

```

1  insheet using "${par_path}\parameters_ES_model2.csv", clear
2  mkmat estimate, matrix(beta)
3
4  insheet using "${par_path}\vres_ES_model2.csv", clear
5  mkmat v, matrix(var)
6
7  insheet using "${par_path}\bound_ES.csv", clear
8  mkmat m, matrix(m)
9
10
11  use "${out_path}\temp_ES.dta", clear
12  sort sa0100 sa0010 im0100
13
14  gen x2 = 0 if _n == 1
15  gen x1 = 6*$seed if _n == 1
16
17  replace x1 = mod(x1[_n-1]*20077 + 12345,2^16) if _n>1
18  replace x2 = mod(int((x1[_n-1]*20077 + 12345 - x1)/2^16)+mod(16838*x1[_n-1]+20077*x2[_n-1],2^16),2^15) if _n>1
19
20  gen double z=2^16*x2+x1
21  format z %16.0g
22
23  gen u=z/2^31
24
25  gen cfood = hi0100*12
26  gen rent = hb2300*12
27  replace rent = 0 if missing(hb2300) == 1
28  gen l_cfood = log(max(cfood,1))
29  gen l_rent = log(max(rent,1))
30  gen head_male = (ra0200 == 1)
31  gen owner = (inlist(hb0300,1,2))
32  gen free_use = (hb0300 == 4)
33  gen hhsz1 = (dh0001 == 1)
34  gen hhsz3 = (dh0001 >= 3)
35  gen agerp_1 = (ra0300 < 30)
36  gen agerp_2 = (ra0300 < 40 & ra0300 >= 30)
37  gen agerp_3 = (ra0300 < 50 & ra0300 >= 40)
38  gen agerp_4 = (ra0300 < 60 & ra0300 >= 50)
39  gen agerp_5 = (ra0300 < 70 & ra0300 >= 60)
40  gen agerp_6 = (ra0300 >= 70)
41  gen number_children_1 = (number_children == 1)
42  gen number_children_2 = (number_children == 2)
43  gen number_children_3 = (number_children >= 3)
44  gen labour_status_1 = (inlist(pe0100a,1,2))
45  gen labour_status_2 = (inlist(pe0100a,3,4,6,7,8,9))
46  gen labour_status_3 = (pe0100a == 5)
47  gen diploma_1 = (pa0200 == 1)
48  gen diploma_2 = (pa0200 == 2)
49  gen diploma_5 = (pa0200 == 5)
50
51  /* computing quintiles */
52  forvalues i = 1/5{
53      _pctile di2000 if im0100 == `i' [weight=hw0010], nq(5)
54      gen q1_`i' = r(r1)
55      gen q2_`i' = r(r2)
56      gen q3_`i' = r(r3)
57      gen q4_`i' = r(r4)
58  }
59
60  gen q1 = (q1_1+q1_2+q1_3+q1_4+q1_5)/5
61  gen q2 = (q2_1+q2_2+q2_3+q2_4+q2_5)/5
62  gen q3 = (q3_1+q3_2+q3_3+q3_4+q3_5)/5
63  gen q4 = (q4_1+q4_2+q4_3+q4_4+q4_5)/5
64
65  gen income_quintile_1 = (di2000 <= q1)
66  gen income_quintile_2 = (di2000 > q1 & di2000 <= q2)
67  gen income_quintile_3 = (di2000 > q2 & di2000 <= q3)
68  gen income_quintile_4 = (di2000 > q3 & di2000 <= q4)
69  gen income_quintile_5 = (di2000 > q4)
70
71  gen lbound = cfood+rent
72  gen ubound = m[1,1]

```

```

73  gen a = log(lbound)
74  gen b = log(ubound)
75  #delimit ;
76  gen Xbeta = beta[1,1]+beta[2,1]*l_cfood+beta[3,1]*l_cfood^2+beta[4,1]*l_cfood^3
77             +beta[5,1]*income_quintile_2+beta[6,1]*income_quintile_3+beta[7,1]*
income_quintile_4+beta[8,1]*income_quintile_5
78             +beta[9,1]*l_rent+beta[10,1]*l_rent^2+beta[11,1]*l_rent^3
79             +beta[12,1]*agerp_1+beta[13,1]*agerp_2+beta[14,1]*agerp_4+beta[15,1]*
agerp_5+beta[16,1]*agerp_6
80             +beta[17,1]*head_male
81             +beta[18,1]*hhsize_1+beta[19,1]*hhsize_3
82             +beta[20,1]*number_children_1+beta[21,1]*number_children_2+beta[22,1]*
number_children_3
83             +beta[23,1]*owner+beta[24,1]*free_use
84             +beta[25,1]*diploma_1+beta[26,1]*diploma_2+beta[27,1]*diploma_5
85             +beta[28,1]*labour_status_2+beta[29,1]*labour_status_3
86             +beta[30,1]*income_quintile_2*l_cfood+beta[31,1]*income_quintile_2*(
l_cfood^2)+beta[32,1]*income_quintile_2*(l_cfood^3)
87             +beta[33,1]*income_quintile_3*l_cfood+beta[34,1]*income_quintile_3*(
l_cfood^2)+beta[35,1]*income_quintile_3*(l_cfood^3)
88             +beta[36,1]*income_quintile_4*l_cfood+beta[37,1]*income_quintile_4*(
l_cfood^2)+beta[38,1]*income_quintile_4*(l_cfood^3)
89             +beta[39,1]*income_quintile_5*l_cfood+beta[40,1]*income_quintile_5*(
l_cfood^2)+beta[41,1]*income_quintile_5*(l_cfood^3);
90  #delimit cr
91
92  gen Phi_a = normal((a-Xbeta)/sqrt(var[1,1]))
93  gen Phi_b = normal((b-Xbeta)/sqrt(var[1,1]))
94
95  gen di3001 = round(exp(Xbeta + invnormal((Phi_a + (Phi_b - Phi_a)*u))*sqrt(var[1,1])))
96  keep sa0100 sa0010 im0100 di3001
97  save "${out_path}\temp_ES.dta", replace
98

```