



```

        name:    main
        log:     C:\Users\Conor\Documents\Conor\Grad School\TA Work\Econ 103 - Econometric
> s\STATA Work\Week 4\wk4_section_log.smcl
        log type:    smcl
        opened on:    5 Feb 2018, 11:55:49

1 . qui log using wk4_section_simple, name(simple) replace

2 .
3 . // Demonstration STATA code for week 4
4 . // Principles of Econometrics 4th Edition
5 . // Covered Problems: 4.15
6 .
7 . set more off

8 . clear all

9 . use cps4.dta, clear

10.
11. ////////////////////////////////////////////
> //////////////////////////////////////////// Question 4.15 ////////////////////////////////////////////
> ////////////////////////////////////////////
>
12. *****
13. *Setup: Does the return to education differ by race and gender? In this exercise,
14. * we will look at the following subsamples ("partitions") of the data:
15. * (i) all males (ii) all females (iii) all whites (iv) all blacks
16. * (v) white males (vi) white females (vii) black males (viii) black females
17. *
18. * Parts (A) - (E)
19. *****
20.
21. // Most of the work for this problem is repetitive. For that reason, I will
22. // use a loop to generate the required results, and then report the output and
23. // include some discussion. The "simple" log file will not record the commands
24. // or the output from the loop, but will show results that I store. The "main"
25. // log file will continue recording the full do file.
26.
27. qui log off simple

28. // Note that at the beginning of the file, we had initiated 2 log file.
29. // log off [name] tells STATA to stop recording using log file [name] where
30. // name was set in the name option, and is NOT the name of the .smcl file
31. // Since we have 2 log files running, wk4_section_log (name = main) will
32. // continue recording while wk4_section_simple (name = simple) will not
33. // document running the loop
34.
35.
36. //////////////////////////////////////////// Warning: Advanced STATA Usage Below ////////////////////////////////////////////
> //////////////////////////////////////////// Exercise in How to Use Loops in STATA ////////////////////////////////////////////
>
37. // Set variable for significance level at 5% significance
38. // Used for hypothesis test in Part E
39. scalar alpha = 0.05

40.
41. // Set variable for null for beta where beta = 0.1
42. // Used for hypothesis test in Part E

```

```

43. scalar nullBeta = 0.1
44.
45. // Generate log wage
46. gen ln_wage = log(wage)
47.
48. // Use this loop to do all the exercises for the 8 partitions of the data:
49.
50. // Create a column vector of 0s for the values we want to store.
51. // As we go through the loop, we will store values of interest in
52. // the rows of the matrix
53. //
54. // Also, for each matrix, assign a column name to make it clear what value
55. // is being stored
56. //
57. // command: matrix NAME = J(n,k,a) creates an n-by-k matrix, with all entries
58. // set to the value a. A can be a number of ., where . means a missing value
59. matrix store_values = J(8,11,0)
60. matrix colnames store_values = "NumObs" "Wage_mean" "Wage_sd" "Wage_cv" ///
    >                                     "Beta" "StdError" "R
    > Sqr" "RMSE" ///
    >
    >                                     "T:Beta=0.1" "TC_95%
    > _2side" "Reject_Null"
61.
62. local starRow "*****"
    > "*****"
63.
64. gen condition = 0
65.
66. forvalues partition = 1/8 {
    2. // at the beginning of each iteration, set condition = 0
    replace condition = 0
    3.
    // What the loop does:
    // (1) depending on the value of the loop (1 through 8), set the value of
    // "condition" = 1 for the observations we want to isolate
    // (2) add a new "element" to the local setRowNames describing which set of
    // observations is being used. Later, we will use this local to assign
    // names to the rows of the storage matrices.
    // (3) Display which loop number in the loop we've reached, and which
    // observations are being used. Useful for troubleshooting as well as
    // reading the output in the log file wk4_section_log
    // (4) For each partition, run the commands associated with each question:
    // (a) summary statistics of wage
    // (b) coefficient of variation for wage
    // (c) run OLS for ln_wage = b1 + b2*educ, interpret coefficient
    // (d) evaluate model_fit (we use R2 and RMSE)
    // (e) test beta2 = 0.1 at 5% confidence, 2-sided test
    83. if `partition' == 1 { // all men
        4. replace condition = 1 if female == 0
        5. local setRowNames "men"
        6. local conditionName = "all men"
        7. }
        8. else if `partition' == 2 { // all women
        9. replace condition = 1 if female == 1
        10. local setRowNames `setRowNames' "women"
        11. local conditionName = "all women"
        12. }
        13. else if `partition' == 3 { // all whites
        14. replace condition = 1 if white == 1
        15. local setRowNames `setRowNames' "white"
        16. local conditionName = "all whites"
        17. }
        18. else if `partition' == 4 { // all blacks
        19. replace condition = 1 if black == 1
        20. local setRowNames `setRowNames' "black"
        21. local conditionName = "all blacks"

```

```

22.     }
23.     else if `partition' == 5 { // white men
24.         replace condition = 1 if white == 1 & female == 0
25.         local setRowNames `setRowNames' "whtMen"
26.         local conditionName = "white men"
27.     }
28.     else if `partition' == 6 { // white women
29.         replace condition = 1 if white == 1 & female == 1
30.         local setRowNames `setRowNames' "whtWomen"
31.         local conditionName = "white women"
32.     }
33.     else if `partition' == 7 { // black men
34.         replace condition = 1 if black == 1 & female == 0
35.         local setRowNames `setRowNames' "blkMen"
36.         local conditionName "black men"
37.     }
38.     else if `partition' == 8 { // black women
39.         replace condition = 1 if black == 1 & female == 1
40.         local setRowNames `setRowNames' "blkWomen"
41.         local conditionName = "black women"
42.     } // if partition ... else if partition ...
43.
84.     // Report where in the loop we are to results window
85.     disp "`starRow'"
44.     disp "Partition: `partition' - Condition: `conditionName'"
45.     disp "`starRow'"
46.
86.     // Part A: Summary statistics for WAGE, by partition
87.     sum wage if condition == 1
47.     matrix store_values[`partition',1] = r(N)
48.     matrix store_values[`partition',2] = r(mean)
49.     matrix store_values[`partition',3] = r(sd)
50.
88.     // Part B: Coefficient of Variation for WAGE, by partition
89.     matrix store_values[`partition',4] = 100*r(sd)/r(mean)
51.
90.     // Part C: Run regression ln(WAGE) = beta1 + beta2*EDUC + e
91.     reg ln_wage educ if condition == 1
52.
92.     // Store beta and standard error for regression
93.     matrix store_values[`partition',5] = _b[educ]
53.     matrix store_values[`partition',6] = _se[educ]
54.
94.     // Part D: Does model fit equally well for each partition?
95.     // --> To evaluate this, we look at the R2 and RMSE
96.     matrix store_values[`partition',7] = e(r2)
55.     matrix store_values[`partition',8] = e(rmse)^2
56.
97.     // Part E: Test the null hypothesis that the rate of return to education
98.     // is 10% against the alternative that it is not, using a two-sided test at
99.     // the 5% level of significance.
100.
101.     // E-1: Calculate t-statistic for 10% return is equivalent to _b[educ] = 0.1
102.     // set null beta = 0.1 earlier
103.     matrix store_values[`partition',9] = (_b[educ]-`nullBeta')/_se[educ]
57.
104.     // E-2: Calculate critical value for 2-sided t-test at significance alpha
105.     // alpha = 0.05 (set above)
106.     // Since the degrees of freedom will vary from one regression to another, we
107.     // need to calculate a different critical value for each. Given the large

```

```

108      // overall size of the sample, the differences are unlikely to be
109      // particularly large, however.
110      matrix store_values['partition',10] = invttail(e(df_r), alpha/2)
58.
111      // E-3: Decide test
112      // Store a logical (0 or 1) value, based on whether we reject the null
113      // for a given regression. Since this is a 2-sided test, we compare the
114      // absolute value to the (positive) critical value
115      matrix store_values['partition',11] = abs(store_values['partition',9])>store
> _values['partition',10]
59.
116
117 } // end loop: forvalues partition
(0 real changes made)
(2,395 real changes made)

```

```

*****
Partition: 1 - Condition: all men
*****

```

Variable	Obs	Mean	Std. Dev.	Min	Max
wage	2,395	22.25795	13.47253	1	173
Source	SS	df	MS	Number of obs	= 2,395
Model	159.850456	1	159.850456	F(1, 2393)	= 614.54
Residual	622.449238	2,393	.260112511	Prob > F	= 0.0000
Total	782.299694	2,394	.326775144	R-squared	= 0.2043
				Adj R-squared	= 0.2040
				Root MSE	= .51001

ln_wage	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
educ	.0883697	.0035647	24.79	0.000	.0813794 .0953599
_cons	1.732617	.0498571	34.75	0.000	1.63485 1.830385

```

(2,395 real changes made)
(2,443 real changes made)
*****
Partition: 2 - Condition: all women
*****

```

Variable	Obs	Mean	Std. Dev.	Min	Max
wage	2,443	18.0538	11.1568	1.14	96.17
Source	SS	df	MS	Number of obs	= 2,443
Model	175.94651	1	175.94651	F(1, 2441)	= 734.02
Residual	585.114023	2,441	.23970259	Prob > F	= 0.0000
Total	761.060533	2,442	.3116546	R-squared	= 0.2312
				Adj R-squared	= 0.2309
				Root MSE	= .48959

ln_wage	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
educ	.1063988	.0039272	27.09	0.000	.0986978 .1140998
_cons	1.242679	.0559434	22.21	0.000	1.132977 1.35238

```

(2,443 real changes made)
(4,116 real changes made)
*****
Partition: 3 - Condition: all whites
*****

```

Variable	Obs	Mean	Std. Dev.	Min	Max
wage	4,116	20.48479	12.63815	1.14	173

Source	SS	df	MS	Number of obs	=	4,116
Model	260.04068	1	260.04068	F(1, 4114)	=	979.75
Residual	1091.91738	4,114	.265415017	Prob > F	=	0.0000
				R-squared	=	0.1923
				Adj R-squared	=	0.1921
Total	1351.95806	4,115	.328543879	Root MSE	=	.51518

ln_wage	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
educ	.0910544	.002909	31.30	0.000	.0853512	.0967577
_cons	1.59244	.0411475	38.70	0.000	1.511769	1.673112

(4,116 real changes made)

(493 real changes made)

Partition: 4 - Condition: all blacks

Variable	Obs	Mean	Std. Dev.	Min	Max
wage	493	16.44389	10.13587	1	72.13

Source	SS	df	MS	Number of obs	=	493
Model	27.1848402	1	27.1848402	F(1, 491)	=	125.25
Residual	106.565497	491	.217037672	Prob > F	=	0.0000
				R-squared	=	0.2033
				Adj R-squared	=	0.2016
Total	133.750337	492	.271850279	Root MSE	=	.46587

ln_wage	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
educ	.1051817	.0093982	11.19	0.000	.0867161	.1236473
_cons	1.245588	.127762	9.75	0.000	.99456	1.496615

(493 real changes made)

(2,065 real changes made)

Partition: 5 - Condition: white men

Variable	Obs	Mean	Std. Dev.	Min	Max
wage	2,065	22.83416	13.67063	1.5	173

Source	SS	df	MS	Number of obs	=	2,065
Model	135.920445	1	135.920445	F(1, 2063)	=	534.92
Residual	524.197917	2,063	.254094967	Prob > F	=	0.0000
				R-squared	=	0.2059
				Adj R-squared	=	0.2055
Total	660.118362	2,064	.319824788	Root MSE	=	.50408

ln_wage	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
educ	.0861452	.0037247	23.13	0.000	.0788407	.0934497
_cons	1.7909	.0521835	34.32	0.000	1.688562	1.893238

(2,065 real changes made)

(2,051 real changes made)

Partition: 6 - Condition: white women

Variable	Obs	Mean	Std. Dev.	Min	Max
wage	2,051	18.11937	11.01337	1.14	96.17

Source	SS	df	MS	Number of obs	=	2,051
Model	144.303713	1	144.303713	F(1, 2049)	=	599.62
Residual	493.10938	2,049	.240658555	Prob > F	=	0.0000
				R-squared	=	0.2264
				Adj R-squared	=	0.2260
Total	637.413093	2,050	.310933216	Root MSE	=	.49057

ln_wage	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
educ	.1057232	.0043175	24.49	0.000	.097256	.1141903
_cons	1.254094	.0616503	20.34	0.000	1.13319	1.374997

(2,051 real changes made)

(214 real changes made)

Partition: 7 - Condition: black men

Variable	Obs	Mean	Std. Dev.	Min	Max
wage	214	16.21332	9.492562	1	72.13

Source	SS	df	MS	Number of obs	=	214
Model	5.57111511	1	5.57111511	F(1, 212)	=	23.12
Residual	51.0759938	212	.240924499	Prob > F	=	0.0000
				R-squared	=	0.0983
				Adj R-squared	=	0.0941
Total	56.6471089	213	.265948868	Root MSE	=	.49084

ln_wage	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
educ	.0761759	.0158412	4.81	0.000	.0449495	.1074022
_cons	1.652099	.2104773	7.85	0.000	1.237203	2.066996

(214 real changes made)

(279 real changes made)

Partition: 8 - Condition: black women

Variable	Obs	Mean	Std. Dev.	Min	Max
wage	279	16.62075	10.61639	3.75	72.13

Source	SS	df	MS	Number of obs	=	279
Model	23.314309	1	23.314309	F(1, 277)	=	120.08
Residual	53.7803451	277	.19415287	Prob > F	=	0.0000
				R-squared	=	0.3024
				Adj R-squared	=	0.2999
Total	77.0946541	278	.2773189	Root MSE	=	.44063

ln_wage	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
educ	.126164	.0115132	10.96	0.000	.1034995	.1488285
_cons	.9395272	.1591768	5.90	0.000	.6261772	1.252877

```

118
119 // Assign row names for all the matrices we created earlier
120 // have STATA report the list of all matrices in a local
121 local allMatrices : all matrices

122
123 // Use foreach ... to loop over the matrices
124 // Assign each matrix the set of rownames that we created during the loop above
125 foreach x of local allMatrices {
126     2.         matrix rownames `x' = `setRowNames'
127     3. }

128
129 // Begin recording with the "simple" log file
130 qui log on simple

131
132 *****
133 *4.15 Part A: For each sample partition, obtain the summary statistics of WAGE.
134 *****
135 // Extract the columns we want from the store_values matrix using
136 // number indexing
137 matrix partA_values = store_values[1...,1..3] // rows 1 and on, columns 1-3

138
139 // Have stata report the contents of a matrix using:
140 // matrix list matName
141 matrix list partA_values

    partA_values[8,3]
           NumObs  Wage_mean  Wage_sd
    men         2395   22.257954   13.472531
    women        2443   18.053799   11.156805
    white        4116   20.484786   12.638152
    black         493   16.443895   10.135868
    whtMen        2065   22.834165   13.670627
    whtWomen      2051   18.119371   11.013366
    blkMen         214   16.213318    9.4925622
    blkWomen      279   16.620753   10.616392

142
143 *****
144 *4.15 Part B: A variable's "coefficient of variation" (CV) is 100 times the ratio
145 * of its sample standard deviation to its sample mean For a variable y, it is:
146 *
147 *  $CV = 100 * se(y) / \bar{y}$ 
148 *
149 * It is a measure of variation that takes into account the size of the variable.
150 * What is the coefficient of variation for WAGE within each sample partition?
151 *****
152
153 // use column name Wage_cv as index to extract the appropriate column of store_value
154 > s
155 // store_values[1..., "Wage_cv"] is rows 1 and on, column with name Wage_cv
156 matrix cvByPartition = store_values[1..., "Wage_cv"]

```

```
156 matrix list cvByPartition
```

```
cvByPartition[8,1]
      Wage_cv
  men 60.529064
  women 61.797546
  white 61.695306
  black 61.639098
  whtMen 59.869178
  whtWomen 60.782278
  blkMen 58.547932
  blkWomen 63.874313
```

```
157
158 *****
159 *4.15 Part C: For each sample partition, estimate the log-linear model:
160 *
161 * ln(WAGE) = beta1 + beta2*educ + e
162 *
163 * What is the approximate percentage return to another year of education for
164 * each group?
165 *****
166
167 // Can join matrices together by having one matrix to the left and one matrix
168 // to the right by using [mat1, mat2]. Note, this only works if mat1 and mat2
169 // have the same number of rows
170 matrix partC_values = [store_values[1...,"Beta"], store_values[1...,"Beta"]*100]

171 // Reset names of columns in partC_values - useful for display
172 matrix colnames partC_values = "Beta" "BetaX100"
```

```
173 matrix list partC_values
```

```
partC_values[8,2]
      Beta  BetaX100
  men .08836967 8.8369667
  women .10639878 10.639878
  white .09105444 9.1054445
  black .1051817 10.51817
  whtMen .08614524 8.6145236
  whtWomen .10572316 10.572316
  blkMen .07617588 7.617588
  blkWomen .12616401 12.616401
```

```
174
175 // Since we have a log-linear regression, i.e. ln(y) = beta1+beta2*x+e
176 // the interpretation of the b2 coefficient is that a 1 unit increase in x leads
177 // to a b2*100 percent increase in y
178
179 *****
180 *4.15 Part D: Does the model fit the data equally well for each sample
181 * partition?
182 *****
183
184 // store_values[1...,"R2".."RMSE"] = rows 1 and on, columns from R2 to RMSE
185 // Recall that R2 = 1 - (SSE/TSS) while RMSE = sqrt((1/df_r)*SSE)
186 matrix partD_values = store_values[1...,"RSqr".."RMSE"]
```

```
187 matrix list partD_values
```

```
partD_values[8,2]
      RSqr  RMSE
  men .20433404 .26011251
  women .23118596 .23970259
  white .19234375 .26541502
  black .20325063 .21703767
  whtMen .20590314 .25409497
  whtWomen .22638963 .24065856
  blkMen .09834774 .2409245
  blkWomen .30241149 .19415287
```



```

188
189 /* Discussion:
190 >
191 > The data fit in a broadly similar range for all (R^2 of 19-23) for all the models,
192 > except when we use the two smallest groups: black-by-gender. The model fits
193 > very poorly for black men but does its best for black women, which netted out
194 > to an average performance for the black partition as a whole. Notably, the model
195 > fits somewhat better for white men and white women separately than it does for
196 > whites as a whole.
197 >
198 > */
199
200 *****
201 *4.15 Part E: For each sample partition, test the null hypothesis that the rate
202 * of return to education is 10% against the alternative that it is not, using a
203 * two-tail test at the 5% level of significance.
204 *****
205
206 // One way to get columns 5-6 and 9-11
207 matrix partE_values = [store_values[1...,5..6], store_values[1...,9..11]]
208
209 matrix list partE_values
210
211 partE_values[8,5]
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999

```

	Beta	StdError	Beta=0.1	TC_95% 2side	Reject_Null
men	.08836967	.00356473	-3.2626106	1.9609558	1
women	.10639878	.0039272	1.6293497	1.9609363	0
white	.09105444	.002909	-3.075134	1.9605408	1
black	.1051817	.00939819	.551351	1.9648072	0
whtMen	.08614524	.00372466	-3.7197368	1.9611146	1
whtWomen	.10572316	.0043175	1.3255746	1.9611224	0
blkMen	.07617588	.01584116	-1.5039381	1.971217	0
blkWomen	.12616401	.0115132	2.2725228	1.968565	1

```

200
201 /* Discussion:
202 >
203 > The table shows the betas for education, along with the standard error, the t-stat
204 > that the beta shown is equal to 0.1 (10% in decimal terms), the critical value,
205 > and a column of 1s and 0s where 1 indicates a rejection of the null and 0 a
206 > failure to reject. We can reject the null of beta = 0.1 to the downside for the
207 > partitions men, whites, and white men; to the upside we can reject the null of
208 > beta = 0.1 for black women. For the other four partition (women, black, white women
209 > and black men) we fail to reject the null.
210 >
211 > A few comments: first note that the critical values change slightly from one
212 > regression to the next. This reflects the changes in the sample size. Notice also
213 > that the standard errors for black, black men, and black women are much larger,
214 > again reflecting the small sample sizes for these groups. This leads to differing
215 > conclusions - for example, the point estimate for black men is lower than the
216 > point estimate for white men but we fail to reject the null for black men because
217 > the beta is so imprecisely (large standard error). Comparing black men to black
218 > women, however, we are able to reject the null because the fit of the model was
219 > better for black women than for black men (though there are more black women
220 > observations, the gap is small compared to black men vs. white men in the sample).
221 >
222 > */
223
224 //Comment: How to think about partitioned regressions

```

```

204 //
205 // One way to think of a regression partition is that it is equivalent to
206 // interacting ALL the right-hand side variables with an indicator variable.
207 // Below, we show how we get identical point estimates for the betas for male
208 // vs. female when we partition and when we interact with female.
209 //
210 // Note that, while the betas are the same the standard errors are not. The
211 // standard errors differ because (1) the sigma_hat^2 for the interaction
212 // regression is a weighted average of the sigma_hat^2 estimates from the separate
213 // regressions and (2) the interaction regression adjusts the standard errors
214 // using information about the relationship between the various RHS terms, which
215 // differs from what OLS uses in the partitioned regressions.
216 //
217 // put a constant in the regression by hand, rather than let STATA do it for us
218 gen onesCol = 1

```

```

219
220 reg ln_wage onesCol educ if female == 0, noconstant

```

Source	SS	df	MS	Number of obs	=	2,395
Model	20879.1724	2	10439.5862	F(2, 2393)	=	40134.89
Residual	622.449238	2,393	.260112511	Prob > F	=	0.0000
Total	21501.6217	2,395	8.9777126	R-squared	=	0.9711
				Adj R-squared	=	0.9710
				Root MSE	=	.51001

ln_wage	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
onesCol	1.732617	.0498571	34.75	0.000	1.63485	1.830385
educ	.0883697	.0035647	24.79	0.000	.0813794	.0953599

```

221 reg ln_wage onesCol educ if female == 1, noconstant

```

Source	SS	df	MS	Number of obs	=	2,443
Model	18442.0587	2	9221.02933	F(2, 2441)	=	38468.63
Residual	585.114023	2,441	.23970259	Prob > F	=	0.0000
Total	19027.1727	2,443	7.78844564	R-squared	=	0.9692
				Adj R-squared	=	0.9692
				Root MSE	=	.48959

ln_wage	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
onesCol	1.242679	.0559434	22.21	0.000	1.132977	1.35238
educ	.1063988	.0039272	27.09	0.000	.0986978	.1140998

```

222 reg ln_wage i.female#c.onesCol i.female#c.educ, noconstant

```

Source	SS	df	MS	Number of obs	=	4,838
Model	39321.2311	4	9830.30778	F(4, 4834)	=	39351.73
Residual	1207.56326	4,834	.249806219	Prob > F	=	0.0000
Total	40528.7944	4,838	8.37717949	R-squared	=	0.9702
				Adj R-squared	=	0.9702
				Root MSE	=	.49981

ln_wage	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
female#c.onesCol						
0	1.732617	.0488593	35.46	0.000	1.636831	1.828404
1	1.242679	.0571103	21.76	0.000	1.130716	1.354641
female#c.educ						
0	.0883697	.0034934	25.30	0.000	.081521	.0952183
1	.1063988	.0040091	26.54	0.000	.0985391	.1142585

223

224

225 //Convert log file (smcl) to pdf