```
*************************************************************************
> **
*
* Practicalities of Running RCTs - Assignment 2
* Zachary Kuloszewski and Jun Wong
*
* Due Nov 3, 2022
*
*************************************************************************
> **

** set options **
version 16
set type double
capture restore
capture log close
macro drop all

clear all
cls
set more off

set graphics off
set scheme plotplainblind

local name zach
if "`name'"=="zach" {
        global main "/Users/zachkuloszewski/Dropbox/My Mac (Zachs-MBP.lan)/Doc
> uments"
        global main $main/GitHub/phd_psets/year2/development/ps2
}
if "`name'"=="jun" {
        global main "/Users/junwong/Dropbox/Second Year/Glennerster - RCT/Assi
> gnments"
}

* import data from Bangladesh Paper
use "$main/data/main.dta", clear

********************** Problem 1.2 - Randomization ************************
> **

set seed 10312022
set sortseed 10312022

* generate random uniform variable
gen rand = runiform()
```

```
gsort rand
gen   rand_id = _n

* calculate number of misfits
local n_mf = mod(_N, 4)

* gen treatment variable (need to treat misfits)
gen treatment = .
forval i=1/4 {
        replace treatment = `i' if rand_id > (`i'-1)*0.25*_N & rand_id <= `i'*
> 0.25*_N
}

* select misfit treatment
replace treatment = floor(runiform()*4) if _n > _N - `n_mf'

drop rand rand_id

********************** Problem 1.3 - Balance Table ************************
> **

* create ever married variable
gen ever_married = (marital_status != 2) * 100 //should engaged to be married
> count?

* create still in school variable
gen in_school = (still_in_school == 1) * 100

* highest class passed variable
gen years_passed = highest_class_passed
replace years_passed = . if inlist(highest_class_passed, 50, 51)
// note: rebecca said that this is fine; one could also replace as zero
// if these don't count as formal education

* find locals
qui count
local n_tot : di %9.0fc `r(N)'

lab var ever_married "Ever Married (\%)"
lab var in_school "Still in-school (\%)"
lab var years_passed "Highest Class Passed"
```

```
foreach var in ever_married in_school years_passed {

        qui summarize `var'
        local `var'_m : di %9.1fc `r(mean)'
        local `var'_sd : di %9.1fc `r(sd)'

        local `var'labs : variable label `var'

        forvalues t=1/4 { // let 1 be control
                qui count if treatment==`t'
                local n_`t' : di %9.0fc `r(N)'

                qui summarize `var' if treatment==`t'
                local `var'_`t'_m : di %9.1fc `r(mean)'
                local `var'_`t'_sd : di %9.1fc `r(sd)'

                if `t'>1 {
                        local `var'_`t'_d = ``var'_`t'_m' - ``var'_1_m '
                        local `var'_`t'_d : di %9.1fc ``var'_`t'_d'
                }
        }
}

* fill balance table
cap file close des
file open des using "$main/output/baseline_balance.tex", write replace
file write des
file write des "\begin{tabular}{lccccccccccc}" _n
file write des "\\" _n
file write des "\toprule" _n
file write des " & \multicolumn{3}{c}{Empowerment} & \multicolumn{3}{c}{Incent
> ive} & \multicolumn{3}{c}{Empow.+Incen.} & \multicolumn{2}{c}{Control} & \mu
> lticolumn{2}{c}{Total} \\" _n
file write des " & \multicolumn{3}{c}{N=`n_2'} & \multicolumn{3}{c}{N=`n_3'} &
>  \multicolumn{3}{c}{N=`n_4'} & \multicolumn{2}{c}{N=`n_1'} & \multicolumn{2}
> {c}{N=`n_tot'} \\" _n
file write des "\cmidrule(lr){2-4} \cmidrule(lr){5-7} \cmidrule(lr){8-10} \cmi
> drule(lr){11-12} \cmidrule(lr){13-14}" _n
file write des " & Mean & SD & Diff & Mean & SD & Diff & Mean & SD & Diff & Me
> an & SD & Mean & SD \\" _n
file write des "\midrule" _n
foreach var of varlist ever_married in_school years_passed  {
        file write des "``var'labs' & ``var'_2_m' & ``var'_2_sd' & ``var'_2_d'
>  & ``var'_3_m' & ``var'_3_sd' & ``var'_3_d' & ``var'_4_m' & ``var'_4_sd' & `
> `var'_4_d' & ``var'_1_m' & ``var'_1_sd'  & ``var'_m' & ``var'_sd' \\" _n
}
file write des "\bottomrule" _n
file write des "\end{tabular} " _n
file close des
```

```
****************** Problem 1.4 - Stratified Randomization ********************
> **

* stratify by unionID with 2:2:1:1 ratio
// (consistent with above with control being t=1 & empowerment t=2)
randtreat, generate(strat_treatment) strata(unionID) misfits(strata) ///
                  unequal(1/3 1/3 1/6 1/6)

replace strat_treatment=strat_treatment + 1

********************** Problem 1.5 - Balance Table ***************************
> **

* find locals
qui count
local n_tot : di %9.0fc `r(N)'

lab var ever_married "Ever Married (\%)"
lab var in_school "Still in-school (\%)"
lab var years_passed "Highest Class Passed"

foreach var in ever_married in_school years_passed {

        qui summarize `var'
        local `var'_m : di %9.1fc `r(mean)'
        local `var'_sd : di %9.1fc `r(sd)'

        local `var'labs : variable label `var'

        forvalues t=1/4 { // let 1 be control
                qui count if strat_treatment==`t'
                local n_`t' : di %9.0fc `r(N)'

                qui summarize `var' if strat_treatment==`t'
                local `var'_`t'_m : di %9.1fc `r(mean)'
                local `var'_`t'_sd : di %9.1fc `r(sd)'

                if `t'>1 {
                        local `var'_`t'_d = ``var'_`t'_m' - ``var'_1_m '
                        local `var'_`t'_d : di %9.1fc ``var'_`t'_d'
                }
        }
}
```

```
* fill balance table
cap file close des
file open des using "$main/output/stratified_balance.tex", write replace
file write des
file write des "\begin{tabular}{lccccccccccc}" _n
file write des "\\" _n
file write des "\toprule" _n
file write des " & \multicolumn{3}{c}{Empowerment} & \multicolumn{3}{c}{Incent
> ive} & \multicolumn{3}{c}{Empow.+Incen.} & \multicolumn{2}{c}{Control} & \mu
> lticolumn{2}{c}{Total} \\" _n
file write des " & \multicolumn{3}{c}{N=`n_2'} & \multicolumn{3}{c}{N=`n_3'} &
>  \multicolumn{3}{c}{N=`n_4'} & \multicolumn{2}{c}{N=`n_1'} & \multicolumn{2}
> {c}{N=`n_tot'} \\" _n
file write des "\cmidrule(lr){2-4} \cmidrule(lr){5-7} \cmidrule(lr){8-10} \cmi
> drule(lr){11-12} \cmidrule(lr){13-14}" _n
file write des " & Mean & SD & Diff & Mean & SD & Diff & Mean & SD & Diff & Me
> an & SD & Mean & SD \\" _n
file write des "\midrule" _n
foreach var of varlist ever_married in_school years_passed  {
        file write des "``var'labs' & ``var'_2_m' & ``var'_2_sd' & ``var'_2_d'
>  & ``var'_3_m' & ``var'_3_sd' & ``var'_3_d' & ``var'_4_m' & ``var'_4_sd' & `
> `var'_4_d' & ``var'_1_m' & ``var'_1_sd'  & ``var'_m' & ``var'_sd' \\" _n
}
file write des "\bottomrule" _n
file write des "\end{tabular}" _n
file close des

********************** Problem 2.2 Testing Survey *************************
> **

* generate in_sample as all unmarried girls
gen in_sample = ever_married==0

* sample five IDs in each treatment arm (presumably from the stratified?)
cap drop rand rand_id
bys strat_treatment: gen rand = runiform() if in_sample==1
gsort strat_treatment rand
bys strat_treatment: gen rand_id = _n
keep if rand_id <= 5
export delimited "$main/output/sampled_ids.csv", replace quote dataf

* summarize results from our survey
import delimited "$main/output/sampled_ids.csv", clear varn(1)
keep memberid strat_treatment
tempfile treat_key
save "`treat_key'", replace
```

```
import delimited "$main/data/Assignment2Q2_WIDE.csv", clear varn(1)
ren id_number memberid
merge 1:1 memberid using "`treat_key'", nogen

gen is_married = (marital_status==2)* 100
replace currently_in_school = currently_in_school * 100

lab var is_married "Married at Endline (%)"
lab var currently_in_school "Enrolled at Endline (%)"

gen treat = "Control" if strat_treatment==1
replace treat = "Empowerment" if strat_treatment==2
replace treat = "Empow.+Incen." if strat_treatment==3
replace treat = "Incentive" if strat_treatment==4
encode treat, gen(streat)

* let's make a cute graph
ciplot is_married currently_in_school, by(treat)
gr export "$main/output/endline_plot.pdf", replace
```