Practical Study Design

Gordana Popovic, Stats Central

UNSW

Summary of the lecture

- Principles of Study Design
  - Asking good questions
  - Population v.s. sample
  - Confounding
  - Independence
- Basic study design with R
  - Randomisation
  - Sample size
- Advanced study design
  - Blocking and stratification
- Really advanced study design
  - Sample size by simulation
  - Factorial designs

Section 1

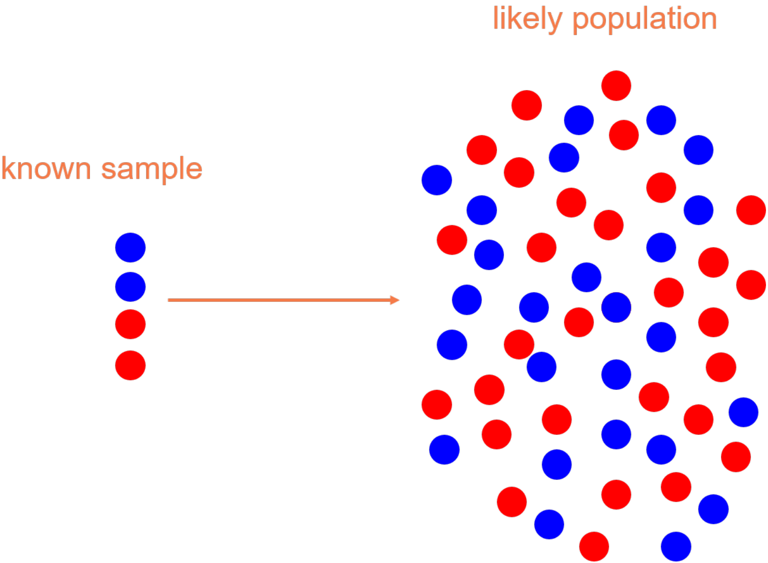
Principles of Study Design

Asking good questions

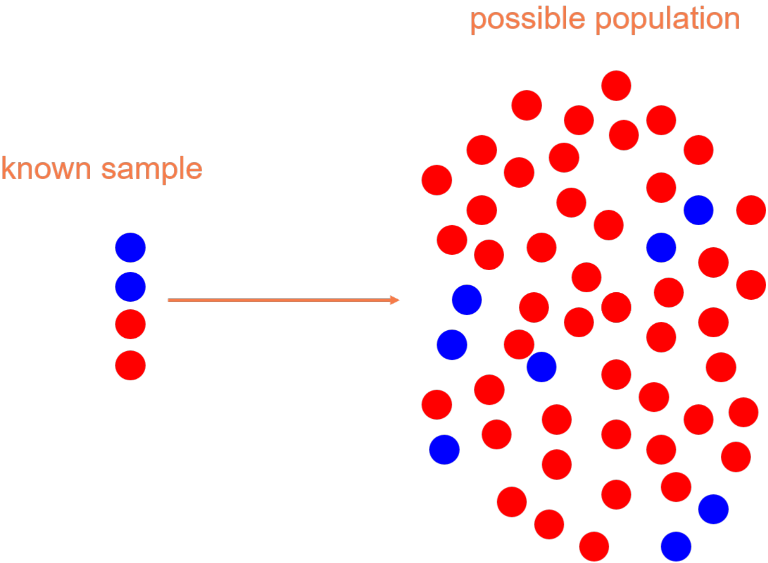
If you ask vague research questions, you will not be able to answer them well using studies and experiments. Things to think about:

- What is your **study population**?
  - in time
  - in space
  - clinically/medically/ecologically
- What is the **response** (dependent variable)?
- What are the **predictors** (independent variables)?
- What is the hypothesised **relationship** between the predictors and response?

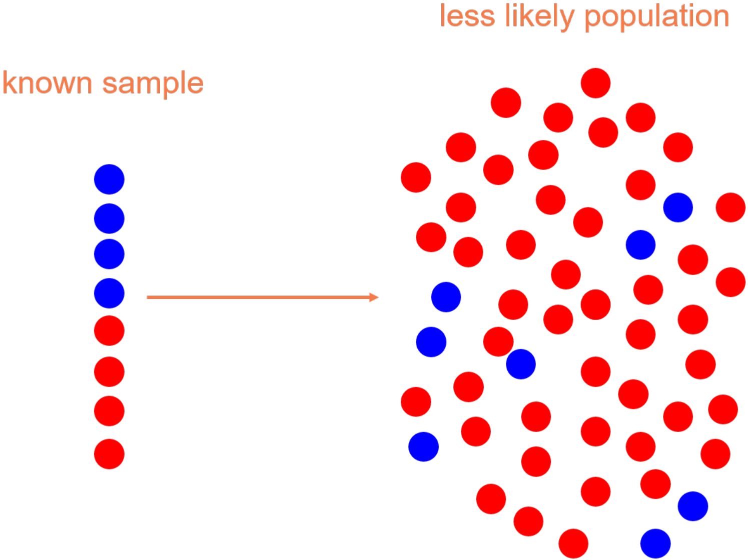
Samples and populations



Samples and populations



Samples and populations



Samples and populations

Scientists generally want to make inferences (draw conclusions) about a (statistical) population, rather than a sample.

- **Biological population** - a group of individuals from the same species (for animals), country (for humans).
- **Statistical population**- the collection of all possible observations, this can be the same or distinct from a biological population. Characteristics of populations are called parameters (e.g. population mean)
- **Sample** - The collection of observations. Characteristics of samples are called statistics (e.g. sample mean)

Any statistic of a sample will differ from the true population parameter.

Confounding

Confounding means that differences due to experimental treatments or predictors cannot be separated from other factors that might be causing the observed differences.

- **Example** - You measure the height of children and their maths ability, you conclude that taller children are better at maths. What is the confounding variable?
- To remove the effects of confounding you have two options
  - Conduct **manipulative experiments** - this can completely remove the effects of confounding - can show causation
  - In **observational studies**, control for the confounding variables - for this to work you have to measure all possible confounders - cannot show causation

Manipulative experiments

To conduct a successful manipulative experiment you need **controls**, **replication**, and **randomisation**.

Many factors can influence the outcome of an experiment, like weather, stress, aliens. These things are often not under our control. We need to know what would have happened if not for the experimental manipulation (treatment) so we can compare what happened to the subjects that received treatment with the background changes in the population of interest. This is why we have **controls**.

Replication is necessary so that you can attribute differences between treatment and control to the treatment, rather than to innate differences between the subjects.

Randomisation allows you to overcome confounding when measuring the treatment effect, because regardless of any confounders, the probability of being in the treatment group is equal for all participants.

Controls

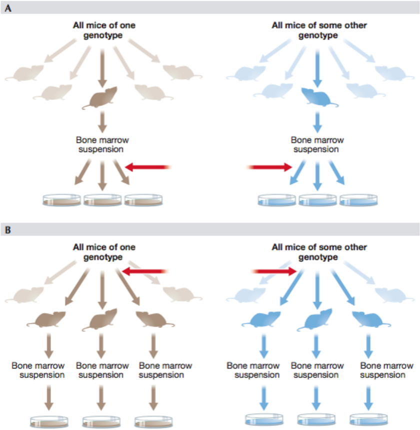
Examples

- Some subjects receiving a sugar pill in a clinical trial
- Excluding predators from some areas and not others in a predator exclusion experiment

Controls should be as similar as possible to the treatment in every way except for the treatment. We give a sugar pill rather than giving no treatment to overcome the **placebo effect**.

Replication

Replication just means you have to multiple subjects/samples in each group.



Observational studies

Observational studies cannot give you the same level of evidence as randomized manipulative experiments, and cannot prove causation. However often observational studies are the only available form of evidence, due to:

- Ethical reasons (we cannot randomize people to smoke)
- Cost (observational studies often use existing data)
- Rare outcomes (no one in your experimental group may develop the outcome)

Even when these are not the case, observational studies are an efficient way to generate hypotheses to later test with manipulative experiments.

Types of observational studies

- Cross sectional
  - one time point for both exposure/treatment and outcome
  - fast / cheap, good for hypothesis generation
  - many *bias* problems
- Case control (retrospective)
  - diagnosed patients matched to similar healthy patients, find out about the past
  - good for rare outcomes, bad for rare exposures
  - a lot of bias can occur, hard to identify good controls
- Cohort (prospective or retrospective)
  - follow people with and without exposure over time, see if they get outcome
  - best evidence of observation studies
  - can estimate temporal association between exposure and outcome
  - expensive and time consuming
  - issues like loss to followup, changes in exposure

Observational studies

You can control for confounding in observational studies by including them in your analysis, however we will never (and measure) all confounding variables, so we can never infer causation.

In observational studies, defining “control” and “treatment” groups can be tricky, and this can also induce a lit of bias, sometimes more than confounding variables.

For example: You are interested in how smoking affects risk of lung cancer using observational data from a longitudinal study with different follow up times for different people. How do you sort your patients into smoking/non smoking group?

How to conduct a good observational study

1. Imagine the ideal clinical trial - target trial
2. Create a detailed written protocol for the target trial, including
   - Eligibility criteria
   - Treatment strategies
   - Assignment procedures
   - Follow-up period
   - Outcome
   - Analysis plan
3. Start trial at time zero - Eligibility criteria need to be met at that point but not later; study outcomes begin to be counted after that point but not earlier.
4. Clearly document any deviations from the target trial protocol (these are inevitable)

See Hernan and Robins (2016) for more detail on how to go about this.

Note that the literature on *causal* inference from observational data is quite confusing. Keep in mind that you cannot actually prove causality with observational data, but you can do a better job of estimating the effect you are interested in.

Section 2

Basic study design with R

Independence and Randomisation

Independence between experimental units means knowing the value of one unit doesn't tell you anything about any other unit.

- Animals in the same aquarium are more similar (exposed to similar conditions) than those in different aquariums.
- Measurements form the same person are more similar than measurements from different people.
- Measurements taken from areas close to one another (in space or time) are more similar than measurements taken further apart.

This dependence must be taken into account in both experimental design and analysis. Independence can be guaranteed by **random** sampling in manipulative experiments and observational studies or **random allocation** of treatments to subjects in manipulative experiments.

If you're not an R user

Basic equal group without stratification

- Randomization.com
- GraphPad
- Excel (Video tutorial)

More complex randomisation

- REDCap
- Excel

Lets randomize with `randomizr`

First we need to install the package and call it up with the `library()` command.

```
install.packages("randomizr")
```

```
library(randomizr)
```

Complete random assignment

The `complete_ra()` function assigns `N` subjects to groups of size given by a vector `m_each`. This can only be done if you know the exact number of subjects beforehand.

To assign 100 subjects equally to 4 groups.

```
Z <- complete_ra(N = 100, num_arms= 4)
table(Z) #number in each treatment
```

```
## Z
## T1 T2 T3 T4
## 25 25 25 25
```

```
Z[1:10] #the first 10 subjects
```

```
##  [1] T3 T2 T1 T2 T4 T2 T1 T4 T3 T3
## Levels: T1 T2 T3 T4
```

Complete random assignment

To assign 100 subjects to three groups called A, B and C with 20, 30 and 50 subjects respectively:

```
Z <- complete_ra(N = 100, m_each = c(20,30,50),
                 conditions=c("A","B","C"))
table(Z) #number in each treatment
```

```
## Z
##  A  B  C
## 20 30 50
```

```
Z[1:10] #the first 10 subjects
```

```
##  [1] C B B C C C B C C B
## Levels: A B C
```

Simple random assignment

If you don't know the total number of subject ahead of time, you can assign the first N subjects to groups with probability given by a vector p with the simple_ra() function. To assign 100 subjects to three groups with unequal probability.

```r
p=c(0.2,0.3,0.5) #must add up to 1
Z <- simple_ra(N = 100, prob_each = p)
table(Z) #number in each treatment
```

```
## Z
## T1 T2 T3
## 24 36 40
```

```r
Z[1:10] #the first 10 subjects
```

```
##  [1] T1 T2 T3 T1 T3 T3 T2 T1 T1 T2
## Levels: T1 T2 T3
```

Power and sample size

Once we have an **experimental** design which has all the properties of good design, a clear question, controls, replication, independence, we still need to decide on a sample size. The sample size required depends on **effect size** and **variation**, as well as the sampling design and intended analysis.

The aim of a study is to detect an effect if one is present. Failure to do so is called type II error, or false negative.

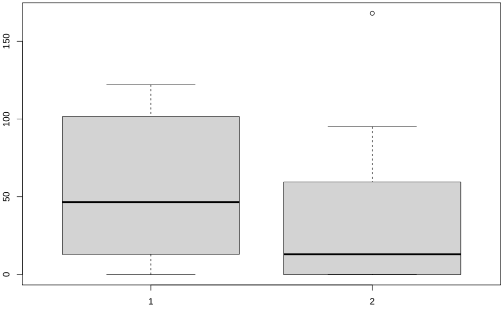
There is no point running an experiment if you have very little chance of detecting and effect. If we have some information about effect size and variability we can conduct a power analysis before starting an experiment, to make sure our sample size is sufficient. The best way to estimate **variability** is to conduct a pilot study. If that is not possible, you can often find estimates of variability in previous literature.

Two sample t-test

This course is not about statistics, but I will mention one statistical test, so I can demonstrate power analysis for some simple experiments.

The two sample t-test tests for differences between two independent groups.

```
##    TR1 TR2
## 1   47   0
## 2   13  61
## 3   13   0
## 4    0   0
## 5   46  58
## 6  122  95
## 7  104   0
## 8   35   0
## 9  110  32
## 10  99 168
## 11  74  26
## 12   0   0
```



Effect size

This is the magnitude of the effect you are hoping to detect. For a simple analysis with two samples (e.g. treatment and control) it is the difference between the population means.

We do not know the true effect size, that is why we are running the experiment. To chose an effect size for the power analysis, we need to consider what effect size is **ecologically/ biologically/ clinically meaningful**.

For example, we don't really care if the treatment and control groups differ by 5%, as this is not clinically interesting, but if they differ by 10% we want to make sure we pick that up. So the effect size we use for the power analysis is 10%, as that is clinically meaningful.

Another common method is to look in previous literature for an effect size, however this may result in an effect size below what we consider relevant.

Not an R user?

- G*Power (basic but free), Video tutorial
- PASS (free trial, then expensive)

Lets calculate sample size with ‘pwr’

Install and call the package.

```
install.packages("pwr")
```

```
library(pwr)
```

Two sample t-test in ‘pwr’

To do power calculations in pwr, you leave one of the values as NULL and enter all the others. It then gives you the value for the one you left out. We want to calculate sample size, so we let n=NULL.

I will ask for a significance level of 0.05, and a power of 0.9 (90%). This means that if the effect size is equal to what I specified (i.e. there is a treatment effect of this size), then I will detect this effect (declare significance at a 0.05 level) 90% of the time with the calculated sample size.

```r
#clinically meaningful difference (same scale as data)
trt.effect <- 0.2
# standard deviation within groups from pilot or literature
sigma <- 0.4
pwr.t.test(n = NULL, d=trt.effect/sigma, sig.level = 0.05,
           power=0.9)
```

Two sample t-test in ‘pwr’

```
## 
##      Two-sample t test power calculation 
## 
##               n = 85.03128
##               d = 0.5
##       sig.level = 0.05
##           power = 0.9
##     alternative = two.sided
## 
## NOTE: n is number in *each* group
```

So we need 86 samples in each group.

Section 3

Advanced study design

Stratification / Blocking

Blocking and stratification are often interchangeably used.

Random allocation of experimental units can lead to high levels of variation between units, which may obscure the effects of the treatment factor of interest. Grouping units into blocks with similar attributes (location, time, age, sex) can explain some variation, this can lead to more precise estimates of parameters of interest and more powerful tests.

Blocking might be unavoidable, for example when sampling patients in hospitals, hospital is a natural blocking factor. Even when blocking is avoidable, it is often possible to create blocking units. For example, instead of allocating all patients randomly to treatments, you may block by some variable which you think will effect the outcome, like age and sex for people, season or location for environment.

Blocking for better power

The way to increase power with blocking is to allocate treatments **within** blocks, so that *e.g* some patients within each hospital get each of the treatments (this is not always possible of course).

Both stratification and blocking **must** be taken into account in analysis, by adding fixed or random effects.

Blocking examples

- Leaves are blocked into trees, we can apply different treatments to leaves on the same tree
- One run of a machine (centrifuge) is a block (if different runs produce slightly different results), so it's good to have some of each of your treatment in each run.
- An individual (person/mouse) can be a blocking factor, and we can (sometimes) sequentially apply all the treatments to each individual.

Blocked/stratified randomisation

`randomizer` calls blocking *block random assignment* and uses a function called `block_ra()`.

All we need in order to block is a blocking variable. Lets introduce a dataset for this that has hair colour, eye colour and sex of 100 people. The first 6 rows of the data look like this.

```
head(datH)
```

```
##    Hair   Eye    Sex
## 1 Black  Blue   Male
## 2 Black Green Female
## 3   Red Hazel   Male
## 4 Brown Brown   Male
## 5 Blond Brown Female
## 6   Red Hazel Female
```

Block randomisation

Say we want to block/stratify by hair colour.

```
table(datH$Hair)
```

```
##
## Black Brown   Red Blond
##    30    21    26    23
```

Block randomisation

We can then stratify by hair into three treatment arms (`num_arms = 3`), with equal numbers of each hair colour in each treatment.

```
datH$trt <- block_ra(blocks = datH$Hair, num_arms = 3)
table(datH$trt, datH$Hair)
```

```
##
##       Black Brown Red Blond
##   T1    10     7   9     8
##   T2    10     7   8     7
##   T3    10     7   9     8
```

```
datH[1:3,]
```

```
##     Hair   Eye    Sex trt
## 1 Black  Blue   Male  T3
## 2 Black Green Female  T3
## 3   Red Hazel   Male  T2
```

Block randomisation

Unequal treatment allocation can be added with a prob_each argument

```
p=c(0.2,0.3,0.5)
datH$trt <- block_ra(blocks = datH$Hair, prob_each = p)
table(datH$trt, datH$Hair)
```

```
##
##       Black Brown Red Blond
##   T1      6     5   5     5
##   T2      9     6   8     7
##   T3     15    10  13    11
```

```
datH[1:3,]
```

```
##     Hair   Eye    Sex trt
## 1 Black  Blue   Male  T1
## 2 Black Green Female  T3
## 3   Red Hazel   Male  T2
```

Power and blocking

Blocking and stratification can have a huge effect on power. Lets consider an experiment with two treatments (study v.s. rest the night before a maths exam) and a set sample size of 50 students. You have the choice of choosing 50 students at random from the whole country (random assignment) or choosing pairs of students from 25 schools , and allocating one of each pair to each treatment.

The response is the students mark in the exam, and you are interested in detecting a difference of 4 marks. You know form previous exam that the standard deviation of exam marks is 7 and that the correlation between two students in a school is 0.6.

Power and blocking

First let's calculate power for the random assignment option (no blocking). Remember we know the sample size, and we want to estimate power, so we set `power=NULL`.

```r
#clinically meaningful difference (same scale as data)
trt.effect <- 4
# standard deviation within groups from pilot or literature
sigma <- 7
```

Power and blocking

```
pwr.t.test(n = 25, d=trt.effect/sigma,
           sig.level = 0.05, power=NULL)
```

```
##
##      Two-sample t test power calculation
##
##               n = 25
##               d = 0.5714286
##       sig.level = 0.05
##           power = 0.507957
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

Power is 0.51, so with this experiment, you have a 51% chance of being able to detect a difference, i.e. half the time you would have wasted your money and time.

Power and blocking

Two sample t-tests for paired data are the same as a one sample t-test for the differences. The major change we need to make to the above code is to calculate the standard deviation of the differences, and add a type="paired" argument.

```
rho=0.6 # correlation
# standard deviation within groups from pilot or literature
sigma <- 7
#formula for standard deviation of differences
sigma_diff=sigma*sqrt(2*(1-rho))
```

Power and blocking

```
pwr.t.test(n = 25, d=trt.effect/sigma_diff,
           sig.level = 0.05, power=NULL, type="paired")
```

```
##
##      Paired t test power calculation
##
##               n = 25
##               d = 0.6388766
##       sig.level = 0.05
##           power = 0.8652638
##     alternative = two.sided
##
## NOTE: n is number of *pairs*
```

Power is now 87%, a huge difference from 51%. With the exact same number of subjects, effort and expense you are now near certain to detect this effect.

Section 4

Really advanced study design

Sample size for more complex analyses

What if you are not doing a t-test, but rather an ANOVA, MANOVA, linear regression, generalised linear model, mixed model . . . ., and still want to estimate a good sample size.

Options

1. For relatively simple things like ANOVA it is still possible to use the `pwr` package.
2. Simplify, often your main research question can be approximately answered by a t-test or ANOVA, if so then it can also give you a good idea of sample size.
3. For really complex analyses, you may have to use simulation.

Sample size by simulation

Basic idea

- Use pilot data to estimate variation for the model you chose
- Simulate data according to that model with the effects size you want, do this many times.
- Calculate what proportion of the simulations that give you a *significant* result, this is your power.

Despite the simple explanation, this is generally very difficult, and everyone, including statisticians find it challenging.

Sample size by simulation

Using pilot data, find values of linear model parameters. In this experiment, I would like to see if there is an effect of Petal length on Sepal length. I have pilot data with 15 observations.

```r
pilot_mod<- lm(Sepal.Length~Petal.Length, data=pilot)
beta_0 = coef(pilot_mod)[1]  #estimated intercept (from pilot
sd = summary(pilot_mod)$sigma  #estimated variability
my_range=range(pilot$Petal.Length) # range of petal length
nsim=500 # number of simulated datasets
```

Then specify a meaningful effect size (here it's the slope of Petal length)

```r
#ecologically meaningful slope (same scale as data)
beta_1 = 1
```

Simulate power for a particular sample size

```r
N = 20 #desired sample size
sim_dat=data.frame(Sepal.Length=NA,
                   Petal.Length=seq(my_range[1],
                                    my_range[2], length = N) )
pval=rep(NA,nsim)
for (i in 1:nsim) {
    mean_y= beta_0 + beta_1 * sim_dat$Petal.Length
    sim_dat$Sepal.Length = rnorm(N, mean = mean_y, sd = sd)
    m = lm(Sepal.Length~Petal.Length, data=sim_dat)
    pval[i] = coef(summary(m))["Petal.Length", "Pr(>|t|)"]
  }  #cycle through all N values
sum(pval < 0.05)/nsim
```

```
## [1] 0.504
```

For this sample size, you only have a 50% chance of detecting the desired effect. You can then make another loop to try different sample sizes to find the one that has enough power for your needs.

Factorial design

You have K treatments each with 2+ levels, you randomize subjects to every combination of treatments. Here is a factorial design for 160 patients experiencing depression to two treatments (K=2), a medications treatment and a counselling treatment, each with two levels.

```
##            counseling
## medication Yes No
##    Active   20 20
##    Placebo  20 20
```

Factorial design

Factorial designs can estimate all main effects (the effect of counselling on depression, the effect of medication on depression), as well as all interactions (does the effect of counselling change by medication). To randomize patients to combinations of treatments you just list the combinations.

```r
dat=data.frame(ID=1:80)
groups=c("A_Y","A_N","P_Y","P_N")
dat$group=complete_ra(N = 80, conditions=groups)
dat[1:3,]
```

```
##   ID group
## 1  1   P_Y
## 2  2   P_Y
## 3  3   P_Y
```

```r
table(dat$group)
```

```
## 
## A_Y A_N P_Y P_N 
##  20  20  20  20
```

Fractional factorial design

Sometimes you don't care about some interactions and can leave out specific combinations of your treatments, this is a fractional factorial design. This can increase your power for detecting particular effects, without increasing your sample size. In R we can use the FrF2 package.

Fractional factorial designs are usually used when each factor has 2 levels only. For factors with more than 2 levels, you can use response surface designs (hard!).

Install and call the package.

```
install.packages("FrF2")
```

```
library(FrF2)
```

Fractional factorial design

If you are interested in main effects only and no interactions, then set `resolution = 3`. If you are interested in main effects and first order interactions only, then set ‘`resolution = 4` and so on.

```r
res=FrF2(nfactors=3, resolution = 3,
         default.levels = c("Yes", "No"))
res
```

```
##     A   B   C
## 1  No  No  No
## 2 Yes Yes  No
## 3  No Yes Yes
## 4 Yes  No Yes
## class=design, type= FrF2
```

The output tells you the combinations of each treatment that you need to randomize patients to. For example, you do not need the treatment combination A=Yes, B=Yes and C=Yes. Note you **must** take design into account when you analyse data.

Fractional factorial design

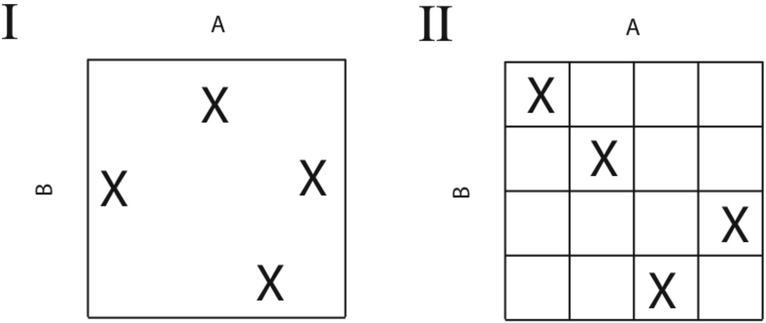
Let's randomize 32 patients to the 4 combinations we need.

```
dat=data.frame(ID=1:32)
groups=c("AY_BY_CN","AY_BN_CY","AN_BN_CN","AN_BY_CY")
dat$group=complete_ra(N = 32, conditions=groups)
head(dat)
```

```
##   ID    group
## 1  1 AN_BN_CN
## 2  2 AN_BN_CN
## 3  3 AY_BN_CY
## 4  4 AN_BN_CN
## 5  5 AN_BY_CY
## 6  6 AN_BY_CY
```

Latin hypercube

What if you have many continuous predictors you are interested in, and want to set up an experiment with samples in a more spread out way than a random sample would.

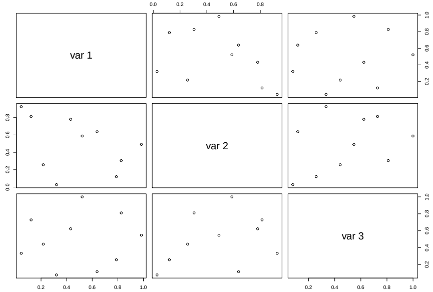


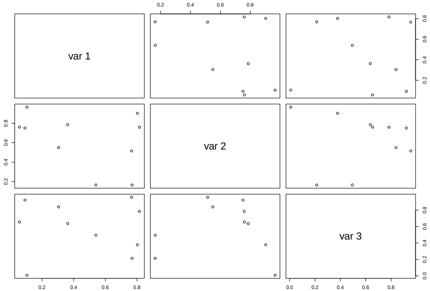
Latin hypercube

To construct a Latin hypercube with 20 samples and 3 continuous predictors.

```
library(lhs)
A <- randomLHS(n=10, k=3)
pairs(A)
```

```
# random sample
A=matrix(runif(30),ncol=3)
pairs(A)
```





Summary

- Always start by defining a precise research question, before collecting any data
- Ideally do a manipulative experiment with controls, replication and randomisation, so you can infer causation
- Ensure independence by random assignment
- Aim to have a trial with high power with appropriate sample size.
- Improve power for free by blocking.
- Otherwise do an observation study, control for confounding variables, use a target trial, and don't interpret effects as causation
- Ask for help, early and often: statscentral.unsw.edu.au