

Stata Code and Result - Chapter #3

March 16, 2019

Question 6

```
.
. // download data from: http://hdl.handle.net/10079/6hdr852
. // copy and paste the url to your web browser
.
.
. import delim "Clingingsmith_et_al_QJE_2009dta.csv",clear
(8 vars, 958 obs)
. set seed 1234567
.
. rename success D
. rename views Y
.
.
. //findit tsrtest
. //package name: st0158.pkg install
.
. cap program drop ate
. program define ate, rclass
1.     args Y D
2.     sum `Y' if `D'==1, meanonly
3.     local Y_treat=r(mean)
4.     sum `Y' if `D'==0, meanonly
5.     local Y_con=r(mean)
6.     return scalar ate_avg = `Y_treat'-`Y_con'
7. end
.
. // ssc install tsrtest
. tsrtest D r(ate_avg) using 3_6_resam.dta, overwrite: ate Y D
Two-sample randomization test for theta=r(ate_avg) of ate Y D by D
Combinations:      8.4503047638e+285 = (958 choose 448)
Assuming null=0
Observed theta: .4748
Minimum time needed for exact test (h:m:s):  2.6e+278:00:00
Reverting to Monte Carlo simulation.
Mode: simulation (10000 repetitions)
progress: |.....|
p=0.00190 [one-tailed test of Ho:  theta(D==0)<=theta(D==1)]
p=0.99830 [one-tailed test of Ho:  theta(D==0)>=theta(D==1)]
p=0.00360 [two-tailed test of Ho:  theta(D==0)==theta(D==1)]
Saving log file to 3_6_resam.dta...done.
.
.
. preserve
. use "3_6_resam.dta", clear
. global ate = theta[1]
. di $ate
.4748337
. drop if _n==1
(1 observation deleted)
. count if theta >= $ate
19
. scalar p_onesided = r(N)/_N
. count if abs(theta) >= $ate
```

```

36
. scalar p_twosided = r(N)/_N
. di "p.value.onesided = "p_onesided
p.value.onesided = .0019
. di "p.value.twosided = "p_twosided
p.value.twosided = .0036
. restore
.
.

```

Question 7

```

.
. clear
. set seed 1234567
. set obs 10
number of observations (_N) was 0, now 10
.
. input D Y
           D          Y
1. 0 1
2. 0 0
3. 0 0
4. 0 4
5. 0 3
6. 1 2
7. 1 11
8. 1 14
9. 1 0
10. 1 3
.
.
. gen Y_star= Y+D*(-7)
.
. cap program drop ate
. program define ate, rclass
1.     args Y D
2.     sum `Y' if `D'==1, meanonly
3.     local Y_treat=r(mean)
4.     sum `Y' if `D'==0, meanonly
5.     local Y_con=r(mean)
6.     return scalar ate_avg = `Y_treat'-`Y_con'
7. end
.
. // findit tsrtest (to install the package)
. tsrtest D r(ate_avg): ate Y_star D
Two-sample randomization test for theta=r(ate_avg) of ate Y_star D by D
Combinations: 252 = (10 choose 5)
Assuming null=0
Observed theta: -2.6
Minimum time needed for exact test (h:m:s): 0:00:00
Mode: exact
progress: |.....|
p=0.83730 [one-tailed test of Ho: theta(D==0)<=theta(D==1)]
p=0.20635 [one-tailed test of Ho: theta(D==0)>=theta(D==1)]
p=0.41270 [two-tailed test of Ho: theta(D==0)==theta(D==1)]
.
. // ate
. di r(obsvStat)
-2.6
.
. // p.value.onesided
. di r(lowertail)
.20634921
.
.

```

Question 8

part(a)

```
. // download data from : http://hdl.handle.net/10079/s1rn910
. // copy and paste the url to your web browser
. use "Titunik_WorkingPaper_2010.csv.dta",clear

.
. set seed 1234567
.
.      rename term2year D
.      rename bills_introduced Y
.      rename texas0_arkansas1 block
.
.      qui tabstat Y if block ==0, by(D) stat(mean) save
.      scalar ate_texas = el(r(Stat2),1,1) - el(r(Stat1),1,1)
.
.
.      qui tabstat Y if block ==1, by(D) stat(mean) save
.      scalar ate_ark = el(r(Stat2),1,1) - el(r(Stat1),1,1)
.
.      di "ate_texas=%18.5f ate_texas
ate_texas=          -16.74167
.      di "ate_arkansas=%18.5f ate_ark
ate_arkansas=        -10.09477
```

part(b)

```
.
.
. qui tabstat Y if block ==0, by(D) stat(v n) save
.
. scalar se_texas = sqrt(el(r(Stat2),1,1)/el(r(Stat2),2,1) + ///
>                      el(r(Stat1),1,1)/el(r(Stat1),2,1))
.
.
. qui tabstat Y if block ==1, by(D) stat(v n) save
.
. scalar se_arkansas = sqrt(el(r(Stat2),1,1)/el(r(Stat2),2,1) + ///
>                          el(r(Stat1),1,1)/el(r(Stat1),2,1))
.
.
. di "se_texas=%18.6f se_texas
se_texas=          9.345871
. di "se_arkansas=%18.6f se_arkansas
se_arkansas=        3.395979
.
.
```

part(c)

```
. qui tabstat Y, by(block) stat(n) save
.
. scalar ate_overall = el(r(Stat1),1,1)/_N*ate_texas + ///
>                      el(r(Stat2),1,1)/_N*ate_ark
.
.
. di %18.4f ate_overall
      -13.2168
.
.
. // same as
. // teffects nnmatch (bills_introduced) (term2year), ematch(texas0_arkansas1)
```

part(e)

```
. scalar se_overall = sqrt((e1(r(Stat1),1,1)/_N)^2*se_texas^2 + ///
> (e1(r(Stat2),1,1)/_N)^2*se_arkansas^2)

. di %18.5f se_overall
      4.74478

.
```

part(f)

```
.
. // calculate probs under block assignment
. bysort block: egen probs=mean(D)
.
.
.
. cap program drop ate_block
.
. program define ate_block, rclass
. 1. args Y D probs
. 2. tempvar ipw
. 3. gen `ipw' = .
. 4. // calculate inverse probability weight under block assignment
. replace `ipw' = `D' / `probs' + (1-`D') / (1-`probs')
. 5. qui reg `Y' `D' [iw=`ipw']
. 6. return scalar ate=_b[`D']
. 7. end
```

```
.
. // ssc install ritest (to install ritest package)
.
. //
. ritest D r(ate), strata(block) reps(10000) nodots: ///
> ate_block Y D probs
(66 missing values generated)
(66 real changes made)

      command:  ate_block Y D probs
      _pm_1:    r(ate)
res. var(s):    D
Resampling:    Permuting D
Clust. var(s):  __000000
Clusters:      66
Strata var(s):  block
Strata:        2
```

T	T(obs)	c	n	p=c/n	SE(p)	[95% Conf. Interval]
_pm_1	-13.2168	65	10000	0.0065	0.0008	.00502 .0082774

Note: Confidence interval is with respect to $p=c/n$.
Note: $c = \#\{ |T| \geq |T(\text{obs})| \}$

```
.
.
. // ate
. di e1(r(b),1,1)
-13.216796

.
. // p.value.twosided
. di e1(r(p),1,1)
.0065

.
```

Question 9

part(b)

```
. // download data from : http://hdl.handle.net/10079/1g1jx43
```

```

. // copy and paste the url to your web browser
.
. use "Camerer_JPEsubset_1998.dta.dta", clear
.
. set seed 1234567
.
.
.      rename treatment D
.      rename pair block
.      rename preexperimentbets covs
.
.      // calculate probs under block assignment
.      bysort block: egen probs=mean(D)
.
.
.      // permutation to calculate F stat and one-side P value
.      ritest D e(F), strata(block) reps(10000) right nodots: ///
>      regress D covs

```

Source	SS	df	MS	Number of obs	=	34
Model	.005024372	1	.005024372	F(1, 32)	=	0.02
Residual	8.49497563	32	.265467988	Prob > F	=	0.8914
				R-squared	=	0.0006
				Adj R-squared	=	-0.0306
Total	8.5	33	.257575758	Root MSE	=	.51524

D	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
covs	-.0000386	.0002809	-0.14	0.891	-.0006109 .0005336
_cons	.5137818	.1335793	3.85	0.001	.2416896 .785874


```

command: regress D covs
       _pm_1: e(F)
res. var(s): D
Resampling: Permuting D
Clust. var(s): __000000
Clusters: 34
Strata var(s): block
Strata: 17

```


T	T(obs)	c	n	p=c/n	SE(p)	[95% Conf. Interval]
_pm_1	.0189265	3736	10000	0.3736	0.0048	.3641064 .3831672

Note: Confidence interval is with respect to p=c/n.
Note: c = #{T >= T(obs)}

```

.
.      // p.value
.      di el(r(p),1,1)
.3736

```

part(c)

```

.
.
. rename experimentbets change
.
. tabstat change, by(D) stat(mean) save
Summary for variables: change
      by categories of: D

```

D	mean
0	571.4118
1	461.2353
Total	516.3235


```

. di "ATE ="%180.4f el(r(Stat2),1,1)-el(r(Stat1),1,1)
ATE = -110.1765
.

```

part(d)

```
. bysort block (D): gen pair_diff = change - change[_n+1]
(17 missing values generated)
. mean(pair_diff)
```

Mean estimation		Number of obs	=	17
	Mean	Std. Err.	[95% Conf. Interval]	
pair_diff	110.1765	104.8377	-112.0695	332.4225

```
.
. // the same as
. // teffects nnmatch (experimentbets block) (D)
```

part(e)

```
.
.
. cap program drop ate_block
.
. program define ate_block, rclass
1. args Y D probs
2. tempvar ipw
3. gen `ipw' = .
4. // calculate inverse probability weight under block assignment
. replace `ipw' = `D' / `probs' + (1-`D') / (1-`probs')
5. qui reg `Y' `D' [iweight=`ipw']
6. return scalar ate=_b[`D']
7. end
.
.
. ritest D r(ate), strata(block) reps(10000) nodots: ///
> ate_block change D probs
(34 missing values generated)
(34 real changes made)
      command: ate_block change D probs
      _pm_1:  r(ate)
  res. var(s):  D
  Resampling:  Permuting D
Clust. var(s):  __000000
   Clusters:   34
Strata var(s):  block
   Strata:     17
```

T	T(obs)	c	n	p=c/n	SE(p)	[95% Conf. Interval]	
_pm_1	-110.1765	3170	10000	0.3170	0.0047	.3078845	.3262222

```
Note: Confidence interval is with respect to p=c/n.
Note: c = #{|T| >= |T(obs)|}
.
.
. // ate
. di el(r(b),1,1)
-110.17647
.
. // p.value.twosided
. di el(r(p),1,1)
.317
```