Statistics/Data analysis

```
      name:  <unnamed>
       log:  A:\_maestria_unibo_(operacional)\4_econometrics_1\4_problem_sets\1_ps1\2_
> log\log.smcl
  log type:  smcl
 opened on:  14 Oct 2021, 19:31:16

1 .
2 . *#########################################################################
3 . * 1. Question 1.
4 . *#########################################################################
5 . *=========================================================================
6 . * 1.1. Generate sample from the random variables and its population parameters.
7 . *=========================================================================
8 . set obs 100 // Set the number of  observations for the random sample.
  Number of observations (_N) was 0, now 100.

9 . set seed 1015 // Set the seed for the pseudo-random number generator.

10.
11. matrix means = (10,15,15,10) //Vector of means for the drawnorm().

12. matrix varcov = (1,0.6,0,0.2\0.6,1,0,0.3\0,0,1,0\0.2,0.3,0,1) //Matrix of var-cov fo
  > r the drawnorm().

13. matrix list means // Displays a matrix.

  means[1,4]
      c1  c2  c3  c4
  r1  10  15  15  10

14. matrix list varcov

  symmetric varcov[4,4]
      c1  c2  c3  c4
  r1   1
  r2  .6   1
  r3   0   0   1
  r4  .2  .3   0   1

15.
16. drawnorm y x1 x2 x3, cov(varcov) means(means) //Generates a sample of the random var
  > iables with the specified parameters.

17. list y in 1/10

     +-----------+
     |         y |
     |-----------|
  1. | 9.703876  |
  2. | 9.624805  |
  3. | 8.257296  |
  4. | 7.851505  |
  5. | 9.348859  |
     |-----------|
  6. | 9.059323  |
  7. | 10.47427  |
  8. | 10.64235  |
  9. | 8.040435  |
 10. | 10.34571  |
     +-----------+
```

```
18. summarize // Calculate mean and sd for the data.
```

Variable	Obs	Mean	Std. dev.	Min	Max
y	100	9.83153	.990959	7.156042	12.93186
x1	100	14.77765	.9627169	12.82143	17.81787
x2	100	15.01672	.9772178	12.76052	17.20855
x3	100	9.938679	1.097261	7.359324	12.43131

```
19.
20. gen constant = 1

21.
22. mata
```

mata (type **end** to exit)

```
: //------------------------------------------------------------------------------
: // 1.1.a. Generate population values of the coefficients of X.
: //------------------------------------------------------------------------------
: //----- Calculate the vector of coeficcients for X.
:
: means = (10,15,15,10)' //Vector of means for the drawnorm().

: varcov = (1,0.6,0,0.2\0.6,1,0,0.3\0,0,1,0\0.2,0.3,0,1) //Matrix of var-cov for the d
> rawnorm().

: means
        1
  1 |  10 |
  2 |  15 |
  3 |  15 |
  4 |  10 |

: varcov
[symmetric]
        1    2    3    4
  1 |   1                |
  2 |  .6    1           |
  3 |   0    0    1      |
  4 |  .2   .3    0    1 |

:
: varcov_x = varcov[2..4,2..4]

: varcov_x
[symmetric]
        1    2    3
  1 |   1           |
  2 |   0    1      |
  3 |  .3    0    1 |

:
: cov_y = (varcov[1,2],varcov[1,3],varcov[1,4])'

: cov_y
        1
  1 |  .6 |
  2 |   0 |
  3 |  .2 |

:
: beta = invsym(varcov_x) * (cov_y)
```

```
: beta
                 1
  1 |  .5934065934
  2 |           0
  3 |   .021978022

:
: //----- Calculate B0.
:
: mean_x = (15,15,10)'

: mean_x
       1
  1 |  15
  2 |  15
  3 |  10

:
: beta_0 = 10 - (mean_x'*beta)

: beta_0
  .8791208791

:
: //-------------------------------------------------------------------------------
: // 1.1.c. OLS Estimator.
: //-------------------------------------------------------------------------------
:
: st_view(y=.,.,"y")

: st_view(x=.,.,("x1","x2","x3","constant"))

:
: beta_hat=invsym(x'x)*(x'y)

: beta_hat
                 1
  1 |  .6141202149
  2 |  .0337280483
  3 |  .1213489504
  4 | -.9562587499

:
: //-------------------------------------------------------------------------------
: // 1.1.d. SST, SSE, SSR.
: //-------------------------------------------------------------------------------
:
: st_view(y=.,.,"y")

: st_view(x=.,.,("x1","x2","x3","constant"))

:
: mean_vector_y = J(rows(y),1,mean(y))

: //mean_vector_y
: sst = (y-mean_vector_y)'(y-mean_vector_y)

: sst
  97.21797076

:
: vector_y_hat = x * beta_hat

: //vector_y_hat
: sse = (vector_y_hat - mean_vector_y)'(vector_y_hat - mean_vector_y)
```

```
: sse
  44.52903988

:
: vector_u_hat = y - vector_y_hat

: //vector_u_hat
: ssr = (vector_u_hat)'(vector_u_hat)

: ssr
  52.68893088

:
: sst
  97.21797076

: sse + ssr
  97.21797076

:
: //---------------------------------------------------------------------------
: // 1.1.e. r2 and adjusted r2.
: //---------------------------------------------------------------------------
:
: r_squared = sse/sst

: r_squared
  .4580330111

:
: adjusted_r_squared = 1 - ((ssr / (rows(x) - cols(x))) / (sst / (rows(x) - 1)))

: adjusted_r_squared
  .4410965427

:
: //---------------------------------------------------------------------------
: // 1.1.f. OLS residuals and fitted values of y.
: //---------------------------------------------------------------------------
:
: //vector_y_hat
:
: //vector_u_hat
:
: //---------------------------------------------------------------------------
: // 1.1.g. Sample average of the OLS residuals and sample covariance between
: /// regressors and the residuals.
> //---------------------------------------------------------------------------
:
: mean_vector_u_hat = mean(vector_u_hat)

: mean_vector_u_hat
  -3.88471e-13

:
: cov_xu_hat = x'vector_u_hat

: cov_xu_hat
                  1
  1 |  -5.92225e-10
  2 |  -5.90944e-10
  3 |  -3.82536e-10
  4 |  -3.88471e-11
```

```
: // Comment: The mean of the residuals and the covariance between the residuals and t
> he
: // regressors is virtually zero.
:
: //---------------------------------------------------------------------------
: // 1.1.h. Comparison betweenthe average fitted value of y and the average value
: /// of y.
> //---------------------------------------------------------------------------
:
: mean_y = mean(y)

:
: mean_vector_y_hat = mean(vector_y_hat)

:
: mean_vector_y_hat - mean_y
  3.89022e-13

:
: // A summary of the key values to compare with the OLS regression output.
:
: beta_hat
                  1
  1    .6141202149
  2    .0337280483
  3    .1213489504
  4   -.9562587499

: sst
  97.21797076

: sse
  44.52903988

: ssr
  52.68893088

: r_squared
  .4580330111

: adjusted_r_squared
  .4410965427

:
: end
```

```
23.
24. *===========================================================================
25. * 1.2. OLS regression in STATA and comparison with results from MATA.
26. *===========================================================================
27.
28. reg y x*
```

Source	SS	df	MS		
Model	44.5290399	3	14.8430133	Number of obs	= 100
Residual	52.6889309	96	.54884303	F(3, 96)	= 27.04
				Prob > F	= 0.0000
				R-squared	= 0.4580
				Adj R-squared	= 0.4411
Total	97.2179708	99	.981999705	Root MSE	= .74084

y	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
x1	.6141202	.0893371	6.87	0.000	.4367875	.7914529
x2	.033728	.0763631	0.44	0.660	-.1178514	.1853075
x3	.121349	.0782833	1.55	0.124	-.0340421	.27674
_cons	-.9562587	1.572035	-0.61	0.544	-4.076724	2.164206

```
29. corr y x*
(obs=100)
```

	y	x1	x2	x3
y	1.0000			
x1	0.6658	1.0000		
x2	0.0782	0.0658	1.0000	
x3	0.4332	0.4985	0.0424	1.0000

```
30. // The results are exactly the same.
31.
32. *================================================================================
33. * 1.3. 1000 random samples from the joint distribution above.
34. *================================================================================
35.
36. capture program drop random_sample

37.
38. program define random_sample, rclass // Define the name of the program.
    1.          drop _all
    2.          scalar drop _all
    3.          matrix drop _all
    4.          set more off
    5.          set obs 100 // Set the number of observations in the sample.
    6.          matrix varcov = (1,0.6,0,0.2\0.6,1,0,0.3\0,0,1,0\0.2,0.3,0,1) //
    7.          matrix  means = (10,15,15,10)'    //Vector of means for the drawnorm().
    8.          drawnorm y x1 x2 x3, cov(varcov) means(means)
    9.
39.         reg y x1 x2 x3
   10.          // Store regression coefficients in r() in order to return them in the si
  > mulation
40.         return scalar beta_0 = _b[_cons]  //_b[namevariable]
   11.          return scalar beta_1 = _b[x1]
   12.      return scalar beta_2 = _b[x2]
   13.          return scalar beta_3 = _b[x3]
   14.          // End of program
41. end

42.
43. *--------------------------------------------------------------------------------
44. * 1.3.a. Estimation of parameters from 1000 replications.
45. *--------------------------------------------------------------------------------
46.
47. simulate ///
  > beta_0_hat = r(beta_0) ///
  > beta_1_hat = r(beta_1)  ///
  > beta_2_hat = r(beta_2) ///
  > beta_3_hat = r(beta_3), reps(1000)  ///
  > saving(0_data\coefficient_estimators, replace) seed(1015): random_sample

      Command: random_sample
   beta_0_hat: r(beta_0)
   beta_1_hat: r(beta_1)
   beta_2_hat: r(beta_2)
   beta_3_hat: r(beta_3)
```

```
Simulations (1,000)
----+--- 1 ---+--- 2 ---+--- 3 ---+--- 4 ---+--- 5
..................................................    50
..................................................   100
..................................................   150
..................................................   200
..................................................   250
..................................................   300
..................................................   350
..................................................   400
..................................................   450
..................................................   500
..................................................   550
..................................................   600
..................................................   650
..................................................   700
..................................................   750
..................................................   800
..................................................   850
..................................................   900
..................................................   950
.................................................. 1,000
```

```
48.
49. *------------------------------------------------------------------------------
50. * 1.3.b. Unbiasness of the estimators of beta (against parameter beta).
51. *------------------------------------------------------------------------------
52.
53. egen mean_beta_0_hat = mean(beta_0_hat)

54. egen mean_beta_1_hat = mean(beta_1_hat)

55. egen mean_beta_2_hat = mean(beta_2_hat)

56. egen mean_beta_3_hat = mean(beta_3_hat)

57.
58. gen diff_b_0 = beta_0_hat - mean_beta_0_hat

59. gen diff_b_1 = beta_1_hat - mean_beta_1_hat

60. gen diff_b_2 = beta_2_hat - mean_beta_2_hat

61. gen diff_b_3 = beta_3_hat - mean_beta_3_hat

62.
63. foreach var in beta_0_hat beta_1_hat beta_2_hat beta_3_hat{
    2. summarize `var'
    3. }
```

Variable	Obs	Mean	Std. dev.	Min	Max
beta_0_hat	**1,000**	**.8302528**	**1.838884**	**-4.651379**	**5.667601**

Variable	Obs	Mean	Std. dev.	Min	Max
beta_1_hat	**1,000**	**.5962717**	**.0876936**	**.1954905**	**.9266147**

Variable	Obs	Mean	Std. dev.	Min	Max
beta_2_hat	**1,000**	**.0033474**	**.0819166**	**-.2419794**	**.2505844**

Variable	Obs	Mean	Std. dev.	Min	Max
beta_3_hat	**1,000**	**.0176864**	**.0868876**	**-.2672546**	**.3348112**

```
64.
65. foreach var in diff_b_0 diff_b_1 diff_b_2 diff_b_3{
    2. summarize `var'
    3. }
```

Variable	Obs	Mean	Std. dev.	Min	Max
diff_b_0	1,000	-1.31e-08	1.838884	-5.481631	4.837348

Variable	Obs	Mean	Std. dev.	Min	Max
diff_b_1	1,000	3.81e-09	.0876936	-.4007812	.330343

Variable	Obs	Mean	Std. dev.	Min	Max
diff_b_2	1,000	-7.69e-10	.0819166	-.2453268	.247237

Variable	Obs	Mean	Std. dev.	Min	Max
diff_b_3	1,000	-1.45e-09	.0868876	-.284941	.3171248

```
66.
67. *------------------------------------------------------------------------------
68. * 1.3.c. beta_hat distribution plots
69. *------------------------------------------------------------------------------
70.
71. foreach var  of varlist beta_0_hat beta_1_hat beta_2_hat beta_3_hat{
    2.         histogram `var', normal name(`var', replace)
    3.         local graphnames `graphnames' `var'
    4. }
  (bin=29, start=-4.6513786, width=.35582689)
  (bin=29, start=.19549048, width=.02521118)
  (bin=29, start=-.24197945, width=.01698496)
  (bin=29, start=-.26725462, width=.02076089)

72.
73. graph combine `graphnames'

74. graph save 3_graphs\betas.gph, replace
  file 3_graphs\betas.gph saved

75.
76. //Comment: They look pretty normal, no pun intended.
77.
78. *##############################################################################
79. * 2. Question 2.
80. *##############################################################################
81. *==============================================================================
82. * 2.1. Load dataset in STATA and MATA and calculate the regression model in MATA.
83. *==============================================================================
84. clear all

85. use 0_data\ps1_group15

86.
87. summarize
```

Variable	Obs	Mean	Std. dev.	Min	Max
workedm	322,542	.5656597	.4956708	0	1
weeksm1	322,542	20.84158	22.28615	0	52
hourswm	322,542	18.80672	18.91383	0	99
incomem	322,542	7168.956	10839.99	0	260308
kidcount	322,542	2.553159	.8104265	2	12
twin_birth_2	322,542	.0094747	.0968762	0	1
same_sex	322,542	.5053233	.4999724	0	1
morekids	322,542	.4022949	.4903616	0	1
blackm	322,542	.1191783	.3239987	0	1
hispm	322,542	.0302472	.1712671	0	1
othracem	322,542	.0288366	.1673472	0	1

educm	322,542	12.12603	2.402849	0	20
agem1	322,542	30.12355	3.506812	21	35
agefstm	322,542	20.13954	2.950454	15	33

```
88.
89. gen constant = 1

90. mata
```

mata (type **end** to exit)

```
: 
: st_view(y = .,.,"hourswm")

: st_view(x = .,.,("morekids","educm","agefstm","blackm","hispm","othracem","agem1","c
> onstant"))

: 
: beta = invsym(x'x) * (x'y)

: beta
```

	1
1	-6.374661875
2	.7717738957
3	-1.60535276
4	5.431590302
5	2.491738478
6	4.365351396
7	.8708522197
8	17.26199847

```
: 
: // I cannot compute the partitioned regression in mata because of the inability
: // to create an identity matrix of dimension = n (Insuficiente memory).
: 
: end
```

```
91.
92. *================================================================================
93. * 2.2. Calculate the regression model in STATA and compare it with the one
94. * obtained in MATA.
95. *================================================================================
96.
97. reg hourswm morekids educm agem1 agefstm blackm hispm othracem
```

Source	SS	df	MS		
Model	8418182.8	7	1202597.54	Number of obs	= 322,542
Residual	106965359	322,534	331.640568	F(7, 322534)	= 3626.21
				Prob > F	= 0.0000
				R-squared	= 0.0730
				Adj R-squared	= 0.0729
Total	115383542	322,541	357.732945	Root MSE	= 18.211

hourswm	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
morekids	-6.374662	.0684161	-93.17	0.000	-6.508756	-6.240568
educm	.7717739	.0150779	51.19	0.000	.7422216	.8013262
agem1	.8708522	.0102514	84.95	0.000	.8507597	.8909447
agefstm	-1.605353	.0133654	-120.11	0.000	-1.631548	-1.579157
blackm	5.43159	.101549	53.49	0.000	5.232557	5.630623
hispm	2.491738	.1917857	12.99	0.000	2.115844	2.867633
othracem	4.365351	.1924748	22.68	0.000	3.988106	4.742597
_cons	17.262	.3179841	54.29	0.000	16.63876	17.88524

```
98.
99. // The results are the same.
100
101 *================================================================================
102 * 2.3.a.
103 *================================================================================
104
105 cls

106 //----- Remove the effect of educm from hourswm.
107 reg hourswm educm
```

Source	SS	df	MS		
Model	206732.46	1	206732.46	Number of obs	= 322,542
Residual	115176809	322,540	357.093102	F(1, 322540)	= 578.93
				Prob > F	= 0.0000
				R-squared	= 0.0018
				Adj R-squared	= 0.0018
Total	115383542	322,541	357.732945	Root MSE	= 18.897

hourswm	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
educm	.3331849	.0138475	24.06	0.000	.3060442	.3603256
_cons	14.76651	.1711802	86.26	0.000	14.431	15.10202

```
108 predict e2_tilda, residuals

109
110 //----- Remove the effect of educm from morekids
111 reg morekids educm
```

Source	SS	df	MS		
Model	1836.36986	1	1836.36986	Number of obs	= 322,542
Residual	75720.0515	322,540	.234761739	F(1, 322540)	= 7822.27
				Prob > F	= 0.0000
				R-squared	= 0.0237
				Adj R-squared	= 0.0237
Total	77556.4213	322,541	.240454458	Root MSE	= .48452

morekids	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
educm	-.0314023	.0003551	-88.44	0.000	-.0320982	-.0307064
_cons	.7830797	.0043891	178.41	0.000	.7744772	.7916822

```
112 predict x1_tilda, residuals

113
114 //----- Regress filtered-out y onto filetred out x, filter being educm.
115 reg e2_tilda x1_tilda
```

Source	SS	df	MS		
Model	1210401.47	1	1210401.47	Number of obs	= 322,542
Residual	113966408	322,540	353.340385	F(1, 322540)	= 3425.60
				Prob > F	= 0.0000
				R-squared	= 0.0105
				Adj R-squared	= 0.0105
Total	115176809	322,541	357.091995	Root MSE	= 18.797

e2_tilda	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
x1_tilda	-3.998152	.0683111	-58.53	0.000	-4.132039	-3.864264
_cons	2.96e-07	.0330981	0.00	1.000	-.0648711	.0648717

```
116 reg hourswm morekids educm
```

Source	SS	df	MS	Number of obs	=	322,542
				F(2, 322539)	=	2005.33
Model	1417133.91	2	708566.954	Prob > F	=	0.0000
Residual	113966408	322,539	353.341481	R-squared	=	0.0123
				Adj R-squared	=	0.0123
Total	115383542	322,541	357.732945	Root MSE	=	18.797

hourswm	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
morekids	-3.998152	.0683112	-58.53	0.000	-4.13204	-3.864264
educm	.2076339	.0139406	14.89	0.000	.1803107	.2349571
_cons	17.89738	.1784834	100.27	0.000	17.54756	18.2472

```
117
118 //----- Remove the effect of morekids from hourswm.
119
120 reg hourswm morekids
```

Source	SS	df	MS	Number of obs	=	322,542
				F(1, 322540)	=	3786.24
Model	1338749.89	1	1338749.89	Prob > F	=	0.0000
Residual	114044792	322,540	353.583406	R-squared	=	0.0116
				Adj R-squared	=	0.0116
Total	115383542	322,541	357.732945	Root MSE	=	18.804

hourswm	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
morekids	-4.154711	.0675207	-61.53	0.000	-4.28705	-4.022373
_cons	20.47814	.0428262	478.17	0.000	20.3942	20.56208

```
121 predict e1_tilda, residuals

122
123 //----- Remove the effect of morekids from educm.
124 reg educm morekids
```

Source	SS	df	MS	Number of obs	=	322,542
				F(1, 322540)	=	7822.27
Model	44094.0925	1	44094.0925	Prob > F	=	0.0000
Residual	1818156.04	322,540	5.63699397	R-squared	=	0.0237
				Adj R-squared	=	0.0237
Total	1862250.13	322,541	5.77368498	Root MSE	=	2.3742

educm	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
morekids	-.7540173	.0085254	-88.44	0.000	-.7707269	-.7373078
_cons	12.42936	.0054074	2298.59	0.000	12.41877	12.43996

```
125 predict x2_tilda, residuals

126
127 //----- Remove the effect of educm from morekids.
```

```
128 reg e1_tilda x2_tilda
```

Source	SS	df	MS		
Model	78384.007	1	78384.007	Number of obs	= 322,542
Residual	113966407	322,540	353.340383	F(1, 322540)	= 221.84
				Prob > F	= 0.0000
Total	114044791	322,541	353.582308	R-squared	= 0.0007
				Adj R-squared	= 0.0007
				Root MSE	= 18.797

e1_tilda	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
x2_tilda	.2076338	.0139406	14.89	0.000	.1803107	.234957
_cons	5.33e-07	.0330981	0.00	1.000	-.0648709	.0648719

```
129 reg hourswm morekids educm
```

Source	SS	df	MS		
Model	1417133.91	2	708566.954	Number of obs	= 322,542
Residual	113966408	322,539	353.341481	F(2, 322539)	= 2005.33
				Prob > F	= 0.0000
Total	115383542	322,541	357.732945	R-squared	= 0.0123
				Adj R-squared	= 0.0123
				Root MSE	= 18.797

hourswm	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
morekids	-3.998152	.0683112	-58.53	0.000	-4.13204	-3.864264
educm	.2076339	.0139406	14.89	0.000	.1803107	.2349571
_cons	17.89738	.1784834	100.27	0.000	17.54756	18.2472

```
130
131 *##############################################################################
132 * n. Close log.
133 *##############################################################################
134
135 log close
      name:  <unnamed>
       log:  A:\_maestria_unibo_(operacional)\4_econometrics_1\4_problem_sets\1_ps1\2_
> log\log.smcl
  log type:  smcl
 closed on:  14 Oct 2021, 19:31:30
```