

Taller Stata

Clase 1

Javiera Vásquez

Introducción

- Stata es una aplicación completa e integrada, basada en comandos, que tiene todos los elementos necesarios para el análisis estadístico.
- Existen 3 formas de trabajar en STATA:
 - Con las ventanas
 - En la ventana de comando (requiere conocer los comandos)
 - Do-file (programación, también requiere conocer los comandos)

¿Cómo se ve STATA?

- Al abrir el programa rápidamente podemos distinguir 4 ventanas:
 - review: aparecen los comandos que han sido ejecutados sin y con error en una sesión (bitacora)
 - results: muestra los resultados luego de la ejecución de un comando
 - variables: presenta el listado de variables de la base de datos, así como su descripción
 - commands: ventada donde se introducen los comandos

¿Cómo se ve STATA?

The screenshot shows the Stata/SE 12.0 interface. The top menu bar includes 'Review' (selected), 'File' (with Open, Save, Print options), 'Help' (with More, Break, Search Help), and 'Stata/SE 12.0' (with Log, Viewer, Graph, Do-file Editor, Data Editor, Data Browser icons). The main window has tabs for 'Review' (selected) and 'Results'. The 'Results' tab displays the following text:

Copyright 1985-2011 StataCorp LP
StataCorp
4905 Lakeway Drive
College Station, Texas 77845 USA
800-STATA-PC <http://www.stata.com>
979-696-4600 stata@stata.com
979-696-4601 (fax)

Single-user Stata network perpetual license:
Serial number: 93611859953
Licensed to: STATAforAll
STATA

Notes:
1. (-set maxvar-) 5000 maximum variables

The bottom of the Results window shows a 'Command' field with a single slash character (/).

To the right of the Results window is a 'Variables' table with columns for 'Name' and 'Label'. Below it is a 'Properties' panel with sections for Variables (Name, Label, Type, Format, Value Label, Notes) and Data (Filename, Label, Notes, Variables: 0, Observations: 0, Size: 0, Memory: 64M).

¿dónde estamos trabajando?: Directorio

- Una forma de facilitar el trabajo en Stata, especialmente cuando trabajamos con do-file (programa) consiste en definir la(s) carpeta(s) de trabajo.
- Esto permite acceder y guardar fácilmente a los archivos
- Para cambiar el directorio, se utiliza el comando:

cd

- Por ejemplo:

```
cd "G:\FNE\FEN_Taller Stata\Bases de datos"
```

¿dónde estamos trabajando?: Directorio

- Para saber en que directorio estamos ubicados utilizamos el comando:
- Por ejemplo:

```
pwd
```

```
. cd "G:\FNE\FEN_Taller Stata\Bases de datos"  
G:\FNE\FEN_Taller Stata\Bases de datos  
  
. pwd  
G:\FNE\FEN_Taller Stata\Bases de datos
```

¿dónde estamos trabajando?: Directorio

- Un error común consiste en no ocupar comillas, y tener nombres de carpeta con espacio:

```
. cd G:\FNE\FEN_Taller Stata\Bases de datos  
invalid syntax  
r(198);
```

Cargar base de datos. Paso 1: memoria

- De la versión 12 de Stata en adelante, no es necesario aumentar la memoria para cargar bases de datos, se ajusta automáticamente.
- Para versiones anteriores el programa viene con 50 megabytes por defecto, esto significa que si queremos abrir una base de datos con mayor tamaño nos entregará un error:

```
no room to add more observations
```

- El comando para aumentar la memoria en versiones anteriores a la 12 es:

```
set mem # [b|k|m|g]
```

Cargar base de datos. Paso 2: limpiar

- Cuando abrimos STATA e cargamos por primera vez la base de datos en la sesión, no es necesario limpiar.
- Cuando ya estamos trabajando en STATA y hemos modificado la base de datos, y queremos abrir otra base de datos, debemos limpiar el STATA antes con el comando:

```
clear
```

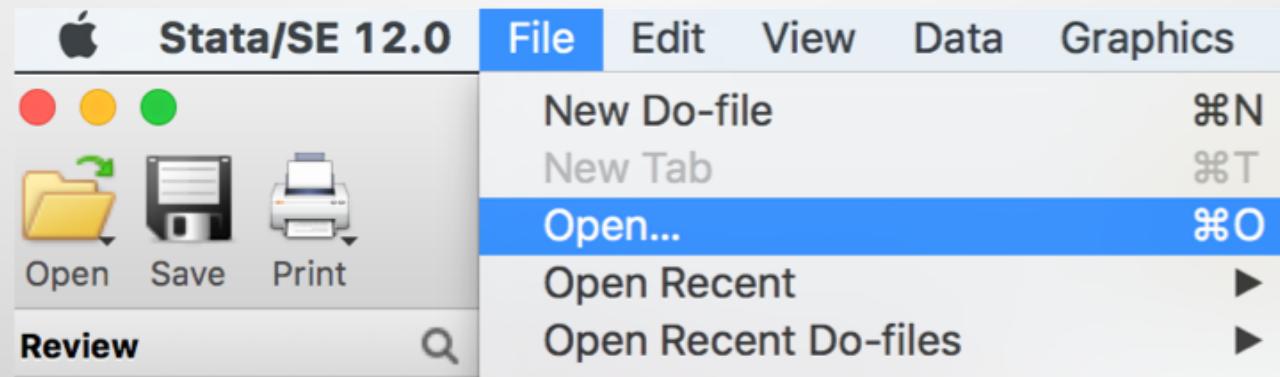
- Sino nos entregará el siguiente error:

```
no; data in memory would be lost
```

Cargar base de datos. Paso 3: abrir

1. Base de datos en formato .dta:

- a. Apretar el icono , buscar la ubicación de la base de datos y abrir (también se puede ir a File/Open):



Cargar base de datos. Paso 3: abrir

1. Base de datos en formato .dta:

b. Usar directamente el comando **use** en la ventana de comando (o en do-file):

- Si ya le indicamos a STATA la carpeta donde estamos trabajando con el comando **cd**:

```
use pcobre.dta [, clear]
```

- Si no le hemos indicado la carpeta donde estamos trabajando antes:

```
use "G:\FNE\FEN_Taller Stata\Bases de datos\pcobre.dta"
```

Cargar base de datos. Paso 3: abrir

2. Base de datos en formato excel:
 - a. Se puede copiar la planilla de cálculo de excel y copiar directamente en el editor de datos de Stata
 - Tener cuidado con el formato de los decimales, para STATA el “.” corresponde al decimal
 - Los número deben estar sin formato “moneda” u otras opciones
 - Los números deben estar sin separador de miles
 - b. Utilizar el comando:

```
import excel
```

Ejemplo: cargar datos desde excel

	A	B	C	D	E	F	G
1	Año	mes	AFP	Cotización Adicional	Comisión Fija por Cotizacion	Comisión Fija por mantención de saldo	Comisión porcentual por mantencion de saldo
2	1981	5	Alameda	2.63%	\$ 228		0.60%
3	1981	5	Concordia	2.61%	\$ 3		2.50%
4	1981	5	Cuprum	2.63%			2.50%
5	1981	5	El Libertador	2.75%		\$ 16	1.50%
6	1981	5	Habitat	2.50%		\$ 12	1.20%
7	1981	5	Invierta	2.40%	\$ 12		1.68%
8	1981	5	Magister				
9	1981	5	Planvital	2.50%	\$ 139		1.92%
10	1981	5	Provida	2.63%		\$ 15	0.80%
11	1981	5	San Cristóbal	2.50%	\$ 98		1.90%
12	1981	5	Santa María	2.50%			2.10%
13	1981	5	Summa	2.65%		\$ 96	1.68%
14	1981	6	Alameda	2.63%	\$ 228		0.60%
15	1981	6	Concordia	2.61%	\$ 3		2.50%
16	1981	6	Cuprum	2.63%			2.50%
17	1981	6	El Libertador	2.75%		\$ 16	1.50%

Ejemplo: cargar datos desde excel

- Copiar y pegar en el editor de Stata:
 - Primero se deben eliminar todos los formatos de los números, y que los decimales estén con punto:

	A	B	C	D	E	F	G
1	Año	mes	AFP	Cotización Adicional	Comisión Fija por Cotizacion	Comisión Fija por mantención de saldo	Comisión porcentual por mantencion de saldo
2	1981	5	Alameda	0.02630	228		0.00600
3	1981	5	Concordia	0.02610	3		0.02500
4	1981	5	Cuprum	0.02630			0.02500
5	1981	5	El Libertador	0.02750		16	0.01500
6	1981	5	Habitat	0.02500		12	0.01200
7	1981	5	Invierta	0.02400	12		0.01680
8	1981	5	Magister				
9	1981	5	Planvital	0.02500	139		0.01920
10	1981	5	Provida	0.02630		15	0.00800
11	1981	5	San Cristóbal	0.02500	98		0.01900
12	1981	5	Santa María	0.02500			0.02100
13	1981	5	Summa	0.02650		96	0.01680
14	1981	6	Alameda	0.02630	228		0.00600
15	1981	6	Concordia	0.02610	3		0.02500
16	1981	6	Cuprum	0.02630			0.02500
17	1981	6	El Libertador	0.02750		16	0.01500

Ejemplo: cargar datos desde excel

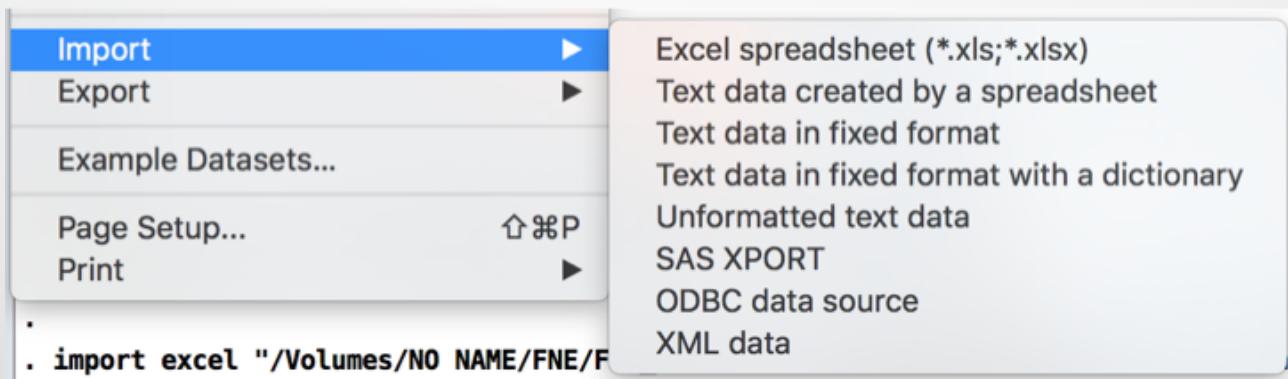
- Copiar y pegar en el editor de Stata:
 - Luego se seleccionan las columnas
 - Se abre el editor de Stata, tipeando **edit** en la ventana de comando o pinchando 
 - Se pega lo copiando en excel, Stata le preguntará si quiere tratar la primera observación como e nombre de la variable o como dato.
- Ocupar el comando **import excel**:

```
import excel "base.xls" sheet ("Comisiones 1981-1987") firstrow clear
```

Si no le indique la carpeta con el comando **cd**, debo entregar la ruta completa

Cargar base de datos. Paso 3: abrir

2. Base de datos en otros formatos:
 - a. Aceptados por el comando import:

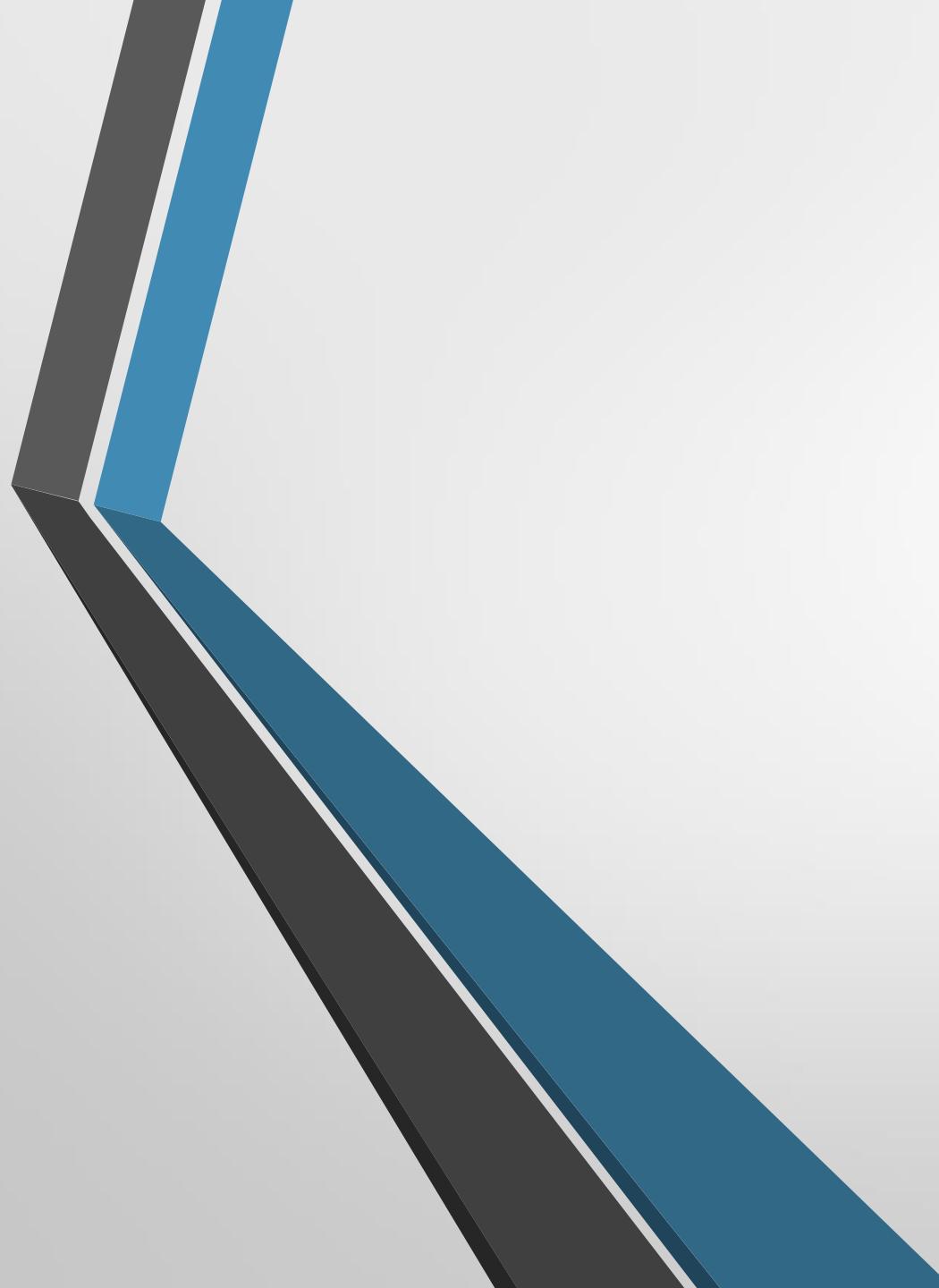


- b. No aceptados por el comando import: Stattransfer

Guardar la base de datos

- Cuando importamos una base de datos desde otro formato, esta no se guarda automáticamente en formato Stata, por lo que si cerramos el programa perderemos la importación de la base de datos.
- Lo mismo sucede si luego de abrir la base, hemos generado nuevas variables, etiquetado variables, o pegado información de otras bases de datos, sino guardamos esta base, al cerrar el programa se pierde todo lo realizados
- El comando para guardar la base de datos es **save**:

```
save "base_excel.dta" [, replace]
```



Taller Stata

Clase 2

Javiera Vásquez

Sintaxis general de un comando en STATA

- Salvo algunas excepciones, el comando básico de STATA tiene la siguiente forma:

```
[by varlist:] command [varlist] [=exp] [if exp] [weight] [, options]
```

- by varlist: indica que el comando se debe repetir para todos los valores de la variable en el listado
- varlist: indica el listado de variables sobre el cual se ejecuta el comando, sino se indica, asume que es a todas las variables
- if exp: restringe el alcance del comando a las observaciones que cumplen con exp.
- weight: ponderador
- , options: opciones específicas de cada comando

Instalación de nuevos comandos

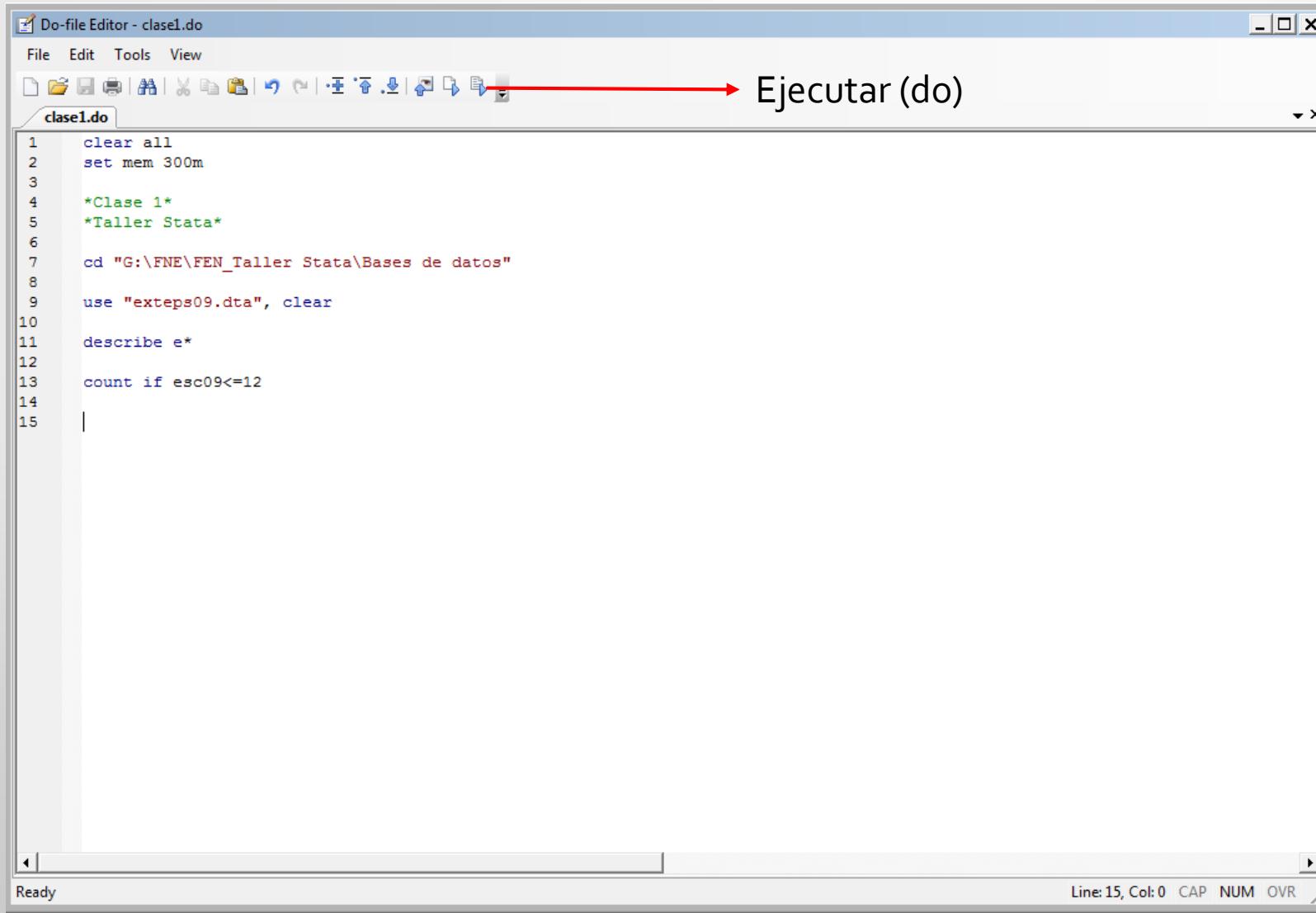
- Algunos comandos no están disponible en Stata y deben ser instalados, esto se hace con el siguiente comando:

```
ssc install
```

Do-file: trabajar en forma ordenada

- Un do-file es un archivo texto reconocido por STATA que contiene los comandos que se quieren ejecutar.
- Cada línea de este archivo texto representa una línea de comando
- Las ventajas de trabajar con un do-file es que:
 - Ordenar el trabajo
 - Corregir errores y hacer modificaciones sin necesidad de comenzar desde o.
 - Reproducir y verificar los resultados
- Para abrir el editor de do-files se debe pinchar el botón 

Do-file: trabajar en forma ordenada



The screenshot shows a Stata Do-file Editor window titled "Do-file Editor - clase1.do". The window has a menu bar with File, Edit, Tools, and View. Below the menu is a toolbar with various icons. A red arrow points from the toolbar to the "Ejecutar (do)" button. The main text area contains a Stata do-file script:

```
1  clear all
2  set mem 300m
3
4  *Clase 1*
5  *Taller Stata*
6
7  cd "G:\FNE\FEN_Taller Stata\Bases de datos"
8
9  use "exteps09.dta", clear
10
11 describe e*
12
13 count if esc09<=12
14
15 |
```

The status bar at the bottom right indicates "Line: 15, Col: 0 CAP NUM OVR".

Do-file: trabajar en forma ordenada

- Otra forma de ejecutar un do-file es con el comando **do**:

```
do clase1.do
```

- El do-file también se puede ejecutar de manera parcial seleccionando las líneas que se quieren ejecutar.
- Para desactivar una línea de comando del do-file se pone “*”, esto sirve para hacer comentarios
- Cuando un comando es muy largo y se quiere continuar en la siguiente línea se utiliza “///”

```
describe esc09 edad09 ///
numper sexo09
```

Exploración de los datos

- Los comandos `browse` y `edit` abren el visor y el editor de datos, respectivamente.
- El comando `describe` entrega una descripción general de toda la base de datos o de un sub-conjunto de variables. Esta descripción incluye el tipo de variable (donde la gran diferencia es entre numérico y *string* (no numérico), y la descripción de la variable (variable label).
- En el editor, las variables en negro y azul son numéricas, la diferencia es que la en color azul son variables numéricas pero etiquetadas (con label). Las variables en rojo son *string*.

Describe

Descripción de la variable

. describe			
Contains data from G:\FNE\FEN_Taller Stata\Bases de datos\exteps09.dta			
obs: 14,243			
vars: 29			
size: 1,865,833			
6 Jan 2014 11:07			
variable name storage display value variable type format label label			
folio double %9.0g			
edad09 int %9.0g			a9. ¿qué edad tiene ud.?
esc09 float %9.0g			Años de escolaridad EPS09
sexo09 byte %9.0g			a8. sexo
b2_09 float %32.0g b2			Estatus laboral Abril 2009
b4_09 float %32.0g b4			Region en que trabaja (Abril 2009)
b6_09 float %32.0g b6			Tipo de trabajo (Abril 2009)
b8_09 float %32.0g b8			Categoría Ocupacional (Abril 2009)
b9a_09 float %9.0g b9a			Tiene contrato (Abril 2009)
b9b_09 float %32.0g b9b			Relación Contractual (Abril 2009)
b10_09 float %40.0g b10			Horario (Abril 2009)
b11_09 float %32.0g b11			Entrega boleta de honorarios (Abril 2009)
b12_09 float %9.0g			Ingreso líquido mensual promedio (Abril 2009)
b13_09 float %9.0g			Horas semanales trabajadas (Abril 2009)
b14_09 float %42.0g b14			Lugar donde realiza la actividad laboral (Abril 2009)
b15_09 float %9.0g			Tamaño de la empresa (Abril 2009)
b15t_09 float %32.0g b15t			Tramos de tamaño de la empresa (Abril 2009)
b16_09 float %32.0g b16			Se encontraba afiliado a algun sindicato (Abril 2009)
b17_09 float %32.0g b17			Institución de afiliación seguro de accidentes del trabajo (Abril 2009)
b18_09 float %60.0g b18			Se encuentra cotizando (Abril 2009)
ocu_0609 double %9.0g			Meses ocupado entre Enero de 2006 y Abril de 2009
ces_0609 double %9.0g			Meses cesante entre Enero de 2002 y Septiembre de 2006
inact_0609 double %9.0g			Meses inactivo entre Enero de 2006 y Abril de 2009
cot_0609 double %9.0g			Meses cotizados entre Enero de 2006 y Abril de 2009
pcot float %9.0g			Proporción del tiempo cotizado entre Enero 2004 y Septiembre 2006
pcot_ocu float %9.0g			Proporción del tiempo ocupado cotizado entre Enero 2004 y Septiembre 2006
ing_entrev_me~1 float %9.0g			
numper float %9.0g			
ytotal float %9.0g			

Indica el formato,
string es no numérico

Indica si la variable tiene un etiquetado en sus categorías (diccionario de códigos)

Edit/Browse

Esta variable es numérica pero está etiquetada

	folio	edad09	esc09	sexo09	b2_09	
1	2409302	19	12	2	cesante	
2	2403092	19	13	2	trabajando	
3	2405087	20	13	1	trabajando	
4	2400943	20	12	2	trabajando	
5	2403511	21	16	1	inactivo	
6	2402495	21	12	2	trabajando	
7	2418471	21	16	2	trabajando	
8	2406608	21	12	1	cesante	
9	2406586	21	12	2	trabajando	
10	1280487	22	12	1	trabajando	
11	2416135	22	12	2	trabajando	
12	2400922	22	16	1	trabajando	
13	2413751	22	13	1	inactivo	
14	575478	22	16	1	inactivo	
15	2410750	22	12	1	trabajando	
16	2409305	22	.	2	inactivo	
17	2401728	22	12	2	trabajando	
18	2400058	22	16	2	inactivo	

label list (diccionario de variables etiquetadas)

- Con el comando `label list`, podemos ver el diccionario (*value label*) de las variables etiquetadas, por ejemplo:

```
. d b2_09

      storage   display     value
variable name    type    format     label     variable label
b2_09          float   %32.0g     b2       Estatus laboral Abril 2009

. label list b2
b2:
      -9 falta en el sistema
      -8 error en cálculo numérico
      -7 elementos de respuesta filtrados
      -6 el usuario se retractó
      -5 no contesta
      -4 no sabe/no está seguro
      -3 no aplicable
      -2 saltada por el usuario
      -1 no se preguntó
      1 trabajando
      2 cesante
      3 buscando trabajo por 1ra. vez
      4 inactivo
```

Ordenar los datos

- Los comandos **sort** y **gsort** permiten ordenar la base de datos en orden ascendente o descendente de acuerdo a una o más variables
- Por ejemplo, si queremos ordenar la base de datos de acuerdo al folio, en orden ascendente:

```
sort folio
```

- Y por ejemplo, si queremos ordenar la base de datos en orden ascendente de mayor a menor edad:

```
gsort -edad09
```

Ordenar las variables

- El comando **order** permite ordenar las variables de izquierda a derecha
- Por ejemplo, si quiero que la variable numper quede después de sexo09, y el resto mantenga el orden:

```
order folio edad09 esc09 sexo09 numper
```

Contar número de observaciones

- El comando **count** cuenta el número de observaciones:

```
. count  
14243  
  
. count if edad09<=30  
1354  
  
. count if sexo==1  
6950
```

- Total de observaciones
- Observaciones edad09 menor o igual a 30
- Observaciones con sexo09 igual a 1 (hombres)

Examinando los datos

- Otro comando útil que es útil para examinar los datos es:
`misstable sum`
- Este comando reporta el número de *missing values* contenidos en cada variable:

The screenshot shows the Stata command `. misstable sum edad09 esc09 sexo09 b2_09` followed by its output. The output includes a header row with columns for Variable, Obs=., Obs>., Obs<., Unique values, Min, and Max. Below this, there are two rows for variables `edad09` and `b2_09`. The `Obs<.` column for `edad09` contains the value `14,124`, which is circled in red. A red arrow points from this circled value to a box labeled "missing value extendido". The `Obs=.` column for `edad09` contains the value `119`, which is also circled in red. A red arrow points from this circled value to a box labeled "missing value del sistema". The `Obs>.` column for `edad09` contains the value `14,241`, which is circled in red. A red arrow points from this circled value to a box labeled "no missing value".

Variable	Obs=.	Obs>.	Obs<.	Unique values	Min	Max
edad09	119	14,241	14,124	22	0	21
b2_09	2		14,241	4	1	4

Estadísticas descriptivas

- El comando **summarize**, el que se puede abbreviar simplemente como **sum**, entrega estadísticas descriptivas básicas como:
 - Número de observaciones
 - Promedio (mean)
 - Desviación estándar (Std. Dev.)
 - Mínimo (Min)
 - Máximo (Max)

```
. sum edad09 esc09 b13_09
```

Variable	Obs	Mean	Std. Dev.	Min	Max
edad09	14243	49.93239	15.22389	19	108
esc09	14124	9.696616	4.324177	0	21
b13_09	8309	94.71645	211.105	2	999

Estadísticas descriptivas

- Podemos utilizar el comando **summarize** con la opción **by varlist**:

```
. by sexo09: sum esc09  
not sorted  
r(5);  
  
. bys sexo09: sum esc09  
  
-> sexo09 = 1  
  
Variable | Obs Mean Std. Dev. Min Max  
esc09 | 6897 9.664492 4.307995 0 21  
  
-> sexo09 = 2  
  
Variable | Obs Mean Std. Dev. Min Max  
esc09 | 7227 9.727273 4.33964 0 21
```

Estadísticas descriptivas

- También se puede utilizar el comando **summarize** filtros (**if**):

```
. sum esc09 if sexo09==1
```

Variable	Obs	Mean	Std. Dev.	Min	Max
esc09	6897	9.664492	4.307995	0	21

- También se puede ocupar el comando **summarize** con la opción **, detail**:

```
. sum esc09, detail
```

Años de escolaridad EPS09

	Percentiles	Smallest		
1%	0	0		
5%	2	0		
10%	3	0	Obs	14124
25%	6	0	Sum of Wgt.	14124
50%	12		Mean	9.696616
		Largest	Std. Dev.	4.324177
75%	12	21		
90%	15	21	Variance	18.69851
95%	17	21	Skewness	-.3412399
99%	17	21	Kurtosis	2.465894

Estadísticas descriptivas

- El comando **tabulate** (o **tab**) permite hacer tablas de distribución de frecuencias, las que muestran en número de observaciones en cada una de las categorías de la variable, así como el porcentaje y el porcentaje acumulado, estas se conocen como tablas de una entrada:

Estatus laboral Abril 2009	Freq.	Percent	Cum.
trabajando	8,309	58.35	58.35
cesante	1,274	8.95	67.29
buscando trabajo por 1ra. vez	13	0.09	67.38
inactivo	4,645	32.62	100.00
Total	14,241	100.00	

N – frecuencia absoluta % - frecuencia relativa

Estadísticas descriptivas

- Podemos incorporar el *missing value* como una categoría:

Estatus laboral Abril 2009	Freq.	Percent	Cum.
trabajando	8,309	58.34	58.34
cesante	1,274	8.94	67.28
buscando trabajo por 1ra. vez	13	0.09	67.37
inactivo	4,645	32.61	99.99
.	2	0.01	100.00
Total	14,243	100.00	

- O podemos hacer una tabla donde se comparan ambas, considerando o no considerando el *missing value* como una categoría, para esto se usa el comando **fre**, sino está instalado en su computador, recuerde ocupar el comando **ssc install** para instalarlo.

Estadísticas descriptivas

```
. ssc install fre  
checking fre consistency and verifying not already installed...  
installing into c:\ado\plus\...  
installation complete.
```

```
. fre b2_09
```

b2_09 — Estatus laboral Abril 2009

		Freq.	Percent	Valid	Cum.
Valid	1 trabajando	8309	58.34	58.35	58.35
	2 cesante	1274	8.94	8.95	67.29
	3 buscando trabajo por 1ra. vez	13	0.09	0.09	67.38
	4 inactivo	4645	32.61	32.62	100.00
	Total	14241	99.99	100.00	
Missing	.	2	0.01		
	Total	14243	100.00		

Estadísticas descriptivas

- Para hacer una tabla de dos entradas, se ponen las dos variables:

. tab b2_09 sexo09			
Estatus laboral Abril 2009	a8. sexo		
	1	2	Total
trabajando	5,060	3,249	8,309
cesante	534	740	1,274
buscando trabajo por	5	8	13
inactivo	1,350	3,295	4,645
Total	6,949	7,292	14,241

- Esta sólo nos entrega la frecuencia absoluta, es decir el número de observaciones

Estadísticas descriptivas

- Si queremos que nos muestre los porcentajes, tenemos varias alternativas:

- Que sume el 100% en las columnas:

. tab b2_09 sexo09, col nofreq

Estatus laboral Abril 2009	a8. sexo		Total
	1	2	
trabajando	72.82	44.56	58.35
cesante	7.68	10.15	8.95
buscando trabajo por	0.07	0.11	0.09
inactivo	19.43	45.19	32.62
Total	100.00	100.00	100.00

Es para que no muestre el número de observaciones

- Que sume el 100% en las filas:

. tab b2_09 sexo09, row nofreq

Estatus laboral Abril 2009	a8. sexo		Total
	1	2	
trabajando	60.90	39.10	100.00
cesante	41.92	58.08	100.00
buscando trabajo por	38.46	61.54	100.00
inactivo	29.06	70.94	100.00
Total	48.80	51.20	100.00

Estadísticas descriptivas

- Que sume el 100% en el total:

```
. tab b2_09 sexo09, cell nofreq
```

Estatus laboral Abril 2009	a8. sexo		Total
	1	2	
trabajando	35.53	22.81	58.35
cesante	3.75	5.20	8.95
buscando trabajo por inactivo	0.04	0.06	0.09
	9.48	23.14	32.62
Total	48.80	51.20	100.00

Estadísticas descriptivas

- También podemos ocupar **by** e **if**:

```
. bys sexo09: tab b2_09
```

-> sexo09 = 1

Estatus laboral Abril 2009	Freq.	Percent	Cum.
trabajando	5,060	72.82	72.82
cesante	534	7.68	80.50
buscando trabajo por 1ra. vez	5	0.07	80.57
inactivo	1,350	19.43	100.00
Total	6,949	100.00	

-> sexo09 = 2

Estatus laboral Abril 2009	Freq.	Percent	Cum.
trabajando	3,249	44.56	44.56
cesante	740	10.15	54.70
buscando trabajo por 1ra. vez	8	0.11	54.81
inactivo	3,295	45.19	100.00
Total	7,292	100.00	

```
. tab b2_09 if sexo09==1
```

Estatus laboral Abril 2009	Freq.	Percent	Cum.
trabajando	5,060	72.82	72.82
cesante	534	7.68	80.50
buscando trabajo por 1ra. vez	5	0.07	80.57
inactivo	1,350	19.43	100.00
Total	6,949	100.00	

Estadísticas descriptivas

- También se puede mezclar el comando **tab** (tabulate) con **summarize**:

```
. tab esc09, summarize(ing_entrev_mensual) means
```

Años de escolaridad EPS09	Summary of ing_entrev_ mensual Mean
0	101186.33
1	75201.093
2	102586.96
3	115778.54
4	129247.77
5	114401.1
6	128952.18
7	167105.26
8	145126.82
9	178890.03
10	231746.11
11	247240.56
12	217041.55
13	371306.02
14	370284.47
15	370182.99
16	476301.6
17	692630.93
18	960761.91
19	941410.25
20	800855.75
21	1840060
Total	230243.43

```
. tab esc09 sexo09, summarize(ing_entrev_mensual) means
```

Años de escolarida d EPS09	Means of ing_entrev_mensual			
	a8. sexo	1	2	Total
0	121164.56	80915.74	101186.33	
1	93559.686	61580.201	75201.093	
2	137300.52	73777.061	102586.96	
3	152217.48	80317.823	115778.54	
4	170856.69	83285.912	129247.77	
5	169263.06	69624.068	114401.1	
6	177323.43	83748.606	128952.18	
7	254680.15	76623.322	167105.26	
8	198081.4	89960.377	145126.82	
9	283087.82	98804.119	178890.03	
10	293087.4	151329.17	231746.11	
11	325590.74	178684.15	247240.56	
12	307354.91	131798.54	217041.55	
13	379531.42	364275.76	371306.02	
14	467029.04	298529.28	370284.47	
15	452353.79	290917.21	370182.99	
16	581758.76	370844.44	476301.6	
17	891904.54	507755.71	692630.93	
18	1192285.7	845000.01	960761.91	
19	1210305.6	710928.57	941410.25	
20	942647.62	668516.66	800855.75	
21	1905647.4	1769007	1840060	
Total	305611.68	158316.65	230243.43	

Estadísticas descriptivas

- Y también se pueden hacer tablas, no con la distribución de frecuencias sino con estadísticas descriptivas, esto se hace con el comando **tabstat**:

```
. tabstat esc09, stats(mean) by(b2_09)  
Summary for variables: esc09  
by categories of: b2_09 (Estatus laboral Abril 2009)  
  
b2_09 | mean  
-----|-----  
trabajando | 10.9357  
cesante | 9.5672  
buscando trabajo | 10.83333  
inactivo | 7.504998  
  
Total | 9.696502
```

Variable(s) sobre la cual(es) se quiere sacar las estadísticas descriptivas

Variable sobre la cual quiero la tabla, una sola.

Listado de estadísticas descriptivas, puede ser más de una.

Estadísticas descriptivas

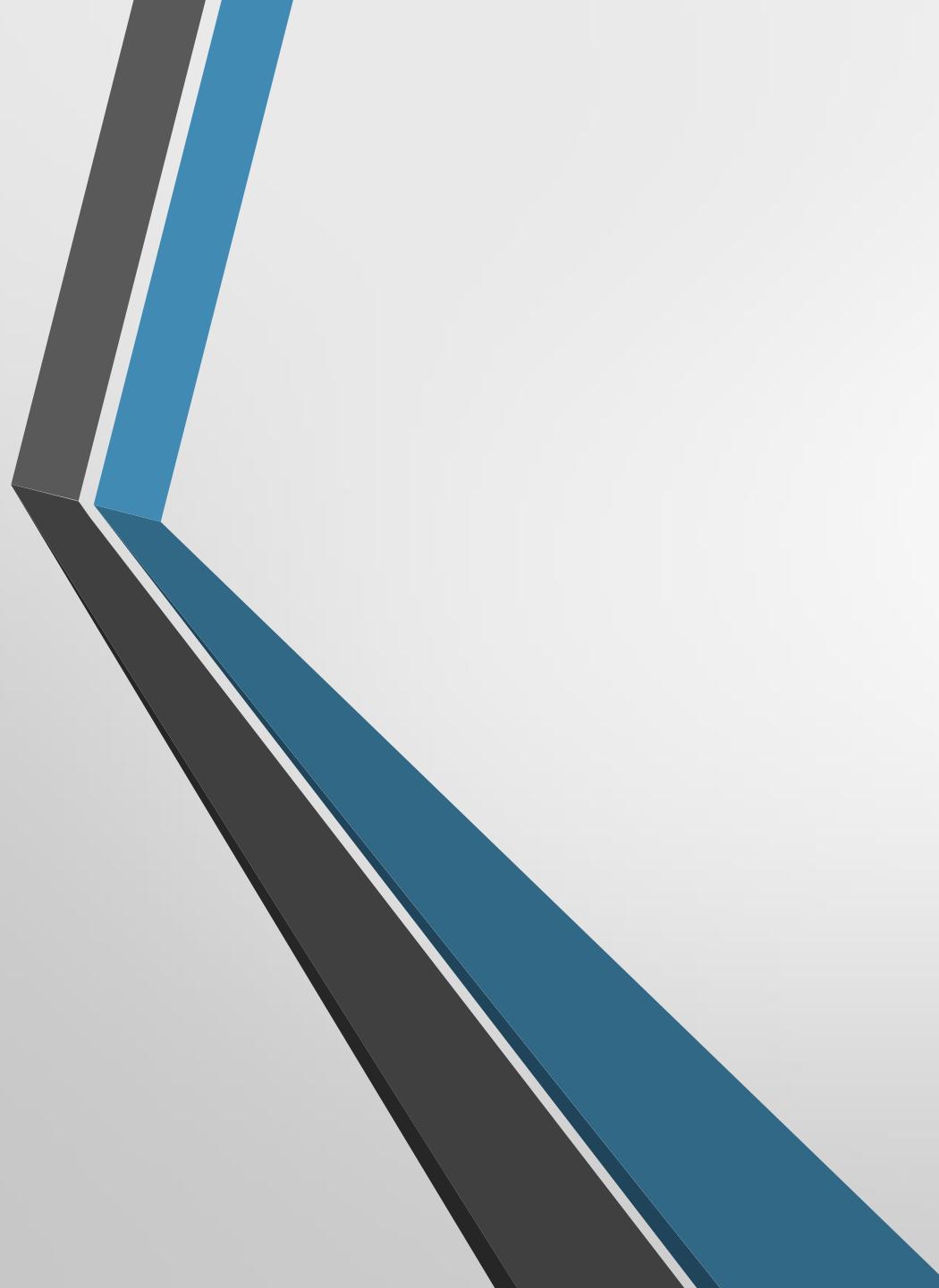
Summary statistics: mean, p50 by categories of: b2_09 (Estatus laboral Abril 2009)			
b2_09	esc09	edad09	
trabajando	10.52157 12	45.62885 45	
cesante	9.107955 10	47.56554 46	
buscando trabajo	9.8 12	44.4 49	
inactivo	6.652924 6	66.59037 69	
Total	9.664443 12	49.84904 48	

También se puede utilizar filtro (*if*), el que siempre va antes de la coma de opciones

Estadísticas descriptivas

- Este es el listado de estadísticas que se puede pedir:

statname	Definition
<u>mean</u>	mean
<u>count</u>	count of nonmissing observations
<u>n</u>	same as count
<u>sum</u>	sum
<u>max</u>	maximum
<u>min</u>	minimum
<u>range</u>	range = max - min
<u>sd</u>	standard deviation
<u>variance</u>	variance
<u>cv</u>	coefficient of variation (sd/mean)
<u>semean</u>	standard error of mean (sd/sqrt(n))
<u>skewness</u>	skewness
<u>kurtosis</u>	kurtosis
<u>p1</u>	1st percentile
<u>p5</u>	5th percentile
<u>p10</u>	10th percentile
<u>p25</u>	25th percentile
<u>median</u>	median (same as p50)
<u>p50</u>	50th percentile (same as median)
<u>p75</u>	75th percentile
<u>p90</u>	90th percentile
<u>p95</u>	95th percentile
<u>p99</u>	99th percentile
<u>iqr</u>	interquartile range = p75 - p25
<u>q</u>	equivalent to specifying p25 p50 p75



Taller Stata

Clase 3

Javiera Vásquez

Estadísticas descriptivas

- El comando **summarize**, el que se puede abbreviar simplemente como **sum**, entrega estadísticas descriptivas básicas como:
 - Número de observaciones
 - Promedio (mean)
 - Desviación estándar (Std. Dev.)
 - Mínimo (Min)
 - Máximo (Max)

```
. sum edad09 esc09 b13_09
```

Variable	Obs	Mean	Std. Dev.	Min	Max
edad09	14243	49.93239	15.22389	19	108
esc09	14124	9.696616	4.324177	0	21
b13_09	8309	94.71645	211.105	2	999

Estadísticas descriptivas

- Podemos utilizar el comando **summarize** con la opción **by varlist**:

```
. by sexo09: sum esc09  
not sorted  
r(5);  
  
. bys sexo09: sum esc09  
  
-> sexo09 = 1  
  
Variable | Obs Mean Std. Dev. Min Max  
esc09 | 6897 9.664492 4.307995 0 21  
  
-> sexo09 = 2  
  
Variable | Obs Mean Std. Dev. Min Max  
esc09 | 7227 9.727273 4.33964 0 21
```

Estadísticas descriptivas

- También se puede utilizar el comando **summarize** filtros (**if**):

```
. sum esc09 if sexo09==1
```

Variable	Obs	Mean	Std. Dev.	Min	Max
esc09	6897	9.664492	4.307995	0	21

- También se puede ocupar el comando **summarize** con la opción **, detail**:

```
. sum esc09, detail
```

Años de escolaridad EPS09

	Percentiles	Smallest		
1%	0	0		
5%	2	0		
10%	3	0	Obs	14124
25%	6	0	Sum of Wgt.	14124
50%	12		Mean	9.696616
		Largest	Std. Dev.	4.324177
75%	12	21		
90%	15	21	Variance	18.69851
95%	17	21	Skewness	-.3412399
99%	17	21	Kurtosis	2.465894

Estadísticas descriptivas

- El comando **tabulate** (o **tab**) permite hacer tablas de distribución de frecuencias, las que muestran en número de observaciones en cada una de las categorías de la variable, así como el porcentaje y el porcentaje acumulado, estas se conocen como tablas de una entrada:

Estatus laboral Abril 2009	Freq.	Percent	Cum.
trabajando	8,309	58.35	58.35
cesante	1,274	8.95	67.29
buscando trabajo por 1ra. vez	13	0.09	67.38
inactivo	4,645	32.62	100.00
Total	14,241	100.00	

N – frecuencia absoluta % - frecuencia relativa

Estadísticas descriptivas

- Podemos incorporar el *missing value* como una categoría:

Estatus laboral Abril 2009	Freq.	Percent	Cum.
trabajando	8,309	58.34	58.34
cesante	1,274	8.94	67.28
buscando trabajo por 1ra. vez	13	0.09	67.37
inactivo	4,645	32.61	99.99
.	2	0.01	100.00
Total	14,243	100.00	

- O podemos hacer una tabla donde se comparan ambas, considerando o no considerando el *missing value* como una categoría, para esto se usa el comando **fre**, sino está instalado en su computador, recuerde ocupar el comando **ssc install** para instalarlo.

Estadísticas descriptivas

```
. ssc install fre  
checking fre consistency and verifying not already installed...  
installing into c:\ado\plus\...  
installation complete.
```

```
. fre b2_09
```

b2_09 — Estatus laboral Abril 2009

		Freq.	Percent	Valid	Cum.
Valid	1 trabajando	8309	58.34	58.35	58.35
	2 cesante	1274	8.94	8.95	67.29
	3 buscando trabajo por 1ra. vez	13	0.09	0.09	67.38
	4 inactivo	4645	32.61	32.62	100.00
	Total	14241	99.99	100.00	
Missing	.	2	0.01		
	Total	14243	100.00		

Estadísticas descriptivas

- Para hacer una tabla de dos entradas, se ponen las dos variables:

. tab b2_09 sexo09			
Estatus laboral Abril 2009	a8. sexo		
	1	2	Total
trabajando	5,060	3,249	8,309
cesante	534	740	1,274
buscando trabajo por	5	8	13
inactivo	1,350	3,295	4,645
Total	6,949	7,292	14,241

- Esta sólo nos entrega la frecuencia absoluta, es decir el número de observaciones

Estadísticas descriptivas

- Si queremos que nos muestre los porcentajes, tenemos varias alternativas:

- Que sume el 100% en las columnas:

. tab b2_09 sexo09, col nofreq

Estatus laboral Abril 2009	a8. sexo		Total
	1	2	
trabajando	72.82	44.56	58.35
cesante	7.68	10.15	8.95
buscando trabajo por	0.07	0.11	0.09
inactivo	19.43	45.19	32.62
Total	100.00	100.00	100.00

Es para que no muestre el número de observaciones

- Que sume el 100% en las filas:

. tab b2_09 sexo09, row nofreq

Estatus laboral Abril 2009	a8. sexo		Total
	1	2	
trabajando	60.90	39.10	100.00
cesante	41.92	58.08	100.00
buscando trabajo por	38.46	61.54	100.00
inactivo	29.06	70.94	100.00
Total	48.80	51.20	100.00

Estadísticas descriptivas

- Que sume el 100% en el total:

```
. tab b2_09 sexo09, cell nofreq
```

Estatus laboral Abril 2009	a8. sexo		Total
	1	2	
trabajando	35.53	22.81	58.35
cesante	3.75	5.20	8.95
buscando trabajo por inactivo	0.04	0.06	0.09
	9.48	23.14	32.62
Total	48.80	51.20	100.00

Estadísticas descriptivas

- También podemos ocupar **by** e **if**:

```
. bys sexo09: tab b2_09
```

-> sexo09 = 1

Estatus laboral Abril 2009	Freq.	Percent	Cum.
trabajando	5,060	72.82	72.82
cesante	534	7.68	80.50
buscando trabajo por 1ra. vez	5	0.07	80.57
inactivo	1,350	19.43	100.00
Total	6,949	100.00	

-> sexo09 = 2

Estatus laboral Abril 2009	Freq.	Percent	Cum.
trabajando	3,249	44.56	44.56
cesante	740	10.15	54.70
buscando trabajo por 1ra. vez	8	0.11	54.81
inactivo	3,295	45.19	100.00
Total	7,292	100.00	

```
. tab b2_09 if sexo09==1
```

Estatus laboral Abril 2009	Freq.	Percent	Cum.
trabajando	5,060	72.82	72.82
cesante	534	7.68	80.50
buscando trabajo por 1ra. vez	5	0.07	80.57
inactivo	1,350	19.43	100.00
Total	6,949	100.00	

Estadísticas descriptivas

- También se puede mezclar el comando **tab** (tabulate) con **summarize**:

```
. tab esc09, summarize(ing_entrev_mensual) means
```

Años de escolaridad EPS09	Summary of ing_entrev_ mensual Mean
0	101186.33
1	75201.093
2	102586.96
3	115778.54
4	129247.77
5	114401.1
6	128952.18
7	167105.26
8	145126.82
9	178890.03
10	231746.11
11	247240.56
12	217041.55
13	371306.02
14	370284.47
15	370182.99
16	476301.6
17	692630.93
18	960761.91
19	941410.25
20	800855.75
21	1840060
Total	230243.43

```
. tab esc09 sexo09, summarize(ing_entrev_mensual) means
```

Años de escolarida d EPS09	Means of ing_entrev_mensual			
	a8. sexo	1	2	Total
0	121164.56	80915.74	101186.33	
1	93559.686	61580.201	75201.093	
2	137300.52	73777.061	102586.96	
3	152217.48	80317.823	115778.54	
4	170856.69	83285.912	129247.77	
5	169263.06	69624.068	114401.1	
6	177323.43	83748.606	128952.18	
7	254680.15	76623.322	167105.26	
8	198081.4	89960.377	145126.82	
9	283087.82	98804.119	178890.03	
10	293087.4	151329.17	231746.11	
11	325590.74	178684.15	247240.56	
12	307354.91	131798.54	217041.55	
13	379531.42	364275.76	371306.02	
14	467029.04	298529.28	370284.47	
15	452353.79	290917.21	370182.99	
16	581758.76	370844.44	476301.6	
17	891904.54	507755.71	692630.93	
18	1192285.7	845000.01	960761.91	
19	1210305.6	710928.57	941410.25	
20	942647.62	668516.66	800855.75	
21	1905647.4	1769007	1840060	
Total	305611.68	158316.65	230243.43	

Estadísticas descriptivas

- Y también se pueden hacer tablas, no con la distribución de frecuencias sino con estadísticas descriptivas, esto se hace con el comando **tabstat**:

```
. tabstat esc09, stats(mean) by(b2_09)  
Summary for variables: esc09  
by categories of: b2_09 (Estatus laboral Abril 2009)  
  
b2_09 | mean  
-----|-----  
trabajando | 10.9357  
cesante | 9.5672  
buscando trabajo | 10.83333  
inactivo | 7.504998  
  
Total | 9.696502
```

Variable(s) sobre la cual(es) se quiere sacar las estadísticas descriptivas

Variable sobre la cual quiero la tabla, una sola.

Listado de estadísticas descriptivas, puede ser más de una.

Estadísticas descriptivas

Summary statistics: mean, p50 by categories of: b2_09 (Estatus laboral Abril 2009)			
b2_09	esc09	edad09	
trabajando	10.52157 12	45.62885 45	
cesante	9.107955 10	47.56554 46	
buscando trabajo	9.8 12	44.4 49	
inactivo	6.652924 6	66.59037 69	
Total	9.664443 12	49.84904 48	

También se puede utilizar filtro (*if*), el que siempre va antes de la coma de opciones

Estadísticas descriptivas

- Este es el listado de estadísticas que se puede pedir:

statname	Definition
<u>mean</u>	mean
<u>count</u>	count of nonmissing observations
<u>n</u>	same as count
<u>sum</u>	sum
<u>max</u>	maximum
<u>min</u>	minimum
<u>range</u>	range = max - min
<u>sd</u>	standard deviation
<u>variance</u>	variance
<u>cv</u>	coefficient of variation (sd/mean)
<u>semean</u>	standard error of mean (sd/sqrt(n))
<u>skewness</u>	skewness
<u>kurtosis</u>	kurtosis
<u>p1</u>	1st percentile
<u>p5</u>	5th percentile
<u>p10</u>	10th percentile
<u>p25</u>	25th percentile
<u>median</u>	median (same as p50)
<u>p50</u>	50th percentile (same as median)
<u>p75</u>	75th percentile
<u>p90</u>	90th percentile
<u>p95</u>	95th percentile
<u>p99</u>	99th percentile
<u>iqr</u>	interquartile range = p75 - p25
<u>q</u>	equivalent to specifying p25 p50 p75

Estadísticas descriptivas

- Se puede cambiar el formato con que se quiere que aparezcan los números

```
. tabstat esc09 edad09 if sexo09==1, stats(mean p50) by(b2_0) format(%3.1f)
```

Summary statistics: mean, p50
by categories of: b2_09 (Estatus laboral Abril 2009)

b2_09	esc09	edad09
trabajando	10.5	45.6
	12.0	45.0
cesante	9.1	47.6
	10.0	46.0
buscando trabajo	9.8	44.4
	12.0	49.0
inactivo	6.7	66.6
	6.0	69.0
Total	9.7	49.8
	12.0	48.0

Ejercicio: práctica para taller 1

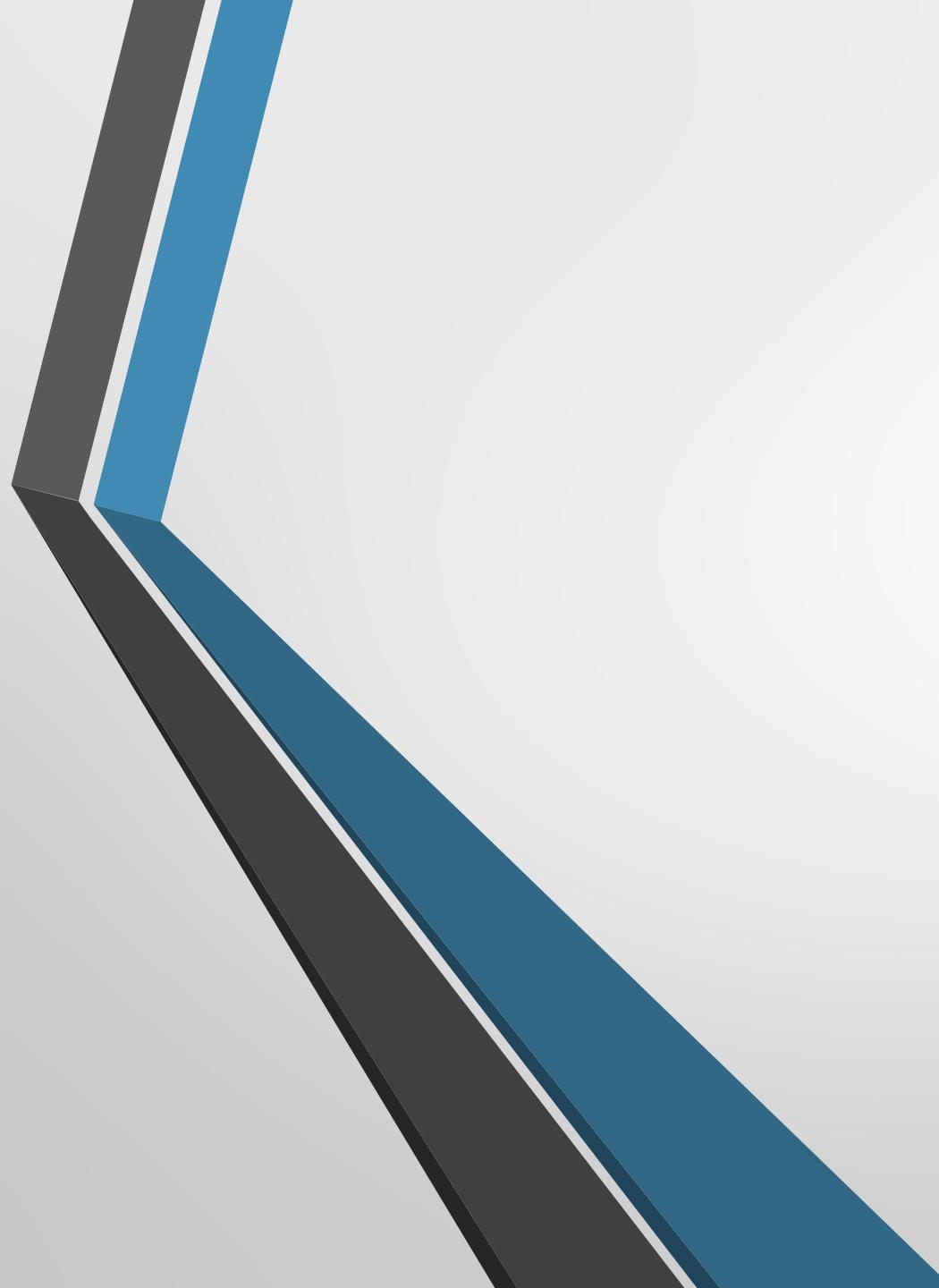
- Para el siguiente ejercicio haga un do-file
- Comience con los comandos recomendados en clases para limpiar todo y ubicar a Stata en la ruta (carpeta) donde esté trabajando
- Abra la base de datos T1.dta, corresponde a la base de datos de la ENE para el trimestre Enero-Febrero-Marzo de 2018.
- ¿La variable edad es numérica o string?
- ¿La variable parentesco es numérica o string?
- ¿Cuál es el código (número) de “hijo(a)” en la variable parentesco?

Ejercicio: práctica para taller 1

- Cuente el número de observaciones totales en la base de datos
- Cuente el número de observaciones en la región metropolitana
- Cuente el número de observaciones que corresponden a hombres entre 25 y 25 años de edad que residen en la región de Antofagasta
- Cuantifique la cantidad de missing values en las siguientes variables: edad, sexo, a1 y efectivas.
- Calcule el promedio de horas efectivas trabajadas
- Calcule el promedio de edad
- Calcule el promedio de horas efectivas trabajadas para hombres y mujeres
- Calcule el promedio y mediana de horas efectivas trabajadas según región

Ejercicio: práctica para taller 1

- Haga una distribución de frecuencias absolutas y relativa de la categoría ocupacional, ¿Qué porcentaje de la muestra trabaja como asalariado del sector privado?.
- Haga una distribución de frecuencias relativa de la categoría ocupacional separando entre hombres y mujeres.
- Haga una tabla de distribución de frecuencias relativas que muestre para cada categoría ocupacional, que porcentaje es hombre y que porcentaje es mujer.



Taller Stata

Clase 4

Javiera Vásquez

Modificación de base de datos

- Para lograr que una base de datos sea más amigable y sea entendida por cualquier usuario, es recomendable incorporar etiquetas (*labels*) tanto a las variables como a sus categorías de respuesta.
- Por ejemplo, en la base de datos las variables `ing_entrev_mensual`, `numper` e `ytotal` no están etiquetadas como variable, es decir, no tienen descripción:

```
. d ing_entrev_mensual numper ytotal

      storage  display    value
variable name   type   format   label   variable label
_____________________________________
ing_entrev_me~1 float  %9.0g
numper           float  %9.0g
ytotal           float  %9.0g
```

Modificación de base de datos

- Para agregar la descripción de la variable se utiliza el comando **label variable**:

```
. label variable ing_entrev_mensual "Ingreso mensual del entrevistado"  
  
. label variable numper "Número de personas en el hogar"  
  
. label variable ytotal "Ingreso total del hogar"
```

```
. d ing_entrev_mensual numper ytotal  
  
      storage  display    value  
variable name   type   format   label   variable label  
  
-----  
ing_entrev_me~l float  %9.0g          Ingreso mensual del entrevistado  
numper           float  %9.0g          Número de personas en el hogar  
ytotal           float  %9.0g          Ingreso total del hogar
```

Modificación de base de datos

- Para etiquetar las categorías de respuesta de una variable, necesitamos de una variable secundaria llamada `value_label`, la que contiene el diccionario o asociación entre número y palabra o etiqueta.
- Habíamos visto que en el resultado de ejecutar el comando `describe` aparece si la variable tiene asociada un `value_label` (diccionario) y con el comando `label list` podemos ver ese diccionario.
- Ahora vamos a definir el diccionario y luego asociarlo a la variable correspondiente para de esta forma etiquetar las categorías de una variable
- Por ejemplo, en la base de datos la variable `sexo09` no está etiquetada, pero sabemos que el código 1 representa a los hombres y el código 2 representa a las mujeres.

Modificación de base de datos

- Para etiquetar las categorías de la variable `sex09`, primero debemos definir el `value label` (diccionario):

```
label define sexo 1 "Hombre" 2 "Mujer"
```

- Este diccionario puede tener cualquier nombre, incluso el mismo nombre de la variable, no genera conflicto ya que no es una variable de la base de datos.
- Luego debemos asociar la variable de la base de datos (`sexo09`) con su respectivo `value label` (diccionario):

```
label values sexo09 sexo
```

Modificación de base de datos

```
. d sexo09  
  
      storage  display    value  
variable name   type   format   label     variable label  
  
sexo09        byte   %9.0g    sexo      a8. sexo  
  
. label list sexo  
sexo:  
      1 Hombre  
      2 Mujer
```

Ahora aparece en azul

Data Editor (Edit) - [exteps09.dta]

File Edit View Data Tools

folio[1] 716324

	folio	edad09	esc09	sexo09	b2_09
1	716324	54	12	Hombre	trabajando
2	2323948	42	8	Hombre	trabajando
3	1337701	39	13	Hombre	trabajando
4	1278025	88	3	Hombre	inactivo
5	911256	29	12	Hombre	trabajando
6	920882	40	12	Hombre	trabajando
7	2148498	47	.	Hombre	trabajando
8	610563	53	12	Hombre	trabajando
9	1180113	83	1	Hombre	inactivo
10	1243640	29	12	Hombre	trabajando
11	1101498	46	12	Hombre	trabajando

Modificación de base de datos

- Cuando una variable que debería ser numérica está en formato *string* (no numérico) no se pueden obtener estadísticas de ella. Mediante el siguiente comando podemos transformar una variable no numérica en numérica:

```
generate nueva_variable=real(variable)
```

- Otras veces tenemos variables *string* porque son texto, y queremos codificar estos textos, para eso podemos utilizar el siguiente comando:

```
encode variable, generate(nueva_variable)
```

Modificación de base de datos

- Veamos el siguiente ejemplo, primero abramos la base de datos ejstring.dta:

```
use ejstring.dta, clear
```

```
. d

Contains data from ejstring.dta
    obs:                 20
    vars:                  3
    size:                300
                                         2 May 2011 12:06

              storage  display      value
variable name   type    format     label  variable label
            pais    str9    %9s
            var1    str4    %9s          var 1
            var2     int    %8.0g

Sorted by:
```

Son tres variables, dos no numéricas y 1 numérica

Modificación de base de datos

The screenshot shows the Stata Data Editor window titled "Data Editor (Edit) - [ejstring.dta]". The menu bar includes File, Edit, View, Data, Tools. Below the menu is a toolbar with icons for file operations. The current view is "var4[22]". The data table has three columns: "pais", "var1", and "var2". The data rows are:

	pais	var1	var2
1	Argentina	11.3	1500
2	Argentina		1350
3	Argentina	12.5	1400
4	Argentina	11.5	2000
5	Argentina	10	2100
6	Bolivia	20	3000
7	Bolivia	20.3	4000

país es no numérica porque es un texto, esta bien, pero la podríamos querer codificar.

var1 es no numérica pero es un número, no está bien, la debemos transformar para que STATA la lea como número

. sum var1

Variable	Obs	Mean	Std. Dev.	Min	Max
var1	0				

Si tratamos de sacar estadísticas de var1, no podremos, es como si no tuviera observaciones

Modificación de base de datos

- Para transformar la variable var1 a formato numérico:

```
generate var1num=real(var1)
```

- Para codificar la variable país:

```
encode pais, generate(pais_cod)
```

Modificación de base de datos

```
. edit pais pais_cod var1 var1num
```

	pais	pais_cod	var1	var1num
1	Argentina	Argentina	11.3	11.3
2	Argentina	Argentina		.
3	Argentina	Argentina	12.5	12.5
4	Argentina	Argentina	11.5	11.5
5	Argentina	Argentina	10	10
6	Bolivia	Bolivia	20	20
7	Bolivia	Bolivia	20.3	20.3
8	Bolivia	Bolivia	22.5	22.5
9	Bolivia	Bolivia	24.1	24.1
10	Bolivia	Bolivia	25.9	25.9

```
. d
```

Contains data from ejstring.dta

obs:

20

vars:

5

size:

460

2 May 2011 12:06

variable name	storage type	display format	value label	variable label
pais	str9	%9s		
var1	str4	%9s		var 1
var2	int	%8.0g		
var1num	float	%9.0g		
pais_cod	long	%9.0g	pais_cod	

Sorted by:

Note: dataset has changed since last saved

```
. label list pais_cod  
pais_cod:
```

- 1 Argentina
- 2 Bolivia
- 3 Chile
- 4 Ecuador

Modificación de base de datos

- El comando **recode** se usa para cambiar los valores de la variable
- Volvamos a la base exteps09.dta.
- Veamos la variable b11_09, que indica para las personas ocupadas si entregan boleta de honorarios:

The image shows two Stata tabulation outputs side-by-side. The left output is for the original dataset and the right is for the modified dataset after applying a recode command.

Left Output (Original Data):

```
. tab b11_09
```

Entrega boleta de honorarios (Abril 2009)	Freq.	Percent	Cum.
sí	697	8.42	8.42
no	7,519	90.80	99.22
no responde	52	0.63	99.84
no sabe	13	0.16	100.00
Total	8,281	100.00	

Right Output (Modified Data):

```
. tab b11_09, nolabel
```

Entrega boleta de honorarios (Abril 2009)	Freq.	Percent	Cum.
1	697	8.42	8.42
2	7,519	90.80	99.22
8	52	0.63	99.84
9	13	0.16	100.00
Total	8,281	100.00	

A red oval highlights the category "no responde" in the first table, and a red arrow points from this category to the value "8" in the second table, indicating that the value has been changed to 8.

Modificación de base de datos

- Queremos pasar las respuesta “no responde” y “no sabe” a *missing value*:

```
. recode b11_09 (8=.) (9=.)
(b11_09: 65 changes made)
```

Entrega boleta de honorarios (Abril 2009)		Freq.	Percent	Cum.
sí		697	8.48	8.48
no		7,519	91.52	100.00
Total		8,216	100.00	

Modificación de base de datos

- Otra alternativa es recodificar como *missing value*, pero distinto al del sistema, por ejemplo:

```
. tab b9b_09
```

Relación Contractual (Abril 2009)	Freq.	Percent	Cum.
plazo indefinido	4,248	79.24	79.24
plazo fijo	572	10.67	89.91
por obra, faena o servicio	462	8.62	98.53
servicios transitorios	36	0.67	99.20
no responde	24	0.45	99.65
no sabe	19	0.35	100.00
Total	5,361	100.00	

```
. tab b9b_09, nolabel
```

Relación Contractual (Abril 2009)	Freq.	Percent	Cum.
1	4,248	79.24	79.24
2	572	10.67	89.91
3	462	8.62	98.53
5	36	0.67	99.20
8	24	0.45	99.65
9	19	0.35	100.00
Total	5,361	100.00	

```
. recode b9b_09 (8=.a) (9=.b)  
(b9b_09: 43 changes made)
```

Modificación de base de datos

- Ahora los valores “.a” y “.b” son reconocidos por STATA como *missing value*, pero no del sistema, distinto de “.”:

```
. tab b9b_09
```

Relación Contractual (Abril 2009)	Freq.	Percent	Cum.
plazo indefinido	4,248	79.88	79.88
plazo fijo	572	10.76	90.64
por obra, faena o servicio	462	8.69	99.32
servicios transitorios	36	0.68	100.00
Total	5,318	100.00	

```
. tab b9b_09, miss
```

Relación Contractual (Abril 2009)	Freq.	Percent	Cum.
plazo indefinido	4,248	29.83	29.83
plazo fijo	572	4.02	33.84
por obra, faena o servicio	462	3.24	37.08
servicios transitorios	36	0.25	37.34
.	8,882	62.36	99.70
.a	24	0.17	99.87
.b	19	0.13	100.00
Total	14,243	100.00	

```
. misstable sum b9b_09
```

Variable	Obs=.	Obs>.	Obs<.	Obs<.		
				Unique values	Min	Max
b9b_09	8,882	43	5,318	4	1	5

Modificación de base de datos

- Se podrían recodificar para todas las variables de una sola vez los códigos 9 por missing value, esto se hace con el siguiente comando:

```
recode _all (9=.a)
```

- También se puede usar para recodificar otros números, por ejemplo:

```
. recode b18_09 (1/6=1)  
(b18_09: 190 changes made)
```

Se encuentra cotizando (Abril 2009)	Freq.	Percent	Cum.
sí, afp (administradora de fondos de pe	5,777	69.53	69.53
sí, inp [servicio de seguro social, caj	133	1.60	71.13
sí, capredena (caja de previsión de la	6	0.07	71.20
sí, dipreca (dirección de previsión de	10	0.12	71.32
sí, otra caja	20	0.24	71.56
sí, no sabe donde cotizó	21	0.25	71.81
no cotizó	2,250	27.08	98.89
no responde	25	0.30	99.19
no sabe	67	0.81	100.00
Total	8,309	100.00	

Se encuentra cotizando (Abril 2009)	Freq.	Percent	Cum.
sí, afp (administradora de fondos de pe	5,967	71.81	71.81
no cotizó	2,250	27.08	98.89
no responde	25	0.30	99.19
no sabe	67	0.81	100.00
Total	8,309	100.00	

```
. label define b18 1 "si cotizó", replace
```

Modificación de base de datos

- El comando **recode** no puede ser utilizado con variables no numéricas (string), en estos casos debemos utilizar el comando:

```
replace
```

- Este también se puede utilizar con variables numéricas, al igual que recode.
- El comando **rename** se utiliza para cambiar el nombre de la variable:

```
rename esc09 esc
```

- También se puede cambiar el comando **rename** para cambiar nombres de variables de minúscula a mayúscula o viceversa:

```
. rename b2_09, upper  
.  
. rename B2_09, lower
```

```
. rename _all, upper  
.  
. rename _all, lower
```

Modificación de base de datos

- El comando **generate** o simplemente **g** sirve para generar nuevas variables utilizando los siguientes operadores:

Arithmetic	Logical	Relational (numeric and string)
+	addition	> greater than
-	subtraction	< less than
*	multiplication	>= > or equal
/	division	<= < or equal
^	power	== equal
-	negation	!= not equal
+	string concatenation	~= not equal

Modificación de base de datos

```
. g neduc=1 if esc<8  
(10050 missing values generated)  
  
. replace neduc=2 if esc==8  
(1662 real changes made)  
  
. replace neduc=3 if esc>8 & esc<12  
(773 real changes made)  
  
. replace neduc=4 if esc==12  
(5159 real changes made)  
  
. replace neduc=5 if esc>12  
(2456 real changes made)  
  
. replace neduc=. if esc==.  
(119 real changes made, 119 to missing)
```

o de manera
alternativa

```
. recode esc09 (1/7=1) (8=2) (9/11=3) (12=4) (else=5), generate(neduc)  
(14081 differences between esc09 and neduc)  
  
. replace neduc=. if esc09==.  
(119 real changes made, 119 to missing)
```

```
. label define neduc 1 "Básica Incompleta" 2 "Básica Completa" 3 "Media Incompleta"  
  
. label define neduc 4 "Media Completa" 5 "Superior", add  
  
. label values neduc neduc  
  
. label variable neduc "Nivel educacional"  
  
. order folio edad09 esc09 neduc
```

Modificación de base de datos

- También existen funciones que pueden ser utilizadas para generar nuevas variables con el comando **generate**, como por ejemplo: `ln(x)`, `sqrt(x)`, `int(x)`, `round(x)`, `sign(x)`, entre otras. Para ver todas las funciones ir a `help functions`.
- Edad al cuadrado:

```
g edad2=edad09*edad09
```

- Logaritmo del ingreso:

```
g ly=ln(ytotal)
```

Modificación de base de datos

- Otro comando para generar variables es **egen**, la diferencia con el comando **generate** es que son funciones ya programadas.
- Una variable estandarizada, consiste en tener media cero y desviación estándar 1, cualquier variable puede ser fácilmente estandarizada con el siguiente comando:

```
. egen ingz=std(ytotal)

. sum ytotal ingz
```

Variable	Obs	Mean	Std. Dev.	Min	Max
ytotal	14243	4889592	7433665	0	2.46e+08
ingz	14243	-5.79e-11	1	-.6577633	32.37644

- Existen muchas otras funciones de este comando las que puede encontrar en `help egen`.

Modificación de base de datos

- El comando **keep** se usa para mantener datos o variables, y el comando **drop** para eliminar observaciones o variables.
- Por ejemplo, si sólo quiero trabajar con los hombres de la base de datos:

```
keep if sexo09==1
```

- O solo quiero trabajar con las personas de 12 años de escolaridad o menos:

```
keep if esc09<=12
```



Taller Stata

Clase 5

Javiera Vásquez

Modificación de base de datos

- El comando **generate** o simplemente **g** sirve para generar nuevas variables utilizando los siguientes operadores:

Arithmetic	Logical	Relational (numeric and string)
+	addition	> greater than
-	subtraction	< less than
*	multiplication	>= > or equal
/	division	<= < or equal
^	power	== equal
-	negation	!= not equal
+	string concatenation	~= not equal

Modificación de base de datos

```
. g neduc=1 if esc<8  
(10050 missing values generated)  
  
. replace neduc=2 if esc==8  
(1662 real changes made)  
  
. replace neduc=3 if esc>8 & esc<12  
(773 real changes made)  
  
. replace neduc=4 if esc==12  
(5159 real changes made)  
  
. replace neduc=5 if esc>12  
(2456 real changes made)  
  
. replace neduc=. if esc==.  
(119 real changes made, 119 to missing)
```

o de manera
alternativa

```
. recode esc09 (1/7=1) (8=2) (9/11=3) (12=4) (else=5), generate(neduc)  
(14081 differences between esc09 and neduc)  
  
. replace neduc=. if esc09==.  
(119 real changes made, 119 to missing)
```

```
. label define neduc 1 "Básica Incompleta" 2 "Básica Completa" 3 "Media Incompleta"  
  
. label define neduc 4 "Media Completa" 5 "Superior", add  
  
. label values neduc neduc  
  
. label variable neduc "Nivel educacional"  
  
. order folio edad09 esc09 neduc
```

Modificación de base de datos

- También existen funciones que pueden ser utilizadas para generar nuevas variables con el comando **generate**, como por ejemplo: `ln(x)`, `sqrt(x)`, `int(x)`, `round(x)`, `sign(x)`, entre otras. Para ver todas las funciones ir a `help functions`.
- Edad al cuadrado:

```
g edad2=edad09*edad09
```

- Logaritmo del ingreso:

```
g ly=ln(ytotal)
```

Modificación de base de datos

- Otro comando para generar variables es **egen**, la diferencia con el comando **generate** es que son funciones ya programadas.
- Una variable estandarizada, consiste en tener media cero y desviación estándar 1, cualquier variable puede ser fácilmente estandarizada con el siguiente comando:

```
. egen ingz=std(ytotal)

. sum ytotal ingz
```

Variable	Obs	Mean	Std. Dev.	Min	Max
ytotal	14243	4889592	7433665	0	2.46e+08
ingz	14243	-5.79e-11	1	-.6577633	32.37644

- Existen muchas otras funciones de este comando las que puede encontrar en `help egen`.

Modificación de base de datos

- El comando **keep** se usa para mantener datos o variables, y el comando **drop** para eliminar observaciones o variables.
- Por ejemplo, si sólo quiero trabajar con los hombres de la base de datos:

```
keep if sexo09==1
```

- O solo quiero trabajar con las personas de 12 años de escolaridad o menos:

```
keep if esc09<=12
```

Combinar bases de datos

- El comando **merge** se usa para pegar dos bases de datos de manera horizontal, esto significa a una base de datos le agregamos más variables (o columnas) a partir de otra base de datos.
- El comando **append** se usa para pegar dos bases de datos de manera vertical, esto significa que a una base de datos le agregamos más filas (o observaciones)

merge

- Estas son las distintas opciones del comando **merge**

One-to-one merge on specified key variables

```
merge 1:1 varlist using filename [, options]
```

Many-to-one merge on specified key variables

```
merge m:1 varlist using filename [, options]
```

One-to-many merge on specified key variables

```
merge 1:m varlist using filename [, options]
```

Many-to-many merge on specified key variables

```
merge m:m varlist using filename [, options]
```

One-to-one merge by observation

```
merge 1:1 _n using filename [, options]
```

merge

- Suponga que a la base de datos exteps09.dta, donde los individuos se identifican a través de la variable **folio**, queremos pegar las variables que están en la base de datos exteps06.dta, con información para los mismos individuos identificados a través del mismo **folio**:

```
. cd "G:\FNE\FEN_Taller Stata\Bases de datos"  
G:\FNE\FEN_Taller Stata\Bases de datos  
  
. use "exteps09.dta", clear  
. merge 1:1 folio using exteps06.dta  
(label b18 already defined)  
(label b17 already defined)  
(label b16 already defined)  
(label b15t already defined)  
(label b14 already defined)  
(label b11 already defined)  
(label b10 already defined)  
(label b8 already defined)  
(label b6 already defined)  
(label b4 already defined)  
(label b2 already defined)  
  
Result # of obs.  
-----  
not matched 4,318  
    from master 1,059 (_merge==1) → 1.059 observaciones están sólo en la base master (exteps09)  
    from using 3,259 (_merge==2) → 3.259 observaciones están sólo en la base using (exteps06)  
matched 13,184 (_merge==3) → 13.184 observaciones están en ambas bases de datos
```

merge

- Ahora suponga que tenemos una base de datos de pacientes, donde una de las variables corresponde al identificador del doctor que atendió al paciente, y tenemos otra base de datos de los doctores con este mismo identificador, y queremos pegar ambas bases de datos:

```
. use "pacientes.dta", clear  
  
. merge m:1 docid using "doctor.dta"  
  
          Result          # of obs.  
_____  
not matched                         31  
      from master                      7  (_merge==1)  
      from using                      24  (_merge==2)  
  
matched                            104  (_merge==3)
```

append

- Muchas veces las bases de datos están separadas en varias bases de datos, por ejemplo, cuando tenemos las mismas variables pero para distintos años o periodos de tiempo.
- Si queremos pegar estas bases de datos, ocupamos el comando **append**.
- Las bases deben tener un conjunto de variables en común, las que deben tener exactamente el mismo nombre.
- Las variables con el mismo nombre deben además tener el mismo formato (numérico o no numérico)
- Las variables que no están en alguna de las bases de datos quedan con missing value.

append

```
. use ext_mensualeps06.dta, clear

. d

Contains data from ext_mensualeps06.dta
    obs:       650,602
    vars:          6
    size: 12,361,438
    2 May 2011 12:26

      storage   display     value
variable name   type   format   label     variable label

folio           double %12.0g
b2              byte   %9.0g    b2        b2. en este periodo, ¿en cuál de las siguientes situaciones se encontraba?
oficio         float  %57.0g   oficio
b6              byte   %9.0g    b6        b6. este trabajo era de tipo:
b8              byte   %9.0g    b8        b8. en esta ocupación, ud. trabajaba como:
fecha          float  %9.0g

Sorted by: folio
```

append

```
. use ext_mensualeps09.dta, clear

. d

Contains data from ext_mensualeps09.dta
    obs:      569,120
    vars:          6
    size:  8,536,800
                                2 May 2011 12:27

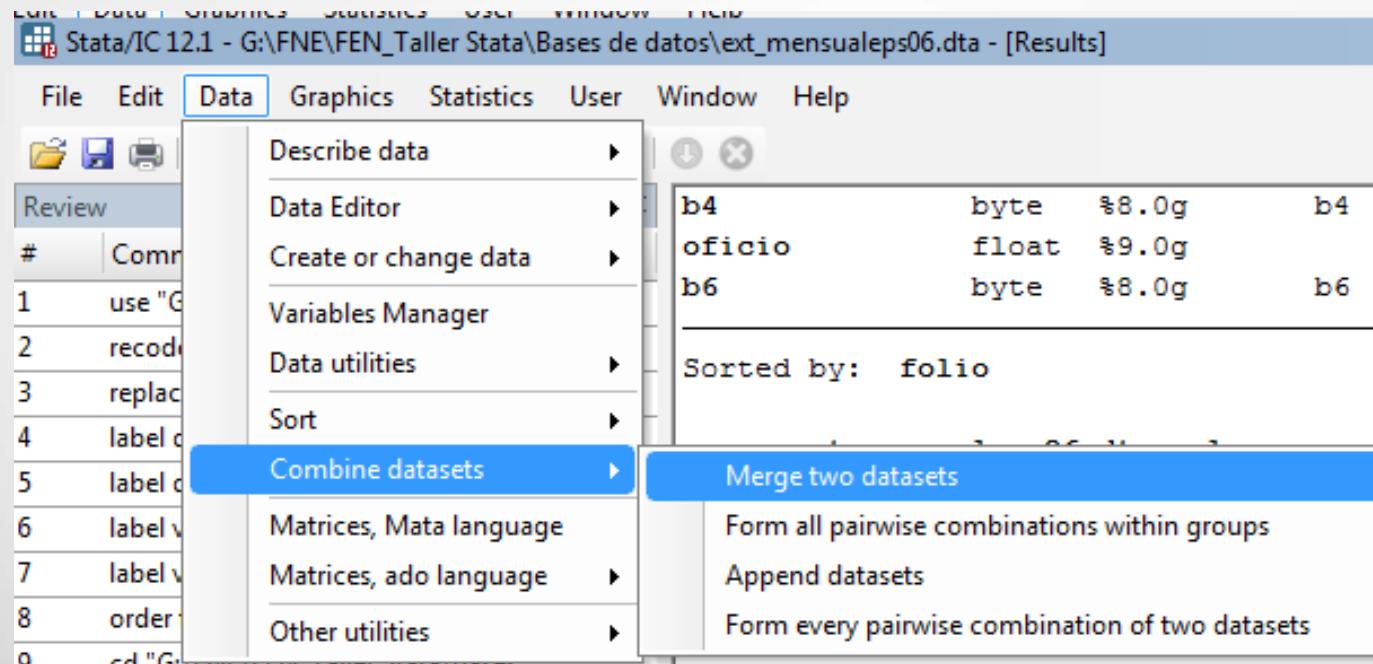
      storage  display    value
variable name   type    format   label   variable label

folio           float   %9.0g
fecha           float   %9.0g
b2              byte   %8.0g   b2       b2. en este periodo ¿en cuál de las siguientes situaciones se encontraba? (1)
b4              byte   %8.0g   b4       b4. ¿en que región trabajaba? si en este periodo realizaba más de un trabajo, re
oficio          float   %9.0g
b6              byte   %8.0g   b6       b6. este trabajo era de tipo:

Sorted by: folio
```

append

```
. use ext_mensualeps06.dta, clear  
. append using ext_mensualeps09.dta
```



Condensar base de datos: collapse

- Muchas bases de datos tienen más de una observación por unidad (individuos, colegios, hospitales, países, etc.).
- Si nos interesa trabajar con sólo una observación por unidad podemos condensar la base de datos a través del comando **collapse**.
- Suponga que tiene una base de datos de hogares, donde se tiene información para cada uno de los miembros del hogar, y usted quiere generar una base de datos condensada con sólo una observación por hogar.

Condensar base de datos: collapse

```
. use "hogar.dta", clear  
  
. list hogar orden if _n<10
```

	hogar	orden
1.	1	1
2.	1	2
3.	1	3
4.	2	1
5.	2	2
<hr/>		
6.	2	3
7.	3	1
8.	3	2
9.	4	1

```
. sum
```

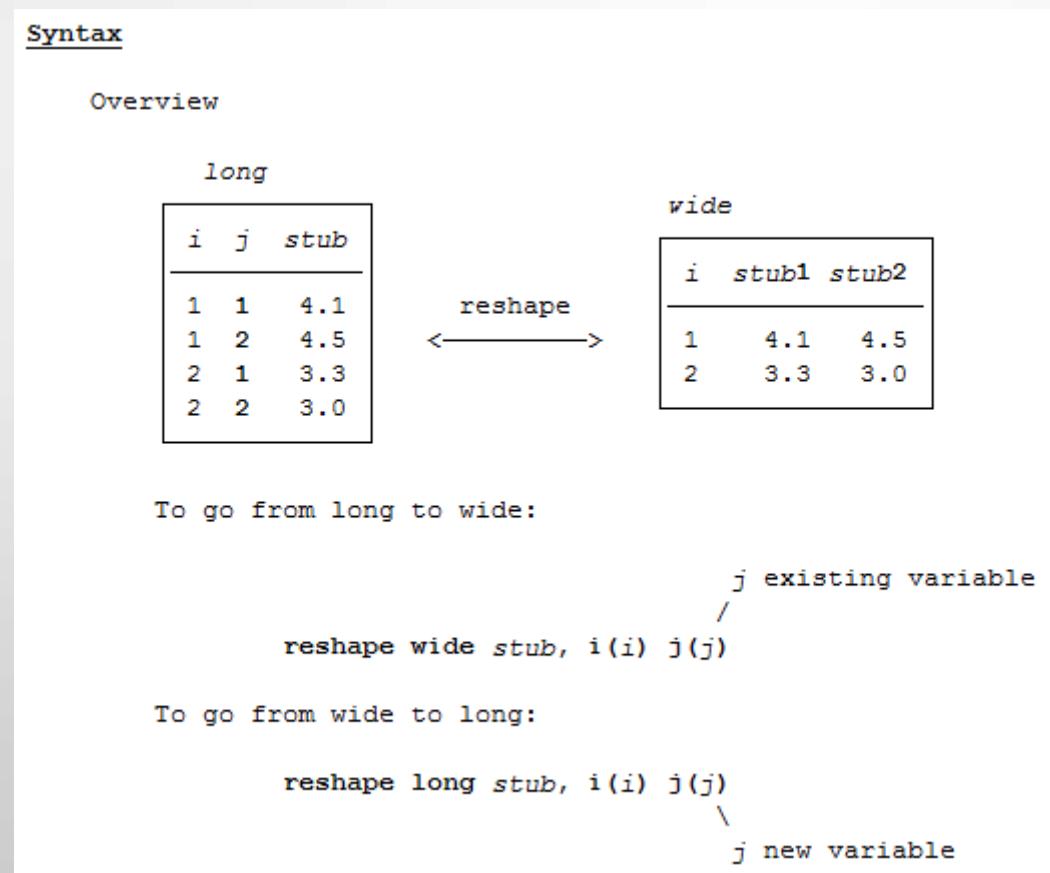
Variable	Obs	Mean	Std. Dev.	Min	Max
hogar	257077	34221.5	19702.98	1	68153
orden	257077	2.814682	1.723089	1	21
educacion	188900	8.71639	4.290771	0	23
sexo	257077	1.50402	.4999848	1	2
edad	257077	31.94582	21.39316	0	107
<hr/>					
ecivil	257077	4.498761	2.799626	1	9
ingreso	86483	229203.7	517893.7	1500	5.40e+07

```
collapse (mean) edad (max) educacion (p50) ingreso, by(hogar)
```

- Se genera una base de datos con observaciones por hogar, con el promedio de edad, máximo de educación, y la mediana del ingreso.

Cambio de estructura: reshape

- Existen dos formas de presentar una base de datos:
 - ❑ wide form (forma horizontal)
 - ❑ long form (forma vertical)



Cambio de estructura: reshape

- Long form:

```
. use "ejreshape.dta", clear  
  
. list if _n<=10
```

	Empresa	año	precio
1.	1	2007	100
2.	1	2008	110
3.	1	2009	120
4.	2	2007	101
5.	2	2008	132
<hr/>			
6.	2	2009	124
7.	3	2008	120
8.	3	2009	118
9.	4	2007	119
10.	4	2008	111

```
reshape wide precio, i(Empresa) j(año)
```

Cambio de estructura: reshape

```
. reshape wide precio, i(Empresa) j(año)
(note: j = 2007 2008 2009)

Data                                long    ->    wide
_____________________________________
Number of obs.                      22    ->      10
Number of variables                 3    ->      4
j variable (3 values)              año    ->  (dropped)
xij variables:                     precio  ->  precio2007 precio2008 precio2009
_____________________________________
```

```
. list if _n<=10
```

	Empresa	pre~2007	pre~2008	pre~2009
1.	1	100	110	120
2.	2	101	132	124
3.	3	.	120	118
4.	4	119	111	123
5.	5	132	142	110
6.	6	123	125	132
7.	7	.	133	135
8.	8	.	145	.
9.	9	.	151	.
10.	10	.	161	.

Cambio de estructura: reshape

```
reshape long precio, i(Empresa) j(year)
```

```
. reshape long precio, i(Empresa) j(year)
(note: j = 2007 2008 2009)

Data                                wide    ->    long
_____________________________________
Number of obs.                      10    ->    30
Number of variables                 4    ->     3
j variable (3 values)              ->    year
xij variables:
    precio2007  precio2008  precio2009  ->    precio
_________________________________
```

```
. list if _n<=10
```

	Empresa	year	precio
1.		1	100
2.		1	110
3.		1	120
4.		2	101
5.		2	132
6.		2	124
7.		3	.
8.		3	120
9.		3	118
10.		4	119



Taller Stata

Clase 6

Javiera Vásquez

Combinar bases de datos

- El comando **merge** se usa para pegar dos bases de datos de manera horizontal, esto significa a una base de datos le agregamos más variables (o columnas) a partir de otra base de datos.
- El comando **append** se usa para pegar dos bases de datos de manera vertical, esto significa que a una base de datos le agregamos más filas (o observaciones)

merge

- Estas son las distintas opciones del comando **merge**

One-to-one merge on specified key variables

```
merge 1:1 varlist using filename [, options]
```

Many-to-one merge on specified key variables

```
merge m:1 varlist using filename [, options]
```

One-to-many merge on specified key variables

```
merge 1:m varlist using filename [, options]
```

Many-to-many merge on specified key variables

```
merge m:m varlist using filename [, options]
```

One-to-one merge by observation

```
merge 1:1 _n using filename [, options]
```

merge

- Suponga que a la base de datos exteps09.dta, donde los individuos se identifican a través de la variable **folio**, queremos pegar las variables que están en la base de datos exteps06.dta, con información para los mismos individuos identificados a través del mismo **folio**:

```
. cd "G:\FNE\FEN_Taller Stata\Bases de datos"  
G:\FNE\FEN_Taller Stata\Bases de datos  
  
. use "exteps09.dta", clear  
La base de datos en la memoria es la master data set  
  
. merge 1:1 folio using exteps06.dta  
(label b18 already defined)  
(label b17 already defined)  
(label b16 already defined)  
(label b15t already defined)  
(label b14 already defined)  
(label b11 already defined)  
(label b10 already defined)  
(label b8 already defined)  
(label b6 already defined)  
(label b4 already defined)  
(label b2 already defined)  
  
La base de datos que se está pegando es la using data set  
  
La variable folio se esta usando para pegar las observaciones 1 a 1  
  
Result # of obs.  
-----  
not matched 4,318  
from master 1,059 (_merge==1) 1.059 observaciones están sólo en la base master (exteps09)  
from using 3,259 (_merge==2) 3.259 observaciones están sólo en la base using (exteps06)  
matched 13,184 (_merge==3) 13.184 observaciones están en ambas bases de datos
```

merge

- Ahora suponga que tenemos una base de datos de pacientes, donde una de las variables corresponde al identificador del doctor que atendió al paciente, y tenemos otra base de datos de los doctores con este mismo identificador, y queremos pegar ambas bases de datos:

```
. use "pacientes.dta", clear  
  
. merge m:1 docid using "doctor.dta"  
  
Result # of obs.  
-----  
not matched 31  
    from master 7 (_merge==1)  
    from using 24 (_merge==2)  
  
matched 104 (_merge==3)
```

append

- Muchas veces las bases de datos están separadas en varias bases de datos, por ejemplo, cuando tenemos las mismas variables pero para distintos años o períodos de tiempo.
- Si queremos pegar estas bases de datos, ocupamos el comando **append**.
- Las bases deben tener un conjunto de variables en común, las que deben tener exactamente el mismo nombre.
- Las variables con el mismo nombre deben además tener el mismo formato (numérico o no numérico)
- Las variables que no están en alguna de las bases de datos quedan con missing value.

append

```
. use ext_mensualeps06.dta, clear

. d

Contains data from ext_mensualeps06.dta
    obs:       650,602
    vars:          6
    size:   12,361,438
    2 May 2011 12:26

      storage  display    value
variable name  type    format   label     variable label

folio           double %12.0g
b2              byte   %9.0g    b2        b2. en este periodo, ¿en cuál de las siguientes situaciones se encontraba?
oficio         float  %57.0g   oficio
b6              byte   %9.0g    b6        b6. este trabajo era de tipo:
b8              byte   %9.0g    b8        b8. en esta ocupación, ud. trabajaba como:
fecha          float  %9.0g

Sorted by: folio
```

append

```
. use ext_mensualeps09.dta, clear

. d

Contains data from ext_mensualeps09.dta
    obs:      569,120
    vars:          6
    size:   8,536,800
                                2 May 2011 12:27

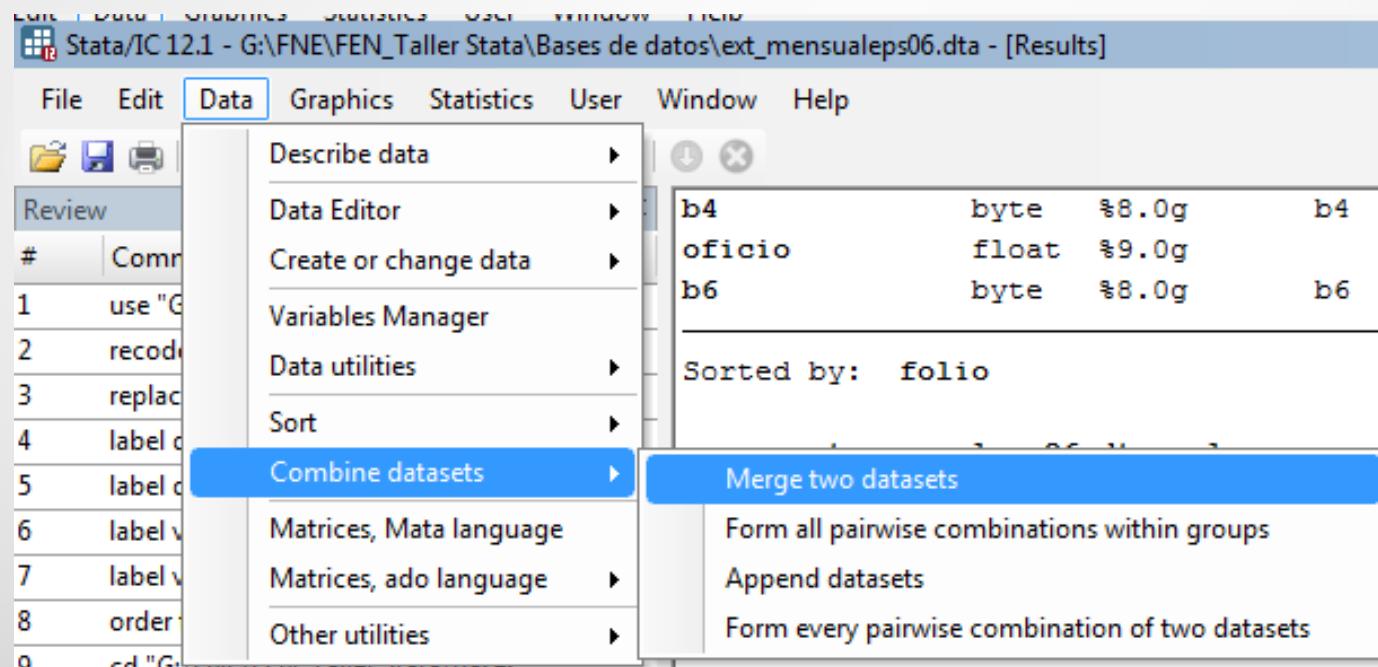
      storage  display    value
variable name  type    format   label   variable label

folio           float   %9.0g
fecha           float   %9.0g
b2              byte   %8.0g   b2       b2. en este periodo ¿en cuál de las siguientes situaciones se encontraba? (1)
b4              byte   %8.0g   b4       b4. ¿en que región trabajaba? si en este periodo realizaba más de un trabajo, re
oficio          float   %9.0g
b6              byte   %8.0g   b6       b6. este trabajo era de tipo:

Sorted by: folio
```

append

```
. use ext_mensualeps06.dta, clear  
  
. append using ext_mensualeps09.dta
```



Condensar base de datos: collapse

- Muchas bases de datos tienen más de una observación por unidad (individuos, colegios, hospitales, países, etc.).
- Si nos interesa trabajar con sólo una observación por unidad podemos condensar la base de datos a través del comando **collapse**.
- Suponga que tiene una base de datos de hogares, donde se tiene información para cada uno de los miembros del hogar, y usted quiere generar una base de datos condensada con sólo una observación por hogar.

Condensar base de datos: collapse

```
. use "hogar.dta", clear  
  
. list hogar orden if _n<10
```

	hogar	orden
1.	1	1
2.	1	2
3.	1	3
4.	2	1
5.	2	2
6.	2	3
7.	3	1
8.	3	2
9.	4	1

```
. sum
```

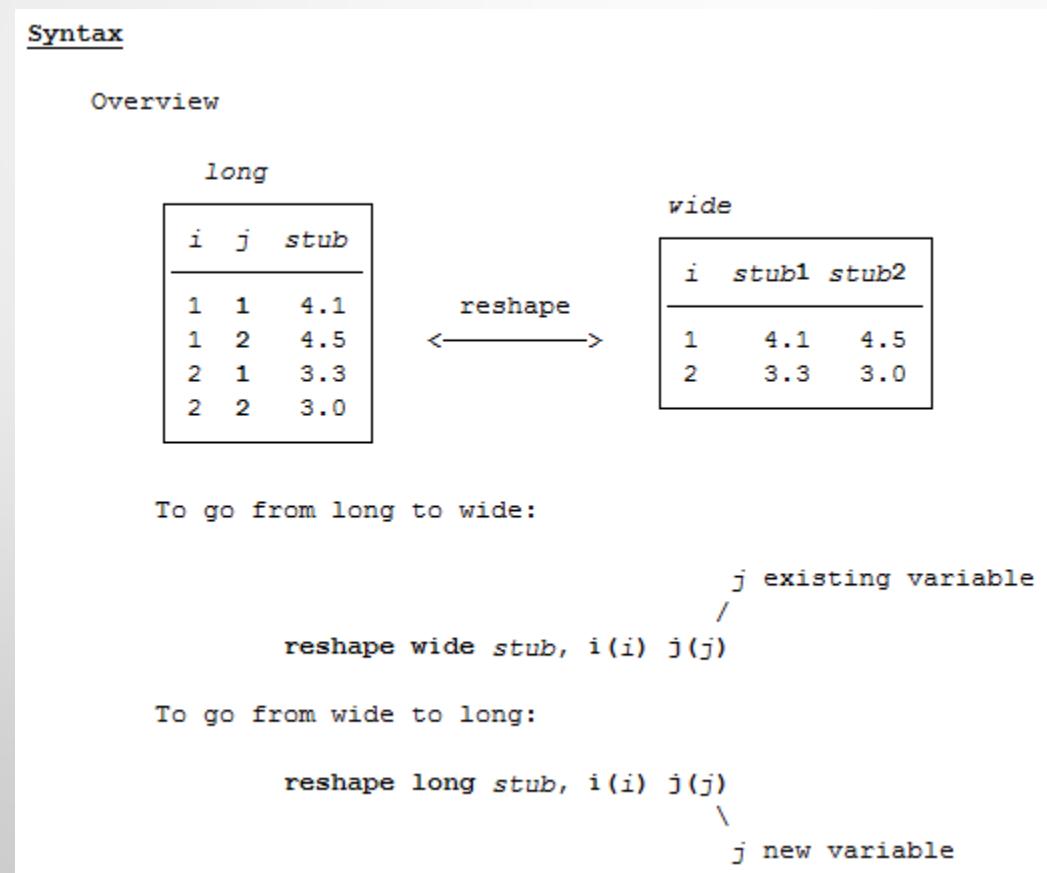
Variable	Obs	Mean	Std. Dev.	Min	Max
hogar	257077	34221.5	19702.98	1	68153
orden	257077	2.814682	1.723089	1	21
educacion	188900	8.71639	4.290771	0	23
sexo	257077	1.50402	.4999848	1	2
edad	257077	31.94582	21.39316	0	107
ecivil	257077	4.498761	2.799626	1	9
ingreso	86483	229203.7	517893.7	1500	5.40e+07

```
collapse (mean) edad (max) educacion (p50) ingreso, by(hogar)
```

- Se genera una base de datos con observaciones por hogar, con el promedio de edad, máximo de educación, y la mediana del ingreso.

Cambio de estructura: reshape

- Existen dos formas de presentar una base de datos:
 - ❑ wide form (forma horizontal)
 - ❑ long form (forma vertical)



Cambio de estructura: reshape

- Long form:

```
. use "ejreshape.dta", clear  
  
. list if _n<=10
```

	Empresa	año	precio
1.	1	2007	100
2.	1	2008	110
3.	1	2009	120
4.	2	2007	101
5.	2	2008	132
6.	2	2009	124
7.	3	2008	120
8.	3	2009	118
9.	4	2007	119
10.	4	2008	111

```
reshape wide precio, i(Empresa) j(año)
```

Cambio de estructura: reshape

```
. reshape wide precio, i(Empresa) j(año)
(note: j = 2007 2008 2009)

Data                                long    ->    wide
_____________________________________
Number of obs.                      22    ->      10
Number of variables                 3    ->      4
j variable (3 values)              año    ->  (dropped)
xij variables:                     precio  ->  precio2007 precio2008 precio2009
_____________________________________
```

```
. list if _n<=10
```

	Empresa	pre~2007	pre~2008	pre~2009
1.	1	100	110	120
2.	2	101	132	124
3.	3	.	120	118
4.	4	119	111	123
5.	5	132	142	110
6.	6	123	125	132
7.	7	.	133	135
8.	8	.	145	.
9.	9	.	151	.
10.	10	.	161	.

Cambio de estructura: reshape

```
reshape long precio, i(Empresa) j(year)
```

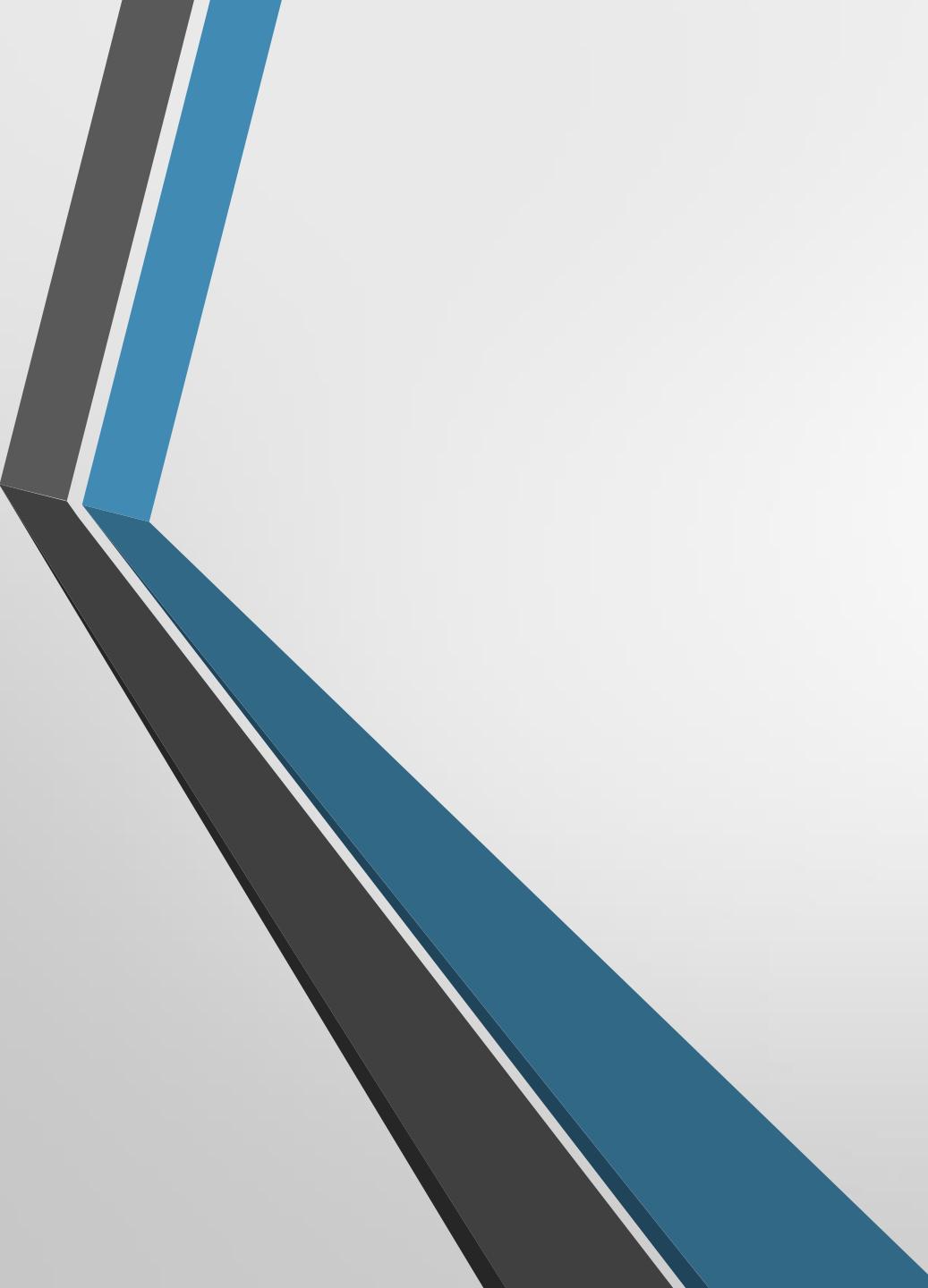
```
. reshape long precio, i(Empresa) j(year)
(note: j = 2007 2008 2009)

Data                                wide    ->    long
_____________________________________
Number of obs.                      10    ->    30
Number of variables                 4    ->     3
j variable (3 values)              ->    year
xij variables:
    precio2007  precio2008  precio2009  ->    precio
_________________________________
```



```
. list if _n<=10
```

	Empresa	year	precio
1.		1	100
2.		1	110
3.		1	120
4.		2	101
5.		2	132
6.		2	124
7.		3	.
8.		3	120
9.		3	118
10.		4	119



Taller Stata

Clase 7

Javiera Vásquez

Factores de expansión

- Cuando trabajamos con muestras de datos que representan una población, la muestra generalmente fue obtenida con un diseño muestral apropiado para lograr la representatividad de la población bajo estudio.
- Si este es el caso, la base de datos por lo general debería contener el factores de expansión o ponderador. Por ejemplo, en la encuesta Casen el factor de expansión se llama `expr`.
- El factor de expansión indica cuantas observación de la población representa cada observación muestral
- Por ejemplo, en la Casen 2017, el mínimo del factor de expansión es 2 y máximo 5828.

Factor de expansión – Casen 2017

	folio	o	region	comuna	expr
1	110110020201	1	RegiOn ...	Iquique	39
2	110110020301	1	RegiOn ...	Iquique	39
3	110110020401	1	RegiOn ...	Iquique	39
4	110110020401	2	RegiOn ...	Iquique	39
5	110110020501	1	RegiOn ...	Iquique	39
6	110110020501	2	RegiOn ...	Iquique	39
7	110110020501	3	RegiOn ...	Iquique	39
8	110110020601	1	RegiOn ...	Iquique	39
9	110110020601	2	RegiOn ...	Iquique	39
10	110110020901	1	RegiOn ...	Iquique	39
11	110110020901	2	RegiOn ...	Iquique	39
12	110110020901	3	RegiOn ...	Iquique	39
13	110110020901	4	RegiOn ...	Iquique	39
14	110110021001	1	RegiOn ...	Iquique	39
15	110110021001	2	RegiOn ...	Iquique	39
16	110110021001	3	RegiOn ...	Iquique	39
17	110110030101	1	RegiOn ...	Iquique	54
18	110110030201	1	RegiOn ...	Iquique	54
19	110110030201	2	RegiOn ...	Iquique	54
20	110110030401	1	RegiOn ...	Iquique	54

. sum expr, d

Factor de Expansión Regional

	Percentiles	Smallest		
1%	9	2		
5%	15	2		
10%	20	2	Obs	216,439
25%	33	2	Sum of Wgt.	216,439
50%	63		Mean	82.27452
			Std. Dev.	87.36169
75%	104	5828	Largest	
90%	171	5828	Variance	7632.065
95%	216	5828	Skewness	19.37765
99%	336	5828	Kurtosis	1075.01

Factores de expansión

- Así, para obtener estadísticas y hace cualquier análisis con los datos, y que estas representen a la población deberíamos considerar el factor de expansión.
- Una alternativa es expandir la base de datos usando el factor de expansión y el comando `expand`. El problema es que se puede agrandar mucho la base de datos

Factores de expansión

- Por ejemplo, dejemos solo las observaciones de la comuna de colina:

```
. cd "C:\jvasquez\Casen"  
C:\jvasquez\Casen  
  
. use "Casen 2017.dta", clear  
  
. keep if comuna==13301  
(215,583 observations deleted)  
  
. sum folio  
  
Variable | Obs Mean Std. Dev. Min Max  
folio | 856 1.33e+12 7755666 1.33e+12 1.33e+12
```

- Son 856 observaciones muestrales
- La variable `expc` representa el factor de expansión comunal, es decir, cuánto representa cada persona entrevistada en la comuna de colina para representar la población de colina.

Factores de expansión

- Al aplicar el comando `expand` con el factor de expansión comunal:

```
. expand expc
```

```
(122,963 observations created)
```

```
. sum folio
```

Variable	Obs	Mean	Std. Dev.	Min	Max
folio	123,819	1.33e+12	8519007	1.33e+12	1.33e+12

- La cantidad de observaciones pasa a ser la población comunal de colina, es decir cerca de 124 mil observaciones.
- Pero en la mayoría de los casos no será eficiente expandir la base de datos
- Es mejor trabajar con los factores de expansión en los comandos de análisis que lo permiten.

Factores de expansión

- Existen 4 tipos de factores de expansión en STATA:
 - ❑ **Frequency weights (fweights)**: son números enteros que indican el número de veces en que la observación es efectivamente observada. Este factor se utiliza cuando la base de datos ha sido condensada y contiene una variable que indica la frecuencia con que cada observación ocurre. Es un error usar el frequency weight como el sampling weight, este error sólo tiene consecuencias en las estimaciones de las varianzas, valores-p y errores estándar. Un ejemplo sería si tenemos datos de pasajeros de aerolineas por ruta, y condensamos la base de datos por ruta, el número de pasajeros en la ruta sería luego el ponderador o factor de expansión si queremos sacar estadísticas representativas que le den mayor peso a las rutas con más pasajeros.

Factores de expansión

- **Sampling weights (pweights):** en la mayoría de las encuestas donde la muestra fue extraída mediante un proceso aleatorio, el inverso de la probabilidad de selección de cada unidad muestral corresponde al factor de expansión. STATA utiliza este valor como el número de sujetos en la población que representa cada observación en el computo de las proporciones, los promedios y parámetros de regresión. Además calcula las varianzas de manera robusta, de manera de obtener estimaciones correctas del error estándar e intervalos de confianza.

Factores de expansión

- **Analytical weights (aweights)**: este tipo de factor de expansión es apropiado cuando se trabaja con datos que representan promedios, y que el ponderador contiene el número de observaciones con la que fue construida el promedio.
- **Importance weights (iweights)**: este ponderador no tiene definición estadística formal, puede ser usando para implementar el propio peso o ponderador (importancia) a las observaciones de acuerdo al criterio discrecional del investigador.

Factores de expansión

Veamos el siguiente ejemplo para el cálculo del salario promedio (ingreso de la ocupación principal) con los datos de la encuesta Casen 2015:

```
. sum yoprcor [pw=expr]
pweights not allowed
r(101);
```

```
. mean yoprcor [pw=expr]
```

Mean estimation Number of obs = 52,796

	Mean	Std. Err.	[95% Conf. Interval]
yoprcor	1465716	14538.54	1437220 1494212

```
. tabstat yoprcor [pw=expr], stats(mean p50)
pweights not allowed
r(101);
```

```
. sum yoprcor [w=expr]
(analytic weights assumed)
```

Variable	Obs	Weight	Mean	Std. Dev.	Min	Max
yoprcor	52,796	13297413	1465716	2136217	20000	9000000

```
. mean yoprcor [w=expr]
(frequency weights assumed)
```

Mean estimation Number of obs = 13,297,413

	Mean	Std. Err.	[95% Conf. Interval]
yoprcor	1465716	585.8113	1464568 1466864

Factores de expansión

- Como el ponderador de la encuesta Casen viene de un diseño muestral deberíamos utilizar `pweights`, sin embargo, el comando `sum` y `tabstat` como acabamos de ver no aceptan esta opción de factor de expansión.
- Pero para estos comandos podemos utilizar `fweights` (`weights`) ya que el cálculo de proporciones, promedios, etc. Es igual con ambos tipos de ponderadores, lo que afecta es en el cálculo de error estándar e intervalos de confianza.
- Podemos notar que el promedio del salario es igual independiente de cual del factor de expansión utilizado.

Factores de expansión

- Veamos otro ejemplo, generemos una variable que indique si la persona tiene un ingreso por debajo del promedio:

```
. sum yoprcor [w=expr]
(analytic weights assumed)
```

Variable	Obs	Weight	Mean	Std. Dev.	Min	Max
yoprcor	52,796	13297413	1465716	2136217	20000	9000000

```
. return list
```

scalars:

```
r(N)      = 52796
r(sum_w)   = 13297413
r(mean)    = 1465716.021173517
r(Var)     = 4563423742337.147
r(sd)      = 2136217.157111408
r(min)    = 20000
r(max)    = 9000000
r(sum)    = 19490231274261
```

```
. g dmenor=1 if yoprcor<r(mean)
(78,557 missing values generated)

. replace dmenor=0 if dmenor==.
(78,557 real changes made)

. replace dmenor=. if yoprcor==.
(71,023 real changes made, 71,023 to missing)
```

Análisis de datos

```
. tab dmenor [pw=expr]  
pweights not allowed  
r(101);
```

```
. tab dmenor [w=expr]  
(frequency weights assumed)
```

dmenor	Freq.	Percent	Cum.
0	4,499,176	33.83	33.83
1	8,798,237	66.17	100.00
Total	13,297,413		100.00

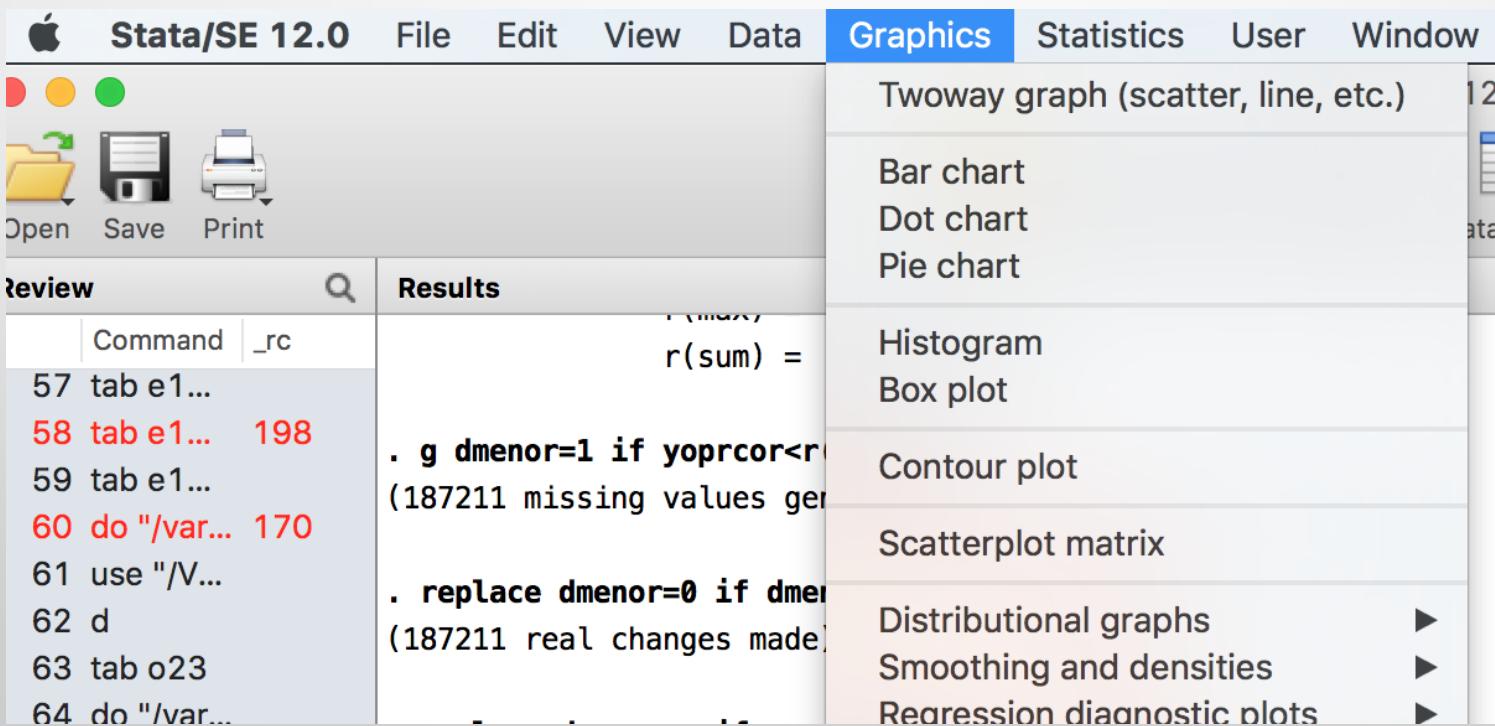
```
. proportion dmenor [pw=expr]
```

Proportion estimation Number of obs = 52,796

	Proportion	Std. Err.	Logit [95% Conf. Interval]	
dmenor				
0	.3383497	.0030137	.3324682	.3442816
1	.6616503	.0030137	.6557184	.6675318

Gráficos

- Existen diferentes tipos de gráficos que se pueden hacer en STATA: histograma, de barras, de torta, de dos entradas, etc.
- Los comandos de los gráficos son difíciles de memorizar con todas sus opciones, en este caso podemos trabajar con las ventanas y luego copiar el comando en el do-file.

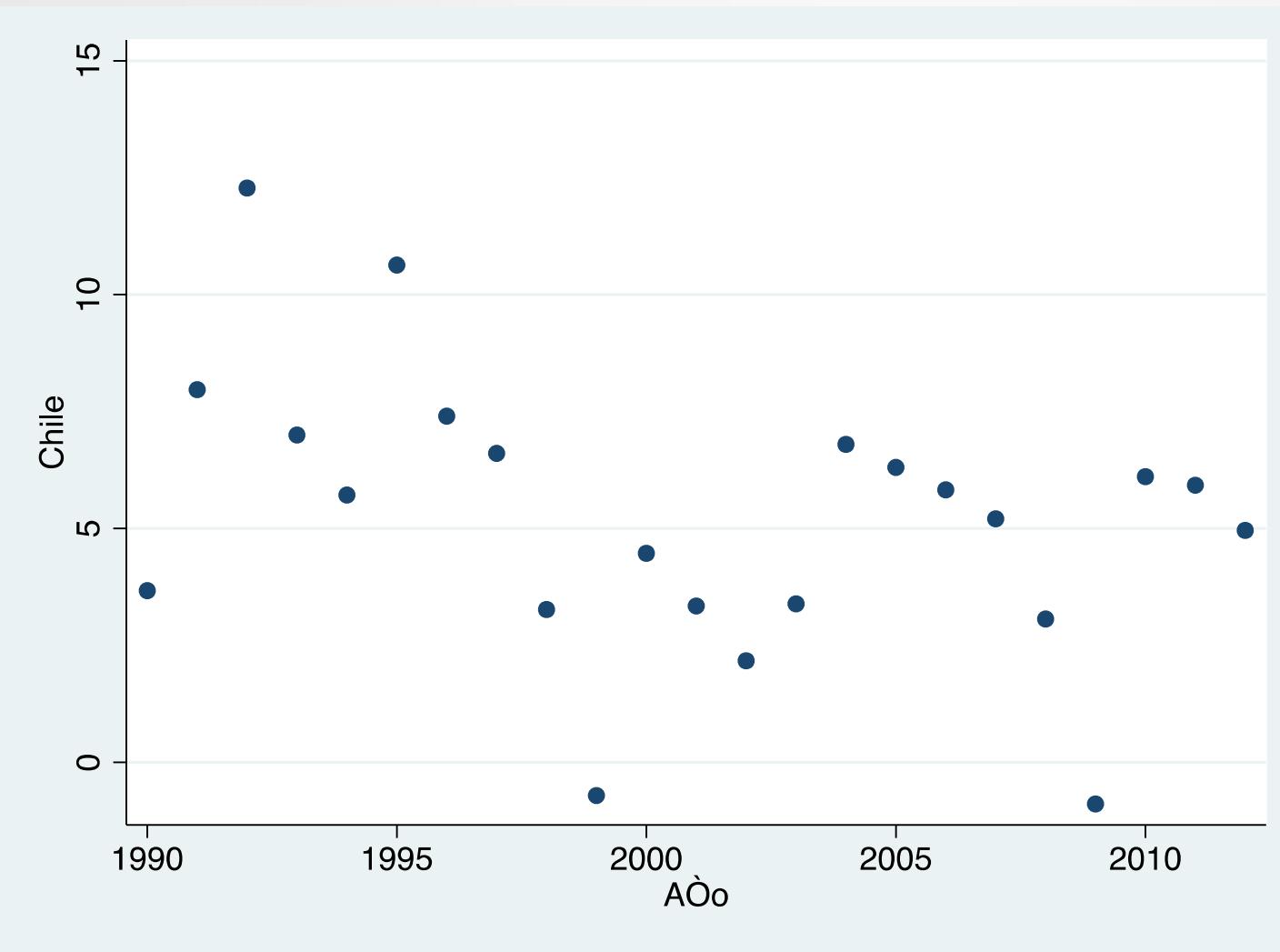


two way graph

- Estos gráficos se utilizan para analizar la relación entre dos variables.
- Por ejemplo, usando la base de tasas de crecimiento del PIB de varios países, podemos graficar la evolución del crecimiento del PIB de Chile:

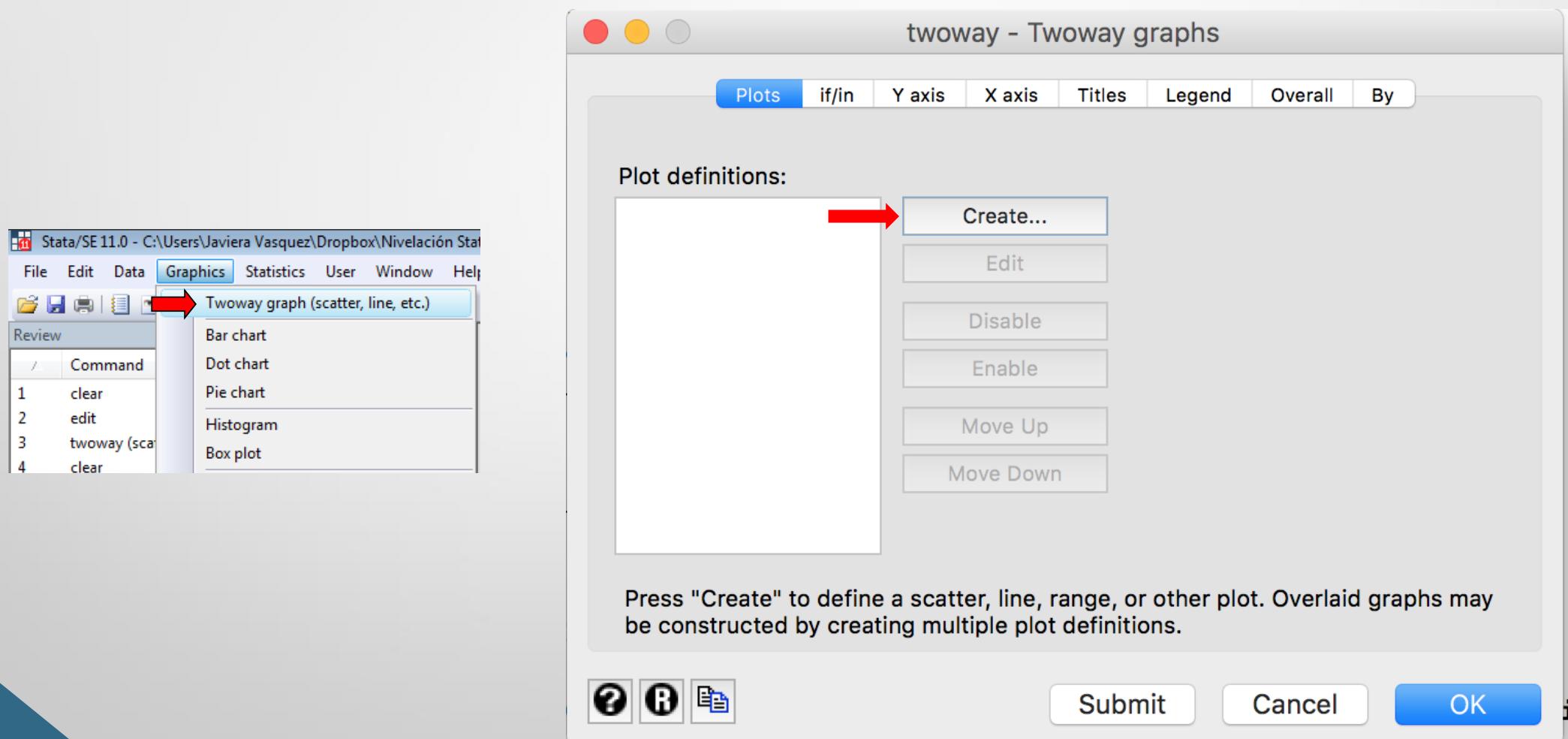
```
twoway (scatter chile año)
```

two way graph



two way graph

- Este mismo gráfico lo podemos hacer a través de las ventanas



two way graph

Plot if/in

Choose a plot category and type

Basic plots Range plots Contour plots Fit plots Immediate plots Advanced plots

Basic plots: (select type)

Scatter Line Connected Area Bar

Plot type: (scatterplot)

Y variable: X variable:

Sort on x variable

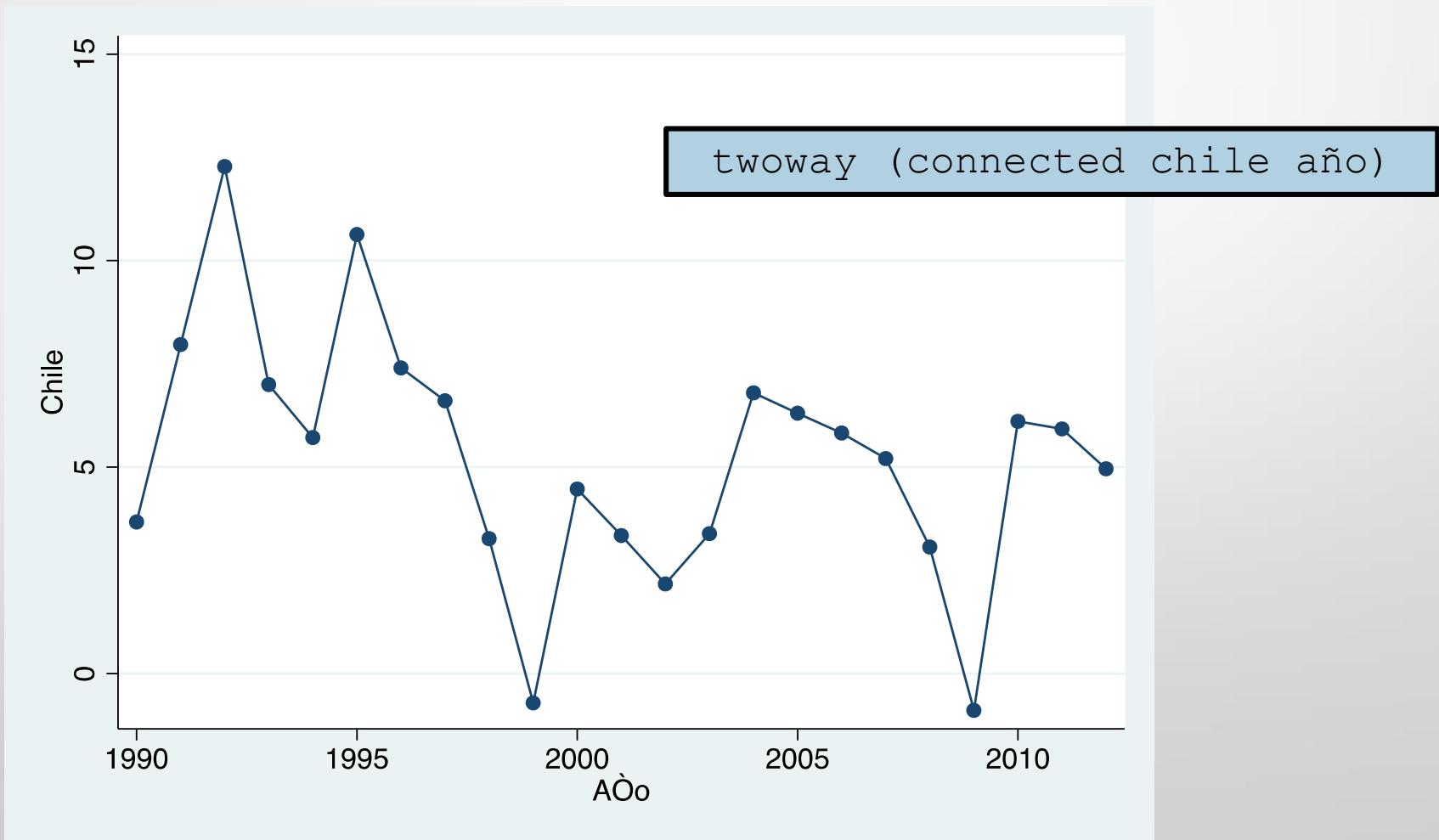
Marker properties Marker weights

?

Submit Cancel Accept

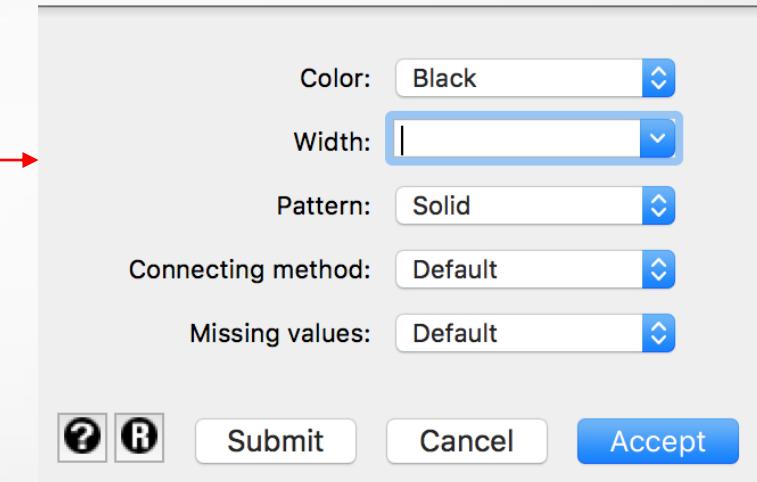
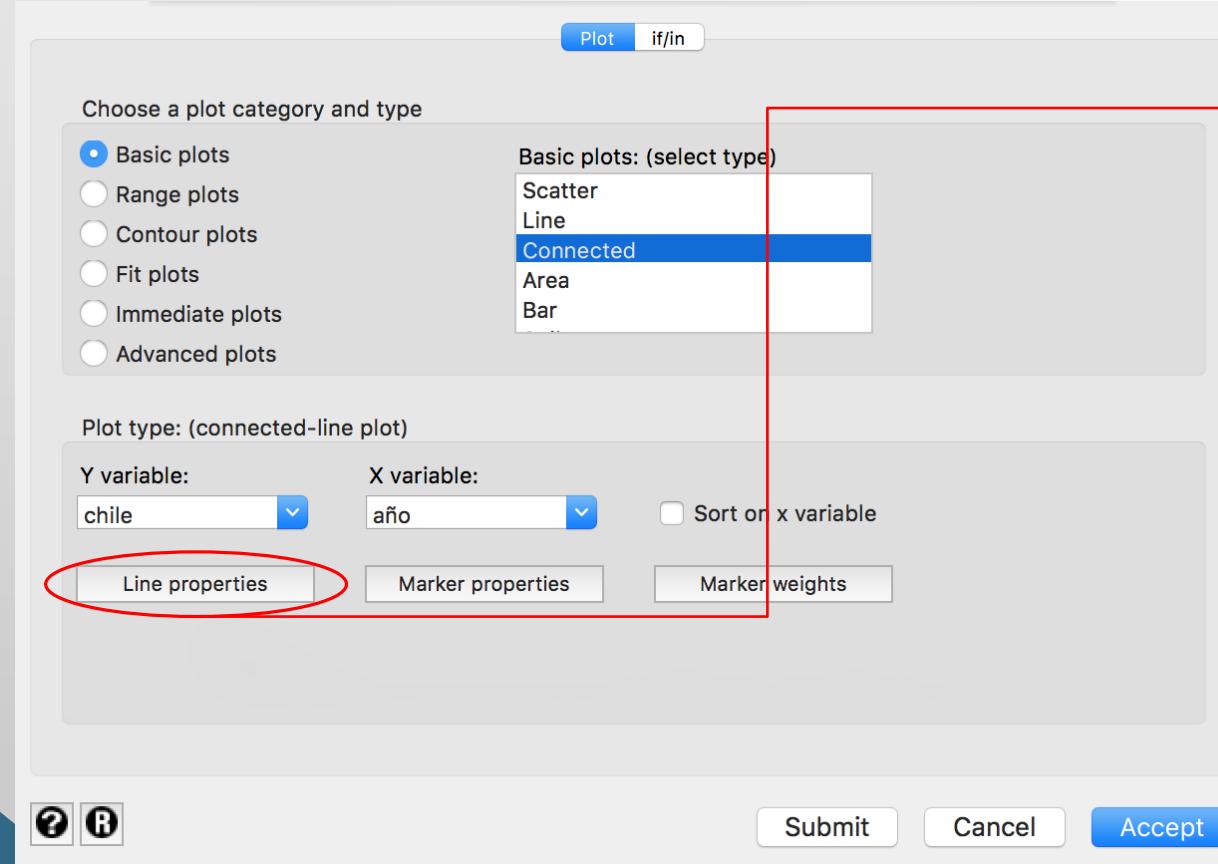
two way graph

- Si en vez de scatter seleccionamos connected, el gráfico aparecerá con los puntos de datos conectados:



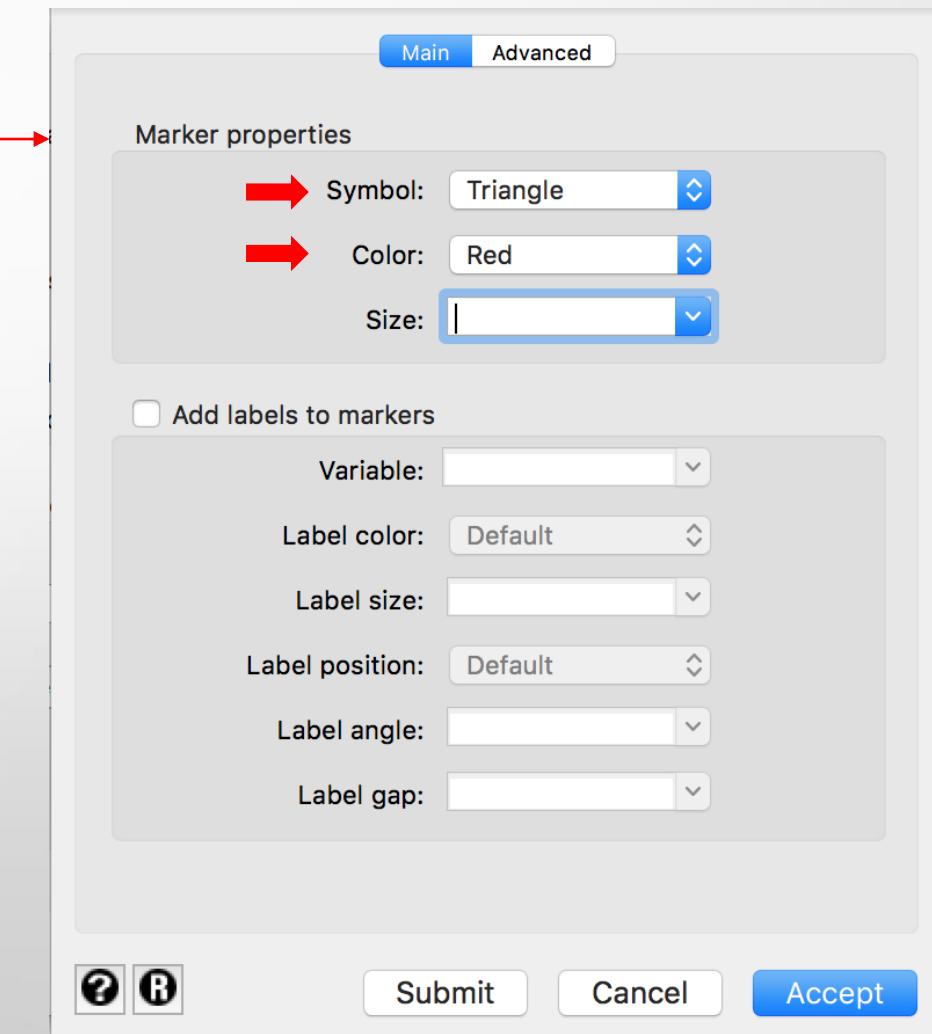
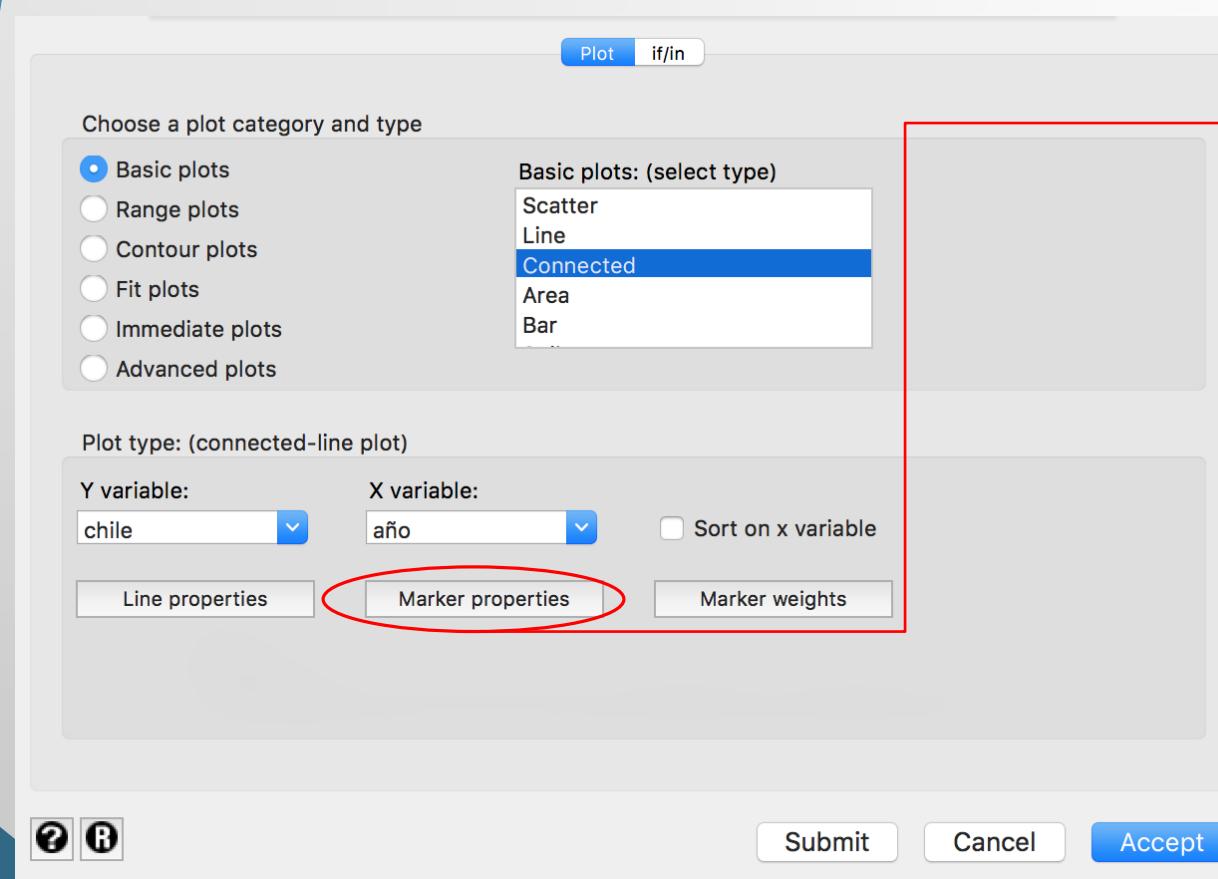
two way graph

- Luego podemos usar las distintas opciones de la venta de gráficos:



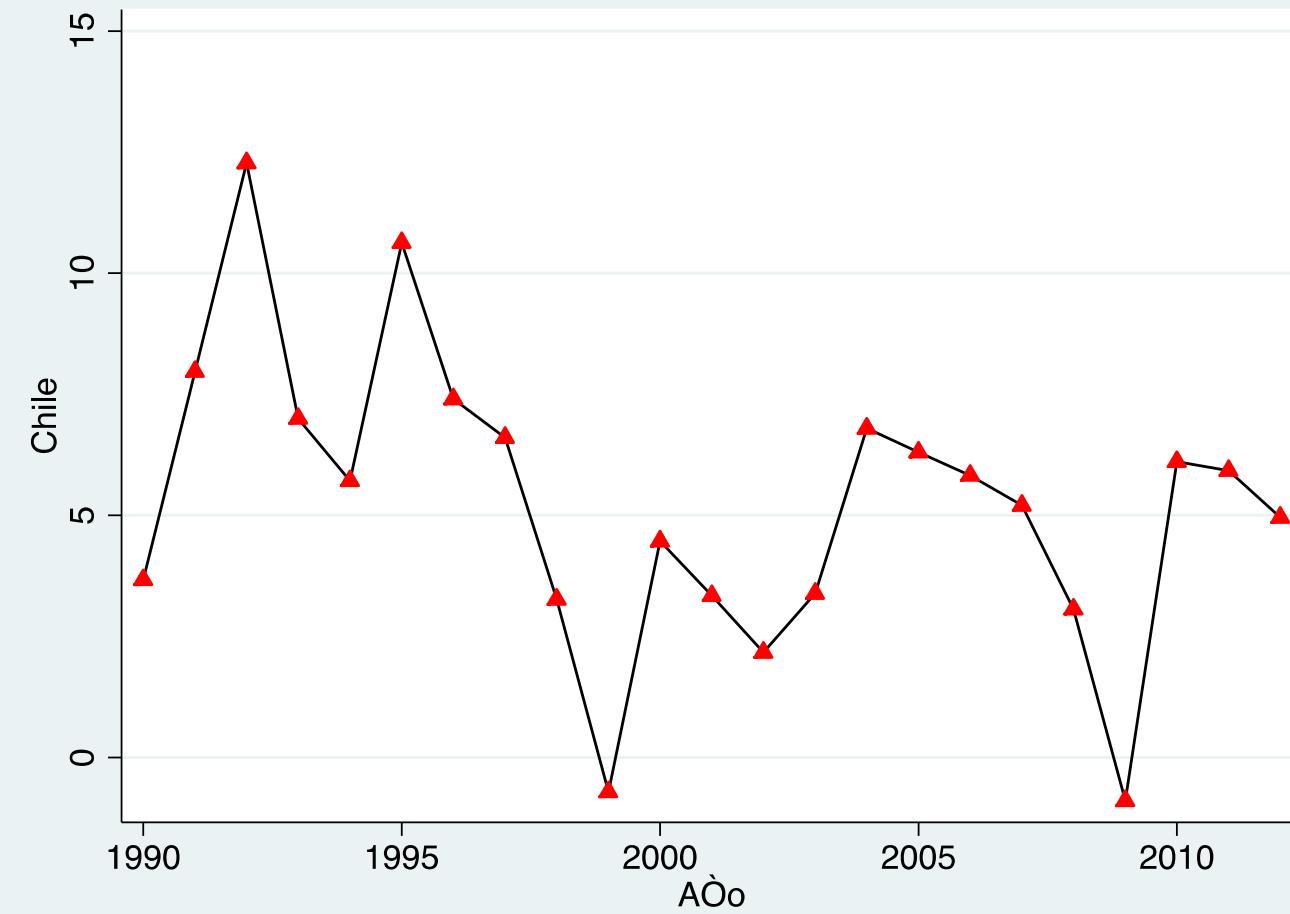
two way graph

- Luego podemos usar las distintas opciones de la venta de gráficos:



two way graph

```
twoway (connected chile año, mcolor(red) msymbol(triangle) lcolor(black) ///
lpattern(solid))
```

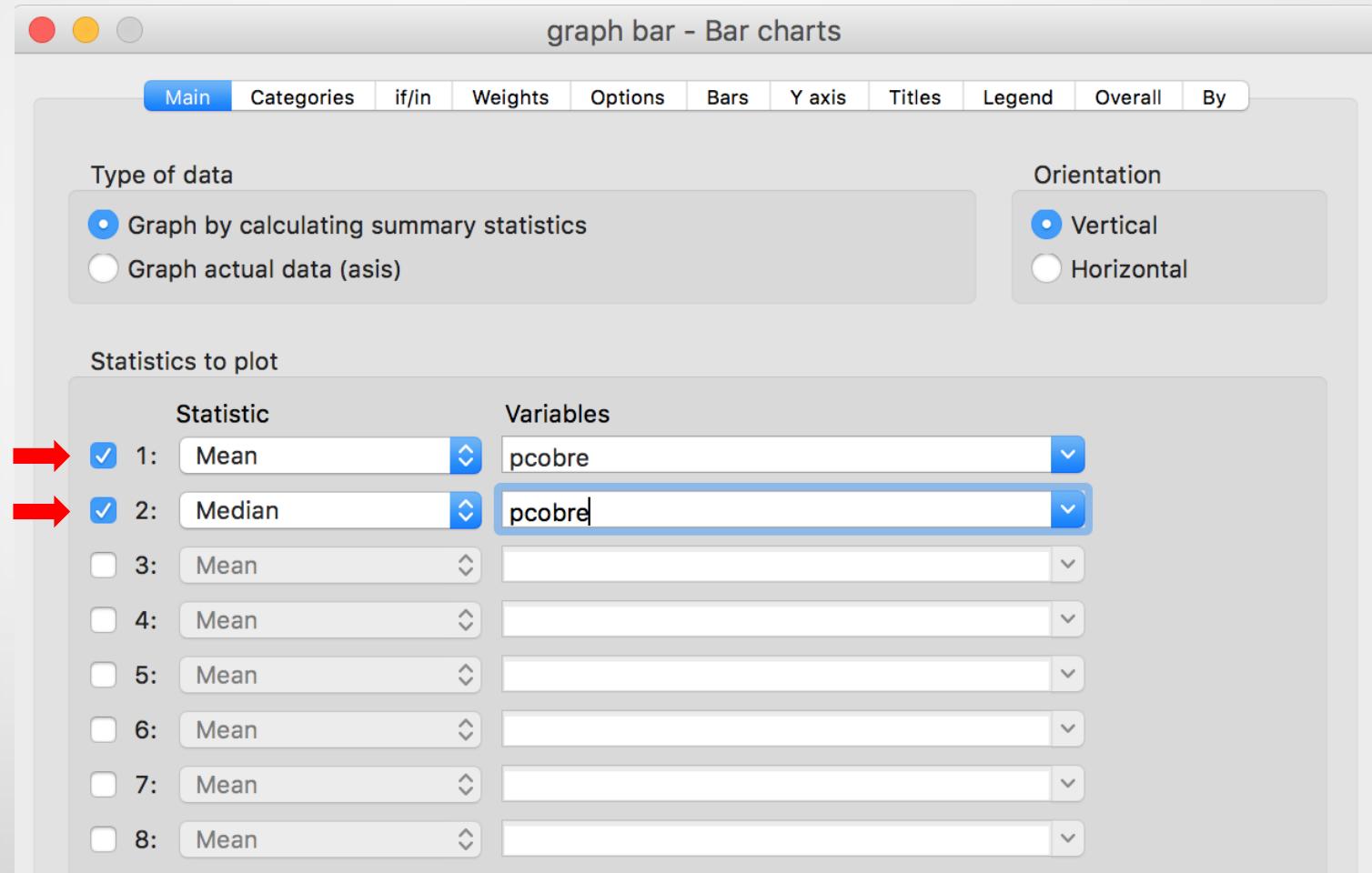
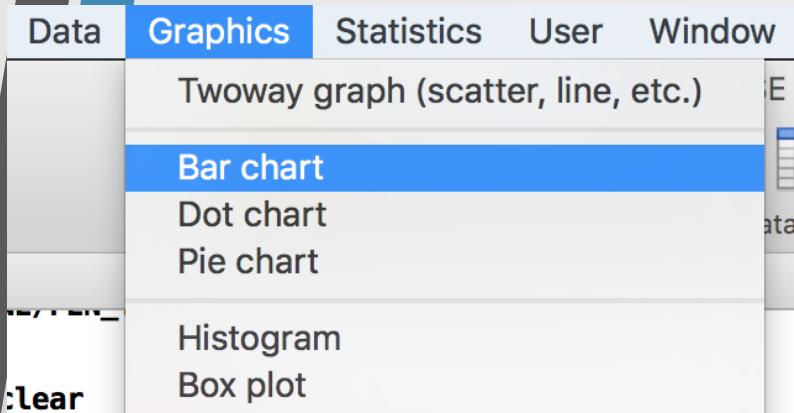


Bar chart

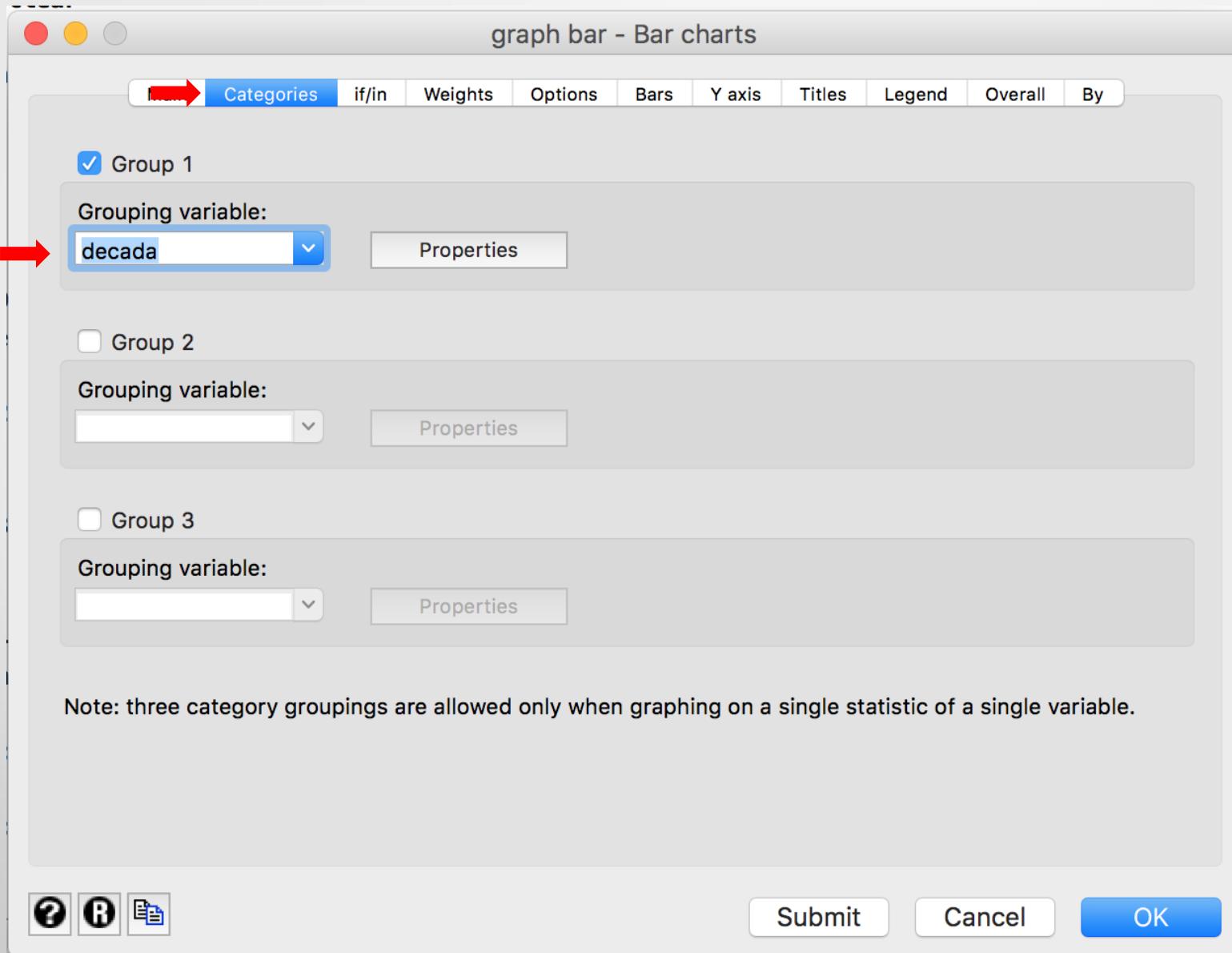
- El gráfico de barras nos permite resumir de manera gráfica y comparativa estadísticas descriptivas, como la media, mediana, entre otras, para una o más variables de interés.
- Trabajemos ahora con la base de datos histórica del precio del cobre, y generemos la variable década:

```
. use pcobre.dta, clear  
  
. g decada=1 if aÑo<1990  
(279 missing values generated)  
  
. replace decada=2 if aÑo>=1990 & aÑo<2000  
(120 real changes made)  
  
. replace decada=3 if aÑo>=2000 & aÑo<2010  
(120 real changes made)  
  
. replace decada=4 if aÑo>=2010  
(39 real changes made)  
  
. label define decada 1 "80" 2 "90" 3 "00"4 "10"  
  
. label values decada decada
```

Bar chart

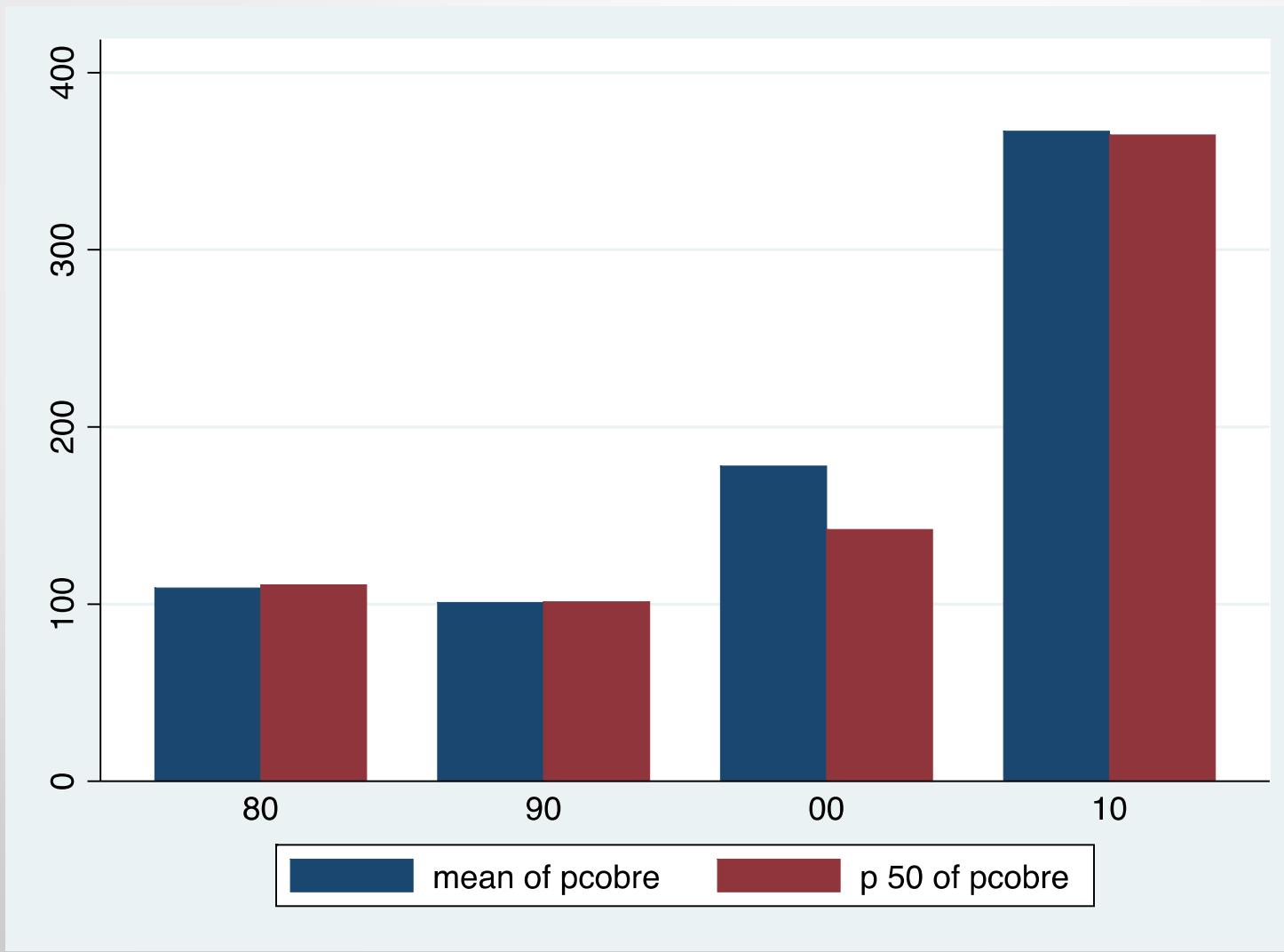


Bar chart



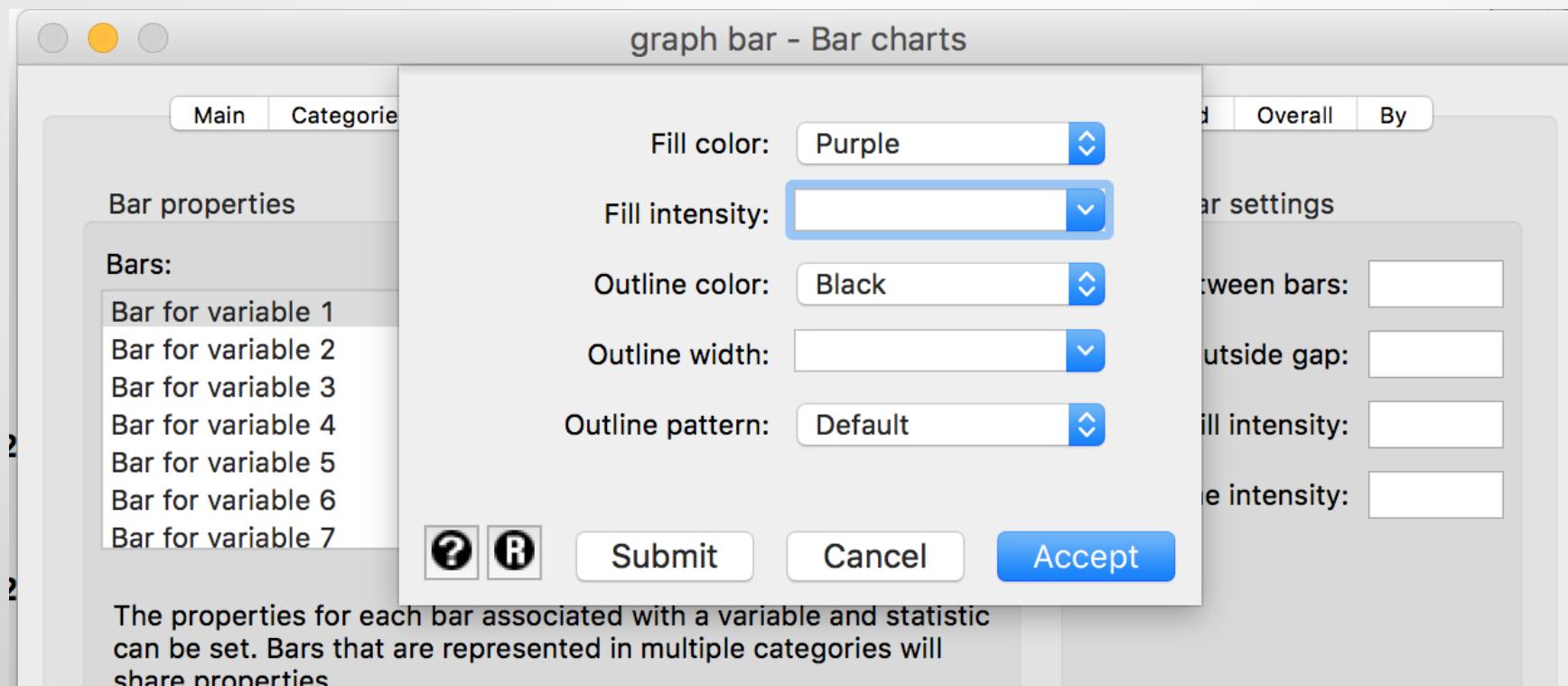
Bar chart

```
graph bar (mean) pcobre (median) pcobre, over(decada)
```



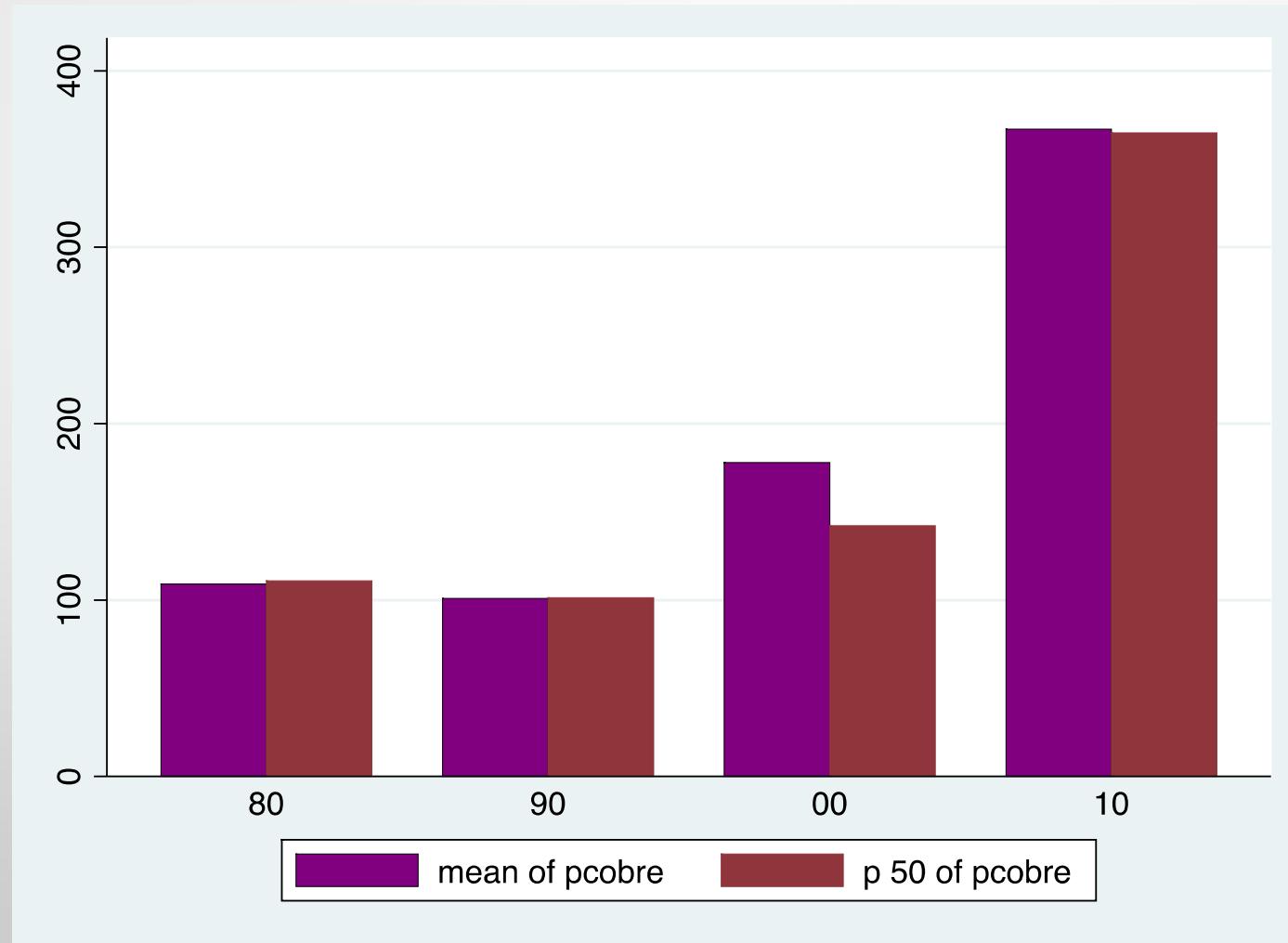
Bar chart

- Podemos cambiar el color de las barras:



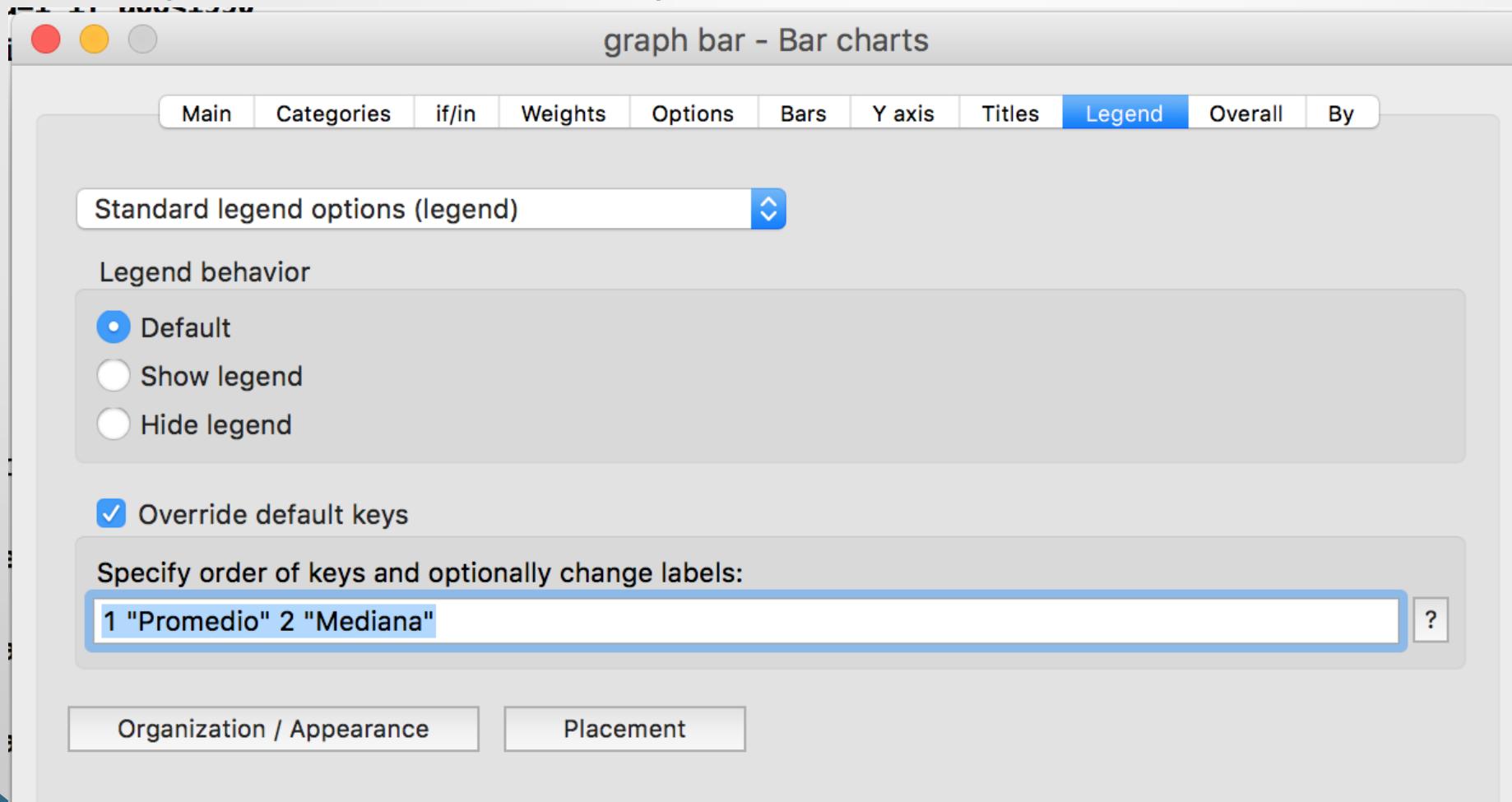
Bar chart

```
graph bar (mean) pcobre (median) pcobre, over(decada) bar(1, fcolor(purple) ///  
lcolor(black))
```



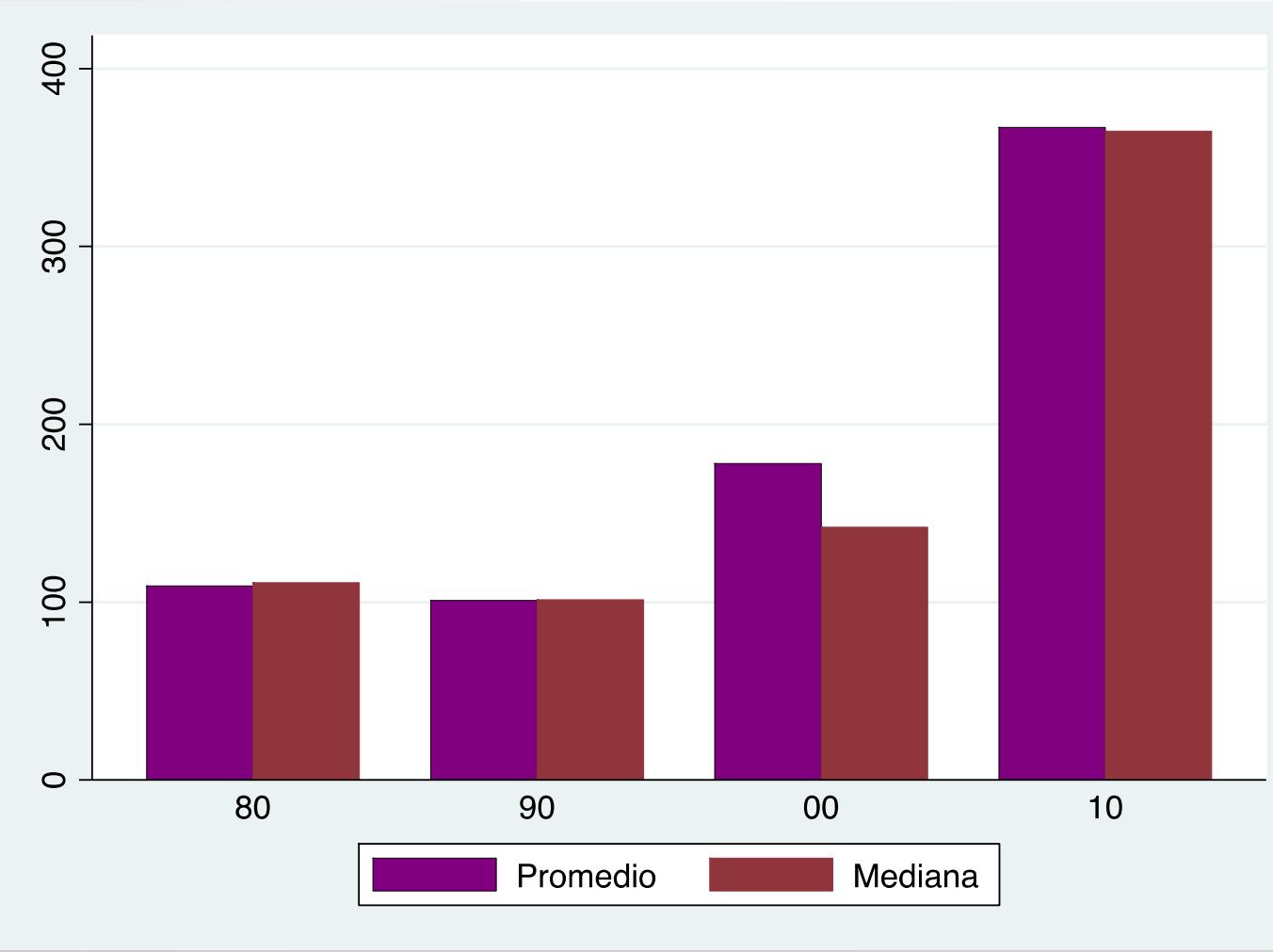
Bar chart

- También podemos cambiar la leyenda:



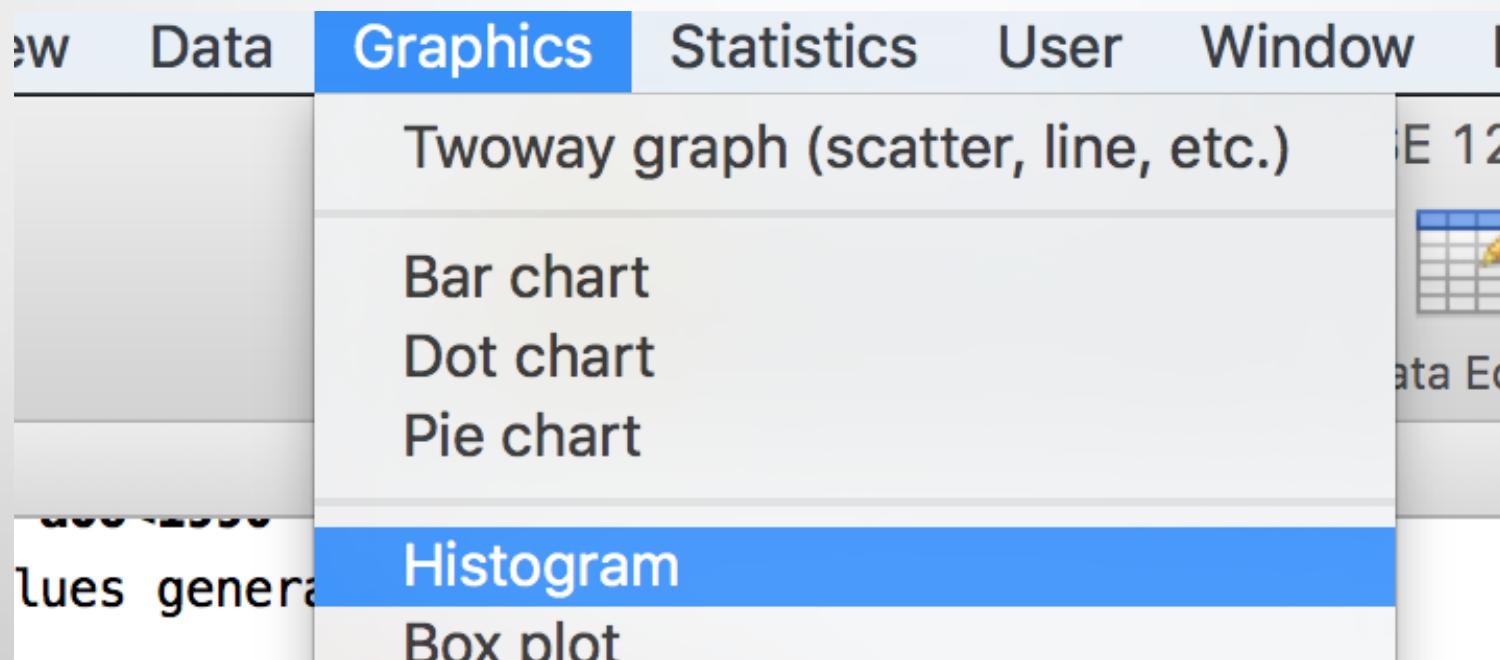
Bar chart

```
graph bar (mean) pcobre (median) pcobre, over(decada) bar(1, fcolor(purple) ///  
lcolor(black)) legend(order(1 "Promedio" 2 "Mediana"))
```

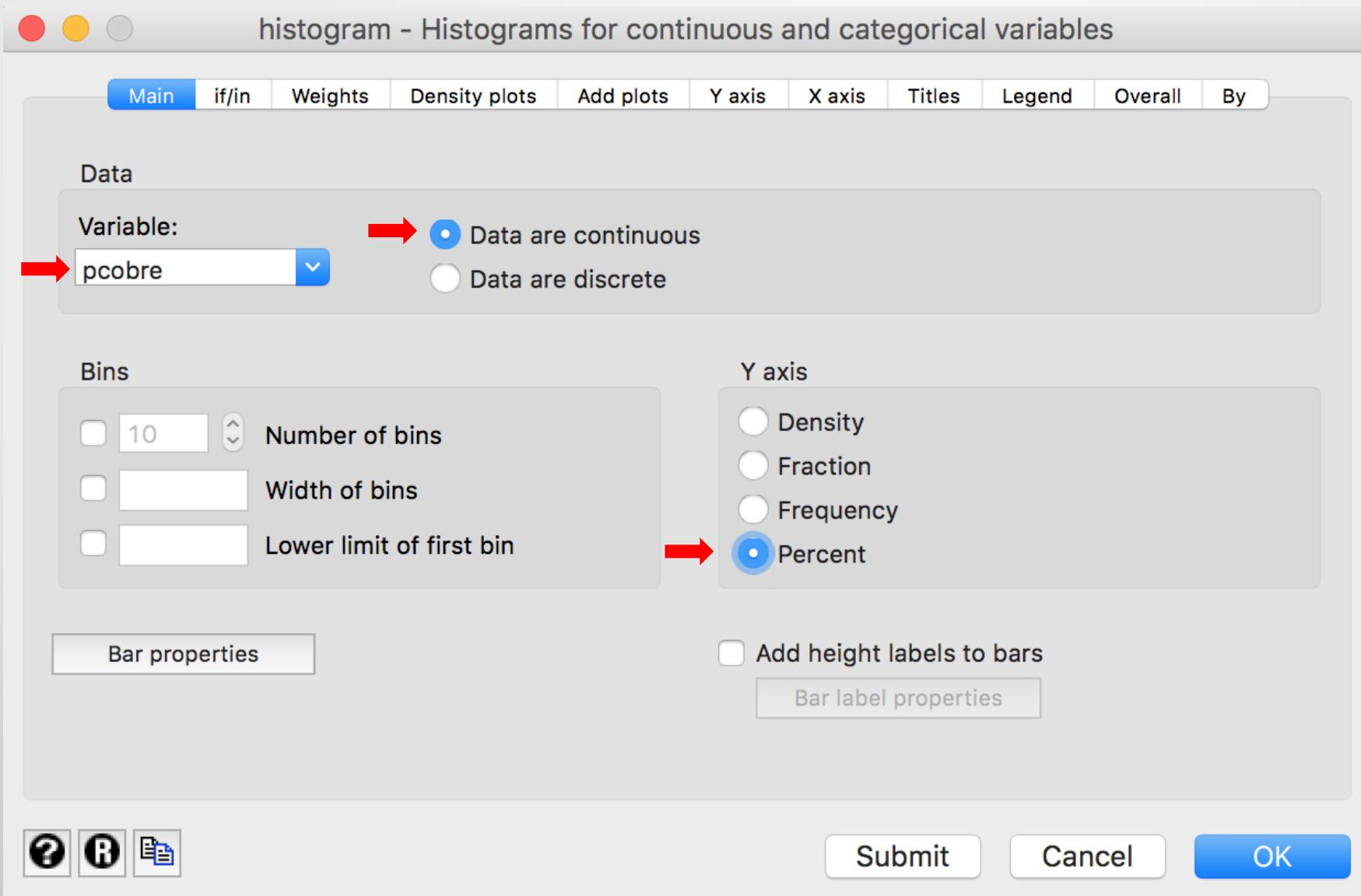


Histograma

- Un histograma es un tipo de gráfico que nos permite caracterizar como se distribuye la muestra o población de datos en los valores que puede tomar la variable:

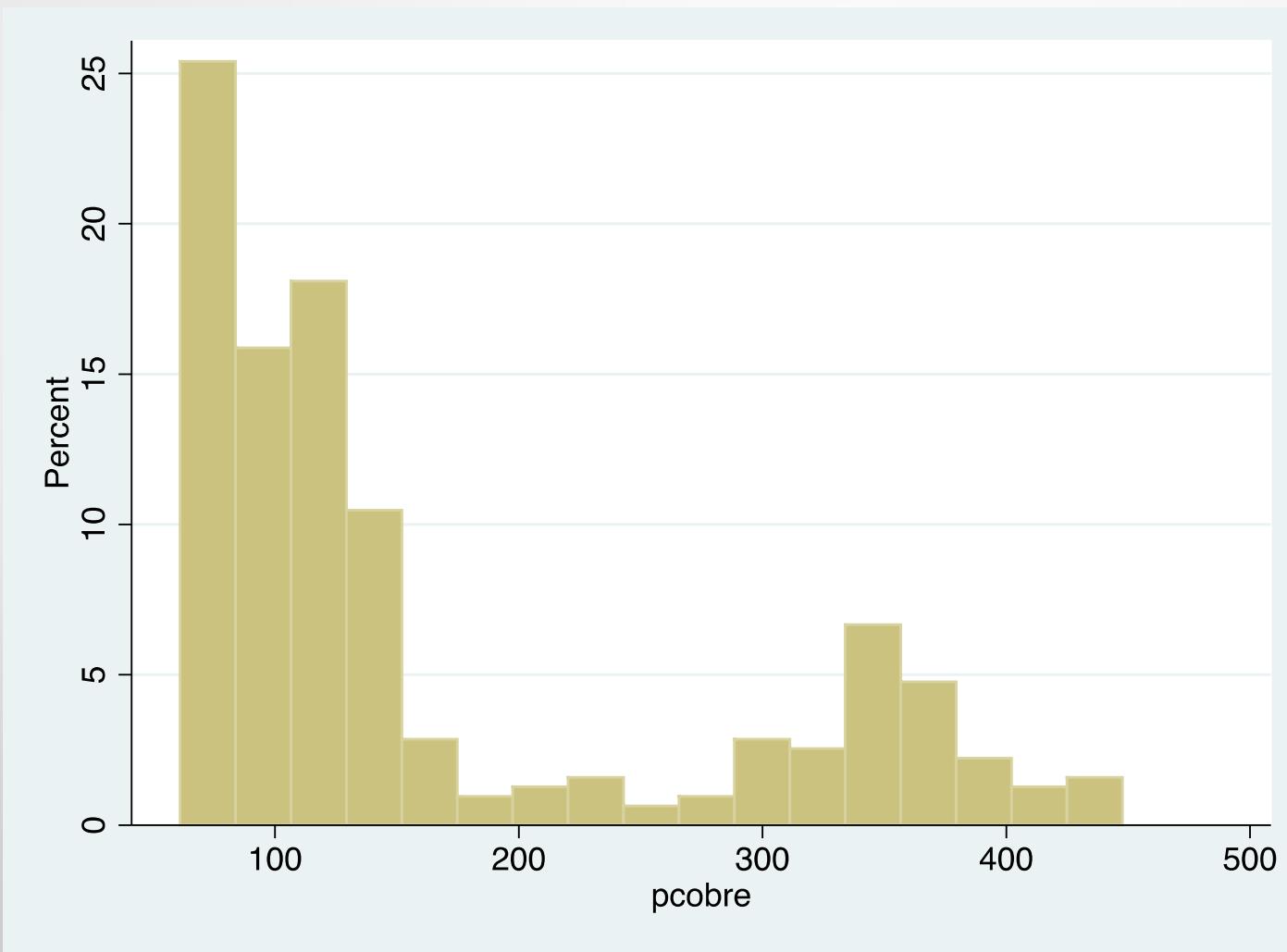


Histograma



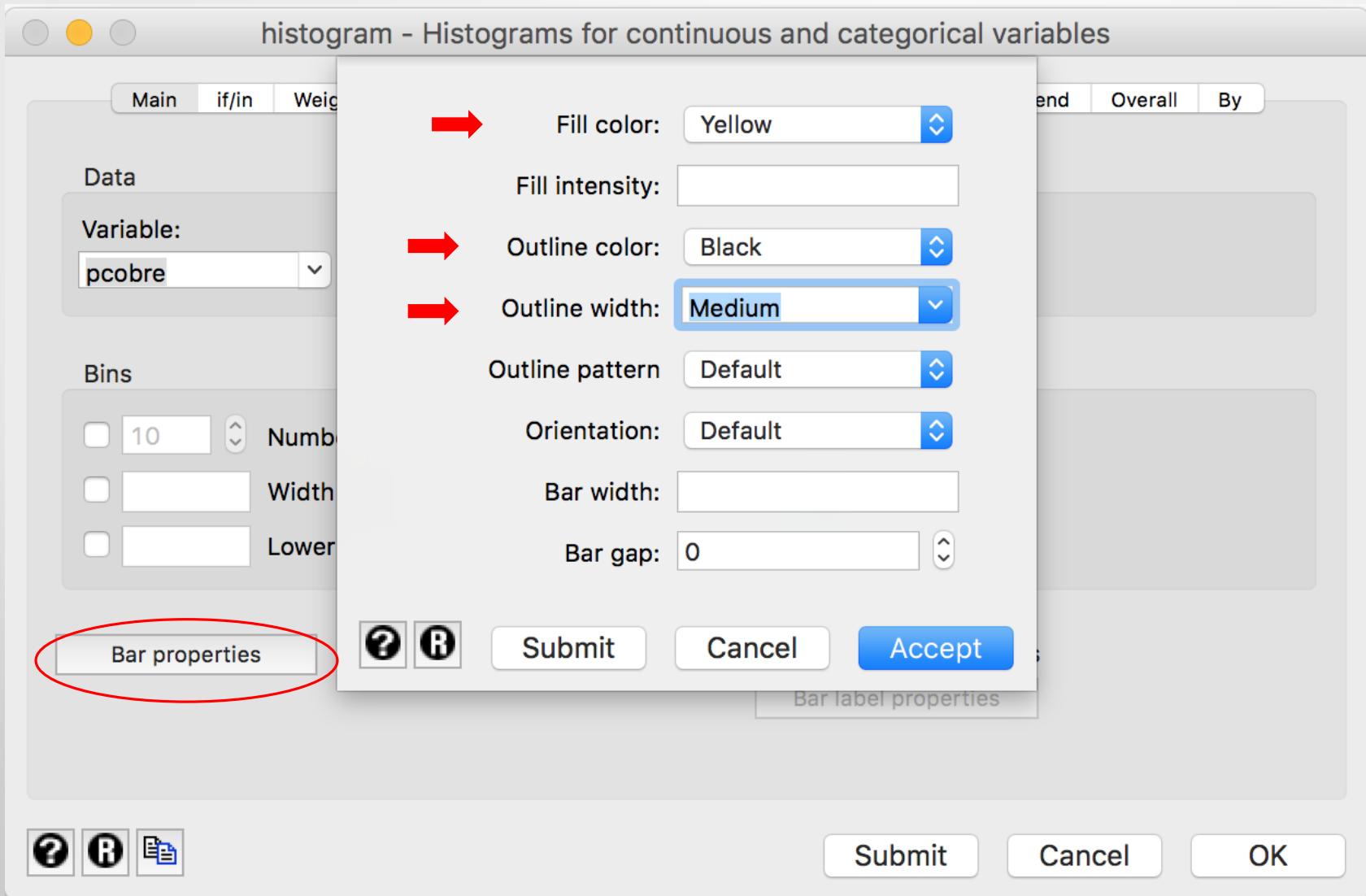
Histograma

histogram pcobre, percent



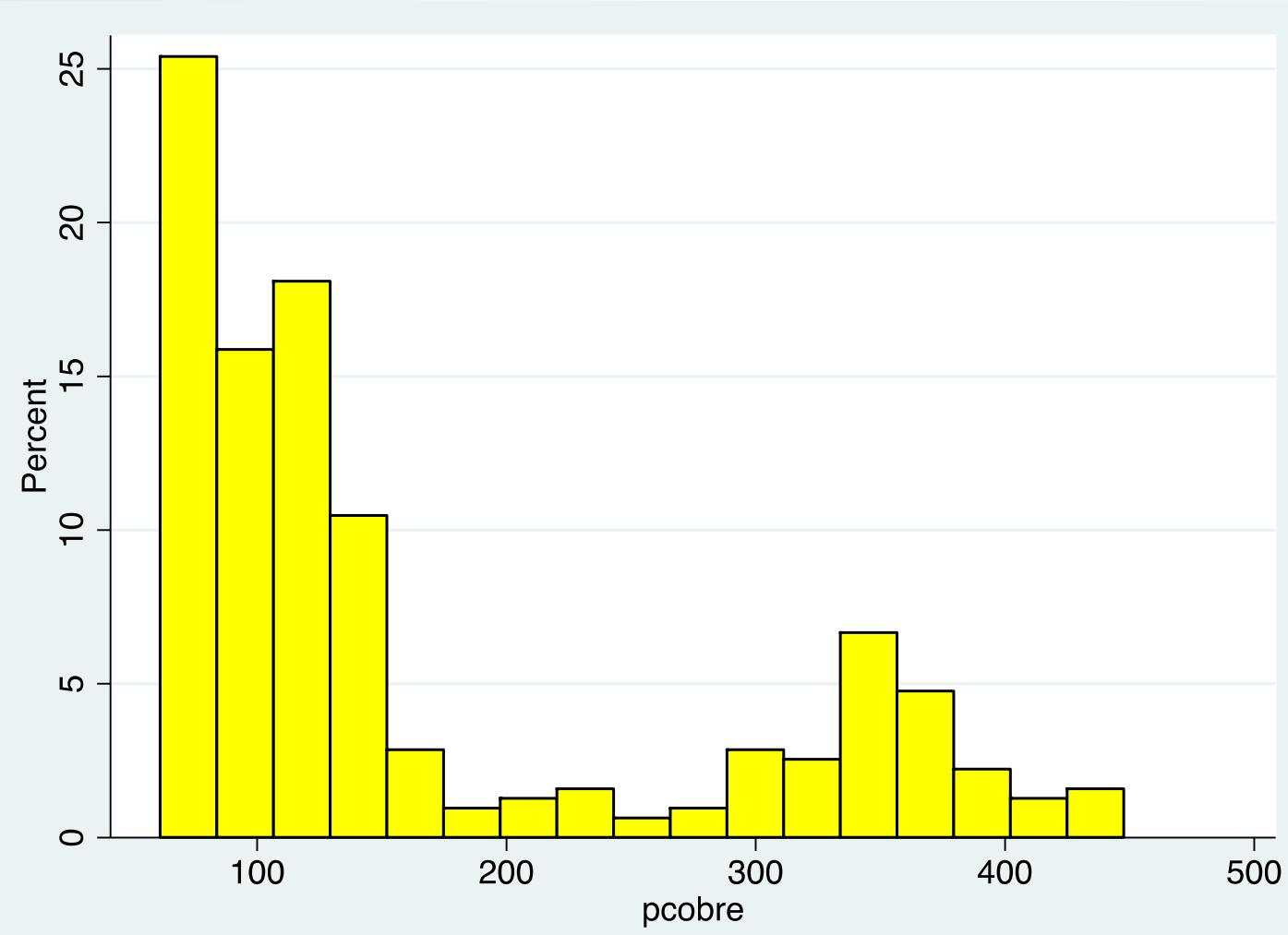
Histograma

- También se puede cambiar el color de las barras:



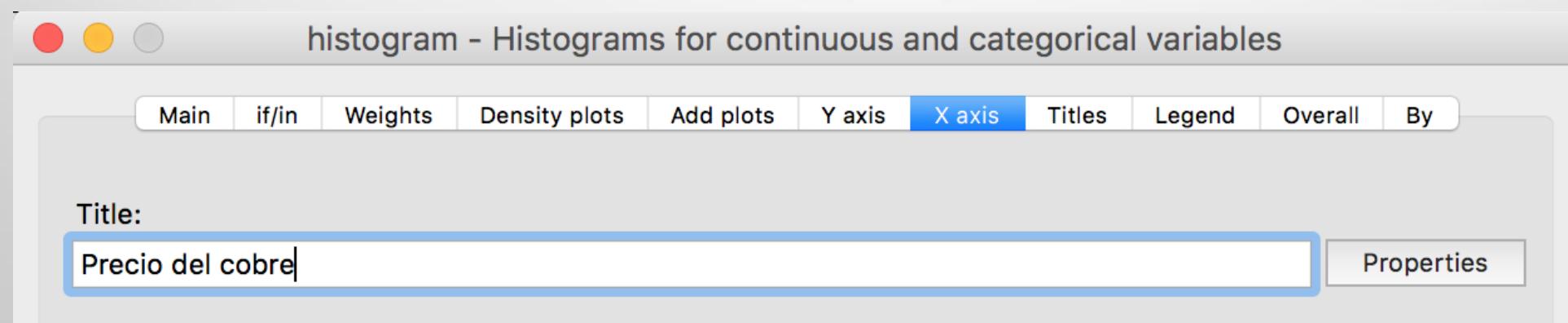
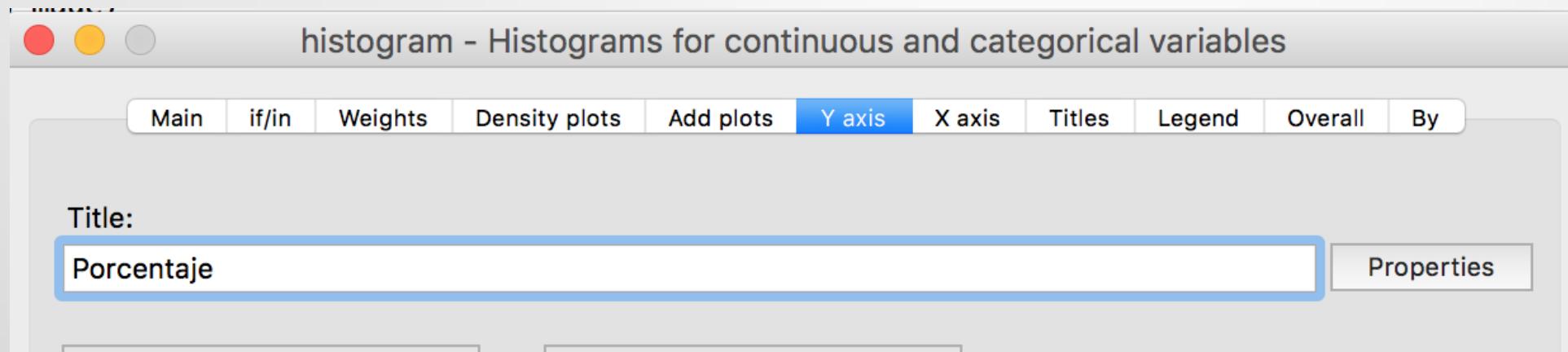
Histograma

```
histogram pcobre, percent fcolor(yellow) lcolor(black) lwidth(medium)
```

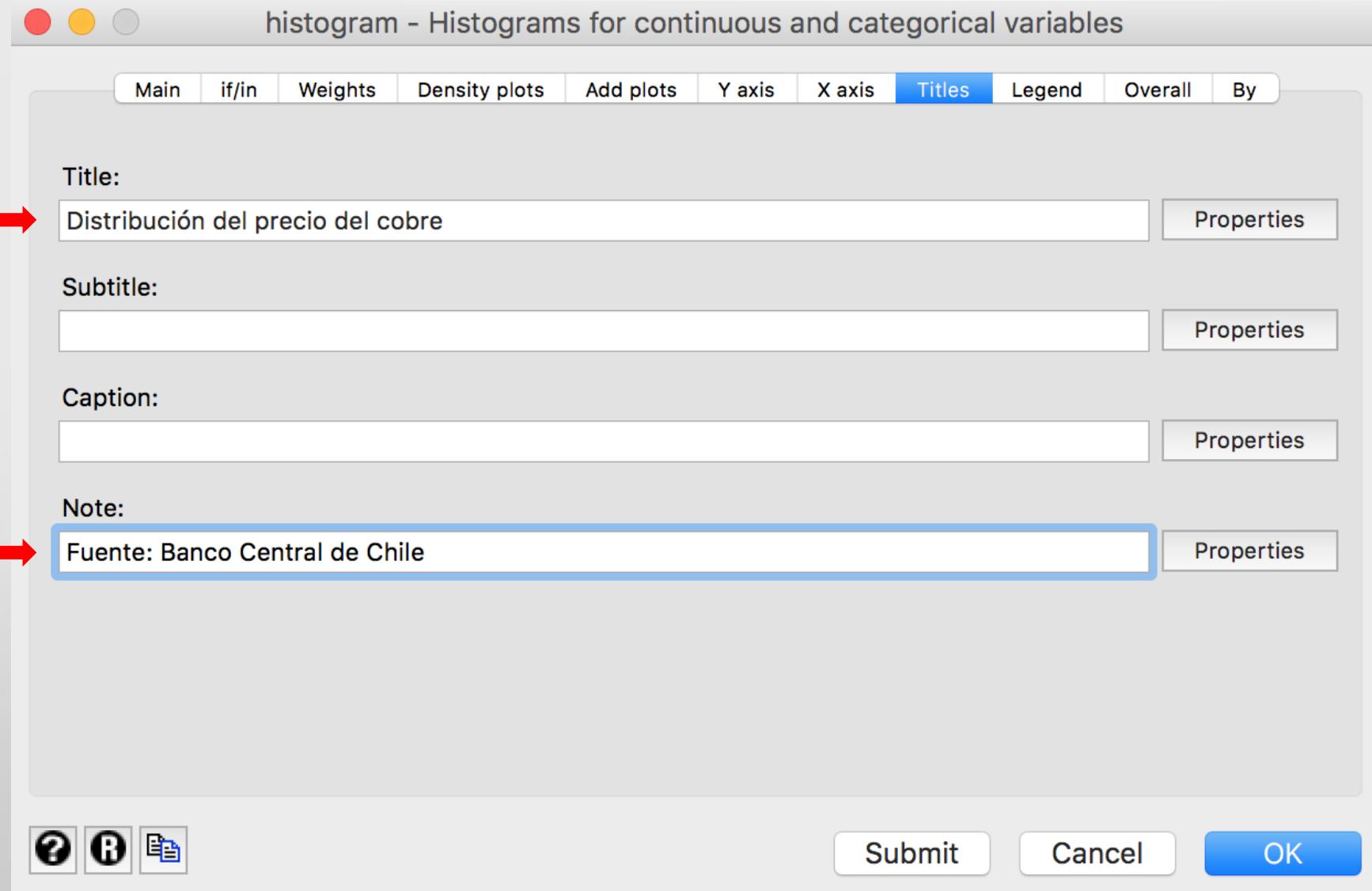


Histograma

- Se le puede cambiar el nombre a los ejes y poner título:

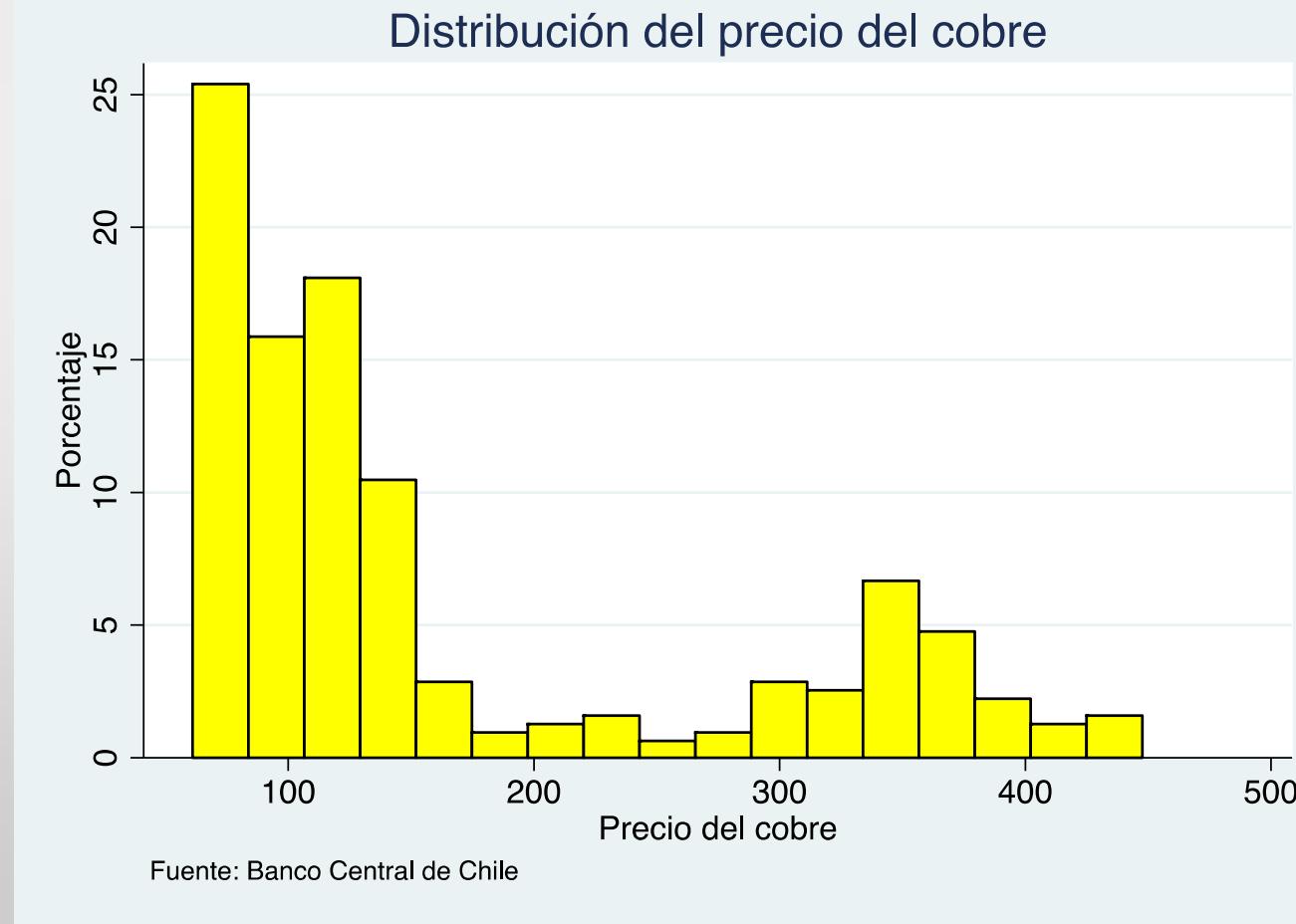


Histograma



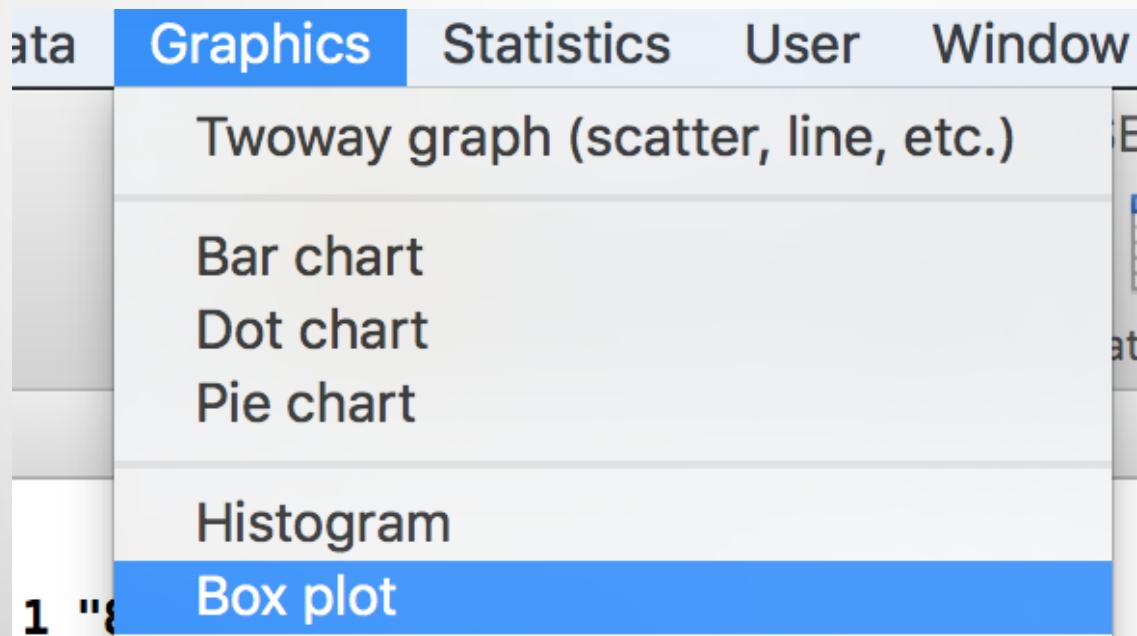
Histograma

```
histogram pcobre, percent fcolor(yellow) lcolor(black) lwidth(medium) ///
ytitle(Porcentaje) xtitle(Precio del cobre) ///
title(Distribución del precio del cobre) note(Fuente: Banco Central de Chile)
```



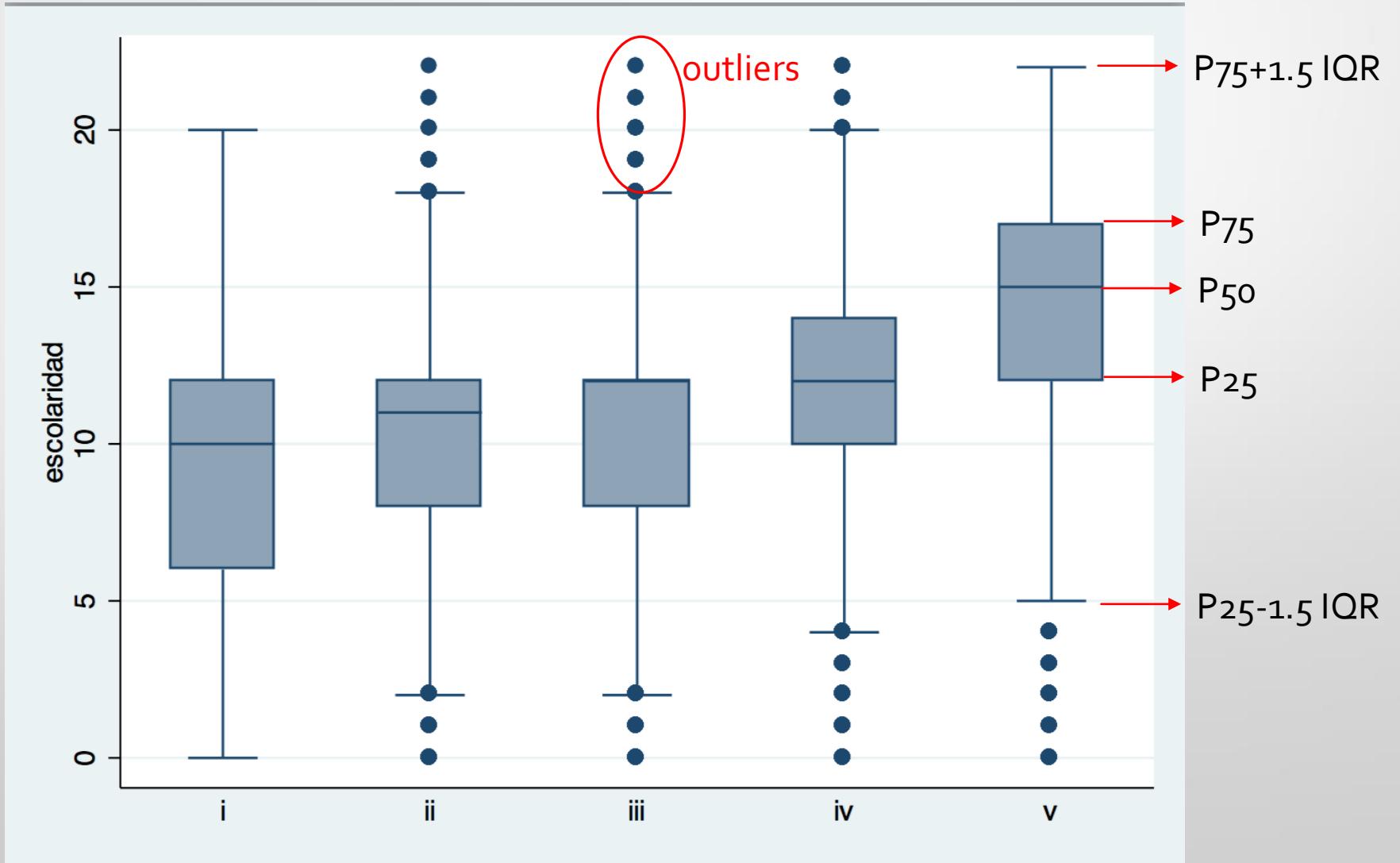
Box plot

- Este tipo de gráfico resume indicadores de tendencia central y dispersión de una variable:



Box plot

```
graph box esc [pweight = expr], over(qaut)
```





Taller Stata

Clase 8

Javiera Vásquez

Correlación entre dos variables

```
use "Casen 2017.dta", clear
```

- Suponga que estamos interesados en calcular la correlación entre el salario por hora y los años de escolaridad.
- El comando para obtener una correlación es correlate o simplemente corr:
- Primero generemos el salario por hora:

```
      storage  display    value
variable name   type    format   label      variable label
o10          double  %10.0g    o10       o10. Cuantas horas trabaja habitualmente
                                         por semana en su trabajo, negocio o ac
. sum o10

      Variable      Obs       Mean     Std. Dev.      Min       Max
           o10      92,296    42.9006    14.81883        1       220
. g horas_men=o10/7*30
(124,143 missing values generated)

. g yph=yoprcor/horas_men
(126,830 missing values generated)
```

Correlación entre dos variables

- Luego podemos calcular la correlación entre yph y esc:

```
correlate yph esc
```

```
. correlate yph esc  
(obs=89,051)
```

	yph	esc
yph	1.0000	
esc	0.2082	1.0000

- La correlación es positiva y moderada ($\rho = 0.2082$)

Correlación entre dos variables

- Existe otro comando para calcular correlación entre dos variables:

```
pwcorr yph esc, star(0.05) obs
```

		yph	esc
yph	1.0000		
	89609		
esc	0.2082*	1.0000	
	89051	1.7e+05	

- La correlación es positiva y moderada ($\rho = 0.2082$), nos entrega el mismo resultado anterior, pero además nos indica si la correlación es significativa al 5% de significancia.

Correlación entre dos variables

- ¿Cuál es la diferencia entre `pwcorr` y `corr`?
 - La diferencia es en como tratan a los missing value.
 - Para calcular la correlación sólo necesitamos dos variables, por lo tanto al pedir una matriz de correlaciones (correlaciones entre varias variables) se debería mirar de a par de variables considerando en el cálculo de la correlación todas las observaciones disponibles para ese par de variables. Esto es lo que hace el comando `pwcorr` (pairwise correlation)
 - Si uno define la muestra de análisis como aquellas donde todo el listado de variables tiene observaciones, no se consideran en el cálculo de ninguna correlación las observaciones que tienen uno o más missing value en el listado de variables. Esto es lo que hace el comando `corr`.

Correlación entre dos variables

```
. corr esc yph edad  
(obs=89,051)
```

	esc	yph	edad
esc	1.0000		
yph	0.2082	1.0000	
edad	-0.3226	0.0274	1.0000

```
. pwcorr esc yph edad
```

	esc	yph	edad
esc	1.0000		
yph	0.2082	1.0000	
edad	-0.4170	0.0277	1.0000

```
. corr esc edad  
(obs=174,058)
```

	esc	edad
esc	1.0000	
edad	-0.4170	1.0000

```
. corr esc yph  
(obs=89,051)
```

	esc	yph
esc	1.0000	
yph	0.2082	1.0000

```
. corr edad yph  
(obs=89,609)
```

	edad	yph
edad	1.0000	
yph	0.0277	1.0000

Test de diferencia de medias de dos variables

- Se puede hacer el test de dos maneras, va depender de que es lo que queremos testear:
 - Testeando que en promedio la diferencia entre dos variables es cero (paired)
 - Testeando que la diferencia de los promedios de las dos variables es cero (unpaired)
- Suponga que tenemos como hipótesis nula que la diferencia entre el ingreso autónomo personal promedio y el ingreso autónomo per-cápita promedio es cero.
- Primero identifiquemos estas variables en la base de datos Casen 2017

. d yau*				
variable	name	storage	display	value
		type	format	label
yaut		double	%10.0g	Ingreso autonomo
yauth		double	%10.0g	Ingreso autonomo del hogar
yautcor		double	%10.0g	Ingreso autonomo corregido
yautcorh		double	%10.0g	Ingreso autonomo del hogar corregido

Test de diferencia de medias de dos variables

- Ahora, generemos el ingreso autónomo per-cápita:

```
g yautpc=yautcorh/numper
```

- Luego podemos hacer el test correspondiente:

Test de diferencia de medias de dos variables

```
. ttest yautcor==yautpc, unpaired
```

Two-sample t test with equal variances

Variable	Obs	Mean	Std. Err.	Std. Dev.	[95% Conf. Interval]	
yautcor	143754	438567.4	1917.892	727166.8	434808.3	442326.4
yautpc	266968	236808.6	744.1943	384517.4	235350	238267.2
combined	410722	307424.8	840.9132	538921.1	305776.7	309073
diff		201758.7	1734.696		198358.8	205158.7

diff = mean(yautcor) - mean(yautpc) t = 116.3078
Ho: diff = 0 degrees of freedom = 410720

Ha: diff < 0 Pr(T < t) = 1.0000 Ha: diff != 0 Pr(|T| > |t|) = 0.0000 Ha: diff > 0 Pr(T > t) = 0.0000

Hipótesis nula

p-value

Valor del estadístico

Test de diferencia de medias de dos variables

- Distinto es si queremos testear que el promedio de la diferencia entre el ingreso autónomo personal y el ingreso autónomo per-cápita es igual a cero:

```
. ttest yautcor==yautpc

Paired t test

Variable      Obs       Mean     Std. Err.    Std. Dev. [95% Conf. Interval]
yautcor        143754   438567.4   1917.892   727166.8   434808.3   442326.4
yautpc        143754   289503.7   1220.309   462679.1   287112     291895.5
diff          143754   149063.6   1304.877   494742.8   146506.1   151621.2

mean(diff) = mean(yautcor - yautpc)                      t = 114.2358
Ho: mean(diff) = 0                                     degrees of freedom = 143753
Ha: mean(diff) < 0          Ha: mean(diff) != 0          Ha: mean(diff) > 0
Pr(T < t) = 1.00000          Pr(|T| > |t|) = 0.00000          Pr(T > t) = 0.00000
```

Se rechaza la hipótesis nula de que el promedio de la diferencia entre ingreso autónomo personal y el ingreso autónomo per-cápita es cero.

Test de diferencia de medias entre grupos

- En otros casos nos interesa testear para una misma variable la diferencia en el promedio entre distintos grupos, por ejemplo, testear el promedio de salario por hora de los hombres es igual al promedio de salario por hora de las mujeres. Esto se hace con el siguiente comando:

```
ttest yph, by(sexo)
```

Test de diferencia de medias entre grupos

```
. ttest yph, by(sexo)

Two-sample t test with equal variances

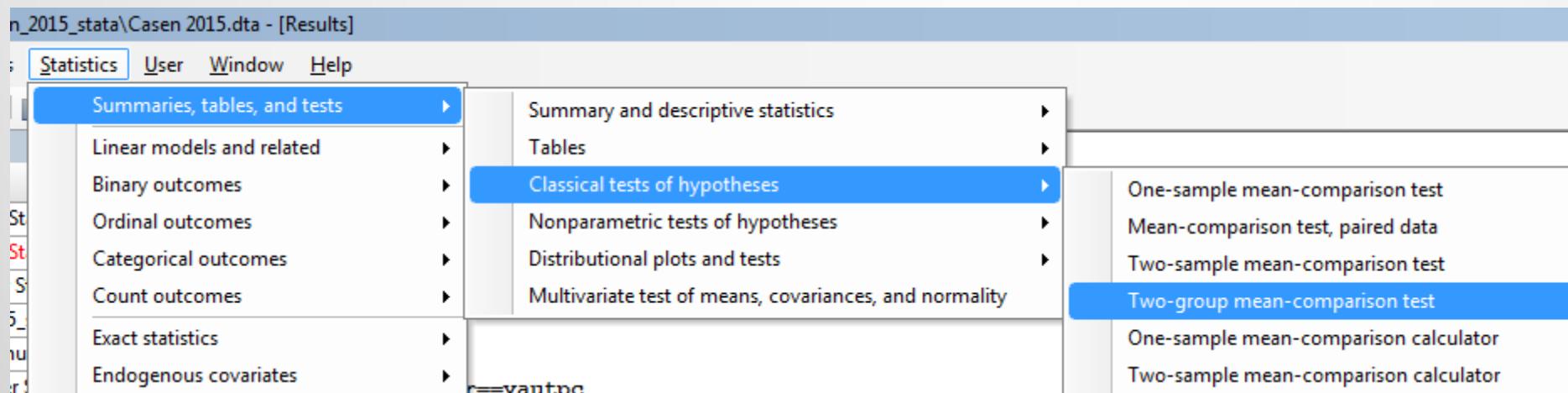
      Group          Obs        Mean    Std. Err.    Std. Dev. [95% Conf. Interval]
      hombre        62854    2730.799   23.77737   5961.153   2684.196   2777.403
      mujer         43762    2361.258   17.14746   3587.14    2327.649   2394.868
      combined      106616   2579.116   15.69525   5124.83    2548.354   2609.879
      diff           369.541   31.88631                307.0443   432.0378

      diff = mean(hombre) - mean(mujer)                      t = 11.5893
      Ho: diff = 0                                         degrees of freedom = 106614

      Ha: diff < 0             Ha: diff != 0            Ha: diff > 0
      Pr(T < t) = 1.00000     Pr(|T| > |t|) = 0.00000  Pr(T > t) = 0.00000
```

La diferencia entre el salario por hora promedio de los hombres y el salario por hora promedio de las mujeres es de 369.5 pesos, esta diferencia es estadísticamente significativa.

Test de diferencia de medias entre grupos



Test de diferencia de proporciones entre grupos

- Suponga que queremos testear si la proporción de mujeres con educación superior es igual a la proporción de hombres con educación superior.
- Primero generemos una variable binaria que tome valor 1 para las personas con educación superior completa y cero para las personas sin educación superior:

```
. label list educ
educ:
    0 sin educ. formal
    1 básica incom.
    2 básica compl.
    3 m. hum. incompleta
    4 m. téc. prof. incompleta
    5 m. hum. completa
    6 m. téc completa
    7 técnico nivel superior incompleta
    8 técnico nivel superior completo
    9 profesional incompleto
    10 postgrado incompleto
    11 profesional completo
    12 postgrado completo
    99 ns/nr
```

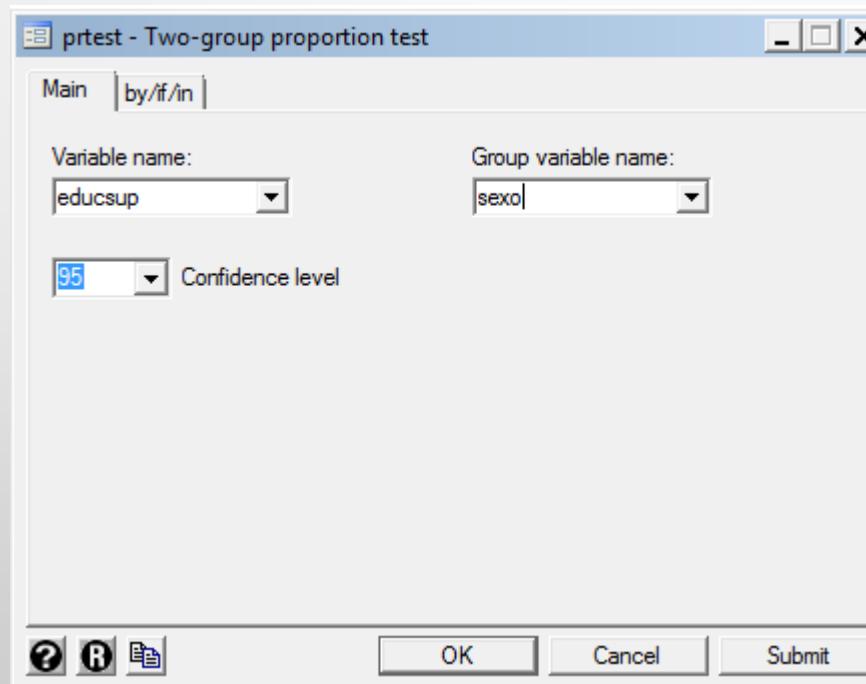
Test de diferencia de proporciones entre grupos

```
. g educsup=1 if educ==8 | educ==10 | educ==11 | educ==12  
(236503 missing values generated)

. replace educsup=0 if educsup==.  
(236503 real changes made)

. replace educsup=. if educ==.  
(0 real changes made)

. replace educsup=. if educ==99  
(403 real changes made, 403 to missing)
```



Test de diferencia de proporciones entre grupos

```
. prtest educsup, by(sexo)

Two-sample test of proportions                                         hombre: Number of obs = 127425
                                                               mujer: Number of obs = 139140

Variable      Mean    Std. Err.      z     P>|z|    [95% Conf. Interval]
-----+-----+-----+-----+-----+-----+
hombre       .1086992   .000872          .1069902   .1104083
mujer        .1194049   .0008693          .1177011   .1211087
-----+-----+-----+-----+-----+
diff         -.0107057   .0012313          -.0131189  -.0082924
under Ho:    .0012337   -8.68    0.000
-----+-----+-----+-----+-----+
diff = prop(hombre) - prop(mujer)                                     z = -8.6780
Ho: diff = 0

Ha: diff < 0              Ha: diff != 0             Ha: diff > 0
Pr(Z < z) = 0.0000          Pr(|Z| < |z|) = 0.0000  Pr(Z > z) = 1.0000
```

La proporción de hombres con educación superior completa es menor a la proporción de mujeres con educación superior completa, la diferencia es de 1 punto porcentual y es estadísticamente significativa.

Modelo de regresión lineal

- Supongamos la relación entre salario por hora y años de escolaridad.
- La correlación nos muestra que existe una relación positiva entre ambas variables.
- Pero, si queremos ver cuánto afecta un año más de escolaridad al salario, es decir, el retorno a la educación, no nos basta con la correlación.
- Debemos estimar un modelo de regresión lineal de la forma:

$$yph = \beta_0 + \beta_1 \cdot esc + u$$

Modelo de regresión lineal

- En este modelo lineal, β_1 mide el efecto marginal de un año más de escolaridad sobre el valor esperado del salario por hora:

$$E[yph] = \beta_0 + \beta_1 \cdot esc$$

$$\frac{\Delta E[yph]}{\Delta esc} = \beta_1$$

$$\Delta E[yph] = \beta_1 \cdot \underbrace{\Delta esc}_1$$

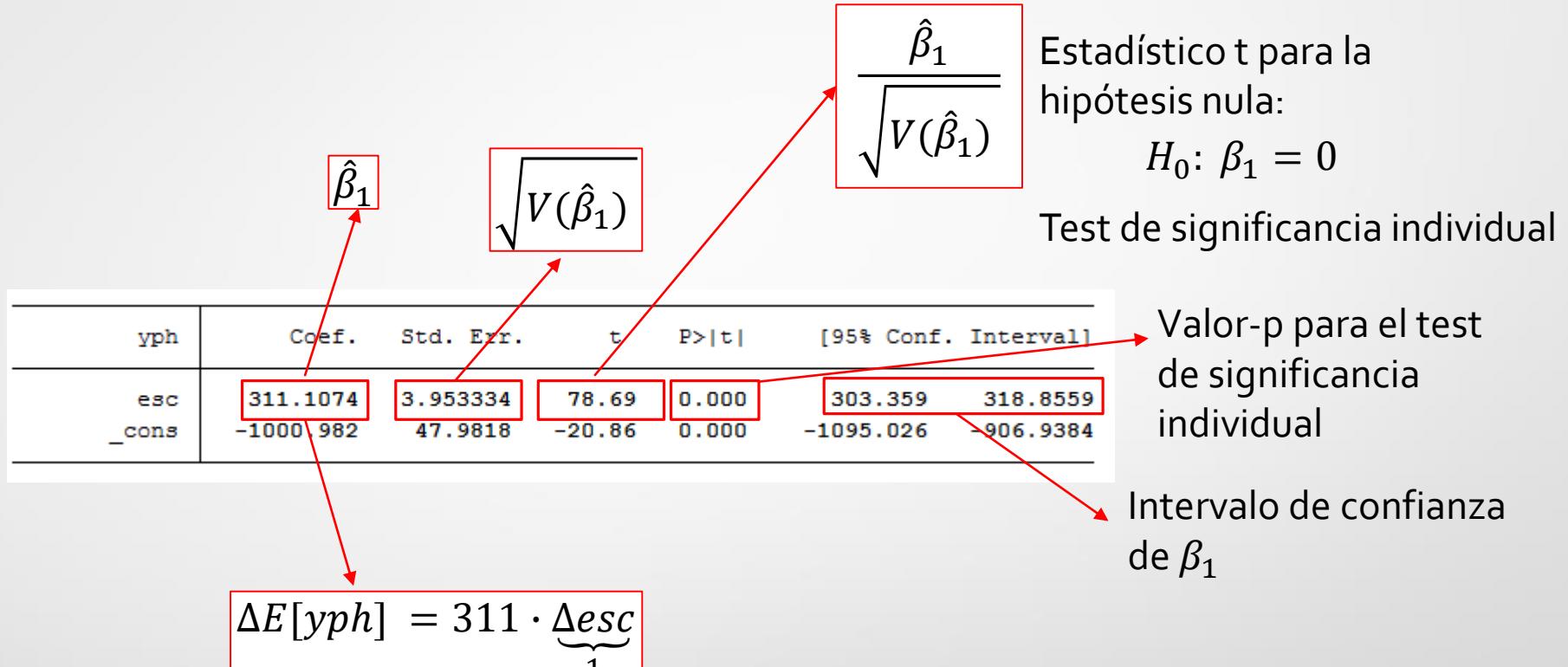
Modelo de regresión lineal

- El estimador de Mínimos Cuadrados Ordinarios (MCO) es el método más utilizado para estimar los coeficientes de la regresión lineal.
- Para estimar por MCO en STATA utilizamos el comando `regress` o simplemente `reg`.

. reg yph esc					
Source	SS	df	MS		
Model	1.5379e+11	1	1.5379e+11	Number of obs =	106412
Residual	2.6424e+12	106410	24832712.7	F(1, 106410) =	6192.89
Total	2.7962e+12	106411	26277690.6	Prob > F =	0.0000
				R-squared =	0.0550
				Adj R-squared =	0.0550
				Root MSE =	4983.2
<hr/>					
ypf	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
esc	311.1074	3.953334	78.69	0.000	303.359 318.8559
_cons	-1000.982	47.9818	-20.86	0.000	-1095.026 -906.9384

Número de observaciones
Test de significancia global
Bondad de ajuste

Modelo de regresión lineal



Un año más de escolaridad aumenta en promedio 311 pesos el salario por hora

Modelo de regresión lineal

- Toda la información de la regresión es guardada en la memoria temporal de STATA, para saber con que nombre la guarda y como, podemos usar el comando `ereturn list`:

```
. ereturn list

scalars:
    e(N) = 106412
    e(df_m) = 1
    e(df_r) = 106410
    e(F) = 6192.894577571646
    e(r2) = .0549976499343486
    e(rmse) = 4983.243191329205
    e(mss) = 153786371850.5557
    e(rss) = 2642448958825.072
    e(r2_a) = .0549887691679727
    e(l1) = -1056965.425037718
    e(l1_0) = -1059975.17484426
    e(rank) = 2

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

matrices:
    e(b) : 1 x 2
    e(V) : 2 x 2

functions:
    e(sample)
```

Modelo de regresión lineal

- Podemos ver la matriz de coeficientes y la matriz de varianzas y covarianzas de la siguiente manera:

```
. matrix list e(b)

e(b) [1,2]
            esc      _cons
y1    311.10744  -1000.9821

. matrix list e(V)

symmetric e(V) [2,2]
            esc      _cons
  esc    15.628853
_cons   -179.81758   2302.2529
```

Modelo de regresión lineal

- Podemos llamar a los coeficientes recién estimados para hacer predicciones:

```
. display _b[_cons]+_b[esc]*12  
2732.3072  
  
. display _b[_cons]+_b[esc]*17  
4287.8444
```

- Se obtiene que el salario por hora promedio predicho por el modelo para una persona con 12 años de escolaridad es de \$2.732, y el salario promedio por hora para una persona con 17 años de escolaridad es de \$4.288.

Modelo de regresión lineal

- La sintaxis general del comando es la siguiente y tiene varias opciones:

Syntax

```
regress depvar [indepvars] [if] [in] [weight] [, options]
```

options	Description
<hr/>	
Model	
<u>noconstant</u>	suppress constant term
<u>hascons</u>	has user-supplied constant
<u>tsscons</u>	compute total sum of squares with constant; seldom used
SE/Robust	
<u>vce(vcetype)</u>	vcetype may be <u>ols</u> , <u>robust</u> , <u>cluster</u> clustvar, <u>bootstrap</u> , <u>jackknife</u> , <u>hc2</u> , or <u>hc3</u>
Reporting	
<u>level(#)</u>	set confidence level; default is <u>level(95)</u>
<u>beta</u>	report standardized beta coefficients
<u>eform(string)</u>	report exponentiated coefficients and label as string
<u>depname(varname)</u>	substitute dependent variable name; programmer's option
<u>display_options</u>	control column formats, row spacing, line width, and display of omitted variables and base and empty cells
<u>noheader</u>	suppress table header
<u>notable</u>	suppress coefficient header
<u>plus</u>	make table extendable
<u>mse1</u>	force mean squared error to 1
<u>coeflegend</u>	display legend instead of statistics

Modelo de regresión lineal

- Por ejemplo, si queremos que haga el cálculo de los errores estándar de manera robusta:

```
. reg yph esc, vce(robust)

Linear regression                                         Number of obs = 106412
                                                          F( 1,106410) = 3565.61
                                                          Prob > F   = 0.0000
                                                          R-squared = 0.0550
                                                          Root MSE   = 4983.2


```

yph	Robust					
	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
esc	311.1074	5.210066	59.71	0.000	300.8958	321.3191
_cons	-1000.982	51.46091	-19.45	0.000	-1101.845	-900.1194

Modelo de regresión lineal

- O con bootstrap:

```
. reg yph esc, vce(boot)
(running regress on estimation sample)

Bootstrap replications (50)
+-----+-----+-----+-----+-----+
| 1 | 2 | 3 | 4 | 5 |
+-----+-----+-----+-----+-----+
..... 50

Linear regression                               Number of obs     =    106412
                                                Replications    =       50
                                                Wald chi2(1)   =    3832.55
                                                Prob > chi2    =    0.0000
                                                R-squared       =    0.0550
                                                Adj R-squared  =    0.0550
                                                Root MSE        = 4983.2432


```

yph	Observed	Bootstrap			Normal-based	
	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
esc	311.1074	5.02535	61.91	0.000	301.2579	320.9569
_cons	-1000.982	49.89057	-20.06	0.000	-1098.766	-903.1984

Modelo de regresión lineal

- Si queremos comparar en una sola tabla las tres estimaciones:

```
. quietly reg yph esc  
  
. estimates store ols  
  
. quietly reg yph esc, vce(robust)  
  
. estimates store robust  
  
. quietly reg yph esc, vce(boot)  
  
. estimates store boot
```

```
. estimates table ols robust boot, b(%6.1f) se(%6.1f)
```

Variable	ols	robust	boot
esc	311.1 4.0	311.1 5.2	311.1 4.5
_cons	-1001.0 48.0	-1001.0 51.5	-1001.0 44.7

legend: b/se

Modelo de regresión lineal

- Podemos cambiar el nivel de confianza del intervalo:

. reg yph esc, level(99)					
Source	SS	df	MS		
Model	1.5379e+11	1	1.5379e+11		
Residual	2.6424e+12	106410	24832712.7		
Total	2.7962e+12	106411	26277690.6		
ypb	Coef.	Std. Err.	t	P> t	[99% Conf. Interval]
esc	311.1074	3.953334	78.69	0.000	300.9241 321.2907
_cons	-1000.982	47.9818	-20.86	0.000	-1124.577 -877.3869

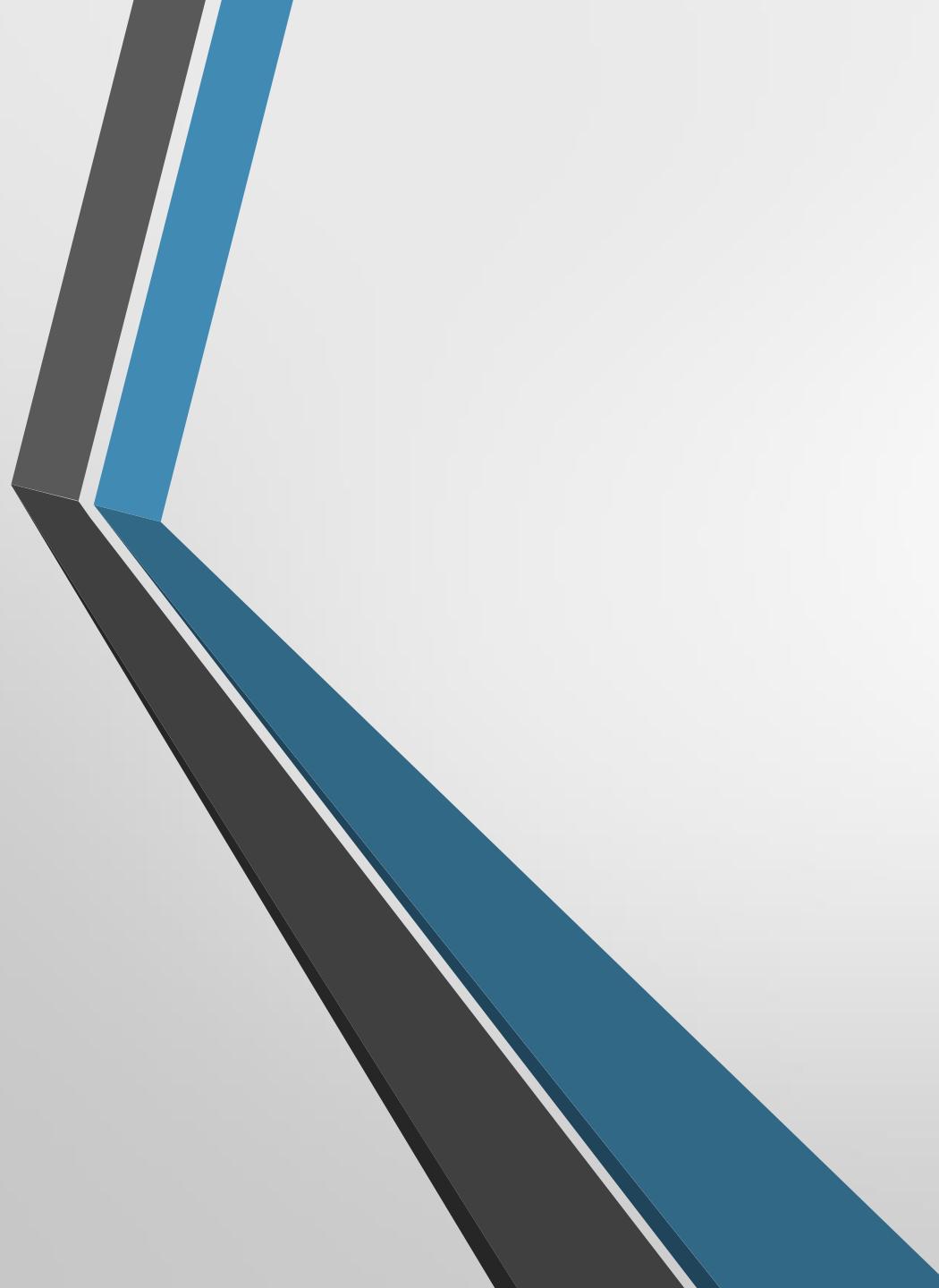
Modelo de regresión lineal

- Y si solo queremos que nos muestre la estimación de los coeficientes:

. reg yph esc, level(99) noheader						
yph	Coef.	Std. Err.	t	P> t	[99% Conf. Interval]	
esc	311.1074	3.953334	78.69	0.000	300.9241	321.2907
_cons	-1000.982	47.9818	-20.86	0.000	-1124.577	-877.3869

- Al igual que el cálculo de estadísticas descriptivas, la estimación del modelo la deberíamos hacer con factor de expansión si queremos extrapolar las conclusiones a la población:

. reg yph esc [pw=expr], noheader (sum of wgt is 7.3038e+06)						
yph	Coef.	Std. Err.	t	P> t	Robust [95% Conf. Interval]	
esc	345.917	7.340983	47.12	0.000	331.5288	360.3053
_cons	-1359.134	73.58668	-18.47	0.000	-1503.363	-1214.905



Taller Stata

Clase 9

Javiera Vásquez

Modelo de regresión lineal

- Sigamos con el ejemplo de la clase anterior:

```
use "Casen 2017.dta", clear
```

```
. d o10

      storage  display    value
variable name   type   format   label   variable label
────────────────────────────────────────────────────────────────
o10          int    %8.0g    o10    o10. ¿cuántas horas trabaja habitualmente por semana en su trabajo.

. sum o10

      Variable |       Obs        Mean     Std. Dev.      Min      Max
────────────────────────────────────────────────────────────────────────
      o10 |    110491    43.20706    14.42574         1      120

. g horas_men=o10/7*30
(156477 missing values generated)

. g yph=yoprcor/horas_men
(160352 missing values generated)
```

Modelo de regresión lineal

- Como se mostraba la clase anterior, una vez estimado el modelo podemos hacer predicciones sobre el valor esperado de la variable dependiente, en función de distintos valores de la(s) variable(s) explicativa(s).
- Si bien esto se puede calcular “manualmente” con el comando `display`, también se puede utilizar el comando `margins`.
- Después de hacer la siguiente regresión:

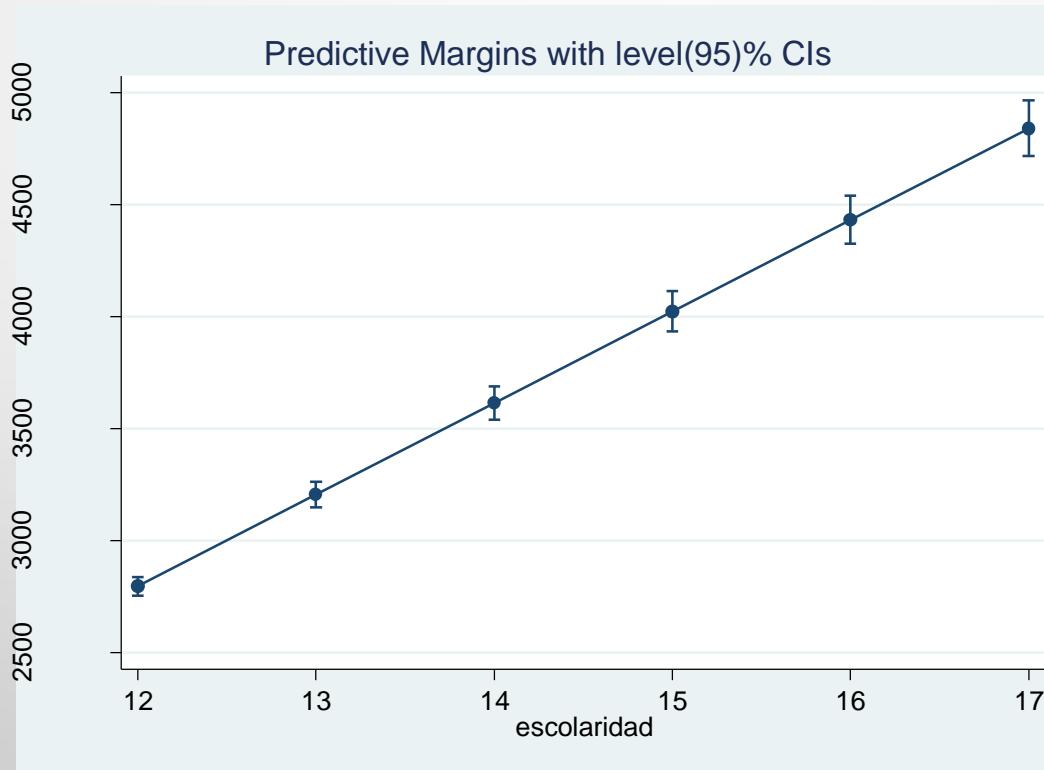
Linear regression						
	Robust					
ypf	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
esc	409.2373	9.141968	44.76	0.000	391.3192	427.1554
edad	64.78335	10.10943	6.41	0.000	44.96902	84.59769
edadc	-.1763659	.1240176	-1.42	0.155	-.4194387	.0667068
2.sexo	-718.2149	42.61828	-16.85	0.000	-801.7461	-634.6836
_cons	-4188.396	208.3265	-20.10	0.000	-4596.713	-3780.079

Modelo de regresión lineal

. margins, at(esc=(12(1)17))						
Predictive margins				Number of obs = 106412		
Model VCE : Robust						
Expression : Linear prediction, predict()						
1._at	: esc	=	12			
2._at	: esc	=	13			
3._at	: esc	=	14			
4._at	: esc	=	15			
5._at	: esc	=	16			
6._at	: esc	=	17			
Delta-method						
	Margin	Std. Err.	z	P> z	[95% Conf. Interval]	
_at						
1	2795.734	21.75347	128.52	0.000	2753.098	2838.37
2	3204.972	29.15195	109.94	0.000	3147.835	3262.108
3	3614.209	37.33109	96.81	0.000	3541.041	3687.377
4	4023.446	45.87522	87.70	0.000	3933.532	4113.36
5	4432.684	54.61329	81.16	0.000	4325.643	4539.724
6	4841.921	63.46526	76.29	0.000	4717.531	4966.31

Modelo de regresión lineal

- Se puede hacer un gráfico con estos efectos marginales, usando el comando marginsplot luego de haber ocupado el comando margins:



Modelo de regresión lineal

- Una vez estimado el modelo, también se puede utilizar el comando predict, para obtener el valor estimado de la variable dependiente según el modelo o para obtener los residuos:

```
. reg yph esc [pw=expr]
(sum of wgt is 7.3038e+06)

Linear regression                               Number of obs = 106412
                                                F( 1,106410) = 2220.42
                                                Prob > F    = 0.0000
                                                R-squared   = 0.0599
                                                Root MSE    = 5187.4


```

yph	Robust					
	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
esc	345.917	7.340983	47.12	0.000	331.5288	360.3053
_cons	-1359.134	73.58668	-18.47	0.000	-1503.363	-1214.905

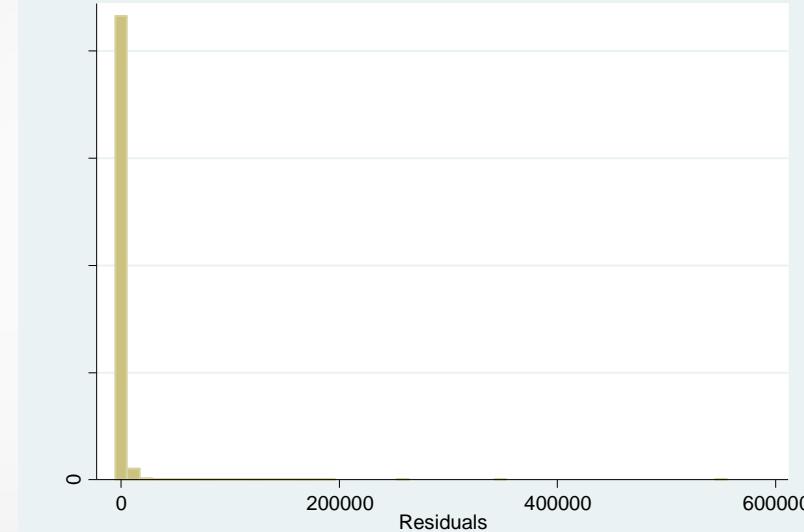
```
. predict ygorro, xb
(54626 missing values generated)

. predict errores, resid
(160556 missing values generated)
```

Modelo de regresión lineal

	yph	esc	ygorro	errores
1	.	9	1754.119	.
2	.	10	2100.036	.
3	.	16	4175.539	.
4	.	15	3829.622	.
5	.	16	4175.539	.
6	1866.667	12	2791.871	-925.204
7	1166.667	.	.	.
8	1944.444	12	2791.871	-847.4261
9	.	12	2791.871	.
10	1944.444	8	1408.202	536.2421
11	.	8	1408.202	.
12	.	9	1754.119	.
13	.	8	1408.202	.

hist errores



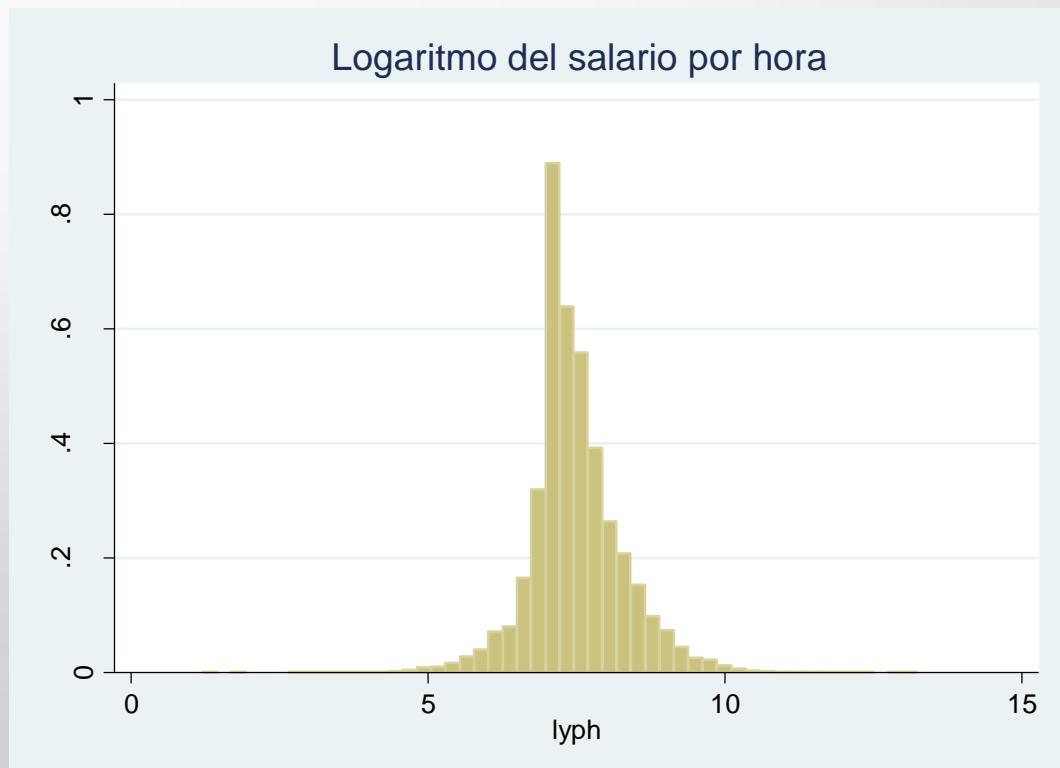
. sum errores, d				
Residuals				
Percentiles	Smallest			
1%	-3204.622	-5473.263		
5%	-2447.382	-5317.708		
10%	-2031.853	-5205.124	Obs	106412
25%	-1496.288	-5074.401	Sum of Wgt.	106412
50%	-717.7966		Mean	-42.34962
		Largest	Std. Dev.	4985.035
75%	394.7428	345478.5	Variance	2.49e+07
90%	1842.847	349283.6	Skewness	42.50878
95%	3429.637	546140.2	Kurtosis	3580.86
99%	12592.51	555478.6		

Los errores del modelo son muy asimétricos, en general, se espera que los errores sean bien comportados y se asume para inferencia que tienen una distribución normal.

Modelo de regresión lineal

- Los errores del modelo son muy asimétricos porque la variable dependiente, el salario por hora es muy asimétrica.
- En estos casos es mejor trabajar con la variable en logaritmo:

```
. g lypth=ln(ypth)  
(160352 missing values generated)
```



Modelo de regresión lineal

- Al cambiar la variable dependiente, ahora el coeficiente (efecto marginal) no tiene la misma interpretación:

Linear regression						
						Number of obs = 106412
						F(1,106410) =11442.14
						Prob > F = 0.0000
						R-squared = 0.2391
						Root MSE = .67413
lyph	Robust					
	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
esc	.0998386	.0009334	106.97	0.000	.0980093	.101668
_cons	6.359365	.0108535	585.93	0.000	6.338092	6.380638

$$\widehat{lyph} = 6,4 + 0,099 \cdot esc$$

$$\Delta \widehat{lyph} = 0,099 \cdot \Delta esc$$

$$100 \cdot \Delta \widehat{lyph} = 9,9 \cdot \Delta esc$$

$$\Delta \% \widehat{yph} = 9,9 \cdot \Delta esc$$

Un aumento de la escolaridad en 1 año aumenta en un 9,9% el salario por hora

Modelo de regresión lineal

- Podemos agregar más variables explicativas:

```
. g edadc=edad*edad

. reg lypm esc edad edadc [pw=expr]
(sum of wgt is 7.3038e+06)

Linear regression                                         Number of obs = 106412
                                                               F( 3,106408) = 4185.49
                                                               Prob > F    = 0.0000
                                                               R-squared = 0.2635
                                                               Root MSE   = .66323


```

lypm	Robust					
	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
esc	.1083155	.0010005	108.26	0.000	.1063545	.1102764
edad	.0270917	.0013773	19.67	0.000	.0243921	.0297912
edadc	-.0002121	.0000162	-13.10	0.000	-.0002438	-.0001804
_cons	5.53345	.0298264	185.52	0.000	5.474991	5.59191

Modelo de regresión lineal

- Si quisiéramos incluir como variables explicativas dummies regionales, tenemos dos opciones:

. quietly tab region, generate(DR_)			
. d DR_*			
storage	display	value	
variable name	type	format	label
DR_1	byte	%8.0g	region==región de tarapacá
DR_2	byte	%8.0g	region==región de antofagasta
DR_3	byte	%8.0g	region==región de atacama
DR_4	byte	%8.0g	region==región de coquimbo
DR_5	byte	%8.0g	region==región de valparaíso
DR_6	byte	%8.0g	region==región del libertador gral. bernardo o higgins
DR_7	byte	%8.0g	region==región del maule
DR_8	byte	%8.0g	region==región del biobío
DR_9	byte	%8.0g	region==región de la araucanía
DR_10	byte	%8.0g	region==región de los lagos
DR_11	byte	%8.0g	region==región de aysén del gral. carlos ibáñez del campo
DR_12	byte	%8.0g	region==región de magallanes y de la antártica chilena
DR_13	byte	%8.0g	region==región metropolitana de santiago
DR_14	byte	%8.0g	region==región de los ríos
DR_15	byte	%8.0g	region==región de arica y parinacota

Modelo de regresión lineal

```
. reg lypf esc edad edadc DR_1-DR_12 DR_14-DR_15 [pw=expr]
(sum of wgt is 7.3038e+06)

Linear regression                                         Number of obs = 106412
                                                               F( 17,106394) = 859.58
                                                               Prob > F    = 0.0000
                                                               R-squared = 0.2788
                                                               Root MSE   = .65637
```

lyph	Robust					
	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
esc	.1051966	.000985	106.80	0.000	.103266	.1071272
edad	.0284848	.0013666	20.84	0.000	.0258062	.0311634
edadcc	-.0002286	.0000161	-14.23	0.000	-.00026	-.0001971
DR_1	.0310776	.0186272	1.67	0.095	-.0054314	.0675866
DR_2	.1224308	.0178434	6.86	0.000	.087458	.1574035
DR_3	.0116185	.0128053	0.91	0.364	-.0134798	.0367167
DR_4	-.1599601	.0116926	-13.68	0.000	-.1828775	-.1370427
DR_5	-.1433072	.0093696	-15.29	0.000	-.1616715	-.1249429
DR_6	-.1218831	.0097709	-12.47	0.000	-.1410339	-.1027323
DR_7	-.1885267	.0105936	-17.80	0.000	-.2092901	-.1677634
DR_8	-.1739807	.008741	-19.90	0.000	-.1911129	-.1568486
DR_9	-.2587632	.0117161	-22.09	0.000	-.2817266	-.2357999
DR_10	-.1457927	.0111752	-13.05	0.000	-.167696	-.1238894
DR_11	.0497474	.0200126	2.49	0.013	.0105231	.0889718
DR_12	.0538814	.0175262	3.07	0.002	.0195303	.0882325
DR_14	-.200737	.0150388	-13.35	0.000	-.2302129	-.1712611
DR_15	-.208597	.0236473	-8.82	0.000	-.2549455	-.1622485
_cons	5.619285	.0299294	187.75	0.000	5.560624	5.677947

Modelo de regresión lineal

```
. reg lypn esc edad edadc i.region [pw=expr]  
(sum of wgt is 7.3038e+06)
```

Linear regression

Number of obs = 106412
F(17,106394) = 859.58
Prob > F = 0.0000
R-squared = 0.2788
Root MSE = .65637

lyph	Robust					
	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
esc	.1051966	.000985	106.80	0.000	.103266	.1071272
edad	.0284848	.0013666	20.84	0.000	.0258062	.0311634
edadc	-.0002286	.0000161	-14.23	0.000	-.00026	-.0001971
region						
2	.0913532	.0246505	3.71	0.000	.0430386	.1396677
3	-.0194591	.0212956	-0.91	0.361	-.0611981	.0222799
4	-.1910377	.0206341	-9.26	0.000	-.2314802	-.1505952
5	-.1743848	.0194152	-8.98	0.000	-.2124383	-.1363313
6	-.1529607	.0196067	-7.80	0.000	-.1913894	-.1145319
7	-.2196043	.0200174	-10.97	0.000	-.2588382	-.1803705
8	-.2050583	.0191238	-10.72	0.000	-.2425407	-.1675759
9	-.2898408	.0206436	-14.04	0.000	-.330302	-.2493796
10	-.1768703	.0203392	-8.70	0.000	-.2167349	-.1370057
11	.0186698	.0262668	0.71	0.477	-.0328127	.0701524
12	.0228038	.0244262	0.93	0.351	-.0250712	.0706789
13	-.0310776	.0186272	-1.67	0.095	-.0675866	.0054314
14	-.2318146	.0227028	-10.21	0.000	-.2763118	-.1873174
15	-.2396746	.0291271	-8.23	0.000	-.2967633	-.1825859
_cons	5.650363	.0339564	166.40	0.000	5.583809	5.716917

En este caso no es necesario generar las dummies antes, se incluyen automáticamente en la regresión.

Por defecto deja la primera región como categoría base, si se quiere cambiar, se debe usar de la siguiente manera:
ib13.region

Modelo de regresión lineal

```
. reg lypf esc edad edadc ib13.region [pw=expr]  
(sum of wgt is 7.3038e+06)
```

Linear regression

Number of obs = 106412
F(17, 106394) = 859.58
Prob > F = 0.0000
R-squared = 0.2788
Root MSE = .65637

lyph	Robust					
	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
esc	.1051966	.000985	106.80	0.000	.103266	.1071272
edad	.0284848	.0013666	20.84	0.000	.0258062	.0311634
edadc	-.0002286	.0000161	-14.23	0.000	-.00026	-.0001971
region						
1	.0310776	.0186272	1.67	0.095	-.0054314	.0675866
2	.1224308	.0178434	6.86	0.000	.087458	.1574035
3	.0116185	.0128053	0.91	0.364	-.0134798	.0367167
4	-.1599601	.0116926	-13.68	0.000	-.1828775	-.1370427
5	-.1433072	.0093696	-15.29	0.000	-.1616715	-.1249429
6	-.1218831	.0097709	-12.47	0.000	-.1410339	-.1027323
7	-.1885267	.0105936	-17.80	0.000	-.2092901	-.1677634
8	-.1739807	.008741	-19.90	0.000	-.1911129	-.1568486
9	-.2587632	.0117161	-22.09	0.000	-.2817266	-.2357999
10	-.1457927	.0111752	-13.05	0.000	-.167696	-.1238894
11	.0497474	.0200126	2.49	0.013	.0105231	.0889718
12	.0538814	.0175262	3.07	0.002	.0195303	.0882325
14	-.200737	.0150388	-13.35	0.000	-.2302129	-.1712611
15	-.208597	.0236473	-8.82	0.000	-.2549455	-.1622485
_cons	5.619285	.0299294	187.75	0.000	5.560624	5.677947

Modelo de regresión lineal

- Si quiero estimar el retorno a la educación, efecto marginal de los años de escolaridad, diferenciado por categoría de variables, por ejemplo sexo, tengo que interactuar las variables.
- Teóricamente lo que estamos buscando es estimar el siguiente modelo:

$$lyph = \beta_0 + \beta_1 \cdot esc + \beta_2 \cdot edad + \beta_3 \cdot edad^2 + \beta_4 \cdot dmujer + \beta_5 \cdot dmujer \cdot esc + u$$

- Donde el retorno a la educación de las mujeres es:

$$\frac{\Delta E[yph]}{\Delta esc} \Big|_{mujer} = \beta_1 + \beta_5$$

Mide la diferencia en retorno a la educación entre mujeres y hombres

- Y en el caso de los hombres:

$$\frac{\Delta E[yph]}{\Delta esc} \Big|_{hombre} = \beta_1$$

Modelo de regresión lineal

- Para esto debo generar la variable dummy de género y la interacción (multiplicación) de esta dummy con la variable de años de escolaridad, pero el comando `reg` permite incorporar esto sin necesidad de generar las variables:

```
. reg lypth edad edadc i.sex0##c.esc [pw=expr]
(sum of wgt is 7.3038e+06)
```

Linear regression

Number of obs = 106412
F(5,106406) = 2639.14
Prob > F = 0.0000
R-squared = 0.2798
Root MSE = .65586

lyph	Robust					
	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
edad	.0291143	.0013665	21.31	0.000	.0264359	.0317927
edadc	-.0002366	.000016	-14.76	0.000	-.0002681	-.0002052
2.sex0	-.2586431	.0225658	-11.46	0.000	-.3028718	-.2144144
esc	.1082852	.0012666	85.49	0.000	.1058028	.1107677
sex0#c.esc						
2	.0049168	.0018813	2.61	0.009	.0012296	.0086041
_cons	5.582025	.0302121	184.76	0.000	5.522809	5.64124

Por defecto deja como categoría base `sexo==1` (hombre)

Ver todas las opciones de variables dummies e interacciones en `help fvvarlist`

Modelo de regresión lineal

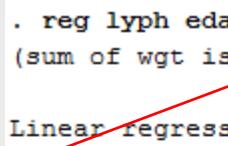
- Según el modelo estimado, el retorno a la educación de los hombres es 10.8%:
- Y el retorno a la educación de las mujeres 11.3%:

```
. display _b[esc]  
.10828524
```

```
. display _b[esc]+_b[2.sexo#c.esc]  
.11320206
```

Modelo de regresión lineal

Podemos cambiar
para que la
categoría base
sean las mujeres
(sexo==2)



Linear regression						
Number of obs = 106412						
F(5,106406) = 2639.14						
Prob > F = 0.0000						
R-squared = 0.2798						
Root MSE = .65586						
Robust						
lyph	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
edad	.0291143	.0013665	21.31	0.000	.0264359	.0317927
edadc	-.0002366	.000016	-14.76	0.000	-.0002681	-.0002052
1.sex0	.2586431	.0225658	11.46	0.000	.2144144	.3028718
esc	.1132021	.0014859	76.19	0.000	.1102898	.1161144
sex0#c.esc						
1	-.0049168	.0018813	-2.61	0.009	-.0086041	-.0012296
_cons	5.323381	.0339608	156.75	0.000	5.256819	5.389944

Modelo de regresión lineal

- También podemos utilizar el comando margins para obtener el retorno a la educación diferenciado entre hombres y mujeres:

```
. margins, dydx(esc) at(sexo=(1 2))

Average marginal effects                               Number of obs     =     106412
Model VCE      : Robust

Expression   : Linear prediction, predict()
dy/dx w.r.t. : esc

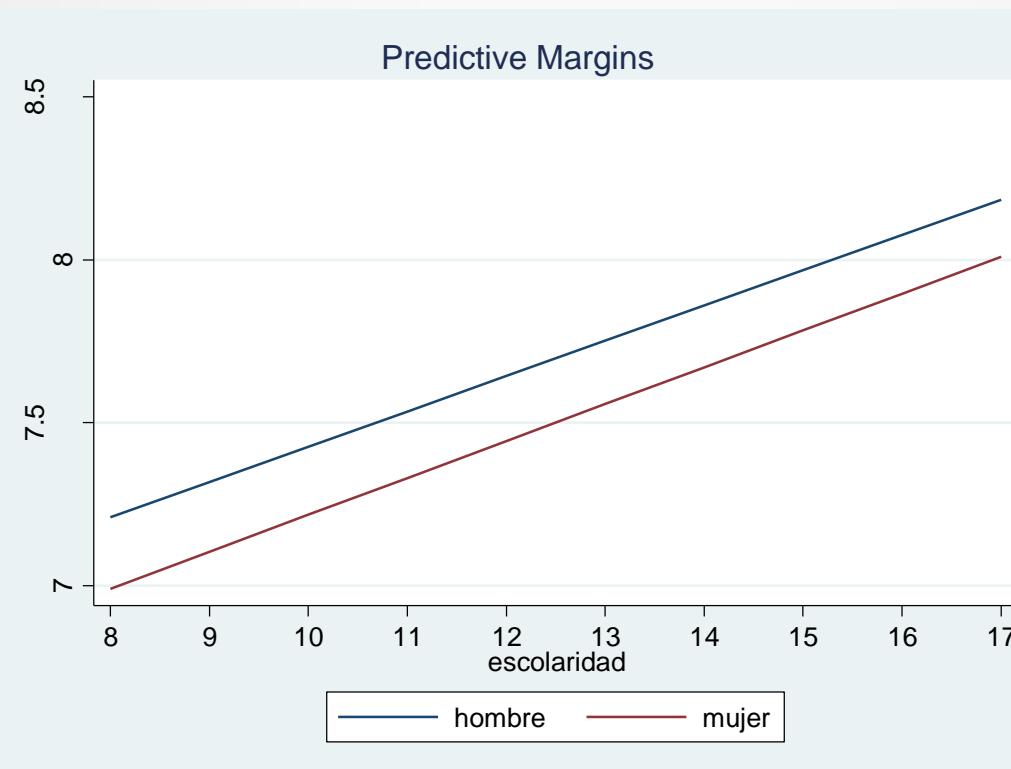
1._at        : sexo          =      1
2._at        : sexo          =      2

                                         Delta-method
                                         dy/dx  Std. Err.      z    P>|z|    [95% Conf. Interval]
-----  
esc
  _at
    1 | .1082852  .0012666    85.49    0.000    .1058028    .1107677
    2 | .1132021  .0014859    76.19    0.000    .1102898    .1161143
```

Modelo de regresión lineal

- Por ejemplo, si queremos graficar el valor predicho del logartimo del salario por hora en función de escolaridad y separando entre hombres y mujeres:

```
. quietly margins, at(esc=(8(1)17) sexo=(1 2))  
  
. marginsplot, noci x(esc) recast(line)  
  
Variables that uniquely identify margins: esc sexo
```



Modelo de regresión lineal

- Continuemos con el modelo de salario (ecuación de Mincer) usando la Casen 2015:

```
. g horas_men=o10/7*30  
(156477 missing values generated)

. g yph=yoprcor/horas_men  
(160352 missing values generated)

. sum yph

      Variable |       Obs        Mean    Std. Dev.       Min       Max
                 |   106616    2579.116     5124.83   3.333333    560000

. g lyph=ln(yph)  
(160352 missing values generated)
```

Modelo de regresión lineal

- El comando `esttab` nos permite hacer una tabla con los resultados de la estimación del modelo (similar a `estimates table`):

```
. reg lypf esc c.edad##c.edad

      Source          SS           df           MS
      Model       16089.3077      3   5363.10258
      Residual    47348.9626106408   .44497559
      Total       63438.2704106411   .596162712

      Number of obs = 106412
      F(  3,106408) = 12052.58
      Prob > F    = 0.0000
      R-squared    = 0.2536
      Adj R-squared = 0.2536
      Root MSE     = .66706

      lypf          Coef.        Std. Err.         t      P>|t|      [95% Conf. Interval]
      esc          .104571      .000563      185.74      0.000      .1034675      .1056745
      edad         .0263487     .000874      30.15      0.000      .0246357      .0280618
      c.edad#c.edad -.0002057    9.83e-06     -20.92      0.000     -.0002249     -.0001864
      _cons        5.570132     .0195937     284.28      0.000      5.531729      5.608536

. eststo
(est1 stored)
```

Modelo de regresión lineal

```
. reg lyp h esc c.edad##c.edad i.sex o
```

Source	SS	df	MS	Number of obs	=	106412
Model	17066.56	4	4266.64	F(4, 106407)	= 9790.46
Residual	46371.7104106407	.	.435795675	Prob > F	=	0.0000
Total	63438.2704106411	.	.596162712	R-squared	=	0.2690
				Adj R-squared	=	0.2690
				Root MSE	=	.66015

lyph	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
esc	.1071826	.0005599	191.43	0.000	.1060852 .10828
edad	.0281598	.0008658	32.52	0.000	.0264628 .0298567
c.edad##c.edad	-.0002279	9.74e-06	-23.39	0.000	-.000247 -.0002088
2.sex o	-.196391	.0041472	-47.35	0.000	-.2045195 -.1882624
_cons	5.588089	.0193943	288.13	0.000	5.550077 5.626102


```
. eststo  
(est2 stored)
```

Modelo de regresión lineal

```
. reg lyph c.edad##c.edad i.sex0##c.esc
```

Source	SS	df	MS	Number of obs = 106412 F(5,106406) = 7852.41 Prob > F = 0.0000 R-squared = 0.2695 Adj R-squared = 0.2695 Root MSE = .65992		
Model	17098.5745	5	3419.71491			
Residual	46339.6958106406	.	.435498899			
Total	63438.2704106411	.	.596162712			
lyph	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
edad	.0283506	.0008658	32.75	0.000	.0266536	.0300475
c.edad#c.edad	-.0002299	9.74e-06	-23.60	0.000	-.000249	-.0002108
2.sex0	-.304698	.0132951	-22.92	0.000	-.3307561	-.2786398
esc	.1035643	.000701	147.74	0.000	.1021904	.1049382
sex0#c.esc						
2	.0092649	.0010806	8.57	0.000	.0071469	.0113828
_cons	5.624355	.0198437	283.43	0.000	5.585462	5.663249

```
. eststo  
(est3 stored)
```

Modelo de regresión lineal

- Una vez hechas las estimaciones de los tres modelos, podemos generar una tabla con estos resultados y que se guarde en Word, de la siguiente manera:

```
. esttab using "tabla1.rtf", replace  
(output written to tabla1.rtf)
```

Pinchando se abre el documento Word.

	(1) lyph	(2) lyph	(3) lyph
esc	0.108*** (108.26)	0.110*** (110.68)	0.108*** (85.49)
edad	0.0271*** (19.67)	0.0290*** (21.26)	0.0291*** (21.31)
edadc	-0.000212*** (-13.10)	-0.000236*** (-14.72)	-0.000237*** (-14.76)
1.sexo		0 (.)	0 (.)
2.sexo		-0.199*** (-33.17)	-0.259*** (-11.46)
1.sexo#c.esc			0 (.)
2.sexo#c.esc			0.00492** (2.61)
_cons	5.533*** (185.52)	5.561*** (188.60)	5.582*** (184.76)
N	106412	106412	106412

t statistics in parentheses
* p < 0.05, ** p < 0.01, *** p < 0.001

Modelo de regresión lineal

- El comando `esttab` tiene muchas opciones, la sintaxis básica es:

```
esttab [ namelist ] [ using filename ] [ , options ]
```

options	description
Main	
<code>b(fmt)</code>	specify format for point estimates
<code>beta[(fmt)]</code>	display beta coefficients instead of point est's
<code>main(name [fmt])</code>	display contents of <code>e(name)</code> instead of point e's
<code>t(fmt)</code>	specify format for t-statistics
<code>abs</code>	use absolute value of t-statistics
<code>not</code>	suppress t-statistics
<code>z[(fmt)]</code>	display z-statistics (affects label only)
<code>se[(fmt)]</code>	display standard errors instead of t-statistics
<code>p[(fmt)]</code>	display p-values instead of t-statistics
<code>ci[(fmt)]</code>	display confidence intervals instead of t-stat's
<code>aux(name [fmt])</code>	display contents of <code>e(name)</code> instead of t-stat's
<code>[no]constant</code>	do not/do report the intercept
Significance stars	
<code>[no]star[(list)]</code>	do not/do report significance stars
<code>staraux</code>	attach stars to t-stat's instead of point est's
Summary statistics	
<code>r2 ar2 pr2[(fmt)]</code>	display (adjusted, pseudo) R-squared
<code>aic bic[(fmt)]</code>	display Akaike's or Schwarz's information crit.
<code>scalars(list)</code>	display any other scalars contained in <code>e()</code>
<code>sfmt(fmt [...])</code>	set format(s) for <code>scalars()</code>
<code>noobs</code>	do not display the number of observations
<code>obslast</code>	place the number of observations last

Modelo de regresión lineal

Layout	
<code>wide</code>	place point est's and t-stat's beside one another
<code>onecell</code>	combine point est's and t-stat's in a single cell
<code>[no]parentheses</code>	do not/do print parentheses around t-statistics
<code>brackets</code>	use brackets instead of parentheses
<code>[no]gaps</code>	suppress/add vertical spacing
<code>[no]lines</code>	suppress/add horizontal lines
<code>noeqlines</code>	suppress lines between equations
<code>compress</code>	reduce horizontal spacing
<code>plain</code>	produce a minimally formatted table
Labeling	
<code>label</code>	make use of variable labels
<code>interaction(str)</code>	specify interaction operator
<code>title(string)</code>	specify a title for the table
<code>mtitles[(list)]</code>	specify model titles to appear in table header
<code>nomtitles</code>	disable model titles
<code>[no]depvars</code>	do not/do use dependent variables as model titles
<code>[no]numbers</code>	do not/do print model numbers in table header
<code>coeflabels(list)</code>	specify labels for coefficients
<code>[no]notes</code>	suppress/add notes in the table footer
<code>addnotes(list)</code>	add lines at the end of the table
Document format	
<code>smcl fixed tab csv scsv rtf html tex booktabs</code>	

Modelo de regresión lineal

- Apliquemos algunas opciones a la tabla anterior:

```
. esttab using "tabla1.rtf", ar2 p(%6.4f) title("Retorno a la educación") label replace  
(output written to tabla1.rtf)
```

Retorno a la educación			
	(1) lyph	(2) lyph	(3) lyph
escolaridad	0.108*** (0.0000)	0.110*** (0.0000)	0.108*** (0.0000)
edad	0.0271*** (0.0000)	0.0290*** (0.0000)	0.0291*** (0.0000)
edadc	-0.000212*** (0.0000)	-0.000236*** (0.0000)	-0.000237*** (0.0000)
hombre		0 (.)	0 (.)
mujer		-0.199*** (0.0000)	-0.259*** (0.0000)
hombre # escolaridad			0 (.)
mujer # escolaridad			0.00492** (0.0090)
Constant	5.533*** (0.0000)	5.561 *** (0.0000)	5.582*** (0.0000)
Observations	106412	106412	106412
Adjusted R ²	0.263	0.280	0.280

p-values in parentheses

* p < 0.05, ** p < 0.01, *** p < 0.001

Modelo de regresión lineal

- Y otras opciones:

```
. esttab using "tabla1.rtf", ar2 p(%6.4f) b(%6.4f) title("Retorno a la educación") label replace drop(1.sexo 1.sexo#c.esc)  
(output written to tabla1.rtf)
```

Retorno a la educación			
	(1) lyph	(2) lyph	(3) lyph
escolaridad	0.1083*** (0.0000)	0.1103*** (0.0000)	0.1083*** (0.0000)
edad	0.0271*** (0.0000)	0.0290*** (0.0000)	0.0291*** (0.0000)
edadc	-0.0002*** (0.0000)	-0.0002*** (0.0000)	-0.0002*** (0.0000)
mujer		-0.1994*** (0.0000)	-0.2586*** (0.0000)
mujer # escolaridad			0.0049** (0.0090)
Constant	5.5335*** (0.0000)	5.5610*** (0.0000)	5.5820*** (0.0000)
Observations	106412	106412	106412
Adjusted R ²	0.263	0.280	0.280

p-values in parentheses
* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$



Taller Stata

Clase 10

Javiera Vásquez

Modelo de regresión lineal

- Continuemos con el modelo de salario (ecuación de Mincer) usando la Casen 2017:

```
. g horas_men=o10/7*30  
(156477 missing values generated)

. g yph=yoprcor/horas_men  
(160352 missing values generated)

. sum yph

      Variable |       Obs        Mean    Std. Dev.       Min       Max
                 |   106616    2579.116     5124.83   3.333333    560000

. g lyph=ln(yph)  
(160352 missing values generated)
```

Modelo de regresión lineal

```
. reg lyp h esc c.edad##c.edad i.sex o
```

Source	SS	df	MS	Number of obs	=	106412
Model	17066.56	4	4266.64	F(4,106407)	= 9790.46
Residual	46371.7104106407	.	.435795675	Prob > F	=	0.0000
Total	63438.2704106411	.	.596162712	R-squared	=	0.2690
				Adj R-squared	=	0.2690
				Root MSE	=	.66015

lyph	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
esc	.1071826	.0005599	191.43	0.000	.1060852 .10828
edad	.0281598	.0008658	32.52	0.000	.0264628 .0298567
c.edad##c.edad	-.0002279	9.74e-06	-23.39	0.000	-.000247 -.0002088
2.sex o	-.196391	.0041472	-47.35	0.000	-.2045195 -.1882624
_cons	5.588089	.0193943	288.13	0.000	5.550077 5.626102


```
. eststo  
(est2 stored)
```

Modelo de regresión lineal

```
. reg lyph c.edad##c.edad i.sex0##c.esc
```

Source	SS	df	MS	Number of obs = 106412 F(5,106406) = 7852.41 Prob > F = 0.0000 R-squared = 0.2695 Adj R-squared = 0.2695 Root MSE = .65992		
Model	17098.5745	5	3419.71491			
Residual	46339.6958106406	.	.435498899			
Total	63438.2704106411	.	.596162712			
lyph	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
edad	.0283506	.0008658	32.75	0.000	.0266536	.0300475
c.edad#c.edad	-.0002299	9.74e-06	-23.60	0.000	-.000249	-.0002108
2.sex0	-.304698	.0132951	-22.92	0.000	-.3307561	-.2786398
esc	.1035643	.000701	147.74	0.000	.1021904	.1049382
sex0#c.esc						
2	.0092649	.0010806	8.57	0.000	.0071469	.0113828
_cons	5.624355	.0198437	283.43	0.000	5.585462	5.663249

```
. eststo  
(est3 stored)
```

Modelo de regresión lineal

- Una vez hechas las estimaciones de los tres modelos, podemos generar una tabla con estos resultados y que se guarde en Word, de la siguiente manera:

```
. esttab using "tabla1.rtf", replace  
(output written to tabla1.rtf)
```

Pinchando se abre el documento Word.

	(1) lyph	(2) lyph	(3) lyph
esc	0.108*** (108.26)	0.110*** (110.68)	0.108*** (85.49)
edad	0.0271*** (19.67)	0.0290*** (21.26)	0.0291*** (21.31)
edadc	-0.000212*** (-13.10)	-0.000236*** (-14.72)	-0.000237*** (-14.76)
1.sexo		0 (.)	0 (.)
2.sexo		-0.199*** (-33.17)	-0.259*** (-11.46)
1.sexo#c.esc			0 (.)
2.sexo#c.esc			0.00492** (2.61)
_cons	5.533*** (185.52)	5.561*** (188.60)	5.582*** (184.76)
N	106412	106412	106412

t statistics in parentheses
* p < 0.05, ** p < 0.01, *** p < 0.001

Modelo de regresión lineal

- El comando `esttab` tiene muchas opciones, la sintaxis básica es:

```
esttab [ namelist ] [ using filename ] [ , options ]
```

options	description
Main	
<code>b(fmt)</code>	specify format for point estimates
<code>beta[(fmt)]</code>	display beta coefficients instead of point est's
<code>main(name [fmt])</code>	display contents of <code>e(name)</code> instead of point e's
<code>t(fmt)</code>	specify format for t-statistics
<code>abs</code>	use absolute value of t-statistics
<code>not</code>	suppress t-statistics
<code>z[(fmt)]</code>	display z-statistics (affects label only)
<code>se[(fmt)]</code>	display standard errors instead of t-statistics
<code>p[(fmt)]</code>	display p-values instead of t-statistics
<code>ci[(fmt)]</code>	display confidence intervals instead of t-stat's
<code>aux(name [fmt])</code>	display contents of <code>e(name)</code> instead of t-stat's
<code>[no]constant</code>	do not/do report the intercept
Significance stars	
<code>[no]star[(list)]</code>	do not/do report significance stars
<code>staraux</code>	attach stars to t-stat's instead of point est's
Summary statistics	
<code>r2 ar2 pr2[(fmt)]</code>	display (adjusted, pseudo) R-squared
<code>aic bic[(fmt)]</code>	display Akaike's or Schwarz's information crit.
<code>scalars(list)</code>	display any other scalars contained in <code>e()</code>
<code>sfmt(fmt [...])</code>	set format(s) for <code>scalars()</code>
<code>noobs</code>	do not display the number of observations
<code>obslast</code>	place the number of observations last

Modelo de regresión lineal

Layout	
<code>wide</code>	place point est's and t-stat's beside one another
<code>onecell</code>	combine point est's and t-stat's in a single cell
<code>[no]parentheses</code>	do not/do print parentheses around t-statistics
<code>brackets</code>	use brackets instead of parentheses
<code>[no]gaps</code>	suppress/add vertical spacing
<code>[no]lines</code>	suppress/add horizontal lines
<code>noeqlines</code>	suppress lines between equations
<code>compress</code>	reduce horizontal spacing
<code>plain</code>	produce a minimally formatted table
Labeling	
<code>label</code>	make use of variable labels
<code>interaction(str)</code>	specify interaction operator
<code>title(string)</code>	specify a title for the table
<code>mtitles[(list)]</code>	specify model titles to appear in table header
<code>nomtitles</code>	disable model titles
<code>[no]depvars</code>	do not/do use dependent variables as model titles
<code>[no]numbers</code>	do not/do print model numbers in table header
<code>coeflabels(list)</code>	specify labels for coefficients
<code>[no]notes</code>	suppress/add notes in the table footer
<code>addnotes(list)</code>	add lines at the end of the table
Document format	
<code>smcl fixed tab csv scsv rtf html tex booktabs</code>	

Modelo de regresión lineal

- Apliquemos algunas opciones a la tabla anterior:

```
. esttab using "tabla1.rtf", ar2 p(%6.4f) title("Retorno a la educación") label replace  
(output written to tabla1.rtf)
```

Retorno a la educación			
	(1) lyph	(2) lyph	(3) lyph
escolaridad	0.108*** (0.0000)	0.110*** (0.0000)	0.108*** (0.0000)
edad	0.0271*** (0.0000)	0.0290*** (0.0000)	0.0291*** (0.0000)
edadc	-0.000212*** (0.0000)	-0.000236*** (0.0000)	-0.000237*** (0.0000)
hombre		0 (.)	0 (.)
mujer		-0.199*** (0.0000)	-0.259*** (0.0000)
hombre # escolaridad			0 (.)
mujer # escolaridad			0.00492** (0.0090)
Constant	5.533*** (0.0000)	5.561 *** (0.0000)	5.582*** (0.0000)
Observations	106412	106412	106412
Adjusted R ²	0.263	0.280	0.280

p-values in parentheses

* p < 0.05, ** p < 0.01, *** p < 0.001

Modelo de regresión lineal

- Y otras opciones:

```
. esttab using "tabla1.rtf", ar2 p(%6.4f) b(%6.4f) title("Retorno a la educación") label replace drop(1.sexo 1.sexo#c.esc)  
(output written to tabla1.rtf)
```

Retorno a la educación			
	(1) lyph	(2) lyph	(3) lyph
escolaridad	0.1083*** (0.0000)	0.1103*** (0.0000)	0.1083*** (0.0000)
edad	0.0271*** (0.0000)	0.0290*** (0.0000)	0.0291*** (0.0000)
edadc	-0.0002*** (0.0000)	-0.0002*** (0.0000)	-0.0002*** (0.0000)
mujer		-0.1994*** (0.0000)	-0.2586*** (0.0000)
mujer # escolaridad			0.0049** (0.0090)
Constant	5.5335*** (0.0000)	5.5610*** (0.0000)	5.5820*** (0.0000)
Observations	106412	106412	106412
Adjusted R ²	0.263	0.280	0.280

p-values in parentheses
* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Test de hipótesis sobre los coeficientes

- Una vez estimado el modelo, también podemos hacer test de hipótesis (más allá del test de significancia individual) sobre los coeficientes:

```
. reg lypf esc edad edadc [pw=expr]
(sum of wgt is    7.3038e+06)
```

Linear regression

Number of obs = 106412
F(3, 106408) = 4185.49
Prob > F = 0.0000
R-squared = 0.2635
Root MSE = .66323

lyph	Robust					
	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
esc	.1083155	.0010005	108.26	0.000	.1063545	.1102764
edad	.0270917	.0013773	19.67	0.000	.0243921	.0297912
edadc	-.0002121	.0000162	-13.10	0.000	-.0002438	-.0001804
_cons	5.53345	.0298264	185.52	0.000	5.474991	5.59191

test (esc=5*edad)

$$(1) \quad esc - 5 * edad = 0$$

F(1,106408) = 14.90
Prob > F = 0.0001

, test (esc=0,11) (edad=0,03)

(1) esc = .11
(2) edad = .03

F(2,106408) = 3.94
Prob > F = 0.0195

Modelo de regresión lineal: post-estimación

- Una vez estimado el modelo de regresión lineal por MCO (regress) existen varios comandos post-estimación:

<u>Title</u>	
[R] regress postestimation — Postestimation tools for regress	
<u>Description</u>	
The following postestimation commands are of special interest after regress:	
Command	Description
dfbeta	DFBETA influence statistics
estat hettest	tests for heteroskedasticity
estat imtest	information matrix test
estat ovtest	Ramsey regression specification-error test for omitted variables
estat szroeter	Szroeter's rank test for heteroskedasticity
estat vif	variance inflation factors for the independent variables
acprplot	augmented component-plus-residual plot
avplot	added-variable plot
avplots	all added-variable plots in one image
cprplot	component-plus-residual plot
lvr2plot	leverage-versus-squared-residual plot
rvfplot	residual-versus-fitted plot
rvpplot	residual-versus-predictor plot

Modelo de regresión lineal: post-estimación

- Test de no linealidades omitidas:

```
. reg lypth edad edadc i.sexo##c.esc [pw=expr]
(sum of wgt is 7.3038e+06)
```

```
Linear regression                                         Number of obs = 106412
                                                               F( 5,106406) = 2639.14
                                                               Prob > F   = 0.0000
                                                               R-squared = 0.2798
                                                               Root MSE  = .65586
```

lyph	Robust					
	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
edad	.0291143	.0013665	21.31	0.000	.0264359	.0317927
edadc	-.0002366	.000016	-14.76	0.000	-.0002681	-.0002052
2.sexo	-.2586431	.0225658	-11.46	0.000	-.3028718	-.2144144
esc	.1082852	.0012666	85.49	0.000	.1058028	.1107677
sexof#c.esc						
2	.0049168	.0018813	2.61	0.009	.0012296	.0086041
_cons	5.582025	.0302121	184.76	0.000	5.522809	5.64124

```
. estat ovtest
```

```
Ramsey RESET test using powers of the fitted values of lypth
Ho: model has no omitted variables
F(3, 106403) = 1195.57
Prob > F = 0.0000
```

La hipótesis nula es que no existen no linealidades omitidas

Modelo de regresión lineal: post-estimación

- Factor de inflación de la varianza (multicolinealidad):

. estat vif	VIF	1/VIF
edad	34.71	0.028809
edadc	35.04	0.028539
2.sexo	11.37	0.087921
esc	1.82	0.550148
sexo#c.esc		
2	12.53	0.079840
Mean VIF	19.09	

Mayor a 10 indica problemas de multicolinealidad

Estimación por Máxima Verosimilitud

- Recordemos que el estimador de Máxima Verosimilitud (MV) se obtiene de:

$$\hat{\theta}_{MV} = \operatorname{Max}_{\{\theta\}} \ln[L(\theta; y)] = \operatorname{Max}_{\{\theta\}} l(\theta; y)$$

- Donde $l(\theta; y)$ es la función de verosimilitud en logaritmo para la muestra de N observaciones:

$$l(\theta; y) = \sum_{i=1}^N l_i(\theta; y_i)$$

- Donde $l_i(\theta; y_i)$ equivalente, funcionalmente, a la función de densidad.

Estimación por Máxima Verosimilitud

- Suponga que usted tiene una muestra aleatoria de 100 observaciones que son *iid* que suponemos fueron generados a partir de una función de densidad de probabilidad exponencial:

$$f(x) = \frac{1}{\theta} \cdot \exp\left(-\frac{x}{\theta}\right) \text{ con } x \geq 0 \text{ y } \theta \geq 0$$

- En este caso, la función de verosimilitud en logaritmo de las 100 observaciones es:

$$\ln L = -n \ln(\theta) - \frac{1}{\theta} \sum_{i=1}^{100} x_i \longrightarrow \hat{\theta}_{MV} = \bar{x}$$

Estimación por Máxima Verosimilitud

- Tenemos una base de datos de 100 observaciones con la variable x que tiene una distribución exponencial. (exponencial.dta)
- A continuación estimaremos por máxima verosimilitud en stata el coeficiente θ , que sabemos corresponde a la media muestral de x .
- Primero debemos definir la log-likelihood:

```
. program define exponencial
  1. args lnf theta
  2. quietly replace `lnf'=-ln(`theta')-(`theta')^(-1)*$ML_y1
  3. end
```

- Luego definimos el problema de máxima verosimilitud:

```
. ml model lf exponencial (x=)
```

Método: linearform

Estimación por Máxima Verosimilitud

- Y luego se maximiza la función de verosimilitud:

```
. ml maximize, difficult

initial:      log likelihood =    -<inf>  (could not be evaluated)
feasible:     log likelihood = -143.27817
rescale:      log likelihood = -106.29644
Iteration 0:   log likelihood = -106.29644
Iteration 1:   log likelihood = -106.10616
Iteration 2:   log likelihood = -106.10616

                                         Number of obs      =          100
                                         Wald chi2(0)      =          .
                                         Prob > chi2      =          .

Log likelihood = -106.10616
```

x	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
_cons	1.062964	.1062964	10.00	0.000	.8546272 1.271302

- El valor estimado del parámetro es 1,063, lo que equivale al promedio muestral de la variable:

```
. sum x

Variable |       Obs        Mean      Std. Dev.       Min       Max
          |      100     1.062964     .9949245     .0067555     5.572033
```

Estimación por Máxima Verosimilitud

- El modelo de regresión lineal, también puede ser estimador por MV.
- En este caso la función de verosimilitud es:

$$\ln L(\beta, \sigma^2) = \ln f(y|x\beta, \sigma^2)$$

- Definimos la función en Stata:

```
. program lfols
  1. args lnf xb lnsigma
  2. quietly replace `lnf'=ln(normalden($ML_y1, `xb', exp(`lnsigma')))
  3. end

. ml model lf lfols (xb: lyp=esc edad edadc) (lnsigma:)
```

Estimación por Máxima Verosimilitud

- Los resultados del modelo lineal estimador por MV son exactamente iguales a los de MCO:

. reg lypth esc edad edadc						
Source	SS	df	MS			
Model	16089.3077	3	5363.10258	Number of obs =	106412	
				F(3, 106408) =	12052.58	
Residual	47348.9626106408		.44497559	Prob > F =	0.0000	
				R-squared =	0.2536	
Total	63438.2704106411		.596162712	Adj R-squared =	0.2536	
				Root MSE =	.66706	
lypht	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
esc	.104571	.000563	185.74	0.000	.1034675	.1056745
edad	.0263487	.000874	30.15	0.000	.0246357	.0280618
edadcc	-.0002057	9.83e-06	-20.92	0.000	-.0002249	-.0001864
_cons	5.570132	.0195934	284.29	0.000	5.53173	5.608535

```
. ml maximize

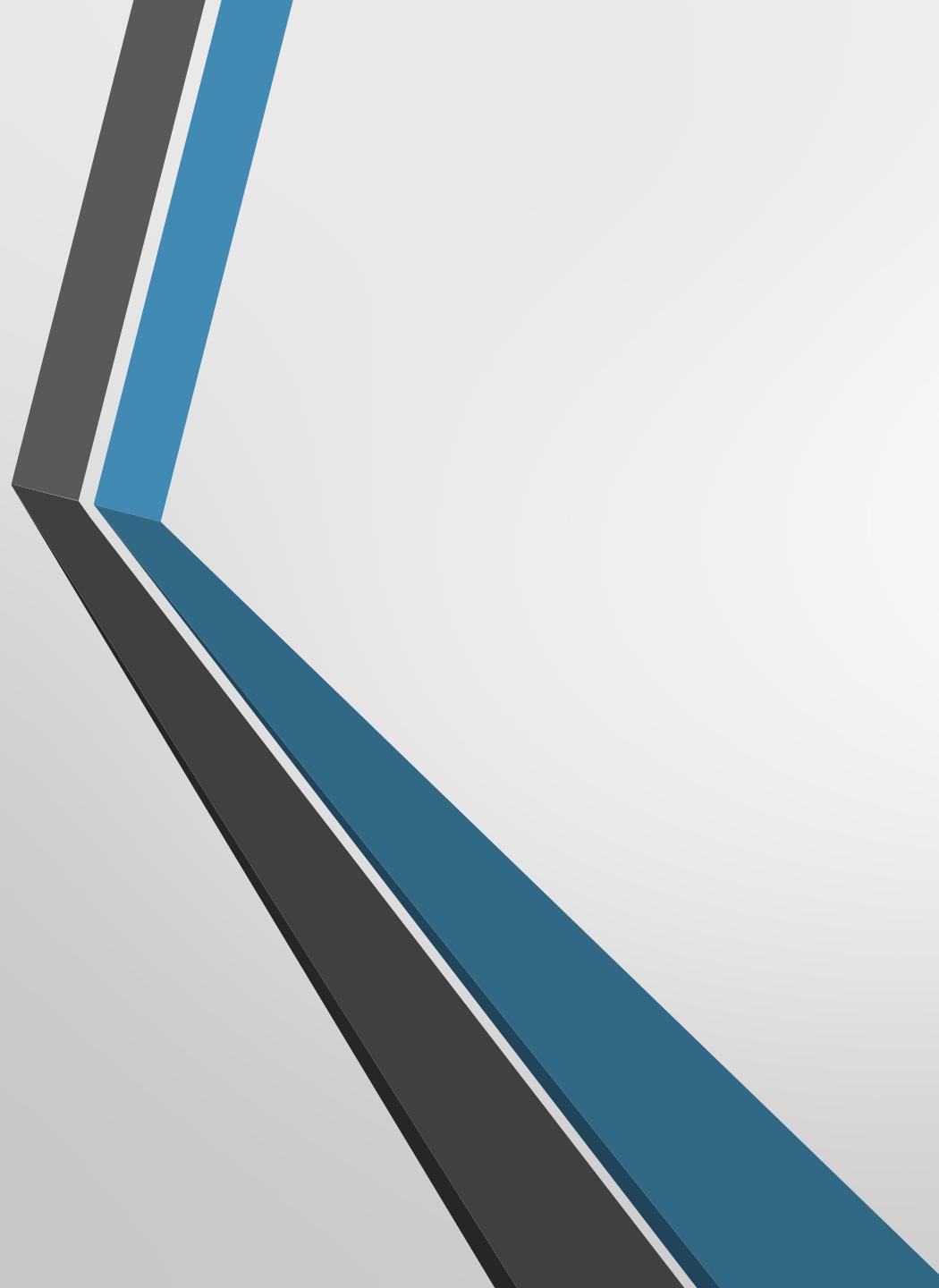
initial:      log likelihood = -3109142.7
alternative:   log likelihood = -1117225.3
rescale:      log likelihood = -340492.5
rescale eq:    log likelihood = -143165.21
Iteration 0:  log likelihood = -143165.21
Iteration 1:  log likelihood = -112649.92
Iteration 2:  log likelihood = -107947.99
Iteration 3:  log likelihood = -107907.31
Iteration 4:  log likelihood = -107907.28
Iteration 5:  log likelihood = -107907.28

Number of obs      =     106412
Wald chi2(3)      =    36159.09
Prob > chi2       =     0.0000

Log likelihood = -107907.28


```

	lypht	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
xb	esc	.104571	.000563	185.74	0.000	.1034675 .1056744
	edad	.0263487	.000874	30.15	0.000	.0246357 .0280618
	edadcc	-.0002057	9.83e-06	-20.92	0.000	-.0002249 -.0001864
	_cons	5.570132	.0195934	284.29	0.000	5.53173 5.608535
lnsigma	_cons	-.4048867	.0021677	-186.79	0.000	-.4091352 -.4006382



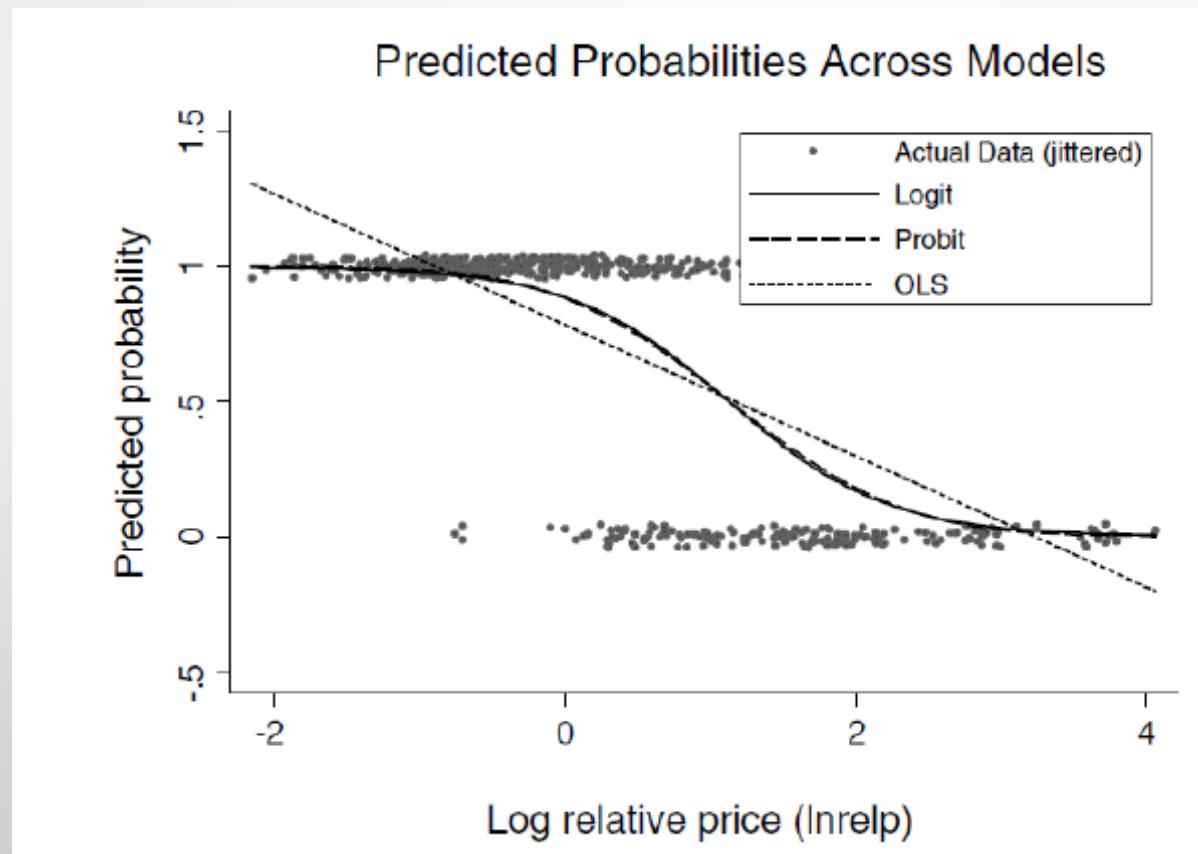
Taller Stata

Clase 11

Javiera Vásquez

Modelos no lineales: Probit-Logit

- Un ejemplo de un modelo no lineal es el modelo de variable dependiente binaria, este no puede ser estimado por MCO.



Modelos no lineales: Probit-Logit

- En este caso la variable dependiente tomar sólo dos valores: 1 y 0. Por ejemplo, cuando estamos interesados en analizar las variables que afectan la decisión de que una mujer trabaje. La variable dependiente toma valor 1 si la mujer trabaja y 0 sino trabaja.
- En este caso, el modelo se estima por máxima verosimilitud, siendo la siguiente función maximizada:

$$\ln L = n_1 \ln F(X\beta) + (n - n_1) \ln(1 - F(X\beta))$$

- Si $F()$ es normal se llama modelo probit y si es logística se llama modelo logit.

Modelos no lineales: Probit-Logit

- Suponga que estamos interesados en estudiar como los meses de lactancia materna afecta la probabilidad de que un niño sea obeso.
- Para esto utilizaremos la base de datos “base_obesidad.dta”.
- La variable dependiente es una variable binaria que toma valor 1 si el niño es obeso y 0 si es que no es obeso.
- Las variables explicativas son: la escolaridad de la madre, los meses de lactancia materna y una variable binaria que toma valor 1 si la madre tuvo diabetes gestacional.

Modelo Probit

- Se define la función de verosimilitud y se define el modelo:

```
. program lfprobit
1. args lnf xb
2. qui replace `lnf'=ln(normal(`xb')) if $ML_y1==1
3. qui replace `lnf'=ln(1-normal(`xb')) if $ML_y1==0
4. end

. ml model lf lfprobit (obeso= esc_madre meses_lechem diabetes )
```

Modelo Probit

- Luego se maximiza la función de verosimilitud para obtener los coeficientes estimados:

```
. ml maximize

initial:    log likelihood = -6338.1378
alternative: log likelihood = -5187.7041
rescale:    log likelihood = -5187.7041
Iteration 0: log likelihood = -5187.7041
Iteration 1: log likelihood = -5086.0822
Iteration 2: log likelihood = -5085.9668
Iteration 3: log likelihood = -5085.9668

                                         Number of obs      =      9144
                                         Wald chi2(3)    =       27.82
                                         Prob > chi2   =     0.0000
Log likelihood = -5085.9668

                                         [95% Conf. Interval]

obeso          Coef.    Std. Err.      z     P>|z|    [95% Conf. Interval]
esc_madre      -.0148317  .0047234    -3.14    0.002    -.0240895  -.005574
meses_lechem   -.0040178  .0014102    -2.85    0.004    -.0067818  -.0012538
diabetes        .182688   .0525013     3.48    0.001     .0797873  .2855887
_cons         -.4799554  .0614715    -7.81    0.000    -.6004373  -.3594735
```

Modelo Logit

- Se define la función de verosimilitud y se define el modelo:

```
. program lflogit  
1. args lnf xb  
2. qui replace `lnf'=ln(invlogit(`xb')) if $ML_y1==1  
3. qui replace `lnf'=ln(1-invlogit(`xb')) if $ML_y1==0  
4. end  
  
. ml model lf lflogit (obeso= esc_madre meses_lechem diabetes )
```

Modelo Logit

- Luego se maximiza la función de verosimilitud para obtener los coeficientes estimados:

```
. ml maximize

initial:    log likelihood = -6338.1378
alternative: log likelihood = -5458.9599
rescale:    log likelihood = -5112.4649
Iteration 0: log likelihood = -5112.4649
Iteration 1: log likelihood = -5086.0131
Iteration 2: log likelihood = -5085.9438
Iteration 3: log likelihood = -5085.9438

                                         Number of obs      =      9144
                                         Wald chi2(3)     =       28.12
                                         Prob > chi2     =     0.0000
Log likelihood = -5085.9438


```

obeso	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
esc_madre	-.0252189	.0080073	-3.15	0.002	-.0409128 -.009525
meses_lechem	-.0068716	.0024087	-2.85	0.004	-.0115926 -.0021507
diabetes	.3062373	.0871003	3.52	0.000	.1355238 .4769508
_cons	-.7677436	.1039795	-7.38	0.000	-.9715397 -.5639476

Modelo Probit

- La estimación del modelo probit se puede hacer directamente ocupando el comando probit de Stata:

```
. probit obeso esc_madre meses_lechem diabetes

Iteration 0:    log likelihood = -5099.8226
Iteration 1:    log likelihood = -5085.9711
Iteration 2:    log likelihood = -5085.9668
Iteration 3:    log likelihood = -5085.9668

Probit regression                                         Number of obs     =      9144
                                                               LR chi2(3)      =       27.71
                                                               Prob > chi2     =     0.0000
                                                               Pseudo R2       =     0.0027

Log likelihood = -5085.9668
```

obeso	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
esc_madre	-.0148317	.0047234	-3.14	0.002	-.0240895 -.005574
meses_lechem	-.0040178	.0014102	-2.85	0.004	-.0067818 -.0012538
diabetes	.182688	.0525013	3.48	0.001	.0797873 .2855887
_cons	-.4799554	.0614715	-7.81	0.000	-.6004373 -.3594735

Modelo Logit

- La estimación del modelo logit se puede hacer directamente ocupando el comando logit de Stata:

```
. logit obeso esc_madre meses_lechem diabetes

Iteration 0:  log likelihood = -5099.8226
Iteration 1:  log likelihood = -5085.9837
Iteration 2:  log likelihood = -5085.9438
Iteration 3:  log likelihood = -5085.9438

Logistic regression                                         Number of obs     =      9144
                                                               LR chi2(3)      =      27.76
                                                               Prob > chi2    =     0.0000
                                                               Pseudo R2       =     0.0027

Log likelihood = -5085.9438


```

obeso	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
esc_madre	-.0252189	.0080073	-3.15	0.002	-.0409128 -.009525
meses_lechem	-.0068716	.0024087	-2.85	0.004	-.0115926 -.0021507
diabetes	.3062373	.0871003	3.52	0.000	.1355238 .4769508
_cons	-.7677436	.1039795	-7.38	0.000	-.9715397 -.5639476

Efectos marginales de un probit/logit

- Hay que tener presente que en un modelo de probabilidad (probit o logit) se tiene que:

$$\Pr[y = 1|X] = F(X\beta)$$

- Por lo tanto, los coeficientes que acompañan a las variables explicativas, ya NO representan los efectos marginales de la variable sobre el valor esperado de la variable dependiente, en este caso sobre la probabilidad de que y sea igual a 1.

Efectos marginales de un probit/logit

- Cuando la variable explicativa es continua, se tiene que el efecto marginal de esta variable sobre la probabilidad de que y sea igual a 1 es:

$$\frac{\Delta \Pr[y = 1|X]}{\Delta x_k} = f(\bar{X}\beta) \cdot \beta_k$$

- Y cuando la variable explicativa es binaria:

$$F(x_i' \beta) \Big|_{x_{-k} = \bar{x}, x_k = 1} - F(x_i' \beta) \Big|_{x_{-k} = \bar{x}, x_k = 0}$$

Efectos marginales de un probit/logit

- El comando margins permite obtener los efectos marginales:

```
. margins, dydx(*) atmeans

Conditional marginal effects                               Number of obs = 9144
Model VCE    : OIM

Expression   : Pr(obeso), predict()
dy/dx w.r.t. : esc_madre meses_lechem diabetes
at           : esc_madre      = 11.45636 (mean)
                  meses_lechem = 13.26444 (mean)
                  diabetes     = .0749125 (mean)

+
+-----+
|           Delta-method
|   dy/dx   Std. Err.      z   P>|z|   [95% Conf. Interval]
+-----+
| esc_madre | -.0046652  .0014854 -3.14  0.002  -.0075766 -.0017539
| meses_lechem | -.0012638  .0004435 -2.85  0.004  -.002133  -.0003945
| diabetes   | .0574634   .0165114  3.48  0.001   .0251017  .0898251
+-----+
```

- También se puede hacer con el comando mfx

```
. mfx

Marginal effects after probit
y = Pr(obeso) (predict)
= .24526023

+
+-----+
|   variable   dy/dx   Std. Err.      z   P>|z|   [  95% C.I.  ]   X
+-----+
| esc_ma~e | -.0046652  .00149 -3.14  0.002  -.007577 -.001754 11.4564
| meses_~m | -.0012638  .00044 -2.85  0.004  -.002133 -.000395 13.2644
| diabetes* | .060389   .01814  3.33  0.001   .024829  .095949 .074913
+-----+
(*) dy/dx is for discrete change of dummy variable from 0 to 1
```

Se produce una diferencia con las variables explicativas binarias. ¿Por qué?

Efectos marginales de un probit/logit

- La diferencia es que el comando margins no hace un trato diferente al calcular los efectos marginales de las variables continuas y las variables binarias, trata a todas las variables como continuas.

```
. margins, at(diabetes=(1 0))

Predictive margins                               Number of obs     =    9144
Model VCE      : OIM
Expression     : Pr(obeso), predict()

1._at        : diabetes      =       1
2._at        : diabetes      =       0

                                         Delta-method
                                         Margin   Std. Err.      z   P>|z|   [95% Conf. Interval]
-----at-----
1          .3016513   .0175158   17.22   0.000   .267321   .3359816
2          .2413222   .0046487   51.91   0.000   .2322109   .2504335
```

```
. display .30165129-.2413222
.06032909
```

Efectos marginales de un probit/logit

- Para que el comando margins reconozca las variables binarias como tal en el cálculo de los efectos marginales, estas deben ser incluidas como variables factores:

```
. probit obeso esc_madre meses_lechem i.diabetes
```

```
Iteration 0:  log likelihood = -5099.8226
Iteration 1:  log likelihood = -5085.9711
Iteration 2:  log likelihood = -5085.9668
Iteration 3:  log likelihood = -5085.9668
```

```
Probit regression                                         Number of obs     =      9144
                                                               LR chi2(3)      =      27.71
                                                               Prob > chi2    =     0.0000
Log likelihood = -5085.9668                                Pseudo R2       =     0.0027
```

obeso	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
esc_madre	-.0148317	.0047234	-3.14	0.002	-.0240895 -.005574
meses_lechem	-.0040178	.0014102	-2.85	0.004	-.0067818 -.0012538
i.diabetes	.182688	.0525013	3.48	0.001	.0797873 .2855887
_cons	-.4799554	.0614715	-7.81	0.000	-.6004373 -.3594735

Efectos marginales de un probit/logit

```
. margins, dydx(*) atmeans
```

Conditional marginal effects

Number of obs = 9144

Model VCE : OIM

Expression : Pr(obeso), predict()

dy/dx w.r.t. : esc_madre meses_lechem 1.diabetes

at : esc_madre = 11.45636 (mean)
meses_lechem = 13.26444 (mean)
0.diabetes = .9250875 (mean)
1.diabetes = .0749125 (mean)

	Delta-method					
	dy/dx	Std. Err.	z	P> z	[95% Conf. Interval]	
esc_madre	-.0046652	.0014854	-3.14	0.002	-.0075766	-.0017539
meses_lechem	-.0012638	.0004435	-2.85	0.004	-.002133	-.0003945
1.diabetes	.060389	.018143	3.33	0.001	.0248294	.0959487

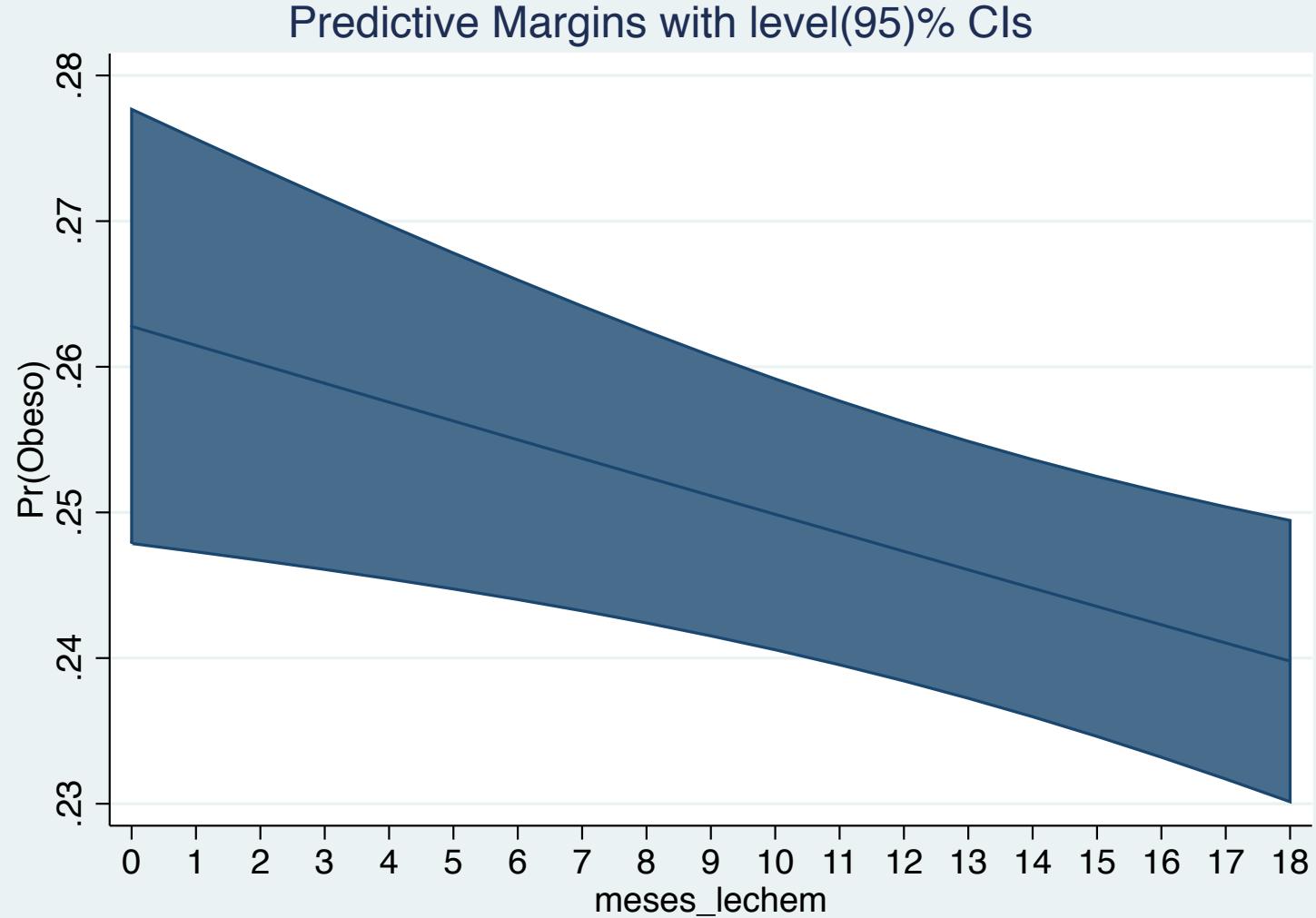
Note: dy/dx for factor levels is the discrete change from the base level.

Predicciones luego de un probit/logit

- Ahora, si queremos graficar la relación entre meses de lactancia materna y la probabilidad de que el niño sea obeso, también ocupamos el comando margins y luego el comando marginsplot.

```
. qui margins, at(meses_lechem=(0(1)18))  
  
. marginsplot, recast(line) recastci(rarea)  
  
Variables that uniquely identify margins: meses_lechem
```

Predicciones luego de un probit/logit



Comparación modelos Probit y Logit

```
. qui probit obeso esc_madre meses_lechem diabetes
```

```
. mfx
```

Marginal effects after probit

y = Pr(obeso) (predict)
= .24526023

variable	dy/dx	Std. Err.	z	P> z	[95% C.I.]	x
esc_ma~e	-.0046652	.00149	-3.14	0.002	-.007577	-.001754	11.4564	
meses_~m	-.0012638	.00044	-2.85	0.004	-.002133	-.000395	13.2644	
diabetes*	.060389	.01814	3.33	0.001	.024829	.095949	.074913	

(*) dy/dx is for discrete change of dummy variable from 0 to 1

```
. estimates store mprobit
```

Comparación modelos Probit y Logit

```
. outreg2 mprobit using "G:\FNE\FEN_Taller Stata\Tabla2.xls", replace mfx  
G:\FNE\FEN_Taller Stata\Tabla2.xls  
dir : seeout
```

A		B
1		(1)
2		obeso
3	VARIABLES	mfx dydx
4		
5		
6	esc_madre	-0.00467*** (0.00149)
7		
8	meses_lechem	-0.00126*** (0.000443)
9		
10	diabetes	0.0604*** (0.0181)
11		
12		
13	Observations	9,144
14	Standard errors in parentheses	
15	*** p<0.01, ** p<0.05, * p<0.1	
16		

Comparación modelos Probit y Logit

```
. qui logit obeso esc_madre meses_lechem diabetes

. mfx

Marginal effects after logit
    y = Pr(obeso) (predict)
    = .24510942

+-----+
| variable | dy/dx     Std. Err.      z   P>|z|   [   95% C.I.   ]   X |
+-----+
| esc_ma~e | -.0046663  .00148   -3.15  0.002  -.007568 -.001765 11.4564
| meses_~m | -.0012715  .00045   -2.85  0.004  -.002144 -.000398 13.2644
| diabetes*| .0603144   .01814    3.32  0.001   .02476  .095869 .074913
+-----+
(*) dy/dx is for discrete change of dummy variable from 0 to 1

. estimates store mlogit

. outreg2 mprobit mlogit using "G:\FNE\FEN_Taller Stata\Tabla2.xls", append mfx
G:\FNE\FEN_Taller Stata\Tabla2.xls
dir : seeout
```

Comparación modelos Probit y Logit

	A	B	C
1			
2		(1)	(2)
3		obeso	obeso
4	VARIABLES	mfx dydx	mfx dydx
5			
6	esc_madre	-0.00467*** (0.00149)	-0.00467*** (0.00148)
7			
8	meses_lechem	-0.00126*** (0.000443)	-0.00127*** (0.000445)
9			
10	diabetes	0.0604*** (0.0181)	0.0603*** (0.0181)
11			
12			
13	Observations	9,144	9,144
14	Standard errors in parentheses		
15	*** p<0.01, ** p<0.05, * p<0.1		
16			

Comparación modelos Probit y Logit

```
. clear all

. use "G:\FNE\FEN_Taller Stata\Bases de datos\base_obesidad.dta", clear

. qui probit obeso esc_madre meses_lechem diabetes

. qui mfx

. outreg2 mprobit using "G:\FNE\FEN_Taller Stata\Table2.xls", replace mfx label stats(coef pval) ctitle(Efectos marginales, Probit)
G:\FNE\FEN_Taller Stata\Table2.xls
dir : seeout

. qui logit obeso esc_madre meses_lechem diabetes

. qui mfx

. outreg2 mprobit mlogit using "G:\FNE\FEN_Taller Stata\Table2.xls", append mfx label stats(coef pval) ctitle(Efectos marginales, Logit)
G:\FNE\FEN_Taller Stata\Table2.xls
dir : seeout
```

Comparación modelos Probit y Logit

	A	B	C
1			
2		(1)	(2)
3		Efectos marginales	Efectos marginales
4	VARIABLES	Probit	Logit
5			
6	obeso		
7			
8	escolaridad de la madre	-0.00467*** (0.00169)	-0.00467*** (0.00162)
9			
10	meses lactancia	-0.00126*** (0.00438)	-0.00127*** (0.00431)
11			
12	diabetes	0.0604*** (0.000873)	0.0603*** (0.000885)
13			
14			
15	Observations	9,144	9,144
16	pval in parentheses		
17	*** p<0.01, ** p<0.05, * p<0.1		
18			
19			

Bondad de ajuste en Probit y Logit

- A través del comando `estat classification` se puede obtener el número de casos que el modelo predice correctamente.

```
. estat classification, cut(0.25)

Probit model for obeso

          True
Classified |   D   ~D   Total
+-----+-----+-----+
+       | 886  2287  3173
-       | 1362  4609  5971
+-----+-----+-----+
Total    | 2248  6896  9144

Classified + if predicted Pr(D) >= .25
True D defined as obeso != 0

Sensitivity           Pr( +| D)  39.41%
Specificity           Pr( -| ~D)  66.84%
Positive predictive value  Pr( D| +)  27.92%
Negative predictive value  Pr(~D| -) 77.19%
False + rate for true ~D  Pr( +| ~D)  33.16%
False - rate for true D  Pr( -| D)  60.59%
False + rate for classified +  Pr(~D| +)  72.08%
False - rate for classified -  Pr( D| -)  22.81%
Correctly classified      60.09%
```

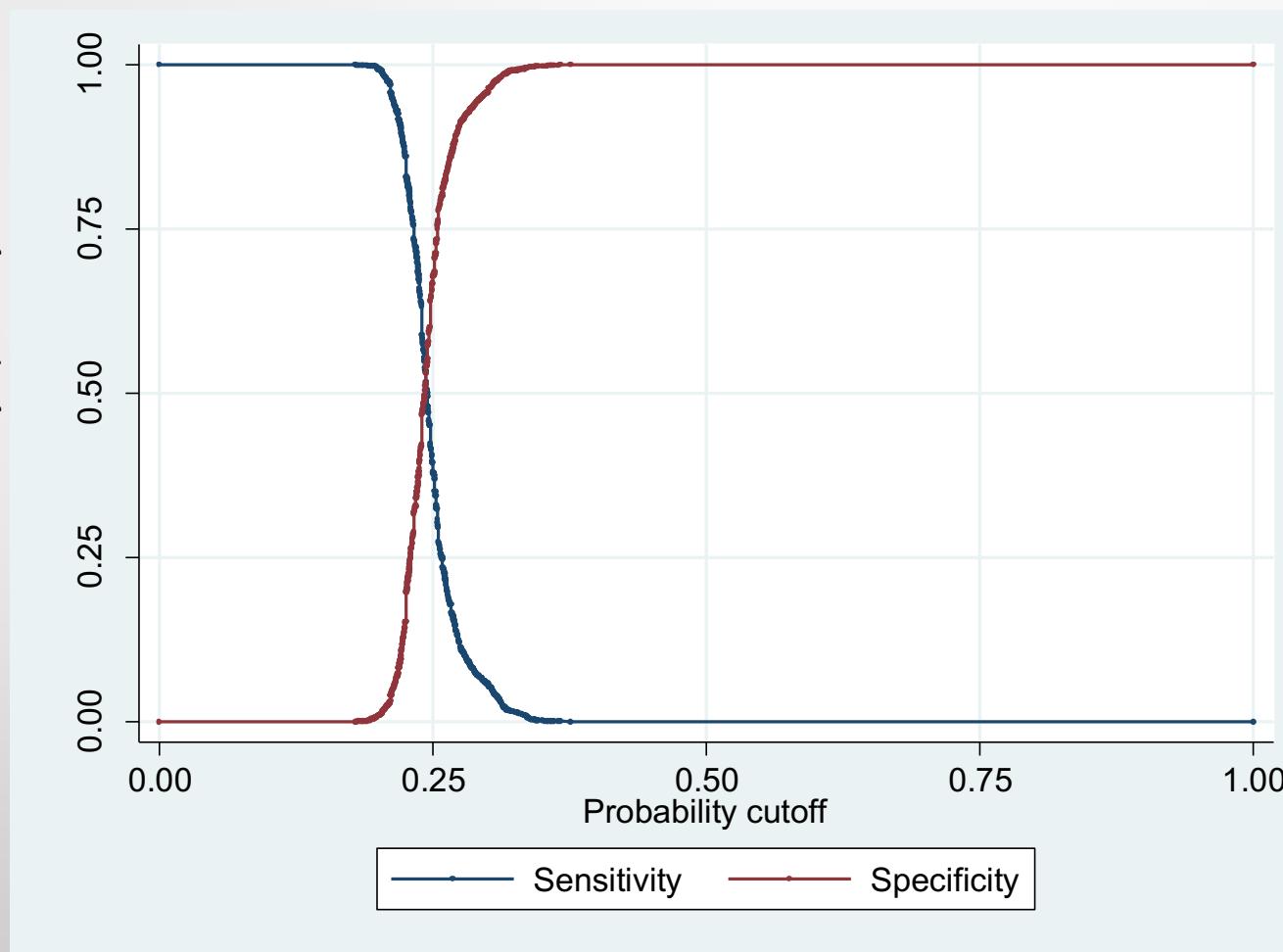
El corte de debe escoger considerando la proporción de observaciones que toman valor 1 en la variable dependiente.

obeso	Freq.	Percent	Cum.
0	7,015	75.36	75.36
1	2,294	24.64	100.00
Total	9,309	100.00	

Pero también se debe considerar que estos dos números sean lo más parecidos, para esto ocupar `lsens`.

Bondad de ajuste en Probit y Logit

- `lSENS` permite obtener el siguiente gráfico:



Bondad de ajuste en Probit y Logit

```
. estat classification, cut(0.244)

Probit model for obeso

      _____ True _____
    Classified |       D       ~D |   Total
    +-----+-----+-----+
      + |     1137     3158 | 4295
      - |     1111     3738 | 4849
    +-----+-----+-----+
      Total |     2248     6896 | 9144

Classified + if predicted Pr(D) >= .244
True D defined as obeso != 0

      _____
      Sensitivity          Pr( +| D)  50.58%
      Specificity          Pr( -| ~D)  54.21%
      Positive predictive value  Pr( D| +)  26.47%
      Negative predictive value  Pr(~D| -) 77.09%
      _____
      False + rate for true ~D  Pr( +|~D)  45.79%
      False - rate for true D  Pr( -| D)  49.42%
      False + rate for classified +  Pr(~D| +)  73.53%
      False - rate for classified -  Pr( D| -) 22.91%
      _____
      Correctly classified  53.31%
```

Uso de matrices para guardar resultados

- También se pueden definir los elementos de las matrices rescatando información de otros comandos de STATA.
- Suponga que se quiere completar la siguiente tabla de datos:

	Ingreso por hora promedio	Escolaridad promedio	Edad promedio
Hombres			
Mujeres			

- Para esto podemos definir una matriz en Stata e ir rellenando los valores

Uso de matrices para guardar resultados

- Primero definimos la matriz con el tamaño necesario, pero con puros ceros:

```
. matrix res=J(2,3,0)

. matrix list res

res[2,3]
    c1   c2   c3
r1     0   0   0
r2     0   0   0
```

- Luego se deben llenar con la información requerida

```
. sum yph if sexo==1

Variable |       Obs        Mean      Std. Dev.       Min       Max
          | 62735  2730.437  5962.874  3.333333  560000

. matrix res[1,1]=r(mean)

. matrix list res

res[2,3]
    c1       c2       c3
r1  2730.4373      0       0
r2      0       0       0
```

Uso de matrices para guardar resultados

```
. sum yph if sex==2

Variable      Obs       Mean    Std. Dev.      Min      Max
yph          43677   2360.171   3587.584   15.55556  186666.7

. matrix res[2,1]=r(mean)

. sum esc if sexo==1

Variable      Obs       Mean    Std. Dev.      Min      Max
esc          62735   11.13229   3.901555      0        22

. matrix res[1,2]=r(mean)

. sum esc if sexo==2

Variable      Obs       Mean    Std. Dev.      Min      Max
esc          43677   12.04153   3.745341      0        22

. matrix res[2,2]=r(mean)
```

Uso de matrices para guardar resultados

```
. sum edad if sexo==1
```

Variable	Obs	Mean	Std. Dev.	Min	Max
edad	62735	43.20521	14.15695	15	99

```
. matrix res[1,3]=r(mean)
```

```
. sum edad if sexo==2
```

Variable	Obs	Mean	Std. Dev.	Min	Max
edad	43677	41.67267	13.07914	15	94

```
. matrix res[2,3]=r(mean)
```

```
. matrix list res
```

```
res[2,3]
```

	c1	c2	c3
r1	2730.4373	11.132287	43.205212
r2	2360.1711	12.041532	41.672665

Nombres de filas y columnas

- Por defecto las columnas son nombradas como c₁, c₂,...etc y las filas como r₁, r₂,....etc.
- Para cambiar los nombres de las filas, usamos el comando matrix rownames:

```
. matrix rownames res=Hombres Mujeres  
  
. matrix list res  
  
res [2,3]  
          c1           c2           c3  
Hombres  2730.4373  11.132287  43.205212  
Mujeres  2360.1711  12.041532  41.672665
```

Nombres de filas y columnas

- Para cambiar los nombres de las filas, usamos el comando `matrix colnames`:

```
. matrix colnames res=Ingreso Escolaridad Edad  
  
. matrix list res  
  
res [2,3]  
          Ingreso    Escolaridad        Edad  
Hombres    2730.4373    11.132287    43.205212  
Mujeres    2360.1711    12.041532    41.672665
```

Procesos iterativos y matrices

- Suponga que queremos hacer una regresión para estimar el retorno a la educación, controlando por edad, edad al cuadrado y género, pero para cada región por separado y guardar los coeficientes del retorno a la educación y género las respectivas regresiones en una matriz.
- Primero debemos definir la matriz que guarde los resultados, la que tendrá 15 filas (una para cada región) y 2 columnas (ya que son 2 coeficientes estimados que se tienen que guardar):

```
. matrix define regR=J(15,2,0)
```

Procesos iterativos y matrices

- Luego se hace un proceso iterativo con el comando `forvalues`, para hacer una regresión para cada región e ir guardando los resultados.

```
forvalues i=1(1)15 {  
  
    qui reg lypn esc edad edadc i.sex if region==`i'  
  
    matrix define b=e(b)  
  
    matrix regR[`i',1]=b[1,1]  
    matrix regR[`i',2]=b[1,5]  
  
}
```

Procesos iterativos y matrices

- Se obtiene el siguiente resultado:

```
. matrix list regR
```

	regR[15,2]	
	c1	c2
r1	.09405477	-.20069843
r2	.08449336	-.23181355
r3	.09879548	-.26828234
r4	.09443317	-.23093731
r5	.09520497	-.19686831
r6	.08204545	-.20910367
r7	.0828229	-.20576355
r8	.10161031	-.2020016
r9	.0977957	-.13818298
r10	.08527623	-.16446349
r11	.11208721	-.1793373
r12	.09391869	-.19566929
r13	.12698749	-.20523704
r14	.09329155	-.16225086
r15	.10087113	-.04362395

Procesos iterativos y matrices

- Luego podemos poner los nombres a las filas y a las columnas:

```
. matrix rownames regR=I II III IV V VI VII VIII IX X XI XII XIII XIV XV  
  
. matrix colnames regR=ESC MUJER  
  
. matrix list regR  
  
regR[15,2]  
          ESC      MUJER  
    I   .09405477  -.20069843  
    II   .08449336  -.23181355  
    III   .09879548  -.26828234  
    IV   .09443317  -.23093731  
    V   .09520497  -.19686831  
    VI   .08204545  -.20910367  
    VII   .0828229  -.20576355  
    VIII   .10161031  -.2020016  
    IX   .0977957  -.13818298  
    X   .08527623  -.16446349  
    XI   .11208721  -.1793373  
    XII   .09391869  -.19566929  
    XIII   .12698749  -.20523704  
    XIV   .09329155  -.16225086  
    XV   .10087113  -.04362395
```

Procesos iterativos y matrices

- Esta matriz la podemos pasar a Excel directamente seleccionando, copiando como tabla y pegando en Excel:

	ESC	MUJER
I	.09405477	-.20069843
II	.08449336	-.23181355
III	.09879548	-.26828234
IV	.09443317	-.23093731
V	.09520497	-.19686831
VI	.08204545	-.20910367
VII	.0828229	-.20576355
VIII	.10161031	-.2020016
IX	.0977957	-.13818298
X	.08527623	-.16446349
XI	.11208721	-.1793373
XII	.09391869	-.19566929
XIII	.12698749	-.20523704
XIV	.09329155	-.16225086
XV	.10087113	-.043
.		

Copy
Copy Table
Copy Table as HTML
Copy as Picture
Select All Ctrl+A
Preferences...
Font...
Print...

Procesos iterativos y matrices

- Y también podemos transformar la matriz en una base de datos en Stata, esto se hace con el comando:

```
Create variables from matrix
```

```
svmat [type] A [, names(col|eqcol|matcol|string) ]
```

Con esto limpiamos la Casen 2015, para luego crear las variables a partir de la matriz, sino limpiamos quedaran mezcladas estas nuevas variables con la base que estaba cargada antes

```
. clear
```

```
. svmat regR
```

```
number of observations will be reset to 15
```

```
Press any key to continue, or Break to abort
```

```
obs was 0, now 15
```

Procesos iterativos y matrices

- Así queda la base de datos en STATA:

	regR1	regR2	
1	.0940548	-.2006984	
2	.0844934	-.2318135	
3	.0987955	-.2682824	
4	.0944332	-.2309373	
5	.095205	-.1968683	
6	.0820455	-.2091037	
7	.0828229	-.2057635	
8	.1016103	-.2020016	
9	.0977957	-.138183	
10	.0852762	-.1644635	
11	.1120872	-.1793373	
12	.0939187	-.1956693	
13	.1269875	-.205237	
14	.0932915	-.1622509	
15	.1008711	-.043624	

Procesos iterativos y matrices

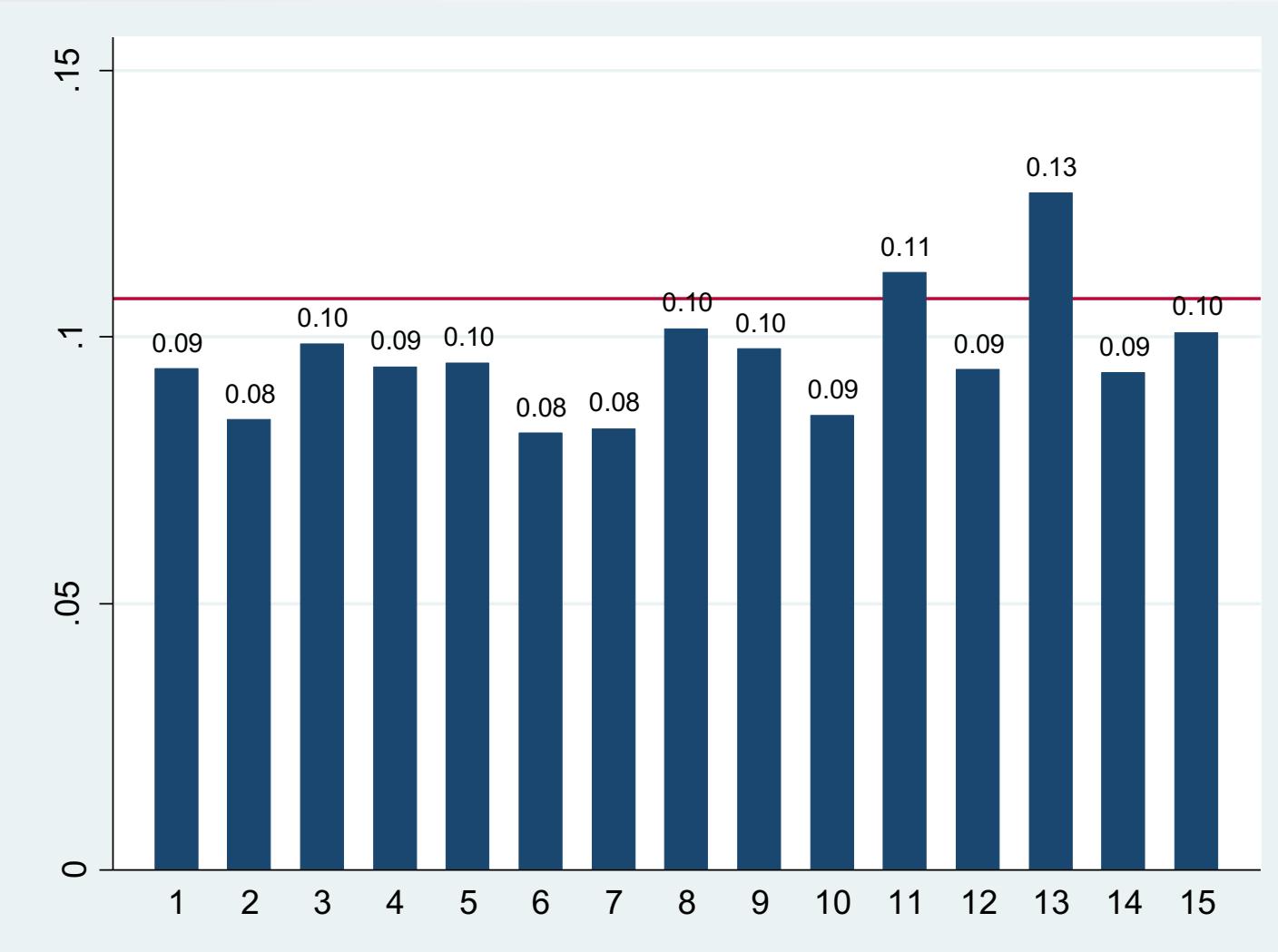
- Vamos a crear la variable `región` como una secuencia de 1 a 15:

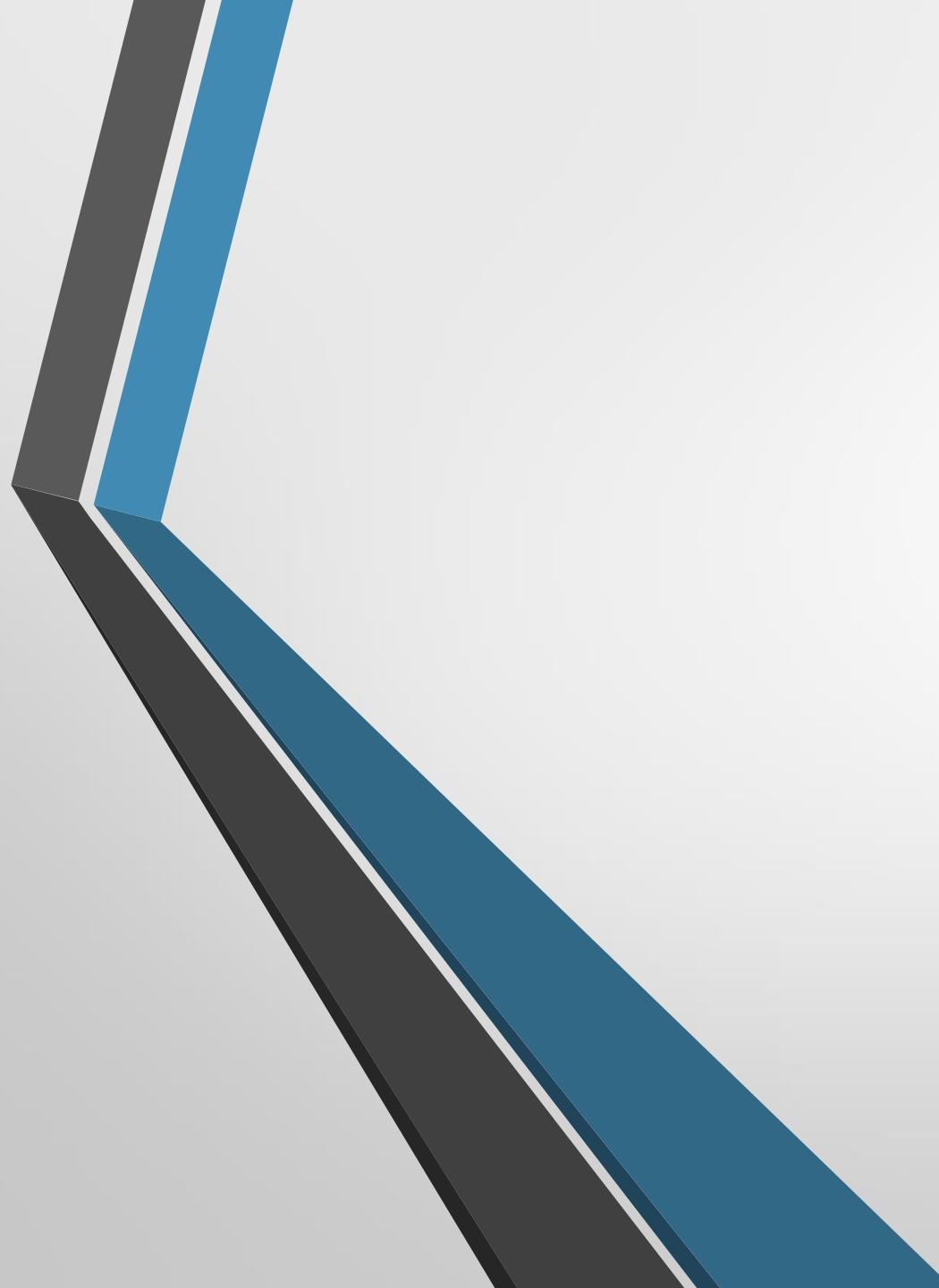
```
. egen region=seq()
```

- Y ahora podemos graficar los resultados:

```
. graph bar (mean) regR1, over(region) blabel(bar, format(%3.2f)) ytitle(Retorno a la educación) yline(0.1071826)
```

Procesos iterativos y matrices





Taller Stata

Clase 12

Javiera Vásquez

Uso de matrices para guardar resultados

- También se pueden definir los elementos de las matrices rescatando información de otros comandos de STATA.
- Suponga que se quiere completar la siguiente tabla de datos:

	Ingreso por hora promedio	Escolaridad promedio	Edad promedio
Hombres			
Mujeres			

- Para esto podemos definir una matriz en Stata e ir rellenando los valores

Uso de matrices para guardar resultados

- Primero definimos la matriz con el tamaño necesario, pero con puros ceros:

```
. matrix res=J(2,3,0)

. matrix list res

res[2,3]
    c1   c2   c3
r1     0   0   0
r2     0   0   0
```

- Luego se deben llenar con la información requerida

```
. sum yph if sexo==1

Variable |       Obs        Mean      Std. Dev.       Min       Max
          | 62735  2730.437  5962.874  3.333333  560000

. matrix res[1,1]=r(mean)

. matrix list res

res[2,3]
    c1       c2       c3
r1  2730.4373      0       0
r2      0       0       0
```

Uso de matrices para guardar resultados

```
. sum yph if sex==2

Variable      Obs       Mean    Std. Dev.      Min      Max
yph          43677   2360.171   3587.584   15.55556  186666.7

. matrix res[2,1]=r(mean)

. sum esc if sexo==1

Variable      Obs       Mean    Std. Dev.      Min      Max
esc          62735   11.13229   3.901555      0        22

. matrix res[1,2]=r(mean)

. sum esc if sexo==2

Variable      Obs       Mean    Std. Dev.      Min      Max
esc          43677   12.04153   3.745341      0        22

. matrix res[2,2]=r(mean)
```

Uso de matrices para guardar resultados

```
. sum edad if sexo==1
```

Variable	Obs	Mean	Std. Dev.	Min	Max
edad	62735	43.20521	14.15695	15	99

```
. matrix res[1,3]=r(mean)
```

```
. sum edad if sexo==2
```

Variable	Obs	Mean	Std. Dev.	Min	Max
edad	43677	41.67267	13.07914	15	94

```
. matrix res[2,3]=r(mean)
```

```
. matrix list res
```

```
res[2,3]
```

	c1	c2	c3
r1	2730.4373	11.132287	43.205212
r2	2360.1711	12.041532	41.672665

Nombres de filas y columnas

- Por defecto las columnas son nombradas como c₁, c₂,...etc y las filas como r₁, r₂,....etc.
- Para cambiar los nombres de las filas, usamos el comando matrix rownames:

```
. matrix rownames res=Hombres Mujeres  
  
. matrix list res  
  
res [2,3]  
          c1           c2           c3  
Hombres  2730.4373  11.132287  43.205212  
Mujeres  2360.1711  12.041532  41.672665
```

Nombres de filas y columnas

- Para cambiar los nombres de las filas, usamos el comando `matrix colnames`:

```
. matrix colnames res=Ingreso Escolaridad Edad  
  
. matrix list res  
  
res [2,3]  
          Ingreso    Escolaridad        Edad  
Hombres    2730.4373    11.132287    43.205212  
Mujeres    2360.1711    12.041532    41.672665
```

Procesos iterativos y matrices

- Suponga que queremos hacer una regresión para estimar el retorno a la educación, controlando por edad, edad al cuadrado y género, pero para cada región por separado y guardar los coeficientes del retorno a la educación y género las respectivas regresiones en una matriz.
- Primero debemos definir la matriz que guarde los resultados, la que tendrá 15 filas (una para cada región) y 2 columnas (ya que son 2 coeficientes estimados que se tienen que guardar):

```
. matrix define regR=J(15,2,0)
```

Procesos iterativos y matrices

- Luego se hace un proceso iterativo con el comando `forvalues`, para hacer una regresión para cada región e ir guardando los resultados.

```
forvalues i=1(1)15 {  
  
    qui reg lypn esc edad edadc i.sex if region==`i'  
  
    matrix define b=e(b)  
  
    matrix regR[`i',1]=b[1,1]  
    matrix regR[`i',2]=b[1,5]  
  
}
```

Procesos iterativos y matrices

- Se obtiene el siguiente resultado:

```
. matrix list regR
```

regR[15,2]

	c1	c2
r1	.09405477	-.20069843
r2	.08449336	-.23181355
r3	.09879548	-.26828234
r4	.09443317	-.23093731
r5	.09520497	-.19686831
r6	.08204545	-.20910367
r7	.0828229	-.20576355
r8	.10161031	-.2020016
r9	.0977957	-.13818298
r10	.08527623	-.16446349
r11	.11208721	-.1793373
r12	.09391869	-.19566929
r13	.12698749	-.20523704
r14	.09329155	-.16225086
r15	.10087113	-.04362395

Procesos iterativos y matrices

- Luego podemos poner los nombres a las filas y a las columnas:

```
. matrix rownames regR=I II III IV V VI VII VIII IX X XI XII XIII XIV XV  
  
. matrix colnames regR=ESC MUJER  
  
. matrix list regR  
  
regR[15,2]  
          ESC      MUJER  
    I   .09405477 -.20069843  
    II   .08449336 -.23181355  
    III   .09879548 -.26828234  
    IV   .09443317 -.23093731  
    V   .09520497 -.19686831  
    VI   .08204545 -.20910367  
    VII   .0828229 -.20576355  
    VIII   .10161031 -.2020016  
    IX   .0977957 -.13818298  
    X   .08527623 -.16446349  
    XI   .11208721 -.1793373  
    XII   .09391869 -.19566929  
    XIII   .12698749 -.20523704  
    XIV   .09329155 -.16225086  
    XV   .10087113 -.04362395
```

Procesos iterativos y matrices

- Esta matriz la podemos pasar a Excel directamente seleccionando, copiando como tabla y pegando en Excel:

	ESC	MUJER
I	.09405477	-.20069843
II	.08449336	-.23181355
III	.09879548	-.26828234
IV	.09443317	-.23093731
V	.09520497	-.19686831
VI	.08204545	-.20910367
VII	.0828229	-.20576355
VIII	.10161031	-.2020016
IX	.0977957	-.13818298
X	.08527623	-.16446349
XI	.11208721	-.1793373
XII	.09391869	-.19566929
XIII	.12698749	-.20523704
XIV	.09329155	-.16225086
XV	.10087113	-.043
.		

Copy
Copy Table
Copy Table as HTML
Copy as Picture
Select All Ctrl+A
Preferences...
Font...
Print...

Procesos iterativos y matrices

- Y también podemos transformar la matriz en una base de datos en Stata, esto se hace con el comando:

```
Create variables from matrix
```

```
svmat [type] A [, names(col|eqcol|matcol|string) ]
```

Con esto limpiamos la Casen 2017, para luego crear las variables a partir de la matriz, sino limpiamos quedaran mezcladas estas nuevas variables con la base que estaba cargada antes

```
. clear
```

```
. svmat regR
```

```
number of observations will be reset to 15
```

```
Press any key to continue, or Break to abort
```

```
obs was 0, now 15
```

Procesos iterativos y matrices

- Así queda la base de datos en STATA:

	regR1	regR2	
1	.0940548	-.2006984	
2	.0844934	-.2318135	
3	.0987955	-.2682824	
4	.0944332	-.2309373	
5	.095205	-.1968683	
6	.0820455	-.2091037	
7	.0828229	-.2057635	
8	.1016103	-.2020016	
9	.0977957	-.138183	
10	.0852762	-.1644635	
11	.1120872	-.1793373	
12	.0939187	-.1956693	
13	.1269875	-.205237	
14	.0932915	-.1622509	
15	.1008711	-.043624	

Procesos iterativos y matrices

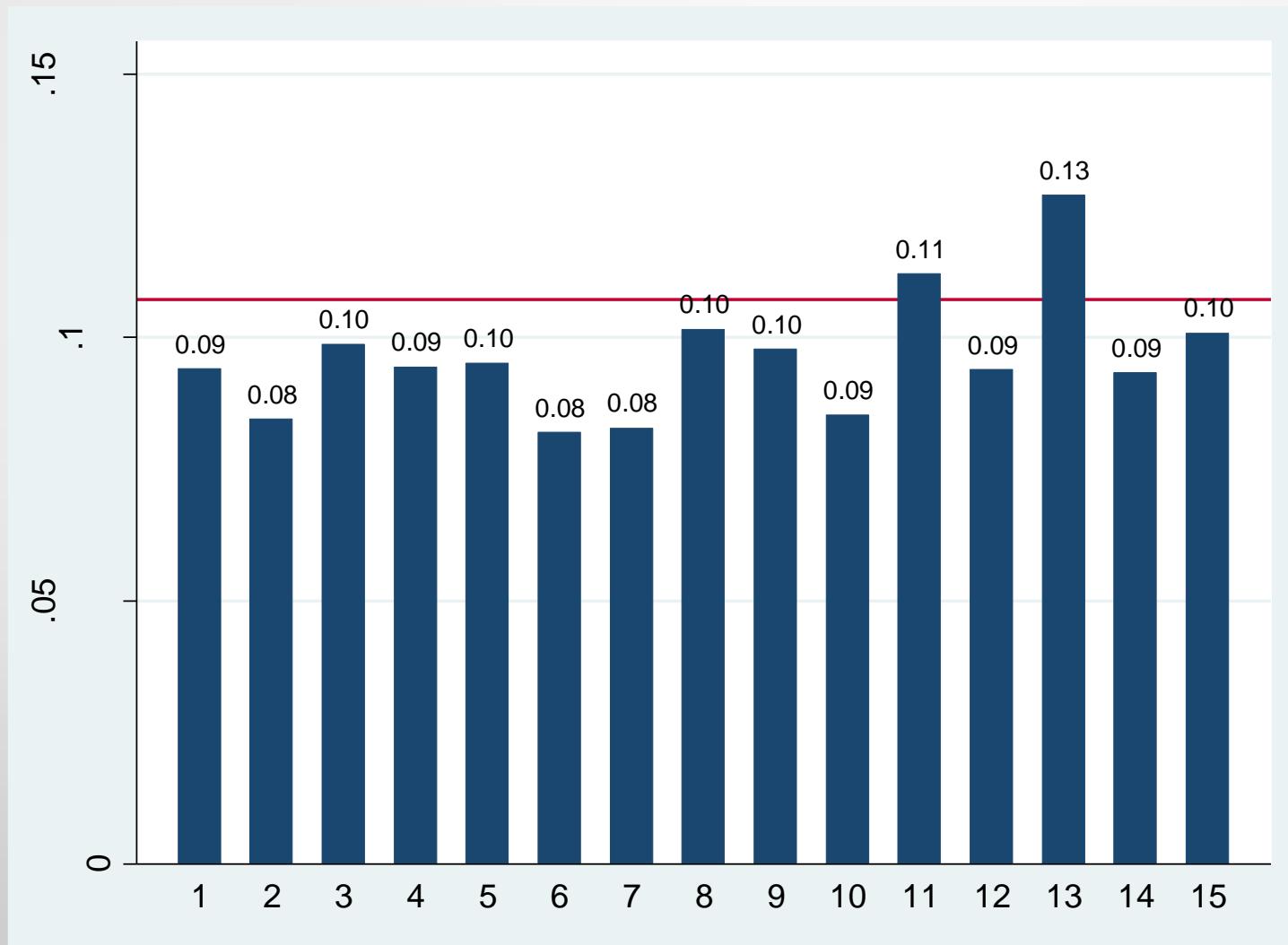
- Vamos a crear la variable `región` como una secuencia de 1 a 15:

```
. egen region=seq()
```

- Y ahora podemos graficar los resultados:

```
. graph bar (mean) regR1, over(region) blabel(bar, format(%3.2f)) ytitle(Retorno a la educación) yline(0.1071826)
```

Procesos iterativos y matrices



Bootstrap

- Este método de simulación sirve para derivar error estándar e intervalos de confianza, considerando la distribución empírica de un estimador, y no bajo supuestos (por ejemplo) de normalidad del error.
- Stata viene con un comando ya programado para hacer bootstrap, pero también se puede programar fácilmente.
- Comencemos con programar bootstrap para calcular el error estándar de ciertos indicadores.
- Para esto utilizaremos una parte de la encuesta de calidad de vida en adultos mayores.

Bootstrap

- Indicador: condiciones excelentes

	De acuerdo	En desacuerdo
<p>2. Se presentan cinco afirmaciones con las que usted puede o no estar de acuerdo. Por favor dígame lo que piensa en cada caso:</p> <p>» Mostrar tarjeta Nº3</p>		
1. En gran parte, su vida está cercana a lo ideal	<input type="checkbox"/>	<input type="checkbox"/>
2. Las condiciones de su vida son excelentes	<input type="checkbox"/>	<input type="checkbox"/>
3. Está satisfecho con su vida	<input type="checkbox"/>	<input type="checkbox"/>
4. Hasta ahora, ha conseguido las cosas que para usted son importantes en la vida	<input type="checkbox"/>	<input type="checkbox"/>
5. Si usted volviera a nacer, no cambiaría casi nada de su vida	<input type="checkbox"/>	<input type="checkbox"/>

Bootstrap

- Indicador: satisfecho con la vida

	De acuerdo	En desacuerdo
<p>2. Se presentan cinco afirmaciones con las que usted puede o no estar de acuerdo. Por favor dígame lo que piensa en cada caso:</p> <p>» Mostrar tarjeta Nº3</p> <p>1. En gran parte, su vida está cercana a lo ideal</p>	<input type="checkbox"/>	<input type="checkbox"/>
<p>2. Las condiciones de su vida son excelentes</p>	<input type="checkbox"/>	<input type="checkbox"/>
<p>3. Está satisfecho con su vida</p>	<input type="checkbox"/>	<input type="checkbox"/>
<p>4. Hasta ahora, ha conseguido las cosas que para usted son importantes en la vida</p>	<input type="checkbox"/>	<input type="checkbox"/>
<p>5. Si usted volviera a nacer, no cambiaría casi nada de su vida</p>	<input type="checkbox"/>	<input type="checkbox"/>

Bootstrap

```
set seed 100

matrix boots=J(300,2,.)

forvalues i=1(1)300 {

    use "G:\FNE\FEN_Taller Stata\Otoño 2018\Bases de datos\base_clase14.dta", clear

    bsample

    gen condiciones_excelentes=1 if s2_2_2==1
    replace condiciones_excelentes=0 if s2_2_2==2 | s2_2_2==8| s2_2_2==9

    gen satisfecho_vida=1 if s2_2_3==1
    replace satisfecho_vida=0 if s2_2_3==2 | s2_2_3==8| s2_2_3==9

    proportion condiciones_excelentes [pw=fexp_cv2018]
    matrix results=r(table)

    matrix boots[`i',1]=results[1,2]

    proportion satisfecho_vida [pw=fexp_cv2018]
    matrix results=r(table)

    matrix boots[`i',2]=results[1,2]

}

clear
svmat boots
sum boots*
```

Bootstrap

Variable	Obs	Mean	Std. Dev.	Min	Max
boots1	300	.4542773	.0118115	.4263015	.4902963
boots2	300	.7731841	.0107695	.743354	.7995453

Bootstrap

- También se puede hacer utilizando el comando bootstrap:

```
clear matrix
scalar drop _all

use "G:\FNE\FEN_Taller Stata\Otoño 2018\Bases de datos\base_clase14.dta", clear

gen condiciones_excelentes=1 if s2_2_2==1
replace condiciones_excelentes=0 if s2_2_2==2 | s2_2_2==8| s2_2_2==9

gen satisfecho_vida=1 if s2_2_3==1
replace satisfecho_vida=0 if s2_2_3==2 | s2_2_3==8| s2_2_3==9

capture program drop bootind
program define bootind, rclass
preserve

tab condiciones_excelentes [aw=fexp_cv2018], matcell(z0)

scalar define ind1=z0[2,1]/(z0[1,1]+z0[2,1])

return scalar ind1_r=ind1

tab satisfecho_vida [aw=fexp_cv2018], matcell(z1)

scalar define ind2=z1[2,1]/(z1[1,1]+z1[2,1])

return scalar ind2_r=ind2

restore
end

set seed 100
bootstrap ind1=r(ind1_r), reps(300) saving("G:\FNE\FEN_Taller Stata\Otoño 2018\Bases de datos\ind1.dta", replace): bootind
bootstrap ind2=r(ind2_r), reps(300) saving("G:\FNE\FEN_Taller Stata\Otoño 2018\Bases de datos\ind2.dta", replace): bootind
```

Bootstrap

- Se obtienen los siguientes resultados:

```
Bootstrap replications (300)
+---+ 1 +---+ 2 +---+ 3 +---+ 4 +---+ 5
..... 50
..... 100
..... 150
..... 200
..... 250
..... 300

Bootstrap results                               Number of obs      =     2523
                                                Replications      =      300

command: bootind
        ind1: r(ind1_r)

                                         Observed   Bootstrap
                                         Coef.       Std. Err.      z      P>|z|
                                                               [95% Conf. Interval]
ind1    .4533228    .0118115    38.38    0.000    .4301726    .476473
```

Bootstrap

```
Bootstrap replications (300)
+---+---+---+---+---+
| 1 | 2 | 3 | 4 | 5 |
+---+---+---+---+---+
..... 50
..... 100
..... 150
..... 200
..... 250
..... 300

Bootstrap results                               Number of obs = 2523
                                                Replications = 300

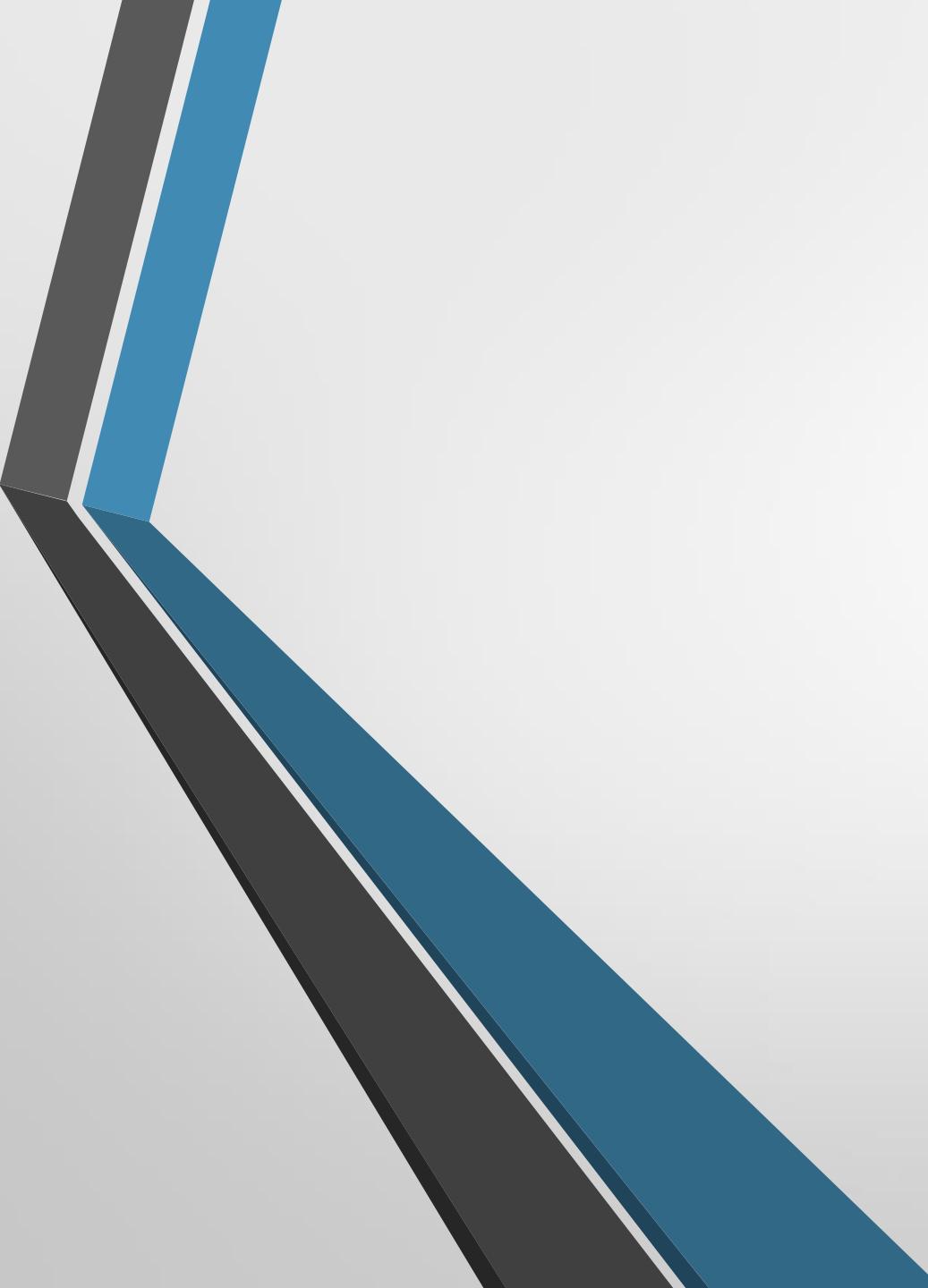
command: bootind
          ind2: r(ind2_r)

      Observed   Bootstrap
      Coef.     Std. Err.      z    P>|z|  Normal-based
                                         [95% Conf. Interval]
ind2    .7722427   .0105272   73.36   0.000   .7516097   .7928756
```

Bootstrap

- También se puede utilizar el comando bootstrap para obtener los intervalos de confianza de los coeficientes estimados de un modelo de regresión lineal:

lyph	Observed Coef.	Bootstrap Std. Err.	z	P> z	Normal-based [95% Conf. Interval]	
esc	.1071826	.0006699	160.01	0.000	.1058697	.1084955
edad	.0281598	.0009893	28.47	0.000	.0262209	.0300987
c.edad##c.edad	-.0002279	.0000118	-19.26	0.000	-.0002511	-.0002047
2.sexo	-.196391	.0043509	-45.14	0.000	-.2049185	-.1878634
_cons	5.588089	.0199115	280.65	0.000	5.549063	5.627115



Taller Stata

Clase 14

Javiera Vásquez

Bootstrap

```
set seed 100

matrix boots=J(300,2,.)

forvalues i=1(1)300 {

    use "G:\FNE\FEN_Taller Stata\Otoño 2018\Bases de datos\base_clase14.dta", clear

    bsample

    gen condiciones_excelentes=1 if s2_2_2==1
    replace condiciones_excelentes=0 if s2_2_2==2 | s2_2_2==8| s2_2_2==9

    gen satisfecho_vida=1 if s2_2_3==1
    replace satisfecho_vida=0 if s2_2_3==2 | s2_2_3==8| s2_2_3==9

    proportion condiciones_excelentes [pw=fexp_cv2018]
    matrix results=r(table)

    matrix boots[`i',1]=results[1,2]

    proportion satisfecho_vida [pw=fexp_cv2018]
    matrix results=r(table)

    matrix boots[`i',2]=results[1,2]

}

clear
svmat boots
sum boots*
```

Bootstrap

Variable	Obs	Mean	Std. Dev.	Min	Max
boots1	300	.4542773	.0118115	.4263015	.4902963
boots2	300	.7731841	.0107695	.743354	.7995453

Bootstrap

- También se puede hacer utilizando el comando bootstrap:

```
clear matrix
scalar drop _all

use "G:\FNE\FEN_Taller Stata\Otoño 2018\Bases de datos\base_clase14.dta", clear

gen condiciones_excelentes=1 if s2_2_2==1
replace condiciones_excelentes=0 if s2_2_2==2 | s2_2_2==8| s2_2_2==9

gen satisfecho_vida=1 if s2_2_3==1
replace satisfecho_vida=0 if s2_2_3==2 | s2_2_3==8| s2_2_3==9

capture program drop bootind
program define bootind, rclass
preserve

tab condiciones_excelentes [aw=fexp_cv2018], matcell(z0)

scalar define ind1=z0[2,1]/(z0[1,1]+z0[2,1])

return scalar ind1_r=ind1

tab satisfecho_vida [aw=fexp_cv2018], matcell(z1)

scalar define ind2=z1[2,1]/(z1[1,1]+z1[2,1])

return scalar ind2_r=ind2

restore
end

set seed 100
bootstrap ind1=r(ind1_r), reps(300) saving("G:\FNE\FEN_Taller Stata\Otoño 2018\Bases de datos\ind1.dta", replace): bootind
bootstrap ind2=r(ind2_r), reps(300) saving("G:\FNE\FEN_Taller Stata\Otoño 2018\Bases de datos\ind2.dta", replace): bootind
```

Bootstrap

- Se obtienen los siguientes resultados:

```
Bootstrap replications (300)
+---+ 1 +---+ 2 +---+ 3 +---+ 4 +---+ 5
..... 50
..... 100
..... 150
..... 200
..... 250
..... 300

Bootstrap results                               Number of obs      =     2523
                                                Replications      =      300

command: bootind
        ind1: r(ind1_r)

                                         Observed   Bootstrap
                                         Coef.       Std. Err.      z      P>|z|
                                                               [95% Conf. Interval]
ind1    .4533228    .0118115    38.38    0.000    .4301726    .476473
```

Bootstrap

```
Bootstrap replications (300)
+---+---+---+---+---+
| 1 | 2 | 3 | 4 | 5 |
+---+---+---+---+---+
..... 50
..... 100
..... 150
..... 200
..... 250
..... 300

Bootstrap results                               Number of obs = 2523
                                                Replications = 300

command: bootind
          ind2: r(ind2_r)

      Observed   Bootstrap
      Coef.     Std. Err.      z    P>|z|  Normal-based
                                         [95% Conf. Interval]
ind2    .7722427   .0105272   73.36  0.000   .7516097   .7928756
```

Bootstrap

- También se puede utilizar el comando bootstrap para obtener los intervalos de confianza de los coeficientes estimados de un modelo de regresión lineal:

lyph	Observed Coef.	Bootstrap Std. Err.	z	P> z	Normal-based [95% Conf. Interval]	
esc	.1071826	.0006699	160.01	0.000	.1058697	.1084955
edad	.0281598	.0009893	28.47	0.000	.0262209	.0300987
c.edad##c.edad	-.0002279	.0000118	-19.26	0.000	-.0002511	-.0002047
2.sexo	-.196391	.0043509	-45.14	0.000	-.2049185	-.1878634
_cons	5.588089	.0199115	280.65	0.000	5.549063	5.627115

Tabout

- tabout es un comando de Stata que permite hacer tablas de calidad para ser publicadas.
- Se pueden hacer tablas básicas de una y dos entradas y tablas de estadísticas descriptivas
- Primero debemos instalar el comando
- La sintaxis básica del comando es:

Syntax

```
tabout [ varlist ] [ if exp ] [ in range ] [ weight = exp ] using  
filename [ , options ]
```

Tabout

- Comencemos con un ejemplo sencillo utilizando la encuesta Casen 2015.
- Generaremos una variable que clasifique a las personas entre ocupados y no ocupados:

```
cd "G:\FNE\FEN_Taller Stata\Clases"

use "G:\FNE\casen_2015_stata\Casen 2015.dta", clear

g ocupado=1 if o1==1 | o2==1 | o3==1
replace ocupado=2 if ocupado==.
replace ocupado=. if o1==.

label variable ocupado "Indicador si está ocupado"
label define ocupado 1 "Ocupado" 2 "No ocupado"
label values ocupado ocupado
```

Tabout: tabla básica de una entrada

- Hagamos una tabla básica de una entrada, la vamos a guardar como archivo Excel, y la llamaremos Tabla1:

```
. tabout ocupado using Tabla1.xls, replace  
Table output written to: Tabla1.xls  
  
Indicador si est. ocupado      No.  
Ocupado 110,526.0  
No ocupado    113,435.0  
Total    223,961.0
```



	A	B
1	Indicador si está ocupado	No.
2	Ocupado	110,526.00
3	No ocupado	113,435.00
4	Total	223,961.00
5		
6		

- Por defecto lo que presenta es la frecuencia de observaciones (No.)
- Si queremos que muestre los porcentajes

Tabout: tabla básica de una entrada

- Con la opción `cells` cambiamos el contenido de la celda:

`cells` determines the contents of table cells. As the table below shows, you can enter any one or more of `freq` `cell` `row` `col` `cum` in a basic table. They can be in any order. When you choose the `svy` option, you can only have one of these choices, and it must come first. The additional choices which are then available are: `se` `ci` `lb` `ub`.

```
. tabout ocupado using Tabla1.xls, replace cells(col)
Table output written to: Tabla1.xls
Indicador si est. ocupado      %
Ocupado    49.4
No ocupado   50.6
Total     100.0
```



	A	B
1	Indicador si está ocupado	%
2	Ocupado	49.4
3	No ocupado	50.6
4	Total	100

Tabout: tabla básica de una entrada

- Si queremos que muestre la frecuencia y el porcentaje al mismo tiempo:

```
. tabout ocupado using Tabla1.xls, replace cells(freq col)
```

	A	B	C
1	Indicador si está ocupado	No.	%
2	Ocupado	110,526.00	49.4
3	No ocupado	113,435.00	50.6
4	Total	223,961.00	100
5			

- Ahora podemos darle más formato a esta tabla, ya que ahora está en la manera más simple.
- La opción format indica el número de puntos decimales, por ejemplo:

Tabout: tabla básica de una entrada

```
. tabout ocupado using Tabla1.xls, replace cells(freq col) format(0 1)
```

	A	B	C
1	Indicador si está ocupado	No.	%
2	Ocupado	110526	49.4
3	No ocupado	113435	50.6
4	Total	223961	100
5			

- También se puede ocupar c para que incluya las comas en los números y p para que incluya un porcentaje, por ejemplo:

```
. tabout ocupado using Tabla1.xls, replace cells(freq col) format(0c 1p)
```

	A	B	C
1	Indicador si está ocupado	No.	%
2	Ocupado	110,526	49.4%
3	No ocupado	113,435	50.6%
4	Total	223,961	100.0%
5			
6			

Tabout: tabla básica de una entrada

- La opción clab se utiliza para etiquetar las columnas de la tabla:

	A	B	C
1	Indicador si está ocupado	No.	%
2	Ocupado	110,526	49.4%
3	No ocupado	113,435	50.6%
4	Total	223,961	100.0%
5			
6			

```
. tabout ocupado using Tabla1.xls, replace cells(freq col) format(0c 1p) clab(N %)
```

	A	B	C
1	Indicador si está ocupado	N	%
2	Ocupado	110,526	49.40%
3	No ocupado	113,435	50.60%
4	Total	223,961	100.00%
5			

Tabout: tabla básica de una entrada

- Si queremos guardar la tabla en formato .tex debemos por una parte cambiar el nombre del archivo con extensión .tex, pero además debemos agregar la opción `style(text)`, esta opción por defecto es `style(tab)` que genera la tabla con formato tabulado.

```
. tabout ocupado using Tabla1.tex, replace cells(freq col) format(0c 1p) clab(N %) style(tex)

Table output written to: Tabla1.tex

Indicador si est. ocupado&N&% \\ 
\hline
Ocupado&110,526&49.4% \\
No ocupado&113,435&50.6% \\
Total&223,961&100.0% \\
```

```
15 \begin{table}[]
16   \centering
17   \begin{tabular}{c|c|c}
18 Indicador si está ocupado & N &% \\
19 \hline
20 Ocupado & 110,526 & 49.4% \\
21 No ocupado & 113,435 & 50.6% \\
22 Total & 223,961 & 100.0% \\
23   \end{tabular}
24   \caption{Caption}
25   \label{tab:my_label}
26 \end{table}
```

Indicador si está ocupado	N	%
Ocupado	110,526	49.4%
No ocupado	113,435	50.6%
Total	223,961	100.0%

Table 1: Caption

Tabout: tabla básica de dos entradas

```
. label variable ocupado "Estatus"
```

```
. tabout ocupado sexo using Tabla2.xls, replace cells(freq col) format(0c 1p) clab(N %) style(tab)
```

	A	B	C	D	E	F	G
1		sexos					
2	Estatus	hombre	hombre	mujer	mujer	Total	Total
3		N	%	N	%	N	%
4	Ocupado	65,031	61.60%	45,495	38.40%	110,526	49.40%
5	No ocupado	40,598	38.40%	72,837	61.60%	113,435	50.60%
6	Total	105,629	100.00%	118,332	100.00%	223,961	100.00%
7							

Tabout: tabla básica de dos entradas

- Código para latex:

```
. tabout ocupado sexo using Tabla2.txt, replace cells(freq col) format(0c 1p) clab(N %) style(tex) cl1(2-7) cl2(2-3 4-5 6-7) font(bold) bt  
  
Table output written to: Tabla2.txt  
  
& \multicolumn{6}{c}{\textbf{sexo}} \\  
\cmidrule(l{.75em}){2-7}  
\textbf{Estatus} & \multicolumn{2}{c}{\textbf{hombre}} & \multicolumn{2}{c}{\textbf{mujer}} & \multicolumn{2}{c}{\textbf{Total}} \\  
\cmidrule(l{.75em}){2-3} \cmidrule(l{.75em}){4-5}\cmidrule(l{.75em}){6-7}  
&\%&\%&\%&\% \\  
\midrule  
Ocupado&65,031&61.6\%&45,495&38.4\%&110,526&49.4\% \\  
No ocupado&40,598&38.4\%&72,837&61.6\%&113,435&50.6\% \\  
\textbf{Total}&105,629&100.0\%&118,332&100.0\%&223,961&100.0\%
```

```
\begin{table}[]  
  \centering  
  \begin{tabular}{cccccc}  
  
& \multicolumn{6}{c}{\textbf{sexo}} \\  
\cmidrule(l{.75em}){2-7}  
\textbf{Estatus} & \multicolumn{2}{c}{\textbf{hombre}} & \multicolumn{2}{c}{\textbf{mujer}} & \multicolumn{2}{c}{\textbf{Total}} \\  
\cmidrule(l{.75em}){2-3} \cmidrule(l{.75em}){4-5}\cmidrule(l{.75em}){6-7}  
&\%&\%&\%&\% \\  
\midrule  
Ocupado&65,031&61.6\%&45,495&38.4\%&110,526&49.4\% \\  
No ocupado&40,598&38.4\%&72,837&61.6\%&113,435&50.6\% \\  
\textbf{Total}&105,629&100.0\%&118,332&100.0\%&223,961&100.0\%  
  
  \end{tabular}  
  \caption{Caption}  
  \label{tab:my_label}  
\end{table}
```

Estatus	sexo					
	hombre		mujer		Total	
	N	%	N	%	N	%
Ocupado	65,031	61.6%	45,495	38.4%	110,526	49.4%
No ocupado	40,598	38.4%	72,837	61.6%	113,435	50.6%
Total	105,629	100.0%	118,332	100.0%	223,961	100.0%

Table 1: Caption

Tabout: tabla básica de más entradas

- También podemos hacer tablas con más entradas, dentro del listado de variables que entreguemos la última variables será la que vaya en las columnas, el resto se pondrán en las filas.
- Por ejemplo, suponga que queremos ver la distribución de frecuencias y porcentual de los universitarios y no universitarios según zona geográfica y sexo.
- Primero generemos la variables con que clasifique a las personas con estudios universitarios y sin estudios universitarios:

```
. g univ=1 if esc>12  
(158849 missing values generated)

. replace univ=0 if esc<=12  
(158849 real changes made)

. replace univ=. if esc==.  
(54626 real changes made, 54626 to missing)

. label define univ 0 "No Univ." 1 "Univ."

. label values univ univ
```

Tabout: tabla básica de más entradas

- Luego hacemos la tabla en Excel:

```
. tabout zona sexo univ using Tabla3.xls, cells(freq row col) format(0c 1p 1p) clab(_ _ _) replace
```

	univ								
	No Univ.	No Univ.	No Univ.	Univ.	Univ.	Univ.	Total	Total	Total
zona									
urbano	117,678	71.10%	74.10%	47,818	28.90%	89.40%	165,496	100.00%	77.90%
rural	41,171	87.90%	25.90%	5,675	12.10%	10.60%	46,846	100.00%	22.10%
Total	158,849	74.80%	100.00%	53,493	25.20%	100.00%	212,342	100.00%	100.00%
sexo									
hombre	74,788	75.00%	47.10%	24,974	25.00%	46.70%	99,762	100.00%	47.00%
mujer	84,061	74.70%	52.90%	28,519	25.30%	53.30%	112,580	100.00%	53.00%
Total	158,849	74.80%	100.00%	53,493	25.20%	100.00%	212,342	100.00%	100.00%

- Así la tabla no se entiende muy bien, podemos usar algunas opciones para mejorar el formato

Tabout: tabla básica de más entradas

- Agregamos la opción layout:

```
. tabout zona sexo univ using Tabla3.xls, cells(freq row col) format(0c 1p 1p) clab(____) replace layout(rb)
```

	univ		
	No Univ.	Univ.	Total
zona			
urbano	117,678	47,818	165,496
rural	41,171	5,675	46,846
Total	158,849	53,493	212,342
 			
urbano	71.10%	28.90%	100.00%
rural	87.90%	12.10%	100.00%
Total	74.80%	25.20%	100.00%
 			
urbano	74.10%	89.40%	77.90%
rural	25.90%	10.60%	22.10%
Total	100.00%	100.00%	100.00%
 			
sexo			
hombre	74,788	24,974	99,762
mujer	84,061	28,519	112,580
Total	158,849	53,493	212,342
 			
hombre	75.00%	25.00%	100.00%
mujer	74.70%	25.30%	100.00%
Total	74.80%	25.20%	100.00%
 			
hombre	47.10%	46.70%	47.00%
mujer	52.90%	53.30%	53.00%
Total	100.00%	100.00%	100.00%

row block

Tabout: tabla básica de más entradas

- En Latex:

```
. tabout zona sexo univ using Tabla3.txt, cells(freq row col) format(0c 1p 1p) clab(____) replace layout(rb) style(tex) bt font(bold)
```

```
\begin{table}[]
  \centering
  \begin{tabular}{ccccccc}
    & \multicolumn{3}{c}{\textbf{univ}} & & & \\
    & \textbf{No Univ.} & \textbf{Univ.} & \textbf{Total} & & & \\
    \midrule
    \textbf{zona} & & & & & & \\
    urbano&117,678&47,818&165,496 & & & \\
    rural&41,171&5,675&46,846 & & & \\
    \textbf{Total}&158,849&53,493&212,342 & & & \\
    \midrule
    urbano&71.1\%&28.9\%&100.0\% & & & \\
    rural&87.9\%&12.1\%&100.0\% & & & \\
    \textbf{Total}&74.8\%&25.2\%&100.0\% & & & \\
    \midrule
    urbano&74.1\%&89.4\%&77.9\% & & & \\
    rural&25.9\%&10.6\%&22.1\% & & & \\
    \textbf{Total}&100.0\%&100.0\%&100.0\% & & & \\
    \midrule
    \textbf{sexo} & & & & & & \\
    hombre&74,788&24,974&99,762 & & & \\
    mujer&84,061&28,519&112,580 & & & \\
    \textbf{Total}&158,849&53,493&212,342 & & & \\
    \midrule
    hombre&75.0\%&25.0\%&100.0\% & & & \\
    mujer&74.7\%&25.3\%&100.0\% & & & \\
    \textbf{Total}&74.8\%&25.2\%&100.0\% & & & \\
    \midrule
    hombre&47.1\%&46.7\%&47.0\% & & & \\
    mujer&52.9\%&53.3\%&53.0\% & & & \\
    \textbf{Total}&100.0\%&100.0\%&100.0\% & & & \\
  \end{tabular}
  \caption{Caption}
  \label{tab:my_label}
\end{table}
```

	No Univ.	univ	Total
zona			
urbano	117,678	47,818	165,496
rural	41,171	5,675	46,846
Total	158,849	53,493	212,342
urbano			
rural	87.9%	12.1%	100.0%
Total	74.8%	25.2%	100.0%
urbano			
rural	25.9%	10.6%	22.1%
Total	100.0%	100.0%	100.0%
sexo			
hombre	74,788	24,974	99,762
mujer	84,061	28,519	112,580
Total	158,849	53,493	212,342
hombre			
mujer	74.7%	25.3%	100.0%
Total	74.8%	25.2%	100.0%
hombre			
mujer	47.1%	46.7%	47.0%
Total	52.9%	53.3%	53.0%
Total	100.0%	100.0%	100.0%

Table 1: Caption

Tabout: tabla básica de más entradas

- Si queremos eliminar alguno de los “*headings*”, por ejemplo el de la primera fila:

	No Univ.	Univ.	Total
zona			
urbano	117,678	47,818	165,496
rural	41,171	5,675	46,846
Total	158,849	53,493	212,342
urbano	71.1%	28.9%	100.0%
rural	87.9%	12.1%	100.0%
Total	74.8%	25.2%	100.0%
urbano	74.1%	89.4%	77.9%
rural	25.9%	10.6%	22.1%
Total	100.0%	100.0%	100.0%
sexo			
hombre	74,788	24,974	99,762
mujer	84,061	28,519	112,580
Total	158,849	53,493	212,342
hombre	75.0%	25.0%	100.0%
mujer	74.7%	25.3%	100.0%
Total	74.8%	25.2%	100.0%
hombre	47.1%	46.7%	47.0%
mujer	52.9%	53.3%	53.0%
Total	100.0%	100.0%	100.0%

Table 1: Caption

```
. tabout zona sexo univ using Tabla3.txt, cells(freq row col) format(0c 1p 1p)  
clab(____) replace layout(rb) style(tex) bt font(bold) h1(nil)
```

	No Univ.	Univ.	Total
zona			
urbano	117,678	47,818	165,496
rural	41,171	5,675	46,846
Total	158,849	53,493	212,342
urbano	71.1%	28.9%	100.0%
rural	87.9%	12.1%	100.0%
Total	74.8%	25.2%	100.0%
urbano	74.1%	89.4%	77.9%
rural	25.9%	10.6%	22.1%
Total	100.0%	100.0%	100.0%
sexo			
hombre	74,788	24,974	99,762
mujer	84,061	28,519	112,580
Total	158,849	53,493	212,342
hombre	75.0%	25.0%	100.0%
mujer	74.7%	25.3%	100.0%
Total	74.8%	25.2%	100.0%
hombre	47.1%	46.7%	47.0%
mujer	52.9%	53.3%	53.0%
Total	100.0%	100.0%	100.0%

Table 1: Caption

Tabout: tabla básica de más entradas

- Ejercicio: Haga una tabla que muestre la distribución porcentual de la categoría ocupacional (o15) y del oficio (oficio1) según sexo y para el total

	hombre %	mujer %	Total %
Categoría ocupacional			
Empleador	3.4	2.2	2.9
Cuenta propia	21.8	19.0	20.6
Empleado sector publico	5.8	13.6	9.0
Empleador empresas publicas	1.9	3.1	2.4
Empleado sector privado	65.3	53.1	60.3
Servicio domestico p. adentro	0.0	0.5	0.2
Servicio domestico p.afuera	0.1	7.5	3.1
ffaa y del orden	1.3	0.3	0.9
Familiar no remunerado	0.3	0.7	0.5
Total	100.0	100.0	100.0
Oficio			
Fuerzas armadas	0.7	0.1	0.4
Poder ejecutivo	4.5	6.0	5.1
Profesionales	8.2	13.0	10.2
Tecnicos	7.0	10.9	8.6
Empleados de oficina	4.7	12.5	7.9
Trabajadores de servicio y comercio	9.2	23.8	15.2
Agricultores	10.4	3.5	7.6
Operarios	21.2	4.0	14.1
Operadores	14.9	2.0	9.6
No calificado	19.2	24.1	21.2
Total	100.0	100.0	100.0
Tamaño muestral	65,031	45,495	110,526

Table 1: Caption

Tabout: tablas de estadísticas descriptivas

- Suponga que queremos hacer una tabla con el ingreso de la ocupación principal (por hora) promedio y mediana, y la edad promedio, de los ocupados según categoría ocupacional y oficio.

```
. tabout o15 oficio1 using Tabla5.xls, replace c(mean yph median yph mean edad) sum
```

	Mean yph	Median yph	Mean edad
Categoría ocupacional			
Empleador	5,199.90	2,500.00	49.8
Cuenta prop	2,355.40	1,348.10	48.4
Empleado se	3,384.40	2,396.90	42
Empleador e	3,662.00	2,333.30	41.2
Empleado se	2,400.60	1,555.60	40.1
Servicio dom	1,937.80	1,814.80	51.3
Servicio dom	1,529.30	1,256.40	48.5
ffaa y del org	3,837.30	3,001.90	36.6
Familiar no remunerado			44.4
Total	2,579.10	1,555.60	42.6
Oficio			
Fuerzas arma	3,812.30	3,111.10	36.1
Poder ejecut	4,584.50	1,970.40	49.2
Profesionale	6,005.60	4,242.40	40.3
Tecnicos	3,517.60	2,333.30	39.1
Empleados c	2,185.10	1,750.00	39
Trabajadores	1,852.30	1,361.10	39.7
Agricultores	1,578.20	1,244.40	47.8
Operarios	2,051.70	1,555.60	43.1
Operadores	2,189.90	1,666.70	44
No calificado	1,538.60	1,283.30	44.2
Total	2,579.10	1,555.60	42.6

Tabout: tablas de estadísticas descriptivas

- En Latex:

```
tabout o15 oficio1 using Tabla5.txt, replace c(mean yph median yph mean  
edad) sum format(0c 0c 1) style(tex) bt font(bold) h2(& promedio &  
mediana & promedio\\) npos(row) nlab(Tamaño muestral)
```

	promedio yph	mediana yph	promedio edad
Categoría ocupacional			
Empleador	5,200	2,500	49.8
Cuenta propia	2,355	1,348	48.4
Empleado sector publico	3,384	2,397	42.0
Empleador empresas publicas	3,662	2,333	41.2
Empleado sector privado	2,401	1,556	40.1
Servicio domestico p. adentro	1,938	1,815	51.3
Servicio domestico p.afuera	1,529	1,256	48.5
ffaa y del orden	3,837	3,002	36.6
Familiar no remunerado			44.4
Total	2,579	1,556	42.6
Oficio			
Fuerzas armadas	3,812	3,111	36.1
Poder ejecutivo	4,585	1,970	49.2
Profesionales	6,006	4,242	40.3
Tecnicos	3,518	2,333	39.1
Empleados de oficina	2,185	1,750	39.0
Trabajadores de servicio y comercio	1,852	1,361	39.7
Agricultores	1,578	1,244	47.8
Operarios	2,052	1,556	43.1
Operadores	2,190	1,667	44.0
No calificado	1,539	1,283	44.2
Total	2,579	1,556	42.6
Tamaño muestral	110,526		

Table 1: Caption

Tabout: tablas de estadísticas descriptivas

- En Excel:

```
tabout o15 oficio1 using Tabla5.xls, replace c(mean yph median yph mean  
edad) sum format(0c 0c 1) bt font(bold) h2(|promedio|mediana|promedio|)  
npos(row) nlab(Tamaño muestral)
```

	promedio yph	mediana yph	promedio edad
Categoría ocupacional			
Empleador	5,200	2,500	49.8
Cuenta propia	2,355	1,348	48.4
Empleado sector publico	3,384	2,397	42.0
Empleador empresas publicas	3,662	2,333	41.2
Empleado sector privado	2,401	1,556	40.1
Servicio domestico p. adentro	1,938	1,815	51.3
Servicio domestico p.afuera	1,529	1,256	48.5
ffaa y del orden	3,837	3,002	36.6
Familiar no remunerado			44.4
Total	2,579	1,556	42.6
Oficio			
Fuerzas armadas	3,812	3,111	36.1
Poder ejecutivo	4,585	1,970	49.2
Profesionales	6,006	4,242	40.3
Técnicos	3,518	2,333	39.1
Empleados de oficina	2,185	1,750	39.0
Trabajadores de servicio y comercio	1,852	1,361	39.7
Agricultores	1,578	1,244	47.8
Operarios	2,052	1,556	43.1
Operadores	2,190	1,667	44.0
No calificado	1,539	1,283	44.2
Total	2,579	1,556	42.6
Tamaño muestral	110,526		

Table 1: Caption



Taller Stata

Clase 14

Javiera Vásquez

Regresión de mediana

- Cuando utilizamos el estimador de MCO en el contexto de un modelo de regresión lineal lo que estamos estimando es:

$$E[\widehat{y_i | X_i}] = \hat{\beta}_0 + x_i' \hat{\beta}$$

- Donde el estimador se obtiene de minimizar la suma de los errores al cuadrado, lo que garantiza que la recta de regresión pasa por el promedio.
- Cuando se tienen distribuciones muy asimétricas, el promedio no es una buena medida de tendencia central.
- Lo mismo sucede con el estimador MCO, no será bueno, cuando la variable sea muy asimétrica o existen muchos valores extremos.

Regresión de mediana

- En la regresión de mediana:

$$\widehat{Med[y_i|X_i]} = \hat{\beta}_0 + x'_i \hat{\beta}$$

- Se minimiza el valor absoluto de los errores.
- En términos generales, para cualquier cuartil:

$$\widehat{q[y_i|X_i]} = \hat{\beta}_0 + x'_i \hat{\beta}$$

Regresión de mediana

- A continuación utilizaremos datos de gastos médicos y gastos totales del hogar, ambas variables en logaritmo, para una muestra de 5.006 hogares correspondientes a la encuesta Vietnam Living Standards del Banco Mundial (1997), esta base de datos se llama `qreg.dta`.
- Cuando hacemos una regresión por MCO, con el comando `reg`, se obtienen los efectos marginales sobre el promedio de la variable dependiente, en este caso, como un aumento de un 1% del ingreso total del hogar aumenta en promedio, porcentualmente, el gasto médico.

Regresión de mediana

```
. reg lnmed lntotal, vce(robust)
```

Linear regression

Number of obs = 5006
F(1, 5004) = 318.05
Prob > F = 0.0000
R-squared = 0.0587
Root MSE = 1.5458

lnmed	Robust					
	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
lntotal	.5736545	.0321665	17.83	0.000	.510594	.636715
_cons	.9352117	.298119	3.14	0.002	.3507677	1.519656

- Un aumento en un 1% del ingreso total aumenta, en promedio, en un 0.57% el gasto médico, es decir, se estima una elasticidad de 0.57.

Regresión de mediana: p50

```
. qreg lnmed lntotal, quantile(.5) nolog
```

Median regression

Number of obs = 5006

Raw sum of deviations 6324.265 (about 6.3716121)

Min sum of deviations 6097.156

Pseudo R2 = 0.0359

lnmed	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
lntotal	.6210917	.0388194	16.00	0.000	.5449886 .6971948
_cons	.5921626	.3646869	1.62	0.104	-.1227836 1.307109

- Un aumento en un 1% del ingreso total aumenta, en la mediana, en un 0.62% el gasto médico, es decir, se estima una elasticidad de 0.62.

Regresión del percentil 10

. qreg lnmed lntotal, quantile(.1) nolog						
.1 Quantile regression					Number of obs =	5006
Raw sum of deviations 2936.097 (about 4.1743875)						
Min sum of deviations 2932.443					Pseudo R2	= 0.0012
<hr/>						
lnmed	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
lntotal	.1512009	.0552585	2.74	0.006	.0428701	.2595317
_cons	2.825072	.5194066	5.44	0.000	1.806807	3.843336

- Un aumento en un 1% del ingreso total aumenta, para el percentil 10, en un 0.15% el gasto médico, es decir, se estima una elasticidad de 0.15.

Regresión del percentil 90

```
. qreg lnmed lntotal, quantile(.9) nolog
```

.9 Quantile regression

Number of obs = 5006

Raw sum of deviations 2687.692 (about 8.2789364)

Min sum of deviations 2505.131

Pseudo R2 = 0.0679

lnmed	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
lntotal	.8003569	.0517226	15.47	0.000	.698958 .9017558
_cons	.6750967	.4857565	1.39	0.165	-.277199 1.627392

- Un aumento en un 1% del ingreso total aumenta, para el percentil 90, en un 0.8% el gasto médico, es decir, se estima una elasticidad de 0.80.

Regresión de mediana

- Naturalmente, el supuesto de normalidad para la inferencia, en este contexto no es apropiado, lo correcto es obtener los intervalos de confianza mediante bootstrap:

```
. bsqreg lnmed lntotal, quantile(.5) nolog reps(100)
(fitting base model)
(bootstrapping .....)
```

Median regression, bootstrap(100) SEs Number of obs = 5006
Raw sum of deviations 6324.265 (about 6.3716121)
Min sum of deviations 6097.156 Pseudo R2 = 0.0359

lnmed	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
lntotal	.6210917	.0454719	13.66	0.000	.5319469 .7102366
_cons	.5921626	.4279015	1.38	0.166	-.2467119 1.431037

Regresión de mediana

- El siguiente gráfico muestra la relación estimada en cada una de las regresiones:

```
. qui reg lnmed lntotal, vce(robust)

. predict predmco
(option xb assumed; fitted values)

. qui qreg lnmed lntotal, quantile(.5)

. predict predp50
(option xb assumed; fitted values)

. qui qreg lnmed lntotal, quantile(.1)

. predict predp10
(option xb assumed; fitted values)

. qui qreg lnmed lntotal, quantile(.9)

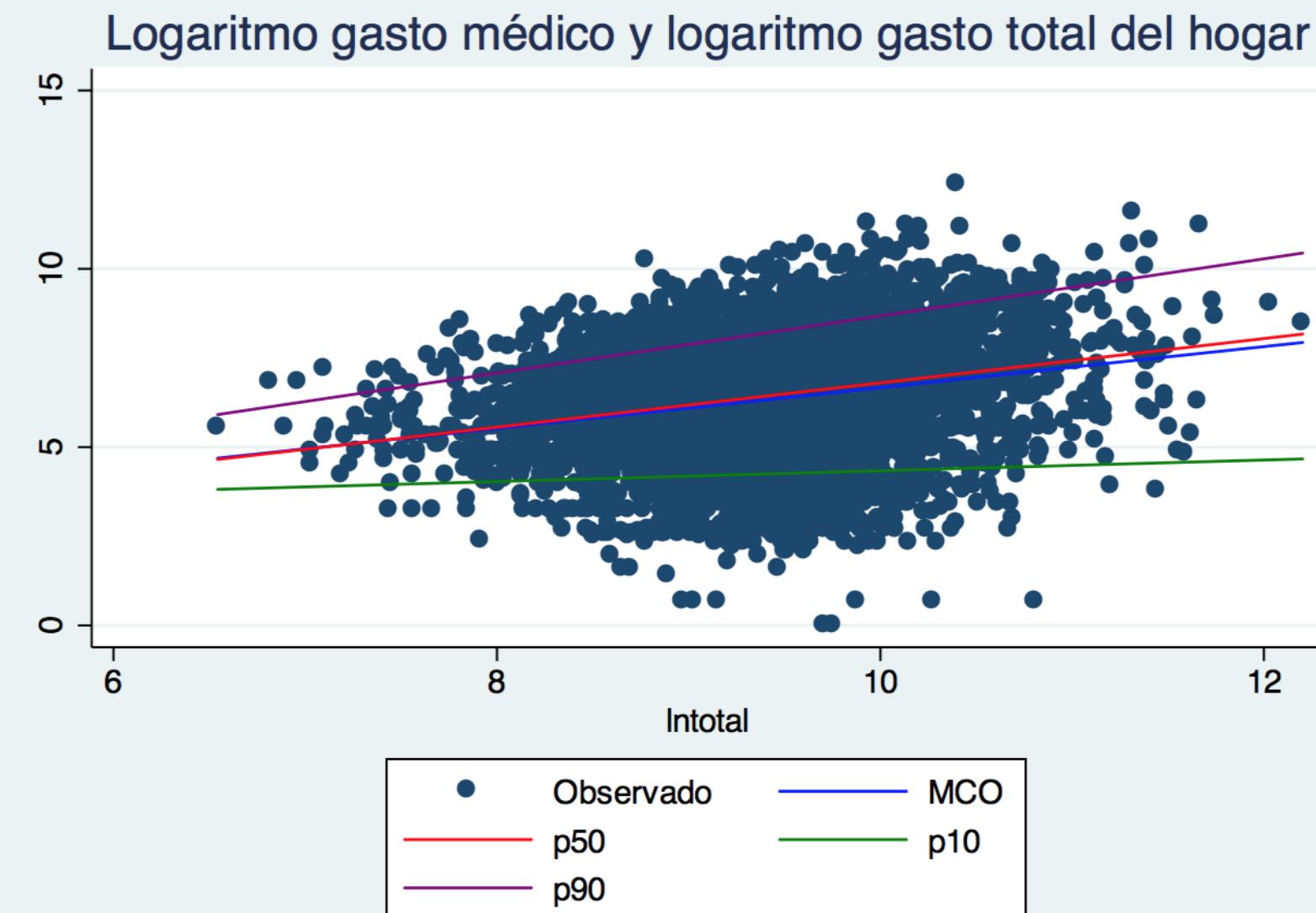
. predict predp90
(option xb assumed; fitted values)
```

Regresión de mediana

- La sintaxis para hacer el gráfico es:

```
twoway (scatter lnmed lntotal) (lfit predmco lntotal, lcolor(blue)) (lfit  
predp50 lntotal, lcolor(red)) (lfit predp10 lntotal, lcolor(green)) (lfit  
predp90 lntotal, lcolor(purple)), title(Logaritmo gasto médico y logaritmo  
gasto total del hogar) legend(order(1 "Observado" 2 "MCO" 3 "p50" 4 "p10" 5  
"p90"))
```

Regresión de mediana



Regresión de mediana

- Se podría estimar la elasticidad del gasto médico al ingreso para cada cuantil del gasto médico, mediante el siguiente proceso iterativo:

```
matrix Q=J(99,2,0)
forvalues i=0.01(0.01)1{

    qui qreg lnmed lntotal, quantile(`i')
    matrix Q[`i'*100,1]=e(q)
    matrix Q[`i'*100,2]=_b[lntotal]

}
```

- Luego se puede transformar la matriz con los resultados, en una base de datos:

```
. svmat Q, name(quantile)

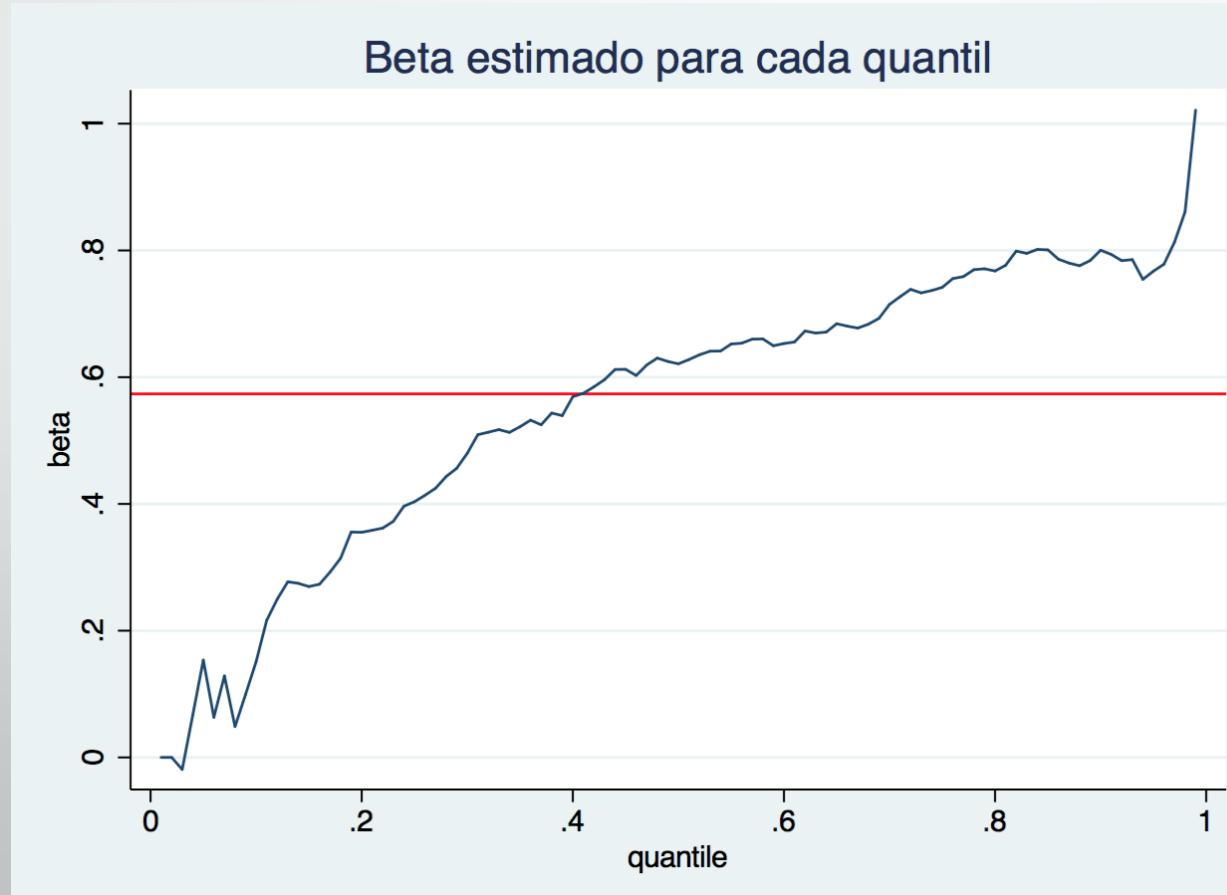
. rename quantile1 quantile

. rename quantile2 beta
```

Regresión de mediana

- Para luego graficar estas estimaciones:

```
twoway (line beta quantile, msize(vtiny) mstyle(p1) clstyle(p1)),  
yline(.5736545, lcolor(red)) title(Beta estimado para cada quantil)
```



Datos de panel

- Los datos de panel o longitudinales corresponden a mediciones repetidas de la misma unidad en diferentes momentos del tiempo.
- Usualmente los datos de panel son observados en intervalos regulares de tiempo.
- Los datos de panel pueden ser balanceados, lo que significa que todas las unidades ($i = 1, \dots, n$) son observadas en todos los momentos del tiempo ($t = 1, \dots, T$). Las estimaciones se pueden hacer tanto como paneles balanceados o no balanceados.
- Los paneles pueden ser cortos, es decir, muchos individuos y pocos momentos en el tiempo, pueden ser largos con muchos periodos de tiempo y pocos individuos, o pueden tener muchos individuos y muchos periodos de tiempo.

Datos de panel

- En modelos de datos de panel, existen altas probabilidades de que los errores estén correlacionados. En general, los métodos de estimación asumen correlación en el tiempo para un individuo (cluster de individuos) e independencia entre los individuos.
- Los regresores (variables explicativas) pueden ser invariantes en el tiempo ($x_{it} = x_i$), como por ejemplo, la variable género. Otros regresores pueden ser invariantes entre individuos ($x_{it} = x_t$), como por ejemplo una tendencia temporal, y en general las variables variarán tanto entre individuos como en el tiempo.
- Todos los comandos de STATA que se refieren a datos de panel comienzan con `xt`.
- Para poder trabajar con estos comandos primero se debe “setear” o indicar al software cual es la variable de individuos y cual es la variable de tiempo
- La base de datos debe estar en formato `long` para poder trabajar con datos de panel.

Datos de panel

- La forma general de un modelo lineal con datos de panel es:

$$y_{it} = \alpha_i + x'_{it}\beta + \varepsilon_{it}$$

- Donde α_i es lo que se conoce como el efecto individual que captura todos los no observables que no varían en el tiempo.
- Existen dos modelos fundamentales, el modelo de efecto fijo y el modelo de efecto aleatorio, la diferencia entre radica en si existe o no correlación del efecto individual con las variables explicativas del modelo.

Datos de panel

- Utilizaremos la base de datos `mus08psidextract.dta`, que contiene 4.165 observaciones, las que corresponden a observaciones para 7 períodos de tiempo para 595 individuos, es un panel balanceado.
- La base de datos está en formato `long`.
- Lo primero que debemos hacer es indicar a Stata que estamos trabajando con datos de panel:

```
. xtset id t
    panel variable: id (strongly balanced)
    time variable: t, 1 to 7
    delta: 1 unit
```

Datos de panel

- El comando `xtdescribe` entrega información sobre el panel y si está o no balanceado, en caso de no estar balanceado, lo permite caracterizar.

```
. xtdescribe
```

```
    id: 1, 2, ..., 595                                n =      595
```

```
    t: 1, 2, ..., 7                                    T =       7
```

```
    Delta(t) = 1 unit
```

```
    Span(t) = 7 periods
```

```
(id*t uniquely identifies each observation)
```

```
Distribution of T_i: min      5%     25%     50%     75%     95%   max
                           7       7       7       7       7       7       7       7
```

Freq.	Percent	Cum.	Pattern
595	100.00	100.00	1111111
595	100.00		XXXXXXX

Datos de panel

- La varianza total de una variable entorno a la media, puede ser descompuesta en la varianza dentro de cada individuo en el tiempo (within), y la varianza entre los individuos (between).
- El comando `xtsum` nos permite obtener la descomposición de la varianza, y así podremos identificar fácilmente las variables que no varían en el tiempo al interior de cada individuo. Por ejemplo:

. xtsum ed exp t							
		Variable	Mean	Std. Dev.	Min	Max	Observations
ed	overall	12.84538	2.787995	4	17	N =	4165
	between		2.790006	4	17	n =	595
	within	0	12.84538	12.84538		T =	7
exp	overall	19.85378	10.96637	1	51	N =	4165
	between		10.79018	4	48	n =	595
	within	2.00024	16.85378	22.85378		T =	7
t	overall	4	2.00024	1	7	N =	4165
	between		0	4	4	n =	595
	within	2.00024	1	7		T =	7

Datos de panel

- El comando `xttab` tabula las variables, pero entregando información sobre la variación `within` y `between` de la variable.
- Por ejemplo:

```
. xttab south
```

south	Overall		Between		Within
	Freq.	Percent	Freq.	Percent	Percent
0	2956	70.97	428	71.93	98.66
1	1209	29.03	182	30.59	94.90
Total	4165	100.00	610	102.52	97.54
	(n = 595)				

Datos de panel

- El resultado nos muestra que de las 4.165 observaciones, 71% corresponden al sur.
- El resultado `between` de la tabla muestra que de las 595 personas, 72% tiene la variable `south` igual a cero al menos una vez, y 31% tiene la variable `south` igual a uno al menos una vez.
- Finalmente, el resultado `within` de la tabla muestra que 95% de las personas que han vivido en el sur, han vivido siempre en el sur durante el periodo de tiempo cubierto por el panel. Y un 99% de las personas que han vivido fuera del sur, han vivido siempre fuera del sur.
- De esta forma, esta tabla nos muestra que la variable `south` es casi como si no variara en el tiempo.

Estimador de Pooled

- Corresponde al estimador de MCO aplicado en los datos de panel.
- En este caso se asume que el efecto individual es constante e igual para todas las unidades, es decir, que no existe efecto individual.
- No se explota la estructura de datos de panel, más allá del aumento en la cantidad de observaciones.
- Además, se asume que todas las observaciones son independientes entre ellas, es decir, se asume que el término de error es independiente en i y en t .

Estimador de Pooled

```
. reg lwage exp exp2 wks ed, vce(robust)
```

Linear regression

Number of obs = 4165
F(4, 4160) = 338.23
Prob > F = 0.0000
R-squared = 0.2836
Root MSE = .39082

lwage	Robust					
	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
exp	.044675	.0025668	17.41	0.000	.0396428	.0497073
exp2	-.0007156	.0000592	-12.09	0.000	-.0008317	-.0005995
wks	.005827	.0012855	4.53	0.000	.0033066	.0083473
ed	.0760407	.0023677	32.12	0.000	.0713988	.0806826
_cons	4.907961	.077262	63.52	0.000	4.756486	5.059436

Estimador de Pooled

- Es muy probable que los modelos de datos de panel, tengan errores con autocorrelación, ya que es probable que los no observables que varían en el tiempo estén correlacionados entre ellos al interior de una unidad (individuo).
- Drukker (2003) elaboró un test de autocorrelación en modelos de datos de paneles lineales.
- Hay que instalar el comando de Stata desarrollado por este autor.

```
. findit xtserial
```

```
. net install st0039
checking st0039 consistency and verifying not already installed...
installing into c:\ado\plus\...
installation complete.
```

Estimador de Pooled

- Si hacemos el test con el comando `xtserial`, se rechaza la hipótesis nula lo que nos muestra claramente que tenemos problemas de autocorrelación.

```
. xtserial lwage exp exp2 wks ed

Wooldridge test for autocorrelation in panel data
H0: no first-order autocorrelation
    F( 1,      594) =      25.829
                  Prob > F =      0.0000
```

Estimador de Pooled

- El estimador MCO asume que los errores son independientes entre ellos (no existe autocorrelación) y que se distribuyen normal.
- La opción `robust` relaja el supuesto de que los errores tienen idéntica distribución, de esta forma, cuando hay problemas de heterocedasticidad es una opción más confiable.
- La opción `cluster`, sin embargo, relaja el supuesto de que los errores son independientes entre ellos.

`vce(cluster clustvar)` specifies that the standard errors allow for intragroup correlation, relaxing the usual requirement that the observations be independent. That is to say, the observations are independent across groups (clusters) but not necessarily within groups. *clustvar* specifies to which group each observation belongs, for example, `vce(cluster personid)` in data with repeated observations on individuals. `vce(cluster clustvar)` affects the standard errors and variance-covariance matrix of the estimators but not the estimated coefficients; see [U] **20.21 Obtaining robust variance estimates**.

Comparación Pooled con distintas varianzas

```
. qui reg lwage exp exp2 wks ed  
  
. estimates store pooled  
  
. qui reg lwage exp exp2 wks ed, vce(robust)  
  
. estimates store pooledr  
  
. qui reg lwage exp exp2 wks ed, vce(cluster id)  
  
. estimates store pooledc
```

```
. estimates table pooled pooledr pooledc, b(%7.4f) se(%7.4f)
```

Variable	pooled	pooledr	pooledc
exp	0.0447	0.0447	0.0447
	0.0024	0.0026	0.0054
exp2	-0.0007	-0.0007	-0.0007
	0.0001	0.0001	0.0001
wks	0.0058	0.0058	0.0058
	0.0012	0.0013	0.0019
ed	0.0760	0.0760	0.0760
	0.0022	0.0024	0.0052
_cons	4.9080	4.9080	4.9080
	0.0673	0.0773	0.1400

legend: b/se

Estimador de Efecto Fijo

- En el estimador de efecto fijo, el efecto individual se asume una constante, de esta forma se permite que exista correlación entre el efecto individual y las variables explicativas del modelo.
- Al igual que en el modelo pooled, debido a la presencia de autocorrelación en los errores, se debe utilizar la opción `cluster` por id para la estimación de las varianzas.
- El estimador de efecto fijo es equivalente, en la estimación de coeficientes, a incluir variables *dummies* para cada uno de los individuos, pero esta estimación es menos eficiente, ya que estima una mayor cantidad de coeficientes.

Estimador de Efecto Fijo

```
. xtreg lwage exp exp2 wks ed, fe vce(cluster id)
note: ed omitted because of collinearity

Fixed-effects (within) regression                               Number of obs     =     4165
Group variable: id                                         Number of groups  =      595

R-sq:  within  =  0.6566                                     Obs per group: min =       7
          between =  0.0276                                     avg =      7.0
          overall =  0.0476                                     max =       7

                                                F(3,594)           =   1059.72
corr(u_i, Xb)  = -0.9107                                     Prob > F        =  0.0000

                                                (Std. Err. adjusted for 595 clusters in id)
```

lwage	Robust					
	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
exp	.1137879	.0040289	28.24	0.000	.1058753	.1217004
exp2	-.0004244	.0000822	-5.16	0.000	-.0005858	-.0002629
wks	.0008359	.0008697	0.96	0.337	-.0008721	.0025439
ed	0	(omitted)				
_cons	4.596396	.0600887	76.49	0.000	4.478384	4.714408
sigma_u	1.0362039					
sigma_e	.15220316					
rho	.97888036					

98% de la varianza se debe a diferencias a través tiempo (dentro de cada unidad): correlación intra-clase

No puede ser estimado debido a que no varía en el tiempo para cada individuo.

Estimador de Efecto Fijo versus dummies

```
. qui xtreg lwage exp exp2 wks ed, fe vce(cluster id)  
  
. estimates store fixed  
  
. qui areg lwage exp exp2 wks ed, absorb(id) vce(cluster id)  
  
. estimates store dummies  
  
. estimates table fixed dummies, b(%7.4f) se(%7.4f)
```

Variable	fixed	dummies
exp	0.1138 0.0040	0.1138 0.0044
exp2	-0.0004 0.0001	-0.0004 0.0001
wks	0.0008 0.0009	0.0008 0.0009
ed	(omitted)	(omitted)
_cons	4.5964 0.0601	4.5964 0.0649

legend: b/se

Estimador de Efecto Aleatorio

- En el estimador de efecto aleatorio, el efecto individual se asume como un variable aleatoria que forma parte del término de error, de esta forma no puede estar correlacionado con las variables explicativas del modelo.
- En este caso el error tiene cierta estructura de correlación:

$$\text{corr}(u_{it}, u_{is}) = \frac{\sigma_\alpha^2}{\sigma_\alpha^2 + \sigma_\epsilon^2}$$

- De esta forma, el estimador de efecto aleatorio es equivalente a un estimador de MGF.

Estimador de Efecto Aleatorio

```
. xtreg lwage exp exp2 wks ed, re vce(cluster id) theta
```

Random-effects GLS regression
Number of obs = 4165
Group variable: id Number of groups = 595

R-sq: within = 0.6340 Obs per group: min = 7
between = 0.1716 avg = 7.0
overall = 0.1830 max = 7

Wald chi2(4) = 1598.50
corr(u_i, X) = 0 (assumed)
Prob > chi2 = 0.0000
theta = .82280511

(Std. Err. adjusted for 595 clusters in id)

lwage	Robust					
	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
exp	.0888609	.0039992	22.22	0.000	.0810227	.0966992
exp2	-.0007726	.0000896	-8.62	0.000	-.0009481	-.000597
wks	.0009658	.0009259	1.04	0.297	-.000849	.0027806
ed	.1117099	.0083954	13.31	0.000	.0952552	.1281647
_cons	3.829366	.1333931	28.71	0.000	3.567921	4.090812
sigma_u	.31951859					
sigma_e	.15220316					
rho	.81505521					(fraction of variance due to u_i)

Comparación de estimadores

```
. qui xtreg lwage exp exp2 wks ed, re vce(cluster id)  
  
. estimates store random  
  
. qui reg lwage exp exp2 wks ed, vce(robust)  
  
. estimates store pooledr  
  
. qui xtreg lwage exp exp2 wks ed, fe vce(cluster id)  
  
. estimates store fixed  
  
. qui xtreg lwage exp exp2 wks ed, re vce(cluster id)  
  
. estimates store random  
  
. estimates table pooledr fixed random, b(%7.4f) se(%7.4f)
```

Variable	pooledr	fixed	random
exp	0.0447	0.1138	0.0889
	0.0026	0.0040	0.0040
exp2	-0.0007	-0.0004	-0.0008
	0.0001	0.0001	0.0001
wks	0.0058	0.0008	0.0010
	0.0013	0.0009	0.0009
ed	0.0760	(omitted)	0.1117
	0.0024		0.0084
_cons	4.9080	4.5964	3.8294
	0.0773	0.0601	0.1334

legend: b/se

¿Efecto fijo o aleatorio?: Test de Hausmann

- Recordemos que el estimador de efecto aleatorio asume que el efecto individual no está correlacionado con las variables explicativas, mientras que el estimador de efecto fijo, permite que exista esta correlación.
- Si los efectos individuales son fijos (están correlacionados con las variables explicativas) el estimador de efecto aleatorio es inconsistente, y se debería utilizar el estimador de efecto fijo (*within*).
- Sin embargo, el estimador *within* es menos deseable que el de efecto aleatorio ya que al utilizar sólo la variación intra grupo (*within*) las estimaciones son menos eficientes, y no se pueden estimar coeficientes para las variables que no varían en el tiempo.

¿Efecto fijo o aleatorio?: Test de Hausman

- Bajo la hipótesis nula de que los efectos individuales son aleatorios, los estimadores de efecto fijo y efecto aleatorio deberían obtener coeficientes estimados muy similares, ya que ambos estimadores son consistentes en este escenario, siendo el de efecto aleatorio más eficiente.
- Bajo la hipótesis alternativa, el estimador de efecto aleatorio es inconsistente.
- El test de Hausman justamente compara ambos estimadores, teniendo como hipótesis nula que son iguales, y por lo tanto que el modelo de efecto aleatorio es el apropiado.

¿Efecto fijo o aleatorio?: Test de Hausman

```
. qui xtreg lwage exp exp2 wks ed, fe  
  
. estimates store fijo  
  
. qui xtreg lwage exp exp2 wks ed, re  
  
. estimates store aleatorio  
  
. hausman fijo aleatorio, sigmamore
```

	Coefficients		(b-B) Difference	sqrt(diag(V_b-V_B)) S.E.
	(b) fijo	(B) aleatorio		
exp	.1137879	.0888609	.0249269	.0012778
exp2	-.0004244	-.0007726	.0003482	.0000285
wks	.0008359	.0009658	-.0001299	.0001108

b = consistent under H_0 and H_a ; obtained from xtreg

B = inconsistent under H_a , efficient under H_0 ; obtained from xtreg

Test: H_0 : difference in coefficients not systematic

$$\begin{aligned} \text{chi2}(3) &= (\mathbf{b}-\mathbf{B})' [(\mathbf{V}_b-\mathbf{V}_B)^{-1}] (\mathbf{b}-\mathbf{B}) \\ &= 1513.02 \end{aligned}$$

Prob>chi2 = 0.0000