```
1  *Created by RM on 2018.09.17
2  *For ECON 675, PS 1, Q 2
3
4  global data "/Users/russellmorton/Desktop/Coursework/Fall
   2018/Econ 675/Problem Sets/Problem Set Data"
5
6  clear
7  set more off
8
9  /* PS 2 Q4: Matrix Implementation of WLS with W = I */
10
11 ***Generate Data
12
13
14 local obs = 1000
15 local beta0 = 0
16 local beta1 = 1
17 local beta2 = 5
18 local alpha = .9
19
20 set obs `obs'
21 g x_1 = runiform() * 100
22 g x_2 = runiform() * 20 + 30
23 g intercept = 1
24 g epsilon = rnormal() * 10
25
26 g y = x_1 * `beta1' + x_2 * `beta2' + epsilon
27
28 **4a: symmetric point estimate of beta
29 *bring data into mata and estimate
30
31 mata: mata clear
32 capture eret clear
33
34 mata:
35
36     X = st_data(.,("intercept", "x_1", "x_2"))
37     y = st_data(.,("y"))
38     Xrows = rows(X)
39     Xcols = cols(X)
40
41     betahat = luinv(X' * X) * (X' * y)
42
43     epsilonhat = y - X * betahat
44     s2 = epsilonhat' * epsilonhat :* (1/(Xrows-Xcols))
45     v0_hat = s2 * X' * X :* (1/Xrows)
46
47     h0inv = luinv(X'* X :* (1/Xrows))
```

```
48
49      AsyVar = h0inv * v0_hat * h0inv
50      denominator = sqrt(diagonal(AsyVar) * (1/Xrows) )
51      t_stats = betahat :/ denominator
52
53      st_matrix("betahat", betahat)
54      st_matrix("denominator", denominator)
55      st_matrix("tstats", t_stats)
56      st_matrix("s2",s2)
57      st_matrix("v0",v0_hat)
58      st_matrix("h0inv",h0inv)
59      st_matrix("asyvar",AsyVar)
60      st_matrix("countcols",Xcols)
61      st_matrix("countrows",Xrows)
62
63  end
64
65  **4a: now use choleksy inverse
66
67
68  ** Now compute the pvalues and confidence intervals
69  g cols = countcols[1,1]
70  local cols = cols
71
72  g ones = 1
73  egen obs = sum(ones)
74  local obs = obs
75
76  g df = `obs' - `cols'
77
78  forv i = 1(1)`cols' {
79
80      local j = `i' - 1
81      g beta_hat_`j' = betahat[`i',1]
82      g t_stat_beta_`j' = tstats[`i',1]
83      g pval_beta_`j' =  2*ttail(df,t_stat_beta_`j')
84      g lb_beta_`j' = beta_hat_`j' + invttail(df,`alpha'+(1-`alpha')/
    2)*denominator[`i',1]
85      g ub_beta_`j' =  beta_hat_`j' + invttail(df,(1-`alpha')/2)*
    denominator[`i',1]
86
87  }
88
89
90  **4a: now use choleksy inverse
91
92  mata: mata clear
93  capture eret clear
```

```
 94
 95  mata:
 96
 97      X = st_data(.,("intercept", "x_1", "x_2"))
 98      y = st_data(.,("y"))
 99      Xrows = rows(X)
100      Xcols = cols(X)
101
102      betahat_chol = cholinv(X' * X) * (X' * y)
103
104      epsilonhat_chol = y - X * betahat_chol
105      s2_chol = epsilonhat_chol' * epsilonhat_chol :* (1/(Xrows-Xcols
     ))
106      v0_hat_chol = s2_chol * X' * X :* (1/Xrows)
107
108      h0inv_chol = cholinv(X'* X :* (1/Xrows))
109
110      AsyVar_chol = h0inv_chol * v0_hat_chol * h0inv_chol
111      denominator_chol = sqrt(diagonal(AsyVar_chol) * (1/Xrows) )
112      t_stats_chol = betahat_chol :/ denominator_chol
113
114      st_matrix("betahatchol", betahat_chol)
115      st_matrix("denominatorchol", denominator_chol)
116      st_matrix("tstatschol", t_stats_chol)
117      st_matrix("s2chol",s2_chol)
118      st_matrix("v0chol",v0_hat_chol)
119      st_matrix("h0invchol",h0inv_chol)
120      st_matrix("asyvarchol",AsyVar_chol)
121
122  end
123
124  **4a: now use choleksy inverse
125
126
127  ** Now compute the pvalues and confidence intervals
128
129  forv i = 1(1)`cols' {
130
131      local j = `i' - 1
132      g beta_hat_chol_`j' = betahatchol[`i',1]
133      g t_stat_beta_chol_`j' = tstatschol[`i',1]
134      g pval_beta_chol_`j' =  2*ttail(df,t_stat_beta_chol_`j')
135      g lb_beta_chol_`j' = beta_hat_chol_`j' + invttail(df,`alpha'+(1
     -`alpha')/2)*denominatorchol[`i',1]
136      g ub_beta_chol_`j' =  beta_hat_chol_`j' + invttail(df,(1-
     `alpha')/2)*denominatorchol[`i',1]
137
138  }
```

```
**Compare Symmetric and Cholesky Inverse

local compare = "lb_beta ub_beta"

di "cols is `cols'"

foreach cibound of local compare {

    forv i = 1(1)`cols' {
    local k = `i' - 1

        g diff_`cibound'_`k' = round(`cibound'_`k',.0000000001) - round(`cibound'_chol_`k',.0000000001)
        su diff_`cibound'_`k'

    }

}


/***************
/* PS 2 Q 5 */
***************/

**Q2: 5a

clear

import delim "$data/LaLonde_1986", delim(",")

local alpha = .95

g educ2 = educ * educ
g black_earn74 = black * earn74
g intercept = 1

mata: mata clear
capture eret clear

mata:

    X = st_data(.,("intercept", "treat", "black", "age", "educ", "educ2", "earn74", "black_earn74", "u74", "u75"))
    y = st_data(.,("earn78"))
    Xrows = rows(X)
    Xcols = cols(X)

    betahat = luinv(X' * X) * (X' * y)
```

```
186
187       epsilonhat = y - X * betahat
188       s2 = epsilonhat' * epsilonhat :* (1/(Xrows-Xcols))
189       v0_hat = s2 * X' * X :* (1/Xrows)
190
191       h0inv = luinv(X'* X :* (1/Xrows))
192
193       AsyVar = h0inv * v0_hat * h0inv
194       denominator = sqrt(diagonal(AsyVar) * (1/Xrows) )
195       t_stats = betahat :/ denominator
196
197       st_matrix("betahat", betahat)
198       st_matrix("denominator", denominator)
199       st_matrix("tstats", t_stats)
200       st_matrix("s2",s2)
201       st_matrix("v0",v0_hat)
202       st_matrix("h0inv",h0inv)
203       st_matrix("asyvar",AsyVar)
204       st_matrix("countcols",Xcols)
205       st_matrix("countrows",Xrows)
206
207   end
208
209   g cols = countcols[1,1]
210   local cols = cols
211
212   g ones = 1
213   egen obs = sum(ones)
214   local obs = obs
215
216   g df = `obs' - `cols'
217
218   forv i = 1(1)`cols' {
219
220       *local j = `i' - 1
221       local j = `i'
222       g beta_hat_`j' = betahat[`i',1]
223       g t_stat_beta_`j' = tstats[`i',1]
224       g pval_beta_`j' =  2*ttail(df,abs(t_stat_beta_`j'))
225       g lb_beta_`j' = beta_hat_`j' + invttail(df,`alpha'+(1-`alpha')/
      2)*denominator[`i',1]
226       g ub_beta_`j' =  beta_hat_`j' + invttail(df,(1-`alpha')/2)*
      denominator[`i',1]
227       g se_beta_`j' = denominator[`i',1]
228
229   }
230
231   *local colsminus = `cols' - 1
```

```
232
233 forv i = 1(1)`cols' {
234     local j = `i'
235
236     local betahat = beta_hat_`j'
237     local pval = pval_beta_`j'
238     local lb = lb_beta_`j'
239     local ub = ub_beta_`j'
240     local tstat = t_stat_beta_`j'
241
242     di "beta is `betahat'; pval is `pval', lb is `lb', ub is `ub'"
243 }
244
245
246 local indepvars "intercept treat black age educ educ2 earn74
    black_earn74 u74 u75"
247
248 **Q2: 5b
249
250 g independentvars = "`indepvars'"
251
252 g independentvars_split = independentvars
253
254 split independentvars_split, g(indep)
255
256 reg earn78 `indepvars', nocons
257 g df_reg = e(df_r)
258
259 forv i = 1(1)`cols' {
260
261     local j = `i'
262     local varrel = indep`i'
263     g beta_hat_reg_`j' = _b[`varrel']
264     g se_beta_reg_`j' = _se[`varrel']
265     g t_stat_beta_reg_`j' = beta_hat_reg_`j' / se_beta_reg_`j'
266     g pval_reg_`j' =  2*ttail(df_reg,abs(t_stat_beta_reg_`j'))
267     g lb_beta_reg_`j' = beta_hat_reg_`j' + invttail(df,`alpha'+(1-
    `alpha')/2)*se_beta_reg_`j'
268     g ub_beta_reg_`j' =  beta_hat_reg_`j' + invttail(df,(1-`alpha'
    )/2)*se_beta_reg_`j'
269
270 }
271
272 g obscounter = [_n]
273
274
275 g var1 = indep1 if obscounter == 1
276 g beta_hat_export = .
```

```
277 g beta_hat_reg_export = .
278 g se_export = .
279 g se_reg_export = .
280 g t_stat_export = .
281 g t_stat_reg_export = .
282 g pval_export = .
283 g pval_reg_export = .
284 g lb_export = .
285 g lb_reg_export = .
286 g ub_export = .
287 g ub_reg_export = .
288
289 forv j = 1(1)`cols' {
290
291     *local j = `i' - 1
292     replace var = indep`j' if obscounter == `j'
293     replace beta_hat_export = beta_hat_`j' if obscounter == `j'
294     replace beta_hat_reg_export = beta_hat_reg_`j' if obscounter ==
     `j'
295     replace se_export = se_beta_`j' if obscounter == `j'
296     replace se_reg_export = se_beta_reg_`j' if obscounter == `j'
297     replace t_stat_export = t_stat_beta_`j' if obscounter == `j'
298     replace t_stat_reg_export = t_stat_beta_reg_`j' if obscounter
    == `j'
299     replace pval_export = pval_beta_`j' if obscounter == `j'
300     replace pval_reg_export = pval_reg_`j' if obscounter == `j'
301     replace lb_export = lb_beta_`j' if obscounter == `j'
302     replace lb_reg_export = lb_beta_reg_`j' if obscounter == `j'
303     replace ub_export = ub_beta_`j' if obscounter == `j'
304     replace ub_reg_export = ub_beta_reg_`j' if obscounter == `j'
305
306 }
307
308 keep var *export
309
310 export excel "$out/STATA_PS1_Q2_5a_5b.xlsx", firstrow(variables)
    replace
311
312
```