



INTRODUCTION TO STATA

Yao Thibaut Kpegli

Master 1 in Economics, ENS Paris-Saclay
2023–2024

INTRODUCTION

PURPOSE

- A quick and intensive reminder about STATA (4.5 hours)
 - to make you operational during your econometrics classes over the year

STATA: DEFINITION

STATA is a fee-based software to conduct statistical and econometrics analysis, optimization, simulation, numerical computation, etc...

INSTALLATION

INSTALLATION

- download the setup of STATA from the intranet of ENS Paris-Saclay
- Install the setup (use the informations of the license provided by ENS)

STATA AS A SIMPLE CALCULATOR

DISPLAY AND SCALAR

- `sca a = 2`
- `sca b = 3`
- `sca c = - 4`
- `di a`
- `di b`

COMMON OPERATIONS

- addition : `di 2+3` or `di a+b`
- subtraction: `di 2-3` or `di a-b`
- product : `di 2*3` or `di a*b`
- division: `di 2/2` or `di a/b`
 - remark: STATA accepts shortened command name (true also for variable)

COMMON MATH FUNCTIONS

COMMON MATH FUNCTIONS

- square/exponent : di 2^3 or di a^b
- square root: di $\text{sqrt}(2)$ or di $\text{sqrt}(a)$
- exponential : $\text{exp}(2)$ or di $\text{exp}(a)$
- logarithm : di $\log(2)$ or $\log(a)$
- min: di $\text{min}(2,3)$ or di $\text{min}(a,b)$
- max: di $\text{max}(2,3)$ or di $\text{max}(a,b)$
- absolute val.: di $\text{abs}(-4)$ or di $\text{abs}(c)$
- sign: di $\text{sign}(-4)$ or di $\text{sign}(c)$
- round: di $\text{round}(4.56789, 0.01)$
- ceiling: di $\text{ceil}(4.56789)$
- integer: di $\text{int}(4.56789)$

MATRIX OPERATIONS

MATRIX OPERATIONS

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$$

- define matrix: matrix A = (1,2 \ 3,4)
- show the matrix: matrix list A
- extract element at line i=2 and column j=1 : di A[2,1]

IDENTY MATRIX

- mat I = I(2)
- mat li I

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

REPETITION MATRIX J(NROW,NCOL,VALUE)

- mat V1 = J(2,1,1)
- mat li V1

MATRIX OPERATIONS

MATRIX OPERATIONS

- transposition:
 - mat define $B = A'$
 - mat li B
- matrix product
 - mat define $C = A * C$
 - mat li C
- matrix addition
 - mat $D = A + C$
 - mat li D
- Substraction
 - mat $E = A - C$
 - mat li E
- division by a scalar
 - mat $F = A / 2$
 - mat li F

KRONECKER PRODUCTS

KRONECKER PRODUCT

$$A = \begin{pmatrix} 0.5 \\ 2 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$$

- Kronecker product
 - `mat A = (0.5 \ 2)`
 - `mat B = (1,0 \ 1,1)`
 - `mat C = A # B`
 - `mat li C`

$$\begin{pmatrix} 0.5 & 0 \\ 0.5 & 0.5 \\ 2 & 0 \\ 2 & 2 \end{pmatrix}$$

MATRIX OPERATIONS

$$A = \begin{pmatrix} 1 & 3 \\ 0 & 4 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

ROW JOIN

- `mat A = (1,2 \ 3,4)`
- `mat B = I(2)`
- `mat C = (A,B)`
- `mat li C`

COLUMN JOIN

- `mat D = (A \ B)`
- `mat li D`

MATRIX OPERATIONS

USEFUL FUNCTIONS

- inverse: `inv()`
 - `mat inv = inv(A)`
 - `mat li inv`
- diagonal vector: `vecdiag()`
 - `mat diag = vecdiag(A)`
 - `mat li diag`
- trace : `trace()`
 - `di trace(A)`
- determinant : `det()`
 - `di det(A)`

EXERCISE

- Create a matrix X that takes the values

$$\begin{pmatrix} 1 & 1 \\ 1 & 4 \\ 1 & 2 \\ 1 & 5 \end{pmatrix}$$

- Compute $B = (X'X)^{-1}$
 - determine the dimension
 - extract the elements on the diagonal
- Create a matrix Y that takes value

$$\begin{pmatrix} 3 \\ 6 \\ 4 \\ 7 \end{pmatrix}$$

- Compute $(X'X)^{-1}X'Y$

RESULT

- `mat X = (1 4 2 5)`
- `mat X = J(4,1,1),X`
- `mat li X`
- `mat B = inv(X'*X)`
- `mat li B`
- `mat diag = vecdiag(B)`
- `mat li diag`
- `mat Y = (3 6 4 7)`
- `mat li Y`
- `mat beta = B*X'*Y`
- `mat li beta`

WORKING DIRECTORY

WORKING DIRECTORY

- Useful to import/export data and results
 - set the work directory: `cd "D:\ENS Paris Saclay\R2023_2024"`

DEFINITION

DEFINITION

- data = matrix of data
 - set of vectors with the same length
 - placed next to each other vertically
- Column = Variable
 - possible of different types: quantitative (in black), characters (in red), dates (in black), numerical but also qualitative (in black).
 - Color “blue” means qualitative variables that are codified and labeled
- Line = Observation

EXAMPLE

- upload the data “auto” integrated in STATA
 - clear all
 - sysuse auto
 - browse
- remark: index of row “_n” and total rows “_N”

CREATION

index of row	one	two	three
1	1	11	21
2	2	12	22
.	.	.	.
.	.	.	.
.	.	.	.
10	10	20	30

CREATION WITH THE HANDS

- clear all
- edit
- start entering directly your data

CREATION

CREATION WITH CODES

- clear all
- set obs 10
- gen one = `_n`
- gen two = `_n+10`
- gen three = `_n +20`
- browse

QUICK INFOS ON THE DATA

- describe content of data: describe
- more detailed description: codebook
- stat des: summarize
 - summarize one
 - stat des (option detail): summarize one, detail

BASIC DATA MANIPULATION

USEFUL SYMBOLS

- Strict inequality : $>$, $<$
- equal or inequal : $=<$, $>=$
- equal: $==$
- different equal: $!=$
- and : $\&$
- or : $|$

RENAME VAR. AND REPLACE OBS.

- rename var.: `rename ratio new`
- replace obs: `replace one = 3 if one == 5`

BASIC DATA MANIPULATION

NEW VARIABLE

- creation of ratio=one/two: `gen ratio = one/two`
- dummy (binary) var. (string):
 - `gen dummy_ch = ""`
 - replace `dummy_ch="Low"` if `one <= 5`
 - replace `dummy_ch="High"` if `one > 5`
- dummy (binary) var. (numerical):
 - `gen dummy_num = (one <= 5)`
- labialization of values
 - `lab define dum 1 "Low" 0 "High"`
 - `label value dummy_num dum`

SUBSET OF DATAFRAME

KEEP AND DROP: APPLY ON VARIABLES AND OBSERVATIONS

- keep lines with $\text{two} \leq 16$: keep if $\text{two} \leq 16$
- drop lines with $\text{two} \geq 14$: drop if $\text{two} \geq 14$
- delete column ratio: drop ratio
- keep columns one and three: keep one three
- you can think of a combination of conditions

EXPORTATION

EXPORTATION IN TEXT FILE

save example , replace

EXPORTATION IN EXCEL FILE

export excel using example.xlsx , firstrow(variables) replace

EXPORTATION IN CSV FILE

export delimited using example.csv, replace

EXPORTATION IN TEXT FILE

export delimited using example.txt, replace

IMPORTATION

EXPORTATION IN TEXT FILE

```
import example , clear
```

EXPORTATION IN EXCEL FILE

```
import excel example.xlsx , firstrow clear
```

EXPORTATION IN CSV FILE

```
import delimited example.csv, clear
```

EXPORTATION IN TEXT FILE

```
import delimited example.txt, clear
```

“VERTICAL MERGE” (APPEND)

TWO DATA

- First data
 - sysuse auto, clear
 - keep if $_n \leq 37$
 - save first37 , replace
- Second data
 - sysuse auto, clear
 - keep if $_n > 37$
 - save after37 , replace

“VERTICAL MERGE” (APPEND)

“VERTICAL MERGE” (APPEND)

- you are merging two data **by using the names of variables:**
 - use first37 , clear
 - append using after37

“HORIZONTAL MERGE” (MERGE)

TWO DATA

- First data
 - sysuse auto, clear
 - keep if `_n <= 37`
 - keep make price
 - rename (make price)(make_1 price_1)
 - gen identifier = `_n`
 - save first37_merge , replace
- Second data
 - sysuse auto, clear
 - keep if `_n > 37`
 - keep make price
 - gen identifier = `_n`
 - save after37 , replace

“HORIZONTAL MERGE” (MERGE)

“HORIZONTAL MERGE” (MERGE)

- you are merging two data by using “identifier(s)” of row:
 - use first37 , clear
 - merge 1:1 identifier using after37

EXERCISE

- Use data "auto" integrated in STATA
- describe the data
- detailed sta des for variables : price, weight, rep78, trunk, length, and foreign
- frequency table for rep78 and look for missing values
- cross tabulation of rep78 and foreign
- generate the dummy/qualitative variables:
 - domestic (counterpart of foreign)
 - small_trunk taking value 1 if trunk < 10
 - standard_trunk taking value 1 if trunk ≥ 10 and trunk < 20
 - huge_trunk taking value 1 if trunk ≥ 20
 - trunk_categ taking values 1, 2, and 3 respectively for small, standard, and huge trunks

EXERCISE

- create a label and applies it on the variable `trunk_categ`
- create `lprice = log(price)`
- Keep variables foreign price trunk (all) weight length
- export your data in STATA and excel formats

EXERCISE: RESULT

- sysuse auto, clear
- d
- codebook
- sum rep78 price weight foreign trunk length, det
- tab rep78
- tab rep78, mis
- tab foreign
- tab foreign, nolabel
- gen domestic = 1- foreign
- sum trunk, det
- gen trunk_small = (trunk<10)

EXERCISE: RESULT

- `gen trunk_standard = (trunk>=10)*(trunk<20)`
- `gen trunk_huge=(trunk>=20)`
- `gen trunk_categ= trunk_small + 2*trunk_standard + 3*trunk_huge`
- `lab def catego 1 "Small" 2 "Standard" 3 "Huge"`
- `lab value trunk_categ catego`
- `keep foreign price trunk* weight length`
- `gen lprice = log(price)`
- `save autoME, replace`
- `export excel using autoME.xlsxs , replace firstrow(variables)`

EGEN

EGEN

- allows to create a variable based on *STATA's specific functions* that involve several variables or rows
 - sysuse auto, clear
 - egen rowmean= rowmean(price - turn)
 - edit
 - sum rowmean
 - egen mean_price = mean(price) , by(rep78)
 - edit
 - ta mean_price rep78, m
- type “help egen” to see all available functions for egen

COLLAPSE

COLLAPSE

- allow to make dataset of summary statistics
 - sysuse auto , clear
 - collapse (mean) price weight (min) trunk , by(rep78)
 - edit

PRESERVE/RESTORE

PRESERVE/RESTORE

- preserve/restore: allow to (i) have two copies of a data (= preserve stage), (ii) manipulate a copy, (iii) restore the un-manipulate copy (== restore stage)
 - sysuse auto , clear
 - preserve
 - collapse (mean) price weight (min) trunk , by(rep78)
 - save data_manipulate , replace
 - restore
 - edit

SCATTER PLOT

- `sysuse auto, clear`
- scatter plot: `scatter price weight`
- remove the color at the background: `scatter price weight , graphregion(color(white))`
- add linear fit:
`graph twoway (scatter price weight, msize(small))(lfit price weight), graphregion(color(white)) title("Scatterplot and OLS fitted line")`

correlation: corr price trunk

USEFUL PLOTS

HISTOGRAM

- `gen lprice = log(price)`
- `hist lprice , graphregion(color(white)) name(gh)`

EMPIRICAL CUMULATIVE DENSITY FUNCTION

- `cumul lprice, gen(cum_lprice)`
- `sort cum_lprice`
- `line cum_lprice lprice, graphregion(color(white)) title("Cumulative of median family income") name(gc)`

EMPIRICAL DENSITY

`kdensity lprice , graphregion(color(white)) name(gk)`

BOX PLOT

`graph box lprice , graphregion(color(white)) name(gb)`

USEFUL PLOTS

PUT ALL PLOTS TOGHETER

```
graph combine gh gc gk gb , graphregion(color(white))
```

PLOT FUNCTIONS

PLOT FUNCTIONS

- $f(x) = x^2$: tw function `y = x^2` , `range(-2 2)`
`graphregion(color(white))` `ytile("f(x)")`
- plot density of `lprice` + normal distribution:
 - `sum lprice`
 - `sca m=r(mean)`
 - `sca sd= r(sd)`
 - tw (`kdensity lprice`) (function `y =`
`(1/(scalar(sd)*sqrt(2*_pi)))*exp(-0.5*((x-`
`scalar(m))/scalar(sd))^2)`, `range(7 10))`,
`graphregion(color(white))` `ytile(density)`
`legend(position(11) col(1) ring(0) label(1 "Kernel empirical`
`pdf") label(2 "Normal pdf"))`

OPTIMIZATION OVER DATA (E.G. ML ROUTINE)

$$\min_{a,b} \sum_{i=1}^n (y_i - ax_i - b)^2$$

with y and x the lprice and weight in data auto

RESULT

- sysuse auto , clear
- gen lprice=log(price)
- program drop _all
- the program:


```

program define ols_ml
  args lnf a b
  tempvar y x
  qui {
    generate double 'y' = $ML_y1 generate double 'x' =
    $ML_y2
    replace 'lnf' = -('y' - 'a'*'x'-'b')^2
  }
end

```
- ml model lf ols_ml (a: lprice weight =) (b:) , maximize
- ml di
- checking with reg: reg lprice weight

OPTIMIZATION OVER DATA (E.G. ML ROUTINE)

$$\max_p \sum_{i=1}^n [y_i \log(p) + (1 - y_i) \log(1 - p)]$$

with y the dummy variable taking value 1 if $\text{rep78} \leq 2$ and 0 otherwise.

RESULT

- `gen rep78small =(rep78<=2)`
- program :
program define bin_ml
args lnf p
tempvar y x
qui {
generate double 'y' = \$ML_y1
replace 'lnf' = 'y'*log('p')+(1-'y')*log(1-'p')
}
end
- `ml model lf bin_ml (p: rep78small =) , maximize`
- `ml di`
- checking with sum: `sum rep78small`
- **rmk: ML routine maximizes functions**
 - so need to adapt if the objective is to minimize functions

FORVALUE

FORVALUE

```
forval i=1/74 {  
  qui sum weight if _n <= 'i'  
  di "mean of lprice of the 'i' first observations is: " r(mean)  
}  
forval i=1/100 {  
  if mod('i',3) == 0 {  
    di "'i' is a multiple of 3"  
  }  
}
```

FOREACH

FOREACH

```
global VAR weight length
foreach x in $VAR {
  scatter lprice 'x' , graphregion(color(white)) name('x')
}
graph combine $VAR , graphregion(color(white))
```

WHILE

WHILE

```
sum lprice
sca min = r(min)
sca j = 1
local i = 1
while (lprice['i']!= r(min)) { local i = 'i' + 1
sca j = 'i'
}
di "the min of lprice is at the row: " j
```

RANDOM VALUES AND RE-ALLOCATION

USEFUL RANDOM VALUES

- normal: `rnormal(mu, sigma)`
 - $N(0, 1)$: `gen sdnorm= rnormal(0,1)`
- uniform: `runiform(a,b)`
 - $U[0, 1]$: `gen sdunif = runif(0,1)`

RANDOM DRAW WITH REPLACEMENT

- `bsample`
 - very useful to conduct “manually” a bootstrapping analysis in order to compute standard deviation/confidence interval

EXERCISE: AN ILLUSTRATION OF CLT

- First replication
- simulates a variable X of $n = 1000$ random values of $U(0, 1)$
- generate

$$Y = \begin{cases} 1 & \text{if } X \leq 0.4 \\ 0 & \text{if } X > 0.4 \end{cases}$$

- remark that Y follows a $\mathcal{B}(p)$ with $p = 0.4$
- Compute the sequence $\{\bar{Y}_1, \bar{Y}_2, \dots, \bar{Y}_{1000}\}$ with

$$\bar{Y}_i = \frac{1}{i} \sum_{j=1}^i Y_j$$

- plot all (\bar{Y}_i, i)

RESULT

- clear all
- set obs 1000
- set seed 123456789
- gen X = runiform()
- gen Y= (X<=0.4)
- gen Ybar=.
- qui forvalue i=1/1000 {
 sum Y if _n<='i'
 replace Ybar = r(mean) in 'i'
}
- gen i = _n
- line Ybar i, graphregion(color(white)) yline(0.4)

EXERCISE

- provides an approximated value for $F(x)$

$$F(x) = \int_{-10}^{10} f(x)dx$$

with $f(\cdot)$ the density function of the standard normal distribution, and $\pi \simeq 3.14$

- hint: $F(x) = (10 - (-10)) \times E[f(x)]$, with $E[.]$ the expectation computed with $U[-10, 10]$
 - this value is almost 1
- 1 simulates a vector x of $n = 100000000$ random values of $U(-10, 10)$
 - 2 compute $f(\cdot)$ over the simulated values of x
 - 3 Compute the mean of the \bar{f} over the n simulated values
 - rmk (central limit theorem): \bar{f} converges in proba towards $E[f(x)]$
 - 4 compute the approximated values as $20 \times \bar{f}$

RESULT

- clear all
- set obs 100000000
- gen x=runiform(-10,10)
- gen f= (1/sqrt(2*_pi))*exp(-0.5*x^2)
- sum f
- di "the approximated value is:" 20*r(mean)

EXERCISE

Provide an approximation for the quantity π

- hint: leverage on (i) the area of a circle $x^2 + y^2 \leq 1$, (ii) the area of a square centered at $(0,0)$ and whose side is 2, and (iii) random draws of x and y from $U[-1, 1]$.

1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024, 2025, 2026, 2027, 2028, 2029, 2030, 2031, 2032, 2033, 2034, 2035, 2036, 2037, 2038, 2039, 2040, 2041, 2042, 2043, 2044, 2045, 2046, 2047, 2048, 2049, 2050, 2051, 2052, 2053, 2054, 2055, 2056, 2057, 2058, 2059, 2060, 2061, 2062, 2063, 2064, 2065, 2066, 2067, 2068, 2069, 2070, 2071, 2072, 2073, 2074, 2075, 2076, 2077, 2078, 2079, 2080, 2081, 2082, 2083, 2084, 2085, 2086, 2087, 2088, 2089, 2090, 2091, 2092, 2093, 2094, 2095, 2096, 2097, 2098, 2099, 2100, 2101, 2102, 2103, 2104, 2105, 2106, 2107, 2108, 2109, 2110, 2111, 2112, 2113, 2114, 2115, 2116, 2117, 2118, 2119, 2120, 2121, 2122, 2123, 2124, 2125, 2126, 2127, 2128, 2129, 2130, 2131, 2132, 2133, 2134, 2135, 2136, 2137, 2138, 2139, 2140, 2141, 2142, 2143, 2144, 2145, 2146, 2147, 2148, 2149, 2150, 2151, 2152, 2153, 2154, 2155, 2156, 2157, 2158, 2159, 2160, 2161, 2162, 2163, 2164, 2165, 2166, 2167, 2168, 2169, 2170, 2171, 2172, 2173, 2174, 2175, 2176, 2177, 2178, 2179, 2180, 2181, 2182, 2183, 2184, 2185, 2186, 2187, 2188, 2189, 2190, 2191, 2192, 2193, 2194, 2195, 2196, 2197, 2198, 2199, 2200, 2201, 2202, 2203, 2204, 2205, 2206, 2207, 2208, 2209, 2210, 2211, 2212, 2213, 2214, 2215, 2216, 2217, 2218, 2219, 2220, 2221, 2222, 2223, 2224, 2225, 2226, 2227, 2228, 2229, 2230, 2231, 2232, 2233, 2234, 2235, 2236, 2237, 2238, 2239, 2240, 2241, 2242, 2243, 2244, 2245, 2246, 2247, 2248, 2249, 2250, 2251, 2252, 2253, 2254, 2255, 2256, 2257, 2258, 2259, 2260, 2261, 2262, 2263, 2264, 2265, 2266, 2267, 2268, 2269, 2270, 2271, 2272, 2273, 2274, 2275, 2276, 2277, 2278, 2279, 2280, 2281, 2282, 2283, 2284, 2285, 2286, 2287, 2288, 2289, 2290, 2291, 2292, 2293, 2294, 2295, 2296, 2297, 2298, 2299, 2300, 2301, 2302, 2303, 2304, 2305, 2306, 2307, 2308, 2309, 2310, 2311, 2312, 2313, 2314, 2315, 2316, 2317, 2318, 2319, 2320, 2321, 2322, 2323, 2324, 2325, 2326, 2327, 2328, 2329, 2330, 2331, 2332, 2333, 2334, 2335, 2336, 2337, 2338, 2339, 2340, 2341, 2342, 2343, 2344, 2345, 2346, 2347, 2348, 2349, 2350, 2351, 2352, 2353, 2354, 2355, 2356, 2357, 2358, 2359, 2360, 2361, 2362, 2363, 2364, 2365, 2366, 2367, 2368, 2369, 2370, 2371, 2372, 2373, 2374, 2375, 2376, 2377, 2378, 2379, 2380, 2381, 2382, 2383, 2384, 2385, 2386, 2387, 2388, 2389, 2390, 2391, 2392, 2393, 2394, 2395, 2396, 2397, 2398, 2399, 2400, 2401, 2402, 2403, 2404, 2405, 2406, 2407, 2408, 2409, 2410, 2411, 2412, 2413, 2414, 2415, 2416, 2417, 2418, 2419, 2420, 2421, 2422, 2423, 2424, 2425, 2426, 2427, 2428, 2429, 2430, 2431, 2432, 2433, 2434, 2435, 2436, 2437, 2438, 2439, 2440, 2441, 2442, 2443, 2444, 2445, 2446, 2447, 2448, 2449, 2450, 2451, 2452, 2453, 2454, 2455, 2456, 2457, 2458, 2459, 2460, 2461, 2462, 2463, 2464, 2465, 2466, 2467, 2468, 2469, 2470, 2471, 2472, 2473, 2474, 2475, 2476, 2477, 2478, 2479, 2480, 2481, 2482, 2483, 2484, 2485, 2486, 2487, 2488, 2489, 2490, 2491, 2492, 2493, 2494, 2495, 2496, 2497, 2498, 2499, 2500, 2501, 2502, 2503, 2504, 2505, 2506, 2507, 2508, 2509, 2510, 2511, 2512, 2513, 2514, 2515, 2516, 2517, 2518, 2519, 2520, 2521, 2522, 2523, 2524, 2525, 2526, 2527, 2528, 2529, 2530, 2531, 2532, 2533, 2534, 2535, 2536, 2537, 2538, 2539, 2540, 2541, 2542, 2543, 2544, 2545, 2546, 2547, 2548, 2549, 2550, 2551, 2552, 2553, 2554, 2555, 2556, 2557, 2558, 2559, 2560, 2561, 2562, 2563, 2564, 2565, 2566, 2567, 2568, 2569, 2570, 2571, 2572, 2573, 2574, 2575, 2576, 2577, 2578, 2579, 2580, 2581, 2582, 2583, 2584, 2585, 2586, 2587, 2588, 2589, 2590, 2591, 2592, 2593, 2594, 2595, 2596, 2597, 2598, 2599, 2600, 2601, 2602, 2603, 2604, 2605, 2606, 2607, 2608, 2609, 2610, 2611, 2612, 2613, 2614, 2615, 2616, 2617, 2618, 2619, 2620, 2621, 2622, 2623, 2624, 2625, 2626, 2627, 2628, 2629, 2630, 2631, 2632, 2633, 2634, 2635, 2636, 2637, 2638, 2639, 2640, 2641, 2642, 2643, 2644, 2645, 2646, 2647, 2648, 2649, 2650, 2651, 2652, 2653, 2654, 2655, 2656, 2657, 2658, 2659, 2660, 2661, 2662, 2663, 2664, 2665, 2666, 2667, 2668, 2669, 2670, 2671, 2672, 2673, 2674, 2675, 2676, 2677, 2678, 2679, 26

- clear all
- set obs 100000 simulate random values on the square whose side is 2:
 - gen x=runiform(-1,1)
 - gen y=runiform(-1,1)
- identify random values that belong to the area of a circle $x^2 + y^2 \leq 1$
 - gen circle = (x^2 + y^2 <=1)
- di "the approximated value is:" 4*r(mean)
 - note that "4" corresponds to the area of the square whose side is 2
- visualization: tw (scatter y x)(scatter y x if circle==1), graphregion(color(white)) legend(label(1 "Square") label(2 "Disque"))

NEXT: LATEX AND ITS CONNECTION WITH R AND STATA !