

## **Randomized Designs**

### **Design 1 – COMPLETELY RANDOMIZED DESIGN**

Assign 30 patients to treatments A, B, and C, according to a completely randomized design with equal allocation.

```
> N = 30; proportions = c(1/3, 1/3, 1/3);
> Schedule = rmultinom(N,1,proportions);
> colSums(Schedule)      # Number of treatments for each patient
[1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
> rowSums(Schedule)      # Number of patients receiving each treatment
[1] 9 9 12
```

Every patient is assigned to one treatment, but the sample sizes are not equal.

Prepare the treatment schedule:

```
> Treatment = numeric(N);
> K = length(proportions);      # Number of treatments
> TreatmentNames = c("TreatmentA","TreatmentB","Placebo");
> Patient = paste0("Patient",1:N);  # Patients' names or ID
> for (k in 1:K){ Treatment[Schedule[k,]==1] = TreatmentNames[k]; }

> data.frame(Patient,Treatment)
```

	Patient	Treatment		Patient	Treatment
1	Patient1	TreatmentB	16	Patient16	TreatmentA
2	Patient2	Placebo	17	Patient17	Placebo
3	Patient3	Placebo	18	Patient18	TreatmentA
4	Patient4	TreatmentA	19	Patient19	TreatmentB
5	Patient5	TreatmentA	20	Patient20	Placebo
6	Patient6	Placebo	21	Patient21	Placebo
7	Patient7	TreatmentB	22	Patient22	TreatmentA
8	Patient8	TreatmentB	23	Patient23	TreatmentB
9	Patient9	TreatmentB	24	Patient24	TreatmentA
10	Patient10	TreatmentB	25	Patient25	Placebo
11	Patient11	TreatmentB	26	Patient26	TreatmentB
12	Patient12	TreatmentA	27	Patient27	TreatmentA
13	Patient13	Placebo	28	Patient28	TreatmentA
14	Patient14	Placebo	29	Patient29	Placebo
15	Patient15	Placebo	30	Patient30	Placebo

## Design 2 – BLOCK RANDOMIZATION

Create a block randomization with 30 patients, 3 treatments, and equal allocation.

```
> N = 30; K = 3;  
> TreatmentGroup = sample( rep(1:K, N/K) );
```

rep creates a non-random vector 1,2,3,1,2,3,... with equal group sizes  
sample shuffles it randomly

```
> Treatment = TreatmentNames[TreatmentGroup];  
> data.frame(Patient,Treatment)
```

	Patient	Treatment		Patient	Treatment
1	Patient1	TreatmentB	16	Patient16	TreatmentA
2	Patient2	TreatmentB	17	Patient17	Placebo
3	Patient3	Placebo	18	Patient18	Placebo
4	Patient4	TreatmentB	19	Patient19	Placebo
5	Patient5	TreatmentA	20	Patient20	Placebo
6	Patient6	Placebo	21	Patient21	TreatmentA
7	Patient7	TreatmentA	22	Patient22	Placebo
8	Patient8	Placebo	23	Patient23	TreatmentB
9	Patient9	TreatmentA	24	Patient24	TreatmentA
10	Patient10	TreatmentA	25	Patient25	TreatmentB
11	Patient11	TreatmentB	26	Patient26	Placebo
12	Patient12	TreatmentB	27	Patient27	TreatmentA
13	Patient13	TreatmentB	28	Patient28	TreatmentB
14	Patient14	TreatmentB	29	Patient29	TreatmentA
15	Patient15	Placebo	30	Patient30	TreatmentA

```
> table(Treatment)  
Treatment  
Placebo TreatmentA TreatmentB  
      10         10         10
```

**Equal group sizes!**

## Design 3 - STRATIFICATION

There are 18 female and 12 male patients.

Randomize them to 3 treatments equally within each sex group.

```
> N = 30; K = 3;  
> Sex = sample( c(rep("Female",12), rep("Male",18)) );  
> Data = data.frame(ID = 1:N, Patient, Sex)  
> Data
```

	ID	Patient	Sex
1	1	Patient1	Male
2	2	Patient2	Female
3	3	Patient3	Female
4	4	Patient4	Female
5	5	Patient5	Female
6	6	Patient6	Female
7	7	Patient7	Male
8	8	Patient8	Male
9	9	Patient9	Male
10	10	Patient10	Male
11	11	Patient11	Male
12	12	Patient12	Female
13	13	Patient13	Female
14	14	Patient14	Male
15	15	Patient15	Female

	ID	Patient	Sex
16	16	Patient16	Male
17	17	Patient17	Male
18	18	Patient18	Male
19	19	Patient19	Male
20	20	Patient20	Male
21	21	Patient21	Female
22	22	Patient22	Male
23	23	Patient23	Male
24	24	Patient24	Male
25	25	Patient25	Male
26	26	Patient26	Female
27	27	Patient27	Female
28	28	Patient28	Female
29	29	Patient29	Male
30	30	Patient30	Male

```

> table(Sex)
Sex
Female    Male
      12      18
> GroupNames = names(table(Sex))
> Nggroups = length(GroupNames)

```

### Prepare a randomized schedule stratified by sex

```

> Schedule = data.frame()

> for (j in 1:Nggroups){
+   Group = Data[Sex==GroupNames[j],]
+   n = nrow(Group);
+   Treatment = TreatmentNames[sample( rep(1:K, n/K ) )];
+   Schedule = rbind(Schedule,
+                     data.frame(Group,Treatment));
+ }

```

### We can return to the original order, sorting by ID

```

> Schedule = Schedule[order(Schedule$ID),]

> table(Schedule$Sex,Schedule$Treatment)

```

	Placebo	TreatmentA	TreatmentB
Female	4	4	4
Male	6	6	6

### Equal allocation within each sex group

> Schedule

	ID	Patient	Sex	Treatment
1	1	Patient1	Male	TreatmentA
2	2	Patient2	Female	TreatmentA
3	3	Patient3	Female	TreatmentA
4	4	Patient4	Female	Placebo
5	5	Patient5	Female	Placebo
6	6	Patient6	Female	TreatmentB
7	7	Patient7	Male	Placebo
8	8	Patient8	Male	TreatmentB
9	9	Patient9	Male	Placebo
10	10	Patient10	Male	TreatmentA
11	11	Patient11	Male	TreatmentA
12	12	Patient12	Female	TreatmentB
13	13	Patient13	Female	TreatmentB
14	14	Patient14	Male	TreatmentB
15	15	Patient15	Female	TreatmentA
16	16	Patient16	Male	TreatmentB
17	17	Patient17	Male	Placebo
18	18	Patient18	Male	TreatmentB
19	19	Patient19	Male	TreatmentA
20	20	Patient20	Male	Placebo
21	21	Patient21	Female	TreatmentB
22	22	Patient22	Male	Placebo
23	23	Patient23	Male	Placebo
24	24	Patient24	Male	TreatmentB
25	25	Patient25	Male	TreatmentA
26	26	Patient26	Female	Placebo
27	27	Patient27	Female	TreatmentA
28	28	Patient28	Female	Placebo
29	29	Patient29	Male	TreatmentB
30	30	Patient30	Male	TreatmentA