lingpc	1568.72	1568.72	1568.72	1491.05	1286.33
	123.17	144.79	146.48	116.62	97.37
	0.00	0.00	0.00	0.00	0.00
_cons	-3159.17	-3159.17	-3159.17	-2845.00	-2036.26
	500.11	566.12	577.91	469.78	380.38
	0.00	0.00	0.00	0.00	0.00
N	1247	1247	1247	1247	1247
r2_a	0.11	0.11	0.11	0.12	0.12
aic	21839.67	21839.67	21839.67	21779.18	21560.01
bic	21849.92	21849.92	21849.92	21789.44	21570.27

legend: b/se/p

Aquí claramente se observa que la ecuación **eq04** tiene menor error estándar y posee los criterios Akaike (aic) y Schwarz (bic) más bajos en comparación con el resto de estimaciones. Nótese ahora que las ecuaciones **eq03** y **eq04** tienen diferentes coeficientes en comparación con las primeras tres ecuaciones, esto ocurre pues en **eq03** y **eq04** se construyen nuevas estimaciones pues se utilizaron diferentes ponderadores que alteran el modelo original. Caso contrario, ocurre cuando se estima **eq01** o **eq02** quienes corrigen heteroscedasticidad utilizando los errores estándar robustos de White y los errores estándar robustos mediante el procesor iterativo bootstrap respectivamente, lo cual en ambos casos se mantienen los coeficientes del modelo original (acuerdense que el problema de heteroscedasticidad es un problema de inferencia, los estimadores del modelo original siguen siendo MELI). Comparando los modelos, se puede concluir que la ecuación **eq04** es la mejor ecuación por poseer menores errores estándar aunque se alteran los coeficientes del modelo original. Por otro lado, si no se desea alterar los coeficientes de la ecuación original la mejor opción es la ecuación **eq02**.

## 8.4. Ejercicio Propuesto

En el archivo **emisiones.csv** se tiene datos para diferentes países en el año 2007, la cual pretende evaluar la implicancia del nivel de desarrollo sobre las emisiones de CO<sub>2</sub>. Aquí se tiene data sobre el dióxido de carbono (*CO2*) en miles de toneladas métricas (*TM*) y representa el nivel de generación de contaminantes. Dicha variable es explicada por la presión de la economía a través del producto bruto interno (*GDP*) en miles de millones de dólares constantes de 2005 y la población total (*POP*) en millones de habitantes. Por lo tanto, el modelo lineal a estimar es:

$$CO2_t = \alpha + \beta_1 GDP_t + \beta_2 POP_t + u_t$$

Dado el modelo a estimar, a continuación se procederá a evaluar la existencia de heteroscedasticidad y su posible corrección a dicho problema.



## Capítulo 9

# Autocorrelación

### 9.1. Problema de Autocorrelación

Tal como se comentó en el capítulo anterior, la tercera condición de Gauss-Markov implica que el término de error para cada observación se determina independientemente de los valores que pueda arrojar en el resto de observaciones de la muestra. Específicamente, la independencia de las perturbaciones implica que su covarianza es cero  $Cov(u_i u_j) = 0$  para todo  $i \neq j$ . Cuando esta condición no se cumple se dice que el error presenta autocorrelación.

Los problemas asociados a la presencia de autocorrelación son similares a los que enfrentamos cuando los errores son heteroscedásticos. Los estimadores MCO se mantienen insesgados pero dejan de ser eficientes. Esto implica que la varianza aumenta por lo que la volatilidad de los estimadores aumenta. Sin embargo, en términos de la estimación en la práctica ocurre lo contrario. Dado que los programas econométricos utilizan el estimador MCO, lo que ocurre es que calculan la varianza siguiendo la fórmula tradicional de MCO la cual nos da desviaciones estándar menores. Por tanto, éstas son usualmente subestimadas lo que conduce a una sobreestimación de los estadísticos-t y a problemas de inferencia dado que nuestras conclusiones serían erróneas.

Con referencia al problema asociado a la eficiencia de los estimadores MCO, y al igual que para el caso de heterocedasticidad, basta encontrar otro procedimiento para la estimación de los parámetros que arroje estimadores de menor varianza para descartar la eficiencia de los estimadores MCO. En este sentido, y como alternativa a la estimación MCO, la estimación por mínimos cuadrados generalizados arroja estimadores más eficientes en el sentido de presentar una menor varianza.

El problema de autocorrelación se da frecuentemente por los siguientes casos:

- Presencia de ciclos económicos.
- Presencia de relación no lineales.
- Mala especificación.

### Ejercicio

En el archivo **curva\_lm.dta** se tiene información trimestral desde 1990 hasta el tercer trimestre del 2009 de las siguientes variables para la economía peruana: *m1* (saldos monetarios nominales), *lr* (tasa de interés por préstamos), *pr* (índice de precios, 2005=100) y *gdp* (producto bruto interno). Las variables *m1* y *gdp* están en millones de dólares. El modelo que vamos a estimar es la curva LM donde incluiremos el rezago del índice de precios (para darle dinámica al modelo), así:

$$\log m1_t = \beta_1 + \beta_2 \log(gdp_t) + \beta_3 lr_t + \beta_4 \Delta \log(pr_t) + e_t$$

Antes de proceder a estimar la ecuación (9), se muestran los pasos en STATA previos a la estimación:

```
. * AUTOCORRELACIÓN
. *****

. *Limpiamos la memoria
. clear
```

```

. *Identificamos la ruta donde se encuentra el archivo
. cd "D:\Econometria-Stata\autocorrelación"
D:\Econometria-Stata\autocorrelación

. *Abrimos el archivo de STATA (.dta)
. use curva_lm.dta

. *Generamos variables
. g logm1=log(m1)
. g loggdp=log(gdp)
. g logpr=log(pr)
. g dlogpr=d.logpr
(1 missing value generated)

. *Declaramos al STATA que los datos son series de tiempo
. tsset year
      time variable:  year, 1 to 79
              delta:  1 unit

```

Luego de generar las variables relevantes para estimar la ecuación (9), procedemos a estimar la curva LM para la economía peruana:

```

. *Estimacion de la Curva LM

. reg logm1 loggdp lr dlogpr

```

Source	SS	df	MS	Number of obs = 78		
Model	97.6242315	3	32.5414105	F( 3, 74) = 859.35		
Residual	2.80219288	74	.037867471	Prob > F = 0.0000		
Total	100.426424	77	1.30423928	R-squared = 0.9721		
				Adj R-squared = 0.9710		
				Root MSE = .1946		
logm1	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
loggdp	1.503526	.0463801	32.42	0.000	1.411112	1.59594
lr	-.0001572	.0000917	-1.71	0.091	-.0003399	.0000255
dlogpr	1.308753	.5190613	2.52	0.014	.2745007	2.343005
_cons	-6.449451	.5015436	-12.86	0.000	-7.448798	-5.450103

## 9.2. Test de Autocorrelación

### 9.2.1. Método Informal (Método Gráfico)

En este caso, se mostrará diferentes comandos para analizar gráficamente la presencia de autocorrelación:



```
. *Test de Autocorrelación
. *****

. *Método Informal (Método Gráfico)
. *****

. *Aquí se gráfica los residuos con los valores predichos.

. rvfplot
```

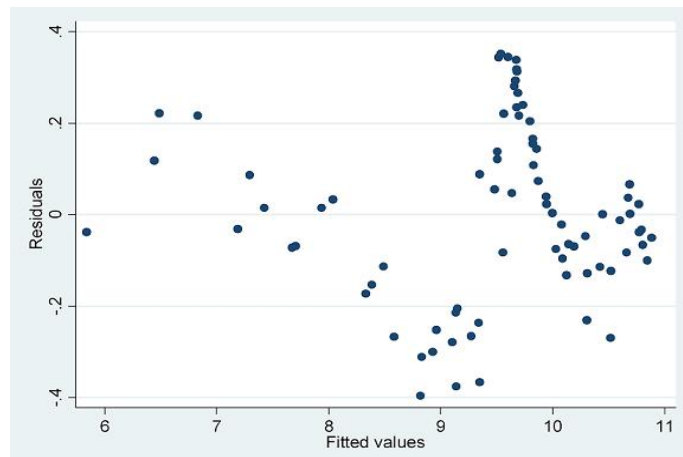


Figura 9.1: Método Gráfico (1) - Autocorrelación

```
. *Capturo el residuo de la regresion anterior y la llamo "res"
. predict res,r
(1 missing value generated)

. *Grafico lineal del residuo "res"
. line res year

. *Alternativamente se muestra el grafico de tipo scatter
. scatter res year

. *Genero el rezago del residuo "res"
. g lres=res[_n-1]
(2 missing values generated)

. *Ploteo el residuo "res" vs su rezago
. list res lres

. *Alternativamente se muestra el grafico de tipo scatter
. scatter res lres
```

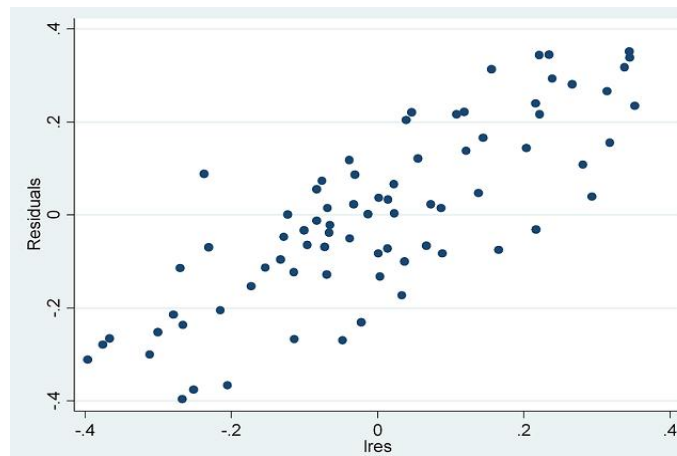


Figura 9.2: Método Gráfico (2) - Autocorrelación

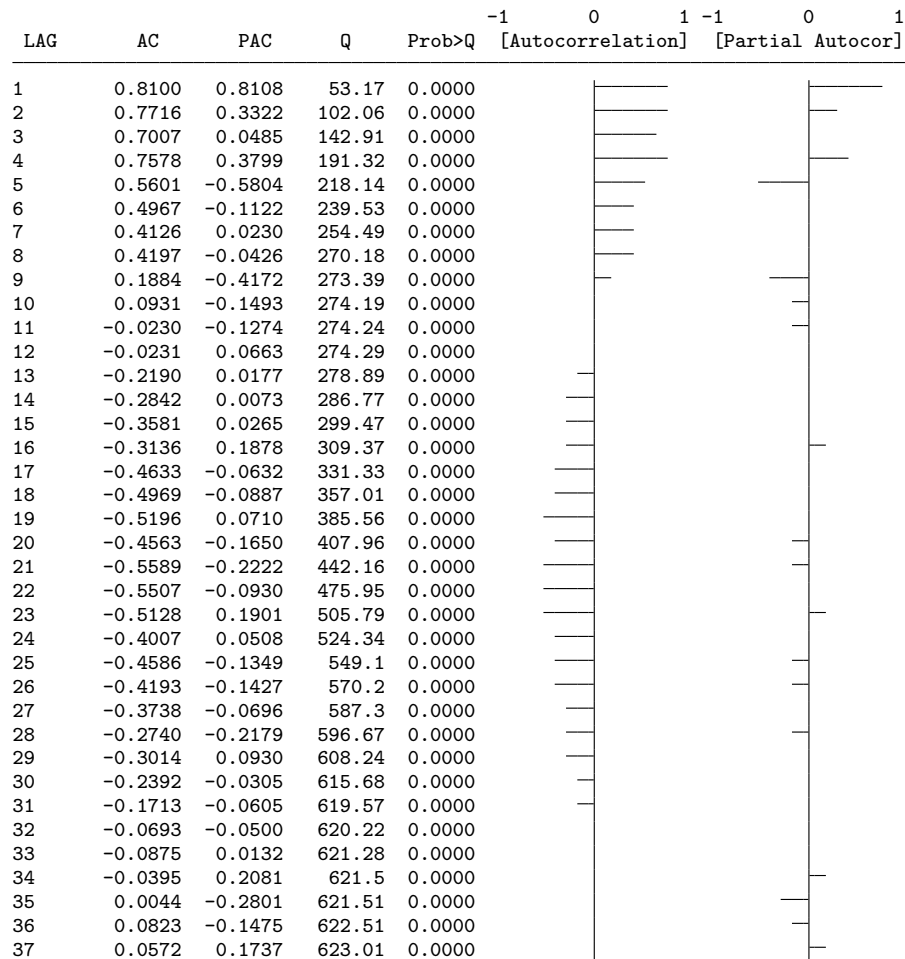
El gráfico anterior nos indica la posible presencia de autocorrelación positiva de grado uno. Se puede medir dicha relación utilizando el grado de correlación entre el residuo y su rezago, la cual se muestra a continuación:

```
. *Calculo el grado de relacion del residuo "res" y su rezago
. corr res lres
(obs=77)
```

	res	lres
res	1.0000	
lres	0.8106	1.0000

También se puede mostrar el correlograma de los residuos:

```
. *Se analiza el correlograma de los residuos
. corrgram res
```



En el correlograma tenemos dos columnas una referida a la autocorrelación y la otra referida a la autocorrelación parcial. Empecemos por la segunda. Tal como se aprecia debajo de dicha columna unas líneas nos indican la magnitud del coeficiente de autocorrelación correspondiente al máximo rezago incluido en la ecuación estimada para cada fila del cuadro de la derecha. Como vemos en cada regresión se va incluyendo un rezago más (y por tanto un parámetro más por estimar). No se incluye un intercepto porque la media de los errores MCO por construcción es cero. Bajo esta perspectiva, cada parámetro que se calcula es el coeficiente de correlación del error contemporáneo con el rezago respectivo. El último parámetro de cada ecuación nos mide la correlación del respectivo rezago con el valor contemporáneo del error. Ese valor es que se registra en la columna de autocorrelación parcial.

La interpretación del gráfico es entonces que cuando las líneas caen dentro del intervalo se puede esperar que los coeficientes de correlación parcial sean estadísticamente iguales a cero. Si dichas líneas salen fuera de la banda se espera que sean diferentes de cero. Como se observa en el gráfico, al parecer el primer rezago sale fuera de la banda de confianza, mientras que los demás no lo hacen. La interpretación es entonces que sólo podría haber autocorrelación de primer orden. En la columna de autocorrelación se registran los estadísticos tanto de Ljung-Box y su probabilidad. Este estadístico toma en cuenta los coeficientes de correlación. Por ello se habla de autocorrelación y no de autocorrelación parcial.

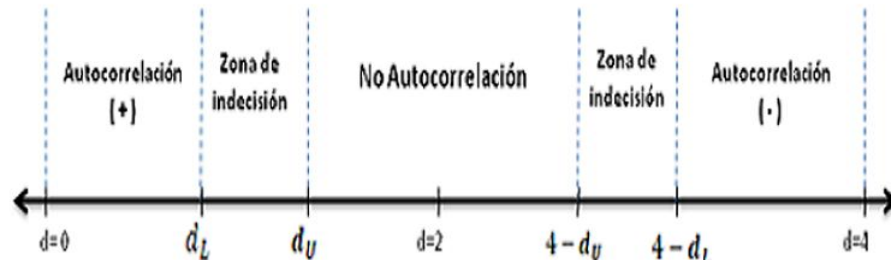
Si observamos la última columna de la tabla se presentan las probabilidades del estadístico consignado. Como nos debemos haber percatado, la hipótesis nula de la prueba es que no existe autocorrelación. Demos una mirada con detenimiento. Si vemos los valores de la probabilidad para cada fila, veremos que en cada una de ellas se rechaza la hipótesis nula. Esto nos llevaría a pensar que incluso tenemos una autocorrelación autorregresiva de orden 37. Esa es una conclusión errónea. Revisando nuestras pruebas, vemos que la hipótesis nula es que no existe autocorrelación de ningún orden. Se utiliza para calcular el estadístico todas las correlaciones parciales. En nuestro caso, aparentemente la primera es distinta de cero, el estadístico será grande a pesar de que las demás sean cercanas a cero. Ello explica las bajas probabilidades observadas. Por lo tanto, vemos que los estadísticos de Ljung-Box sólo pueden detectar la autocorrelación pero no indican el orden de ésta. Por tanto su interpretación debe ser comparada con los gráficos del correlograma para detectar posibles patrones de autocorrelación. En todo caso no son definitivos sino sólo referenciales.

### 9.2.2. Método Formal

#### Prueba de Durbin Watson

Posterior a la estimación de la ecuación (9) y según el criterio de decisión mostrado a continuación, se puede testear la existencia de **autocorrelación solo de primer orden**. Si bien se requeriría la tabla estadística de Savin-White para determinar el límite inferior ( $d_L$ ) y superior ( $d_U$ ) y analizar la existencia de autocorrelación de primer orden, existe una regla práctica. Si el estadístico Durbin-Watson

se aproxima a 0 entonces existe autocorrelación positiva de orden 1 y si por el contrario dicho estadístico tiende a 4 existe autocorrelación negativa de orden 1. Por último, si el estadístico Durbin-Watson se aproxima o es igual a 2, significa que la regresión estimada carace de problemas de autocorrelación.



A continuación se analiza dicho test y uno alternativo a través de STATA:

```
. *Estimación de la Curva LM
. reg logm1 loggdp lr dlogpr

. *Prueba de Durbin Watson
. dwstat

Durbin-Watson d-statistic( 4, 78) = .378536

. *DW indicates the presence of positive autocorrelation de grado uno
. *asymptotic test:
. di 1-normprob(78*.5*(1-.5*.378536))
4.028e-13
. * 4.028e-13, Clearly this test indicates autocorrelation.

. *Alternativa del test de durbin watson
. reg logm1 loggdp lr dlogpr

. estat durbinalt

Durbin's alternative test for autocorrelation
```

lags(p)	chi2	df	Prob > chi2
1	140.640	1	0.0000

```

H0: no serial correlation

. *Conclusión: Se rechaza la H0 y por ende existe autocorrelación positiva de orden 1
. *La ventaja del "estat durbinalt" es que pueden evaluar para diferentes rezagos
. *Por ejemplo, escribir: estat durbinalt, lags(2)

```

## Prueba Breusch Godfrey

A diferencia del test de Durbin-Watson, la prueba Breusch Godfrey permite evaluar si existe autocorrelación de orden uno o mas. Asimismo, esta prueba sirve tanto para modelos de regresión estáticos y dinámicos a diferencia del test de Durbin-Watson que solo sirve para modelos estáticos y de orden uno. Los comandos en STATA son:

```
. *Estimación de la Curva LM
. reg logm1 loggdp lr dlogpr

. *Test Breusch Godfrey: Prueba de Autocorrelación de orden 1 y 2
. bgodfrey, lags(1 2)
```

Breusch-Godfrey LM test for autocorrelation

lags(p)	chi2	df	Prob > chi2
1	51.348	1	0.0000
2	54.367	2	0.0000

H0: no serial correlation

```
. *Lo anterior se puede reafirmar de la sgte. manera

. reg logm1 loggdp lr dlogpr

. predict e, resid
(1 missing value generated)

. g e1=e[_n-1]
(2 missing values generated)

. g e2=e1[_n-1]
(3 missing values generated)

. *Estimando el residuo vs los rezagos y las v.explicativas
. reg e e1 e2 loggdp lr dlogpr
```

Source	SS	df	MS	Number of obs = 76		
Model	2.06611243	5	.413222486	F( 5, 70) = 40.14		
Residual	.720543543	70	.010293479	Prob > F = 0.0000		
Total	2.78665597	75	.037155413	R-squared = 0.7414		
				Adj R-squared = 0.7230		
				Root MSE = .10146		

e	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
e1	.4938558	.1066552	4.63	0.000	.2811387	.706573
e2	.3350593	.1043956	3.21	0.002	.1268488	.5432697
loggdp	-.0667521	.0358289	-1.86	0.067	-.1382105	.0047063
lr	.0007142	.0002189	3.26	0.002	.0002776	.0011508
dlogpr	-2.62366	.8358578	-3.14	0.002	-4.290726	-.9565947
_cons	.7388106	.3928109	1.88	0.064	-.0446259	1.522247

```

. *Probando si son significativos
. test e1 e2

( 1)  e1 = 0
( 2)  e2 = 0

          F( 2, 70) = 79.55
          Prob > F = 0.0000

. *Alternativamente se crea el Test Portmanteau

. wntestq e

Portmanteau test for white noise
-----
Portmanteau (Q) statistic = 623.0129
Prob > chi2(37)          = 0.0000

. wntestq e, lags(2)

Portmanteau test for white noise
-----
Portmanteau (Q) statistic = 102.0558
Prob > chi2(2)           = 0.0000

```

## 9.3. Medidas Correctivas

Dado que las pruebas formales e informales indican la existencia de autocorrelación, usando STATA se plantean las posibles soluciones y posteriormente se compara los modelos estimados:

### 9.3.1. Método de Estimación Prais-Winsten

```
. *Método de Estimación Prais-Winsten AR(1)
```

```
. prais logm1 loggdp lr dlogpr
```

```
Iteration 0: rho = 0.0000
Iteration 1: rho = 0.8108
Iteration 2: rho = 0.9285
Iteration 3: rho = 0.9795
Iteration 4: rho = 0.9867
Iteration 5: rho = 0.9885
Iteration 6: rho = 0.9891
Iteration 7: rho = 0.9894
Iteration 8: rho = 0.9895
Iteration 9: rho = 0.9895
Iteration 10: rho = 0.9895
Iteration 11: rho = 0.9895
Iteration 12: rho = 0.9895
Iteration 13: rho = 0.9895
```

```
Prais-Winsten AR(1) regression -- iterated estimates
```

Source	SS	df	MS	Number of obs = 78		
Model	.816242285	3	.272080762	F( 3, 74) = 45.06		
Residual	.446798088	74	.006037812	Prob > F = 0.0000		
				R-squared = 0.6463		
				Adj R-squared = 0.6319		
Total	1.26304037	77	.016403122	Root MSE = .0777		

logm1	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
loggdp	.5540982	.0995559	5.57	0.000	.3557288	.7524676
lr	.000013	.0000357	0.36	0.717	-.0000581	.0000841
dlogpr	-.3697259	.2588672	-1.43	0.157	-.88553	.1460782
_cons	3.446655	1.121369	3.07	0.003	1.212278	5.681032

rho	.9895021
-----	----------

```
Durbin-Watson statistic (original) 0.378536
Durbin-Watson statistic (transformed) 1.748292
```

```
. *Agregandole los errores estándar robustos
. prais logm1 loggdp lr dlogpr, vce(robust)
```

```
. prais logm1 loggdp lr dlogpr, r
```

```
Iteration 0: rho = 0.0000
Iteration 1: rho = 0.8108
Iteration 2: rho = 0.9285
Iteration 3: rho = 0.9795
Iteration 4: rho = 0.9867
Iteration 5: rho = 0.9885
Iteration 6: rho = 0.9891
Iteration 7: rho = 0.9894
Iteration 8: rho = 0.9895
Iteration 9: rho = 0.9895
```



```
Iteration 10: rho = 0.9895
Iteration 11: rho = 0.9895
Iteration 12: rho = 0.9895
Iteration 13: rho = 0.9895
```

Prais-Winsten AR(1) regression -- iterated estimates

```
Linear regression                                Number of obs =      78
                                                F( 4,      74) = 162.17
                                                Prob > F      = 0.0000
                                                R-squared     = 0.6463
                                                Root MSE     = .0777
```

logm1	Semirobust		t	P> t	[95% Conf. Interval]	
	Coef.	Std. Err.				
loggdp	.5540982	.1076855	5.15	0.000	.3395302	.7686661
lr	.000013	.0000431	0.30	0.764	-.0000729	.0000988
dlogpr	-.3697259	.3069739	-1.20	0.232	-.9813847	.241933
_cons	3.446655	1.658338	2.08	0.041	.1423445	6.750965
rho	.9895021					

```
Durbin-Watson statistic (original)    0.378536
Durbin-Watson statistic (transformed) 1.748292
```

### 9.3.2. Método de Estimación Cochrane-Orcutt

```
. *Método de Estimación Cochrane-Orcutt
```

```
. prais logm1 loggdp lr dlogpr, corc
```

```
Iteration 0: rho = 0.0000
Iteration 1: rho = 0.8108
Iteration 2: rho = 0.9388
Iteration 3: rho = 0.9497
Iteration 4: rho = 0.9547
Iteration 5: rho = 0.9575
Iteration 6: rho = 0.9593
Iteration 7: rho = 0.9605
Iteration 8: rho = 0.9612
Iteration 9: rho = 0.9618
Iteration 10: rho = 0.9622
Iteration 11: rho = 0.9625
Iteration 12: rho = 0.9627
Iteration 13: rho = 0.9628
Iteration 14: rho = 0.9629
Iteration 15: rho = 0.9630
Iteration 16: rho = 0.9630
Iteration 17: rho = 0.9631
Iteration 18: rho = 0.9631
Iteration 19: rho = 0.9631
Iteration 20: rho = 0.9631
Iteration 21: rho = 0.9632
Iteration 22: rho = 0.9632
Iteration 23: rho = 0.9632
Iteration 24: rho = 0.9632
```

```
Iteration 25: rho = 0.9632
Iteration 26: rho = 0.9632
Iteration 27: rho = 0.9632
Iteration 28: rho = 0.9632
Iteration 29: rho = 0.9632
Iteration 30: rho = 0.9632
```

Cochrane-Orcutt AR(1) regression -- iterated estimates

Source	SS	df	MS	Number of obs =	77
Model	.348385887	3	.116128629	F( 3, 73) =	31.99
Residual	.264970218	73	.003629729	Prob > F	= 0.0000
				R-squared	= 0.5680
				Adj R-squared	= 0.5502
Total	.613356105	76	.008070475	Root MSE	= .06025

logm1	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
loggdp	.1404915	.0983169	1.43	0.157	-.0554538	.3364368
lr	-.0000235	.0000283	-0.83	0.408	-.0000798	.0000328
dlogpr	-.1853531	.2026405	-0.91	0.363	-.5892151	.218509
_cons	9.2946	1.163146	7.99	0.000	6.976452	11.61275
rho	.96319					

```
Durbin-Watson statistic (original)    0.378536
Durbin-Watson statistic (transformed) 2.068840
```

```
. *Agregandole los errores estándar robustos
. prais logm1 loggdp lr dlogpr, corc r
```

```
Iteration 0: rho = 0.0000
Iteration 1: rho = 0.8108
Iteration 2: rho = 0.9388
Iteration 3: rho = 0.9497
Iteration 4: rho = 0.9547
Iteration 5: rho = 0.9575
Iteration 6: rho = 0.9593
Iteration 7: rho = 0.9605
Iteration 8: rho = 0.9612
Iteration 9: rho = 0.9618
Iteration 10: rho = 0.9622
Iteration 11: rho = 0.9625
Iteration 12: rho = 0.9627
Iteration 13: rho = 0.9628
Iteration 14: rho = 0.9629
Iteration 15: rho = 0.9630
Iteration 16: rho = 0.9630
Iteration 17: rho = 0.9631
Iteration 18: rho = 0.9631
Iteration 19: rho = 0.9631
Iteration 20: rho = 0.9631
Iteration 21: rho = 0.9632
Iteration 22: rho = 0.9632
Iteration 23: rho = 0.9632
Iteration 24: rho = 0.9632
Iteration 25: rho = 0.9632
Iteration 26: rho = 0.9632
Iteration 27: rho = 0.9632
Iteration 28: rho = 0.9632
```

```
Iteration 29: rho = 0.9632
Iteration 30: rho = 0.9632
```

```
Cochrane-Orcutt AR(1) regression -- iterated estimates
```

```
Linear regression                                Number of obs =      77
                                                F( 4,    73) = 2177.66
                                                Prob > F      = 0.0000
                                                R-squared     = 0.5680
                                                Root MSE     = .06025
```

logm1	Semirobust		t	P> t	[95% Conf. Interval]	
	Coef.	Std. Err.				
loggdp	.1404915	.0919931	1.53	0.131	-.0428505	.3238334
lr	-.0000235	.0000258	-0.91	0.365	-.0000748	.0000278
dlogpr	-.1853531	.1842933	-1.01	0.318	-.552649	.1819429
_cons	9.2946	1.058076	8.78	0.000	7.185858	11.40334
rho	.96319					

```
Durbin-Watson statistic (original)    0.378536
Durbin-Watson statistic (transformed) 2.068840
```

Nótese que usando la estimación por Prais-Winsten o Cochrane-Orcutt e incluso agregándole a dichas estimaciones los errores estándar robustos, las probabilidades individuales indican que las variables no son significativas con un  $R^2$  relativamente alto. Lo anterior puede ser indicios de que quizás se este omitiendo alguna variable relevante en el modelo. Probemos ahora incluyendo el rezago de la variable dependiente como regresora.

### 9.3.3. Estimación de Modelos Dinámicos

```
. *Genero el rezago de la variable dependiente

. g llogm1=l.logm1
(1 missing value generated)

. *Estimación de un modelo dinámico en la Curva LM

. reg logm1 loggdp lr dlogpr llogm1
```

Source	SS	df	MS	Number of obs = 78		
Model	100.096858	4	25.0242146	F( 4, 73) = 5542.95		
Residual	.329565937	73	.004514602	Prob > F = 0.0000		
				R-squared = 0.9967		
				Adj R-squared = 0.9965		
Total	100.426424	77	1.30423928	Root MSE = .06719		

logm1	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
loggdp	.2116228	.0574787	3.68	0.000	.097068	.3261776
lr	-.0000761	.0000318	-2.39	0.019	-.0001396	-.0000127
dlogpr	1.242146	.1792462	6.93	0.000	.8849088	1.599383
llogm1	.8677781	.0370799	23.40	0.000	.7938779	.9416783
_cons	-.9704093	.2912059	-3.33	0.001	-1.550782	-.3900367

```
. *Evaluo si se corrigio el problema de autocorrelación
. bgodfrey, lags(1 2)
```

Breusch-Godfrey LM test for autocorrelation

lags(p)	chi2	df	Prob > chi2
1	3.625	1	0.0569
2	4.276	2	0.1179

H0: no serial correlation

Al incluir el rezago de la variable dependiente como regresora se corrige el problema de la autocorrelación. Procedemos a evaluar diferentes estimaciones incluyendo el rezago de la variable dependiente:

```
. prais logm1 loggdp lr dlogpr llogm1

. *Guarda la ecuación anterior
. estimates store eq01

. prais logm1 loggdp lr dlogpr llogm1, vce(robust)

. *Guarda la ecuación anterior
. estimates store eq02
. prais logm1 loggdp lr dlogpr llogm1, corc

. *Guarda la ecuación anterior
. estimates store eq03

. prais logm1 loggdp lr dlogpr llogm1, corc vce(robust)

. *Guarda la ecuación anterior
. estimates store eq04

. *Resumen

. estimates table eq01 eq02 eq03 eq04, ///
. b(%7.2f) se(%7.2f) p(%7.2f) stats(N r2_a aic bic)
```

Variable	eq01	eq02	eq03	eq04
loggdp	0.17 0.05 0.00	0.17 0.05 0.00	0.15 0.04 0.00	0.15 0.04 0.00
lr	-0.00 0.00 0.12	-0.00 0.00 0.34	0.00 0.00 0.01	0.00 0.00 0.00
dlogpr	1.12 0.17 0.00	1.12 0.29 0.00	0.12 0.31 0.69	0.12 0.28 0.67
llogm1	0.89 0.03 0.00	0.89 0.03 0.00	0.89 0.03 0.00	0.89 0.02 0.00
_cons	-0.74 0.24 0.00	-0.74 0.26 0.01	-0.52 0.21 0.02	-0.52 0.18 0.00
N	78	78	77	77
r2_a	1.00	1.00	1.00	1.00
aic	-199.52	-199.52	-207.28	-207.28
bic	-187.73	-187.73	-195.56	-195.56

legend: b/se/p

### 9.3.4. Estimación de Modelos Dinámicos

El problema de la autocorrelación también se puede corregir utilizando los errores estandar robustos de Newey-West (HAC). Posteriormente se comparará con el resto de modelos:

```
. *Newey-HAC para máximo 2 rezagos
. newey logm1 loggdp lr dlogpr, lag(2)
Regression with Newey-West standard errors      Number of obs =      78
maximum lag: 2                                F( 3, 74) =    2397.46
                                              Prob > F      =     0.0000
```

logm1	Newey-West		t	P> t	[95% Conf. Interval]	
	Coef.	Std. Err.				
loggdp	1.503526	.0462225	32.53	0.000	1.411426	1.595626
lr	-.0001572	.0000787	-2.00	0.049	-.000314	-4.38e-07
dlogpr	1.308753	.4686707	2.79	0.007	.3749061	2.2426
_cons	-6.449451	.5200937	-12.40	0.000	-7.48576	-5.413141

```
. *Resumen
. estimates table eq01 eq02 eq03 eq04 eq05, ///
```

```
. b(%7.4f) se(%7.2f) p(%7.2f) stats(N r2_a aic bic)
```

Variable	eq01	eq02	eq03	eq04	eq05
loggdp	0.1658 0.05 0.00	0.1658 0.05 0.00	0.1548 0.04 0.00	0.1548 0.04 0.00	1.5035 0.05 0.00
lr	-0.0000 0.00 0.12	-0.0000 0.00 0.34	0.0002 0.00 0.01	0.0002 0.00 0.00	-0.0002 0.00 0.05
dlogpr	1.1194 0.17 0.00	1.1194 0.29 0.00	0.1225 0.31 0.69	0.1225 0.28 0.67	1.3088 0.47 0.01
llogm1	0.8950 0.03 0.00	0.8950 0.03 0.00	0.8853 0.03 0.00	0.8853 0.02 0.00	
_cons	-0.7406 0.24 0.00	-0.7406 0.26 0.01	-0.5206 0.21 0.02	-0.5206 0.18 0.00	-6.4495 0.52 0.00
N	78	78	77	77	78
r2_a	0.9982	0.9982	0.9984	0.9984	
aic	-2.0e+02	-2.0e+02	-2.1e+02	-2.1e+02	.
bic	-1.9e+02	-1.9e+02	-2.0e+02	-2.0e+02	.

legend: b/se/p

## 9.4. Ejercicio Propuesto

En el archivo **pbi.csv** se tiene información anual de 1950-2011 de las siguientes variables para la economía peruana: *pbi* (producto bruto interno), *cons* (consumo privado) y *inv* (inversión bruta fija). Todas las variables están en millones de soles 1994. El modelo a estimar es el siguiente:

$$pbi_t = \beta_1 + \beta_2 cons_t + \beta_3 inv_t + e_t$$

Posterior a la estimación, se le pide evaluar la presencia de autocorrelación y si este existe corregirlo de la mejor manera.



# Capítulo 10

## Multicolinealidad

### 10.1. Problema de Multicolinealidad

La colinealidad está referida a la existencia de una sola relación lineal entre las variables explicativas y, por lo tanto, la multicolinealidad se refiere a la existencia de más de una relación lineal. Es importante anotar que la multicolinealidad se refiere sólo a relaciones lineales entre las variables independientes y no a cualquier otro tipo de relación, así pues, si  $X_i = X_i^2$ , entonces existirá multicolinealidad en el modelo.

El problema de la multicolinealidad está definido por el alto grado de intercorrelación entre variables explicativas. Dentro de las violaciones de los supuestos del modelo lineal general, la multicolinealidad es un problema de grado y no teórico como la heterocedasticidad o autocorrelación, más aún, los estimadores obtenidos bajo multicolinealidad, conservan las propiedades que los definen como MELI.

Una cuestión importante que debe analizarse al estudiar los resultados de un modelo de regresión es el grado de relación lineal existente entre las observaciones de las variables explicativas. A este respecto, las posibles situaciones son tres:

- **Multicolinealidad Perfecta:** se da cuando existe una relación lineal exacta



entre algunos o todos los regresores incluidos en el modelo.

- **Ortogonalidad:** Supone la ausencia de relación lineal entre algunos o todos los regresores incluidos en el modelo (raramente ocurre esto).
- **Multicolinealidad Imperfecta:** consiste en la existencia de una relación lineal fuerte entre los regresores del modelo.

Las posibles fuentes de multicolinealidad son cuatro principalmente:

- El método de recolección de información empleado.
- Restricciones sobre el modelo o en la población que es objeto de muestreo.
- Especificación del modelo.
- Un modelo sobredeterminado (es cuando un modelo tiene mas variables explicativas que observaciones).

Las consecuencias del problema de multicolinealidad son las siguientes:

- Varianzas y covarianzas grandes.
- Intervalos de confianza más amplios.
- Estadísticos t poco significativos y un  $R^2$  alto.
- Sensibilidad de los estimadores y sus errores estándar ante pequeños cambios en la muestra.
- Transformación de variables del modelo.

Las posibles correcciones son:

- Suprimir variables.
- Empleo de información adicional.

- Método de primeras diferencias.
- Empleo de cocientes o ratios entre las variables.
- Aumentar el tamaño de muestra.
- No hacer nada.

### Ejercicio

En el archivo **demanda\_mineria.dta** se tiene información estadística de 1980-2010 para las siguiente variables: *pbimineria* y *pbimundial* en US\$ 94, cobre (Miles TMF), *plomo* (Miles TMF), *zinc* (Miles TMF), *oro* (Miles Oz), *plata* (Miles Oz), *hierro* (Miles TMF) y *estanho* (Miles TMF). Teniendo estas variables se pide estimar la demanda de minería según la siguiente ecuación:

$$\begin{aligned}
 pbimineria_t = & \beta_1 + \beta_2 pbimundial_t + \beta_3 cobre_t + \beta_4 plomo_t + \beta_5 zinc_t \\
 & + \beta_6 oro_t + \beta_7 plata_t + \beta_8 hierro_t + \beta_9 estanho_t + e_t
 \end{aligned}$$

## 10.2. Detección de Multicolinealidad

Luego de la estimación (10), se procede a realizar en STATA diferentes pruebas para detectar la multicolinealidad:

```

. * MULTICOLINEALIDAD
. *****

. clear

. cd "D:\Econometria-Stata\multicolinealidad"
D:\Econometria-Stata\multicolinealidad

. use demanda_mineria.dta

. reg pbi_mineria pbi_mundial cobre plomo zinc oro plata hierro estanho

```

Source	SS	df	MS	Number of obs = 31		
Model	3.7342e+19	8	4.6677e+18	F( 8, 22) = 9247.42		
Residual	1.1105e+16	22	5.0476e+14	Prob > F = 0.0000		
				R-squared = 0.9997		
				Adj R-squared = 0.9996		
Total	3.7353e+19	30	1.2451e+18	Root MSE = 2.2e+07		

pbi_mineria	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
pbi_mundial	2.78e-06	2.35e-06	1.18	0.249	-2.09e-06	7.66e-06
cobre	964011.9	93145.86	10.35	0.000	770839.2	1157185
plomo	988878.9	309207.1	3.20	0.004	347622.7	1630135
zinc	395228.5	74204.88	5.33	0.000	241337	549120
oro	206091.8	10727.13	19.21	0.000	183845.1	228338.5
plata	4957.275	976.594	5.08	0.000	2931.943	6982.607
hierro	29015.88	7308.849	3.97	0.001	13858.26	44173.51
estanho	-1265900	1509886	-0.84	0.411	-4397213	1865413
_cons	-6.78e+07	8.06e+07	-0.84	0.409	-2.35e+08	9.94e+07

```
. *Detección de Multicolinealidad
```

```
. * 1. High R2 but few significant t-ratios.
```

```
. * 2.
```

```
. *graph matrix pbi_mineria pbi_mundial cobre plomo zinc oro plata hierro estanho,  
. *half maxis(ylabel(none) xlabel(none))
```

```
. graph matrix pbi_mineria pbi_mundial cobre plomo zinc oro plata hierro estanho
```

```
. *3. Factor de inflación de la varianza.
```

```
. vif
```

Variable	VIF	1/VIF
cobre	56.44	0.017718
zinc	40.26	0.024840
oro	39.24	0.025482
estanho	33.93	0.029472
plata	33.01	0.030290
pbi_mundial	32.42	0.030840
plomo	13.08	0.076462
hierro	2.89	0.345988
Mean VIF	31.41	

```
. ** Interpretation: If a VIF is in excess of 20, or a tolerance (1/VIF) is .05 or less,
```

```
. *4. Matriz de correlación
```

```
. cor pbi_mineria pbi_mundial cobre plomo zinc oro plata hierro estanho  
(obs=31)
```

	pbi_mi-a	pbi_mu-l	cobre	plomo	zinc	oro	plata
pbi_mineria	1.0000						
pbi_mundial	0.9726	1.0000					
cobre	0.9761	0.9480	1.0000				
plomo	0.9278	0.8771	0.8633	1.0000			
zinc	0.9821	0.9652	0.9688	0.9150	1.0000		
oro	0.9758	0.9468	0.9086	0.9371	0.9395	1.0000	
plata	0.9710	0.9528	0.9773	0.8663	0.9634	0.9118	1.0000
hierro	0.6374	0.6037	0.7134	0.4955	0.6159	0.5330	0.6369
estanho	0.9166	0.9114	0.8231	0.9254	0.8923	0.9643	0.8297

	hierro	estanho
hierro	1.0000	
estanho	0.4429	1.0000

```
. *5to. Farrar-Glauber Multicollinearity Tests
```

```
. findit fgtest
```

```
. fgtest pbi_mineria pbi_mundial cobre plomo zinc oro plata hierro estanho
```

```
=====
* Farrar-Glauber Multicollinearity Tests
=====
```

Ho: No Multicollinearity - Ha: Multicollinearity

```
* (1) Farrar-Glauber Multicollinearity Chi2-Test:
    Chi2 Test = 481.7276    P-Value > Chi2(28) 0.0000
```

```
* (2) Farrar-Glauber Multicollinearity F-Test:
```

Variable	F_Test	DF1	DF2	P_Value
pbi_mund-l	103.253	23.000	7.000	0.000
cobre	182.157	23.000	7.000	0.000
plomo	39.686	23.000	7.000	0.000
zinc	128.988	23.000	7.000	0.000
oro	125.657	23.000	7.000	0.000
plata	105.191	23.000	7.000	0.000
hierro	6.211	23.000	7.000	0.009
estanho	108.199	23.000	7.000	0.000

\* (3) Farrar-Glauber Multicollinearity t-Test:

Variable	pbi_-1	cobre	plomo	zinc	oro	plata	hierro	est
a-o								
pbi_mu-1	.							
cobre	14.278	.						
plomo	8.759	8.205	.					
zinc	17.694	18.736	10.874	.				
oro	14.109	10.434	12.879	13.149	.			
plata	15.058	22.123	8.317	17.234	10.648	.		
hierro	3.631	4.882	2.736	3.749	3.021	3.962	.	
estanho	10.621	6.951	11.706	9.477	17.454	7.129	2.369	.

## 10.3. Medidas Correctivas

Dado que se ha demostrado la existencia de una alta correlación entre las variables regresoras, a continuación se plantea la eliminación de aquellas que generan mayor colinealidad. Otra posible solución es quedarse con todas las variables regresoras siempre y cuando todas sean relevantes en el modelo.

```
. *MEDIDAS CORRECTIVAS
. *****

. *Primer modelo alternativo

. reg pbi_mineria pbi_mundial plomo oro plata hierro estanho
```

Source	SS	df	MS	Number of obs = 31		
Model	3.7210e+19	6	6.2017e+18	F( 6, 24) = 1043.77		
Residual	1.4260e+17	24	5.9417e+15	Prob > F = 0.0000		
				R-squared = 0.9962		
				Adj R-squared = 0.9952		
Total	3.7353e+19	30	1.2451e+18	Root MSE = 7.7e+07		

pbi_mineria	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
pbi_mundial	.0000165	7.41e-06	2.23	0.035	1.22e-06	.0000318
plomo	2574976	966177.1	2.67	0.014	580884.4	4569068
oro	233959.7	34393.54	6.80	0.000	162975	304944.5
plata	14165.8	2686.568	5.27	0.000	8620.997	19710.6
hierro	82592.54	19997.78	4.13	0.000	41319.15	123865.9
estanho	-6321754	4829970	-1.31	0.203	-1.63e+07	3646814
_cons	-8.43e+08	2.22e+08	-3.80	0.001	-1.30e+09	-3.85e+08

. vif

Variable	VIF	1/VIF
oro	34.27	0.029179
estanho	29.50	0.033903
pbi_mundial	27.39	0.036511
plata	21.23	0.047114
plomo	10.85	0.092184
hierro	1.84	0.544023
Mean VIF	20.84	

. \*Conclusion: Todavia el vif es alto

. \*Segundo modelo alternativo

. reg pbi\_mineria pbi\_mundial plomo oro plata hierro

Source	SS	df	MS	Number of obs = 31		
Model	3.7200e+19	5	7.4401e+18	F( 5, 25) = 1217.46		
Residual	1.5278e+17	25	6.1111e+15	Prob > F = 0.0000		
				R-squared = 0.9959		
				Adj R-squared = 0.9951		
Total	3.7353e+19	30	1.2451e+18	Root MSE = 7.8e+07		

pbi_mineria	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
pbi_mundial	.0000111	6.22e-06	1.78	0.087	-1.74e-06	.0000239
plomo	2002086	873543.7	2.29	0.031	202989.3	3801183
oro	203657.7	25794.91	7.90	0.000	150532.1	256783.3
plata	16440.7	2077.603	7.91	0.000	12161.8	20719.61
hierro	88553.63	19748.02	4.48	0.000	47881.83	129225.4
_cons	-7.60e+08	2.15e+08	-3.53	0.002	-1.20e+09	-3.16e+08

. vif

Variable	VIF	1/VIF
----------	-----	-------

```

pbi_mundial      18.77    0.053286
      oro         18.74    0.053354
      plata       12.34    0.081027
      plomo        8.62    0.115988
      hierro       1.74    0.573784

Mean VIF      12.04
. *Conclusion: Todavia el vif es alto

. *Tercer modelo alternativo

. reg pbi_mineria pbi_mundial oro cobre

```

Source	SS	df	MS
Model	3.7289e+19	3	1.2430e+19
Residual	6.4565e+16	27	2.3913e+15
Total	3.7353e+19	30	1.2451e+18

Number of obs = 31  
F( 3, 27) = 5197.80  
Prob > F = 0.0000  
R-squared = 0.9983  
Adj R-squared = 0.9981  
Root MSE = 4.9e+07

pbi_mineria	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
pbi_mundial	7.04e-06	3.68e-06	1.91	0.067	-5.17e-07 .0000146
oro	223227.3	11649.44	19.16	0.000	199324.6 247130
cobre	1622987	85256.33	19.04	0.000	1448056 1797919
_cons	3.83e+08	7.43e+07	5.16	0.000	2.31e+08 5.36e+08

```

. vif

```

Variable	VIF	1/VIF
pbi_mundial	16.81	0.059499
cobre	9.98	0.100194
oro	9.77	0.102362
Mean VIF	12.19	

```

. *Conclusion: El vif es aceptable, es menor a 20

```

## 10.4. Ejercicio Propuesto

En el archivo **vbpagr.dta** se tiene de la agricultura para los años 1948 y 1988. Las variables incluidas son: *OUTPUT*: VBP Agropecuario (var. dependiente), *FERT*: Cantidad de fertilizantes utilizados, *LABOR*: Horas de trabajo utilizados, *LAND*: cantidad de acres cultivados, *MACH*: cantidad de horas máquina utilizadas y **SEEDFEED**: Cantidad de semillas y ganados. Se le pide estimar por MCO una función de producción tipo Cobb-Douglas y evaluar la presencia de multicolinealidad, si esta existe corregirla de la mejor manera.

## Parte III

### Modelos de Elección Discreta





# Capítulo 11

## Modelo de Elección Discreta Binaria

Usualmente un modelo de regresión se supuso implícitamente que incluía una variable dependiente ( $Y$ ) numérica y un conjunto de variables explicativas ( $X's$ ) que pueden ser numéricas o discretas (variables dummies). En este capítulo se tratará modelos de regresión en los cuales la variable dependiente o de respuesta puede ser una variable categórica, esto se da en los casos en el que el propósito es explicar o predecir la probabilidad que  $n$ -individuos elija alguna de las alternativas que pueda representar dicha variable endógena y encontrar la probabilidad de que un acontecimiento suceda.

### 11.1. Tipos de Variables de Elección Discreta

En general la variable endógena categórica puede agruparse en dicotómicas (si tiene dos alternativas: si o no, uno u otro) o en policotómicas (si tiene más de dos alternativas).

- **Variables Dicotómicas o Binarias:**

- Estudiar en una institución pública vs privada
- Decisión de trabajar de una mujer casada (si o no)

- Resultado de un examen (aprobado o desaprobado)
- Decisión de acudir al médico (va o no va), etc.

■ **Variables Policotómicas o Múltiples:**

- Alternativas múltiples no ordenadas:
- Decisión de utilizar algún medio de transporte (avión, bus, auto, etc.)
- Decisión de dónde se atiende un paciente (Ministerio de Salud, Es-salud, privado), etc.

■ **Alternativas Múltiples Ordenadas:**

- Estado de salud de un individuo (muy pobre salud, buena salud, muy buena salud).
- Pertenecer a un Nivel de Ingresos (menos de S/. 1000, de S/. 1000 a menos de S/. 5000, de S/. 5,000 o más), etc.

■ **Alternativas Múltiples Secuenciales:**

- Nivel de educación alcanzado (primaria, secundaria, universitaria)
- Nivel de clasificación (local, provincial, departamental, nacional), etc.

■ **Alternativas Múltiples Secuenciales:**

- Número de choques en Lima
- Número de casos con resultado positivo, etc.

## 11.2. Modelos de Elección Discreta para Variables Dicotómicas

### 11.2.1. Modelo Lineal de Probabilidad (MLP)

Considerando que la probabilidad no se observa, el modelo de probabilidad lineal (MPL) se plantea como un modelo de regresión clásico, es decir:

$$Y_i = X_i\beta + \mu_i \quad (11.1)$$

Donde la variable dependiente  $Y_i$  puede tomar valores 0 o 1, y  $X_i$  es el vector fila que representa las k-variables explicativas. Por lo tanto,  $Y_i$  tendrá una función de distribución (cdf) tipo Bernoulli, esto es:

$$F(Y_i) = P^{Y_i}(1 - P)^{1-Y_i} \quad (11.2)$$

El efecto impacto de la variable  $X_i$ , sobre la probabilidad que  $Y_i = 1$  si  $X_i$  es una variable numérica el efecto impacto se calcula como:

$$\beta_i = \frac{\partial P(Y_i = 1 \| X_i)}{\partial X_i} \quad (11.3)$$

Por otro lado, si  $X_i$  es una variable dummy (0, 1) el efecto impacto se calcula como:

$$\beta_i = P(Y_i = 1 \| X_i = 1) - P(Y_i = 1 \| X_i = 0) \quad (11.4)$$

Si se utiliza mínimos cuadrados ordinarios (MCO) para estimar los parámetros del modelo lineal de probabilidad se tiene algunos problemas, tales como:

- El efecto impacto de un cambio en una variable regresora  $X_i$  en la probabilidad es una constante igual a  $\beta$ , cualquiera sea el valor de  $X_i$ .
- Los valores predichos para la probabilidad  $P_i$  no está restringido al rango 0 y 1, esto le quita realismo pues las probabilidades deben ser siempre positivas y permanecer en el rango 0, 1.

- Los errores no se distribuyen normalmente, sino siguen la distribución Bernoulli. Dado que  $Y_i$  sólo puede tomar los valores 0 ó 1, la función de distribución que está asociada es una Bernoulli, esto es, . Sin embargo, este no cumplimiento de normalidad quizá no sea tan crítico ya que a medida que el tamaño de muestra aumenta indefinidamente, los estimadores MCO tienden a ser normalmente distribuidos. Por consiguiente, en muestras grandes, la inferencia estadística MLP seguirá el procedimiento MCO usual bajo el supuesto de normalidad.

### 11.2.2. Modelo Logístico (Logit)

La expresión anterior se puede estimar de otra forma, si se supone que los errores tienen una función de distribución logística, el cual tiene la siguiente especificación:

$$\log\left(\frac{P_i}{1 - P_i}\right) = X_i\beta \quad (11.5)$$

Adicionalmente, si la variable explicativa  $X_i$  es numérica el efecto impacto se obtiene como la derivada de la probabilidad que  $Y_i = 1$  dado un cambio unitario en la variable explicativa,  $X_i$ . Por ejemplo, para el caso de un individuo representativo el efecto marginal de  $X_i$  es:

$$\frac{\partial P(Y_i = 1) \| X_i}{\partial X_i} = P_i(1 - P_i)\beta \quad (11.6)$$

Si la variable  $X_i$  es categórica, esto es, una variable dummy que toma valores 0 ó 1, el *efecto marginal* se obtiene como la diferencia entre la probabilidad que  $Y_i = 1$  dado que  $X_i = 1$  y la probabilidad que  $Y_i = 1$  dado que  $X_i = 0$ . Para el caso de un individuo representativo, el resto de variables explicativas debe tomar su valor promedio.

$$\frac{\partial P(Y_i = 1) \| X_i}{\partial X_i} = \frac{\exp(X_i\beta)}{1 + \exp(X_i\beta)} \| X_i = 1 - \frac{\exp(X_i\beta)}{1 + \exp(X_i\beta)} \| X_i = 0 \quad (11.7)$$

### 11.2.3. Modelo Probabilístico (Probit)

El modelo supone que los errores siguen una función de densidad normal estándar,  $\phi(\cdot)$ , de modo que la función de probabilidad es:

$$P_i = \Phi(X_i\beta) = \int_{-\infty}^{X_i\beta} \phi(z)dz \quad (11.8)$$

donde:  $\phi(z) = \frac{1}{\sqrt{2\pi}} \exp(-\frac{z^2}{2})$ .

Siendo  $\Phi(X_i\beta)$  la función de distribución o acumulativa (cdf), con  $\Phi^{-1}(P_i) = X_i\beta$  y  $\phi(z_i)$  función de densidad de probabilidad normal estándar (pdf). Igual que el modelo anterior, si la variable explicativa  $X_i$  es numérica el efecto impacto es:

$$\frac{\partial P(Y_i = 1) \| X_i}{\partial X_i} = \phi(X_i\beta)\beta \quad (11.9)$$

Si la variable  $X_i$  es una variable dummy que toma valores 0 ó 1, el *efecto marginal* es:

$$\frac{\partial P(Y_i = 1) \| X_i}{\partial X_i} = \Phi(X_i\beta) \| X_i = 1 - \Phi(X_i\beta) \| X_i = 0 \quad (11.10)$$

### 11.2.4. Relaciones entre Modelos Logit y Probit

Si bien los modelos son muy semejantes, la principal diferencia es que la distribución logística tiene extremos más anchos, lo cual significa que la probabilidad condicional  $P_i$  se aproxima a cero o a uno a una tasa menor en el modelo logit en comparación con el probit.

Amemiya (1981) demostró que los coeficientes de los modelos MLP, logit y Probit están relacionados de la siguiente manera:

- $\beta_{MLP} = 0,25\beta_{LOGIT}$ , excepto para la intersección.
- $\beta_{PROBIT} = 0,625\beta_{LOGIT}$ , excepto para la intersección.
- $\beta_{MLP} = 0,25\beta_{LOGIT} + 0,5$ , para la intersección.

### Aplicación

El Parque Nacional de Tingo María se encuentra situado en los Distritos de Rupa Rupa, Damaso Beraum. Departamento de Huánuco. Se realizó una encuesta a los visitantes del Parque Nacional Tingo María, particularmente a la Cueva de las Lechuzas (atractivo turístico y único lugar del país que sirve como hábitat natural a las colonias de lechuzas que están en peligros de extinción). La agricultura, el cultivo de la hoja de coca, la caza ilegal de especies y la deforestación están destruyendo el ecosistema del Parque Nacional y su belleza paisajística. Teniendo información de 92 encuestados se plantea las siguientes preguntas:

1. ¿Cuánto es la Disposición de Pagar (adicional a la tarifa de entrada) de los visitantes para invertir en protección y conservación de dicho atractivo turístico?.
2. ¿Estaría Ud. dispuesto a pagar la cantidad S/. 10 adicionales a la tarifa de ingreso para proteger y conservar el entorno natural y evitar los daños ambientales al área? SI / NO.

Lista de Variables del archivo: **logit\_probit.csv**

**IMPOR** Importancia de las características del área. 1 al 10

**LUGVIS** Exclusividad de la visita. 1al 5

**NVISIT** Número de visitas realizadas.

**REGRES** Si piensan volver. Si 1 No 0

**PROTEC** Si la cueva está bien protegida. Si 1 No 0

**DAP1\_X** cuanto pagaría para proteger y conservar el entorno y evitar los daños ambientales (Variable numérica).

**RDAP1** pagaría la cantidad de S/. 10 para proteger y conservar el entorno natural y evitar los daños ambientales al área? SI / NO (Variable dicotómica)

**EDAD** Edad.

**GENER** Género 1 HOMBRE 2 MUJER.

**HIJOS** Número de hijos.

**TIEMPO** Tiempo.

**GASTO** Gasto.

## Estimación en Mínimos Cuadrados Ordinarios (MCO)

Importación de la base de datos (Excel a Stata)

```
. *****
. *TEMA: ELECCION DISCRETA BINARIA*
. *****

. *Limpiando la memoria
. clear

. *DEFINO LA RUTA
. cd "D:\Econometria-Stata\eleccion-discreta-binaria"
D:\Econometria-Stata\eleccion-discreta-binaria

. *Importar un archivo en Excel al Stata(.dta)
. insheet using logit_probit.csv
(13 vars, 92 obs)

. *Visualizar la data
. browse
```

Una vez importada la base de datos, se quiere determinar la Disposición de Pagar de los visitantes para invertir en protección y conservación de dicho atractivo turístico. En este caso la variable dependiente sería *dap1\_x* y las variables explicadas anteriores serían las variables regresoras:

```
. *Estimación en Mínimos Cuadrados Ordinarios (MCO)
. *****
.
. reg dap1_x import lugvis nvisit regres protec edad gener hijos tiempo gasto
```

Source	SS	df	MS
Model	88.6794405	10	8.86794405
Residual	1126.04882	81	13.9018373
Total	1214.72826	91	13.3486622

```

Number of obs =      92
F( 10,      81) =    0.64
Prob > F       =    0.7772
R-squared      =    0.0730
Adj R-squared  =   -0.0414
Root MSE      =    3.7285
```



dap1_x	Coef.	Std. Err.	t	P> t	[95 % Conf. Interval]	
import	.1960699	.1592004	1.23	0.222	-.120689	.5128287
lugvis	.6489333	.7140249	0.91	0.366	-.7717522	2.069619
nvisit	-.1890576	.4109541	-0.46	0.647	-1.006727	.6286121
regres	-1.47543	2.087338	-0.71	0.482	-5.628578	2.677717
protec	.3205851	.8609684	0.37	0.711	-1.392472	2.033642
edad	-.01271	.0809495	-0.16	0.876	-.173774	.1483541
gener	.6207085	.8346738	0.74	0.459	-1.04003	2.281448
hijos	-.2240379	.7264512	-0.31	0.759	-1.669448	1.221372
tiempo	-.088766	.0786056	-1.13	0.262	-.2451664	.0676344
gasto	.0016019	.0009418	1.70	0.093	-.0002721	.0034759
_cons	7.644663	3.93016	1.95	0.055	-.1751215	15.46445

Calculo de la Disposición a Pagar en MCO:

```
. *Calculo de la Disposición a Pagar en MCO:
. egen mda1_x=median(dap1_x)

. br mda1_x
```

- La media de la disposición a Pagar – indica cuanto pagaría adicional los visitantes para invertir en protección y conservación de dicho atractivo turístico. En este caso el DAP es 7 soles.
- El  $R^2$  cuadrado Indica la bondad del modelo diseñado. Un valor 0.07 indica que el modelo explica la realidad apenas en un 7 %. El valor bajo se puede deber a pocas observaciones (pocas encuestas); o al hecho de haber omitido variables importantes que pueden explicar mejor la realidad. Por ejemplo? Propósito principal de la visita, época de la visita, forma de pago, grado de instrucción, estado civil, ocupación, residencia, ruta, medio de transporte, etc.
- Hay que analizar los signos de los coeficientes betas, si el signo es positivo entonces la variable afecta positivamente la DAP; caso contrario el efecto es inverso. El coeficiente en si es el tamaño o magnitud de impacto de la variable independiente sobre la DAP.
- En todos los casos, las probabilidades son mayores al 5 %, entonces los coeficientes estimados (individualmente) son no significativos. Lo anterior, hace referencia quizás a plantear otro modelo o eliminar algunas variables que posiblemente estén correlacionadas. Por ejemplo comprobar que el siguiente modelo es mejor que el estimado anteriormente:

```
. *Modelo Alternativo
. reg dap1_x rdap1 edad tiempo regres , nocons
```

Source	SS	df	MS			
Model	5070.86971	4	1267.71743	Number of obs = 92		
Residual	1244.13029	88	14.1378442	F( 4, 88) = 89.67		
Total	6315	92	68.6413043	Prob > F = 0.0000		
				R-squared = 0.8030		
				Adj R-squared = 0.7940		
				Root MSE = 3.76		

dap1_x	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
rdap1	-2.796118	.8343532	-3.35	0.001	-4.45422	-1.138017
edad	.065583	.0316849	2.07	0.041	.0026159	.12855
tiempo	.0891388	.0462791	1.93	0.057	-.0028311	.1811087
regres	6.286062	.9524425	6.60	0.000	4.393282	8.178841

### Estimación de Modelos Probabilísticos (MLP, Logit y Probit)

Hay que recordar que para este tipo de modelos de regresión, la variable dependiente es una dicotómica: SI y NO. En nuestro caso, la variable dependiente sería RDAP1 (pagaría la cantidad de S/. 10 adicionales para proteger y conservar el entorno natural y evitar los daños ambientales en el lugar turístico? SI / NO). Es importante tener en cuenta, que se pueden incluir todas las variables explicativas o solo algunas. Sin embargo, se procede a eliminar aquellas que fueron altamente no significativas.

```
. *Estimación de Modelos Probabilísticos (MLP, Logit y Probit)
. *****
```

```
. *****
. *Modelo de Probabilidad Lineal*
. *****
```

```
. reg rdap1 dap1_x edad tiempo regres , nocons
```

Source	SS	df	MS			
Model	33.9897631	4	8.49744077	Number of obs = 92		
Residual	18.0102369	88	.204661783	F( 4, 88) = 41.52		
Total	52	92	.565217391	Prob > F = 0.0000		
				R-squared = 0.6536		
				Adj R-squared = 0.6379		
				Root MSE = .4524		

rdap1	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
dap1_x	-.0404771	.0120782	-3.35	0.001	-.06448	-.0164741
edad	.0103356	.0037452	2.76	0.007	.0028927	.0177784
tiempo	.0148721	.0054588	2.72	0.008	.004024	.0257203
regres	.4315095	.1323492	3.26	0.002	.1684933	.6945257

```
. *Para evitar problemas de heteroscedasticidad
```

```
. reg rdap1 dap1_x edad tiempo regres , nocons robust
```

Linear regression

Number of obs = 92  
 F( 4, 88) = 67.29  
 Prob > F = 0.0000  
 R-squared = 0.6536  
 Root MSE = .4524

rdap1	Coef.	Robust Std. Err.	t	P> t	[95% Conf. Interval]	
dap1_x	-.0404771	.0118323	-3.42	0.001	-.0639913	-.0169629
edad	.0103356	.0037068	2.79	0.006	.0029692	.017702
tiempo	.0148721	.0042757	3.48	0.001	.0063751	.0233692
regres	.4315095	.142156	3.04	0.003	.1490044	.7140146

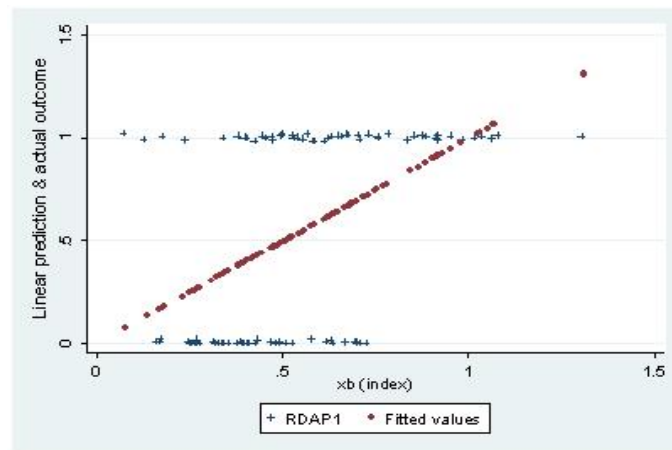
```
. *Vista gráfica:

. predict yhat
(option xb assumed; fitted values)

. predict xb, xb

. label var xb "xb (index)"

. scatter rdap1 yhat xb, symbol(+ o) ///
  jitter(2) ltitle("Linear prediction & actual outcome")
```



```
. *Estimando la probabilidad individual para MPL
. reg rdap1 dap1_x edad tiempo regres , nocons
```

Source	SS	df	MS	Number of obs = 92	
Model	33.9897631	4	8.49744077	F( 4, 88) =	41.52
Residual	18.0102369	88	.204661783	Prob > F =	0.0000
				R-squared =	0.6536
				Adj R-squared =	0.6379
Total	52	92	.565217391	Root MSE =	.4524

rdap1	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
dap1_x	-.0404771	.0118323	-3.42	0.001	-.0639913	-.0169629
edad	.0103356	.0037068	2.79	0.006	.0029692	.017702
tiempo	.0148721	.0042757	3.48	0.001	.0063751	.0233692
regres	.4315095	.142156	3.04	0.003	.1490044	.7140146

dap1_x	-.0404771	.0120782	-3.35	0.001	-.06448	-.0164741
edad	.0103356	.0037452	2.76	0.007	.0028927	.0177784
tiempo	.0148721	.0054588	2.72	0.008	.004024	.0257203
regres	.4315095	.1323492	3.26	0.002	.1684933	.6945257

```
. predict rdap1mpl
(option xb assumed; fitted values)
```

```
. *****
. *Modelo Logit*
. *****
```

```
. logit rdap1 dap1_x edad tiempo regres , nocons
```

```
Iteration 0: log likelihood = -63.769541
```

```
Iteration 1: log likelihood = -51.686026
```

```
Iteration 2: log likelihood = -51.628551
```

```
Iteration 3: log likelihood = -51.62851
```

```
Iteration 4: log likelihood = -51.62851
```

```
Logistic regression
```

```
Number of obs = 92
```

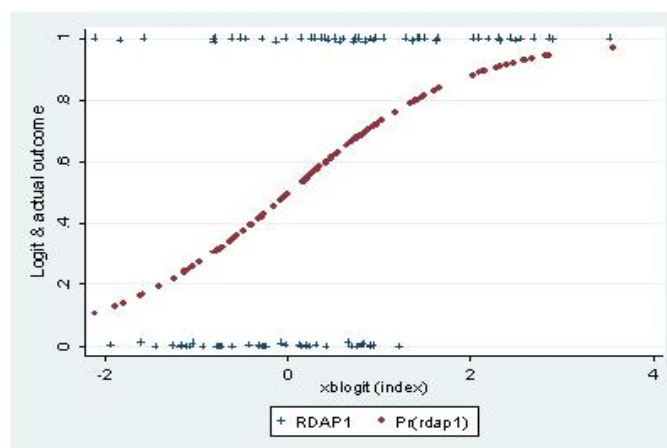
```
Wald chi2(4) = 16.78
```

```
Prob > chi2 = 0.0021
```

```
Log likelihood = -51.62851
```

rdap1	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
dap1_x	-.2451406	.0685435	-3.58	0.000	-.3794835	-.1107977
edad	.0323642	.0206005	1.57	0.116	-.0080121	.0727405
tiempo	.0771123	.0345736	2.23	0.026	.0093492	.1448753
regres	.5405781	.6398627	0.84	0.398	-.7135297	1.794686

```
. *Vista gráfica:
. predict yhatt
(option pr assumed; Pr(rdap1))
. predict xbb, xb
. label var xbb "xblogit (index)"
. scatter rdap1 yhatt xbb, symbol(+ o) jitter(2) ///
. lltitle("Logit & actual outcome")
```



```
. *Estimando la probabilidad individual para el modelo Logit
```

```
. logit rdap1 dap1_x edad tiempo regres , nocons
```

```
Iteration 0: log likelihood = -63.769541
Iteration 1: log likelihood = -51.686026
Iteration 2: log likelihood = -51.628551
Iteration 3: log likelihood = -51.62851
Iteration 4: log likelihood = -51.62851
```

```
Logistic regression                Number of obs   =          92
                                Wald chi2(4)      =          16.78
Log likelihood = -51.62851         Prob > chi2     =          0.0021
```

rdap1	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
dap1_x	-.2451406	.0685435	-3.58	0.000	-.3794835	-.1107977
edad	.0323642	.0206005	1.57	0.116	-.0080121	.0727405
tiempo	.0771123	.0345736	2.23	0.026	.0093492	.1448753
regres	.5405781	.6398627	0.84	0.398	-.7135297	1.794686

```
. predict rdap1logit
(option pr assumed; Pr(rdap1))
```

```
. *Cálculo de la DAP -- Modelo Logit
```

```
. logit rdap1 dap1_x edad tiempo regres , nocons
```

```
Iteration 0: log likelihood = -63.769541
Iteration 1: log likelihood = -51.686026
Iteration 2: log likelihood = -51.628551
Iteration 3: log likelihood = -51.62851
Iteration 4: log likelihood = -51.62851
```

```
Logistic regression                Number of obs   =          92
                                Wald chi2(4)      =          16.78
Log likelihood = -51.62851         Prob > chi2     =          0.0021
```

rdap1	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
dap1_x	-.2451406	.0685435	-3.58	0.000	-.3794835	-.1107977
edad	.0323642	.0206005	1.57	0.116	-.0080121	.0727405
tiempo	.0771123	.0345736	2.23	0.026	.0093492	.1448753
regres	.5405781	.6398627	0.84	0.398	-.7135297	1.794686

```
. *DAP por persona encuestada
```

```
. gen dap_logit1= -(_b[edad]*edad+_b[tiempo]*tiempo+_b[regres]*regres)/_b[dap1_x]
```

```
. *DAP promedio
```

```
. egen dap_logit2= median(-(_b[edad]*edad+_b[tiempo]*tiempo+_b[regres]*regres)/_b[dap1_x])
```

```
. *Visualizar
```

```
. br dap_logit1 dap_logit2
```

```
. *****
. *Modelo Probit*
. *****
```

```
. probit rdap1 dap1_x edad tiempo regres , nocons
```

```
Iteration 0: log likelihood = -63.769541
Iteration 1: log likelihood = -51.713968
Iteration 2: log likelihood = -51.571368
Iteration 3: log likelihood = -51.571295
Iteration 4: log likelihood = -51.571295
```

```
Probit regression              Number of obs   =          92
                              Wald chi2(4)    =         19.29
Log likelihood = -51.571295    Prob > chi2   =         0.0007
```

rdap1	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
dap1_x	-.1473299	.0392988	-3.75	0.000	-.2243541	-.0703058
edad	.0195594	.0122553	1.60	0.110	-.0044606	.0435795
tiempo	.045675	.0197966	2.31	0.021	.0068743	.0844757
regres	.3453639	.3929503	0.88	0.379	-.4248045	1.115532

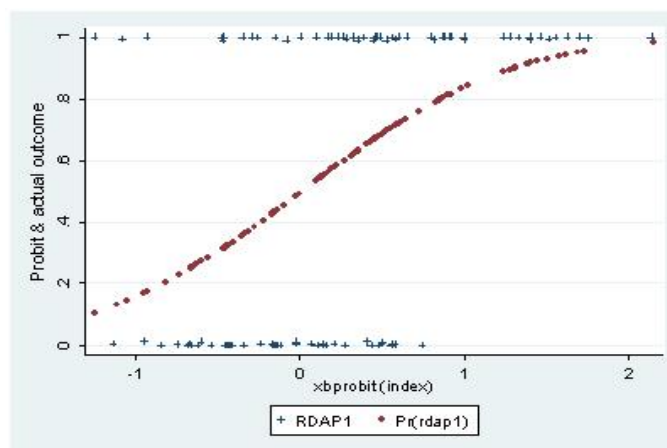
```
. *Vista gráfica:
```

```
. predict yhathtt
(option pr assumed; Pr(rdap1))
```

```
. predict xbbb, xb
```

```
. label var xbbb "xbprobit (index)"
```

```
. scatter rdap1 yhathtt xbbb, symbol(+ o) ///
jitter(2) ltitle("Probit & actual outcome")
```



```
. *Estimando la probabilidad individual para el modelo Probit
```

```
. probit rdap1 dap1_x edad tiempo regres , nocons
```

```
Iteration 0: log likelihood = -63.769541
Iteration 1: log likelihood = -51.713968
Iteration 2: log likelihood = -51.571368
Iteration 3: log likelihood = -51.571295
Iteration 4: log likelihood = -51.571295
```

```
Probit regression                      Number of obs   =          92
                                      Wald chi2(4)      =          19.29
Log likelihood = -51.571295           Prob > chi2     =          0.0007
```

rdap1	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
dap1_x	-.1473299	.0392988	-3.75	0.000	-.2243541	-.0703058
edad	.0195594	.0122553	1.60	0.110	-.0044606	.0435795
tiempo	.045675	.0197966	2.31	0.021	.0068743	.0844757
regres	.3453639	.3929503	0.88	0.379	-.4248045	1.115532

```
. predict rdap1probit
```

```
(option pr assumed; Pr(rdap1))
```

```
. *Cálculo de la DAP -- Modelo Probit
```

```
. probit rdap1 dap1_x edad tiempo regres , nocons
```

```
Iteration 0: log likelihood = -63.769541
Iteration 1: log likelihood = -51.713968
Iteration 2: log likelihood = -51.571368
Iteration 3: log likelihood = -51.571295
Iteration 4: log likelihood = -51.571295
```

```
Probit regression                      Number of obs   =          92
                                      Wald chi2(4)      =          19.29
Log likelihood = -51.571295           Prob > chi2     =          0.0007
```

rdap1	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
dap1_x	-.1473299	.0392988	-3.75	0.000	-.2243541	-.0703058
edad	.0195594	.0122553	1.60	0.110	-.0044606	.0435795
tiempo	.045675	.0197966	2.31	0.021	.0068743	.0844757
regres	.3453639	.3929503	0.88	0.379	-.4248045	1.115532

```
. *DAP por persona encuestada
```

```
. gen dap_probit1= -(_b[edad]*edad+_b[tiempo]*tiempo+_b[regres]*regres)/_b[dap1_x]
```

```
. *DAP promedio
```

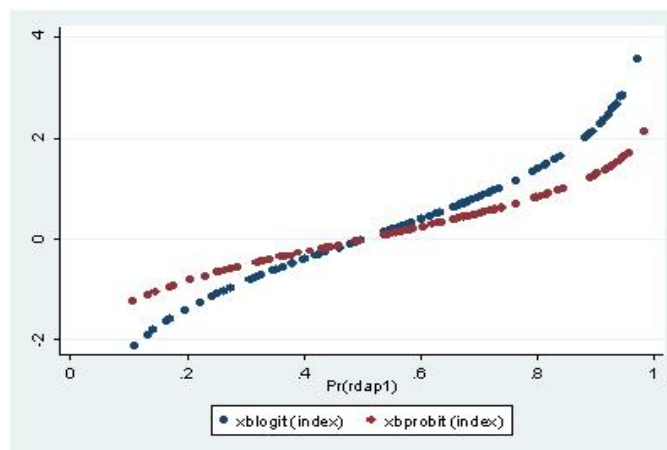
```
. egen dap_probit2= median(-(_b[edad]*edad+_b[tiempo]*tiempo+_b[regres]*regres)/_b[dap1_x])
```

```
. *visualizar
```

```
. br dap_probit1 dap_probit2
```

```
. *Comparaciones dos últimos modelos (Logit vs. Probit): Vista gráfica
```

```
. twoway (scatter xbb yhatt) (scatter xbbb yhattt)
```



Pronóstico (probabilidad individual) de que los visitantes paguen la cantidad de S/. 15 adicionales para proteger y conservar el entorno natural y evitar los daños ambientales al área

```
. list rdap1mpl rdap1logit rdap1probit in 1/10
```

	rdap1mpl	rdap1l-t	rdap1p-t
1.	.3908928	.3634279	.3762003
2.	.4731561	.4332702	.4454062
3.	.9055933	.9075889	.9162604
4.	.6130841	.699643	.7017308
5.	.4816584	.8023702	.7992527
6.	.2698136	.2426241	.2515068
7.	.4998367	.6003659	.6052126
8.	.4653244	.3994535	.4094213
9.	.7161391	.6779036	.6757985
10.	1.066361	.9438509	.9534355

Efecto Impacto de las variables explicativas en cada uno de los tres modelos anteriores

```
. *Efectos Marginales (MFX)
. *****

. *Modelo de probabilidad lineal

. reg rdap1 dap1_x edad tiempo regres , nocons
```

Source	SS	df	MS
Model	33.9897631	4	8.49744077
Residual	18.0102369	88	.204661783

```
Number of obs =      92
F( 4,      88) =    41.52
Prob > F       =    0.0000
R-squared      =    0.6536
Adj R-squared  =    0.6379
```



Total	52	92	.565217391	Root MSE	=	.4524
rdap1	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
dap1_x	-.0404771	.0120782	-3.35	0.001	-.06448	-.0164741
edad	.0103356	.0037452	2.76	0.007	.0028927	.0177784
tiempo	.0148721	.0054588	2.72	0.008	.004024	.0257203
regres	.4315095	.1323492	3.26	0.002	.1684933	.6945257

.mfx

Marginal effects after regress

```
y = Fitted values (predict)
= .55576647
```

variable	dy/dx	Std. Err.	z	P> z	[	95% C.I.	]	X
dap1_x	-.0404771	.01208	-3.35	0.001	-.06415	-.016804		7.44565
edad	.0103356	.00375	2.76	0.006	.002995	.017676		28.9891
tiempo	.0148721	.00546	2.72	0.006	.004173	.025571		9.73478
regres*	.4315095	.13235	3.26	0.001	.17211	.690909		.956522

(\*)  $dy/dx$  is for discrete change of dummy variable from 0 to 1

```
. *Modelo Logit
```

```
. logit rdap1 dap1_x edad tiempo regres , nocons
```

```
Iteration 0:    log likelihood = -63.769541
```

```
Iteration 0: log likelihood = -55.739911
Iteration 1: log likelihood = -51.686026
```

```
Iteration 1: log likelihood = -51.633323
Iteration 2: log likelihood = -51.628551
```

```
Iteration 3: log likelihood = -51.62851
```

```
Iteration 4: log likelihood = -51.62851
```

Logistic regression	Number of obs =	92
---------------------	-----------------	----

Wald chi2(4) = 16.78

Log likelihood = -51.62851

```
Prob > chi2      =    0.0021
```

rdap1	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
dap1_x	-.2451406	.0685435	-3.58	0.000	-.3794835	-.1107977
edad	.0323642	.0206005	1.57	0.116	-.0080121	.0727405
tiempo	.0771123	.0345736	2.23	0.026	.0093492	.1448753
regres	.5405781	.6398627	0.84	0.398	-.7135297	1.794686

. mfx

### Marginal effects after logit

```
y = Pr(rdap1) (predict)
= .59404788
```

variable	dy/dx	Std. Err.	z	P> z	[ 95% C.I. ]	X
dap1_x	-.0591169	.01648	-3.59	0.000	-.091424 -.02681	7.44565
edad	.0078048	.00493	1.58	0.113	-.00185 .01746	28.9891
tiempo	.018596	.00823	2.26	0.024	.002471 .034721	9.73478
regres*	.1337379	.15849	0.84	0.399	-.176899 .444375	.956522

(\*)  $dy/dx$  is for discrete change of dummy variable from 0 to 1

```
. *Modelo Probit

. probit rdap1 dap1_x edad tiempo regres , nocons

Iteration 0:   log likelihood = -63.769541
Iteration 1:   log likelihood = -51.713968
Iteration 2:   log likelihood = -51.571368
Iteration 3:   log likelihood = -51.571295
Iteration 4:   log likelihood = -51.571295

Probit regression                               Number of obs   =          92
                                                Wald chi2(4)      =         19.29
Log likelihood = -51.571295                    Prob > chi2       =         0.0007
```

	rdap1	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
dap1_x		-.1473299	.0392988	-3.75	0.000	-.2243541 -.0703058
edad		.0195594	.0122553	1.60	0.110	-.0044606 .0435795
tiempo		.045675	.0197966	2.31	0.021	.0068743 .0844757
regres		.3453639	.3929503	0.88	0.379	-.4248045 1.115532

```
. mfx

Marginal effects after probit
y = Pr(rdap1) (predict)
= .59678259
```

variable	dy/dx	Std. Err.	z	P> z	[ 95% C.I. ]	X
dap1_x	-.0570379	.01511	-3.77	0.000	-.086662 -.027414	7.44565
edad	.0075723	.00471	1.61	0.108	-.001653 .016797	28.9891
tiempo	.0176828	.00759	2.33	0.020	.002806 .03256	9.73478
regres*	.1365816	.15559	0.88	0.380	-.168371 .441534	.956522

(\*) dy/dx is for discrete change of dummy variable from 0 to 1

Las variables sombreadas de amarillo representan el efecto impacto de cada variable explicativa para un modelo Probit.

### Conclusiones:

- La edad, el tiempo y si piensa volver los individuos (visitantes), aumentan la probabilidad de que dicho individuo esté dispuesto a pagar 10 soles. Por otro lado, si se incrementa el pago de la DAP (*DAP1\_X*) y así tener más recaudación para evitar los daños ambientales disminuye la probabilidad de que los visitantes paguen 10 soles.
- Si los visitantes desean volver, aumentan la probabilidad de que dicha persona pague 10 soles adicionales en 13.65 % para un modelo probit, 13.37 % para un modelo logit y 43.15 % para un modelo MPL.

- El tiempo que se demoran en llegar a dicha área aumentan la probabilidad de que dicha persona pague los 10 soles adicionales en 1.76 % para un modelo probit y 1.85 % para un modelo logit. Con respecto a los resultados del Seudo-R2 de los modelos logit y probit, se dan los siguientes resultados:

McFadden R-squared Logit 0.1820

McFadden R-squared Probit 0.1826

El pseudo R2 de McFadden del modelo Probit es ligeramente superior al del modelo Logit.

### Clasificación

Para determinar si el modelo predice correctamente la probabilidad de la variable dependiente realizamos la prueba de clasificación.

```
. *Clasificación
. *****

. logit rdap1 dap1_x edad tiempo regres, nocons
Iteration 0:   log likelihood = -63.769541
Iteration 1:   log likelihood = -51.686026
Iteration 2:   log likelihood = -51.628551
Iteration 3:   log likelihood = -51.62851
Iteration 4:   log likelihood = -51.62851

Logistic regression               Number of obs   =       92
                                Wald chi2(4)        =       16.78
Log likelihood = -51.62851        Prob > chi2     =       0.0021
```

rdap1	Coef.	Std. Err.	z	P> z	[95 % Conf. Interval]	
dap1_x	-.2451406	.0685435	-3.58	0.000	-.3794835	-.1107977
edad	.0323642	.0206005	1.57	0.116	-.0080121	.0727405
tiempo	.0771123	.0345736	2.23	0.026	.0093492	.1448753
regres	.5405781	.6398627	0.84	0.398	-.7135297	1.794686

```
. estat classification
```

Logistic model for rdap1

Classified	True		Total
	D	~D	
+	40	15	55
-	12	25	37
Total	52	40	92

Classified + if predicted Pr(D) >= .5  
True D defined as rdap1 != 0

Sensitivity	Pr( +  D)	76.92%
Specificity	Pr( -  ~D)	62.50%
Positive predictive value	Pr( D  +)	72.73%
Negative predictive value	Pr(~D  -)	67.57%
<hr/>		
False + rate for true ~D	Pr( +  ~D)	37.50%
False - rate for true D	Pr( -  D)	23.08%
False + rate for classified +	Pr(~D  +)	27.27%
False - rate for classified -	Pr( D  -)	32.43%
<hr/>		
Correctly classified		70.65%

```
. probit rdap1 dap1_x edad tiempo regres, nocons
```

```
Iteration 0: log likelihood = -63.769541
Iteration 1: log likelihood = -51.713968
Iteration 2: log likelihood = -51.571368
Iteration 3: log likelihood = -51.571295
Iteration 4: log likelihood = -51.571295
```

```
Probit regression               Number of obs   =          92
                                Wald chi2(4)       =         19.29
Log likelihood = -51.571295     Prob > chi2    =         0.0007
```

rdap1	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
dap1_x	-.1473299	.0392988	-3.75	0.000	-.2243541	-.0703058
edad	.0195594	.0122553	1.60	0.110	-.0044606	.0435795
tiempo	.045675	.0197966	2.31	0.021	.0068743	.0844757
regres	.3453639	.3929503	0.88	0.379	-.4248045	1.115532

```
. estat classification
```

```
Probit model for rdap1
```

Classified	True		Total
	D	~D	
+	40	15	55
-	12	25	37
Total	52	40	92

```
Classified + if predicted Pr(D) >= .5
```

```
True D defined as rdap1 != 0
```

Sensitivity	Pr( +  D)	76.92%
Specificity	Pr( -  ~D)	62.50%
Positive predictive value	Pr( D  +)	72.73%
Negative predictive value	Pr(~D  -)	67.57%
<hr/>		
False + rate for true ~D	Pr( +  ~D)	37.50%
False - rate for true D	Pr( -  D)	23.08%
False + rate for classified +	Pr(~D  +)	27.27%
False - rate for classified -	Pr( D  -)	32.43%
<hr/>		
Correctly classified		70.65%

La prueba relaciona probabilidad estimadas contra probabilidad observadas aprox. 71 % de las probabilidades estimadas coinciden con las observadas

### 11.3. Ejercicio Propuesto

La base de datos **fishing.dta** contiene informaci on de 630 individuos a cionados a la pesca en California del Sur para 1989 y fue utilizada por Herriges y Kling (1999) para explicar las preferencias de los individuos a partir de modelos de utilidad aleatoria. En este modelo, los individuos eligen pescar en el muelle o pescar en un bote alquilado dependiendo del precio relativo de ambas opciones de pesca. El precio de pescar en un bote alquilado o en el muelle varia entre individuos por varios factores, como por ejemplo, las diferencias en accesibilidad. De este modo se espera que la probabilidad de pescar en un bote alquilado disminuya mientras el precio relativo de esta opci on con respecto al precio de la pesca en el muelle se incrementa. Usted observa que la variable *dcharter* toma el valor de 1 cuando el individuo ha elegido pescar desde un bote privado y toma el valor de 0 en caso que el individuo haya elegido pescar desde el muelle. Por otro lado, la variable *lnrelp* es el logaritmo del precio relativo de pesca en bote alquilado con respecto al precio de pesca en el muelle. El modelo de eleccion binaria puede escribirse de la siguiente manera:

$$dcharter_i = \beta_0 + \beta_1 \lnrelp_i + \mu_i$$

- Estime el Modelo de Probabilidad Lineal (MPL). Interprete el coeficiente resultante.
- Que problemas puede generar el MPL con respecto a las predicciones de las probabilidades de pesca en bote alquilado para cada individuo? Corrija la matriz de varianzas y covarianzas del MPL construyendo las variables ((ponderadas))
- Estime el modelo Logit. y Probit Que informacion brindan los parametros estimados?.
- Obtenga los siguientes estad sticos de bondad de ajuste: (i) tasa de predicci on, (ii) Pseudo R2 de McFadden y (iii) Prueba del Ratio de Verosimilitud. Interprete cada uno de los estadisticos resultantes. Los estadisticos deberan de ser calculados utilizando los datos proporcionados por las estimaciones y comandos de calculo en Stata.
- Obtenga los efecto impacto de los modelos MLP, Logit y Probit. Interprete los resultados.

## Parte IV

# Econometría de Series de Tiempo



## Capítulo 12

# Introducción a Series de Tiempo en STATA

En el presente capítulo nos centraremos en aplicar análisis de series temporales y la forma de hacerlo con Stata. Existen diversos libros que cubren la teoría del análisis de series de tiempo, así que no hay necesidad de detenerme en las pruebas y demostraciones.

En un principio nos centraremos en un Análisis Univariado de datos que son observados en distintos puntos discretos del tiempo, y posteriormente se abarcará el análisis multivariado.

### 12.1. Análisis de Serie Temporal Univariado en STATA

Como es de saber, en este tipo de análisis el *tiempo* juega un papel. Es decir, a diferencia de los conjuntos de datos de corte transversal, existe un orden natural de las observaciones en un conjunto de datos a través del tiempo.

En segundo lugar, muchas series tiempo exhiben el fenómeno de *autocorrelación*;



es decir, el valor de una variable en el tiempo  $t$  está a menudo correlacionada con sus valores en los tiempos  $t-1$ ,  $t+1$ , y así sucesivamente.

Stata está diseñado para trabajar con los datos que se recogieron en los puntos equidistantes en el tiempo. Por ejemplo, usted puede tener los datos semanales de inventarios, datos de ventas mensuales, trimestrales datos económicos, o los datos anuales de la climatología.

Todos los conjuntos de datos de series de tiempo debe tener una variable que identifica el período en que se tomó cada observación. Stata necesita esto por tres razones: **i.** La variable temporal representa el orden de clasificación de los datos. A diferencia de los datos transversales, el orden en que aparecen las observaciones en el conjunto de datos es significativa. **ii.** Stata usa la variable de tiempo para identificar las lagunas y los datos que faltan en la serie, **iii.** Stata a menudo necesita referirse a un valor de una variable de un período anterior a la época actual, y que a menudo trabajan con los cambios de período a período en las variables. Stata cuenta con herramientas que hacen llegar los valores rezagados y las diferencias es fácil, pero para que funcionen, Stata debe saber la variable que representa el tiempo.

Se ilustra mediante el uso de un conjunto de datos del Índice de Producto Bruto Interno año base 1994 (*pbi*) y Consumo Privado (*consumo*) que se encuentra contenida en la base de datos *indice\_pbi* que abarca el periodo enero 1992 hasta mayo 2012.

El comando **tsset** se utiliza para identificar la variable temporal. La variable  $t$  se numerará consecutivamente, el cual representa el mes en que se observa el índice.

```
. *Limpiando la memoria
. clear

. *Incluyendo la ruta donde se encuentra el archivo
. cd "D:\Econometria-Stata\introducción-serie-tiempo"
D:\Econometria-Stata\introducción-serie-tiempo

. *Abriendo la base de datos en formato Stata (.dta)
```

```
. use indice_pbi.dta,clear

. *listamos algunas observaciones de la base de datos

. list in 1/10
```

	t	pbi
1.	1	85.0194
2.	2	80.4039
3.	3	84.653
4.	4	83.4272
5.	5	87.1733
6.	6	87.9181
7.	7	84.174
8.	8	81.6815
9.	9	80.9778
10.	10	85.7191

```


. *Establecemos la base de datos como serie temporal

. tsset t
    time variable:  t, 1 to 244
                delta:  1 unit

. * hacemos una descripción de la base

. describe
Contains data from indice_pbi.dta
  obs:                244
 vars:                 2                      23 Jul 2012 10:40
size:                 3,416 (98.8% of memory free)
```

---

variable name	storage type	display format	value label	variable label
t	int	%8.0g		
pbi	float	%8.0g		

---

```
Sorted by:  t
```

Cuando usamos el comando *tsset*, Stata hace una nota de la variable temporal y ordena el conjunto de datos según dicha variable, si es necesario. Si usamos **describe** al conjunto de datos, se observará que está ordenada por *t*.

En la base de datos, el primer mes observado es Enero 1992. Así que, el Stata puede mostrar los meses en vez de los valores genéricos de la variable *t* cuando se listen los datos, a través de la función **ym()**, que toma como argumento el número de años y el mes de inicio, y así el Stata devuelve una serie cuyos valores se expresan en fecha mensual.

```
. *creamos una variable para las fechas mensuales

. display ym(1992, 1) //mes de enero 1992
384

. generate time = ym(1992, 1) + t - 1

. tsset time, monthly
    time variable:  time, 1992m1 to 2012m4
        delta: 1 month
```

Ahora usamos el comando **tsset** con la variable *time*, y usamos la opción **monthly** para indicarle al Stata que está variable lo interprete con periodicidad mensual.

Nuestra variable *time* está etiquetada con las fechas mensuales de *1992m1* hasta *2012m4*, lo que indica el mes y el año de cada observación que fue recolectada. Otra forma era haber utilizado el código *%tm*:

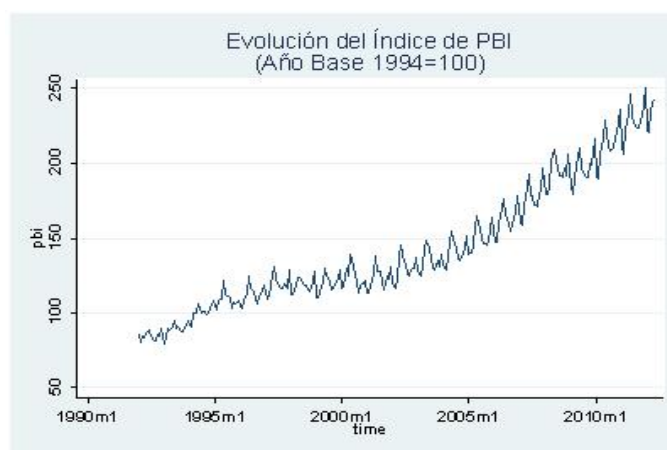
```
. tsset time, format(%tm)
    time variable:  time, 1992m1 to 2012m4
        delta: 1 month
```

Cuando se trabaja con variables temporales, siempre es útil graficarlos para saber si existe ciertas particularidades en la información, como por ejemplo la presencia de tendencia, Estacionalidad, Quiebres, Outliers, etc. Si graficamos la serie del *pbi* con el comando **tsline**, se puede apreciar una amrcada estacionalidad en los meses de Junio y Diciembre, además de la tendencia creciente a lo largo del tiempo.

```
. *graficamos la evolución del pbi
. tsline pbi, title("Evolución del Índice de PBI" "(Año Base 1994=100)")
```

## 12.2. Operadores de Serie de Tiempo

Debido a que las series cronológicas de datos tiene un *orden temporal natural*, a menudo nos queremos referir no al valor de la serie en sí en el tiempo *t*, sino más bien a sus valores rezagados o los cambios en el valor de la serie de tiempo *t-1* a *t*.

Figura 12.1: Comando `tsline`

Stata cuenta con una serie de operadores que se pueden aplicar a las variables para hacer más fácil para referirse a tales valores. En esta discusión, que se entrelazan tanto el álgebra y el código de Stata.

### 12.2.1. Operador de Rezagos

El operador de rezago (*lag*) **L** se utiliza comúnmente en el análisis de series de tiempo y se define de tal manera que:

$$L.x_t = x_{t-1}$$

$$L^2.x_t = L.(L.x_t) = L.x_{t-1} = x_{t-2}$$

y en general,  $L^n.x_t = x_{t-n}$ . En Stata, se indica el grado del rezago entre el operador del rezago y el nombre de la variable a la que queremos rezagar, tal y como se muestra a continuación:

```
. *Operadores de Series de Tiempo

. *Operador de Rezagos

. generate lpbi=L.pbi
(1 missing value generated)
```

```
. gen l2pbi=L2.pbi
(2 missing values generated)

. list time pbi lpbi l2pbi in 1/10
```

	time	pbi	lpbi	l2pbi
1.	1992m1	85.0194	.	.
2.	1992m2	80.4039	85.01944	.
3.	1992m3	84.653	80.40388	85.01944
4.	1992m4	83.4272	84.65303	80.40388
5.	1992m5	87.1733	83.42715	84.65303
6.	1992m6	87.9181	87.17332	83.42715
7.	1992m7	84.174	87.91812	87.17332
8.	1992m8	81.6815	84.17403	87.91812
9.	1992m9	80.9778	81.68151	84.17403
10.	1992m10	85.7191	80.97782	81.68151

y en general,  $L\#x$  donde  $\#$  es un número entero mayor o igual a 0, y se refiere al número de periodos rezagados de la variable  $x$ . El caso especial de  $L0x$  indica el valor corriente de la variable  $x$ , y a veces es útil en un contexto de programación.

### 12.2.2. Operador de Adelanto

El operador de adelanto (*forward*)  $F$  es opuesto al operador  $L$ . Este da el valor posterior en el tiempo en la variable.

$$F.x_t = x_{t+1}$$

$$F^2.x_t = F.(F.x_t) = F.x_{t+1} = x_{t+2}$$

y en el Stata escribimos de la siguiente forma:

```
. *Operador de adelanto

. generate fpbi=F.pbi
(1 missing value generated)

. gen f2pbi=F2.pbi
(2 missing values generated)

. list time pbi fpbi f2pbi in -10/1
```

	time	pbi	fpbi	f2pbi
235.	2011m7	226.421	223.8375	223.1113
236.	2011m8	223.838	223.1113	229.0115
237.	2011m9	223.111	229.0115	230.4544
238.	2011m10	229.012	230.4544	249.8597
239.	2011m11	230.454	249.8597	223.0286
240.	2011m12	249.86	223.0286	220.4585
241.	2012m1	223.029	220.4585	235.0209
242.	2012m2	220.459	235.0209	243.5336
243.	2012m3	235.021	243.5336	.
244.	2012m4	243.534	.	.

### 12.2.3. Operador de Diferencia

A menudo no nos interesa trabajar con los valores en niveles de la variable, sino con los cambios periodo a periodo del mismo. Y veremos más adelante que hay razones estadísticas para utilizar valores diferenciados de una serie en lugar de su nivel.

El Operador de Diferencia **D** ( $\Delta$ ) se define como:

$$\Delta x_t = x_t - x_{t-1}$$

$$\Delta^2 x_t = \Delta \Delta x_t = \Delta(x_t - x_{t-1}) = (x_t - x_{t-1}) - (x_{t-1} - x_{t-2})$$

En Stata se usa la letra *D* para obtener la primera diferencia, *D2* para obtener la segunda diferencia y así sucesivamente.

```
. *Operador de diferencia

. generate dpbi=D.pbi
(1 missing value generated)

. gen d2pbi=D2.pbi
(2 missing values generated)

. list time pbi dpbi d2pbi in 1/10
```

	time	pbi	dpbi	d2pbi
1.	1992m1	85.0194	.	.
2.	1992m2	80.4039	-4.615555	.
3.	1992m3	84.653	4.249146	8.8647
4.	1992m4	83.4272	-1.225876	-5.475021
5.	1992m5	87.1733	3.746162	4.972038
6.	1992m6	87.9181	.7448044	-3.001358
7.	1992m7	84.174	-3.744095	-4.488899
8.	1992m8	81.6815	-2.492516	1.251579
9.	1992m9	80.9778	-.7036896	1.788826
10.	1992m10	85.7191	4.741295	5.444984

### 12.2.4. Operador de Diferencia Estacional

Cuando se trata de datos mensuales, por ejemplo, muchas veces queremos comparar el cambio en una variable por cada mes de este año con el mismo mes del año anterior. Esto con la final de comparar dos periodos que no se van a ver influidos por *factores estacionales*, por lo que nos dará una mejor idea de los cambios en las variables a través del tiempo.

El Operador de Diferencia Estacional **DS** ( $\Delta_s$ ) se define como:

$$\Delta_s x_t = x_t - x_{t-s}$$

donde  $s$  depende de la frecuencia de la data. Por ejemplo, para datos mensuales  $s = 12$ , tal que.

$$\Delta_{12} x_t = x_t - x_{t-12}$$

Además, se puede usar el operador de diferencia estacional mas de una vez. Por ejemplo, aplicando dos veces este operador para ver si el efecto estacional ha sido cada vez mas fuerte, se ha mantenido igual o disminuye con el tiempo, dependiendo de si la segunda diferencia estacional es positiva, cero o negativo, respectivamente. Entonces:

$$\Delta_{12}^2 x_t = \Delta_{12}(x_t - x_{t-12}) = (x_t - x_{t-12}) - (x_{t-12} - x_{t-24})$$

En Stata, se usa la letra **S** seguido por el número que representa el periodo de diferenciación estacional

```
. *Operador de diferencia estacional

. generate s12pbi=S12.pbi
(12 missing values generated)

. gen s12pbi2=S12S12.pbi
(24 missing values generated)

. list time pbi s12pbi s12pbi2 in -15/1
```

	time	pbi	s12pbi	s12pbi2
230.	2011m2	206.023	15.82724	5.526276
231.	2011m3	222.872	16.46817	-.3625946
232.	2011m4	233.337	16.8952	-1.453201
233.	2011m5	246.076	16.93275	-1.835114
234.	2011m6	232.614	11.97296	-11.59868
235.	2011m7	226.421	13.82661	-4.194199
236.	2011m8	223.838	15.98938	-1.536041
237.	2011m9	223.111	12.41335	-7.327576
238.	2011m10	229.012	11.58698	-5.468887
239.	2011m11	230.454	11.26797	-9.005386
240.	2011m12	249.86	14.06331	-5.274368
241.	2012m1	223.029	11.71666	-7.864197
242.	2012m2	220.459	14.4352	-1.392044
243.	2012m3	235.021	12.14847	-4.319702
244.	2012m4	243.534	10.19682	-6.69838

### 12.2.5. Combinando Operadores de Serie Temporales

También podemos especificar más de un operador de series de tiempo para una variable. Por ejemplo:

$$\Delta\Delta_{12}y_t = \Delta(\Delta_{12}y_t) = (y_t - y_{t-12}) - (y_{t-12} - y_{t-13})$$

Es necesario de saber, que los operadores de serie de tiempo son conmutativos, esto quiere decir que no existe diferencia alguna en especificar el orden de los operadores.



```
. *Combinando Operadores de Serie Temporal

. generate x=DS12L.pbi
(14 missing values generated)

. gen w=LS12D.pbi
(14 missing values generated)

. list time pbi x w in -15/1
```

	time	pbi	x	w
230.	2011m2	206.023	.2431793	.2431793
231.	2011m3	222.872	-3.753616	-3.753616
232.	2011m4	233.337	.6409302	.6409302
233.	2011m5	246.076	.4270325	.4270325
234.	2011m6	232.614	.0375519	.0375519
235.	2011m7	226.421	-4.959793	-4.959793
236.	2011m8	223.838	1.853653	1.853653
237.	2011m9	223.111	2.162766	2.162766
238.	2011m10	229.012	-3.576035	-3.576035
239.	2011m11	230.454	-.8263702	-.8263702
240.	2011m12	249.86	-.3190002	-.3190002
241.	2012m1	223.029	2.795334	2.795334
242.	2012m2	220.459	-2.346649	-2.346649
243.	2012m3	235.021	2.718536	2.718536
244.	2012m4	243.534	-2.286728	-2.286728

### 12.2.6. Expresiones con Operadores

Los operadores también pueden ser usados en otras expresiones o como lista de variables. Por ejemplo, los operadores pueden ser usados para obtener estadísticas descriptivas variables.

```
. *Expresiones con Operadores
```

```
. sum pbi L.pbi F.pbi D.pbi S12.pbi
```

Variable	Obs	Mean	Std. Dev.	Min	Max
pbi					
--.	244	142.5583	42.27827	79.58114	249.8597
L1.	243	142.1428	41.86326	79.58114	249.8597
F1.	243	142.7951	42.2031	79.58114	249.8597
D1.	243	.6523218	8.124517	-26.83107	19.40523
S12.	232	7.621459	6.514746	-8.172592	24.84401

También se usa con la condicional **if** para limitar la muestra:

```
. sum pbi if D.pbi>0
```

Variable	Obs	Mean	Std. Dev.	Min	Max
pbi	127	145.6637	44.10635	83.49815	249.8597

O por último, es posible realizar una regresión:

```
. regress pbi L.pbi
```

Source	SS	df	MS	Number of obs = 243		
Model	415101.072	1	415101.072	F( 1, 241) = 6281.71		
Residual	15925.4931	241	66.0808844	Prob > F = 0.0000		
Total	431026.565	242	1781.10151	R-squared = 0.9631		
				Adj R-squared = 0.9629		
				Root MSE = 8.129		
pbi	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
pbi L1.	.9893186	.0124824	79.26	0.000	.9647301	1.013907
_cons	2.170609	1.849326	1.17	0.242	-1.472297	5.813514

En el caso de que se quiera especificar una lista de variables, Stata nos permite usar paréntesis para agrupar un conjunto de variables que serán afectados por un operador. Por ejemplo:

```
. sum L(0 1 2 3).pbi
```

Variable	Obs	Mean	Std. Dev.	Min	Max
pbi					
--.	244	142.5583	42.27827	79.58114	249.8597
L1.	243	142.1428	41.86326	79.58114	249.8597
L2.	242	141.759	41.51943	79.58114	249.8597
L3.	241	141.4324	41.29324	79.58114	249.8597

```
. sum L(1/3).pbi
```

Variable	Obs	Mean	Std. Dev.	Min	Max
pbi					
L1.	243	142.1428	41.86326	79.58114	249.8597
L2.	242	141.759	41.51943	79.58114	249.8597
L3.	241	141.4324	41.29324	79.58114	249.8597

```
. reg pbi L( pbi lpbi)
```

Source	SS	df	MS	Number of obs = 242		
Model	411395.58	2	205697.79	F( 2, 239) = 3126.89		
Residual	15722.2339	239	65.7834056	Prob > F = 0.0000		
				R-squared = 0.9632		
				Adj R-squared = 0.9629		
Total	427117.814	241	1772.27309	Root MSE = 8.1107		

pbi	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
pbi L1.	.8874169	.0644332	13.77	0.000	.7604873	1.014346
lpbi L1.	.1037251	.0648495	1.60	0.111	-.0240244	.2314747
_cons	1.999572	1.864951	1.07	0.285	-1.674269	5.673413

### 12.2.7. Cambios Porcentuales

Para muchas variables económicas, generalmente se analiza en términos porcentuales con respecto al mismo periodo del año anterior. La *tasa de cambio* en una variable  $X$  desde el periodo  $t-1$  hasta  $t$  esatá dado por:

$$\Delta \%X = \frac{X_t - X_{t-1}}{X_{t-1}} \times 100$$

Por lo tanto, una forma de calcular una variable que sea igual al cambio porcentual de  $X$ .

```
. *Cambio Porcentual
. gen var1_pbi=(pbi-L.pbi)/L.pbi *100
(1 missing value generated)
```

Alternativamente, se puede calcular de la siguiente forma:

$$Y = \ln(X)$$

$$dY = d\ln(X) = dX/X$$

Si el cambio de  $X$  es relativamente pequeño, no importa cual de las dos formulas se usa para calcular el cambio porcentual.

```
. gen ln_pbi=ln(pbi)
. gen var2_pbi=D.ln_pbi*100
(1 missing value generated)
```

```
. sum var*
```

Variable	Obs	Mean	Std. Dev.	Min	Max
var1_pbi	243	.574856	5.319213	-13.16981	12.52799
var2_pbi	243	.4330761	5.314392	-14.12158	11.8032

Algo importante que mencionar, es que el operador de series temporales no puede ser combinado con funciones del Stata, es decir, no se puede por ejemplo escribir  $100*D.ln(X)$ .

## 12.3. Ejercicio Propuesto

Se presenta el archivo **indice\_produccion.dta** que contiene información de los Índices de Producción Mensual del Producto Bruto Interno, de los sectores Agropecuario, Pesca, Minería e Hidrocarburos, Manufactura, Construcción y Comercio. A través de estos índices, se pide lo siguiente:

- Establecer la base de datos como serie de tiempo.
- Graficar la evolución de cada uno de los índices.
- Por medio de los operadores de rezagos, calcular lo siguiente:
  - Las variaciones porcentuales mensual a 12 meses, el cual se define de la siguiente manera:

$$\Delta \%X_t = \frac{X_t - X_{t-12}}{X_{t-12}} \times 100$$

- Las variaciones porcentuales acumuladas al mes  $j$  en el año  $T$ , el cual se define de la siguiente manera:

$$Var.Acuml.X_{(j,T)} = \frac{\sum_{t=0}^{t=j} X_{(t,T)}}{\sum_{t=0}^{t=j} X_{(t,T-1)}} \times 100 - 100$$

- Las variaciones porcentuales anualizadas, el cual se define de la siguiente manera:

$$Var.Anual.X_{(j,T)} = \frac{\sum_{t=j-12}^{t=j} X_{(t,T)}}{\sum_{t=j-12}^{t=j} X_{(t,T-1)}} \times 100 - 100$$

- Graficar cada una de estas tasas de variación.

# Capítulo 13

## Series de Tiempo Estacionarios

### 13.1. La Naturaleza de Series de Tiempo

Una primera característica de serie de tiempo es que las observaciones están típicamente correlacionadas. Esto es, que el valor de  $X_t$  esta correlacionado con  $X_{t-2}$ ,  $X_{t-1}$ ,  $X_{t+1}$ , y así sucesivamente. En segundo lugar, muchas series temporales exhiben un comportamiento tendencial. Es decir, si hacemos una regresión de una variable con otra variable que indica el tiempo (*tendencia*), ambas variables pueden parecer altamente correlacionadas. En tercer lugar, en datos de serie de tiempo asumimos que la media poblacional puede incluso no existir, a diferencia de la data de sección transversal en donde la media muestral es un estimador de la media poblacional.

Comencemos con algunas definiciones que nos van a permitir caracterizar una variable temporal.

Un **Proceso Estocástico Discreto (PED)** es una sucesión de variables  $\{y_t\}$  donde  $t = \dots, -2, -1, 0, 1, 2, \dots$

Una **Serie Temporal** es la realización particular de un PED, conjunto de valores observados de distintas variables (*pbi, consumo, inversión, etc.*) correpon-

dientes a periodos de tiempos consecutivos (el cual tienen la misma amplitud), y en donde la serie tiene un carácter discreto.

La **Caracterización de una Serie Temporal** consiste en encontrar el valor promedio, la varianza, la covarianza y la autocorrelación de una variable.

Podemos definir el primer momento, conocido como el promedio o valor esperado de una serie  $y_t$  está dado por:

$$E(y_t) = \mu = \int_{-\infty}^{\infty} y_t f(y_t) dy_t$$

donde  $f(y_t)$  representa la función de densidad de probabilidad de  $y_t$ . El análogo de la muestra es:

$$\bar{y}_t = \frac{1}{T} \sum_{t=0}^{t=T} y_t$$

De manera similar, el segundo momento, o varianza, es:

$$E[(y_t - \mu)^2] = \sigma^2 = \int_{-\infty}^{\infty} (y_t - \mu)^2 f(y_t) dy_t$$

y su análogo de la muestra es:

$$s^2 = \frac{1}{T} \sum_{t=0}^{t=T} (y_t - \bar{y})^2$$

Recordemos que la covarianza entre dos variables  $u_i$  y  $v_i$  se define como:

$$Cov(u_i, v_i) = E[(u_i - \mu_u)(v_i - \mu_v)]$$

En el análisis de series de tiempo, el valor de una variable en el tiempo  $t$  se puede correlacionar con su valor en el tiempo  $t + j$  para algún entero  $j$ , por lo que vamos a hacer un uso extensivo de la *autocovarianza* define como:

$$\gamma_j = Cov(y_t, y_{t+j}) = E[(y_t - \mu)(y_{t+j} - \mu)]$$

$\gamma_j$  es simétrica, es decir,  $\gamma_j = \gamma_{-j}$ . Se puede estimar  $\gamma_j$  como:

$$\hat{\gamma}_j = \frac{1}{T} \sum_{t=0}^{t=T-j} (y_t - \bar{y})(y_{t+j} - \bar{y})$$

Debido a que las correlaciones son invariantes en escala y por lo tanto más fácil de interpretar que las covarianzas, se define la *Función de Autocorrelación* (ACF) como:

$$\rho_j = \gamma_j / \gamma_0$$

donde  $\gamma_0$  es la autocovarianza de orden cero, que es la varianza.

## 13.2. Estacionariedad

En nuestras definiciones de la media, varianza y autocovarianza, hemos supuesto implícitamente que son independientes del tiempo. Entonces, se dice que una PED es **estacionario** si determinadas propiedades estocásticas (funciones de distribución, momentos entre otras) de  $y_t$  y  $y_{t-k}$  no dependen de  $t$  y  $t-k$  (su ubicación absoluta en la secuencia) pero sólo de  $k$  (su separación relativa en la secuencia).

Existe dos tipos de estacionariedad:

- La **Estacionariedad Estricta** se presenta cuando la función de distribución no cambia a lo largo del tiempo. Un proceso estocástico es estaci  $y_{t1}, y_{t2}, \dots, y_{tn}$  tienen la misma distribución independientemente del valor de  $t$ .
- Mientras que la **Estacionariedad Débil** se presenta cuando sus dos primeros momentos es constante en el tiempo. Es decir, si cumplen con las siguientes tres propiedades:



- *Propiedad 1.* Las esperanzas matemáticas de las variables aleatorias no dependen del tiempo (esperanzas constantes):

$$E[y_t] = 0$$

- *Propiedad 2.* Las varianzas no dependen del tiempo y son finitas:

$$Var[y_t] = \sigma^2 < \infty$$

- *Propiedad 3.* Las covarianzas entre dos periodos de tiempos distintos solamente dependen del lapso de tiempo transcurrido entre esos dos periodos:

$$Cov(y_t, y_{t+j}) = Cov(y_{t'}, y_{t'+j}) = 0 \text{ para todo } t, t', j.$$

Un ejemplo de serie estacionaria en sentido débil es una variable *ruido blanco* ( $\epsilon_t$ ) el cual tiene una media igual a cero, varianza constante y no está serialmente correlacionada:

$$\begin{aligned} E[\epsilon_t] &= 0 \\ Var[\epsilon_t] &= \sigma_\epsilon \\ \gamma_k = Cov(\epsilon_t, \epsilon_{t+s}) &= 0 \text{ para todo } s \end{aligned}$$

```
. *Incluyendo la ruta donde se encuentra el archivo
```

```
. clear all
```

```
. set mem 200m
```

#### Current memory allocation

settable	current value	description	memory usage (1M = 1024k)
set maxvar	5000	max. variables allowed	1.909M
set memory	200M	max. data space	200.000M
set matsize	400	max. RHS vars in models	1.254M
			203.163M

```
. cd "D:\Econometria-Stata\estacionariedad"
D:\Econometria-Stata\estacionariedad
```

```
. *Generando una variable Ruido Blanco con 10,000 observaciones : et= N(0,1)
. set seed 11111
```

```
. set obs 10000
obs was 0, now 10000
```

```

. gen t=_n

. tsset t
    time variable:  t, 1 to 10000
        delta: 1 unit

. gen et=rnormal(0,1)

. sum et

```

Variable	Obs	Mean	Std. Dev.	Min	Max
et	10000	.0056196	1.001315	-3.83599	3.607195

```

. tsline et, yline(0) title("Ruido Blanco . et - N(0,1)")

```

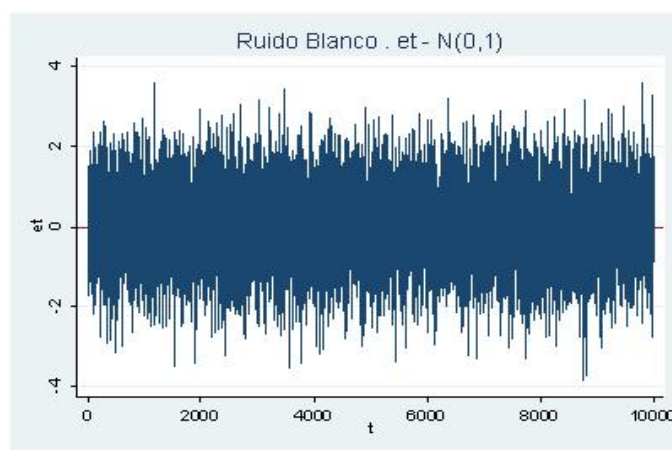


Figura 13.1: Ruido Blanco

Se puede notar que la variable *ruido blanco* tiene un valor promedio cercano a cero (.0056196) y una desviación estándar cercana a la unidad (1.001315). Además, en la gráfica se nota que los valores oscilan alrededor de su promedio.

### 13.3. Procesos Autoregresivos y de Media Móvil

Como se mencionó anteriormente, una característica clave de los datos de series de tiempo es de autocorrelación: el valor de  $y$  en el momento  $t$  es probable que se correlaciona con su valor en los tiempos  $t - 2, t - 1, t + 1$ , y así sucesivamente. En esta sección, se desarrollan tres maneras de modelar este tipo de dependencias.

Los modelos que se desarrollan en esta sección nos permite entender cómo el choque en el tiempo  $t$  influye en los otros períodos  $t + 1, t + 2$ , y así sucesivamente.

### 13.3.1. Procesos de Media Móvil (MA)

Un **proceso de media móvil de primer orden**, escrito MA(1), se describe mediante la ecuación:

$$y_t = \mu + \epsilon_t + \theta\epsilon_{t-1} \quad (13.1)$$

donde  $\epsilon_t$  es un proceso ruido blanco. La media de (10.1) es:

$$E[y_t] = \mu + E[\epsilon_t] + \theta E[\epsilon_{t-1}] = \mu \quad (13.2)$$

Como se ve en (10.2), la media depende solamente de la constante  $\mu$  y no del tiempo. La varianza es:

$$\begin{aligned} \gamma_0 = E[(y_t - \mu)^2] &= E[(\epsilon_t + \theta\epsilon_{t-1})^2] \\ &= E[\epsilon_t^2 + \theta\epsilon_t\epsilon_{t-1} + \theta^2\epsilon_{t-1}^2] \\ &= (1 + \theta^2)\sigma^2 \end{aligned} \quad (13.3)$$

La autocovarianza de orden 1 es:

$$\gamma_1 = E[(\epsilon_t + \theta\epsilon_{t-1})(\epsilon_{t+1} + \theta\epsilon_t)] = \theta\sigma^2 \quad (13.4)$$

Así que la primera Autocorrelación es:

$$\rho_1 = \frac{\gamma_1}{\gamma_0} = \frac{\theta\sigma^2}{(1 + \theta^2)\sigma^2} = \frac{\theta}{(1 + \theta^2)} \quad (13.5)$$

La autocovarianza de segundo orden es:

$$\gamma_2 = E[(\epsilon_t + \theta\epsilon_{t-1})(\epsilon_{t+2} + \theta\epsilon_{t+1})] = 0 \quad (13.6)$$

Y la Autocorrelación de segundo orden  $\rho_2 = 0$ .

Para explorar las propiedades de un proceso  $MA(1)$ , vamos a simular algunos datos. Supongamos que queremos una muestra de  $T = 10000$  observaciones del proceso  $MA(1)$ :

$$y_1 = 1 + \epsilon_t + 0,5\epsilon_{t-1}$$

Para una variable ruido blanco con media 0 y varianza 1, usamos la función **rnormal()**, además establecemos una semilla de 11111 para que se obtenga los mismos resultados.

El proceso  $MA(1)$  depende de valores anteriores de  $\epsilon_t$ , así que necesitamos tener cuidado. Para  $t = 1$ , necesitamos  $\epsilon_0$ . dado que  $\epsilon_t$  es un ruido blanco, podemos calcular un valor aleatoria para  $\epsilon_0$  y usarlo para calcular  $y_1$ . Entonces, tipeamos lo siguiente:

```
. *PROCESO MÉDIA MÓVIL
. *=====

. *Proceso MA(1)

. clear

. set obs 10000
obs was 0, now 10000

. set seed 11111

. gen t=_n

. tsset t
      time variable:  t, 1 to 10000
              delta:  1 unit

. gen et=rnormal(0,1)

. sum et, d
```

et				
	Percentiles	Smallest		
1%	-2.344029	-3.83599		
5%	-1.636566	-3.714058		
10%	-1.26873	-3.5222	Obs	10000
25%	-.669957	-3.500743	Sum of Wgt.	10000
50%	.0031478		Mean	.0056196
		Largest	Std. Dev.	1.001315
75%	.675252	3.261993		
90%	1.301161	3.429939	Variance	1.002631
95%	1.66943	3.600897	Skewness	-.0063962
99%	2.316044	3.607195	Kurtosis	3.010815

```
. gen yt=.
(10000 missing values generated)

. replace yt=et in 1
(1 real change made)

. global mu=1

. global theta=0.5

. forvalues i=2(1)10000 {
  2. quietly replace yt=$mu+et+$theta*et[`i'-1] in `i'
  3. }
//
```

```
. *Caracterizamos la Serie temporal yt
. *-----

. sum yt, d
```

yt				
	Percentiles	Smallest		
1%	-1.592231	-3.587869		
5%	-.8197443	-3.336658		
10%	-.4194317	-3.129482	Obs	10000
25%	.2584961	-2.583009	Sum of Wgt.	10000
50%	1.002547		Mean	1.008289
		Largest	Std. Dev.	1.116081
75%	1.764957	4.689604		
90%	2.446433	4.713771	Variance	1.245636
95%	2.852232	4.738758	Skewness	.0100336
99%	3.599518	4.933245	Kurtosis	2.969727

```
. *media:
. disp "E(yt)=" $mu
E(yt)=1

. *varianza:
. disp "Var(yt)=" (1+$theta^2)
Var(yt)=1.25

. *Autocovarianza de Primer Orden:
. disp "Ac1=" $theta
Ac1=.5
```

```

. *Autocorrelación de Primer Orden:
. disp "Rho1=" $theta/(1+$theta^2)
Rho1=.4

. *Autocorrelación de Orden Superior:
. cor F2.yt F1.yt yt L.yt L2.yt
(obs=9996)

```

	F2. yt	F. yt	yt	L. yt	L2. yt
yt					
F2.	1.0000				
F1.	0.3990	1.0000			
--.	0.0071	0.3990	1.0000		
L1.	-0.0033	0.0072	0.3990	1.0000	
L2.	-0.0068	-0.0034	0.0072	0.3990	1.0000

```

. *graficamos el Proceso MA(1)
. line yt t, name(ma1,replace) yline($mu) title("Proceso MA(1)")

```

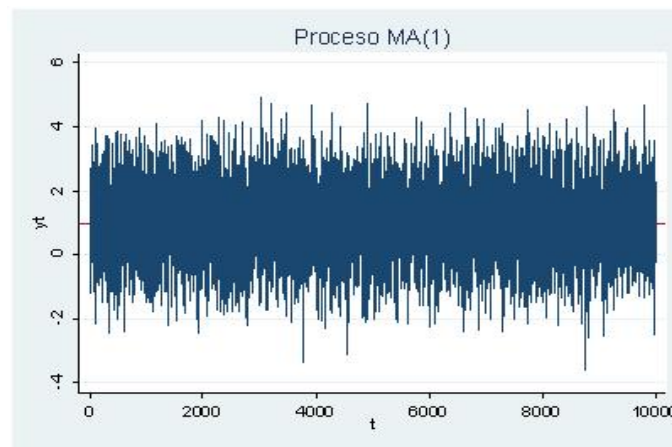


Figura 13.2: Proceso Media Móvil - MA

De la expresión (10.2), el valor esperado de  $y_t$  es 1, y la media muestral es cercano a este valor (1.008289). La ecuación (10.3) indica la varianza de  $y_t$  es  $(1 + 0,5^2) \cdot 1 = 1,25$  cercano también a 1.245636. En tanto, la correlación de primer orden  $\rho_1 = 0,5/(1 + 0,5^2) = 0,4$ , y la autocorrelación muestral esta alrededor de 0.399. La autocorrelación de segundo orden es cero, y se observa en el último cuadro que las correlaciones superiores de primer orden son casi nulas.

Además, en el gráfico se puede observar el proceso MA(1) para  $y_t$  que oscila alrededor de su media.

En general, un Proceso de Media Móvil de orden  $q$   $MA(q)$  es:

$$y_t = \mu + \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \dots + \theta_q \epsilon_{t-q}$$

La media es  $\mu$ , mientras que la varianza es:

$$\gamma_0 = E[(y_t - \mu)^2] = (1 + \theta_1^2 + \theta_2^2 + \dots + \theta_q^2) \sigma^2 = \left( \sum_{i=0}^q \theta_i^2 \right) \sigma^2$$

donde  $\theta_0 \equiv 1$ , y las autocovarianzas son:

$$\begin{aligned} \gamma_1 &= (\theta_1 + \theta_1 \theta_2 + \theta_2 \theta_3 + \dots + \theta_{q-1} \theta_q) \sigma^2 \\ \gamma_2 &= (\theta_2 + \theta_1 \theta_3 + \theta_2 \theta_4 + \dots + \theta_{q-2} \theta_q) \sigma^2 \\ \vdots &= \vdots \\ \gamma_q &= \theta_q \sigma^2 \\ \gamma_m &= 0, m > q \end{aligned}$$

tanto la media, la varianza y autocovarianza son todos independientes de  $t$ , así que concluimos que un proceso  $MA(q)$ . La  $\gamma_m = 0$  para  $m$  mayor a  $q$  quiere decir que si los shocks ocurren en el tiempo  $t$  para un proceso  $MA(q)$ , estos shocks afectan a la serie en el periodo  $t, t+1, t+2, \dots, t+q$ , pero no tiene efecto para el periodo  $t+q+1$  en adelante.

### 13.3.2. Procesos Autoregresivos (AR)

Un *Proceso Autoregresivo de Primer Orden*, simbolizado como  $AR(1)$ , se escribe de la siguiente manera:

$$y_t = \beta + \phi y_{t-1} + \epsilon_t \quad (13.7)$$

donde  $\epsilon_t$  es un proceso ruido blanco,  $\beta$  y  $\phi$  son parámetros. Note que la variable  $y_t$  depende de su valor pasado  $t - 1$ .

¿Cuál es la emdia descrito en (10.7)?. Tenemos lo siguiente:

$$\begin{aligned} E[y_t] &= \beta + \phi E[y_{t-1}] + E[\epsilon_t] \\ &= \beta + \phi E[y_{t-1}] \end{aligned} \quad (13.8)$$

Si asumimos que  $y_t$  es estacionaria, entonces,  $E[y_t] = E[y_{t-1}] = \mu$ , así que se puede escribir (10.8) como:

$$\mu = \beta + \phi\mu$$

Despejando  $\mu$ , se tiene:

$$\mu = \frac{\beta}{1-\phi}$$

Sin embargo, esta derivación es algo engañoso, ya que hace un supuesto implícito acerca de  $\phi$ . Usando *sustituciones recursivas*, podemos reescribir la ecuación (10.7) como:

$$\begin{aligned} y_t &= \beta + \phi y_{t-1} + \epsilon_t \\ &= \beta + \phi(\beta + \phi y_{t-2} + \epsilon_{t-1}) + \epsilon_t \\ &= (1 + \phi)\beta + \phi^2(\beta + \phi y_{t-3} + \epsilon_{t-2}) + \epsilon_t + \phi\epsilon_{t-1} \\ &= (1 + \phi + \phi^2)\beta + \phi^3(\beta + \phi y_{t-4} + \epsilon_{t-3}) + \epsilon_t + \phi\epsilon_{t-1} + \phi^2\epsilon_{t-2} \\ &= \vdots \\ &= (1 + \phi + \phi^2 + \dots)\beta + \epsilon_t + \phi\epsilon_{t-1} + \phi^2\epsilon_{t-2} + \dots \end{aligned} \quad (13.9)$$

Recordando de la siguiente secuencia:



$$1 + \phi + \phi^2 + \dots = \frac{1}{1-\phi}$$

Siempre que  $|\phi| < 1$ . Entonces, podemos reescribir la expresión (10.9):

$$y_t = \frac{\beta}{1-\phi} + \epsilon_t + \phi\epsilon_{t-1} + \phi^2\epsilon_{t-2} + \dots \quad (13.10)$$

el cual se simplifica a un proceso de media móvil con  $\mu = \frac{\beta}{1-\phi}$ .

A partir de ahora, asumiremos que  $|\phi| < 1$ , por tanto, podemos caracterizar la serie  $y_t$  como:

$$y_t = \frac{\beta}{1-\phi} \quad (13.11)$$

$$\gamma_0 = E[(y_t - \mu)^2] = (1 + \phi^2 + \phi^4 + \dots)\sigma^2 = \frac{\sigma^2}{1-\phi^2} \quad (13.12)$$

La Autocovarianza de primer orden es:

$$\begin{aligned} E[(y_t - \mu)(y_{t+1} - \mu)] &= E[(\epsilon_t + \phi\epsilon_{t-1} + \phi^2\epsilon_{t-2} + \dots)(\epsilon_{t+1} + \phi\epsilon_t + \phi^2\epsilon_{t-1} + \dots)] \\ &= (\phi + \phi^3 + \phi^5 + \dots)\sigma^2 \\ &= \frac{\phi}{1-\phi^2}\sigma^2 \end{aligned}$$

así que:

$$\rho_1 = \phi$$

La Autocovarianza de segundo orden es:

$$\begin{aligned}
E[(y_t - \mu)(y_{t+2} - \mu)] &= E[(\epsilon_t + \phi\epsilon_{t-1} + \phi^2\epsilon_{t-2} + \dots)(\epsilon_{t+2} + \phi\epsilon_{t+1} + \phi^2\epsilon_t + \dots)] \\
&= (\phi^2 + \phi^4 + \phi^6 + \dots)\sigma^2 \\
&= \frac{\phi^2}{1 - \phi^2}\sigma^2
\end{aligned}$$

asi que:

$$\rho_2 = \phi^2$$

Se puede ver un patron en  $\rho_1$  y  $\rho_2$ . Por lo tanto,  $\rho_j = \phi^j$  para un proceso  $AR(1)$ .

Ahora, simulemos el siguiente proceso  $AR(1)$  con un muestra de  $T = 10000$  valores:

$$\begin{aligned}
y_t &= 10 + 0,7y_{t-1} + \epsilon_t \\
\epsilon_t &\sim N(0, 2)
\end{aligned}$$

En este caso  $\epsilon$  tiene varianza 2. En primer lugar, sabemos que la media condicional es:

$$\mu = \frac{\beta}{1-\phi} = \frac{10}{1-0,7} \approx 33,33$$

asi que esta cifra lo podemos usar para el primer valor de  $y_t$  ( $y_0$ ).

```

. *PROCESO AUTOREGRESIVO
. *=====

. *Proceso AR(1)

. clear

. set obs 10000
obs was 0, now 10000

. set seed 11111

```

```

. gen t=_n

. tsset t
    time variable:  t, 1 to 10000
        delta: 1 unit

. gen et=rnormal(0,sqrt(2))

. sum et, d

              et
-----
Percentiles      Smallest
 1%    -3.314957    -5.42491
 5%    -2.314455    -5.252471
10%    -1.794255    -4.981143      Obs          10000
25%    -.9474622    -4.950798      Sum of Wgt.    10000
50%     .0044516                                Mean          .0079473
                                Largest          Std. Dev.    1.416073
75%     .9549506      4.613155
90%     1.840119      4.850666      Variance        2.005263
95%     2.360931      5.092437      Skewness         -.0063962
99%     3.275381      5.101345      Kurtosis         3.010815

. global sigma=2
. global beta=10
. global phi=0.7
. global mu=$beta/(1-$phi)

. gen yt=.
(10000 missing values generated)
. replace yt=$mu in 1
(1 real change made)

. quietly replace yt=$beta+$phi*L.yt+et in 2/l

. *Caracterizamos la Serie temporal yt
. *-----

. sum yt, d

              yt
-----
Percentiles      Smallest
 1%    28.76227     25.98178
 5%    30.11402     26.1061
10%    30.82553     26.80658      Obs          10000
25%    32.00679     26.89886      Sum of Wgt.    10000
50%    33.37155                                Mean          33.35945
                                Largest          Std. Dev.    1.97937
75%    34.70212     39.83002
90%    35.89807     39.96432      Variance        3.917904
95%    36.65289     40.10463      Skewness         -.0110678
99%    37.90582     40.1847      Kurtosis         2.908804

. *media:
. disp "E(yt)=" $beta/(1-$phi)
E(yt)=33.333333

. *varianza:

```

```

. disp "Var(yt)=" $sigma/(1-$phi^2)
Var(yt)=3.9215686

. *Autocovarianza de Primer Orden:
. disp "Ac1=" $sigma*$phi/(1-$phi^2)
Ac1=2.745098

. *Autocorrelación de Primer Orden:
. disp "Rho1=" $phi
Rho1=.7

. *Autocorrelación de Segundo Orden:
. disp "Rho2=" $phi^2
Rho2=.49

. *Autocorrelación de Orden Superior:
. cor F2.yt F1.yt yt L.yt L2.yt
(obs=9996)

```

	F2. yt	F. yt	yt	L. yt	L2. yt
yt	1.0000				
F2.	1.0000				
F1.	0.6987	1.0000			
--.	0.4929	0.6987	1.0000		
L1.	0.3428	0.4929	0.6987	1.0000	
L2.	0.2372	0.3428	0.4929	0.6987	1.0000

```

. *graficamos el Proceso AR(1)
. line yt t, name(ar1,replace) yline($mu) title("Proceso AR(1)")

```

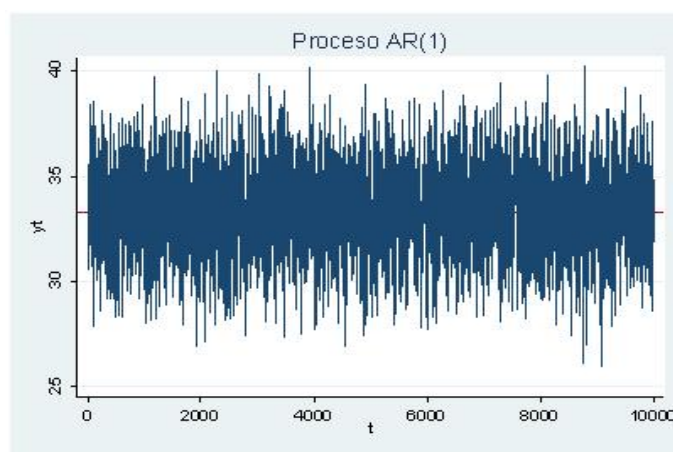


Figura 13.3: Proceso Autoregresivo - AR

El valor promedio muestral (33.35945) está cercano al valor poblacional (33.33). Mientras que la varianza poblacional es  $2/(1 - 0.7^2) \approx 3.92$  y la muestral es 3.917904.

Con respecto a las correlaciones estimadas ( $\rho_1 = 0,6987$  y  $\rho_2 = 0,4929$ ) son muy parecidas a las poblacionales 0,7 y 0,49 respectivamente.

Un *Proceso Autoregresivo de Segundo orden AR(2)* puede ser escrito como:

$$y_t = \beta + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \epsilon_t \quad (13.13)$$

La forma más sencilla de encontrar la media es asumir que  $y_t$  es estacionaria y se toma el valor esperado:

$$E[y_t] = E[\beta + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \epsilon_t]$$

$$\begin{aligned} \mu &= \beta + \phi_1 \mu + \phi_2 \mu \\ &= \frac{\beta}{1 - \phi_1 - \phi_2} \end{aligned} \quad (13.14)$$

La varianza es:

$$\gamma_0 = E[(y_t - \mu)^2] = \frac{(1 - \phi_2)\sigma^2}{(1 + \phi_2)[(1 - \phi_2)^2 - \phi_1^2]}$$

y las correlaciones son:

$$\begin{aligned} \rho_1 &= \frac{\phi_1}{1 - \phi_2} \\ \rho_2 &= \phi_1 \rho_1 + \phi_2 \\ \rho_j &= \phi_1 \rho_{j-1} + \phi_2 \rho_{j-2}, \text{ para } j > 2 \end{aligned}$$

Generalizando un *Proceso Autoregresivo de Orden p AR(p)*, este se define como:

$$y_t = \beta + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + \epsilon_t$$

Las formulas de varianza y autocorrelaciones son un pocos complejas en este caso<sup>1</sup>.

### 13.3.3. Procesos Autoregresivos y Medias Móviles (AR-MA)

Estos procesos estan compuesto por una parte Autoregresiva (AR) y otra de Medias Móviles (MA), el cual nos permite obtener modelos más parsimoniosos.

Un ejemplo sencillo es el proceso ARMA(1,1), el cual posee un termino autoregresivo y de media móvil de primer orden en ambos.

$$y_t = \beta + \phi_1 y_{t-1} + \epsilon_t + \theta_1 \epsilon_{t-1}$$

La media de este proceso es similar al modelo  $AR(1)$ :

$$E[y_t] = \mu = \frac{\beta}{1-\phi}$$

La varianza esta dado por la siguiente expresión:

$$\gamma_0 = E[(y_t - \mu)^2] = \frac{(1 + 2\phi\theta + \theta^2)\sigma^2}{1 - \phi^2}$$

Mientras que la Autocovarianza de primer orden es:

$$\gamma_1 = \theta\sigma^2 + \phi\gamma_0$$

---

<sup>1</sup>Ver Hamilton (1994, capítulo 3.4) para más detalles.

Para ordenes superiores, la Autocovarianza se clacula de la siguiente forma:

$$\gamma_k = \phi \gamma_{k-1}, k > 1$$

Por lo tanto, la Autocorrlación se obtiene como:

$$\begin{aligned}\rho_1 &= \frac{\theta\sigma^2}{\gamma_0} + \phi \\ \rho_k &= \phi \rho_{k-1}, k > 1\end{aligned}$$

Para este caso, simulemos el siguiente proceso  $ARMA(1, 1)$  con un muestra de  $T = 10000$  valores, basandose en los dos ejemplos anteriores:

$$\begin{aligned}y_t &= 10 + 0,7y_{t-1} + \epsilon_t + +0,5\epsilon_{t-1} \\ \epsilon_t &\sim N(0, 2)\end{aligned}$$

```
. *PROCESO AUTOREGRESIVO Y DE MEDIA MÓVIL
. *=====

. *Proceso ARMA(1,1)

. clear

. set obs 10000
obs was 0, now 10000

. set seed 11111

. gen t=_n

. tsset t
      time variable:  t, 1 to 10000
      delta: 1 unit

. gen et=rnormal(0,sqrt(2))

. sum et, d
```

```

                                et
-----
      Percentiles      Smallest
  1%      -3.314957      -5.42491
  5%      -2.314455      -5.252471
 10%      -1.794255      -4.981143
 25%      -.9474622      -4.950798
 50%       .0044516
                        Largest
 75%       .9549506      4.613155
 90%       1.840119      4.850666
 95%       2.360931      5.092437
 99%       3.275381      5.101345
                                Obs      10000
                                Sum of Wgt. 10000
                                Mean      .0079473
                                Std. Dev. 1.416073
                                Variance  2.005263
                                Skewness  -.0063962
                                Kurtosis  3.010815

. global sigma=2

. global beta=10

. global phi=0.7

. global theta=0.5

. global mu=$beta/(1-$phi)

. gen yt=.
(10000 missing values generated)

. replace yt=$mu in 1
(1 real change made)

. quietly replace yt=$beta+$phi*L.yt+et+$theta*L.et in 2/1

. *Caracterizamos la Serie temporal yt
. *-----

. sum yt, d

                                yt
-----
      Percentiles      Smallest
  1%       27.00448      23.19296
  5%       28.82331      23.79176
 10%       29.77382      24.31368
 25%       31.47618      24.56132
 50%       33.40345
                        Largest
 75%       35.24392      41.90961
 90%       36.91316      41.99406
 95%       37.97047      42.40366
 99%       39.79942      42.65474
                                Obs      10000
                                Sum of Wgt. 10000
                                Mean      33.37256
                                Std. Dev. 2.763146
                                Variance  7.634977
                                Skewness  -.0087895
                                Kurtosis  2.888375

. *media:
. disp "E(yt)=" $beta/(1-$phi)
E(yt)=33.33333

. *varianza:
. disp "Var(yt)=" ($sigma*(1+2*$phi*$theta+$theta^2))/(1-$phi^2)
Var(yt)=7.6470588

```



```

. *Autocovarianza de Primer Orden:
. disp "Ac1=" $theta*$sigma + $phi*(1+2*$phi*$theta+$theta^2)*$sigma/(1-$phi^2)
Ac1=6.3529412

. *Autocorrelación de Primer Orden:
. disp "Rho1=" $phi + $theta*$sigma/((1+2*$phi*$theta+$theta^2)*$sigma/(1-$phi^2))
Rho1=.83076923

. *Autocorrelación de Orden Superior:
. cor F2.yt F1.yt yt L.yt L2.yt
(obs=9996)

```

	F2. yt	F. yt	yt	L. yt	L2. yt
yt					
F2.	1.0000				
F1.	0.8312	1.0000			
--.	0.5834	0.8312	1.0000		
L1.	0.4072	0.5834	0.8312	1.0000	
L2.	0.2849	0.4072	0.5834	0.8312	1.0000

```

. *graficamos el Proceso AR(1)
. line yt t, name(arma1,replace) yline($mu) title("Proceso ARMA(1,1)")

```

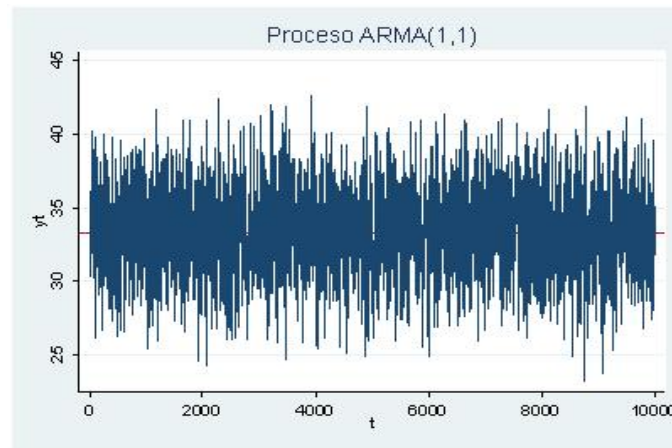


Figura 13.4: Proceso Autoregresivo de Media Móvil - ARMA

Se puede observar que el valor promedio muestral (33,37256) está cercano al valor poblacional (33.33). Mientras que tanto la varianza poblacional (7,6470588) y la muestral (7,634977) están cercanos.

Con respecto a las correlación estimadas ( $\rho_1 = 0,8312$  es muy parecida a la poblacional 0,83076923).

Un proceso mas general es el  $ARMA(p, q)$ :

$$y_t = \mu + \phi_1 y_{t-1} + \dots + \phi_p y_{t-p} + \epsilon_t + \theta_1 \epsilon_{t-1} + \dots + \theta_q \epsilon_{t-q}$$

Esencialmente el proceso  $ARMA$  nos permite capturar la dinámica de nuestra serie utilizando pocos parámetros, en vez de emplear un proceso  $AR$  o  $MA$ , haciendo que el modelo final cumpla con la propiedad de *parsinomia*.

## 13.4. Función de Autocorrelación Muestral (FAS) y Parcial (FAP)

Como se ilustró en la sección anterior, un factor que distingue entre un *proceso autoregresivo* y un *proceso de media móvil* es la manera en que los shocks afectan a las futuras realizaciones de una serie.

En un proceso  $MA(q)$ , un shock en el periodo  $t$  no tiene efecto alguno sobre la serie en el periodo  $t + q + 1$  en adelante. Sin embargo, en un proceso  $AR(p)$ , el efecto de un shock decae gradualmente a través del tiempo.

Las dos herramientas ( $FAS$  y  $FAP$ ) que describiremos en esta sección nos permite ver el impacto del shock y diferenciar en base a sus propiedades si es un proceso autoregresivo o de media móvil, además de encontrar el orden de cada proceso.

### 13.4.1. Función de Autocorrelación Muestral (FAS)

La *Función de Autocorrelación Muestral (FAS)* en el rezago  $j$  ( $\gamma_j$ ) se define como:

$$\rho_j = \frac{\gamma_j}{\gamma_0}$$

donde  $\gamma_0$  es la varianza.

Para ilustrar esta herramienta, simularemos una muestra de 1000 observaciones de un proceso  $MA(1)$ ,  $y_t = \epsilon_t + 0,7\epsilon_t$ , donde  $\epsilon$  es un ruido blanco con media 0 y varianza 1. Para esto utilizaremos en comando **sim\_arma**

```
. *FUNCIÓN DE AUTOCORRELACIÓN MUESTRAL (FAS) Y PARCIAL (FAP)
. *=====

. *FUNCIÓN DE AUTOCORRELACIÓN MUESTRAL (FAS)
. *-----

. *proceso MA(1): y(t) = e(t) + 0.7e(t-1)

. clear all

. set seed 11111

. sim_arma y, ma(0.7) sigma(1) nobs(1000) time(t)
```

Este comando genera una nueva variable  $y$ , con un componente **ar(0.7)** donde el coeficiente que acompaña al primer rezago es 0,7. La opción **sigma(1)** especifica una desviación estándar igual a 1 para el proceso ruido blanco  $\epsilon : t$ , **nobs(1000)** indica que se desea 1000 observaciones en nuestra base de datos y **time(t)** genera una nueva variable  $t$  que indica la frecuencia temporal (de 1 hasta 1000).

De la expresión (10,5), la autocorrelación de primer orden es  $0,7/(1 + 0,7^2) = 0,47$ . Además,  $\rho_2 = \rho_3 = \dots = 0$ . Para dibujar las autocorrelaciones podemos usar el comando **ac**:

```
. ac y , title("FAS MA(1)")
```

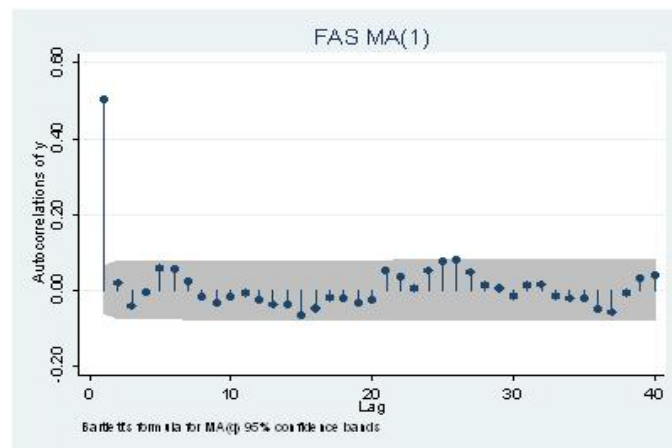


Figura 13.5: FAS para un Proceso  $MA(1)$

La autocorrelación con un rezago es cercano a la poblacional 0.47, y las demás autocorrelaciones son aproximadamente iguales a 0. Estos no son idénticamente a 0 porque estamos trabajando con una muestra en lugar de la población. Pero si la muestra hubiese sido mas grande, entonces, estos se acercarían al valor poblacional. Los intervalos de confianza de las bandas se deriva de Davis(2002).

Ahora, consideremos un proceso  $AR(1)$ :

$$y_t = 0,95y_{t-1} + \epsilon_t$$

$$\epsilon \sim (0, 1)$$

```
. *proceso AR(1): y(t) = 0.95y(t-1) + e(t)
. clear all
. set seed 11111
. sim_arma y, ar(0.95) sigma(1) nobs(1000) time(t)
. ac y , title("FAS AR(1)")
```

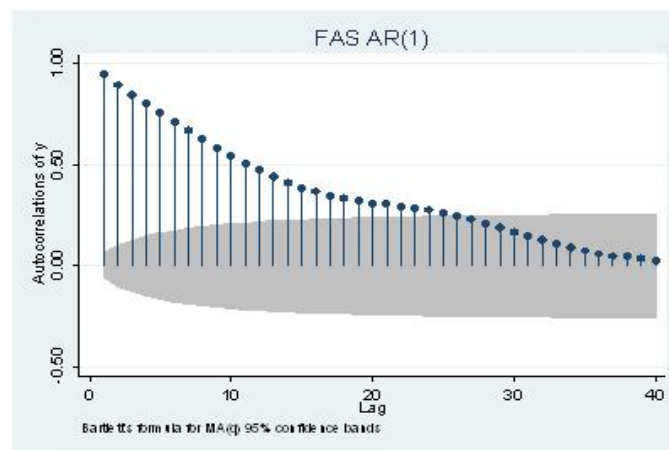


Figura 13.6: FAS para un Proceso  $AR(1)$

En este caso se observa que un shock ocurrido en el periodo  $t$  tiene un efecto persistente y cuyo impacto decrece gradualmente en los periodos futuros.

¿Que sucede si el término  $AR$  es negativo, es decir,  $\phi < 0$ ?

```

. *¿Que sucede si phi<0?

. *proceso AR(1): y(t) = -0.75y(t-1) + e(t)
. clear all

. set seed 11111

. sim_arma y, ar(-0.75) sigma(1) nobs(1000) time(t)

. ac y , title("FAS AR(1) con phi<0")

```

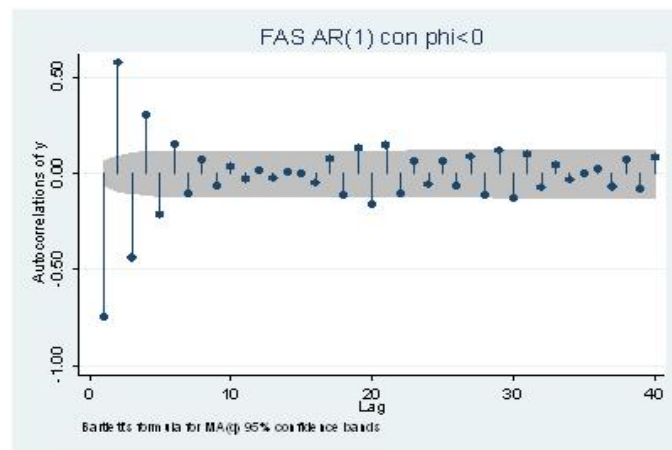


Figura 13.7: FAS para un Proceso AR(1) con  $\Phi < 0$

Con un valor negativo de  $\phi$ , un shock positivo en el periodo  $t$  impactaría negativamente en el periodo  $t+1$ . dado que el impacto en el periodo  $t+1$  es negativo, en el periodo  $t+2$  el impacto es positivo, y así sucesivamente. Estas autocorrelaciones muestran un patrón de *oscilación amortiguada*.

Finalmente , exploremos un proceso *ARMA*:

$$y_t = 0,5y_{t-1} + \epsilon_t + 0,5\epsilon_{t-1}$$

$$\epsilon \sim (0, 1)$$

```

. *proceso ARMA(1,1): y(t) = 0.5y(t-1) + e(t) + 0.5e(t-1)

. clear all

```

```

. set seed 11111

. sim_arma ar1 , ar(0.5) nobs(1000)
. sim_arma ma1 , ma(0.5) nobs(1000)
. sim_arma arma11 , ar(0.5) ma(0.5) nobs(1000)

. ac ar1, gen(ar1_ac)
. label var ar1 "AR(1)-only AC"
. ac ma1, gen(ma1_ac)
. label var ma1 "MA(1)-only AC"
. ac arma11, gen(arma11_ac)
. label var arma11 "ARMA(1,1) AC"

. tsline ar1_ac ma1_ac arma11_ac in 1/20 , title("FAS ARMA(1,1)")

```

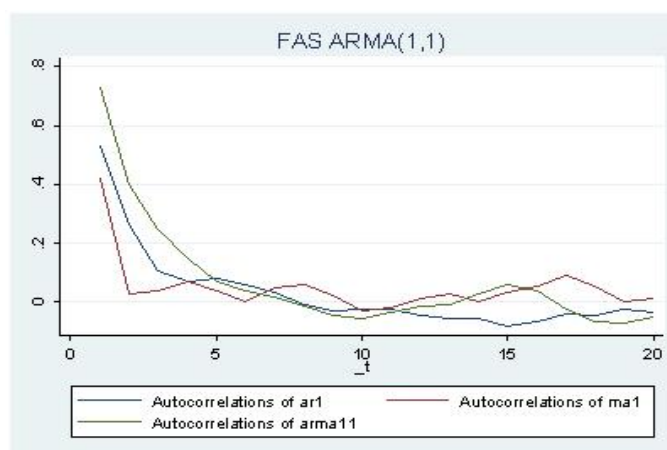


Figura 13.8: FAS para un Proceso ARMA(1,1)

Las autocorrelaciones para el proceso *ARMA* comienzan con valores altos y decrecen rápidamente a comparación de un proceso *AR*.

### 13.4.2. Función de Autocorrelación Parcial (FAP)

Hemos notado que las autocorrelaciones para un proceso autoregresivo decaen gradualmente, pero usando la función de autocorrelación para un proceso *AR* no permite detectar de qué orden es dicho proceso, es decir, si es un *AR*(1) o un *AR*(5).

La *Función de Autocorrección Parcial (FAP)* remedia esto, la motivación surge en saber si sigue proceso de orden  $p$  ( $AR(p)$ ), entonces las autocorrelaciones parciales de los rezagos  $p + 1$  en adelante son iguales a 0 en la población. La *FAP* mide la correlación entre  $y_t$  y  $y_{t+j}$  después de controlar el efecto de  $y_{t+1}, y_{t+2}, \dots, y_{t+j-1}$ . Desde una perspectiva de regresión, las autocorrelaciones parciales  $\phi_{11}, \phi_{22}, \dots, \phi_{jj}$  son los coeficiente de la siguiente regresión:

$$(y_t - \mu) = \phi_{11}(y_{t-1} - \mu) + \phi_{22}(y_{t-2} - \mu) + \dots + \phi_{jj}(y_{t-j} - \mu) + \epsilon_t$$

donde  $\epsilon_t$  es un ruido blanco.

A nivel poblacional, la primera FAS es igual a la primera FAP ( $\rho_1 = \phi_{11}$ ). Si bien es cierto, la técnica basada en la regresión ha demostrado que funciona bien para calcular el FAP, sin embargo, esto no garantiza que la primera FAS sea igual a la primera FAP<sup>2</sup>.

En STATA, el comando **pac** nos permite obtener la *FAP*. Ilustremos un ejemplo para un proceso  $AR(3)$  para una muestra de  $T = 10000$  observaciones:

$$y_t = 0,7y_{t-1} + 0,4y_{t-2} - 0,3y_{t-3}$$

```
. *FUNCIÓN DE AUTOCORRELACIÓN PARCIAL (FAP)
. *-----

. *proceso MA(3) : y(t) = 0.7y(t-1) + 0.4y(t-2) - 0.3y(t-3)

. clear

. set seed 11111

. sim_arma y, ar(0.7 0.4 -0.3) nobs(10000)

. pac y , title("FAP AR(3)")
```

Note que las tres primeras FAP son estadísticamente significativos, mientras que los demás son iguales a cero.

<sup>2</sup>Existe una opción para calcular el FAP a través de las ecuaciones de Yule-Walker. Este método garantiza que la primera FAS sea igual a la primera FAP, pero los resultados simulados para calcular el FAP por el método de Yule-Walker puede estar seriamente sesgado.

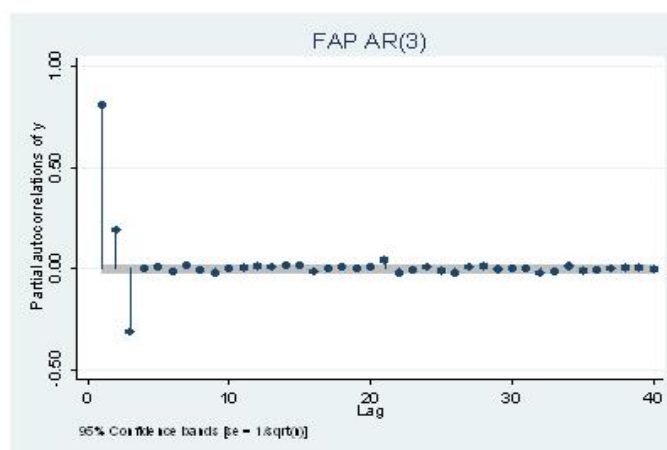


Figura 13.9: FAP para un Proceso AR

Como se indicó en el capítulo referente al tema de *Autocorrelación*, el comando **corrgram** muestra la *FAS* y *FAP* de manera conjunta. Si utilizamos este comando para los primeros 20 rezagos, tendremos lo siguiente:

```
. corrgram y , lags(20)
```

LAG	AC	PAC	Q	Prob>Q	-1	0	1	-1	0	1
					[Autocorrelation]			[Partial Autocor]		
1	0.8131	0.8133	6613	0.0000						
2	0.7286	0.1998	11924	0.0000						
3	0.5357	-0.3098	14796	0.0000						
4	0.4218	-0.0015	16576	0.0000						
5	0.2915	0.0067	17426	0.0000						
6	0.2081	-0.0138	17860	0.0000						
7	0.1379	0.0161	18050	0.0000						
8	0.0899	-0.0074	18131	0.0000						
9	0.0489	-0.0243	18155	0.0000						
10	0.0218	-0.0015	18159	0.0000						
11	0.0022	0.0053	18160	0.0000						
12	-0.0051	0.0126	18160	0.0000						
13	-0.0080	0.0068	18160	0.0000						
14	-0.0022	0.0134	18160	0.0000						
15	0.0071	0.0171	18161	0.0000						
16	0.0115	-0.0153	18162	0.0000						
17	0.0179	0.0022	18166	0.0000						
18	0.0210	0.0093	18170	0.0000						
19	0.0244	0.0014	18176	0.0000						
20	0.0267	0.0069	18183	0.0000						



## 13.5. Ejercicio Propuesto

1. Generar de forma manual una serie con  $T = 8000$  observaciones que sigan los siguientes procesos:

$$MA(2) : y_t = \epsilon_t + 0,3\epsilon_{t-1} + 0,7\epsilon_{t-2} \\ \epsilon_t \sim N(0, 1)$$

$$AR(2) : y_t = 5 + \phi_t + 0,3\phi_{t-1} + 0,7\phi_{t-2} \\ \epsilon_t \sim N(0, 1)$$

$$ARMA(2, 2) : y_t = 5 + \phi_t + 0,3\phi_{t-1} + 0,7\phi_{t-2} + \epsilon_t + 0,3\epsilon_{t-1} + 0,7\epsilon_{t-2} \\ \epsilon_t \sim N(0, 1)$$

Además, compare los resultados de la media, varianza, autocovarianza y autocorrelación de primer y segundo orden poblacional con la muestral.

2. Simule una serie de  $T = 5000$  observaciones para cada uno de los siguientes procesos y grafica las funciones de autocorrelación muestral y parcial:
  - AR(1) con  $\phi_1 = 0,95$ . ¿Qué sucede si  $\phi_1 = -0,95$ ?
  - MA(1) con  $\theta_1 = 0,95$ . ¿Qué sucede si  $\theta_1 = -0,95$ ?
  - ARMA(1,1) con  $\phi_1 = 0,95$  y  $\theta_1 = 0,95$ . ¿Qué sucede si tienen signos contrarios ambos parámetros?

# Capítulo 14

## Procesos Estocásticos No Estacionarios

Una *serie no estacionaria* se puede deber a:

### 1. Serie No Estacionaria en Media

- Presencia de una tendencia estocástica.
- Presencia de una tendencia determinística.

### 2. Serie No Estacionaria en Varianza

- Presencia de una varianza heteroscedástica.

Para el presente capítulo se desarrollará el tema de *Series No Estacionarias en Media*.

### 14.1. Serie No Estacionaria en Media

En la realidad, casi siempre se trabaja con variables económicas que no son estacionarias. Es decir, a medida que transcurre el tiempo esta variable se va alejando

de su valor promedio sin tener una trayectoria definida. La diferencia entre procesos estacionarios y no estacionarios es saber si la evolución a largo plazo (*tendencia*) de las series observadas es **determinística** o **estocástico**.

Un proceso es *determinístico* si la tendencia puede ser predecible y constante. Mientras que, un proceso *estocástico* no puede ser predecible. A estos últimos tipos de procesos se le conoce en la literatura como **Caminata Aleatoria** o **Random Walk**<sup>1</sup>.

Hay que diferenciar dos tipos de procesos Random Walk:

1. Caminata Aleatoria sin Variaciones o simplemente Random Walk.
2. Caminata Aleatoria con Variaciones o Random Walk with Drift.

### 14.1.1. Proceso Estacionario de Tendencia Determinística

Si  $\epsilon$  es ruido blanco que se distribuye con media 0 y varianza  $\sigma_\epsilon^2$ . Se dice que  $Y_t$  presenta una tendencia determinística si:

$$Y_t = \beta_0 + \beta_1 t + \epsilon_t \quad (14.1)$$

El valor de  $Y_t$  depende linealmente de  $t$  más un choque aleatorio. Si realizamos la caracterización de esta serie, obtenemos:

$$\begin{aligned} E[Y_t] &= E[\beta_0 + \beta_1 t + \epsilon_t] \\ &= \beta_0 + \beta_1 t \end{aligned} \quad (14.2)$$

---

<sup>1</sup>El término Caminata Aleatoria o Random Walk se compara con el camino de un borracho. Al dejar la cantina, el borracho se mueve una distancia aleatoria  $\epsilon$ , al tiempo  $t$ , y continúa caminando de manera indefinida, con lo cual a la larga se alejará cada vez más de la cantina.

$$\begin{aligned} \text{Var}[Y_t] &= \text{Var}[\beta_0 + \beta_1 t + \epsilon_t] \\ &= \sigma^2 \end{aligned} \quad (14.3)$$

Este tipo de procesos no son estacionarios en media pero si en varianza. Para convertirlo en un proceso estacionario es necesario restar a la serie original su media, como se muestra a continuación:

$$\begin{aligned} Y_t - E[Y_t] &= (\beta_0 + \beta_1 t + \epsilon_t) - (\beta_0 + \beta_1 t) \\ &= \epsilon_t \end{aligned}$$

Por lo tanto:

$$\begin{aligned} E[Y_t - E[Y_t]] &= 0 \\ \text{Var}[Y_t - E[Y_t]] &= \sigma^2 \end{aligned}$$

### 14.1.2. Proceso Estacionario de Tendencia Estocástica

#### Random Walk

Supongase que  $\epsilon$  es un término de error ruido blanco que se distribuye con media cero y varianza  $\sigma_\epsilon^2$ . Entonces, decimos que  $Y_t$  es un random walk si:

$$Y_t = Y_{t-1} + \epsilon_t \quad (14.4)$$

Como se observa en la anterior ecuación, el valor de  $Y$  en el tiempo  $t$  es igual a su valor pasado ( $t - 1$ ) más un choque aleatorio, siendo un modelo  $AR(1)$ .

Si caracterizamos este proceso, tendríamos con respecto a la *media* lo siguiente:

$$\begin{aligned}
Y_1 &= Y_0 + \epsilon_1 \\
Y_2 &= Y_1 + \epsilon_2 = Y_0 + \epsilon_1 + \epsilon_2 \\
Y_3 &= Y_2 + \epsilon_3 = Y_0 + \epsilon_1 + \epsilon_2 + \epsilon_3
\end{aligned}$$

Así sucesivamente hasta obtener la siguiente expresión:

$$Y_t = Y_0 + \Sigma \epsilon_t \quad (14.5)$$

Por lo tanto:

$$\begin{aligned}
E[Y_t] &= E[Y_0 + \Sigma \epsilon_t] \\
&= Y_0 + \Sigma E[\epsilon_t] \\
&= Y_0
\end{aligned} \quad (14.6)$$

Mientras que la varianza de la variable  $Y$  es:

$$\begin{aligned}
Var[Y_t] &= Var[Y_0 + \Sigma \epsilon_t] \\
&= Var[\Sigma \epsilon_t] \\
&= \Sigma Var[\epsilon_t] \\
&= \Sigma \sigma_\epsilon^2 \\
&= t \sigma_\epsilon^2
\end{aligned} \quad (14.7)$$

Según la expresión (14.6), el promedio de  $y$  es igual a su valor inicial, mientras que, la expresión (14.7) muestra que a medida que el tiempo  $t$  se incrementa, la varianza de  $y$  también lo hace. Esto conlleva a decir que la serie  $y$  no cumple con la propiedad de Estacionariedad, ya que la varianza depende del tiempo<sup>2</sup>.

---

<sup>2</sup>A menudo se iguala  $Y_0$  es igual a cero, o mejor dicho,  $E[Y_0] = 0$

Otra característica de las series random walk, es la *persistencia de los choques aleatorios*, tal como se muestran en la ecuación (14.5) donde la serie  $Y_t$  es la suma de su valor inicial ( $Y_0$ ) y con la sumatoria de los errores ( $\epsilon_t$ ). Dado esto, los impactos de los choques no se desvanecen, es por ello, se dice que el proceso random walk tiene *memoria infinita*.

Se obtiene resultados interesantes si la ecuación (14.4) se expresa de la siguiente manera:

$$Y_t - Y_{t-1} = \Delta Y_t = \epsilon_t \quad (14.8)$$

siendo  $\Delta$  el operador de primera diferencia. Sabemos que la serie  $Y_t$  no es estacionaria, pero probaremos a continuación que su primera diferencia si lo es:

$$E[\Delta Y_t] = E[\epsilon_t] = 0 \quad (14.9)$$

$$Var[\Delta Y_t] = Var[\epsilon_t] = \sigma_\epsilon \quad (14.10)$$

Dado que la media y varianza de  $Y_t$  no depende del tiempo, podemos decir que la primera diferencia es estacionaria.

### Random Walk with Drift

Este tipo de procesos se define de la siguiente manera:

$$Y_t = \beta + Y_{t-1} + \epsilon_t \quad (14.11)$$

donde  $\beta$  se conoce como el *parámetro de variación o drift*. Si realizamos el proceso iterativo la variable  $Y_t$  análogamente que en el anterior caso, tendremos:

$$\begin{aligned}
Y_1 &= \beta + Y_0 + \epsilon_1 \\
Y_2 &= \beta + Y_1 + \epsilon_2 = 2\beta + Y_0 + \epsilon_1 + \epsilon_2 \\
Y_3 &= \beta + Y_2 + \epsilon_3 = 3\beta + Y_0 + \epsilon_1 + \epsilon_2 + \epsilon_3
\end{aligned}$$

Así sucesivamente hasta obtener la siguiente expresión:

$$Y_t = t\beta + Y_0 + \Sigma\epsilon_t \quad (14.12)$$

Si caracterizamos la variable  $Y_t$  se obtiene:

$$\begin{aligned}
E[Y_t] &= E[t\beta + Y_0 + \Sigma\epsilon_t] \\
&= t\beta + Y_0 + \Sigma E[\epsilon_t] \\
&= t\beta + Y_0
\end{aligned} \quad (14.13)$$

$$\begin{aligned}
Var[Y_t] &= Var[t\beta + Y_0 + \Sigma\epsilon_t] \\
&= Var[\Sigma\epsilon_t] \\
&= \Sigma Var[\epsilon_t] \\
&= \Sigma\sigma_\epsilon^2 \\
&= t\sigma_\epsilon^2
\end{aligned} \quad (14.14)$$

Como se observa en las expresiones (4.13) y (4.14), tanto la media como la varianza dependen directamente del tiempo, originando que la Serie  $Y_t$  no sea estacionaria.

Si nuevamente despejamos el término rezagado de  $Y_t$  para obtener al lado izquierdo la primera diferencia ( $\Delta Y_t$ ), y luego caracterizamos esta nueva serie se conseguiría lo siguiente:

$$Y_t - Y_{t-1} = \Delta Y_t = \beta + \epsilon_t \quad (14.15)$$

$$E[\Delta Y_t] = E[\beta + \epsilon_t] = \beta \quad (14.16)$$

$$Var[\Delta Y_t] = Var[\beta + \epsilon_t] = \sigma_\epsilon \quad (14.17)$$

Tal como se muestra en las expresiones (14.16) y (14.17), la primera diferencia de la serie  $Y_t$  es estacionaria, ya que sus dos primeros momentos no dependen del tiempo.

En conclusión, se puede decir que una serie un proceso Random Walk con o sin drift es *NO ESTACIONARIA*, sin embargo, su primera diferencia si lo es.

## 14.2. Proceso de Raíz Unitaria

Sea el siguiente proceso AR(1):

$$Y_t = \rho Y_{t-1} + \epsilon_t, -1 \leq \rho \leq 1 \quad (14.18)$$

donde  $\epsilon_t$  es nuevamente un ruido blanco.

Un proceso de Raíz Unitaria se da cuando  $\rho = 1$ , lo cual conlleva a que este modelo se convierta en un random walk puro, y por lo tanto, se tendría un proceso no estacionario de  $Y_t$ .

Si manipulamos la expresión (14.18), restando  $Y_{t-1}$  en ambos lados, se obtiene:

$$\begin{aligned} Y_t - Y_{t-1} &= \rho Y_{t-1} - Y_{t-1} + \epsilon_t \\ &= (\rho - 1)Y_{t-1} + \epsilon_t \end{aligned} \quad (14.19)$$

O que es lo mismo:



$$\Delta Y_t = \delta Y_{t-1} + \epsilon_t \quad (14.20)$$

donde  $\delta = \rho - 1$ . Así que en la práctica se estimará el modelo (14.20), para probar la hipótesis nula de que  $\delta = 0$ , que indirectamente es lo mismo decir que  $\rho = 1$ , y nos lleva a conclusión de que existe Raíz Unitaria y por lo tanto la serie  $Y$  no es estacionaria.

Sin embargo, el estadístico usual para contrastar esta hipótesis no es el  $t$  - *student*. Dickey & Fuller demostraron que bajo la hipótesis nula de que  $\delta = 0$ , el valor estimado de  $t$  del coeficiente de  $Y_{t-1}$  en (14.20) sigue una *distribución estadístico tau* ( $\tau$ ), por lo tanto, tuvieron que calcular *valores críticos del estadístico*  $\tau$  en abse a simulaciones de Montecarlo.

### 14.2.1. Pruebas de Raíz Unitaria

Dentro de los test existentes para probar la presencia de Unit Root tenemos: Dickey & Fuller (DF), Dickey & Fuller Aumentado (ADF), Phillips Perron (PP) y Kwiatkowski, Phillips, Smichdt y Shin (KPSS).

A manera de ejemplo, analizaremos la serie del Índice General de la Bolsa de Lima (IGBVL) desde el mes de Enero de 1992 hasta Junio 2012, cuya información provierne del BCRP y se encuentra en el archivo **igbvl\_mensual.csv**.

```
. *****
. * PROCESOS ESTOCÁTICOS NO ESTACIONARIOS *
. *****

. *Incluyendo la ruta donde se encuentra el archivo

. clear all
. set mem 200m
Current memory allocation

```

settable	current value	description	memory usage (1M = 1024k)
set maxvar	5000	max. variables allowed	1.909M
set memory	200M	max. data space	200.000M
set matsize	400	max. RHS vars in models	1.254M
			203.163M

```
. cd "D:\Econometria-Stata\no-estacionario"
D:\Econometria-Stata\no-estacionario
```

```

. insheet using igbvl_mensual.csv,delimiter(";")
(3 vars, 247 obs)

. *creamos nuestra variable temporal
. gen time=ym(year,month)
(1 missing value generated)

. format %tm time

. *establecemos una base de datos de serie de tiempo
. tsset time
      time variable:  time, 1992m1 to 2012m6
              delta:  1 month

. *Graficamos el IGB
. tsline igb91

```

### Dickey - Fuller (DF)

El modelo más simple para evaluar la presencia de raíz unitaria es el desarrollado por Dickey & Fuller:

$$\Delta Y_t = \alpha + \delta Y_{t-1} + \epsilon_t$$

Ahora el contraste es el siguiente:

Ho:  $\delta = 0 \Rightarrow$  Existe Unit Root, por lo tanto,  $Y_t$  No es Estacionaria

Ha:  $\delta \neq 0 \Rightarrow$  No Existe Unit Root, por lo tanto,  $Y_t$  es Estacionaria

Si:

$\tau$ -calculado  $>$   $\tau$ -crítico  $\Rightarrow$  Se rechaza la Ho.

$\tau$ -calculado  $<$   $\tau$ -crítico  $\Rightarrow$  Se acepta la Ho.

```

. * PRUEBAS DE RAÍZ UNITARIA
. *****

. *Prueba de Dickey-Fuller (DF)

```

```
. dfuller igb91, noconstant regress
```

Dickey-Fuller test for unit root				Number of obs = 245	
Test Statistic		1% Critical Value	Interpolated Dickey-Fuller	5% Critical Value	10% Critical Value
Z(t)		0.825	-2.581	-1.950	-1.620
D.igb91	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
igb91 L1.	.005005	.0060686	0.82	0.410	-.0069485 .0169585

Según el test de DF, el  $\tau$ -calculado (0.825) es menor (en valor absoluto) al  $\tau$ -crítico (1.950) a un nivel de significancia del 95 %. Por lo tanto, se acepta la  $H_0$ , es decir, el IGBVL es no estacionaria. las estimaciones que originan este test se muestran en el cuadro de abajo.

### Dickey - Fuller Aumentado (ADF)

En esta prueba se puede excluir la constante e incluir una tendencia lineal, también existe la opción de pedir utilizar los rezagos de la variable diferenciada ( $\Delta Y_{t-i}$ ) de acuerdo a algún criterio de información.

Por lo cual, en terminos generales, la prueba ADF consiste en estimar el siguiente modelo:

$$\Delta Y_t = \beta_1 + \beta_2 t + \delta Y_{t-1} + \alpha_i \sum_{i=1}^m \Delta Y_{t-i} + \epsilon_t \quad (14.21)$$

El contraste es similar al anterior caso:

$H_0: \delta = 0 \Rightarrow$  Existe Unit Root, por lo tanto,  $Y_t$  es Estacionaria

$H_a: \delta \neq 0 \Rightarrow$  No Existe Unit Root, por lo tanto,  $Y_t$  No es Estacionaria

Si:

$\tau$ -calculado  $>$   $\tau$ -crítico  $\Rightarrow$  Se rechaza la  $H_0$ .

$\tau$ -calculado  $<$   $\tau$ -crítico  $\Rightarrow$  Se acepta la  $H_0$ .

. \*Prueba de Dickey-Fuller Aumentado (ADF)

. dfuller igb91, trend regress

Dickey-Fuller test for unit root

Number of obs = 245

	Test Statistic	Interpolated Dickey-Fuller		
		1% Critical Value	5% Critical Value	10% Critical Value
Z(t)	-1.701	-3.992	-3.431	-3.131

MacKinnon approximate p-value for Z(t) = 0.7503

D.igb91	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
igb91						
L1.	-.0233522	.013726	-1.70	0.090	-.0503899	.0036856
_trend	2.749317	1.388493	1.98	0.049	.0142433	5.484391
_cons	-115.1746	126.6946	-0.91	0.364	-364.7397	134.3904

Según el test de ADF, el  $\tau$ -calculado (1.701) es menor (en valor absoluto) al  $\tau$ -crítico (3.431) a un nivel de significancia del 95 %. Por lo tanto, se acepta la  $H_0$ , es decir, el IGBVL es no estacionaria. las estimaciones que originan este test se muestran en el cuadro de abajo.

### Phillips Perron - PP

Este contraste estadístico estima una regresión haciendo una corrección sobre la matriz de varianzas y covarianzas de los residuos. La corrección es mediante un método no paramétrico.

Al igual que la prueba ADF, la prueba PP es una prueba de hipótesis sobre  $\rho = 1$  en la ecuación:  $\Delta Y_t = \Delta\beta + \rho Y_{t-1} + \Delta\epsilon$ ; pero a diferencia de la prueba ADF, no existen términos de diferencias retardados. Mas bien, la ecuación es estimada por MCO y luego el estadístico  $t$  del coeficiente  $\rho$  es corregido.

La hipótesis nula  $H_0$  del test de Phillips-Perron es la trayectoria de raíz unitaria con tendencia y la alternativa la estacionariedad con tendencia, si el valor  $t$ -Student asociado al coeficiente de  $Y_{t-1}$  es mayor en valor absoluto al valor crítico de MacKinnon, se rechaza la hipótesis de existencia de raíz unitaria.

. \*Prueba de Phillips-Perron (PP)

. pperron igb91, trend regress					
Phillips-Perron test for unit root					
				Number of obs =	245
				Newey-West lags =	4
	Test Statistic	1% Critical Value	Interpolated Dickey-Fuller		
			5% Critical Value	10% Critical Value	
Z(rho)	-8.926	-28.367	-21.280	-17.983	
Z(t)	-2.120	-3.992	-3.431	-3.131	
MacKinnon approximate p-value for Z(t) = 0.5346					
igb91	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
igb91					
L1.	.9766478	.013726	71.15	0.000	.9496101 1.003686
_trend	2.749317	1.388493	1.98	0.049	.0142433 5.484391
_cons	-115.1746	126.6946	-0.91	0.364	-364.7397 134.3904

Según el test de PP, el valor t-Student asociado al coeficiente de  $Y_{t-1}$  (8.926) es mayor en valor absoluto al valor crítico de MacKinnon (21.280) a un nivel de significancia del 95%. Por lo tanto, se acepta la  $H_0$ , es decir, el IGBVL es no estacionaria. las estimaciones que originan este test se muestran en el cuadro de abajo.

### Elliot, Rothenberg y Stock (ERS)o (DF-GLS)

Antes de aplicar la regresión propuesta por Dickey & Fuller, se debe primero extraer la tendencia de la serie original. Pero se trata de una cuasidiferencia  $Y_t - aY_{t-1}$ , donde  $a$  toma el valor uno en el caso anterior (ADF). Aquí el valor de  $a$  representa el punto específico contra el cual contrastamos la hipótesis nula (valor menor a uno).

El contraste es el siguiente:

$H_0$ : La serie tiene Unit Root.

$H_a$ : La serie no tiene Unit Root.

Si:

$t\text{-calculado} > t\text{-crítico} \Rightarrow$  Se rechaza la  $H_0$ .

$t$ -calculado  $< t$ -crítico  $\Rightarrow$  Se acepta la  $H_0$ .

```
. *Prueba de Elliot, Rothenberg y Stock (ERS) o DF-GLS
. dfgls igb91

DF-GLS for igb91                                Number of obs =   230
Maxlag = 15 chosen by Schwert criterion
```

[lags]	DF-GLS tau Test Statistic	1% Critical Value	5% Critical Value	10% Critical Value
15	-2.436	-3.480	-2.803	-2.525
14	-2.174	-3.480	-2.813	-2.534
13	-1.954	-3.480	-2.823	-2.543
12	-1.961	-3.480	-2.833	-2.552
11	-2.271	-3.480	-2.842	-2.560
10	-2.042	-3.480	-2.851	-2.569
9	-1.896	-3.480	-2.860	-2.577
8	-2.010	-3.480	-2.868	-2.584
7	-1.706	-3.480	-2.876	-2.592
6	-1.838	-3.480	-2.884	-2.599
5	-1.855	-3.480	-2.892	-2.606
4	-2.188	-3.480	-2.899	-2.612
3	-2.619	-3.480	-2.906	-2.618
2	-2.100	-3.480	-2.912	-2.624
1	-1.462	-3.480	-2.918	-2.630

```
Opt Lag (Ng-Perron seq t) = 12 with RMSE  810.7147
Min SC = 13.58869 at lag  3 with RMSE  851.5482
Min MAIC = 13.53372 at lag  5 with RMSE  835.3038
```

El test de DFGLS nos brinda dos informaciones importantes: El primero, es el *número de rezagos óptimos (15)* incluidos en el modelo para probar raíz unitaria; y segundo, los estadísticos calculados  $t$ , el cual son en todos los rezagos menores al valor  $t$  crítico. Por lo tanto, se puede decir que existe se acepta la  $H_0$  (existe raíz unitaria).

### Kwiatkowski, Phillips, Smichdt y Shin - KPSS

Proponen contrastar como hipótesis nula la hipótesis de estacionariedad en tendencias, ésta es la principal diferencia con los otros test de raíces unitarias. KPSS es frecuentemente utilizado con las otras pruebas de raíces unitarias para investigar si la serie es fraccionalmente integrada.

El contraste es el siguiente:

$H_0$ : La serie es estacionaria.

$H_a$ : La serie no es estacionaria.

Si:

valor calculado  $>$  valor crítico  $\Rightarrow$  Se rechaza la  $H_0$ .

valor calculado  $<$  valor crítico  $\Rightarrow$  Se acepta la  $H_0$ .

```
. *Prueba de Kwiatkowski, Phillips, Smichdt y Shin (KPSS)

. findit kpss

. kpss igb91
KPSS test for igb91

Maxlag = 15 chosen by Schwert criterion
Autocovariances weighted by Bartlett kernel

Critical values for H0: igb91 is trend stationary
10%: 0.119  5% : 0.146  2.5%: 0.176  1% : 0.216

Lag order      Test statistic
  0             3.72
  1             1.89
  2             1.27
  3             .967
  4             .785
  5             .664
  6             .579
  7             .515
  8             .466
  9             .427
 10             .394
 11             .368
 12             .345
 13             .326
 14             .31
 15             .296
```

Al igual que el test de DFGLS, el KPSS nos brinda dos informaciones importantes: El primero, es el *número de rezagos óptimos (15)*; y segundo, el valor de los estadísticos calculados, el cual son en todos los rezagos mayores al valor  $t$  crítico. Por lo tanto, se puede decir que existe se rechaza la  $H_0$  (la serie no es estacionaria).

### 14.2.2. Transformación de Series No estacionarias

Dado la serie analizada presenta raíz unitaria según todas las pruebas, entonces, es necesario convertirla en estacionaria. Una de las formas mas usuales es el método de la **diferenciación**, es decir, calcular  $\Delta Y_t$ . A partir de esta nueva serie aplicar

otra vez las pruebas de raíz unitaria y verificar que no exista este problema. El procedimiento de diferenciación se utilizó hasta que la serie se convierta en estacionaria.

Si generamos la diferencia de la serie IGBVL ( $d_{igb91}$ ) y contrastamos las pruebas de raíz unitaria veremos lo siguiente:

```
. *generamos la diferencia del igb91
. g d_igb91=D.igb91
(2 missing values generated)
```

```
. *pruebas de raíz unitaria
```

```
. dfuller d_igb91, noconstant regress //DF
```

Dickey-Fuller test for unit root		Number of obs = 244			
		Interpolated Dickey-Fuller			
	Test Statistic	1% Critical Value	5% Critical Value	10% Critical Value	
Z(t)	-14.424	-2.581	-1.950	-1.620	
D.d_igb91	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
d_igb91 L1.	-.9241252	.0640686	-14.42	0.000	-1.050326 - .7979245

```
. dfuller d_igb91, trend regress // ADF
```

Dickey-Fuller test for unit root		Number of obs = 244			
		Interpolated Dickey-Fuller			
	Test Statistic	1% Critical Value	5% Critical Value	10% Critical Value	
Z(t)	-14.536	-3.992	-3.431	-3.131	
MacKinnon approximate p-value for Z(t) = 0.0000					
D.d_igb91	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
d_igb91 L1.	-.9368331	.0644488	-14.54	0.000	-1.063788 - .8098782
_trend	.7661244	.8095957	0.95	0.345	-.8286628 2.360912
_cons	-17.14068	114.1104	-0.15	0.881	-241.9217 207.6404

```
. dfglS d_igb91 //ERS
```

DF-GLS for d_igb91			Number of obs = 229	
Maxlag = 15 chosen by Schwert criterion				
[lags]	DF-GLS tau Test Statistic	1% Critical Value	5% Critical Value	10% Critical Value



15	-3.624	-3.480	-2.803	-2.525
14	-3.135	-3.480	-2.813	-2.534
13	-3.549	-3.480	-2.823	-2.543
12	-4.056	-3.480	-2.833	-2.552
11	-4.207	-3.480	-2.842	-2.560
10	-3.775	-3.480	-2.851	-2.569
9	-4.301	-3.480	-2.860	-2.577
8	-4.827	-3.480	-2.868	-2.584
7	-4.815	-3.480	-2.877	-2.592
6	-6.001	-3.480	-2.884	-2.599
5	-6.081	-3.480	-2.892	-2.606
4	-6.609	-3.480	-2.899	-2.612
3	-6.164	-3.480	-2.906	-2.619
2	-5.575	-3.480	-2.913	-2.624
1	-7.586	-3.480	-2.919	-2.630

Opt Lag (Ng-Perron seq t) = 15 with RMSE 804.3016  
 Min SC = 13.59831 at lag 3 with RMSE 855.5125  
 Min MAIC = 14.20697 at lag 2 with RMSE 867.3959

. pperron d\_igb91, trend regress //PP

Phillips-Perron test for unit root			Number of obs =	244
			Newey-West lags =	4
		Interpolated Dickey-Fuller		
	Test Statistic	1% Critical Value	5% Critical Value	10% Critical Value
Z(rho)	-274.702	-28.360	-21.276	-17.980
Z(t)	-14.899	-3.992	-3.431	-3.131

MacKinnon approximate p-value for Z(t) = 0.0000

d_igb91	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
d_igb91						
L1.	.0631669	.0644488	0.98	0.328	-.063788	.1901218
_trend	.7661244	.8095957	0.95	0.345	-.8286628	2.360912
_cons	-17.14068	114.1104	-0.15	0.881	-241.9217	207.6404

. kpss d\_igb91 //KPSS

KPSS test for d\_igb91

Maxlag = 15 chosen by Schwert criterion  
 Autocovariances weighted by Bartlett kernel

Critical values for H0: d\_igb91 is trend stationary

10%: 0.119 5% : 0.146 2.5%: 0.176 1% : 0.216

Lag order	Test statistic
0	.0499
1	.0469
2	.0396
3	.0344
4	.0326
5	.0317
6	.0313

---

7	.0318
8	.032
9	.0326
10	.0329
11	.033
12	.0334
13	.0336
14	.0336
15	.0337

Como se puede observar en todos los casos, la primera diferencia del IGBVL deja de ser no estacionaria. Eliminandose el problema de raíz unitaria.

### 14.3. Ejercicio Propuesto

Se la base de datos de **sector\_real.xls** el cual contiene información anual entre el periodo 1992-2011 de algunas series macroeconómicas. Se pide analizar la estacionariedad de todas las series. Si en caso no fueran estacionarias, convertirlas a estacionarias.



# Capítulo 15

## Modelos de Vectores Autoregresivos

El Modelo de Vectores Autoregresivos (VAR) representa un sistema lineal de  $n$  variables con  $n$  ecuaciones en el que cada variable es explicado por sus propios valores rezagados y los valores pasados de las restantes  $n-1$  variables. La importancia de los modelos VARs es que es un enfoque coherente y creíble de descripción de datos, predicción, inferencia estructural y análisis de políticas. La metodología de Vectores Autoregresivos parte del supuesto del no conocimiento de las variables (el modelo teórico detrás de la forma reducida) por lo que busca ver las dinámicas entre las variables. Existen tres variedades de VAR:

- **Forma Reducida del VAR**

Expresa cada variable como una función lineal de sus propios valores pasados y los valores pasados de todas las demás variables, considerando el termino de error serialmente no correlacionado. Cada ecuación es estimada por MCO. El numero de valores rezagados a incluir se basan en diferentes métodos ( Akaike, BIC, etc). El término de error en estas regresiones son los movimientos sorpresas en las variables después de considerar los valores pasados. Si las variables se encuentran correlacionadas, el término de error del modelo en la forma reducida también puede estar correlacionado entre las ecuaciones.

- **VAR Recursivo**

Construye los términos de error en cada regresión como no correlacionado con el término de error de la ecuación anterior. Se incluye algunos valores contemporáneos como regresores. La estimación de cada ecuación se hace con MCO, produciéndose residuos que no se encuentran correlacionados. Los resultados dependen del orden de las variables, donde hay  $n$  representaciones de VARs.

### ■ VAR Estructural

Usa la teoría económica para establecer las relaciones contemporáneas entre las variables. VAR estructurales requiere *supuestos de identificación* que permita que las correlaciones sean interpretadas por causalidad. Estos supuestos de identificación puede involucrar todo el VAR o solo algunas ecuaciones. Esto produce variables instrumentales que permitan relaciones contemporáneas sean estimadas usando regresiones de variables instrumentales. El número de VARs estructurales es limitado solamente por la inventiva del investigador.

A continuación se realizara una especificación del VAR no restringido (a la Cholesky), ordenando las variables desde la más endógena hasta la más exógena. De acuerdo al esquema tradicional de política monetaria, el mecanismo de transmisión va desde la tasa de interés, pasando con la demanda agregada y terminando en la variación del índice de precios.

### Ejercicio

En el archivo **phillips.dta** se tiene información de algunas variables económicas peruanas: *inflacion* (variación porcentual del índice de precios), *desempleo* (tasa porcentual), *interes* (tasa de referencia de la política monetaria) durante el primer trimestre 2005 hasta el último trimestre del 2011. Utilizando un modelo parecido al de Stock y Watson se plantea un VAR reducido que involucren las siguientes ecuaciones:

$$\begin{aligned} inflacion_t = & \beta_{11}inflacion(-1) + \beta_{12}inflacion(-2) + \beta_{13}desempleo(-1) + \\ & + \beta_{14}desempleo(-2) + \beta_{15}interes(-1) + \beta_{16}interes(-2) + \beta_{17} \end{aligned}$$

$$\begin{aligned}
desempleo_t &= \beta_{21}inflacion(-1) + \beta_{22}inflacion(-2) + \beta_{23}desempleo(-1) + \\
&\quad + \beta_{24}desempleo(-2) + \beta_{25}interes(-1) + \beta_{26}interes(-2) + \\
\\ 
interes_t &= \beta_{31}inflacion(-1) + \beta_{32}inflacion(-2) + \beta_{33}desempleo(-1) + \\
&\quad + \beta_{34}desempleo(-2) + \beta_{35}interes(-1) + \beta_{36}interes(-2) +
\end{aligned}$$

Dado el sistema de ecuaciones, se procederá a realizar los pasos en STATA (en orden) y estimar un VAR con el objetivo de encontrar la función impulso respuesta y realizar proyecciones:

```

. *1ER PASO
. *****

. *Limpiando la memoria

. clear

. *Identificando la ruta donde se encuentra el archivo

. cd "D:\Econometria-Stata\var"
D:\Econometria-Stata\var

. *Abriendo un archivo de formato STATA (.dta)

. use phillips.dta

. *Definiendo a la base de datos como una serie de tiempo

. tsset year
    time variable:  year, 1 to 28
    delta: 1 unit

. *2do PASO
. *****
. *Asumiendo que las variables son estacionarias (sgte. cap. realizaremos dicho analisis)

. *3er PASO
. *****

. *Estimación VAR

. var inflacion desempleo interes

```



```
. *Rezago óptimo del VAR
. varsoc, maxlag(3)
```

Selection-order criteria  
Sample: 4 - 28                      Number of obs       =       25

lag	LL	LR	df	p	FPE	AIC	HQIC	SBIC
0	-97.2078				.608426	8.01662	8.05719	8.16289
1	-67.1157	60.184	9	0.000	.113489	6.32926	6.49153	6.91432*
2	-55.2493	23.733*	9	0.005	.093706*	6.09994*	6.38392*	7.1238
3	-50.2664	9.9659	9	0.353	.142213	6.42131	6.82699	7.88396

Endogenous: inflacion desempleo interes  
Exogenous: \_cons

Nótese que cuando escribimos solamente el comando **var** sin ninguna opción, Stata por default realiza usando 2 rezagos en la estimación. Posteriormente se determinó el rezago óptimo usando los criterios de información de AIC, BIC y HQIC a través del comando **varsoc**. Según dicho resultado mostrado arriba, este nos indica que la estimación var se debería usar 2 rezagos. Teniendo claro esto procedemos a realizar el cuarto paso, que es realizar el test de causalidad de Granger.

```
. *4to PASO
. *****

. *Causalidad de granger
. var inflacion desempleo interes, lags(1/2)
```

Vector autoregression

Sample: 3 - 28                      No. of obs       =       26  
Log likelihood = -58.94595                      AIC               =   6.149689  
FPE               = .0979673                      HQIC              =   6.442304  
Det(Sigma\_ml) = .0186982                      SBIC               =   7.165843

Equation	Parms	RMSE	R-sq	chi2	P>chi2
inflacion	7	.625039	0.3629	14.8097	0.0218
desempleo	7	.84337	0.1596	4.93706	0.5519
interes	7	.463316	0.9278	333.8844	0.0000



	Coef.	Std. Err.	z	P> z	[95 % Conf. Interval]	
inflacion						
inflacion						
L1.	.4587205	.2044218	2.24	0.025	.0580612	.8593798
L2.	-.1625398	.2408203	-0.67	0.500	-.6345389	.3094594
desempleo						
L1.	-.4179195	.145849	-2.87	0.004	-.7037784	-.1320606
L2.	.0647335	.128392	0.50	0.614	-.1869101	.3163771
interes						
L1.	.1119754	.1961098	0.57	0.568	-.2723928	.4963436
L2.	-.0691278	.1628153	-0.42	0.671	-.3882399	.2499842
_cons	3.28333	1.441859	2.28	0.023	.4573388	6.109321
desempleo						
inflacion						
L1.	-.334926	.2758278	-1.21	0.225	-.8755386	.2056866
L2.	-.2506085	.3249407	-0.77	0.441	-.8874806	.3862636
desempleo						
L1.	-.0962437	.1967952	-0.49	0.625	-.4819553	.2894678
L2.	.1721826	.1732402	0.99	0.320	-.1673621	.5117272
interes						
L1.	.352897	.2646125	1.33	0.182	-.165734	.871528
L2.	-.1731846	.2196879	-0.79	0.431	-.603765	.2573957
_cons	7.285505	1.945511	3.74	0.000	3.472374	11.09864
interes						
inflacion						
L1.	.5491019	.1515296	3.62	0.000	.2521094	.8460944
L2.	-.3430815	.1785103	-1.92	0.055	-.6929553	.0067924
desempleo						
L1.	-.2673601	.108112	-2.47	0.013	-.4792557	-.0554645
L2.	.2070807	.0951718	2.18	0.030	.0205475	.3936139
interes						
L1.	1.438602	.1453683	9.90	0.000	1.153685	1.723519
L2.	-.6414245	.1206884	-5.31	0.000	-.8779693	-.4048797
_cons	1.125724	1.068791	1.05	0.292	-.9690684	3.220517

```
. vargranger
```

```
Granger causality Wald tests
```

Equation	Excluded	chi2	df	Prob > chi2
inflacion	desempleo	8.2631	2	0.016
inflacion	interes	.35172	2	0.839
inflacion	ALL	8.4553	4	0.076
desempleo	inflacion	2.6594	2	0.265
desempleo	interes	2.4175	2	0.299
desempleo	ALL	3.3026	4	0.509
interes	inflacion	14.336	2	0.001
interes	desempleo	8.6811	2	0.013
interes	ALL	16.629	4	0.002

La hipótesis nula es que el rezago de las variables *inflacion* e *interes* si ayudan a explicar o predecir las variables *inflacion*, *interes* y *desempleo* en un 10 % y 5 % respectivamente. Caso contrario es que el se observa en la variable *desempleo* quien no ayuda a explicar a ninguna variable.

```
. *5to PASO
```

```
. *****
```

```
. *Prueba de Estabilidad del VAR
```

```
. varstable, graph
```

```
Eigenvalue stability condition
```

Eigenvalue	Modulus
.7116242 + .3886478i	.810837
.7116242 - .3886478i	.810837
.6697269	.669727
-.3698625	.369863
.03898298 + .1720191i	.176381
.03898298 - .1720191i	.176381

```
All the eigenvalues lie inside the unit circle.
VAR satisfies stability condition.
```

En este caso el comando **varstable** realiza la prueba de estabilidad del var(2) estimado. Dado que los resultados del *modulus* de cada eigenvalor es estrictamente menor a 1, se cumple la condición de estabilidad.

```

. *6to PASO
. *****

. *Test de autocorrelación

. varlmar

Lagrange-multiplier test

```

lag	chi2	df	Prob > chi2
1	7.5555	9	0.57949
2	9.3872	9	0.40232

```

H0: no autocorrelation at lag order

. *Prueba de la significancia conjunta de los coeficientes del var(2).

. varwle

Equation: inflacion

```

lag	chi2	df	Prob > chi2
1	11.58681	3	0.009
2	.589807	3	0.899

```

Equation: desempleo

```

lag	chi2	df	Prob > chi2
1	2.800837	3	0.423
2	1.524876	3	0.677

```

Equation: interes

```

lag	chi2	df	Prob > chi2
1	146.4554	3	0.000
2	30.813	3	0.000

```

Equation: All

```

lag	chi2	df	Prob > chi2
1	171.7505	9	0.000
2	35.01551	9	0.000

Con respecto al test autocorrelación, existe evidencia para concluir que no existe tal problema hasta de orden 2 ya que la probabilidad es mayor a 0.05 y por ende no se rechaza la hipótesis nula. En la prueba de significancia individual y conjunta de la estimación var(2) concluye que todos los coeficientes para cada ecuación son

significativos excepto para la variable *desempleo* tanto para el primer y segundo rezago (*probabilidad* < 0,05). Sin embargo, la última tabla nos permite aseverar que de manera global todos los rezagos asociados a cada ecuación son signi.cativas (probabilidad menor a 0.05).

```
. *7 mo PASO
. *****

. *Pronóstico 3 anhos (2012-2015) usando var(2)

. fcast compute f1_, step(16)

. fcast graph f1_inflacion f1_desempleo f1_interes

. br f1_inflacion f1_desempleo f1_interes
```

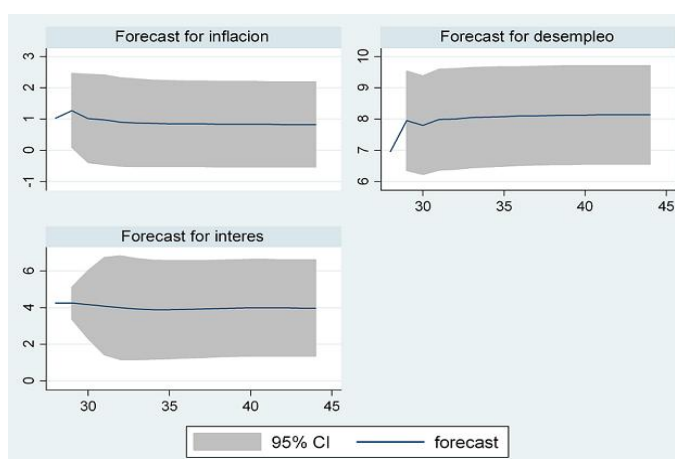


Figura 15.1: Proyección

```
. *8 vo PASO
. *****

. *Impulso Respuesta: caso var(2)

. varbasic inflacion desempleo interes, lags(1/2) irf
```

Vector autoregression

Sample: 3 - 28  
Log likelihood = -58.94595

No. of obs = 26  
AIC = 6.149689

FPE	=	.0979673	HQIC	=	6.442304
Det(Sigma_ml)	=	.0186982	SBIC	=	7.165843

Equation	Parms	RMSE	R-sq	chi2	P>chi2
inflacion	7	.625039	0.3629	14.8097	0.0218
desempleo	7	.84337	0.1596	4.93706	0.5519
interes	7	.463316	0.9278	333.8844	0.0000

	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
inflacion						
inflacion						
L1.	.4587205	.2044218	2.24	0.025	.0580612	.8593798
L2.	-.1625398	.2408203	-0.67	0.500	-.6345389	.3094594
desempleo						
L1.	-.4179195	.145849	-2.87	0.004	-.7037784	-.1320606
L2.	.0647335	.128392	0.50	0.614	-.1869101	.3163771
interes						
L1.	.1119754	.1961098	0.57	0.568	-.2723928	.4963436
L2.	-.0691278	.1628153	-0.42	0.671	-.3882399	.2499842
_cons	3.28333	1.441859	2.28	0.023	.4573388	6.109321
desempleo						
inflacion						
L1.	-.334926	.2758278	-1.21	0.225	-.8755386	.2056866
L2.	-.2506085	.3249407	-0.77	0.441	-.8874806	.3862636
desempleo						
L1.	-.0962437	.1967952	-0.49	0.625	-.4819553	.2894678
L2.	.1721826	.1732402	0.99	0.320	-.1673621	.5117272
interes						
L1.	.352897	.2646125	1.33	0.182	-.165734	.871528
L2.	-.1731846	.2196879	-0.79	0.431	-.603765	.2573957
_cons	7.285505	1.945511	3.74	0.000	3.472374	11.09864
interes						
inflacion						
L1.	.5491019	.1515296	3.62	0.000	.2521094	.8460944
L2.	-.3430815	.1785103	-1.92	0.055	-.6929553	.0067924
desempleo						
L1.	-.2673601	.108112	-2.47	0.013	-.4792557	-.0554645
L2.	.2070807	.0951718	2.18	0.030	.0205475	.3936139
interes						
L1.	1.438602	.1453683	9.90	0.000	1.153685	1.723519
L2.	-.6414245	.1206884	-5.31	0.000	-.8779693	-.4048797
_cons	1.125724	1.068791	1.05	0.292	-.9690684	3.220517

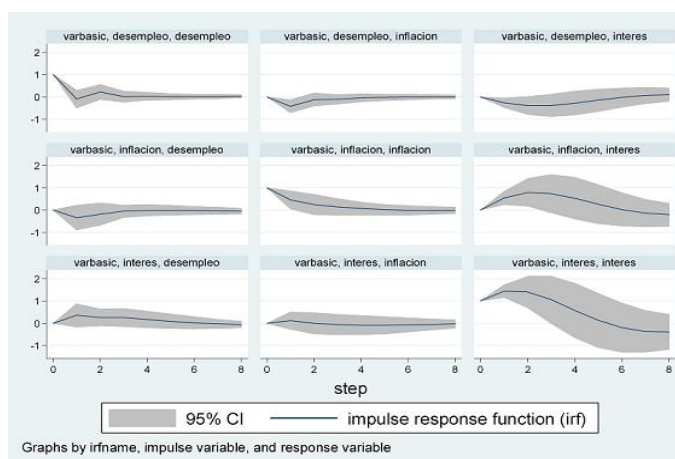


Figura 15.2: Función de Impulso Respuesta

La función impulso respuesta sirve para encontrar la respuesta de valores actuales y futuros de cada variable ante un incremento unitario en la innovación. De las funciones impulso respuesta, graficando la reacción hasta 8 periodos con bandas de confianza analíticas. Aquí se observa un posible trade-off entre la tasa de inflación y la tasa de desempleo en el corto plazo al igual que la tasa de interés y tasa de desempleo. También se observa una relación directa entre la tasa de interés y tasa de inflación.

### 15.1. Ejercicio Propuesto

En el archivo `tc_var.csv` se tiene información de algunas variables económicas peruanas: `tc_nominal` (definido como el precio relativo soles por dólar estadounidense) y `it_creal` (definido como el índice [Dic 01=100] del precio relativo entre bienes de ambos países) durante enero 1998 hasta el noviembre del 2011. Trabajando en logaritmos y asumiendo estacionalidad se pide estimar un VAR, encontrar el orden de dicha estimación, realizar la función impulso respuesta y proyectar 2 años ambas variables. Analizando esto se puede evidenciar la importancia que tienen las perturbaciones nominales en la dinámica del tipo de cambio real peruano.



## Capítulo 16

# Modelos de Corrección de Errores

El modelo de corrección de errores es utilizado cuando las variables están cointegradas. Si dos o más series temporales no estacionarias tienen una relación de largo plazo común (o de equilibrio), se debe evaluar la existencia de cointegración. La prueba de cointegración es un test para corroborar si una combinación lineal de series son estacionarias o no. Si se demuestra que existe cointegración entre estas dos series no es posible usar el enfoque de vectores autoregresivos (VAR) y por tanto es necesario usar el modelo de corrección de errores para conseguir resultados correctos.

Formalmente consideremos dos series temporales,  $y_t$  y  $x_t$ . De manera general la discusión sobre la relación dinámica de estas dos variables relacionadas se da a través del siguiente sistema de ecuaciones:

$$\begin{aligned}y_t &= \beta_{10} + \beta_{11}y(t-1) + \beta_{12}x(t-1) + v_t^y \\x_t &= \beta_{20} + \beta_{21}y(t-1) + \beta_{22}x(t-1) + v_t^x\end{aligned}$$

Tal como se observa, las ecuaciones describen un sistema en el cual cada variable está en función de su propio rezago y el rezago de la otra variable del sistema. En este caso, el sistema contiene dos variables llamadas  $y$  y  $x$ . Juntos las ecuaciones constituyen un sistema llamado vector autoregresivo (VAR). En este ejemplo, dado



que el número máximo de rezagos es 1, tenemos un VAR(1).

Si  $y$  y  $x$  son estacionarias, el sistema puede ser estimado usando MCO aplicado para cada ecuación. Si por el contrario,  $y$  y  $x$  no son estacionarias en sus niveles pero estacionarias en diferencias ( $I(1)$ ), luego se toma diferencias y se estima usando MCO:

$$\begin{aligned}\Delta y_t &= \beta_{10} + \beta_{11}\Delta y(t-1) + \beta_{12}\Delta x(t-1) + v_t^{\Delta y} \\ \Delta x_t &= \beta_{20} + \beta_{21}\Delta y(t-1) + \beta_{22}\Delta x(t-1) + v_t^{\Delta x}\end{aligned}$$

Si por otro lado,  $y$  y  $x$  son  $I(1)$  y cointegradas, el sistema de ecuaciones es modificado para permitir la relación de cointegración entre estas dos variables  $I(1)$ . Introduciendo la relación de cointegración el modelo correcto a usar es el de corrección de errores.

### Ejercicio

En el archivo **vecm.dta** se tiene información anual durante 1950-2011 de las siguiente variables económicas peruanas: *pbi* (Producto bruto interno en millones de soles 1994) y *cpr* (Consumo privado en millones de soles 1994). A partir del uso de estas variables (en términos logaritmos) se le pide estimar un modelo de corrección de errores realizando todo los pasos previos que permitan el uso de dicho modelo.

A continuación se procederá a realizar los pasos en STATA (en orden) y estimar un modelo de corrección de errores realizando todo los pasos previos, esto me permitirá encontrar la función impulso respuesta y realizar proyecciones.

```
. *1er PASO
. *****

. *Limpiando la memoria ram

. clear

. *Especificando la ruta donde se encuentra el archivo
```

```

. cd "D:\Econometria-Stata\vecm"
D:\Econometria-Stata\vecm

. *Abriendo un archivo en formato Stata (.dta)

. use vecm.dta

. *Identificando la data como time series

. tsset year
    time variable:  year, 1950 to 2011
                delta:  1 unit

. *Generando variables

. g lpbi=log(pbi)
. g lcpr=log(cpr)

. *2do PASO
. *****

. *Test para encontrar el rezago optimo incluyendole un maximo de rezagos: (3)

. dfgls lpbi, maxlag(3)

DF-GLS for lpbi
                                     Number of obs =   58

      [lags]      DF-GLS tau      1% Critical      5% Critical      10% Critical
                Test Statistic      Value      Value      Value
-----
      3          -1.338          -3.724          -3.061          -2.770
      2          -1.366          -3.724          -3.096          -2.802
      1          -1.731          -3.724          -3.127          -2.829

Opt Lag (Ng-Perron seq t) =  1 with RMSE  .0431345
Min SC   = -6.146849 at lag  1 with RMSE  .0431345
Min MAIC = -6.188669 at lag  2 with RMSE  .0422051

. *Rezago óptimo:1

. dfgls lcpr, maxlag(3)

DF-GLS for lcpr
                                     Number of obs =   58

      [lags]      DF-GLS tau      1% Critical      5% Critical      10% Critical
                Test Statistic      Value      Value      Value
-----
      3          -1.471          -3.724          -3.061          -2.770
      2          -1.316          -3.724          -3.096          -2.802
      1          -1.897          -3.724          -3.127          -2.829

Opt Lag (Ng-Perron seq t) =  2 with RMSE  .0388019
Min SC   = -6.288549 at lag  2 with RMSE  .0388019
Min MAIC = -6.361305 at lag  2 with RMSE  .0388019

. *Rezago óptimo:2

. *Raiz unitaria utilizando el rezago optimo

. dfuller lpbi, lags(1)

```

Augmented Dickey-Fuller test for unit root		Number of obs = 60		
		Interpolated Dickey-Fuller		
Test	1% Critical	5% Critical	10% Critical	
Statistic	Value	Value	Value	
Z(t)	-0.645	-3.566	-2.922	-2.596

MacKinnon approximate p-value for Z(t) = 0.8605

. dfuller lpbi, lags(1) noconstant

Augmented Dickey-Fuller test for unit root		Number of obs = 60		
		Interpolated Dickey-Fuller		
Test	1% Critical	5% Critical	10% Critical	
Statistic	Value	Value	Value	
Z(t)	2.765	-2.616	-1.950	-1.610

. dfuller lpbi, lags(1) trend

Augmented Dickey-Fuller test for unit root			Number of obs = 60	
		Interpolated Dickey-Fuller		
	Test Statistic	1% Critical Value	5% Critical Value	10% Critical Value
Z(t)	-2.069	-4.128	-3.490	-3.174

MacKinnon approximate p-value for Z(t) = 0.5637

. dfuller lcpr, lags(2)

Augmented Dickey-Fuller test for unit root			Number of obs = 59	
		Interpolated Dickey-Fuller		
	Test	1% Critical	5% Critical	10% Critical
	Statistic	Value	Value	Value
Z(t)	-1.219	-3.567	-2.923	-2.596

MacKinnon approximate p-value for Z(t) = 0.6653

. dfuller lcpr, lags(2) noconstant

Augmented Dickey-Fuller test for unit root		Number of obs = 59		
Test Statistic	Interpolated Dickey-Fuller			
	1% Critical Value	5% Critical Value	10% Critical Value	
Z(t)	3.500	-2.616	-1.950	-1.610

. dfuller lcpr, lags(2) trend

Augmented Dickey-Fuller test for unit root		Number of obs = 59		
		Interpolated Dickey-Fuller		
Test Statistic	1% Critical Value	5% Critical Value	10% Critical Value	
Z(t)	-2.046	-4.130	-3.491	-3.175

MacKinnon approximate p-value for Z(t) = 0.5763

```
. *Son no estacionarias ambas series (en logaritmos)
```

```
. *3er PASO
. *****
```

```
. *Raiz unitaria en diferencias, evaluando si son integrables de orden (1)
```

```
. dfuller d.lpb1, lags(1)
```

Augmented Dickey-Fuller test for unit root		Number of obs = 59		
Test Statistic	1% Critical Value	5% Critical Value	10% Critical Value	
Z(t)	-4.983	-3.567	-2.923	-2.596

MacKinnon approximate p-value for Z(t) = 0.0000

```
. dfuller d.lpb1, lags(1) noconstant
```

Augmented Dickey-Fuller test for unit root		Number of obs = 59		
Test Statistic	1% Critical Value	5% Critical Value	10% Critical Value	
Z(t)	-3.421	-2.616	-1.950	-1.610

```
. dfuller d.lpb1, lags(1) trend
```

Augmented Dickey-Fuller test for unit root		Number of obs = 59		
Test Statistic	1% Critical Value	5% Critical Value	10% Critical Value	
Z(t)	-4.948	-4.130	-3.491	-3.175

MacKinnon approximate p-value for Z(t) = 0.0003

```
. dfuller d.lcpr, lags(2)
```

Augmented Dickey-Fuller test for unit root		Number of obs = 58		
Test Statistic	1% Critical Value	5% Critical Value	10% Critical Value	
Z(t)	-3.917	-3.569	-2.924	-2.597

MacKinnon approximate p-value for Z(t) = 0.0019

```
. dfuller d.lcpr, lags(2) noconstant
```

Augmented Dickey-Fuller test for unit root		Number of obs = 58		
Test Statistic	1% Critical Value	5% Critical Value	10% Critical Value	
Z(t)	-2.589	-2.617	-1.950	-1.610

```
. dfuller d.lcpr, lags(2) trend
```

Augmented Dickey-Fuller test for unit root				Number of obs	=	58
	Test Statistic	1% Critical Value	Interpolated Dickey-Fuller 5% Critical Value	10% Critical Value		
Z(t)	-3.896	-4.132	-3.492	-3.175		

MacKinnon approximate p-value for Z(t) = 0.0123

```
. *Son estacionarias en primeras diferencias, es decir son integrables de orden (1)
. *El pvalue de MacKinnon es menor a 0.05
```

```
. *4to PASO
. *****
```

```
. *Prueba de cointegracion: METODO DE ENGLE Y GRANGER
```

```
. *Estimamos MCO
```

```
. reg lcpr lpbi
```

Source	SS	df	MS	Number of obs = 62		
Model	18.8529442	1	18.8529442	F( 1, 60)	=15159.80	
Residual	.074616847	60	.001243614	Prob > F	= 0.0000	
				R-squared	= 0.9961	
				Adj R-squared	= 0.9960	
Total	18.9275611	61	.310287887	Root MSE	= .03526	

lcpr	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
lpbi	.9537096	.0077459	123.13	0.000	.9382156	.9692036
_cons	.1972969	.0871087	2.26	0.027	.0230537	.3715402

```
. *Obtener los residuos, por que si hay cointegracion deberia ser estacionario
```

```
. predict u1, residuals
```

```
. *graficando los residuos
```

```
. tsline u1
```

```
. *otro grafico
```

```
. twoway (scatter u1 l.u1)
```

```
. *Test de raiz unitaria a los errores
```

```
. dfgls u1
```

DF-GLS for u1  
Maxlag = 10 chosen by Schwert criterion

Number of obs = 51

[lags]	DF-GLS tau Test Statistic	1% Critical Value	5% Critical Value	10% Critical Value
10	-3.553	-3.724	-2.721	-2.440
9	-3.787	-3.724	-2.773	-2.492
8	-3.506	-3.724	-2.826	-2.545
7	-3.204	-3.724	-2.880	-2.598
6	-3.094	-3.724	-2.934	-2.650
5	-3.691	-3.724	-2.987	-2.700
4	-2.935	-3.724	-3.037	-2.746
3	-2.880	-3.724	-3.083	-2.789
2	-2.885	-3.724	-3.124	-2.827
1	-2.715	-3.724	-3.160	-2.860

Opt Lag (Ng-Perron seq t) = 5 with RMSE .0164335

Min SC = -7.919204 at lag 1 with RMSE .0176557

Min MAIC = -7.70731 at lag 1 with RMSE .0176557

. dfuller u1,lags(5)

Augmented Dickey-Fuller test for unit root

Number of obs = 56

Test Statistic	Interpolated Dickey-Fuller		
	1% Critical Value	5% Critical Value	10% Critical Value
Z(t)	-3.367	-3.572	-2.925
			-2.598

MacKinnon approximate p-value for Z(t) = 0.0121

. \*Las estimaciones estan cointegradas, los residuos son ruido blanco  
. \*(integradas de orden 0)  
. \*por tanto, las estimaciones por el metodo de correccion de errores son super  
. \*consistentes

. \*5to PASO  
. \*\*\*\*\*

. \*OTRA PRUEBA DE COINTEGRATION: METODO DEL TEST DE JOHANSEN

. \*Test mas fuerte por el uso de maxima verosimilitud

. \*Asimismo calcula los eigenvalores, traza y si existe o no relacion de cointegracion

. \*Estimation de un var para calcular los rezagos a usar en el test de Johansen

. var lcpr lpbi

## Vector autoregression

Sample: 1952 - 2011  
 Log likelihood = 261.1643  
 FPE = 7.93e-07  
 Det(Sigma\_ml) = 5.68e-07

No. of obs = 60  
 AIC = -8.372144  
 HQIC = -8.235609  
 SBIC = -8.023087

Equation	Parms	RMSE	R-sq	chi2	P>chi2
lcpr	5	.042872	0.9938	9613.436	0.0000
lpbi	5	.045309	0.9937	9395.88	0.0000

		Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
lcpr	lcpr					
	L1.	1.120006	.2879655	3.89	0.000	.5556035 1.684408
	L2.	-.3015458	.2894079	-1.04	0.297	-.8687748 .2656832
	lpbi					
	L1.	.3423983	.2787885	1.23	0.219	-.204017 .8888137
	L2.	-.1754269	.284481	-0.62	0.537	-.7329994 .3821456
_cons		.1226998	.1127753	1.09	0.277	-.0983356 .3437353
lpbi	lcpr					
	L1.	.0176456	.3043344	0.06	0.954	-.5788388 .6141299
	L2.	-.0144649	.3058587	-0.05	0.962	-.6139368 .5850071
	lpbi					
	L1.	1.423183	.2946356	4.83	0.000	.8457082 2.000659
	L2.	-.4329815	.3006517	-1.44	0.150	-1.022248 .1562851
_cons		.0958273	.1191857	0.80	0.421	-.1377724 .3294271

. varsoc

## Selection-order criteria

Sample: 1952 - 2011  
 Number of obs = 60

lag	LL	LR	df	p	FPE	AIC	HQIC	SBIC
0	66.9515				.000393	-2.16505	-2.13774	-2.09524
1	251.026	368.15	4	0.000	9.7e-07	-8.16753	-8.08561	-7.95809
2	261.164	20.277*	4	0.000	7.9e-07*	-8.37214*	-8.23561*	-8.02309*

Endogenous: lcpr lpbi

Exogenous: \_cons

. \*Conclusión: Usar 2 rezagos

```
. vecrank lcpr lpbi, lags(2)
```

Trend: constant      Number of obs = 60  
Sample: 1952 - 2011      Lags = 2

maximum				trace	5%
rank	parms	LL	eigenvalue	statistic	critical value
0	6	257.4197	.	7.4892*	15.41
1	9	260.94945	0.11100	0.4298	3.76
2	10	261.16432	0.00714		

```
. vecrank lcpr lpbi, trend(constant) lags(2)
```

```
Trend: constant                                Number of obs =      60
Sample: 1952 - 2011                            Lags =              2
```

maximum				trace	5%
rank	parms	LL	eigenvalue	statistic	critical value
0	6	257.4197	.	7.4892*	15.41
1	9	260.94945	0.11100	0.4298	3.76
2	10	261.16432	0.00714		

```
. vecrank lcpr lpbi, trend(trend) lags(2)
```

```
Trend: trend                                Number of obs =      60
Sample: 1952 - 2011                          Lags =              2
```

maximum				trace	5%
rank	parms	LL	eigenvalue	statistic	critical value
0	8	257.44348	.	11.4774*	18.17
1	11	260.99836	0.11174	4.3676	3.74
2	12	263.18216	0.07021		

. \*Otra alternativa, no incluye ni tendencia ni constante

```
. vecrank lcpr lpbi, trend(none)lags(2)
```

```

Trend: none                      Number of obs =      60
Sample: 1952 - 2011              Lags =                2

```

maximum				trace	5%
rank	parms	LL	eigenvalue	statistic	critical value
0	4	253.58941	.	13.7971	12.53
1	7	257.60947	0.12541	5.7570	3.84
2	8	260.48795	0.09149		



```
. *6to PASO
. *****
```

. \*A continuación se estima el Modelo de Corrección de Errores

```
. vec lcpr lpbi, lags(2)
```

## Vector error-correction model

Sample: 1952 - 2011	No. of obs	=	60
	AIC	=	-8.398315
Log likelihood = 260.9494	HQIC	=	-8.275433
Det(Sigma_ml) = 5.72e-07	SBIC	=	-8.084163

Equation	Parms	RMSE	R-sq	chi2	P>chi2
D_lcprr	4	.042611	0.5140	59.22012	0.0000
D_lpbir	4	.045063	0.4985	55.67505	0.0000

	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
D_lcpr						
_ce1						
l1.	-.1900002	.1650901	-1.15	0.250	-.5135708	.1335704
lcpr						
LD.	.3151055	.2994898	1.05	0.293	-.2718837	.9020947
lpbi						
LD.	.1676374	.295004	0.57	0.570	-.4105597	.7458346
_cons	-.0006973	.0167657	-0.04	0.967	-.0335574	.0321629
D_lpbi						
_ce1						
l1.	-.0067558	.1745929	-0.04	0.969	-.3489516	.33544
lcpr						
LD.	.0303912	.3167289	0.10	0.924	-.5903861	.6511684
lpbi						
LD.	.4238325	.3119848	1.36	0.174	-.1876465	1.035312
_cons	.0196099	.0177308	1.11	0.269	-.0151418	.0543615

Cointegrating equations

Equation	Parms	chi2	P>chi2
_ce1	1	1695.186	0.0000

Identification: beta is exactly identified

Johansen normalization restriction imposed

beta	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
_ce1					
lcpr	1	.	.	.	.
lpbi	-.951507	.0231102	-41.17	0.000	-.9968021 -.9062119
_cons	-.3175352	.	.	.	.

. vec lcpr lpbi, lags(2)alpha

Vector error-correction model

Sample: 1952 - 2011

No. of obs = 60

AIC = -8.398315

Log likelihood = 260.9494

HQIC = -8.275433

Det(Sigma\_ml) = 5.72e-07

SBIC = -8.084163

Equation	Parms	RMSE	R-sq	chi2	P>chi2
D_lcpr	4	.042611	0.5140	59.22012	0.0000
D_lpbi	4	.045063	0.4985	55.67505	0.0000

	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
D_lcpr					
_ce1					
L1.	-.1900002	.1650901	-1.15	0.250	-.5135708 .1335704
lcpr					
LD.	.3151055	.2994898	1.05	0.293	-.2718837 .9020947
lpbi					
LD.	.1676374	.295004	0.57	0.570	-.4105597 .7458346
_cons	-.0006973	.0167657	-0.04	0.967	-.0335574 .0321629
D_lpbi					
_ce1					
L1.	-.0067558	.1745929	-0.04	0.969	-.3489516 .33544
lcpr					
LD.	.0303912	.3167289	0.10	0.924	-.5903861 .6511684
lpbi					
LD.	.4238325	.3119848	1.36	0.174	-.1876465 1.035312
_cons	.0196099	.0177308	1.11	0.269	-.0151418 .0543615

## Cointegrating equations

Equation	Parms	chi2	P>chi2
_ce1	1	1695.186	0.0000

Identification: beta is exactly identified

Johansen normalization restriction imposed

beta	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
_ce1						
lcpr	1	.	.	.	.	.
lpbi	-.951507	.0231102	-41.17	0.000	-.9968021	-.9062119
_cons	-.3175352	.	.	.	.	.

## Adjustment parameters

Equation	Parms	chi2	P>chi2
D_lcpr	1	1.324544	0.2498
D_lpbi	1	.0014973	0.9691

alpha	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
D_lcpr						
_ce1						
L1.	-.1900002	.1650901	-1.15	0.250	-.5135708	.1335704
D_lpbi						
_ce1						
L1.	-.0067558	.1745929	-0.04	0.969	-.3489516	.33544

En la primera salida se muestra los coeficientes del modelo de corrección de errores de cada ecuación denotado por `_ce1`. Usando la opción **alpha** se obtendrá los parámetros ajustados de corto plazo. Esto quiere decir cuando las variables responden si hay un cambio o shock en el sistema.

```
. *7mo PASO
. *****

. *CREACION DE IMPULSO RESPUESTA

. irf set vec_eg,replace
(file vec_eg.irf created)
(file vec_eg.irf now active)

. irf create vec_eg, step(50) replace
irfname vec_eg not found in vec_eg.irf
(file vec_eg.irf updated)

. irf graph irf
```

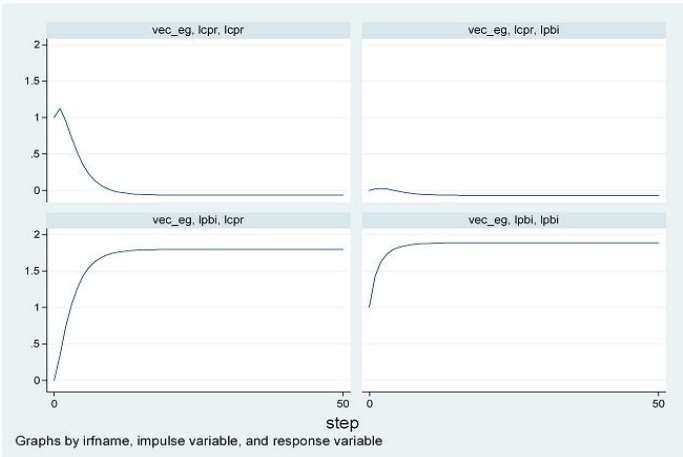


Figura 16.1: Función de Impulso Respuesta en un MCE

```
. *8vo PASO
. *****

. *Evaluando la estabilidad del modelo estimado

. vecstable

Eigenvalue stability condition
+-----+-----+
| Eigenvalue | Modulus |
+-----+-----+
| 1          | 1       |
| .6753039   | .675304 |
| .4983913   | .498391 |
| .3816708   | .381671 |
+-----+-----+

The VECM specification imposes a unit modulus.
```

```
. vecstable, graph

Eigenvalue stability condition
+-----+-----+
| Eigenvalue | Modulus |
+-----+-----+
| 1          | 1       |
| .6753039   | .675304 |
| .4983913   | .498391 |
| .3816708   | .381671 |
+-----+-----+

The VECM specification imposes a unit modulus.
```

```

. *Test de autocorrelacion

. veclmar

Lagrange-multiplier test

```

lag	chi2	df	Prob > chi2
1	7.9734	4	0.09256
2	6.2656	4	0.18017

```

H0: no autocorrelation at lag order

**No existe autocorrelación bajo ningún rezago (no se rechaza la hipótesis nula)

. *9no PASO
. *****

. *Pronóstico

. fcast compute f_1, step(24)

. fcast graph f_1lpbi f_1lcpr

```

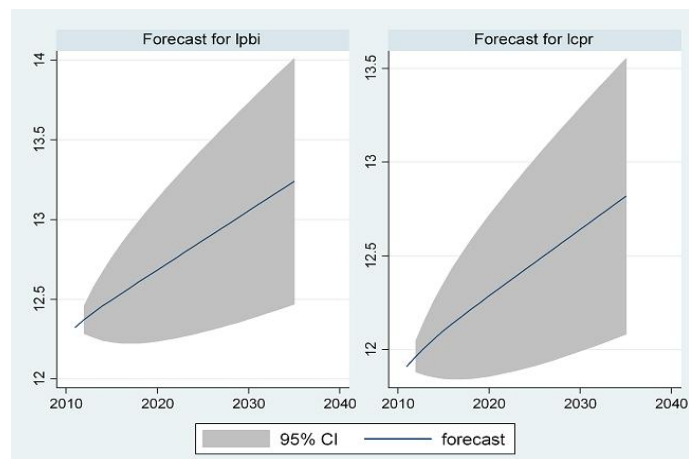


Figura 16.2: Proyección en un MCE

## 16.1. Ejercicio Propuesto

En el archivo **pbi\_cpr\_inv.csv** se tiene información anual durante 1950-2011 de las siguiente variables económicas peruanas: *pbi* (Producto bruto interno en

---

millones de soles 1994) y *cpr* (Consumo privado en millones de soles 1994) e *inv* (Inversion Bruta Fija en millones de soles 1994). A partir del uso de estas variables (en términos logaritmos) se le pide estimar un modelo de corrección de errores realizando todo los pasos previos que permitan el uso de dicho modelo.



## Parte V

### Modelos de Panel de Datos





# Capítulo 17

## Modelos de Datos de Panel Estáticos

Los datos de panel (o datos longitudinales) consiste en observaciones de un corte transversal (unidades transversales: hogares, empresas, regiones, países, etc.) repetidas sobre el tiempo. Es decir:  $Y_{it}, X_{it}'$   $i = 1, \dots, N; t = 1, \dots, T$ . Los datos de panel pueden ser balanceados ( $T_i = T$  para todo  $i$ ) o no balanceados ( $T_i \neq T$  para algun  $i$ ). Se pueden tener paneles:

**Lema 1** *De muchos individuos y pocos periodos temporales (“short panels”-micro panel).*

**Lema 2** *De muchos individuos y pocos periodos temporales (“long panels”-macro panel).*

Para cada observación debe conocerse el individuo  $i$  y el periodo temporal  $t$  al que se refiere.

- Para paneles balanceados, el número total de observaciones es simplemente  $NT$ .
- Para paneles no balanceados, el número total de observaciones es  $\sum_{i=1}^N T_i$ .

p.e., un panel balanceado				p.e., un panel NO balanceado				
individuo	año	renta	edad	individuo	año	renta	edad	sexo
1	2000	1800	20	1	2000	800	19	2
1	2001	1950	30	1	2001	950	20	2
2	2000	800	20	2	2000	1900	29	1
2	2001	850	21	2	2001	1950	30	1
				2	2002	2100	31	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
500	2000	2200	54	1000	2000	2100	49	1
500	2001	2400	55	1000	2001	2200	50	1

Figura 17.1: Datos de Panel balanceado y No balanceado

## 17.1. Modelo Agrupado (Pooled)

Sea el modelo MCO en panel (pooled) o de promedio poblacional:

$$Y_{it} = \beta_1 X_{1it} + \dots + \beta_k X_{kit} + u_{it}$$

Donde principalmente se cumplen los siguientes supuestos:

- Supone que los regresores estan incorrelacionados con  $u_{it}$ .
- Pero no una estructura en  $u_{it}$  (a diferencia de efectos aleatorios que lo veremos luego).
- Se puede estimar consistentemente por MCO, usando errores estándar robustos por la probable correlación entre individuos y en el tiempo para un individuo.

## 17.2. Modelos con efectos individuales (One-Way)

Estos tipos de modelos tienen la siguientes características:

$$Y_{it} = \beta_1 X_{1it} + \dots + \beta_k X_{kit} + u_{it}$$

$$Y_{it} = \beta_1 X_{1it} + \dots + \beta_k X_{kit} + \alpha_i + \epsilon_{it}$$

Donde:

- $X_{1it} + \dots + X_{kit}$ : variables explicativas (observables).
- $u_{it} = \alpha_i + \epsilon_{it}$
- $\alpha_i$ : efectos individuales (heterogeneidad inobservada permanente en el tiempo).
- $\epsilon_{it}$ : error idiosincrático

Existen dos modelos sustancialmente diferentes según el tratamiento de  $\alpha_i$ :

1. Modelos de Efectos Fijos.
2. Modelos de Efectos Aleatorios.

### 17.3. Modelo de Efectos Fijos (FE)

- Siguiendo la última ecuación (3), los efectos fijos (FE) permiten que los regresores  $X_{1it}, \dots, X_{kit}$  estén correlacionados con  $\alpha_i$ , es decir,  $E[\alpha_i | X_{1it}, \dots, X_{kit}] \neq 0$ .
- El otro supuesto fundamental es :  $E[\epsilon_{it} | \alpha_i, X_{1it}, \dots, X_{kit}] = 0$ , es decir los regresores deben seguir siendo incorrelacionados con  $\epsilon_{it}$ .
- Se necesita estimar  $\alpha_i$  junto con los parámetros de los regresores [Efectos fijos(FE) con variables dummy por individuo], en paneles cortos se necesita que  $N \rightarrow \infty$ . Por otro lado, los parámetros de los regresores podrían estar sesgados por estimar infinitos parámetros auxiliares  $\alpha_i$ .
- La solución a lo anterior, es estimar por *modelos transformados* de tal manera que se elimine  $\alpha_i$ : [*modelos en primeras diferencias*] y [*modelos intra-grupos o estimadores within- desviaciones respecto a la media*].

## 17.4. Modelo de Efectos Aleatorios (RE)

- En efectos aleatorios (RE), el efecto individual  $\alpha_i$  se trata como puramente aleatorio.
- Siguen los supuestos fundamentales:  $\alpha_i$  y  $\epsilon_{it}$  no están correlacionados con los regresores, es decir:  $E[\alpha_i|X_{it}] = 0 \rightarrow Var[\alpha_i|X_{it}] = \delta_\alpha^2$  y  $E[\epsilon_{it}|X_{it}] = 0 \rightarrow Var[\epsilon_{it}|x_{it}] = \delta_e^2$ .
- Dado lo anterior, esto implica que los regresores son exógenos con respecto al término de error compuesto, es decir dado que:  $u_{it} = \alpha_i + \epsilon_{it}$  se tiene que  $E[u_{it}|X_{it}] = 0$ .
- Además, se tiene una estructura de correlación:  $corr(u_{it}, u_{is}) = \delta_\alpha^2 / (\delta_\alpha^2 + \delta_e^2)$ ,  $t \neq s$ .
- Se puede estimar eficientemente utilizando MCGF (solo si cumple el supuesto de  $\alpha_i$ ):

$$(y_{it} - \theta_i y_i) = \beta(1 - \theta_i) + (X_{it} - \theta_i X_i) + \alpha_i(1 - \theta_i) + (\epsilon_{it} - \theta_i \epsilon_i)$$

- $\theta_i = 1 - (\frac{\delta_\alpha^2}{\delta_\alpha^2 + \delta_e^2})^{1/2}$ . Si:  $\theta_i = 0 \rightarrow$  “pooled”,  $\theta_i = 1 \rightarrow$  “within”.

Nota:

**Lema 3** *En términos de consistencia, si tanto los regresores están o no correlacionados con  $\alpha_i$  se puede estimar por efectos fijos (FE).*

Sin embargo, si no existe correlación, otro estimador es más eficiente (proporciona menos varianza)  $\rightarrow$  efectos aleatorios (RE).

Luego de conocer tres tipos de especificación en un modelo de panel de datos, es necesario responder a la pregunta de cual modelo es mejor. A continuación se explica los test que permitan concluir el mejor modelo.

## 17.5. Comparación de Modelos

### 17.5.1. Modelo Pooled vs. Modelo de Efectos Fijos: Prueba F

Para comparar entre el FE y Pooled se utiliza la prueba F, éste sirve para contrastar la hipótesis de que todos los efectos individuales son constantes. El estadístico F relaciona el modelo no restringido (con efectos individuales – FE) con el modelo restringido (efectos individuales constantes – Pooled). La hipótesis nula y la formula del estadístico utilizado los siguientes:

$$H_0 : \alpha_1 = \alpha_2 = \dots = \alpha_i$$

La hipótesis nula se contrasta con el siguiente estadístico:

$$F[n-1, (\sum_{i=1}^n T_i) - n - k] = \frac{((R_{nr}^2 - R_r^2)/(n-1))}{(1-R_{nr}^2)/((\sum_{i=1}^n T_i) - n - k)}$$

Donde:

- $R_{nr}^2$  = bondad de ajuste del modelo restringido (Pooled).
- $R_r^2$  = bondad de ajuste del modelo no restringido (FE).
- $n$  = número de unidades transversales.
- $T_i$  = número de años en que está presente la unidad transversal  $i$ .
- $k$  = número de unidades transversales.

### 17.5.2. Modelo Pooled vs. Modelo de Efectos Aleatorios: Prueba LM

- Para elegir entre los modelos Pooled o RE, se utiliza el LM Test de Breusch y Pagan. La cual tiene como hipótesis nula que:  $Var(\alpha_i) = 0$ , es decir :  $cov(u_{it}, u_{is}) = cov(\epsilon_{it}, \epsilon_{is})$ .

$$LM = \frac{NT}{2(T-1)} \frac{\sum_{i=1}^N (\sum_{t=1}^T \epsilon_{it})^2}{\sum_{i=1}^N \sum_{t=1}^T \epsilon_{it}} - 1 \sim \chi^2$$

- Si el valor de la probabilidad asociado al test es mayor a 0.05, la hipótesis nula no se rechaza es mejor el modelo pooled.
- Si por otro lado, la probabilidad asociado al test es menor a 0.05, la hipótesis nula se rechaza y por lo tanto es mejor elegir un modelo de efectos aleatorios (RE).

### 17.5.3. Modelo de Efecto Fijo vs. Modelo de Efecto Aleatorio: Prueba Hausman

$H_0 : cov(\alpha_i, X_{it}) = 0 \rightarrow$  efectos aleatorios (RE).

$H_a : cov(\alpha_i, X_{it}) \neq 0 \rightarrow$  efectos fijos (FE).

Donde el test es:

$$(\beta_{FE} - \beta_{RE})[Var(\beta_{FE}) - Var(\beta_{RE})]^{-1}(\beta_{FE} - \beta_{RE}) \sim \chi^2$$

- Si se tiene una probabilidad asociada a dicho test menor a 0.05, se rechaza la hipótesis nula y se prefiere elegir el modelo de efectos fijos.
- Por otro lado, si dicha probabilidad es mayor a 0.05, no se rechaza la hipótesis nula y se prefiere usar el modelo de efectos aleatorios.

### Ejercicio

En el archivo **lm\_panel.dta** se tiene información trimestral desde 1996 hasta el tercer trimestre del 2009 para los siguientes países: Chile, Colombia, México y Perú. Las siguientes variables son las que se usaran: *logm1* (logaritmo de los saldos monetarios nominales), *lrate* (tasa de interés por prestamos) y *loggdp* (logaritmo del producto bruto interno). Las variables *m1* y *gdp* están en millones de dólares. El modelo que vamos a estimar es el siguiente (no se incluirá el rezago del índice de precios por falta de datos), así:

$$\log m1_{it} = \beta_1 + \beta_2 \log gdp_{it} + \beta_3 lrate_{it} + u_{it}$$

El objetivo de este ejercicio es estimar los tres tipos de modelos panel, escoger el mejor modelo y corregir si existe problemas de heteroscedasticidad y/o autocorrelación.

A continuación se muestra los pasos en STATA:

```
. *1er PASO
. *****

. *Limpiando la memoria

. clear

. *Incluyendo la ruta donde se encuentra el archivo

. cd "D:\Econometria-Stata\panel data"
D:\Econometria-Stata\panel data

. *Abriendo la base de datos en formato Stata (.dta)

. use panel.dta

. *Declarando panel data: primero identificador transversal: code y luego el temporal:year

. xtset code year
      panel variable:  code (strongly balanced)
      time variable:  year, 1 to 55
      delta: 1 unit

. *2do PASO
. *****

. *Heterogeneidad entre paises
.
. bysort code: egen logm1_mean=mean(logm1)

. twoway scatter logm1 code, msymbol(circle_hollow)|| ///
connected logm1_mean code, symbol(diamond)||, ///
xlabel(1 "Chi" 2 "Col" 3 "Mex" 4 "Per")
```



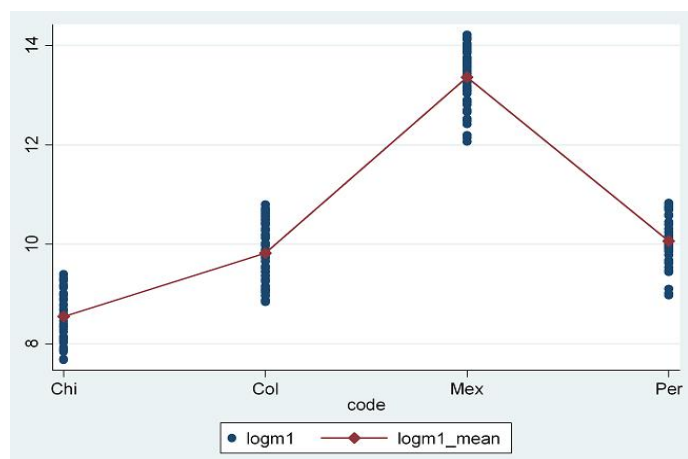


Figura 17.2: Heterogeneidad entre Individuos

```
. graph box logm1, over(code)
```

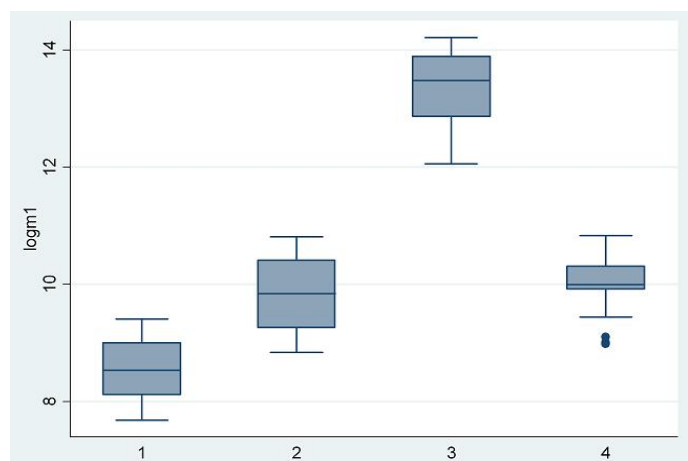


Figura 17.3: Caja y Bigote de la Heterogeindad entre Individuos

```
. *3ro PASO
. *****

. *Modelos Pooled

. reg logm1 loggdp l_rate
```

Source	SS	df	MS				
Model	104.602356	2	52.3011778	Number of obs = 220			
Residual	653.969477	217	3.01368423	F( 2, 217) = 17.35			
Total	758.571833	219	3.46379832	Prob > F = 0.0000			
				R-squared = 0.1379			
				Adj R-squared = 0.1299			
				Root MSE = 1.736			
logm1	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]		
loggdp	-.5519155	.1149364	-4.80	0.000	-.7784502	-.3253808	
l_rate	-.0274896	.0122999	-2.23	0.026	-.0517322	-.0032471	
_cons	16.50759	1.131745	14.59	0.000	14.27697	18.73821	

```
. *Guardamos la ecuación anterior
```

```
. estimates store betas_OLS
```

```
. *4to PASO
```

```
. *****
```

```
. *Efectos fijos
```

```
. xtreg logm1 loggdp l_rate ,fe
```

```
Fixed-effects (within) regression      Number of obs   =      220
Group variable: code                  Number of groups =       4
R-sq:  within = 0.9563                 Obs per group:  min =      55
      between = 0.2707                      avg =      55.0
      overall  = 0.1190                      max =      55
corr(u_i, Xb) = -0.7496                 F(2,214)        =    2340.34
                                      Prob > F         =     0.0000
```

logm1	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]		
loggdp	1.267732	.0293479	43.20	0.000	1.209884	1.32558	
l_rate	.0014483	.0015951	0.91	0.365	-.0016958	.0045924	
_cons	-2.318828	.3187803	-7.27	0.000	-2.947179	-1.690476	
sigma_u	3.0339649						
sigma_e	.11407446						
rho	.9985883	(fraction of variance due to u_i)					

```
F test that all u_i=0:      F(3, 214) = 16680.38      Prob > F = 0.0000
```

```
. *Guardamos la ecuación anterior
```

```
. estimates store betas_FE
```

```
. *Estimador de efectos fijos con variables dummy 1
.
. xi:regress logm1 loggdp l_rate i.code
i.code      _Icode_1-4      (naturally coded; _Icode_1 omitted)
Source      SS      df      MS      Number of obs =      220
Model       755.787054      5    151.157411      F( 5, 214) =11615.89
Residual    2.78477847     214    .013012984      Prob > F      = 0.0000
Total       758.571833     219    3.46379832      R-squared      = 0.9963
                        Adj R-squared = 0.9962
                        Root MSE   = .11407
```

	Source	SS	df	MS			
	Model	755.787054	5	151.157411			
	Residual	2.78477847	214	.013012984			
	Total	758.571833	219	3.46379832			

	logm1	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
	loggdp	1.267732	.0293479	43.20	0.000	1.209884 1.32558
	l_rate	.0014483	.0015951	0.91	0.365	-.0016958 .0045924
	_Icode_2	-.6793208	.0633168	-10.73	0.000	-.8041253 -.5545163
	_Icode_3	5.691146	.0273827	207.84	0.000	5.637172 5.745121
	_Icode_4	-.3745498	.0672343	-5.57	0.000	-.5070761 -.2420235
	_cons	-3.478147	.2927645	-11.88	0.000	-4.055218 -2.901075

```
. *Guardamos la ecuación anterior
. estimates store betas_FE_D1

. test _Icode_2 _Icode_3 _Icode_4
( 1) _Icode_2 = 0
( 2) _Icode_3 = 0
( 3) _Icode_4 = 0
      F( 3, 214) =16680.38
      Prob > F = 0.0000

. *Generamos dicotomicas que identifiquen cada pais
. g d1=pais==1
. g d2=pais==2
. g d3=pais==3
. g d4=pais==4

. *Estimador de efectos fijos con variables dummy 2
. regress logm1 loggdp l_rate d1 d2 d3 d4, nocons
```

	Source	SS	df	MS			
	Model	24783.0272	6	4130.50453			
	Residual	2.78477847	214	.013012984			
	Total	24785.812	220	112.662782			

	logm1	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
	loggdp	1.267732	.0293479	43.20	0.000	1.209884 1.32558
	l_rate	.0014483	.0015951	0.91	0.365	-.0016958 .0045924
	d1	-3.478147	.2927645	-11.88	0.000	-4.055218 -2.901075
	d2	-4.157467	.3516467	-11.82	0.000	-4.850602 -3.464333
	d3	2.213	.2766494	8.00	0.000	1.667693 2.758307
	d4	-3.852696	.355298	-10.84	0.000	-4.553028 -3.152364

```
. *Guardamos la ecuación anterior
. estimates store betas_FE_D2

. *5to PASO
. *****

. *Efectos aleatorios
. xtreg logm1 loggdp l_rate logpr,re

Random-effects GLS regression           Number of obs   =       220
Group variable: code                   Number of groups  =        4
R-sq:  within = 0.2024                 Obs per group:   min =       55
      between = 0.1994                      avg =      55.0
      overall  = 0.1997                      max =       55

Random effects u_i ~ Gaussian           Wald chi2(3)      =       53.89
corr(u_i, X)      = 0 (assumed)         Prob > chi2       =      0.0000
```

logm1	Coef.	Std. Err.	z	P> z	[95 % Conf. Interval]	
loggdp	-1.014515	.1586038	-6.40	0.000	-1.325373	-.7036571
l_rate	.0389616	.0201478	1.93	0.053	-.0005274	.0784506
logpr	3.605482	.8829798	4.08	0.000	1.874874	5.336091
_cons	3.789323	3.300892	1.15	0.251	-2.680307	10.25895
sigma_u	0					
sigma_e	.1142695					
rho	0	(fraction of variance due to u_i)				

```
. *Guardamos la ecuación anterior
. estimates store betas_RE

. *6mo PASO
. *****

. *Creación de una tabla para los coeficientes betas de las estimaciones anteriores
. *y comparación
.
. estimates table betas_OLS betas_FE betas_FE_D1 betas_FE_D2 betas_RE, ///
star stats(N r2_a r2_o r2_b r2_w sigma_u sigma_e rho aic bic)
```

Variable	betas_OLS	betas_FE	betas_FE_D1	betas_FE_D2	betas_RE
loggdp	-.55191547***	1.2677321***	1.2677321***	1.2677321***	-1.0145149***
l_rate	-.02748963*	.00144831	.00144831	.00144831	.03896163
_Icode_2			-.67932081***		
_Icode_3			5.6911463***		
_Icode_4			-.37454981***		

d1					-3.4781465***
d2					-4.1574673***
d3					2.2129998***
d4					-3.8526963***
logpr					3.6054824***
_cons		16.507594***	-2.3188276***	-3.4781465***	3.7893226
N		220	220	220	220
r2_a		.12994812	.95525748	.99624315	.9998845
r2_o			.11904283		.19967257
r2_b			.27066639		.19944302
r2_w			.95627899		.20239818
sigma_u			3.0339649		0
sigma_e			.11407446		.1142695
rho			.9985883		0
aic		870.00824	-330.94807	-324.94807	-324.94807
bic		880.18913	-320.76719	-304.58631	-304.58631

legend: \* p<0.05; \*\* p<0.01; \*\*\* p<0.001

```
. *Genero un tabla igual a lo anterior en excel (llamado producto_I)

. outreg2 [betas_OLS betas_FE betas_FE_D1 betas_RE] using producto_I, excel replace
producto_I.xml
dir : seeout
```

```
. *7mo PASO
. *****
```

```
. *TEST DE BPG
```

```
. xtreg logm1 loggdp l_rate ,re vce(robust)
```

```
Random-effects GLS regression           Number of obs   =       220
Group variable: code                    Number of groups  =        4
```

```
R-sq:  within = 0.9563                    Obs per group: min =       55
        between = 0.2710                  avg           =      55.0
        overall = 0.1189                  max           =       55
```

```
Random effects u_i ~ Gaussian           Wald chi2(2)      =      758.81
corr(u_i, X)      = 0 (assumed)         Prob > chi2       =      0.0000
```

(Std. Err. adjusted for 4 clusters in code)

logm1	Coef.	Robust Std. Err.	z	P> z	[95% Conf. Interval]	
loggdp	1.263901	.0542515	23.30	0.000	1.15757	1.370232
l_rate	.0012722	.0017168	0.74	0.459	-.0020926	.004637
_cons	-2.277054	1.964482	-1.16	0.246	-6.127368	1.57326
sigma_u	1.4530548					
sigma_e	.11407446					
rho	.99387446	(fraction of variance due to u_i)				

```
. xttest0
```

Breusch and Pagan Lagrangian multiplier test for random effects

```
logm1[code,t] = Xb + u[code] + e[code,t]
```

Estimated results:

	Var	sd = sqrt(Var)
logm1	3.463798	1.861128
e	.013013	.1140745
u	2.111368	1.453055

Test: Var(u) = 0

chi2(1) = 4485.81  
Prob > chi2 = 0.0000

```
. *Aleatorios es mejor que pooled
```

```
. *TEST DE HAUSMAN
```

```
. xtreg logm1 loggdp l_rate ,fe
```

Fixed-effects (within) regression  
Group variable: code

Number of obs = 220  
Number of groups = 4

R-sq: within = 0.9563  
between = 0.2707  
overall = 0.1190

Obs per group: min = 55  
avg = 55.0  
max = 55

corr(u\_i, Xb) = -0.7496

F(2,214) = 2340.34  
Prob > F = 0.0000

logm1	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
loggdp	1.267732	.0293479	43.20	0.000	1.209884	1.32558
l_rate	.0014483	.0015951	0.91	0.365	-.0016958	.0045924
_cons	-2.318828	.3187803	-7.27	0.000	-2.947179	-1.690476
sigma_u	3.0339649					
sigma_e	.11407446					
rho	.9985883	(fraction of variance due to u_i)				

F test that all u\_i=0: F(3, 214) = 16680.38 Prob > F = 0.0000

```
. estimates store fixed
```

```
. xtreg logm1 loggdp l_rate ,re
```

Random-effects GLS regression  
Group variable: code

Number of obs = 220  
Number of groups = 4

R-sq: within = 0.9563  
between = 0.2710  
overall = 0.1189

Obs per group: min = 55  
avg = 55.0  
max = 55

Random effects u\_i ~ Gaussian  
corr(u\_i, X) = 0 (assumed)

Wald chi2(2) = 4466.52  
Prob > chi2 = 0.0000

logm1	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
loggdp	1.263901	.0299864	42.15	0.000	1.205129	1.322673
l_rate	.0012722	.0016302	0.78	0.435	-.0019229	.0044673
_cons	-2.277054	.811419	-2.81	0.005	-3.867406	-.6867018
sigma_u	1.4530548					
sigma_e	.11407446					
rho	.99387446	(fraction of variance due to u_i)				

```
. estimates store random
```

```
. hausman fixed random,sigmamore
```

	Coefficients		(b-B) Difference	sqrt(diag(V_b-V_B)) S.E.
	(b) fixed	(B) random		
loggdp	1.267732	1.263901	.0038313	.0014306
l_rate	.0014483	.0012722	.0001761	.0000685

b = consistent under Ho and Ha; obtained from xtreg  
B = inconsistent under Ha, efficient under Ho; obtained from xtreg

Test: Ho: difference in coefficients not systematic

chi2(2) = (b-B)'[(V\_b-V\_B)^(-1)](b-B)  
= 11.53  
Prob>chi2 = 0.0031

```
. *se rechaza la hipotesis nula, mejor es efectos fijos que aleatorios
```

```
. *8vo PASO  
. *****
```

```
. *Dado el mejor modelo: EFECTOS FIJOS
```

```
. *Se evalua si es necesario usar "Time-Effects" (Two way)
```

```
. *Testiando time effects
```

```
. xi: xtreg logm1 loggdp l_rate i.year,fe  
i.year _Iyear_1-55 (naturally coded; _Iyear_1 omitted)
```

Fixed-effects (within) regression	Number of obs	=	220
Group variable: code	Number of groups	=	4
R-sq: within = 0.9737	Obs per group: min	=	55
between = 0.2674	avg	=	55.0
overall = 0.0757	max	=	55
	F(56,160)	=	105.80
corr(u_i, Xb) = -0.6526	Prob > F	=	0.0000

logm1	Coef.	Std. Err.	t	P> t	[95 % Conf. Interval]	
loggdp	.9041342	.0958689	9.43	0.000	.7148026	1.093466
l_rate	.0022614	.0026811	0.84	0.400	-.0030336	.0075563
_Iyear_2	.0244641	.0723771	0.34	0.736	-.1184735	.1674017
_Iyear_3	.0423186	.0728033	0.58	0.562	-.1014607	.1860979
_Iyear_4	.2243885	.0728079	3.08	0.002	.0806001	.3681769
_Iyear_5	.1647555	.07346	2.24	0.026	.0196792	.3098318
_Iyear_6	.2028327	.0743117	2.73	0.007	.0560744	.3495911
_Iyear_7	.2170709	.0745979	2.91	0.004	.0697475	.3643944
_Iyear_8	.3345421	.0747793	4.47	0.000	.1868603	.4822238
_Iyear_9	.2261649	.0753039	3.00	0.003	.0774472	.3748826
_Iyear_10	.2510923	.0757554	3.31	0.001	.1014829	.4007018
_Iyear_11	.2223255	.0779852	2.85	0.005	.0683124	.3763385
_Iyear_12	.3312363	.0794332	4.17	0.000	.1743635	.4881091
_Iyear_13	.2475885	.0777398	3.18	0.002	.0940601	.4011169
_Iyear_14	.317447	.0767372	4.14	0.000	.1658987	.4689954
_Iyear_15	.3112534	.0774796	4.02	0.000	.1582389	.464268
_Iyear_16	.4317114	.0788023	5.48	0.000	.2760846	.5873382
_Iyear_17	.2965809	.0811128	3.66	0.000	.1363911	.4567708
_Iyear_18	.3408118	.080979	4.21	0.000	.1808864	.5007373
_Iyear_19	.3203846	.0812502	3.94	0.000	.1599235	.4808457
_Iyear_20	.4452123	.0821397	5.42	0.000	.2829944	.6074302
_Iyear_21	.3306046	.0833187	3.97	0.000	.1660584	.4951509
_Iyear_22	.3635176	.0830068	4.38	0.000	.1995873	.527448
_Iyear_23	.3742054	.0833491	4.49	0.000	.2095992	.5388116
_Iyear_24	.506074	.0842567	6.01	0.000	.3396753	.6724727
_Iyear_25	.4189348	.085779	4.88	0.000	.2495297	.58834
_Iyear_26	.4522092	.0863341	5.24	0.000	.2817079	.6227106
_Iyear_27	.4602026	.0865912	5.31	0.000	.2891934	.6312118
_Iyear_28	.5376398	.0877836	6.12	0.000	.3642758	.7110038
_Iyear_29	.4251446	.0902647	4.71	0.000	.2468806	.6034085
_Iyear_30	.4490152	.0900736	4.98	0.000	.2711286	.6269017
_Iyear_31	.4285727	.0905972	4.73	0.000	.2496521	.6074933
_Iyear_32	.5283993	.0921649	5.73	0.000	.3463826	.7104159
_Iyear_33	.4442867	.0939688	4.73	0.000	.2587076	.6298658
_Iyear_34	.4598856	.0953521	4.82	0.000	.2715745	.6481966
_Iyear_35	.4491913	.0966262	4.65	0.000	.2583641	.6400186
_Iyear_36	.5415413	.098178	5.52	0.000	.3476494	.7354331
_Iyear_37	.4689264	.0997919	4.70	0.000	.2718473	.6660056
_Iyear_38	.4943226	.1010743	4.89	0.000	.2947108	.6939345
_Iyear_39	.4678318	.1027074	4.55	0.000	.2649947	.6706688
_Iyear_40	.5931801	.1038023	5.71	0.000	.3881807	.7981795
_Iyear_41	.4946224	.1073342	4.61	0.000	.2826479	.706597
_Iyear_42	.5252911	.1094521	4.80	0.000	.309134	.7414482
_Iyear_43	.5193787	.1109765	4.68	0.000	.300211	.7385465
_Iyear_44	.6500232	.1108873	5.86	0.000	.4310317	.8690147
_Iyear_45	.591606	.1141441	5.18	0.000	.3661825	.8170294
_Iyear_46	.5873623	.1158788	5.07	0.000	.3585132	.8162114
_Iyear_47	.6183196	.1167704	5.30	0.000	.3877095	.8489297
_Iyear_48	.6995092	.1197684	5.84	0.000	.4629784	.93604
_Iyear_49	.6112433	.1219573	5.01	0.000	.3703897	.852097
_Iyear_50	.6032773	.1242892	4.85	0.000	.3578184	.8487362
_Iyear_51	.6166061	.1235303	4.99	0.000	.372646	.8605662
_Iyear_52	.7305238	.1238529	5.90	0.000	.4859266	.9751211
_Iyear_53	.688161	.1216999	5.65	0.000	.4478156	.9285063
_Iyear_54	.7001898	.1212826	5.77	0.000	.4606686	.939711
_Iyear_55	.6856295	.1220343	5.62	0.000	.4446237	.9266352
_cons	.8941145	.9591038	0.93	0.353	-1.000021	2.78825
sigma_u	2.7173076					
sigma_e	.10231141					
rho	.99858435	(fraction of variance due to u_i)				



---

F test that all  $u_i=0$ :       $F(3, 160) = 12852.53$       Prob > F = 0.0000

. testparm \_Iyear\_\*

```
( 1) _Iyear_2 = 0
( 2) _Iyear_3 = 0
( 3) _Iyear_4 = 0
( 4) _Iyear_5 = 0
( 5) _Iyear_6 = 0
( 6) _Iyear_7 = 0
( 7) _Iyear_8 = 0
( 8) _Iyear_9 = 0
( 9) _Iyear_10 = 0
(10) _Iyear_11 = 0
(11) _Iyear_12 = 0
(12) _Iyear_13 = 0
(13) _Iyear_14 = 0
(14) _Iyear_15 = 0
(15) _Iyear_16 = 0
(16) _Iyear_17 = 0
(17) _Iyear_18 = 0
(18) _Iyear_19 = 0
(19) _Iyear_20 = 0
(20) _Iyear_21 = 0
(21) _Iyear_22 = 0
(22) _Iyear_23 = 0
(23) _Iyear_24 = 0
(24) _Iyear_25 = 0
(25) _Iyear_26 = 0
(26) _Iyear_27 = 0
(27) _Iyear_28 = 0
(28) _Iyear_29 = 0
(29) _Iyear_30 = 0
(30) _Iyear_31 = 0
(31) _Iyear_32 = 0
(32) _Iyear_33 = 0
(33) _Iyear_34 = 0
(34) _Iyear_35 = 0
(35) _Iyear_36 = 0
(36) _Iyear_37 = 0
(37) _Iyear_38 = 0
(38) _Iyear_39 = 0
(39) _Iyear_40 = 0
(40) _Iyear_41 = 0
(41) _Iyear_42 = 0
(42) _Iyear_43 = 0
(43) _Iyear_44 = 0
(44) _Iyear_45 = 0
(45) _Iyear_46 = 0
(46) _Iyear_47 = 0
(47) _Iyear_48 = 0
(48) _Iyear_49 = 0
(49) _Iyear_50 = 0
(50) _Iyear_51 = 0
(51) _Iyear_52 = 0
(52) _Iyear_53 = 0
(53) _Iyear_54 = 0
(54) _Iyear_55 = 0
```

F( 54,    160) =    1.96  
 Prob > F =    0.0007

```
. *El modelo debe incluir time effects
```

```
. *9no PASO
```

```
. *****
```

```
. *testeo si los residuos estan correlacionados entre entidades
```

```
. findit xtcsd
```

```
. xtreg logm1 loggdp l_rate ,fe
```

```
Fixed-effects (within) regression      Number of obs   =      220
Group variable: code                  Number of groups =       4

R-sq:  within = 0.9563                  Obs per group:  min =      55
      between = 0.2707                      avg   =     55.0
      overall  = 0.1190                      max   =      55

corr(u_i, Xb) = -0.7496                  F(2,214)        =    2340.34
                                          Prob > F         =     0.0000
```

logm1	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
loggdp	1.267732	.0293479	43.20	0.000	1.209884	1.32558
l_rate	.0014483	.0015951	0.91	0.365	-.0016958	.0045924
_cons	-2.318828	.3187803	-7.27	0.000	-2.947179	-1.690476
sigma_u	3.0339649					
sigma_e	.11407446					
rho	.9985883	(fraction of variance due to u_i)				

```
F test that all u_i=0:      F(3, 214) = 16680.38      Prob > F = 0.0000
```

```
. xtcsd, pesaran abs
```

```
Pesaran's test of cross sectional independence =      1.598, Pr = 0.1101
```

```
Average absolute value of the off-diagonal elements =      0.219
```

```
. *No hay dependencia entre cada unidad transversal
```

```
. *Si hubiera problema de autocorrelacion o ht entre entidad,
```

```
. *usar los errores estandar robustos
```

```
. *de xtscd "Driscoll and Kraay standard errors"
```

```
. *10mo PASO
```

```
. *****
```

```
. *Testing heteroscedasticidad
```

```
. xtreg logm1 loggdp l_rate ,fe
```

```
Fixed-effects (within) regression      Number of obs   =      220
Group variable: code                  Number of groups =       4

R-sq:  within = 0.9563                  Obs per group:  min =      55
      between = 0.2707                      avg   =     55.0
      overall  = 0.1190                      max   =      55
```

corr(u\_i, Xb) = -0.7496      F(2,214) = 2340.34  
 Prob > F = 0.0000

logm1	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
loggdp	1.267732	.0293479	43.20	0.000	1.209884	1.32558
l_rate	.0014483	.0015951	0.91	0.365	-.0016958	.0045924
_cons	-2.318828	.3187803	-7.27	0.000	-2.947179	-1.690476
sigma_u	3.0339649					
sigma_e	.11407446					
rho	.9985883	(fraction of variance due to u_i)				

F test that all u\_i=0:      F(3, 214) = 16680.38      Prob > F = 0.0000

. xttest3

Modified Wald test for groupwise heteroskedasticity  
 in fixed effect regression model

H0: sigma(i)^2 = sigma^2 for all i

chi2 (4) = 82.91

Prob>chi2 = 0.0000

. \*Si no existe instalar: findit xttest3  
 . \*Se rechaza la hipotesis nula y existe ht  
 . \*corregir con la opcion robust.

. \*Testing autocorrelacion

. xtserial logm1 loggdp l\_rate

Wooldridge test for autocorrelation in panel data

H0: no first-order autocorrelation

F( 1, 3) = 0.102

Prob > F = 0.7703

. \*No hay autocorrelation, no se rechaza la hipotesis nula

. \*11vo PASO  
 . \*\*\*\*\*

. \*Tomando en cuenta los problemas encontrados, aqui se tiene el mejor modelo

. xi: xtreg logm1 loggdp l\_rate i.year,fe vce(robust)  
 i.year      \_Iyear\_1-55      (naturally coded; \_Iyear\_1 omitted)

Fixed-effects (within) regression      Number of obs = 220  
 Group variable: code      Number of groups = 4

R-sq:    within = 0.9737      Obs per group: min = 55  
           between = 0.2674                            avg = 55.0  
           overall = 0.0757                            max = 55

corr(u\_i, Xb) = -0.6526      F(3,3) = .  
 Prob > F = .

(Std. Err. adjusted for 4 clusters in code)

logm1	Coef.	Robust Std. Err.	t	P> t	[95% Conf. Interval]	
loggdp	.9041342	.201337	4.49	0.021	.26339	1.544878
l_rate	.0022614	.0027875	0.81	0.477	-.0066096	.0111323
_Iyear_2	.0244641	.0203292	1.20	0.315	-.0402324	.0891606
_Iyear_3	.0423186	.0446591	0.95	0.413	-.0998065	.1844437
_Iyear_4	.2243885	.0788259	2.85	0.065	-.0264708	.4752478
_Iyear_5	.1647555	.0686025	2.40	0.096	-.0535684	.3830794
_Iyear_6	.2028327	.0922517	2.20	0.115	-.0907533	.4964187
_Iyear_7	.2170709	.1231619	1.76	0.176	-.1748851	.6090269
_Iyear_8	.3345421	.0671376	4.98	0.016	.1208804	.5482038
_Iyear_9	.2261649	.1072274	2.11	0.125	-.1150806	.5674104
_Iyear_10	.2510923	.1423951	1.76	0.176	-.2020726	.7042572
_Iyear_11	.2223255	.1659424	1.34	0.273	-.3057774	.7504283
_Iyear_12	.3312363	.1166043	2.84	0.066	-.0398506	.7023233
_Iyear_13	.2475885	.1514211	1.64	0.201	-.2343011	.7294782
_Iyear_14	.317447	.1572253	2.02	0.137	-.182914	.8178081
_Iyear_15	.3112534	.1756659	1.77	0.175	-.2477937	.8703006
_Iyear_16	.4317114	.0979449	4.41	0.022	.1200072	.7434157
_Iyear_17	.2965809	.1168066	2.54	0.085	-.0751497	.6683116
_Iyear_18	.3408118	.1302598	2.62	0.079	-.073733	.7553567
_Iyear_19	.3203846	.1287391	2.49	0.089	-.0893206	.7300899
_Iyear_20	.4452123	.0653147	6.82	0.006	.2373519	.6530727
_Iyear_21	.3306046	.0931123	3.55	0.038	.0342798	.6269295
_Iyear_22	.3635176	.1036726	3.51	0.039	.033585	.6934502
_Iyear_23	.3742054	.1271968	2.94	0.060	-.0305915	.7790024
_Iyear_24	.506074	.0573184	8.83	0.003	.3236612	.6884868
_Iyear_25	.4189348	.0798912	5.24	0.014	.1646855	.6731842
_Iyear_26	.4522092	.0736948	6.14	0.009	.2176794	.6867391
_Iyear_27	.4602026	.0887808	5.18	0.014	.1776623	.7427429
_Iyear_28	.5376398	.0350672	15.33	0.001	.4260404	.6492392
_Iyear_29	.4251446	.0987511	4.31	0.023	.1108744	.7394147
_Iyear_30	.4490152	.074895	6.00	0.009	.2106659	.6873645
_Iyear_31	.4285727	.0787774	5.44	0.012	.1778678	.6792777
_Iyear_32	.5283993	.073017	7.24	0.005	.2960265	.7607721
_Iyear_33	.4442867	.1146978	3.87	0.030	.079267	.8093063
_Iyear_34	.4598856	.1199835	3.83	0.031	.0780447	.8417265
_Iyear_35	.4491913	.1226731	3.66	0.035	.0587909	.8395918
_Iyear_36	.5415413	.1168944	4.63	0.019	.1695311	.9135514
_Iyear_37	.4689264	.1569907	2.99	0.058	-.0306879	.9685408
_Iyear_38	.4943226	.1272317	3.89	0.030	.0894145	.8992307
_Iyear_39	.4678318	.1252363	3.74	0.033	.069274	.8663896
_Iyear_40	.5931801	.1187517	5.00	0.015	.2152593	.9711009
_Iyear_41	.4946224	.1523003	3.25	0.048	.0099349	.97931
_Iyear_42	.5252911	.1193485	4.40	0.022	.1454709	.9051113
_Iyear_43	.5193787	.119718	4.34	0.023	.1383828	.9003747
_Iyear_44	.6500232	.1243715	5.23	0.014	.2542177	1.045829
_Iyear_45	.591606	.1260949	4.69	0.018	.1903158	.9928962
_Iyear_46	.5873623	.1105153	5.31	0.013	.2356534	.9390713
_Iyear_47	.6183196	.1239071	4.99	0.015	.2239919	1.012647
_Iyear_48	.6995092	.1315727	5.32	0.013	.2807861	1.118232
_Iyear_49	.6112433	.1518297	4.03	0.028	.1280535	1.094433
_Iyear_50	.6032773	.1426512	4.23	0.024	.1492974	1.057257
_Iyear_51	.6166061	.1495451	4.12	0.026	.1406868	1.092525
_Iyear_52	.7305238	.1388305	5.26	0.013	.2887031	1.172345
_Iyear_53	.688161	.1674338	4.11	0.026	.1553118	1.22101
_Iyear_54	.7001898	.166793	4.20	0.025	.1693799	1.231
_Iyear_55	.6856295	.1577677	4.35	0.022	.1835421	1.187717
_cons	.8941145	2.056735	0.43	0.693	-5.651334	7.439563

sigma_u	2.7173076	
sigma_e	.10231141	
rho	.99858435	(fraction of variance due to u_i)

## 17.6. Ejercicio Propuesto

En el archivo **prod.arroz.csv** se tiene información anual 1997-2009 para las siguientes provincias: Lambayeque, Chiclayo y Ferreñafe. El objetivo es estimar una función de producción para el arroz usando un modelo panel, las variables que se usarán son: *pd* (producción de arroz en TM), *sc* (superficie cosechada en ha.), *pr* (precio real en soles/TM), *tmin* y *pp* son la temperatura mínima (°C) y la precipitación (mm) respectivamente, tambien se le agregó sus términos cuadráticos de estas últimas variables. El modelo a estimar sería el siguiente:

$$pd_{it} = \beta_1 + \beta_2 sc_{it} + \beta_3 pr_{it} + \beta_4 tmin + \beta_5 tmin_{it}^2 + \beta_6 pp_{it} + \beta_7 pp_{it}^2 + u_{it}$$

El objetivo de este ejercicio es estimar los tres tipos de modelos panel, escoger el mejor modelo y corregir si existe problemas de autocorrelación y/o heteroscedasticidad.