Escuela Nacional de Estadística e Informática

SOFTWARE "R"

Lima – Perú

www.inei.gob.pe

Regresión Lineal Simple y Múltiple

Contenido

1.1 Introducción

Vamos a utilizar un archivo de datos que fue creado por un muestreo aleatorio de 400 escuelas primarias del Departamento de Educación. Este archivo de datos contiene una medida del rendimiento académico, así como otros atributos de las escuelas primarias, tales como, el tamaño de la matrícula, la pobreza, etc.

Puede acceder a este archivo de datos, que se encuentra en la web, desde el interior de Stata con el comando ***use*** de Stata, como se muestra a continuación.

use http://www.ats.ucla.edu/stat/stata/webbooks/reg/elemapi

Una vez que haya leído el archivo, es probable que desee guardar una copia del mismo en el equipo (por lo que no es necesario leerlo en la web cada vez). Digamos que usted está utilizando Windows y desea almacenar el archivo en una carpeta llamada c:\regstata (se puede elegir un nombre diferente si lo desea). En primer lugar, puede crear esta carpeta desde el Stata con el comando mkdir.

mkdir c:\regstata

A continuación, puede cambiar a ese directorio con el comando cd.

cd c:\regstata

Luego, si se guarda el archivo se guardará en la carpeta c:\regstata. Vamos a guardar el archivo como elemapi.

save elemapi

1.2 Un primer análisis de regresión

Vamos a explorar y realizar un análisis de regresión con las variables ***api00***, ***acs_k3***, ***meals*** y ***full***. Estas miden el rendimiento académico de la escuela (api00), tamaño promedio de jardín de infantes hasta 3er grado (acs_k3), el porcentaje de estudiantes que reciben comidas gratis

(comidas), que es un indicador de la pobreza, y el porcentaje de profesores con credenciales completas de enseñanza (completo). Esperamos que un mayor rendimiento académico se asocie con un menor tamaño de la clase, y con un menor porcentaje de estudiantes que reciben comidas gratis, y con un mayor porcentaje de docentes con credenciales completas de enseñanza. A continuación, se muestra el comando de Stata para las pruebas de este modelo de regresión seguido por la salida de Stata.

regress api00 acs_k3 meals full

```
      Source |       SS       df       MS              Number of obs =     313
-------------+------------------------------           F(  3,   309) =  213.41
       Model |  2634884.26     3  878294.754           Prob > F      =  0.0000
    Residual |  1271713.21   309  4115.57673           R-squared     =  0.6745
-------------+------------------------------           Adj R-squared =  0.6713
       Total |  3906597.47   312  12521.1457           Root MSE      =  64.153

------------------------------------------------------------------------------
       api00 |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      acs_k3 |  -2.681508   1.393991    -1.92   0.055    -5.424424    .0614073
       meals |  -3.702419   .1540256   -24.04   0.000    -4.005491   -3.399348
        full |   .1086104    .090719     1.20   0.232    -.0698947    .2871154
       _cons |   906.7392   28.26505    32.08   0.000     851.1228    962.3555
------------------------------------------------------------------------------
```

Vamos a centrarnos en los tres predictores, tanto si son estadísticamente significativas y, en tal caso, la dirección de la relación. El promedio de alumnos (acs_k3, b=- 2,68), no es significativa (p = 0,055), pero por un valor ajustado. El coeficiente es negativo lo que indicaría que un mayor tamaño de clase se relaciona con menor rendimiento académico, que es lo que cabría esperar. A continuación, el efecto de las comidas (b=-3,70, p =. 000) es significativa y su coeficiente es negativo lo que indica que cuanto mayor es la proporción de estudiantes que reciben comidas gratuitas, menor es el rendimiento académico. Tenga en cuenta que no estamos diciendo que las comidas gratuitas están causando bajo rendimiento académico. La variable de las comidas está muy relacionada con el nivel de ingresos y funciona más como un indicador de la pobreza. Por lo tanto, mayores niveles de pobreza se asocia con un menor rendimiento académico. Este resultado también tiene sentido. Finalmente, el porcentaje de maestros con credenciales completas (full, b=0.11, p =. 232) parece no estar relacionado con el rendimiento académico. Esto parece indicar que el porcentaje de maestros con credenciales completas no es un factor importante para predecir el rendimiento académico, el resultado fue algo inesperado.

¿Debemos tomar estos resultados y describirlos para su publicación? A partir de estos resultados, podemos concluir que la reducción del tamaño de las clases están relacionadas con un mayor rendimiento, menos estudiantes que reciben comidas gratuitas se asocia con un mayor rendimiento, y que el porcentaje de maestros con credenciales completas no estaba relacionado con el rendimiento académico en las escuelas. Antes de escribir esto para su publicación, debemos hacer una serie de comprobaciones para asegurarnos que los resultados sean consistentes con los supuestos del modelo.

1.3 Examinando los datos

En primer lugar, vamos a usar el comando ***describe*** para aprender más acerca de este archivo de datos. Podemos comprobar cuantas observaciones tiene y ver los nombres de las variables que contiene. Para hacer esto, simplemente escriba

describe

```
Contains data from http://www.ats.ucla.edu/stat/stata/webbooks/reg/elemapi.dta
  obs:           400
 vars:            21                          25 Feb 2001 16:58
 size:        14,800 (92.3% of memory free)
-------------------------------------------------------------------------------
              storage  display     value
variable name   type   format      label      variable label
-------------------------------------------------------------------------------
snum            int    %9.0g                  school number
dnum            int    %7.0g       dname      district number
api00           int    %6.0g                  api 2000
api99           int    %6.0g                  api 1999
growth          int    %6.0g                  growth 1999 to 2000
meals           byte   %4.0f                  pct free meals
ell             byte   %4.0f                  english language learners
yr_rnd          byte   %4.0f       yr_rnd     year round school
mobility        byte   %4.0f                  pct 1st year in school
acs_k3          byte   %4.0f                  avg class size k-3
acs_46          byte   %4.0f                  avg class size 4-6
not_hsg         byte   %4.0f                  parent not hsg
hsg             byte   %4.0f                  parent hsg
some_col        byte   %4.0f                  parent some college
col_grad        byte   %4.0f                  parent college grad
grad_sch        byte   %4.0f                  parent grad school
avg_ed          float  %9.0g                  avg parent ed
full            float  %4.0f                  pct full credential
emer            byte   %4.0f                  pct emer credential
enroll          int    %9.0g                  number of students
mealcat         byte   %18.0g      mealcat    Percentage free meals in 3
                                                categories
-------------------------------------------------------------------------------
Sorted by:  dnum
```

No vamos a entrar en todos los detalles de esta salida. Tenga en cuenta que hay 400 observaciones y 21 variables. Tenemos las variables sobre el rendimiento académico en los años 2000 y 1999 y el cambio en el rendimiento, ***api00***, ***api99*** y ***growth***, respectivamente. También tenemos diferentes características de las escuelas, por ejemplo, tamaño de la clase, educación de los padres, el porcentaje de maestros con credenciales completas y de emergencia, y el número de estudiantes. Tenga en cuenta que cuando hicimos el análisis de regresión, se mostró que había 313 observaciones, pero el comando ***describe*** indica que tenemos 400 observaciones en el archivo de datos.

Si desea obtener más información sobre el archivo de datos, se puede enumerar todas o algunas de las observaciones. Por ejemplo, a continuación se enumeran las cinco primeras observaciones.

list in 1/5

```
Observation 1

        snum          906        dnum           41       api00          693
       api99          600      growth           93       meals           67
         ell            9      yr_rnd           No    mobility           11
      acs_k3           16      acs_46           22     not_hsg            0
         hsg            0    some_col            0    col_grad            0
    grad_sch            0      avg_ed            .        full        76.00
        emer           24      enroll          247     mealcat  47-80% free

Observation 2

        snum          889        dnum           41       api00          570
       api99          501      growth           69       meals           92
         ell           21      yr_rnd           No    mobility           33
      acs_k3           15      acs_46           32     not_hsg            0
         hsg            0    some_col            0    col_grad            0
    grad_sch            0      avg_ed            .        full        79.00
        emer           19      enroll          463     mealcat 81-100% free

Observation 3

        snum          887        dnum           41       api00          546
       api99          472      growth           74       meals           97
         ell           29      yr_rnd           No    mobility           36
      acs_k3           17      acs_46           25     not_hsg            0
         hsg            0    some_col            0    col_grad            0
    grad_sch            0      avg_ed            .        full        68.00
        emer           29      enroll          395     mealcat 81-100% free

Observation 4

        snum          876        dnum           41       api00          571
       api99          487      growth           84       meals           90
         ell           27      yr_rnd           No    mobility           27
      acs_k3           20      acs_46           30     not_hsg           36
         hsg           45    some_col            9    col_grad            9
    grad_sch            0      avg_ed         1.91        full        87.00
        emer           11      enroll          418     mealcat 81-100% free

Observation 5

        snum          888        dnum           41       api00          478
       api99          425      growth           53       meals           89
         ell           30      yr_rnd           No    mobility           44
      acs_k3           18      acs_46           31     not_hsg           50
         hsg           50    some_col            0    col_grad            0
    grad_sch            0      avg_ed          1.5        full        87.00
        emer           13      enroll          520     mealcat 81-100% free
```

Esto ocupa mucho espacio en la página, pero no nos da mucha información. El listado de los datos puede ser muy útil, pero es más útil si listamos sólo las variables de interés. Vamos a explorar las primeras 10 observaciones de las variables que vimos en nuestro análisis de regresión anterior.

list api00 acs_k3 meals full in 1/10

```
     api00  acs~3  meals    full
  1.   693    16    67     76.00
  2.   570    15    92     79.00
  3.   546    17    97     68.00
  4.   571    20    90     87.00
  5.   478    18    89     87.00

  6.   858    20     .    100.00
  7.   918    19     .    100.00
  8.   831    20     .     96.00
  9.   860    20     .    100.00
 10.   737    21    29     96.00
```

Vemos que entre las 10 primeras observaciones, tenemos cuatro valores que faltan para las comidas. Es probable que los datos que faltan para las comidas tengan algo que ver con el hecho de que el número de observaciones en el análisis de regresión anterior fue de 313 y no 400.

Otra herramienta útil para conocer las variables es el libro de códigos. Vamos a generar el libro de códigos para las variables que se incluyeron en el análisis de regresión, así como la variable ***yr_rnde***. Algunos comentarios sobre esta salida aparecen entre corchetes y en negrita.

codebook api00 acs_k3 meals full yr_rnd

```
api00 ------------------------------------------------------------------ api 2000
                  type:  numeric (int)

                 range:  [369,940]                    units:  1
         unique values:  271                  coded missing:  0 / 400

                  mean:   647.622
             std. dev:   142.249

          percentiles:        10%       25%       50%       75%       90%
                            465.5     523.5       643     762.5       850
[Los resultados muestran que API no tiene valores perdidos, y su rango de valores va de
369 hasta 940]
[Esto tiene sentido ya que las puntuaciones de API van de 200 a 1000]

acs_k3 ----------------------------------------------------- avg class size k-3
                  type:  numeric (byte)

                 range:  [-21,25]                     units:  1
         unique values:  14                   coded missing:  2 / 400

                  mean:   18.5477
             std. dev:   5.00493

          percentiles:        10%       25%       50%       75%       90%
                              17        18        19        20        21
[En el rango de valores de la variable promedio de alumnos por salón (-21,25), 2 son
valores perdidos.]
[Un tamaño de clase no puede tener como valor a -21]
```

```
meals ------------------------------------------------------------ pct free meals
                  type:  numeric (byte)

                 range:  [6,100]                      units:  1
         unique values:  80                   coded missing:  85 / 400

                  mean:   71.9937
             std. dev:   24.3856

           percentiles:        10%       25%       50%       75%       90%
                               33        57        77        93        99
```

[El rango de valores de la variable porcentaje que reciben comidas gratis, está entre 6 a 100, pero faltan 85 valores. Existen valores perdidos]
[¡Esto puede ser una cantidad alta de valores faltantes!]

```
full ------------------------------------------------------- pct full credential
                  type:  numeric (float)

                 range:  [.42,100]                    units:  .01
         unique values:  92                   coded missing:  0 / 400

                  mean:   66.0568
             std. dev:   40.2979

           percentiles:        10%       25%       50%       75%       90%
                                67       .95        87        97       100
```

[El porcentaje de acreditados va de 0.42 hasta 100 y no hay valores perdidos]

```
yr_rnd ----------------------------------------------------- year round school
                  type:  numeric (byte)
                 label:  yr_rnd

                 range:  [0,1]                        units:  1
         unique values:  2                    coded missing:  0 / 400

            tabulation:  Freq.   Numeric  Label
                          308         0  No
                           92         1  Yes
```

[La variable yr_rnd se codifica 0 = No (no todo el año) y 1 = Sí (durante todo el año)]
[308 son no todo el año y 92 durante todo el año, y no hay valores faltantes]

El comando ***codebook*** ha puesto al descubierto una serie de peculiaridades que merecen un examen más minucioso. Vamos a usar el comando resumen para conocer más sobre estas variables. Como se muestra a continuación, el comando también muestra un resumen de la alta cantidad de valores que faltan para la variable *meals* (400 - 315 = 85) y vemos una valor raro (mínimo) en la variable *acs_k3* que es -21.

summarize api00 acs_k3 meals full

```
    Variable |     Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
       api00 |     400    647.6225     142.249        369        940
      acs_k3 |     398    18.54774    5.004933        -21         25
       meals |     315    71.99365    24.38557          6        100
        full |     400     66.0568    40.29793         .42        100
```

Vamos a obtener un resumen más detallado de acs_k3. En Stata, la coma después de la lista de variables indica las opciones a seguir, en este caso, la opción es el detalle. Como se puede ver a

continuación, la opción de detalle da los percentiles, los cuatro mayores y menores valores, medidas de valor central y de la varianza, etc. Tenga en cuenta que el comando *summarize*, y otros comandos, se pueden abreviar: se podría haber escrito *sum acs_k3, d*.

summarize acs_k3, detail

```
                    avg class size k-3
-------------------------------------------------------------
      Percentiles      Smallest
 1%           -20            -21
 5%            16            -21
10%            17            -21       Obs                 398
25%            18            -20       Sum of Wgt.         398

50%            19                      Mean           18.54774
                        Largest       Std. Dev.      5.004933
75%            20              23
90%            21              23       Variance       25.04935
95%            21              23       Skewness      -7.078785
99%            23              25       Kurtosis       55.33497
```

Por alguna razón existen valores negativos en la variable ***acs_k3***; es como si un signo negativo se ha escrito incorrectamente delante de las cifras. Hagamos una tabulación del tamaño de la clase para ver que encontramos.

tabulate acs_k3

```
   avg class |

   size k-3 |      Freq.     Percent        Cum.
------------+-----------------------------------
        -21 |          3        0.75        0.75
        -20 |          2        0.50        1.26
        -19 |          1        0.25        1.51
         14 |          2        0.50        2.01
         15 |          1        0.25        2.26
         16 |         14        3.52        5.78
         17 |         20        5.03       10.80
         18 |         64       16.08       26.88
         19 |        143       35.93       62.81
         20 |         97       24.37       87.19
         21 |         40       10.05       97.24
         22 |          7        1.76       98.99
         23 |          3        0.75       99.75
         25 |          1        0.25      100.00
------------+-----------------------------------
      Total |        398      100.00
```

Echemos un vistazo a la escuela y el número de distrito al cual pertenecen estas observaciones para ver si proceden de un mismo distrito. Efectivamente, todos ellos corresponden al distrito 140.

list snum dnum acs_k3 if acs_k3 < 0

```
          snum     dnum  acs~3

 37.       602      140   -21
 96.       600      140   -20
```

```
173.       595       140   -21
223.       596       140   -19
229.       611       140   -20
282.       592       140   -21
```

Echemos un vistazo a todas las observaciones para el distrito 140.

list dnum snum api00 acs_k3 meals full if dnum == 140

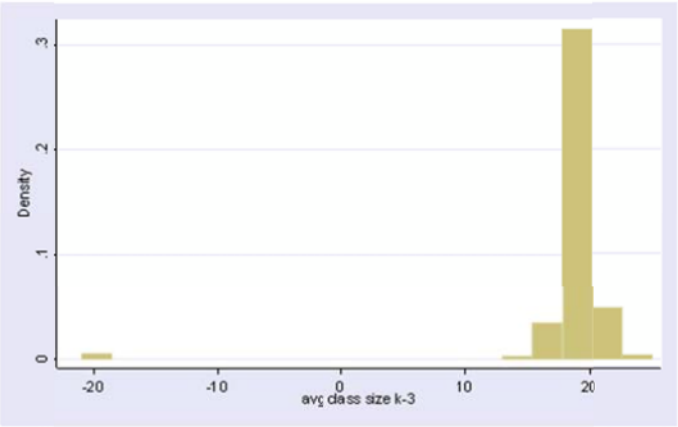
```
        dnum       snum    api00  acs~3  meals     full
 37.     140        602      864    -21      .   100.00
 96.     140        600      843    -20      .    91.00
173.     140        595      713    -21     63    92.00
223.     140        596      800    -19      .    94.00
229.     140        611      857    -20      .   100.00
282.     140        592      804    -21      .    97.00
```

Todas las observaciones del distrito 140 parecen tener este problema. Cuando se encuentra un problema, se necesita volver a la fuente original de los datos para verificar los valores. Tenemos que averiguar que genera este error, y que los datos reales no tenían el problema detectado. Imaginemos que nos registramos con el distrito 140 y que había un problema con los datos allí, un guión se puso accidentalmente delante de las cifras haciéndolas negativas.

Echemos un vistazo a algunos métodos gráficos para la inspección de los datos. Para cada variable, es útil el uso de un histograma, diagrama de caja, y diagramas de tallo y hoja. Estos gráficos pueden mostrar información sobre la forma de las variables más que las simples estadísticas numéricas. Ya sabemos el problema con ***acs_k3***, pero vamos a ver cómo estos métodos gráficos ponen de manifiesto el problema con esta variable.

En primer lugar, se muestra un histograma para acs_k3. Esto nos muestra las observaciones negativas.

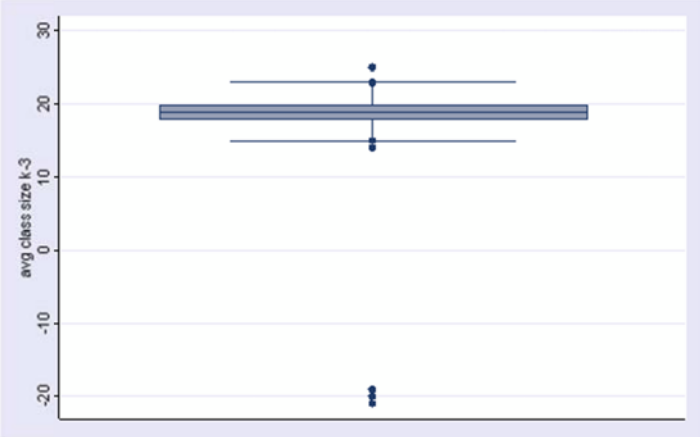
histogram acs_k3



Del mismo modo, un diagrama de caja también habría puesto de manifiesto a estas

observaciones negativas. Usted puede ver en la parte inferior del gráfico las observaciones negativas.

graph box acs_k3



Finalmente, un diagrama de tallo y hojas también ayuda a identificar estas observaciones. Este gráfico muestra los valores exactos de las observaciones, lo que indica que hay tres -21, dos 20, y un -19 años.

stem acs_k3

```
 -2* | 11100
 -1. | 9
 -1s |
 -1f |
 -1t |
 -1* |
 -0. |
 -0s |
 -0f |
 -0t |
 -0* |
  0* |
  0t |
  0f |
  0s |
  0. |
  1* |
  1t |
  1f | 445
  1s | 66666666666667777777777777777777
  1. | 88888888888888888888888888888888888888888888888888888888888888 ... (207)
  2* | 00000000000000000000000000000000000000000000000000000000000000 ... (137)
  2t | 2222222333
  2f | 5
```

Se recomienda generar todos estos gráficos de las variables que se vaya a analizar. Vamos a omitir, por razones de espacio, mostrar las gráficas de todas las variables. Sin embargo, en el examen de las variables, el diagrama de tallo y hojas para la variable ***full*** parece inusual. Hasta

ahora, no hemos visto nada problemático en esta variable, pero si observamos el gráfico de tallo y hojas para esta variable, observamos algo inusual. Muestra 104 observaciones en las que el porcentaje que están totalmente acreditados es menor que uno. Esto es más del 25% de las escuelas, y esto parece muy raro.

stem full

```
Stem-and-leaf plot for full (pct full credential)

full rounded to nearest multiple of .1
plot in units of .1

   0** | 04,04,05,05,05,05,05,05,05,05,05,06,06,06,06,06,06,06,06, ... (104)
   0** |
   0** |
   0** |
   0** |
   1** |
   1** |
   1** |
   1** |
   1** |
   2** |
   2** |
   2** |
   2** |
   2** |
   2** |
   3** |
   3** |
   3** |
   3** | 70
   3** |
   4** | 10
   4** |
   4** | 40,40,50,50
   4** | 60
   4** | 80
   5** |
   5** | 30
   5** |
   5** | 70
   5** | 80,80,80,90
   6** | 10
   6** | 30,30
   6** | 40,50
   6** |
   6** | 80,80,90,90,90
   7** | 00,10,10,10
   7** | 20,30,30
   7** | 40,50,50,50,50
   7** | 60,60,60,60,70,70
   7** | 80,80,80,80,90,90,90
   8** | 00,00,00,00,00,00,00,00,00,00,10,10,10,10
   8** | 20,20,20,30,30,30,30,30,30,30,30,30
   8** | 40,40,40,40,50,50,50,50,50,50,50,50
   8** | 60,60,60,60,60,70,70,70,70,70,70,70,70,70,70,70
   8** | 80,80,80,80,80,80,90,90,90,90,90
   9** | 00,00,00,00,00,00,00,00,00,10,10,10,10,10,10,10,10
   9** | 20,20,20,20,20,20,20,30,30,30,30,30,30,30,30,30,30,30,30
   9** | 40,40,40,40,40,40,40,40,40,40,50,50,50,50,50,50,50,50,50,50, ... (27)
   9** | 60,60,60,60,60,60,60,60,60,60,60,60,60,60,60,60,70,70,70, ... (28)
   9** | 80,80,80,80,80,80,80,80,80
  10** | 00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00, ... (81)
```

Echemos un vistazo a la distribución de frecuencias de la variable ***full*** para ver si podemos entender lo anterior. Los valores van desde 0,42 hasta 1,0, y luego salta a 37 y desde allí

empieza a subir. Parece como si algunas de las cifras están en proporciones y no en porcentaje, por ejemplo, 0,42 se ha introducido en lugar de 42 o 0,96, que realmente debería haber sido 96.

tabulate full

pct full credential	Freq.	Percent	Cum.
0.42	1	0.25	0.25
0.85	1	0.25	0.50
0.86	1	0.25	0.75
0.87	1	0.25	1.00
0.88	1	0.25	1.25
0.90	3	0.75	2.00
0.91	1	0.25	2.25
0.92	1	0.25	2.50
0.93	1	0.25	2.75
0.94	1	0.25	3.00
0.96	2	0.50	3.50
0.97	2	0.50	4.00
0.98	1	0.25	4.25
0.99	2	0.50	4.75
0.60	1	0.25	5.00
0.61	1	0.25	5.25
0.62	4	1.00	6.25
0.63	2	0.50	6.75
0.64	1	0.25	7.00
0.65	1	0.25	7.25
0.66	2	0.50	7.75
0.67	5	1.25	9.00
0.68	3	0.75	9.75
0.69	1	0.25	10.00
0.70	1	0.25	10.25
0.71	2	0.50	10.75
0.72	2	0.50	11.25
0.73	5	1.25	12.50
0.75	4	1.00	13.50
0.76	3	0.75	14.25
0.77	2	0.50	14.75
0.79	3	0.75	15.50
0.80	5	1.25	16.75
73.00	2	0.50	33.75
74.00	1	0.25	34.00
75.00	4	1.00	35.00
76.00	4	1.00	36.00
77.00	2	0.50	36.50
78.00	4	1.00	37.50
79.00	3	0.75	38.25
80.00	10	2.50	40.75
81.00	4	1.00	41.75
82.00	3	0.75	42.50
83.00	9	2.25	44.75
84.00	4	1.00	45.75
85.00	8	2.00	47.75
86.00	5	1.25	49.00
87.00	12	3.00	52.00
88.00	6	1.50	53.50
89.00	5	1.25	54.75
90.00	9	2.25	57.00
91.00	8	2.00	59.00
92.00	7	1.75	60.75
93.00	12	3.00	63.75
94.00	10	2.50	66.25
95.00	17	4.25	70.50
96.00	17	4.25	74.75
97.00	11	2.75	77.50
98.00	9	2.25	79.75

```
     100.00 |          81       20.25      100.00
------------+-----------------------------------
      Total |         400      100.00
```

Vamos a ver de qué distrito(s) provienen estos datos.

tabulate dnum if full <= 1

```
   district |
     number |      Freq.     Percent        Cum.
------------+-----------------------------------
        401 |         104      100.00      100.00
------------+-----------------------------------
      Total |         104      100.00
```

Observamos que las 104 observaciones, con valores menores o iguales a uno provienen del distrito 401. Vamos a contar cuántas observaciones hay en el distrito 401, con el comando ***count***, y observamos que el distrito 401 cuenta con un total de 104 observaciones.

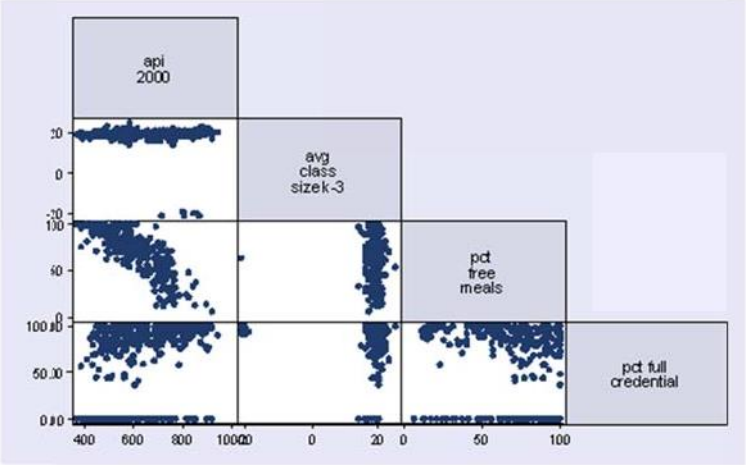
count if dnum==401

```
104
```

Todas las observaciones de este distrito se han registrado como proporciones en vez de porcentajes.

Otra técnica gráfica útil para filtrar los datos es un diagrama de dispersión matricial. Si bien esto es probablemente más relevante como herramienta de diagnóstico en busca de las no linealidades y los valores extremos en los datos, también puede ser una herramienta útil de detección de datos, en la revelación de información en la distribución conjunta de las variables que no se desprende del examen de las distribuciones univariantes. Echemos un vistazo a la matriz de dispersión para las variables de nuestro modelo de regresión. Esto revela los problemas que ya hemos identificado, es decir, los valores negativos y los valores de porcentaje introducidos como proporciones.

graph matrix api00 acs_k3 meals full, half



Hemos identificado tres problemas en nuestros datos. Hay muchos valores perdidos para la variable ***meals***, existen valores negativos en la variable ***acs_k3*** y más de la cuarta parte de los valores de la variable ***full*** son proporciones en lugar de porcentajes. La versión corregida de los datos se denomina ***elemapi2***. Vamos a utilizar ese archivo de datos y repetir el análisis y ver si los resultados son los mismos que el análisis inicial. En primer lugar, vamos a repetir el análisis de regresión inicial.

regress api00 acs_k3 meals full

```
      Source |       SS       df       MS              Number of obs =     313
-------------+------------------------------           F(  3,   309) =  213.41
       Model |  2634884.26     3  878294.754           Prob > F      =  0.0000
    Residual |  1271713.21   309  4115.57673           R-squared     =  0.6745
-------------+------------------------------           Adj R-squared =  0.6713
       Total |  3906597.47   312  12521.1457           Root MSE      =  64.153
```

```
---
 ------------------------------------------------------------------------------
       api00 |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      acs_k3 |  -2.681508   1.393991    -1.92   0.055    -5.424424    .0614073
       meals |  -3.702419   .1540256   -24.04   0.000    -4.005491   -3.399348
        full |   .1086104    .090719     1.20   0.232    -.0698947    .2871154
       _cons |   906.7392   28.26505    32.08   0.000     851.1228    962.3555
------------------------------------------------------------------------------
```

Ahora, vamos a utilizar el archivo de datos corregidos y repetir el análisis de regresión. Vemos una gran diferencia en los resultados. En el análisis original (arriba), ***acs_k3*** fue casi significativa, pero en el análisis corregido (abajo) los resultados muestran que esta variable no es significativa, quizás debido a las inconsistencias en los casos en que existen valores negativos en ***acs_k3***. Asimismo, el porcentaje de maestros con credenciales completas no fue significativa en el análisis original, pero es significativa en el análisis corregido, tal vez debido a los casos en que se le da el valor como proporción y no como porcentaje. Además, tenga en cuenta que el análisis se corrige basado en 398 observaciones en lugar de 313 observaciones, debido a la obtención de los datos completos de la variable ***meals*** que tenía varios valores que faltaban.

use *http://www.ats.ucla.edu/stat/stata/webbooks/reg/elemapi2*

regress api00 acs_k3 meals full

```
      Source |       SS       df       MS              Number of obs =     398
-------------+------------------------------           F(  3,   394) =  615.55
       Model |  6604966.18     3  2201655.39           Prob > F      =  0.0000
    Residual |  1409240.96   394   3576.7537           R-squared     =  0.8242
-------------+------------------------------           Adj R-squared =  0.8228
       Total |  8014207.14   397  20186.9197           Root MSE      =  59.806

------------------------------------------------------------------------------
       api00 |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      acs_k3 |  -.7170622   2.238821    -0.32   0.749    -5.118592    3.684468
       meals |  -3.686265   .1117799   -32.98   0.000    -3.906024   -3.466505
        full |   1.327138  .2388739      5.56   0.000      .857511    1.796765
       _cons |   771.6581   48.86071    15.79   0.000     675.5978    867.7184
------------------------------------------------------------------------------
```

A partir de este punto, usaremos el archivo de datos corregido, ***elemapi2***.

Hasta ahora hemos cubierto algunos de los temas de comprobación/verificación de los datos, pero en realidad no hemos discutido el análisis de regresión en sí. Ahora vamos a describir más sobre la realización de análisis de regresión en Stata.

1.4 Regresión lineal simple

Vamos a comenzar por mostrar algunos ejemplos de regresión lineal simple usando Stata. En

este tipo de regresión, sólo tenemos una variable de predicción. Esta variable puede ser continua, lo que significa que puede asumir todos los valores dentro de un rango, por ejemplo, la edad o altura, o puede ser dicotómica, lo que significa que la variable puede asumir sólo uno de dos valores, por ejemplo, 0 ó 1. Sólo hay una respuesta o variable dependiente, y es continua.

En Stata, la variable dependiente está en la lista inmediatamente después del comando seguida de una o más variables de predicción. Vamos a examinar la relación entre el tamaño de la escuela y el rendimiento académico para ver si el tamaño de la escuela se relaciona con el rendimiento académico. Para este ejemplo, ***api00*** es la variable dependiente **y *enroll*** es el predictor.

regress api00 enroll

```
      Source |       SS       df       MS              Number of obs =     400
-------------+------------------------------           F(  1,   398) =   44.83
       Model |  817326.293     1  817326.293           Prob > F      =  0.0000
    Residual |  7256345.70   398  18232.0244           R-squared     =  0.1012
-------------+------------------------------           Adj R-squared =  0.0990
       Total |  8073672.00   399  20234.7669           Root MSE      =  135.03

------------------------------------------------------------------------------
       api00 |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      enroll |  -.1998674   .0298512    -6.70   0.000    -.2585532   -.1411817
       _cons |   744.2514   15.93308    46.71   0.000     712.9279    775.5749
------------------------------------------------------------------------------
```

Vamos a revisar esta salida con un poco de detalle. En primer lugar, vemos que el F-test es estadísticamente significativo, lo que significa que el modelo es estadísticamente significativo. El R-cuadrado de 0.1012 significa que aproximadamente el 10% de la varianza de ***api00*** se explica por el modelo, en este caso, ***enroll***. El t-test para ***enroll*** es igual a -6,70, y es estadísticamente significativo, lo que significa que el coeficiente de regresión para ***enroll*** es significativamente distinto de cero. Tenga en cuenta que $(-6,70)^2 = 44,89$, es lo mismo que la estadística F (con algunos errores de redondeo). El coeficiente para ***enroll*** es de -.1998674, o aproximadamente -.2, lo que significa que para una unidad de incremento de la matrícula, se esperaría una disminución de 0.2 unidades en ***api00***. En otras palabras, una escuela con 1100 alumnos se espera que tenga una calificación API de 20 unidades menor que una escuela con 1000 alumnos. La constante 744.2514, es el valor esperado cuando ***enroll*** es igual a cero. En la mayoría de los casos, la constante no es muy interesante

Además de obtener la tabla de regresión, puede ser útil ver un diagrama de dispersión entre la variable explicada y la independiente junto a la línea de regresión. Después de ejecutar una regresión, se puede crear una variable que contiene los valores pronosticados usando el comando ***predict***. Usted puede obtener estos valores en cualquier momento después de ejecutar un comando de regresión, pero recuerde que una vez que se ejecuta una nueva regresión, los valores pronosticados se basan en la regresión más reciente. Para crear los

valores pronosticados sólo tiene que teclear ***predict*** y el nombre de una nueva variable donde se guardaran los valores ajustados. Para este ejemplo, el nombre de la nueva variable es ***fv***.

predict fv

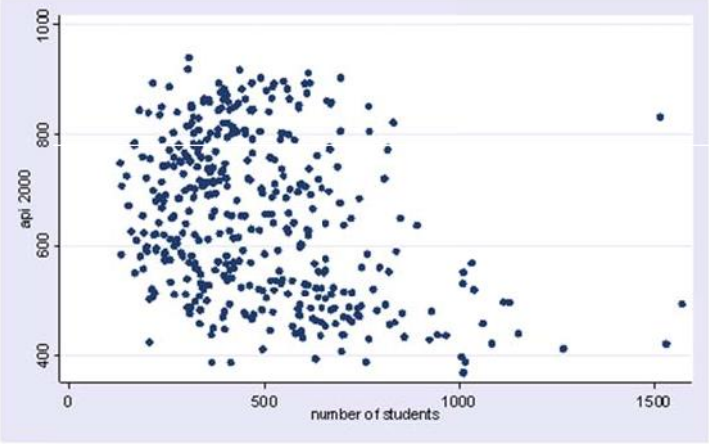
Si usamos el comando ***list***, vemos que un valor ajustado se ha generado para cada observación.

list api00 fv in 1/10

```
     api00        fv
 1.    369   542.5851
 2.    386   671.4996
 3.    386   661.7062
 4.    387   541.7857
 5.    387   592.1523
 6.    394   618.5348
 7.    397   543.5845
 8.    406   604.5441
 9.    411   645.5169
10.    412    491.619
```

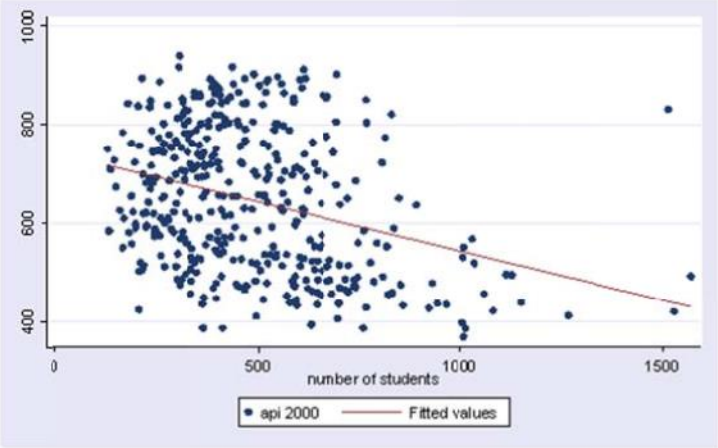
A continuación se puede mostrar un diagrama de dispersión entre las variables ***enroll*** y ***api00***.

scatter api00 enroll



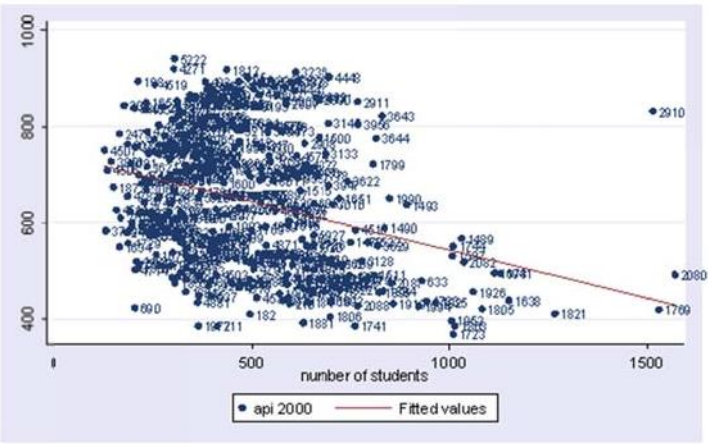
Podemos combinar dispersión con la recta ajustada para mostrar un diagrama de dispersión con los valores ajustados.

twoway (scatter api00 enroll) (lfit api00 enroll)



Como se puede observar, algunos de los puntos parecen ser valores atípicos. Si utiliza la opción ***mlabel(snum)*** en el comando de dispersión, puede ver el número de escolares por cada punto. Esto nos permite ver, por ejemplo, que uno de los valores extremos es la escuela 2910.

twoway (scatter api00 enroll, mlabel(snum)) (lfit api00 enroll)



Como vimos anteriormente, el comando ***predict*** puede ser utilizado para generar los valores pronosticados (ajustado) después de ejecutar una regresión. Usted también puede obtener los residuos mediante el comando ***predict*** seguido de un nombre de variable, en este caso ***e***, con la opción ***residual***.

predict e, residual

Este comando se puede acortar para ***predict e, resid*** o incluso ***predict e, r***. La siguiente tabla muestra algunos de los otros valores que se pueden crear con opciones de ***predict***.

Value to be created	Option after Predict
predicted values of y (y is the dependent variable)	no option needed
residuals	
standardized residuals	studentized or jackknifed residuals leverage

```
standard error of the residual                    resid
Cook's D                                          rstandard
standard error of predicted individual y          rstudent
standard error of predicted mean y                lev or hat
                                                  stdr
                                                  cooksd
                                                  stdf
                                                  stdp
```

1.5 Regresión múltiple

Ahora, echemos un vistazo a un ejemplo de regresión múltiple, en el que tenemos un

resultado (dependiente) y más de una variable predictora.

Para este ejemplo de regresión múltiple, se regresará la variable dependiente, ***api00***, en todas las variables de predicción en el conjunto de datos.

regress api00 ell meals yr_rnd mobility acs_k3 acs_46 full emer enroll

```
      Source |       SS       df       MS              Number of obs =     395
-------------+------------------------------           F(  9,   385) =  232.41
       Model |  6740702.01     9   748966.89           Prob > F      =  0.0000
    Residual |  1240707.78   385  3222.61761           R-squared     =  0.8446
-------------+------------------------------           Adj R-squared =  0.8409
       Total |  7981409.79   394  20257.3852           Root MSE      =  56.768

------------------------------------------------------------------------------
       api00 |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         ell |  -.8600707   .2106317    -4.08   0.000    -1.274203   -.4459382
       meals |  -2.948216   .1703452   -17.31   0.000     -3.28314   -2.613293
      yr_rnd |  -19.88875   9.258442    -2.15   0.032    -38.09218    -1.68531
    mobility |  -1.301352   .4362053    -2.98   0.003    -2.158995   -.4437089
      acs_k3 |     1.3187   2.252683     0.59   0.559      -3.1104    5.747801
      acs_46 |   2.032456   .7983213     2.55   0.011      .462841    3.602071
        full |    .609715   .4758205     1.28   0.201    -.3258169    1.545247
        emer |  -.7066192   .6054086    -1.17   0.244     -1.89694    .4837018
      enroll |   -.012164   .0167921    -0.72   0.469    -.0451798    .0208517
       _cons |   778.8305   61.68663    12.63   0.000     657.5457    900.1154
------------------------------------------------------------------------------
```

Vamos a examinar el resultado de este análisis de regresión. Al igual que con la regresión simple, miramos el p-valor de la F-test para ver si el modelo global es importante. Con un p-valor de cero a cuatro decimales, el modelo es estadísticamente significativo. El R-cuadrado es 0,8446, lo que significa que aproximadamente el 84% de la variabilidad de ***api00*** se explica por las variables en el modelo. En este caso, R-cuadrado ajustado indica que aproximadamente el 84% de la variabilidad de ***api00*** se explica por el modelo, incluso después de tomar en cuenta el número de variables predictoras en el modelo. Los coeficientes para cada una de las variables indica la cantidad de cambio que se podría esperar en ***api00*** dado un cambio de una unidad en el valor de esa variable, ya que todas las otras variables en el modelo se mantienen constantes. Por ejemplo, considere la variable ***ell***. Es de esperar una disminución de 0,86 en la puntuación ***api00*** por cada unidad de incremento en ***ell***, suponiendo que todas las otras variables del modelo se mantienen constantes. La interpretación de gran parte de la salida de la regresión múltiple es la misma que para la regresión simple.

Podemos preguntarnos lo que un cambio de 0,86 en ***ell*** significa, y cómo se puede comparar la fuerza de dicho coeficiente con el coeficiente de otra variable, por ejemplo las comidas. Para resolver este problema, podemos agregar una opción para el comando regresión llamada beta, lo que nos dará los coeficientes de regresión estandarizados. Los coeficientes beta son utilizados por algunos investigadores para comparar la fuerza relativa de los predictores diferentes dentro del modelo. Debido a que los coeficientes beta se miden en desviaciones estándar, en lugar de las unidades de las variables, pueden ser comparados entre sí. En otras

palabras, los coeficientes beta son los coeficientes que se obtendrían si la variable dependiente y las variables de predicción se transformaran todas a puntuaciones estándar antes de ejecutar la regresión; también de les conoce como z-score.

regress api00 ell meals yr_rnd mobility acs_k3 acs_46 full emer enroll, beta

```
      Source |       SS       df       MS              Number of obs =     395
-------------+------------------------------           F(  9,   385) =  232.41
       Model |  6740702.01     9   748966.89           Prob > F      =  0.0000
    Residual |  1240707.78   385  3222.61761           R-squared     =  0.8446
-------------+------------------------------           Adj R-squared =  0.8409
       Total |  7981409.79   394  20257.3852           Root MSE      =  56.768

------------------------------------------------------------------------------
       api00 |      Coef.   Std. Err.      t    P>|t|                     Beta
-------------+----------------------------------------------------------------
         ell |  -.8600707   .2106317    -4.08   0.000               -.1495771
       meals |  -2.948216   .1703452   -17.31   0.000               -.6607003
      yr_rnd |  -19.88875   9.258442    -2.15   0.032               -.0591404
    mobility |  -1.301352   .4362053    -2.98   0.003               -.0686382
      acs_k3 |     1.3187   2.252683     0.59   0.559                .0127287
      acs_46 |   2.032456   .7983213     2.55   0.011                .0549752
        full |    .609715   .4758205     1.28   0.201                .0637969
        emer |  -.7066192   .6054086    -1.17   0.244               -.0580132
      enroll |   -.012164   .0167921    -0.72   0.469               -.0193554
       _cons |   778.8305   61.68663    12.63   0.000                       .
------------------------------------------------------------------------------
```

Debido a que los coeficientes en la columna de Beta están todos en las mismas unidades estandarizadas, se pueden comparar estos coeficientes para evaluar la fuerza relativa de cada uno de los predictores. En este ejemplo, la variable comidas tiene el coeficiente beta más grande, -0,66 (en valor absoluto), y acs_k3 tiene la menor Beta, 0.013. Por lo tanto, un aumento de una desviación estándar en las comidas lleva a una disminución de la desviación estándar de 0,66 en la variable ***api00***, cuando las otras variables se mantienen constantes. A su vez, un aumento de una desviación estándar en ***acs_k3***, conduce a un aumento de 0.013 desviación estándar en la variable ***api00*** cuando las otras variables en el modelo se mantienen constantes.

En la interpretación de estos resultados, recordar que la diferencia entre los números que aparecen en la columna “Coef.” y la columna de “Beta” es la unidad de medida. Por ejemplo, para describir el coeficiente de primas para ***ell*** se dirá: "Una disminución de una unidad en ***ell*** daría lugar a un aumento de 0,86 unidades en la predicción de ***api00***". Sin embargo, para el coeficiente estandarizado (Beta) se dice: "Una disminución de una desviación estándar en ***ell*** daría lugar a un aumento de 0,15 desviaciones estándar en la predicción de ***api00***".

El comando ***listcoef*** da una salida más amplia en relación con los coeficientes estandarizados. No es parte de Stata, pero se puede descargar a través de Internet como este.

findit listcoef

Una vez descargado el comando lo ejecutamos de la siguiente manera:

listcoef

```
regress (N=395): Unstandardized and Standardized Estimates

 Observed SD: 142.32844
 SD of Error: 56.768104

------------------------------------------------------------------------------
   api00 |      b         t     P>|t|    bStdX    bStdY   bStdXY      SDofX
---------+--------------------------------------------------------------------
     ell |  -0.86007   -4.083   0.000 -21.2891  -0.0060  -0.1496    24.7527
   meals |  -2.94822  -17.307   0.000 -94.0364  -0.0207  -0.6607    31.8960
  yr_rnd | -19.88875   -2.148   0.032  -8.4174  -0.1397  -0.0591     0.4232
mobility |  -1.30135   -2.983   0.003  -9.7692  -0.0091  -0.0686     7.5069
  acs_k3 |   1.31870    0.585   0.559   1.8117   0.0093   0.0127     1.3738
  acs_46 |   2.03246    2.546   0.011   7.8245   0.0143   0.0550     3.8498
    full |   0.60972    1.281   0.201   9.0801   0.0043   0.0638    14.8924
    emer |  -0.70662   -1.167   0.244  -8.2569  -0.0050  -0.0580    11.6851
  enroll |  -0.01216   -0.724   0.469  -2.7548  -0.0001  -0.0194   226.4732
------------------------------------------------------------------------------
```

Comparemos los resultados del comando ***regress*** con los del comando ***listcoef***. Nos daremos cuenta que los valores que figuran en el Coef, t, y P> |. T | son los mismos en las dos salidas. Los valores que figuran en la columna de las "Beta" de la salida de regresión son los mismos que los valores en la columna de "bStadXY" del comando listcoef. La columna "bStdX" da el cambio en desviaciones estándar que se espera en Y ante un cambio de una unidad en X. La columna bStdY da el cambio que se espera en X ante una variación de una desviación estándar en Y. La columna "SDofX" da la desviación estándar de cada variable predictora en el modelo.

Por ejemplo, el bStdX para ***ell*** es -21,3, lo que significa que un aumento de una desviación estándar en ***ell*** llevaría a una disminución de 21,3 unidades en ***api00***. El valor bStdY para ***ell*** de -0,0060 significa que, para un aumento de un uno por ciento, en los estudiantes del idioma Inglés, esperaríamos una disminución de 0,006 desviaciones estándar en ***api00***. Dado que los valores de bStdX están en unidades estándar para las variables de predicción, puede utilizar estos coeficientes para comparar la fuerza relativa de los predictores como si fuera comparar los coeficientes Beta. La diferencia es que los coeficientes "BStdX" se interpretan como cambios en las unidades de la variable de resultado en lugar de en unidades estandarizadas de la variable de resultado. Por ejemplo, el BStdX para ***meals*** frente a ***ell*** es -94 contra -21, o cerca de 4 veces más grande, la misma proporción que la relación de los coeficientes Beta.

Hasta el momento, nos hemos preocupado con las pruebas de una sola variable a la vez, por ejemplo, observando el coeficiente de ***ell*** y determinar si es o no significativo. Sin embargo, también podemos probar conjuntos de variables, utilizando el comando ***test***, para ver si el conjunto de variables son significativas. En primer lugar, vamos a empezar por probar, una sola variable, la variable ***ell***, utilizando el comando ***test***.

test ell==0

```
 ( 1)  ell = 0.0

       F(  1,   385) =   16.67
             Prob > F =    0.0001
```

Si se compara este resultado con la salida de la última regresión se puede ver que el resultado de la prueba F, 16,67, es la misma que la mostrada en los resultados de la prueba t de la regresión (-4.083^2=16,67). Tenga en cuenta que podría obtener los mismos resultados si escribe el comando siguiente, ya que por defecto asume la prueba igual a 0.

test ell

```
 ( 1)  ell = 0.0

       F(  1,   385) =   16.67
            Prob > F =    0.0001
```

Quizás una prueba más interesante sería ver si la contribución de la variable "tamaño de clase" es importante. Dado que la información sobre el tamaño de la clase está contenida en dos variables, acs_k3 y acs_46, debemos incluir ambas variables en el comando.

test acs_k3 acs_46

```
 ( 1)  acs_k3 = 0.0
 ( 2)  acs_46 = 0.0

       F(  2,   385) =    3.95
            Prob > F =    0.0200
```

El valor de la prueba F, 3,95, significa que la contribución colectiva de estas dos variables es significativa. Otra manera de expresar esta significancia, es que hay una diferencia significativa entre un modelo con ***acs_k3*** y ***acs_46*** en comparación con un modelo sin ellas, es decir, hay una diferencia significativa entre un modelo completo y uno reducido.

Finalmente, como parte de realizar un análisis de regresión múltiple, se podría estar interesado en ver las correlaciones entre las variables en el modelo de regresión. Usted puede hacer esto con el comando ***correlate***, como se muestra a continuación.

correlate api00 ell meals yr_rnd mobility acs_k3 acs_46 full emer enroll

```
(obs=395)

             |    api00      ell    meals   yr_rnd mobility   acs_k3   acs_46
-------------+---------------------------------------------------------------
       api00 |   1.0000
         ell |  -0.7655   1.0000
       meals |  -0.9002   0.7711   1.0000
      yr_rnd |  -0.4831   0.5104   0.4247   1.0000
    mobility |  -0.2106  -0.0149   0.2207   0.0321   1.0000
      acs_k3 |   0.1712  -0.0553  -0.1888   0.0222   0.0397   1.0000
      acs_46 |   0.2340  -0.1743  -0.2137  -0.0419   0.1280   0.2708   1.0000
        full |   0.5759  -0.4867  -0.5285  -0.4045   0.0235   0.1611   0.1212
        emer |  -0.5902   0.4824   0.5402   0.4401   0.0612  -0.1111  -0.1283
      enroll |  -0.3221   0.4149   0.2426   0.5920   0.1007   0.1084   0.0281

             |     full     emer   enroll
-------------+---------------------------
        full |   1.0000
        emer |  -0.9059   1.0000
      enroll |  -0.3384   0.3417   1.0000
```

Si nos fijamos en las correlaciones con ***api00***, vemos que ***meals*** y ***ell*** tienen las dos mayores correlaciones con la variable ***api00***. Estas correlaciones son negativas, lo que significa que el valor de una variable disminuye cuando el valor de la otra variable tiende a subir. Sabiendo que estas variables están fuertemente asociadas con ***api00***, podemos predecir que serían las variables predictoras estadísticamente más significativas en el modelo de regresión.

También podemos usar el comando ***pwcorr*** para calcular correlaciones por parejas. La diferencia más importante entre ***correlate*** y ***pwcorr*** es la forma en que los datos faltantes se manejan. Con ***correlate***, una observación o el caso se elimina si alguna variable tiene un valor perdido, en otras palabras, se utiliza eliminación por lista para el cálculo de la correlación, también llamada por casos. Por otro lado, ***pwcorr*** utiliza la eliminación por parejas, lo que significa que la observación se elimina sólo si hay un valor que falta para el par de variables que se correlacionan. Dos opciones que puede utilizar con ***pwcorr***, pero no con ***correlate***, es la opción de ***SIG***, que dará a los niveles de significación de las correlaciones y la opción de ***obs***, que dará el número de observaciones utilizadas en la correlación. Esta opción no es necesaria con ***correlate*** pues, Stata muestra el número de observaciones en la parte superior de la salida.

pwcorr api00 ell meals yr_rnd mobility acs_k3 acs_46 full emer enroll, obs sig

	api00	ell	meals	yr_rnd	mobility	acs_k3	acs_46	full	emer
api00	1.0000								
	400								
ell	-0.7676	1.0000							
	0.0000								
	400	400							
meals	-0.9007	0.7724	1.0000						
	0.0000	0.0000							
	400	400	400						
yr_rnd	-0.4754	0.4979	0.4185	1.0000					
	0.0000	0.0000	0.0000						
	400	400	400	400					
mobility	-0.2064	-0.0205	0.2166	0.0348	1.0000				
	0.0000	0.6837	0.0000	0.4883					
	399	399	399	399	399				
acs_k3	0.1710	-0.0557	-0.1880	0.0227	0.0401	1.0000			
	0.0006	0.2680	0.0002	0.6517	0.4245				
	398	398	398	398	398	398			
acs_46	0.2329	-0.1733	-0.2131	-0.0421	0.1277	0.2708	1.0000		
	0.0000	0.0005	0.0000	0.4032	0.0110	0.0000			
	397	397	397	397	396	395	397		
full	0.5744	-0.4848	-0.5276	-0.3977	0.0252	0.1606	0.1177	1.0000	
	0.0000	0.0000	0.0000	0.0000	0.6156	0.0013	0.0190		
	400	400	400	400	399	398	397	400	
emer	-0.5827	0.4722	0.5330	0.4347	0.0596	-0.1103	-0.1245	-0.4558	1.0000
	0.0000	0.0000	0.0000	0.0000	0.2348	0.0277	0.0131	0.0000	
	400	400	400	400	399	398	397	400	400

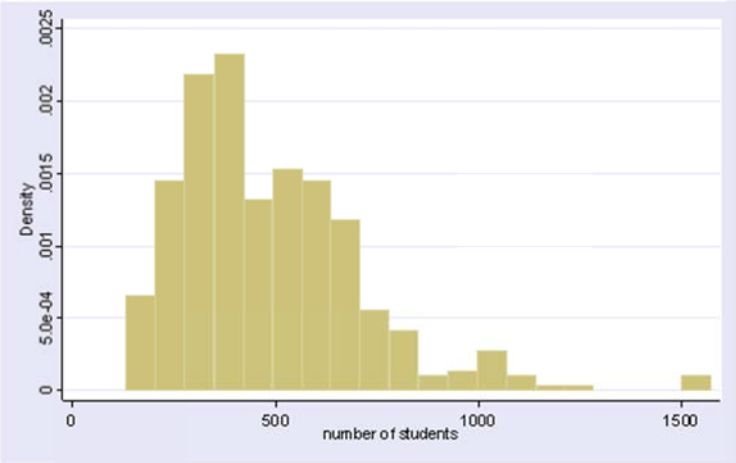
```
      enroll |  -0.3182   0.4030   0.2410   0.5918   0.1050   0.1089   0.0283
             |   0.0000   0.0000   0.0000   0.0000   0.0360   0.0298   0.5741
             |      400      400      400      400      399      398      397
             |

             |     full     emer   enroll
-------------+---------------------------
        full |   1.0000
             |
             |      400
             |
        emer |  -0.9057   1.0000
             |   0.0000
             |      400      400
             |
      enroll |  -0.3377   0.3431   1.0000
             |   0.0000   0.0000
             |      400      400      400
             |
```

1.6 Transformación de variables

Nos hemos preocupado por los posibles errores en los datos de las variables. En el siguiente capítulo, nos centraremos en los diagnósticos de regresión para verificar si sus datos cumplen los supuestos de la regresión lineal. Aquí, nos centraremos en el tema de la normalidad. Algunos investigadores creen que la regresión lineal requiere que el resultado (dependiente) y las variables de predicción se distribuyan normalmente. Debemos aclarara este tema. Lo que se necesita es que los residuos se distribuyan normalmente. De hecho, los residuos deben ser normales sólo para que el t-test sea válido. La estimación de los coeficientes de regresión no requieren residuales distribuidos normalmente. Como estamos interesados en tener validez en las pruebas t, vamos a investigar cuestiones relativas a la normalidad.

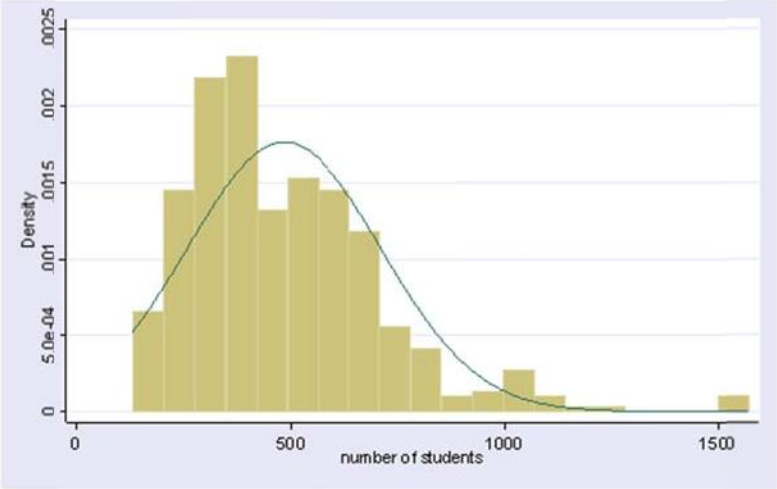
Una causa común de la no normalidad de los residuos es que la distribución de la variable dependiente y/o variables de predicción no se distribuyen normalmente. Por lo tanto, vamos a explorar la distribución de nuestras variables y cómo podemos transformarlas a una forma más normal. Vamos a empezar por hacer un histograma de la variable ***enroll***, que vimos anteriormente en la regresión simple.

histogram enroll



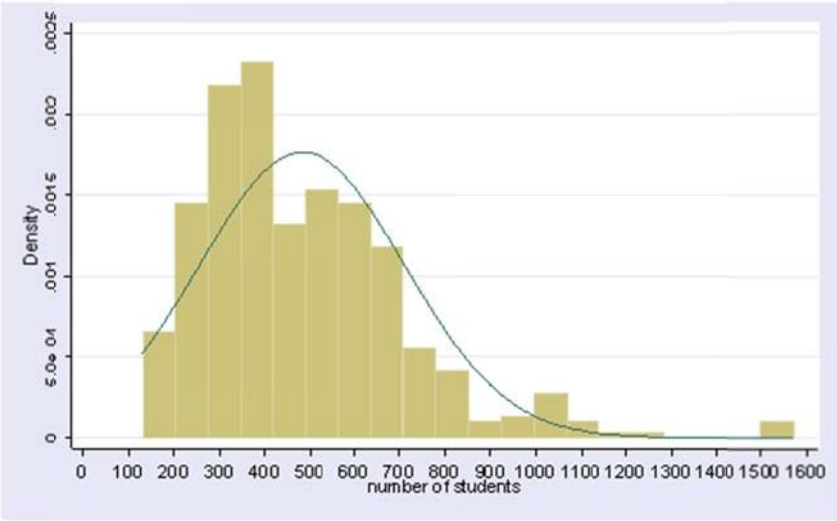
Podemos utilizar la opción normal para superponer una curva normal en este gráfico y la opción bin(20) opción de utilizar 20 intervalos. Observamos que la distribución se ve sesgada a la derecha.

histogram enroll, normal bin(20)



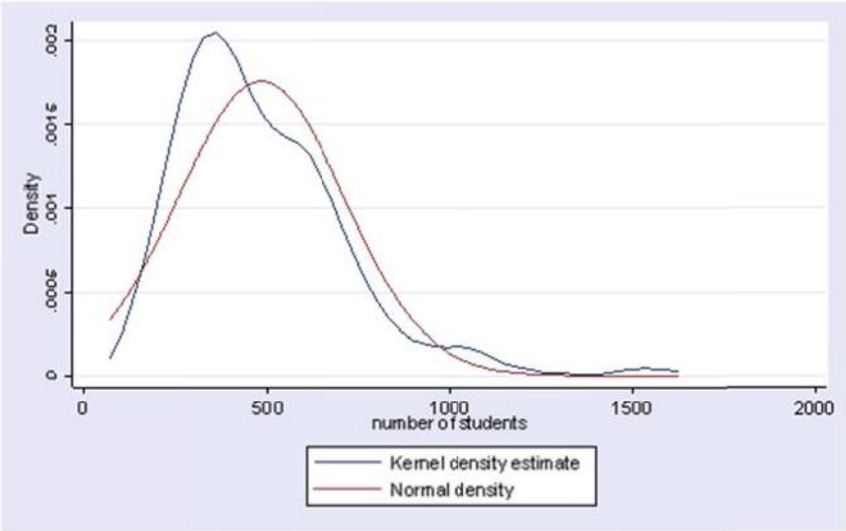
También es posible que desee modificar las etiquetas de los ejes. Por ejemplo, podemos utilizar la opción ***xlabel()*** para el etiquetado del eje X, a continuación definimos el etiquetado del eje x como de 0 a 1600 con incremento de 100.

histogram enroll, normal bin(20) xlabel(0(100)1600)



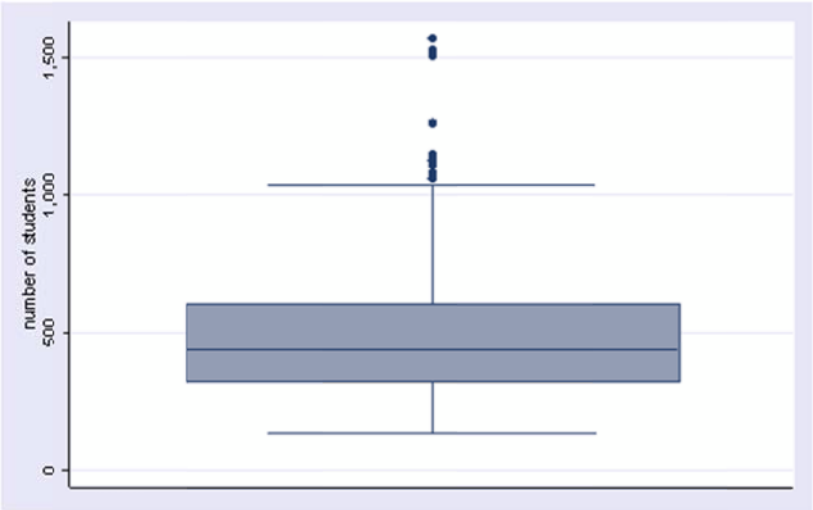
Los histogramas son sensibles al número de intervalos o columnas que se utilizan para graficar la distribución. Una alternativa a los histogramas es el diagrama de densidad del núcleo, que se aproxima a la densidad de probabilidad de la variable. Los intervalos del gráfico de densidad de Kernel tiene la ventaja de ser suave y de ser independiente de la elección del origen, a diferencia de los histogramas. Stata implementa intervalos de densidad kernel con el comando kdensity.

kdensity enroll, normal



La trama creada por el comando ***kdensity*** también nos indica que la variable ***enroll*** no se muestra normal. Ahora vamos a hacer un diagrama de caja para ***enroll***, utilizando el comando ***graph box***.

graph box enroll

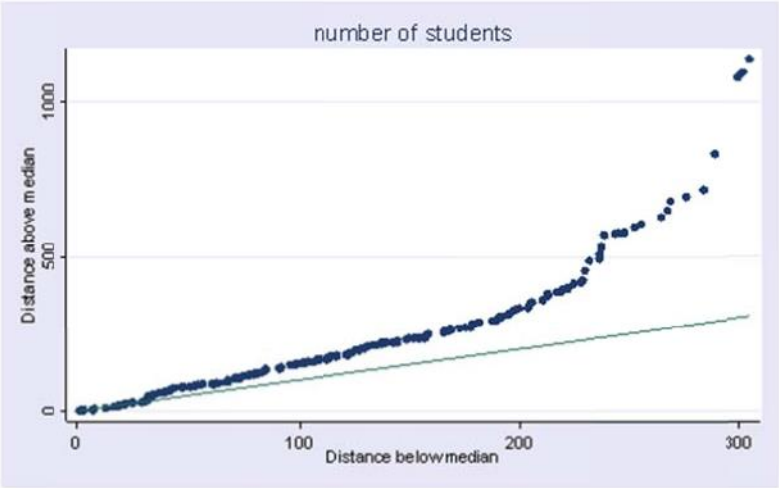


Tenga en cuenta los puntos en la parte superior del diagrama de caja que indican los posibles valores extremos, es decir, estos puntos de datos son mayores a 1,5*(rango intercuartil), esto es, por encima del percentil 75. Este diagrama de caja también confirma que la distribución de ***enroll*** es sesgada a la derecha.

Hay tres tipos de gráficos que se utilizan a menudo para examinar la distribución de las variables; gráficos de simetría, gráficos normal cuantil y gráficos de probabilidad normal.

Un gráfico de simetría grafica la distancia por encima de la mediana para el valor i-ésimo en contra de la distancia por debajo de la mediana para el valor i-ésimo. Una variable que es simétrica tendría puntos que están en la línea diagonal. Como era de esperar, esta distribución no es simétrica.

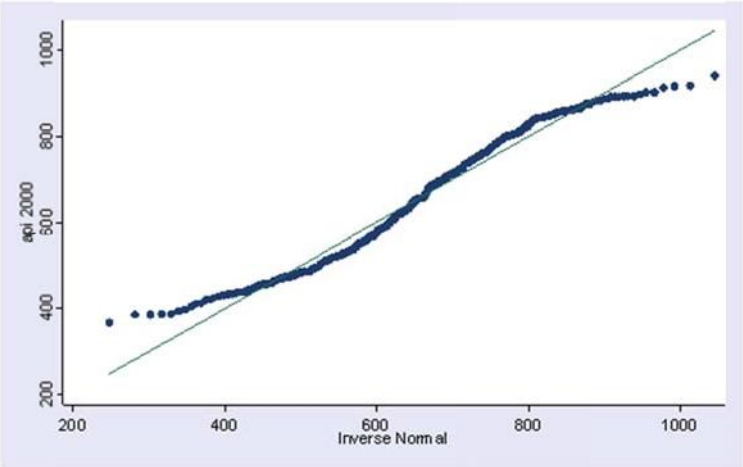
symplot enroll



Una gráfico normal cuantil graficas los cuantiles de una variable en contra de los cuantiles de una normal (gaussiana) de distribución, ***qnorm*** es sensible a la no normalidad. De hecho vemos

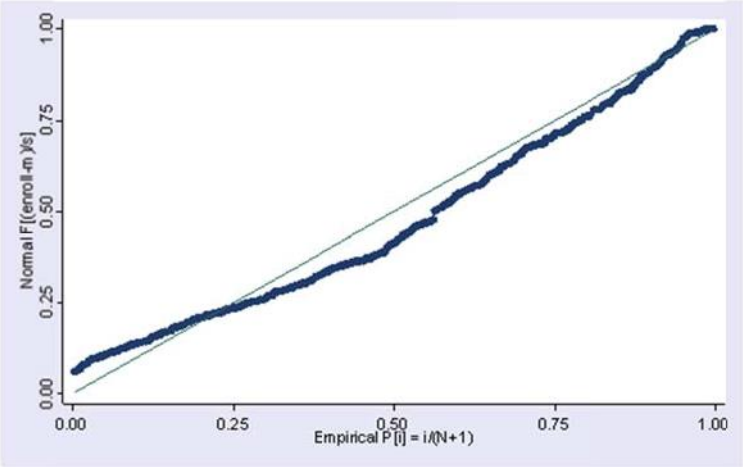
desviaciones importantes a la normal, en las colas de la línea diagonal. Este gráfico es típico de las variables que están fuertemente sesgadas a la derecha.

qnorm api00



Finalmente, el gráfico de probabilidad normal también es útil para examinar la distribución de las variables, ***pnorm*** es sensible a las desviaciones de la normalidad cerca del centro de la distribución. Una vez más, vemos indicios sobre la no-normalidad de la variable ***enroll***.

pnorm enroll



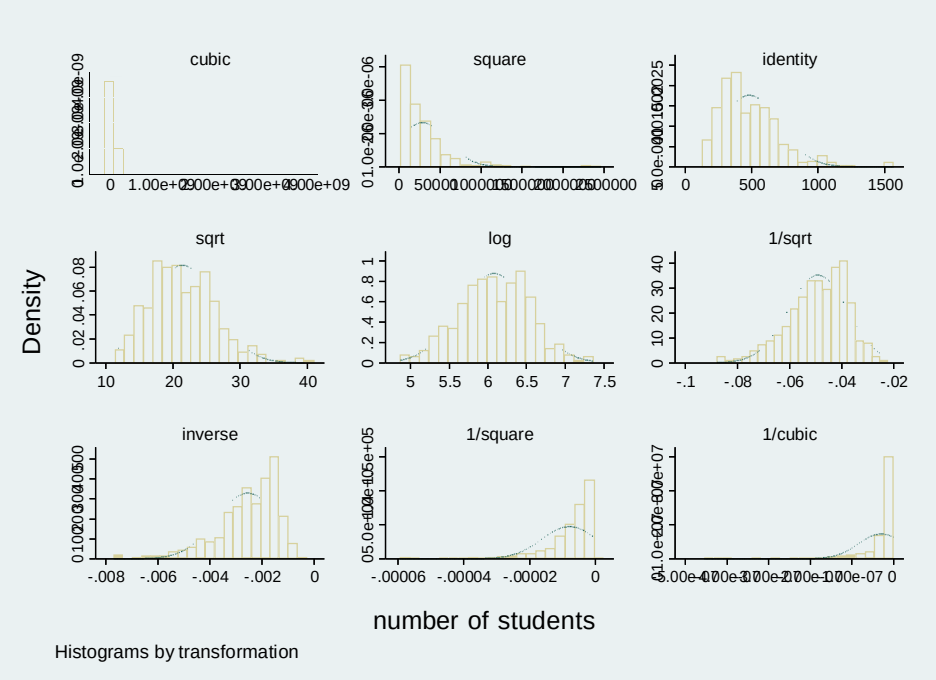
Habiendo concluido que la variable ***enroll*** no se distribuye normalmente, ¿cómo debemos abordar este problema? En primer lugar, podemos intentar ingresar la variable tal y como está en la regresión, pero vemos problemas, que es probable que existan, entonces podemos tratar de transformar la variable ***enroll*** para que se acerque a una distribución normal. Algunas transformaciones útiles son el logaritmo, la raíz cuadrada o elevar la variable a una potencia. Seleccionar la transformación adecuada es algo así como un arte. Stata incluye los comandos ***ladder*** y ***gladder*** para ayudar en la tranformación. ***Ladder*** reporta resultados numéricos y ***gladder*** produce una vista gráfica. Vamos a empezar con ***ladder*** y buscar la transformación con el mínimo valor de chi-cuadrado.

ladder enroll

```
ladder enroll

Transformation         formula               chi2(2)       P(chi2)
-----------------------------------------------------------------
cube                   enroll^3                   .        0.000
square                 enroll^2                   .        0.000
raw                    enroll                     .        0.000
square-root            sqrt(enroll)           20.56        0.000
log                    log(enroll)             0.71        0.701
reciprocal root        1/sqrt(enroll)         23.33        0.000
reciprocal             1/enroll               73.47        0.000
reciprocal square      1/(enroll^2)               .        0.000
reciprocal cube        1/(enroll^3)               .        0.000
```

La transformación *logaritmo* tiene el valor chi-cuadrado más pequeño. Vamos a verificar estos resultados en forma gráfica con ***gladder***.

gladder enroll

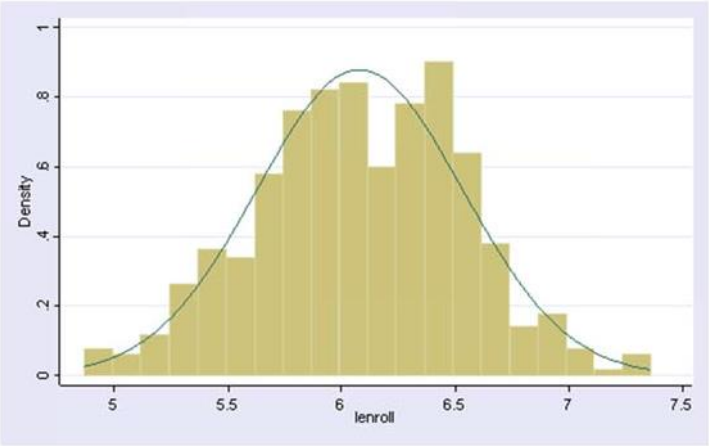


Esto también indica que la transformación logarítmica ayudaría a que ***enroll*** se aproxime más a distribución normal. Vamos a usar el comando generar con la función log para crear la variable ***lenroll*** que será el logaritmo de matrículas. Tenga en cuenta que la función ***log*** de Stata le dará el logaritmo natural, no logaritmo base 10. Para obtener logaritmo base 10, deberá utilizar ***log10 (var)***.

generate lenroll = log(enroll)

Ahora generamos el histograma para la variable transformada y observamos si lo hemos normalizado.

hist lenroll, normal



Podemos ver que ***lenroll*** se aproxima bastante a una distribución normal. A continuación, se utilizan los comandos ***symplot***, ***qnorm*** y ***pnorm*** para ayudarnos a evaluar si ***lenroll*** parece normal, así como ver cómo ***lenroll*** impacta en los residuos, que es realmente el factor importante.

1.7 Resumen

En este capítulo se han discutido los fundamentos de cómo realizar regresiones simples y múltiples, algunas interpretaciones de los resultados, así como algunos comandos relacionados. Hemos examinado algunas de las herramientas y técnicas para la detección de datos erróneos y las consecuencias que estos datos pueden tener en sus resultados. Por último, nos referimos a los supuestos de la regresión lineal y se ilustra cómo se puede comprobar la normalidad de las variables y cómo se pueden transformar las variables para lograr la normalidad. El siguiente capítulo se aborda lo referente a la discusión de los supuestos de regresión lineal y cómo se puede utilizar Stata para evaluar estos supuestos para los datos. En particular, el siguiente capítulo abordará los siguientes temas.

- Comprobación de los puntos que ejercen una influencia indebida sobre los coeficientes
- Comprobación de la varianza constante del error (homocedasticidad)
- Comprobación de las relaciones lineales
- Verificación de las especificaciones del modelo
- Comprobación de la multicolinealidad
- Comprobar la normalidad de los residuos

1.8 Autoevaluación

- Hacer cinco gráficos de ***api99***: histograma, diagrama kdensity, diagrama de caja, gráfico de simetría y el gráfico cuantil normal.
- ¿Cuál es la correlación entre api99 y ***meals***?
- Regresar ***api99*** en ***meals***. ¿Qué interpretación tienen los resultados?
- Crear y listar los valores pronosticados.
- Graficar ***meals*** y ***api99*** con y sin la línea de regresión.

- Observa las correlaciones entre las variables api99, ***ell***, ***meals***, ***avg_ed***, con los comandos ***corr*** y ***pwcorr***. Explicar en qué son diferentes estos comandos. Hacer un diagrama de dispersión matricial de estas variables y relacionar los resultados de la correlación de la matriz de dispersión.
- Realizar una regresión para predecir ***api99*** en función de ***meals*** y ***ell***. Interpretar el resultado.