

|  |
| --- |
| Exercise: How to do Random Assignment using Stata |

**Contents**

[Introduction: 1](#_Toc372210114)

[Part 1: Simple Randomization 1](#_Toc372210115)

[Part 2: Stratified Randomization by Language and Gender 2](#_Toc372210116)

[Part 3: Stratified Randomization by Gender, Language and Average Pre-test score 3](#_Toc372210117)

[Part 3: Randomization given Multiple Treatments 4](#_Toc372210118)

# Introduction:

In this example, we will randomly assign schools to treatment and control for our experiment using Stata (version 13). We use baseline data from the Pratham Balsakhi program in Vadodara India. We will learn how to perform simple randomization, stratified randomization, matching using groups of 4 and randomization for multiple treatments. For the purpose of this exercise, we will be using the following dataset:

Dataset:

Balsakhi\_data.dta

Important variables:

schoolid

language

gender

pretest\_mean

# Part 1: Simple Randomization

In this example we will do a simple randomization where we will randomly assign half the schools to treatment and half to control. Follow the steps below:

1. Create a random number. While we want randomization to be *random*, we also want it to be *reproducible*. What this means is that if we had to rerun our randomization for some reason (say we lose the output, or someone doesn’t believe that we actually randomly assigned individuals to treatment and control), we would like to be able generate the same list of random numbers every time we do the randomization. Generating a reproducible random number involves three steps.
   1. Sort by a variable with only unique observations. Here, schoolid is unique.
   2. Set a “seed”—basically a starting point that stata uses to algorithmically produce a random number.
   3. Generate a random number

sort schoolid

set seed 20131115

gen random = uniform()

1. Sort by this random number (because the sorting is random, there is no statistical reason why the first half of the observations would be any different from the second half)

sort random, stable

Just in case the random number produces two identical random numbers, for example (this is almost impossible), *stable* ensures that those two numbers will always be sort by the original order. Otherwise Stata may “randomly” flip the order sometimes.

1. Create a variable “treatment” which equals 1 if treatment and 0 if control

gen treatment = 0

replace treatment = 1 if \_n <= \_N/2

*OR*

gen treatment =\_n <= \_N/2

*Note: \_N* gives us the total number of observations, and using *\_N/2*, we spit the sample in half—assigning half to treatment or half to control (+/- 1 if we have an odd number of observations). We could chose to divide the sample into any proportion our design requires.

*Note also:* we could have flipped the assignment:

gen treatment = \_n > \_N

Now we have a variable that defines which schools are treatment and which are control.

# Part 2: Stratified Randomization by Language and Gender

In this example we will perform stratified randomization using the gender composition of the school and it’s language of instruction. Follow the steps below:

1. Drop the variables previously created

drop random treatment

1. Create a random number

sort schoolid

set seed 20131115

gen random = uniform()

1. Within each language and gender, sort randomly

sort language gender random

1. Figure out how many schools there are in each stratum

by language gender: gen strata\_size = \_N

*Note:* here, *\_N* gives us the total number of observations within each stratum. Browse the data to view the values of strata\_size, and how they change as the strata change

1. Assign a value reflecting the current (random) order of these schools in each stratum

by language gender: gen strata\_index = \_n

*Note:* here, *\_n* gives us a running number (1, 2, 3…) within each stratum. Browse the data to view how values of strata\_index restart as the strata changes

1. Create a variable “treatment” which equals 1 if treatment and 0 if control, which is based only on the random order

gen treatment = 0

replace treatment = 1 if strata\_index <= (strata\_size/2)

Now we have a variable that defines which schools are treatment and which are control, stratified by language and gender. Again, we are splitting the sample into half treatment, half control.

# Part 3: Stratified Randomization by Gender, Language and Average Pre-test score

In this example we will perform stratified randomization using the gender composition of the school and its language of instruction as well as the mean pre-test score of the school (which is a continuous variable). Follow the steps below:

1. Drop the variables previously created

drop random treatment strata\_size strata\_index

1. Create a random number

sort schoolid

set seed 20131115

gen random = uniform()

1. Sort by language, gender and pretest\_mean

sort language gender pretest\_mean

1. Figure out how many schools are in each *discrete* stratum

by language gender: gen strata\_size = \_N

1. We want to split schools into groups of 4, where each group represents 2 treatment schools and 2 control schools, and have similar pretest\_means. So, for example, if there are 40 schools within a stratum, we want to break the stratum into 10 groups: (40/4 = 10)

by language gender: gen group = group(strata\_size/4)

1. Within each group, sort randomly

sort language group random

1. Figure out how many schools are in each group (ideally, it should be 4, but may very well be less if the number of schools within a stratum is not a multiple of 4).

sort language gender group

by language gender group: gen groupsize = \_N

1. Assign a value reflecting the current (random) order of these schools in each group (should take on a value of 1, 2, 3 or 4).

by language gender group: gen groupindex = \_n

1. If the group has less than 4 schools, make the value equal to 0.

replace groupindex = 0 if groupsize != 4

1. Create a variable “treatment” which equals 1 if treatment and 0 if control (for the schools that are in groups of 4). Here we will assign a school to the treatment group if it’s at the top of the order (groupindex equals 1 or 2).

gen treatment = 0

replace treatment = 1 if groupindex == 1 | groupindex == 2

1. For the schools in groups of less than 4, we cannot stratify by our other observations and assign to both groups. So instead, we can choose to randomize as if they are all equivalent and part of the same language/gender. So this is our new “unstratified” stratum. First we need to see how many observations are in this final group, randomly sort, and then divide.

sum school if groupindex == 0

scalar oddSCHOOL = r(N)

sort groupindex random

replace treatment = 1 if \_n <= (oddSCHOOL/2) & groupindex == 0

1. Now we have a variable that defines which schools are treatment and which are control, stratified by language, gender and pretest\_mean.

# Part 3: Randomization given Multiple Treatments

In this example we will perform randomization for multiple treatment arms (Treatment 1, Treatment 2, and Treatment 1 & 2 combined), without stratification. Follow the steps below:

1. Drop the variables previously created

drop random strata\_size group groupsize groupindex treatment

1. Create a random number

sort schoolid

set seed 20131115

gen random = uniform()

1. Sort by this random number (because the sorting is random, there is no statistical reason why the frst half of the observations would be any different from the second half)

sort random

1. Create a variable “treatment” which equals 1 if treatment 1, 2 if treatment 2, 3 if treatment 1&2 and 0 if control

gen treatment = 0

replace treatment = 1 if \_n <= \_N/4

replace treatment = 2 if \_n > \_N/4 & \_n <= \_N/2

replace treatment = 3 if \_n > \_N/2 & \_n<= \_N\*3/4