Statistics/Data analysis

```
      name:  <unnamed>
       log:  C:\data\AsianBaro\logofSimulationforEntropy.smcl
  log type:  smcl
 opened on:  10 Sep 2024, 22:02:19

1 .
2 . *do file for Simulation of Education's Entropy
3 . *filename entropySimulEduc.do
4 . * Wendy Olsen
5 . * grateful thanks to Mr Ziyang Zhou - Univ. of Manchester
6 . * Univ of Manchester 2024
7 .
8 . * Stata 18
9 . *This file is part 4 of the entropy project.
10. *This file has two aims.  4a) First, calculate Entropy for 2 variables, one datafram
  > e, using Stata.
11.
12. *Second, do an aggregate exercise in **simulating** EDUC and calculate Entropy manua
  > lly using that S=1000 repeated samples bloc of vectors.
13.  ssc install estout
  checking estout consistency and verifying not already installed...
  all files already exist and are up to date.

14.
15.  ***Data already exist but results go in \results folder ***
16. cd "C:\data\AsianBaro"
  C:\data\AsianBaro

17.
18. ** Part 4a Calculate Entropy in Stata for a single variable, then 2 variables.  One
  > needs to recognise the  number of cells in 2-var exercise depends on the unique valu
  > es of each one, k*j.  Whilst N is still the sum of all cell values.
19. cd "C:\data\AsianBaro"
  C:\data\AsianBaro

20. use "C:/data/AsianBaro/data/AsianBaro2019revForEntropy.dta", clear

21. tab income inc2_2
```

Household Income Decile	inc2_2 0	1	Total
Worst-Off	1,293	0	1,293
2	0	124	124
3	0	985	985
4	0	126	126
5	0	2,125	2,125
6	0	125	125
7	0	353	353
8	0	131	131
9	0	20	20
Best-Off	0	36	36
Total	1,293	4,025	5,318

```
22. summ( edu2_1 edu2_2 edu2_3 edu2_4 edu2_5 ) if edu1_4==1
```

Variable	Obs	Mean	Std. dev.	Min	Max
edu2_1	1,381	1	0	1	1
edu2_2	1,381	1	0	1	1
edu2_3	1,381	1	0	1	1
edu2_4	1,381	1	0	1	1
edu2_5	1,381	0	0	0	0

```
23. egen sumedu1_1 = sum(edu1_1)

24. egen sumedu1_2 = sum(edu1_2)

25. egen sumedu1_3 = sum(edu1_3)

26. egen sumedu1_4 = sum(edu1_4)

27. egen sumedu1_5 = sum(edu1_5)

28. egen countedu1_1 = sum(edu1_1)

29. egen countedu1_2 = sum(edu1_2)

30. egen countedu1_3 = sum(edu1_3)

31. egen countedu1_4 = sum(edu1_4)

32. egen countedu1_5 = sum(edu1_5)

33. gen Nedu=countedu1_1+countedu1_2+countedu1_3+countedu1_4+countedu1_5

34. gen entropyofEduc1 = -((sumedu1_1/5318)*ln(sumedu1_1/5318)+(sumedu1_2/5318)*ln(sumed
  > u1_2/5318)+(sumedu1_3/5318)*ln(sumedu1_3/5318)+(sumedu1_4/5318)*ln(sumedu1_4/5318)+(
  > sumedu1_5/5318)*ln(sumedu1_5/5318))

35. summ(entropyofEduc1)
```

Variable	Obs	Mean	Std. dev.	Min	Max
entropyofE~1	**5,318**	**1.529034**	**0**	**1.529034**	**1.529034**

```
36.
37.
38. summ( edu2_1 edu2_2 edu2_3 edu2_4 edu2_5 ) if edu1_4==1
```

Variable	Obs	Mean	Std. dev.	Min	Max
edu2_1	**1,381**	**1**	**0**	**1**	**1**
edu2_2	**1,381**	**1**	**0**	**1**	**1**
edu2_3	**1,381**	**1**	**0**	**1**	**1**
edu2_4	**1,381**	**1**	**0**	**1**	**1**
edu2_5	**1,381**	**0**	**0**	**0**	**0**

```
39. egen sumedu2_1 = sum(edu2_1)

40. egen sumedu2_2 = sum(edu2_2)

41. egen sumedu2_3 = sum(edu2_3)

42. egen sumedu2_4 = sum(edu2_4)

43. egen sumedu2_5 = sum(edu2_5)

44. egen countedu2_1 = sum(edu2_1)

45. egen countedu2_2 = sum(edu2_2)

46. egen countedu2_3 = sum(edu2_3)
```

```
47. egen countedu2_4 = sum(edu2_4)

48. egen countedu2_5 = sum(edu2_5)

49. gen Nedu2=countedu2_1+countedu2_2+countedu2_3+countedu2_4+countedu2_5

50. summ(Nedu2)
```

Variable	Obs	Mean	Std. dev.	Min	Max
Nedu2	**5,318**	**15884**	**0**	**15884**	**15884**

```
51. #This cumulative amount depends upon Nedu and the actual data.
  Unknown #command
52. gen entropyofEduc2 = -((sumedu2_1/5318)*ln(sumedu2_1/5318)+(sumedu2_2/5318)*ln(sumed
  > u2_2/5318)+(sumedu2_3/5318)*ln(sumedu2_3/5318)+(sumedu2_4/5318)*ln(sumedu2_4/5318)+(
  > sumedu2_5/5318)*ln(sumedu2_5/5318))

53. *Suppose the N is still 5318, the raw number of respondents:
54. summ(entropyofEduc2)
```

Variable	Obs	Mean	Std. dev.	Min	Max
entropyofE~2	**5,318**	**1.245304**	**0**	**1.245304**	**1.245304**

```
55. *Suppose the N is the number of responses, which are in columns 1 to 5.  First colum
  > na has 5318 but the others are data, empirical, unknown.
56.
57. gen entropyofEduc2withN2 = -((sumedu2_1/Nedu2)*ln(sumedu2_1/Nedu2)+(sumedu2_2/Nedu2)
  > *ln(sumedu2_2/Nedu2)+(sumedu2_3/Nedu2)*ln(sumedu2_3/Nedu2)+(sumedu2_4/Nedu2)*ln(sume
  > du2_4/Nedu2)+(sumedu2_5/Nedu2)*ln(sumedu2_5/Nedu2))

58. summ(entropyofEduc2withN2)
```

Variable	Obs	Mean	Std. dev.	Min	Max
entropyof~N2	**5,318**	**1.511146**	**0**	**1.511146**	**1.511146**

```
59. summ(Nedu2)
```

Variable	Obs	Mean	Std. dev.	Min	Max
Nedu2	**5,318**	**15884**	**0**	**15884**	**15884**

```
60. ** Part 4b Aggregate Exercise - see separate do file.
61. * Hypothesis.  Using simulation, the MSE of H is higher for ordinal education than f
  > or cumulative education when it is multinomial in 5 categories.
62. *   We emulated education in five levels from the Asian Barometers, unweighted.
63. *this dataset has nothing in common with the rest of the data.
64.
65. use "data\edtmp.dta", clear

66. *Note the edtmp file has the standard, distinct encodings.
67. de

  Contains data from data\edtmp.dta
   Observations:         1,000
      Variables:             5                  10 Sep 2024 15:23
```

Variable name	Storage type	Display format	Value label	Variable label
X1	long	%12.0g		
X2	long	%12.0g		
X3	long	%12.0g		
X4	long	%12.0g		
X5	long	%12.0g		

```
  Sorted by:
```

```
68. summ(X1)
```

Variable	Obs	Mean	Std. dev.	Min	Max
X1	**1,000**	**1594.078**	**33.56514**	**1464**	**1701**

```
69. *drop N
70. *drop p1 p2 p3 p4 p5
71. *drop hsim
72. gen N=5318

73. gen p1 = X1/N

74. gen p2 = X2/N

75. gen p3 = X3/N

76. gen p4 = X4/N

77. gen p5 = X5/N

78. gen hsim =- [(p1*ln(p1))+(p2*ln(p2))+(p3*ln(p3))+(p4*ln(p4))+(p5*ln(p5))]

79. summ(hsim)
```

Variable	Obs	Mean	Std. dev.	Min	Max
hsim	**1,000**	**1.528825**	**.0054934**	**1.508636**	**1.545974**

```
80.  *Helpful notes egen [type] newvar = fcn(arguments) [if] [in] [, options]
81.  * &      pctile(exp) [, p(#) autotype]
82. egen hsimUL = pctile(hsim ), p(97.5)

83. egen hsimLL = pctile(hsim), p(2.5)

84. summ hsimUL hsim hsimLL
```

Variable	Obs	Mean	Std. dev.	Min	Max
hsimUL	**1,000**	**1.539725**	**0**	**1.539725**	**1.539725**
hsim	**1,000**	**1.528825**	**.0054934**	**1.508636**	**1.545974**
hsimLL	**1,000**	**1.517561**	**0**	**1.517561**	**1.517561**

```
85. egen hsimmode=pctile(hsim), p(50)

86. egen hsimmean=mean(hsim)

87. summ (hsimmean hsimmode)
```

Variable	Obs	Mean	Std. dev.	Min	Max
hsimmean	**1,000**	**1.528825**	**0**	**1.528825**	**1.528825**
hsimmode	**1,000**	**1.529082**	**0**	**1.529082**	**1.529082**

```
88. *the MSE is defined as the sum of squared deviations, divided by N.
89. *there is one squared deviation per Sample drawn. The SquDev's are the value (Hi - H
  > mean )^2 for all 1000 sample replicates, i.
90. gen hsimMSEsubs=(hsim- hsimmean)^2

91. egen tempsum=total(hsimMSEsubs)
```

```
92. gen hsimMSEaggreg =tempsum/1000

93. summarize(hsimMSEaggreg)
```

Variable	Obs	Mean	Std. dev.	Min	Max
hsimMSEagg~g	**1,000**	**.0000301**	**0**	**.0000301**	**.0000301**

```
94.
95. * generate relative entropy for distinct encoding.
96. gen RSIsim =- [(p1*ln(p1))+(p2*ln(p2))+(p3*ln(p3))+(p4*ln(p4))+(p5*ln(p5))] / ln(5)

97. summ(RSIsim)
```

Variable	Obs	Mean	Std. dev.	Min	Max
RSIsim	**1,000**	**.9499122**	**.0034132**	**.9373684**	**.9605677**

```
98.  *Helpful notes egen [type] newvar = fcn(arguments) [if] [in] [, options]
99.  * &      pctile(exp) [, p(#) autotype]
100 egen RSIsimUL = pctile(RSIsim ), p(97.5)

101 egen RSIsimLL = pctile(RSIsim), p(2.5)

102 summarize  RSIsimUL RSIsim RSIsimLL
```

Variable	Obs	Mean	Std. dev.	Min	Max
RSIsimUL	**1,000**	**.9566852**	**0**	**.9566852**	**.9566852**
RSIsim	**1,000**	**.9499122**	**.0034132**	**.9373684**	**.9605677**
RSIsimLL	**1,000**	**.9429135**	**0**	**.9429135**	**.9429135**

```
103
104 *the MSE is defined as the sum of squared deviations, divided by N.
105 egen RSIsimmean=pctile(RSIsim), p(50)

106 gen RSIsimMSEsubs=(RSIsim- RSIsimmean)^2

107 egen tempsum2 = total(RSIsimMSEsubs)

108 gen RSIsimMSEaggreg = tempsum2/1000

109 summarize(RSIsimMSEaggreg)
```

Variable	Obs	Mean	Std. dev.	Min	Max
RSIsimMSEa~g	**1,000**	**.0000117**	**0**	**.0000117**	**.0000117**

```
110
111 *Step 2. Create a block of data, EdCumtmp.dta"
112 *This is cumulative encodings of the previous dataset.
113
114 gen X1cum=(X1+X2+X3+X4+X5)

115 gen X2cum=(X2+X3+X4+X5)

116 gen X3cum=(X3+X4+X5)

117 gen X4cum=(X4+X5)
```

```
118 gen X5cum=(X5)

119 gen newN = (X1+2*X2+3*X3+4*X4+5*X5)

120 gen p1cum = X1cum/newN

121 gen p2cum = X2cum/newN

122 gen p3cum = X3cum/newN

123 gen p4cum = X4cum/newN

124 gen p5cum = X5cum/newN

125 gen hsimcum =- [(p1cum*ln(p1cum))+(p2cum*ln(p2cum))+(p3cum*ln(p3cum))+(p4cum*ln(p4cu
  > m))+(p5cum*ln(p5cum))]

126 summ(hsimcum)
```

Variable	Obs	Mean	Std. dev.	Min	Max
hsimcum	**1,000**	**1.512514**	**.0025669**	**1.503691**	**1.520409**

```
127  *Helpful notes egen [type] newvar = fcn(arguments) [if] [in] [, options]
128  * &      pctile(exp) [, p(#) autotype]
129 egen hsimcumUL = pctile(hsimcum ), p(97.5)

130 egen hsimcumLL = pctile(hsimcum), p(2.5)

131 summarize  hsimcumUL hsimcum hsimcumLL
```

Variable	Obs	Mean	Std. dev.	Min	Max
hsimcumUL	**1,000**	**1.517264**	**0**	**1.517264**	**1.517264**
hsimcum	**1,000**	**1.512514**	**.0025669**	**1.503691**	**1.520409**
hsimcumLL	**1,000**	**1.507122**	**0**	**1.507122**	**1.507122**

```
132
133 *the MSE is defined as the sum of squared deviations, divided by N.
134 egen hsimcummean=pctile(hsimcum), p(50)

135 gen hsimcumMSEsubs=(hsimcum- hsimcummean)^2

136 egen tempsum3 = total(hsimcumMSEsubs)

137 gen hsimcumMSEaggreg  = tempsum3/1000

138 summarize(hsimcumMSEaggreg)
```

Variable	Obs	Mean	Std. dev.	Min	Max
hsimcumMSE~g	**1,000**	**6.59e-06**	**0**	**6.59e-06**	**6.59e-06**

```
139
140 * generate relative entropy for cumulative encoding.
141 gen RSIsimcum =- [(p1cum*ln(p1cum))+(p2cum*ln(p2cum))+(p3cum*ln(p3cum))+(p4cum*ln(p4
  > cum))+(p5cum*ln(p5cum))] / ln(5)

142 summ(RSIsimcum)
```

Variable	Obs	Mean	Std. dev.	Min	Max
RSIsimcum	**1,000**	**.9397781**	**.0015949**	**.934296**	**.9446833**

```
143  *Helpful notes egen [type] newvar = fcn(arguments) [if] [in] [, options]
144  * &      pctile(exp) [, p(#) autotype]
145 egen RSIsimcumUL = pctile(RSIsimcum ), p(97.5)

146 egen RSIsimcumLL = pctile(RSIsimcum), p(2.5)

147 summarize  RSIsimcumUL RSIsimcum RSIsimcumLL
```

Variable	Obs	Mean	Std. dev.	Min	Max
RSIsimcumUL	**1,000**	**.9427294**	**0**	**.9427294**	**.9427294**
RSIsimcum	**1,000**	**.9397781**	**.0015949**	**.934296**	**.9446833**
RSIsimcumLL	**1,000**	**.9364276**	**0**	**.9364276**	**.9364276**

```
148
149 *the MSE is defined as the sum of squared deviations, divided by N.
150 egen RSIsimcummean=pctile(RSIsimcum), p(50)

151 gen RSIsimcumMSEsubs=(RSIsimcum- RSIsimcummean)^2

152 egen tempsum4= total(RSIsimcumMSEsubs)

153 gen RSIsimcumMSEaggreg  = tempsum4/1000

154 summarize(RSIsimcumMSEaggreg)
```

Variable	Obs	Mean	Std. dev.	Min	Max
RSIsimcumM~g	**1,000**	**2.54e-06**	**0**	**2.54e-06**	**2.54e-06**

```
155
156
157 log close
       name:  <unnamed>
        log:  C:\data\AsianBaro\logofSimulationforEntropy.smcl
   log type:  smcl
  closed on:  10 Sep 2024, 22:02:24
```