



UNIVERSITY OF THESSALY
SCHOOL OF MEDICINE
LABORATORY OF
BIOMATHEMATICS



MASTER PROGRAM IN
“Research Methodology in Biomedicine,
Biostatistics and Clinical Bioinformatics”

Develop a software in Python
for performing different types of randomization
(simple, stratified and block randomization)

By

Kosmas Ziannas

September 2017

EVALUATION COMMITTEE

Kowald Axel, Professor, Supervisor

Zintzaras Elias, Professor

Ioannis Stefanidis, Professor



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΣΧΟΛΗ ΙΑΤΡΙΚΗΣ
ΕΡΓΑΣΤΗΡΙΟ
ΒΙΟΜΑΘΗΜΑΤΙΚΩΝ □



ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
«Μεθοδολογία Βιοιατρικής Έρευνας,
Βιοστατιστική και Κλινική Βιοπληροφορική»

Ανέπτυξε ένα πρόγραμμα στην python
για την εκτέλεση διαφόρων τύπων randomization
(simple, stratified and block randomization)

Από τον

Κοσμά Ζιάννα

Σεπτέμβριος 2017

ΤΡΙΜΕΛΗΣ ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ

Contents

Summary.....	1
Introduction	1
Methods.....	3
1. Simple randomization	3
2. Block randomization	4
3. Stratified randomization.....	4
Python 3.....	6
Implementation	8
Conclusion.....	11
References	12

Summary

In this article, we are developing an application with Python, in order to explore randomization algorithms. Based on the user's choices at the beginning, and at the input he enters, our application produces, based on the proper randomization algorithm, the output. For the implementation of the software we used Python 3.

Introduction

Randomization is a wide field of research. Techniques and methods implemented concerning randomization are widely used in clinical trials and researches, in experimental control, but mostly in biological and medical researches. As stated in Rosenberger et. Al research [1], "The principle of randomization is now a fundamental feature of the scientific method and is employed in many fields of empirical research".

Randomization produces the comparable groups and its aim is to eliminate the source of bias in treatment assignments, when used in clinical trials. Alongside, the use of these methods includes probability theory in order to express the likelihood of chance as a source for the difference of end outcome. In simple terms randomizations is the more former categorization of a given sample.

Randomization has three types, simple, stratified and block. The aim of this software is to make an application which will interact with the user's inputs in order to produce the final outcome or the randomization. Specifically, this application asks the user for the number of subjects and the number of treatments and then performs a randomization of the subjects into the different treatment groups using either one of the aforementioned types.

Randomization creates groups with similar characteristics

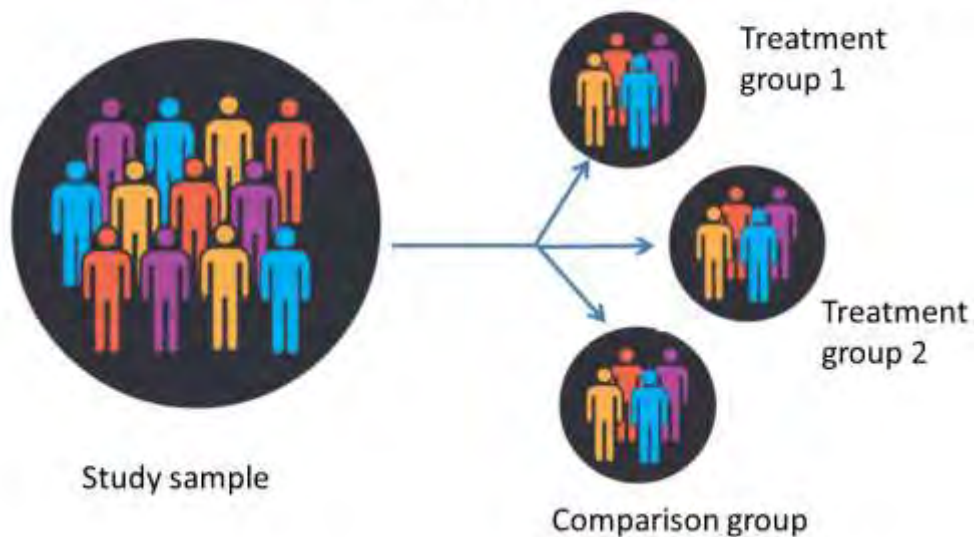


Figure 1: Randomization outcome performed on a study sample.

Study group is the main group in which the randomization methods will be implemented. Comparison group refers to the variables used from researchers in order to determine whether there are effects of the treatment compared with the no treatment group, in a clinical trial.

Methods

In this section, we introduce the methods used in our application and the experiments conducted in details. The purpose of this section is to explain in detail how these methods work and how they were implemented.

1. Simple randomization

Simple randomization is a method based on a single sequence of random Assignments[2]. In simple randomization, each sample has equal terms to be selected, ie equal probabilities. Simple randomization can be simulated with the coin - toss process, which is the distribution of equal probabilities in all the subjects that are involved in the process. In addition, group treatments can be more than two.

In order to implement the simple randomization method, one has to define the group of subject, their numbers and the number of treatment groups that will be used. This procedure can take place through random assignment of values, or let a user, when an interface exists, to enter the data. Following, we present an example referring to the function of simple randomization, in order to understand how this randomization method works.

Example:

In this example we have two treatment groups (A, B). In each group, there are some numbers corresponding to them.

Two treatments	
0-4	A
5-9	B

Table 1

Inserted input (random digits): 779526432008

Treatment assigned											
7	7	9	5	2	6	4	3	2	0	0	8
B	B	B	B	A	B	A	A	A	A	A	B

Table 2

Simple randomization is easy for implementation, though at any point in time, there may be an imbalance in the number of subjects on each treatment and the output may not be acceptable. Furthermore, it is desirable to use restricted methods for randomization, so as to have an accurate and reliable output.

2. Block randomization

The second method is called block randomization. Block randomization is a commonly used technique in clinical trial. Its aim is to reduce bias and achieve balance in the allocation of participants to treatment groups, especially when the sample size is small. This method increases the probability that each group will contain an equal number of individuals by sequencing participant assignments by block [3].

Example:

In this example we have two treatment groups, A and B and block size = 4.

The number of possible allocations – outputs derives from the mathematical equation below:

$$2 \text{ groups } (A, B) \Rightarrow 4C_2 \Rightarrow 6 \text{ possible allocations}$$

This means that we have 6 possible allocations within each block:

1. AABB
2. BABA
3. ABBA
4. BAAB
5. BBAA
6. ABAB

3. Stratified randomization

The stratified randomization method addresses the need to control the possible influence of covariates that would jeopardize the final outcome of the clinical research. Thus, this method defines the stratum to be sampled at the first stage. Next, it determines the number of participants necessary for each stratum, and finally the algorithm randomly samples the participants from each sample in order to produce the final result [4].

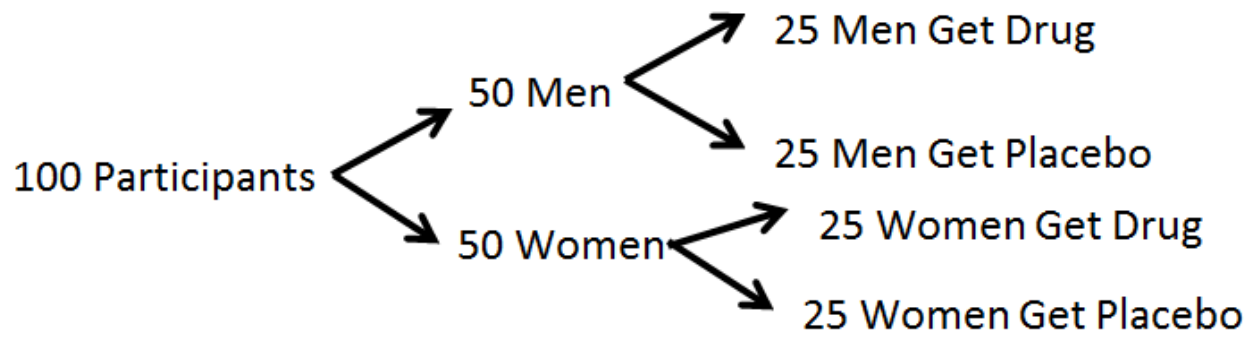


Figure 2: Sample and variables used for stratified randomization method[5].

Stratified randomization takes into account various variables in order to find the best outcome and allocation. However, this method is rarely applicable in trials, because clinical research subjects are often enrolled one at a time on a continuous basis.

Python 3

For the development of this application we used Python 3 on the Anaconda (Spyder) environment. Python is an easy-to-interpret programming language. It belongs to the category of high level programming languages as it has dynamic semantics. Its philosophy makes the code readable and consists of a large standard library. Some of its main features are [6]:

- Ease of learning.
- Readability (easily comprehensible and comprehensible).
- Ease of maintenance.
- Quick Application Development.
- Easy to interpret.
- Very high data structures.
- Easily expandable.
- Open Source.
- Can be used almost everywhere.
- No segmentation faults.
- Automatic memory management.

Another advantage of this language is that it has plenty of ready-made libraries that can be easily used. Libraries can be expanded with new sections written in C or C ++. Python programs are compact, readable, easy to write, and easy to read, as opposed to other programming languages [7].

The interface used to develop our application in the Python programming language is Anaconda - Spyder. This environment is easy to use and can be used by any user to develop the application he desires and to execute easily and directly. Spyder is a complete open source development environment (IDE) for scientific programming in

Python language. It integrates the NumPy, SciPy, Matplotlib and IPython libraries, as well as other open source software. Spyder is extensible with plugins and includes support for interactive tools for data inspection as well as incorporating special code quality assurance systems[8].

Implementation

This section displays the implementation of the algorithms used for the development of this application (Figure 3). Additionally, in Figure 5 we present the initial output of the system, waiting for the user's input. In all the randomization methods we calculate the sample and use random function in order to produce the needed values for the final allocation. The final output concerns the allocation sequence that has been produced.

```
import random
from random import randint

# How ask for input
nr_subjects = int(input("Please enter the number of subjects: ") * nr_input)
nr_treatment = int(input("Please enter the number of treatment: ") * nr_input)
rand_type = int(input("Please enter the number of the wanted randomization type: 1. Simple randomization 2. Block randomization 3. Stratified randomization"))
treat=[]
for i in range(1, nr_treatment):
    treat.append(i)

if rand_type == 1:
    sample = 100/nr_treatment
    random.seed(nr_treatment)
    for j in range(1,nr_subjects+1):
        subject_val = random.random()
        if subject_val < (1/nr_treatment):
            print("Subject ",j,"with val ",subject_val,"belongs to group A")
        elif subject_val < ((1/nr_treatment)*2):
            print("Subject ",j,"with val ",subject_val,"belongs to group B")

elif rand_type == 2:
    print("Block randomization is under construction")
    seq = ""
    block = 4
    a1="A1A1"
    a2="A1A1"
    a3="A1A1"
    a4="B1B1"
    a5="B1B1"
    a6="B1B1"
    for i in range(1,nr_subjects):
        num = randint(1, nr_subjects)
        if num == 1:
            seq = seq + a1
        elif num == 2:
            seq = seq + a2
        elif num == 3:
            seq = seq + a3
        elif num == 4:
            seq = seq + a4
        elif num == 5:
            seq = seq + a5
        elif num == 6:
            seq = seq + a6
```

Επιπρόσθετα το Windows
Μην ξεχάσετε να ελέγξετε για ενοχές

```

    seq= seq + as3
elif num == 4:
    seq= seq + as4
elif num == 5:
    seq= seq + as5
elif num == 6:
    seq= seq + as6
print("Allocation sequence",seq)
elif rand_type == 3:
    #print("stratified randomization is under construction")
    seq = ""
    block = 4
    as1 = "AABB"
    as2="ABAB"
    as3 = "ABBA"
    as4 = "BBAA"
    as5 = "BABA"
    as6 = "BAAB"

    strata = block * nr_treatment
    print("Strata", strata)
    for j in range(1,strata):
        for i in range(1,nr_subjects):
            num = randint(1, nr_subjects)
            if num == 1:
                seq = seq + as1;
            elif num ==2:
                seq = seq + as2;
            elif num == 3:
                seq= seq + as3
            elif num == 4:
                seq= seq + as4
            elif num == 5:
                seq= seq + as5
            elif num == 6:
                seq= seq + as6
    print("Allocation sequence",seq)

```

Figure 3: Application's algorithm

The Output of each algorithm is displayed in Figure 4, Figure 5 and Figure 6.

```

Please enter the number of subjects: 3

Please enter the number of treatment: 2

Please enter the number of the wanted randomization type:
 1. Simple randomization
 2. Block randomzation
 3. Stratified randomization1
Subject 1 with val 0.9560342718892494 belongs to group B
Subject 2 with val 0.9478274870593494 belongs to group B
Subject 3 with val 0.05655136772680869 belongs to group A

```

Figure 4: Simple randomization.

```
Please enter the number of subjects: 3
Please enter the number of treatment: 2
Please enter the number of the wanted randomization type:
1. Simple randomization
2. block randomzation
3. stratified randomization2
Allocation sequence ABABABAB
```

Figure 5: Block randomization.

```
Please enter the number of subjects: 3
Please enter the number of treatment: 2
Please enter the number of the wanted randomization type:
1. Simple randomization
2. Block randomzation
3. Stratified randomization3
Strata 8
Allocation sequence AABBBABBAABABABABABBBAAABBABBBAAABBABBAABABAABBAABBABABABBA
```

Figure 6: Stratified randomization.

Conclusion

The aim of this research concerns the presentation of the randomization methods through the use of an application. The application has been implemented in Python 3 language and can work properly in any command line or other tool used for python scripts.

The result of this work refers to the understanding of these algorithms, how they work, and the results they produce. It is important for the reader both through this research and with the use of the application, to understand how these methods work and maybe then use them to analyze the results of a clinical trial.

Generally, the aim of randomization is to ensure a fair comparison between like groups. Randomly selecting patients for treatment means that groups will typically be similar for important prognostic factors, and so would likely have similar outcomes if they received identical treatment. Thus any observed differences in outcome are likely to result from differences in treatment. As a conclusion, we can use clustering algorithm to help direct us in choosing the appropriate surgical technique for different patients.

References

- [1].Rosenberger, W. F., &Lachin, J. M. (2015). Randomization in clinical trials: theory and practice. John Wiley & Sons.
- [2].Suresh, K. P. (2011). An overview of randomization techniques: an unbiased assessment of outcome in clinical research. Journal of human reproductive sciences, 4(1), 8.
- [3].Efird, J. (2010). Blocked randomization with randomly selected block sizes. International journal of environmental research and public health, 8(1), 15-20.
- [4].Kahan, B. C., & Morris, T. P. (2012). Improper analysis of trials randomised using stratified blocks or minimisation. Statistics in medicine, 31(4), 328-340.
- [5].<http://www.stomponstep1.com/confounding-placebo-stratification-randomization-blinding/>
- [6].Dimitris Leventeas (Group: TasPython), "Python Guide Through Examples", Department of Computer Engineering & Informatics, 2008.
- [7].Nikolaos Aggelidakis, "Introduction to programming with Python", Heraklion, August 2015. <http://aggelid.mysch.gr/pythonbook>
- [8].<https://wiki.python.org/moin/PythonBooks>