Data Simulation and Power Simulation in R

Julian Quandt

Material at: bit.ly/3LyaiCr

Click on Code and then Download ZIP to download the material!

A few words before we start

- I know that many of you are here to learn about how to simulate power for mixed-effects models.

  → I think that there are already good R packages to achieve this, and also tutorials that talk about this e.g. faux or simR

- **Focus here:** Understand the underlying machinery and reasoning behind simulations by simulating data from scratch **without using any packages**! This will make us greatly flexible in what we can simulate.
- I have not given this workshop before and do not know how far we get but I plan on having simulated power for mixed-effects models by the end of this.
- However, if we don't get this far, I tried to make the slides "wordy" and self-explanatory, so you can also make use of them later on.
- I wrote a very long 4-part tutorial on power simulation, that contains all information talked about here and also many points that I don't have the time to talk about today but that are very important, so if after today you think that power simulation is a useful skill, I strongly encourage reading the tutorial.

Table of Content

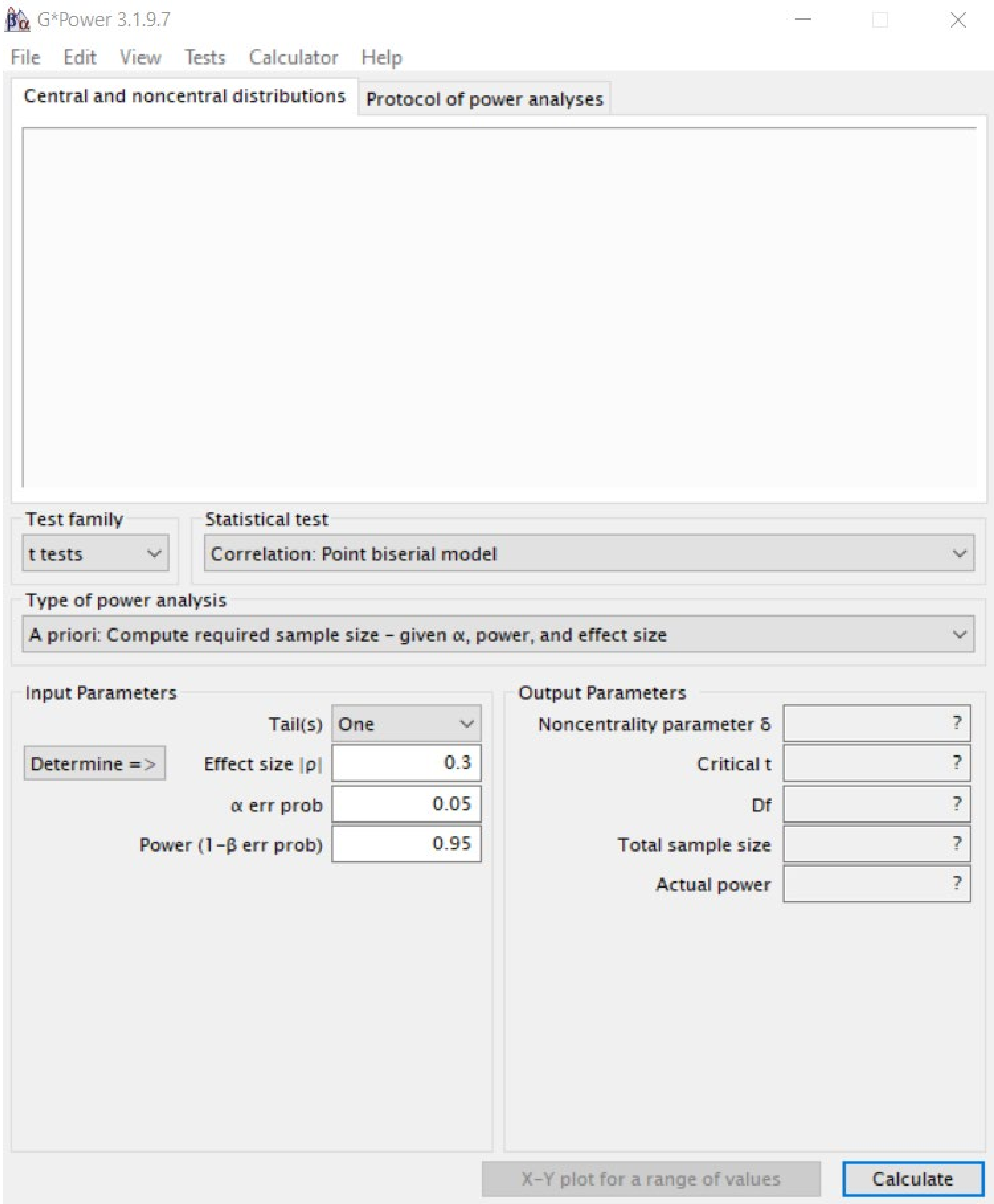
Why are we here?

Replication crisis has sparked renewed focus on research-practice / methodology

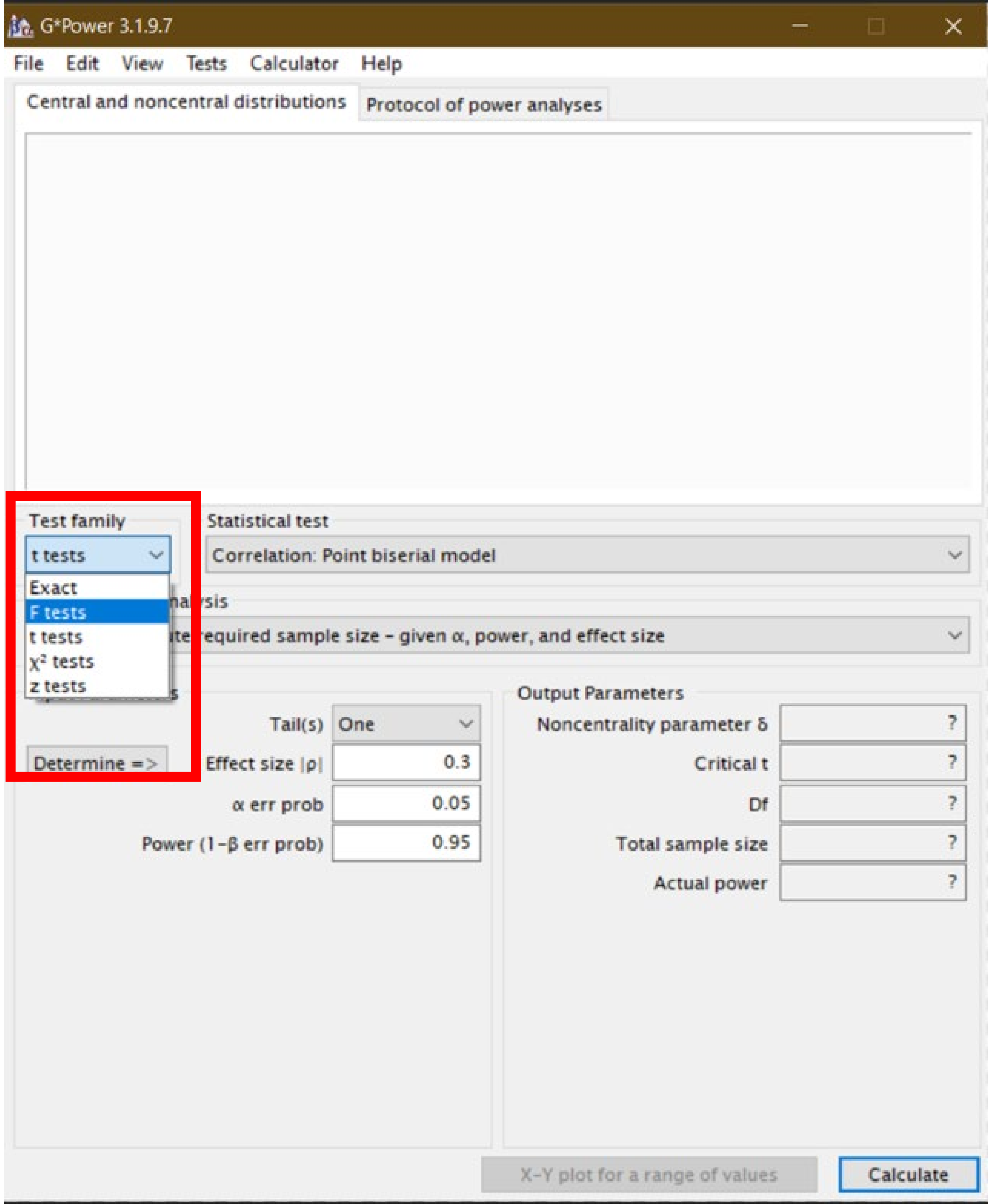
Power analysis is a standard step of planning a research project and preregistration

So before collecting data to investigate a research question (for example whether people rate environmental preservation as more important after seeing two different video clips) we would conduct power analysis…

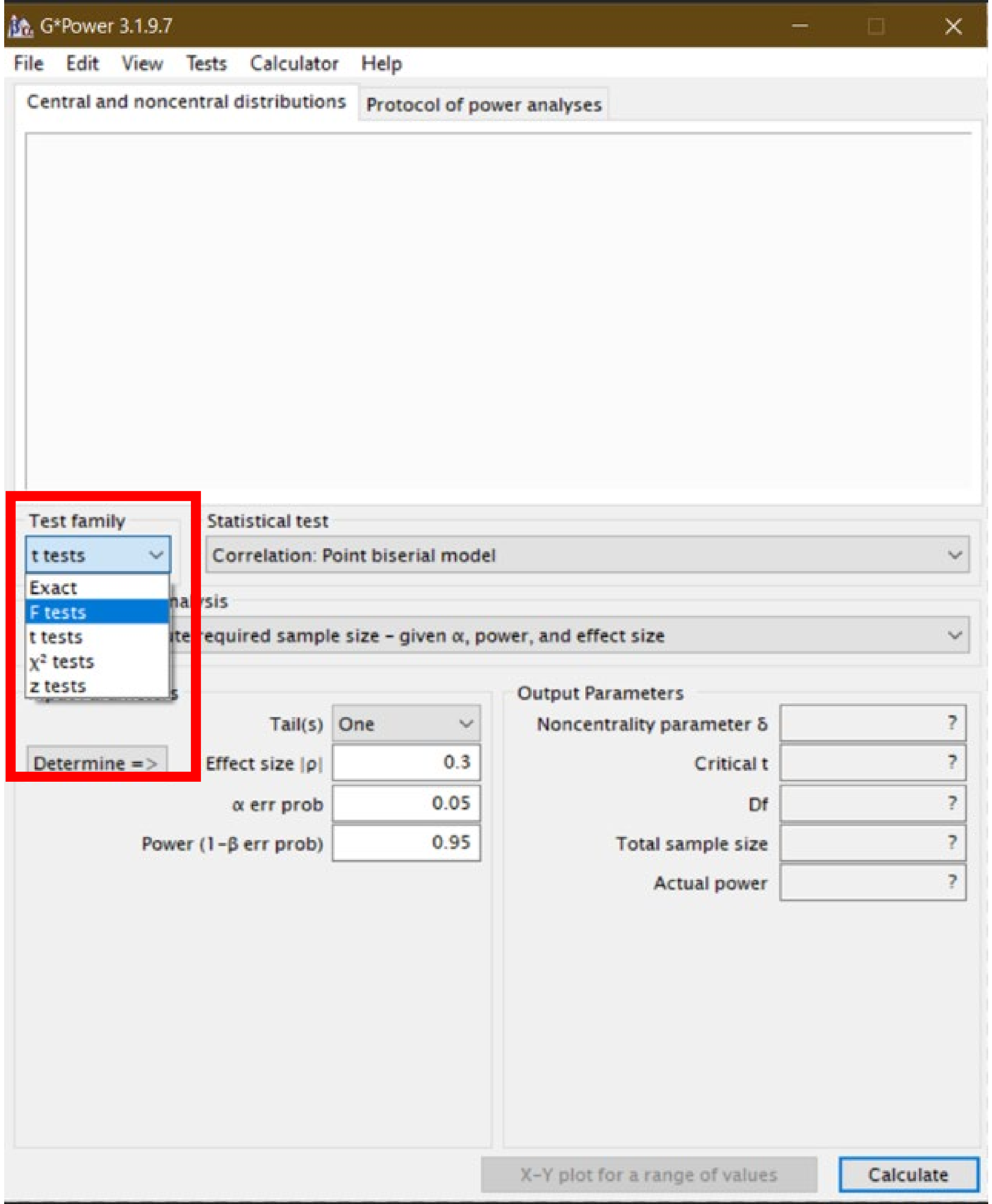
- Open G*Power


G*Power 3.1.9.7
File Edit View Tests Calculator Help
Central and noncentral distributions
Protocol of power analyses
Test family
t tests
Statistical test
Correlation: Point biserial model
Type of power analysis
A priori: Compute required sample size – given α, power, and effect size
Input Parameters
Tail(s)
One
Determine =>
Effect size |ρ|
0.3
α err prob
0.05
Power (1-β err prob)
0.95
Output Parameters
Noncentrality parameter δ
Critical t
Df
Total sample size
Actual power
X-Y plot for a range of values
Calculate

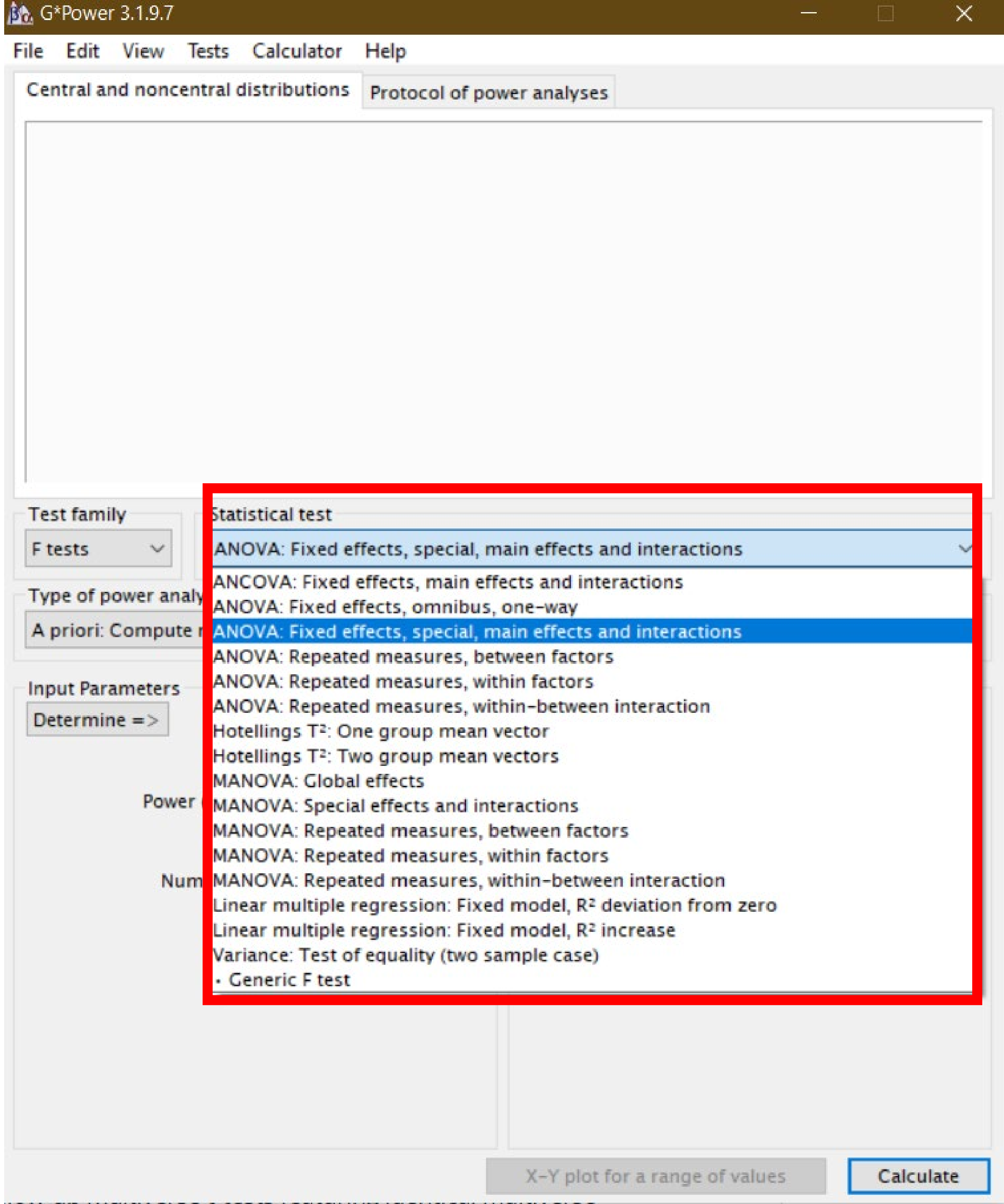
- Open G*Power
- Select the right model family



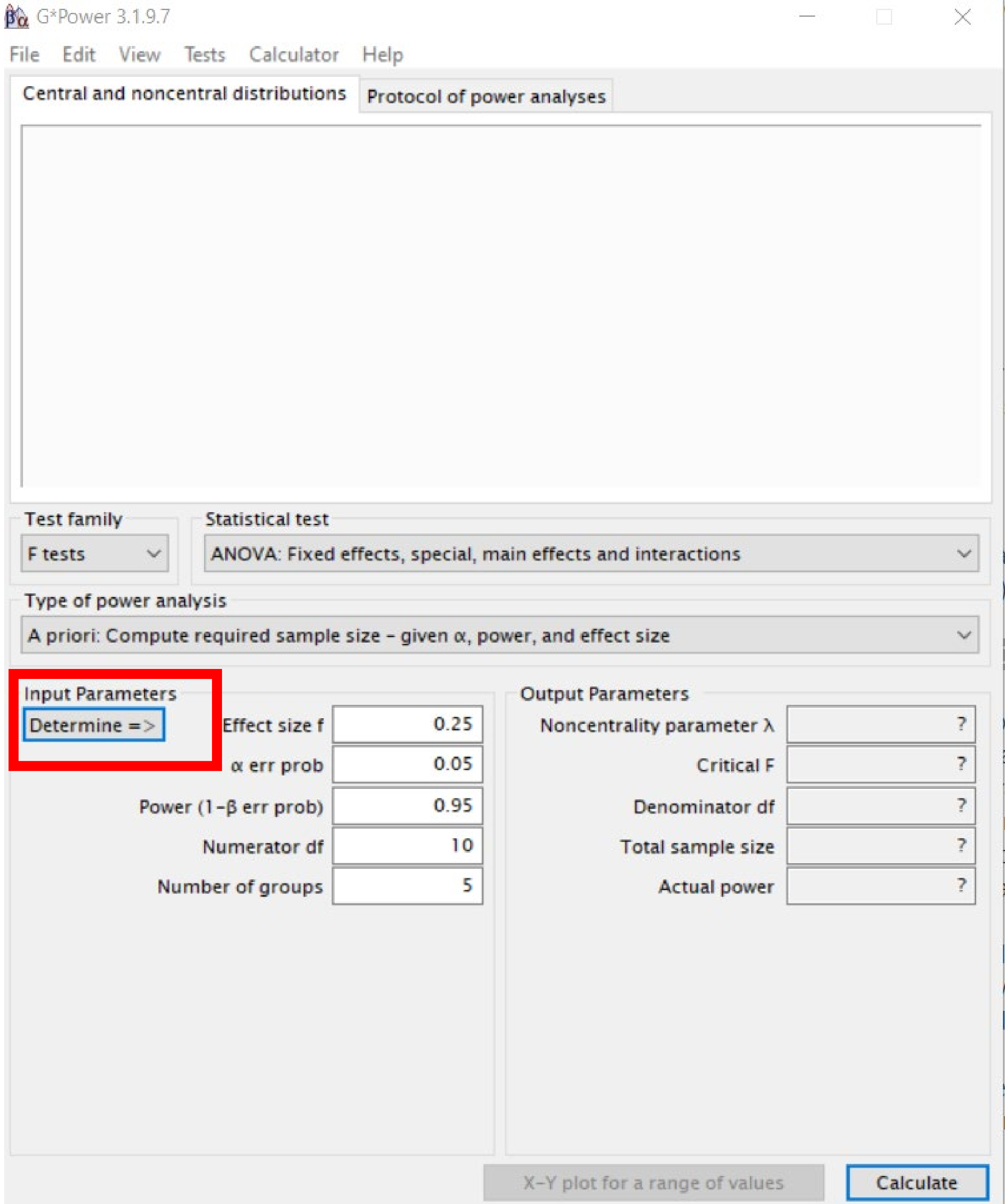
- Open G*Power
- ~~Select the right test family~~
- Realize we are not sure, so randomly click through everything until it says “ANOVA” somewhere



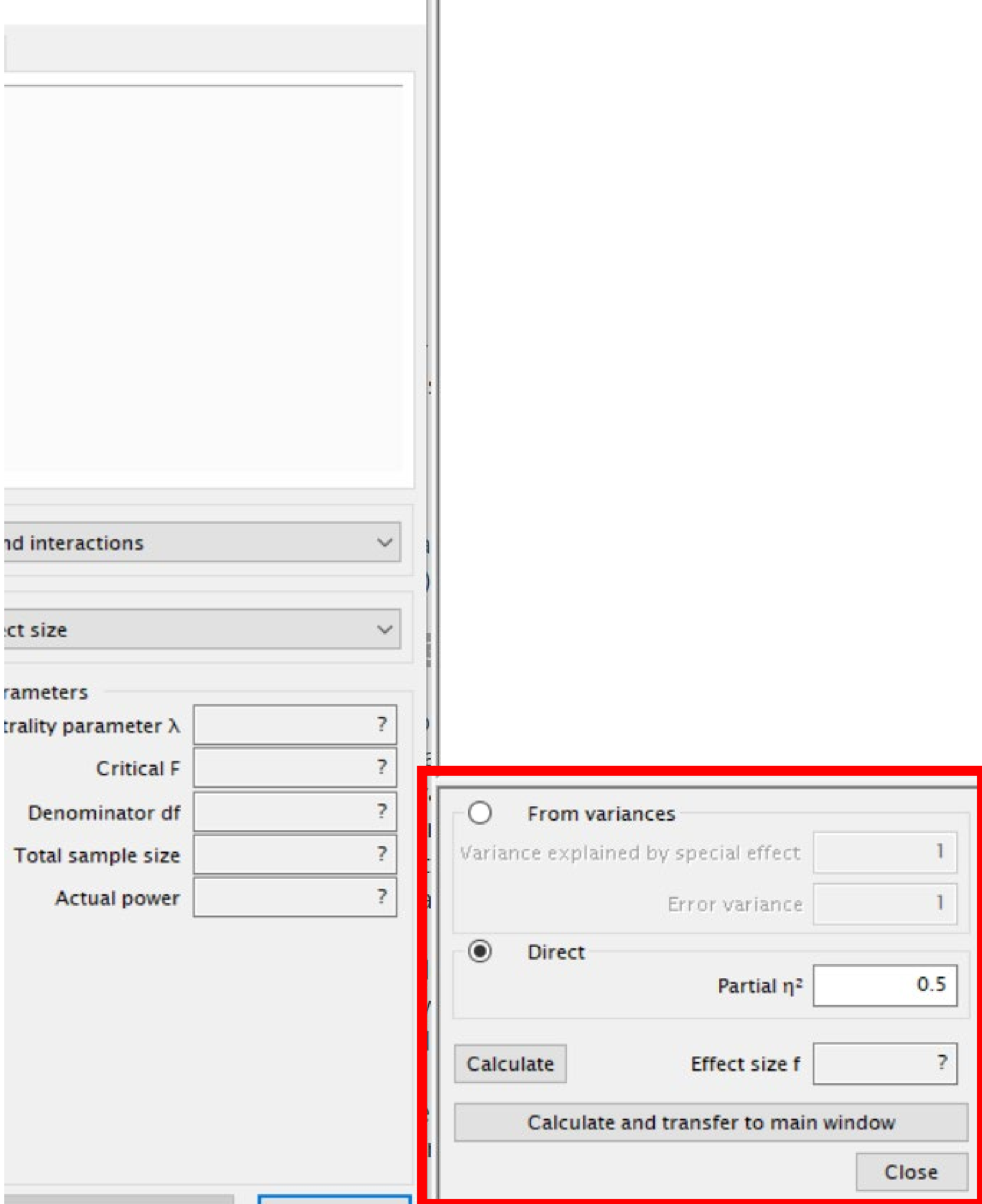
- Open G*Power
- ~~Select the right test family~~
- Randomly click through everything it says “ANOVA” somewhere
- We are interested in an interaction so select “ANOVA: Fixed effects, main effects and interactions”



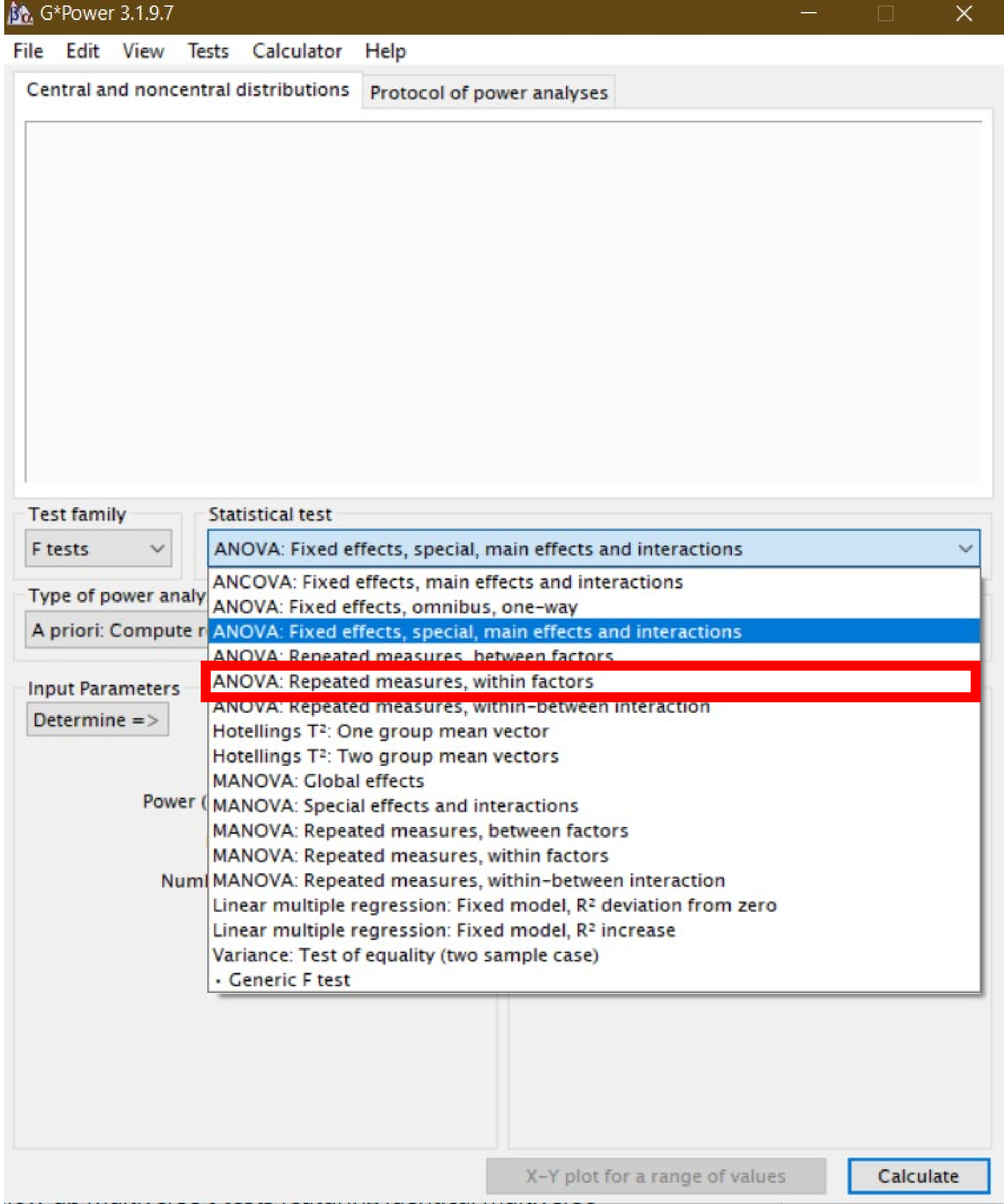
- Open G*Power
- ~~Select the right test family~~
- Randomly click through everything it says "ANOVA" somewhere
- We are interested in an interaction so select "ANOVA: Fixed effects, main effects and interactions"
- Realize being unsure about what to use for effect size f so click on "Determine"



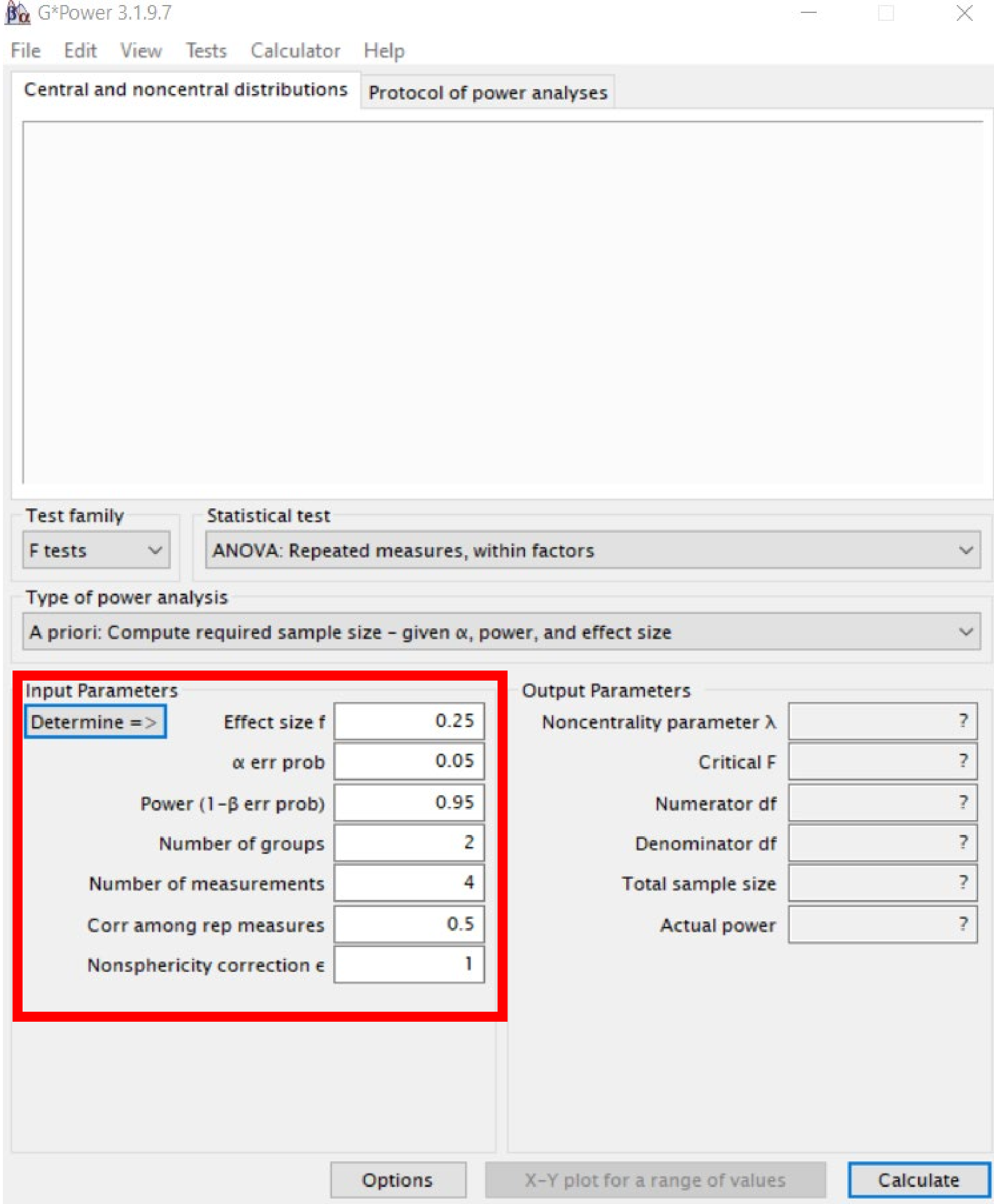
- Open G*Power
- ~~Select the right test family~~
- Randomly click through everything it says “ANOVA” somewhere
- We are interested in an interaction so select “ANOVA: Fixed effects, main effects and interactions”
- Realize being unsure about what to use for effect size f so click on “Determine”
- Ok no idea what to do with this either…



- Open G*Power
- ~~Select the right test family~~
- Randomly click through everything it says “ANOVA” somewhere
- We are interested in an interaction so select “ANOVA: Fixed effects, main effects and interactions”
- Realize being unsure about what to use for effect size f so click on “Determine”
- Ok no idea what to do with this either…
- Also realize we are interested in interactions but also have a “within-subject” design so probably we have to change that to “ANOVA: Repeated measures, within factors”



- Open G*Power
- ~~Select the right test family~~
- Randomly click through everything it says "ANOVA" somewhere
- We are interested in an interaction so select "ANOVA: Fixed effects, main effects and interactions"
- Realize being unsure about what to use for effect size f so click on "Determine"
- Ok no idea what to do with this either…
- Also realize we are interested in interactions but also have a "within-subject" design so probably we have to change that to "ANOVA: Repeated measures, within factors"
- Ok.. Now there are even more things to specify now so look up all of this is in the G*Power manual



- Open G*Power
- ~~Select the right test family~~
- Randomly click through everything it says "ANOVA" somewhere
- We are interested in an interaction so select "ANOVA: Fixed effects, main effects and interactions"
- Realize being unsure about what to use for effect size f so click on "Determine"
- Ok no idea what to do with this either…
- Also realize we are interested in interactions but also have a "within-subject" design so probably we have to change that to "ANOVA: Repeated measures, within factors"
- Ok.. Now there are even more things to specify now so look up all of this is in the [G*Power manual]
- The selected analysis is nowhere to be found so go to one of the other ANOVA's instead

January 21, 2021

This manual is not yet complete. We will be adding help on more tests in the future. If you cannot find help for your test in this version of the manual, then please check the G*Power website to see if a more up-to-date version of the manual has been made available.

Contents

- Open G*Power
- ~~Select the right test family~~
- Randomly click through everything it says "ANOVA" somewhere
- We are interested in an interaction so select "ANOVA: Fixed effects, main effects and interactions"
- Realize being unsure about what to use for effect size f so click on "Determine"
- Ok no idea what to do with this either…
- Also realize we are interested in interactions but also have a "within-subject" design so probably we have to change that to "ANOVA: Repeated measures, within factors"
- Ok.. Now there are even more things to specify now so look up all of this is in the G*Power manual
- The selected analysis is nowhere to be found so go to one of the other ANOVA's instead
- Try reading this for a while, realize it is still not entirely clear

11 F test: Fixed effects ANOVA - special, main effects and interactions

This procedure may be used to calculate the power of main effects and interactions in fixed effects ANOVAs with factorial designs. It can also be used to compute power for planned comparisons. We will discuss both applications in turn.

11.0.1 Main effects and interactions

To illustrate the concepts underlying tests of main effects and interactions we will consider the specific example of an $A \times B \times C$ factorial design, with $i = 3$ levels of A, $j = 3$ levels of B, and $k = 4$ levels of C. This design has a total number of $3 \times 3 \times 4 = 36$ groups. A general assumption is that all groups have the same size and that in each group the dependent variable is normally distributed with identical variance.

In a three factor design we may test three main effects of the factors A, B, C, three two-factor interactions $A \times B$, $A \times C$, $B \times C$, and one three-factor interaction $A \times B \times C$. We write μ_{ijk} for the mean of group $A = i, B = j, C = k$. To indicate the mean of means across a dimension we write a star ($\star$) in the corresponding index. Thus, in the example $\mu_{ij\star}$ is the mean of the groups $A = i$, $B = j$, $C = 1, 2, 3, 4$. To simplify the discussion we assume that the grand mean $\mu_{\star\star\star}$ over all groups is zero. This can always be achieved by subtracting a given non-zero grand mean from each group mean.

In testing the **main effects**, the null hypothesis is that all means of the corresponding factor are identical. For the main effect of factor A the hypotheses are, for instance:

$$H_0 : \mu_{1\star\star} = \mu_{2\star\star} = \mu_{3\star\star}$$

$$H_1 : \mu_{i\star\star} \neq \mu_{j\star\star} \text{ for at least one index pair } i, j.$$

The assumption that the grand mean is zero implies that $\sum_i \mu_{i\star\star} = \sum_j \mu_{\star j\star} = \sum_k \mu_{\star\star k} = 0$. The above hypotheses are therefore equivalent to

$$H_0 : \mu_{i\star\star} = 0 \text{ for all } i$$

$$H_1 : \mu_{i\star\star} \neq 0 \text{ for at least one } i.$$

In testing **two-factor interactions**, the residuals $\delta_{ij\star}$, $\delta_{i\star k}$, and $\delta_{\star ik}$ of the groups means after subtraction of the main effects are considered. For the $A \times B$ interaction of the example, the $3 \times 3 = 9$ relevant residuals are $\delta_{ij\star} = \mu_{ij\star} - \mu_{i\star\star} - \mu_{\star j\star}$. The null hypothesis of no interaction effect states that all residuals are identical. The hypotheses for the $A \times B$ interaction are, for example:

$$H_0 : \delta_{ij\star} = \delta_{kl\star} \text{ for all index pairs } i, j \text{ and } k, l.$$

$$H_1 : \delta_{ij\star} \neq \delta_{kl\star} \text{ for at least one combination of } i, j \text{ and } k, l.$$

The assumption that the grand mean is zero implies that $\sum_{i,j} \delta_{ij\star} = \sum_{i,k} \delta_{i\star k} = \sum_{j,k} \delta_{\star jk} = 0$. The above hypotheses are therefore equivalent to

$$H_0 : \delta_{ij\star} = 0 \text{ for all } i, j$$

$$H_1 : \delta_{ij\star} \neq 0 \text{ for at least one } i, j.$$

In testing the **three-factor interactions**, the residuals δ_{ijk} of the group means after subtraction of all main effects and all two-factor interactions are considered. In a three factor design there is only one possible three-factor interaction. The $3 \times 3 \times 4 = 36$ residuals in the example are calculated as $\delta_{ijk} = \mu_{ijk} - \mu_{i\star\star} - \mu_{\star j\star} - \mu_{\star\star k} - \delta_{ij\star} - \delta_{i\star k} - \delta_{\star jk}$. The null hypothesis of no interaction states that all residuals are equal. Thus,

$$H_0 : \delta_{ijk} = \delta_{lmn} \text{ for all combinations of index triples } i, j, k \text{ and } l, m, n.$$

$$H_1 : \delta_{ijk} \neq \delta_{lmn} \text{ for at least one combination of index triples } i, j, k \text{ and } l, m, n.$$

The assumption that the grand mean is zero implies that $\sum_{i,j,k} \delta_{ijk} = 0$. The above hypotheses are therefore equivalent to

$$H_0 : \delta_{ijk} = 0 \text{ for all } i, j, k$$

$$H_1 : \delta_{ijk} \neq 0 \text{ for at least one } i, j, k.$$

It should be obvious how the reasoning outlined above can be generalized to designs with 4 and more factors.

11.0.2 Planned comparisons

Planned comparison are specific tests between levels of a factor planned before the experiment was conducted.

One application is the **comparison between two sets of levels**. The general idea is to subtract the means across two sets of levels that should be compared from each other and to test whether the difference is zero. Formally this is done by calculating the sum of the component-wise product of the mean vector $\vec{\mu}$ and a nonzero contrast vector $\vec{c}$ (i.e. the scalar product of $\vec{\mu}$ and c): $C = \sum_{i=1}^{k} c_i \mu_i$. The contrast vector c contains negative weights for levels on one side of the comparison, positive weights for the levels on the other side of the comparison and zero for levels that are not part of the comparison. The sum of weights is always zero. Assume, for instance, that we have a factor with 4 levels and mean vector $\vec{\mu} = (2, 3, 1, 2)$ and that we want to test whether the means in the first two levels are identical to the means in the last two levels. In this case we define $\vec{c} = (-1/2, -1/2, 1/2, 1/2)$ and get $C = \sum_i \vec{\mu}_i \vec{c}_i = -1 - 3/2 + 1/2 + 1 = -1$.

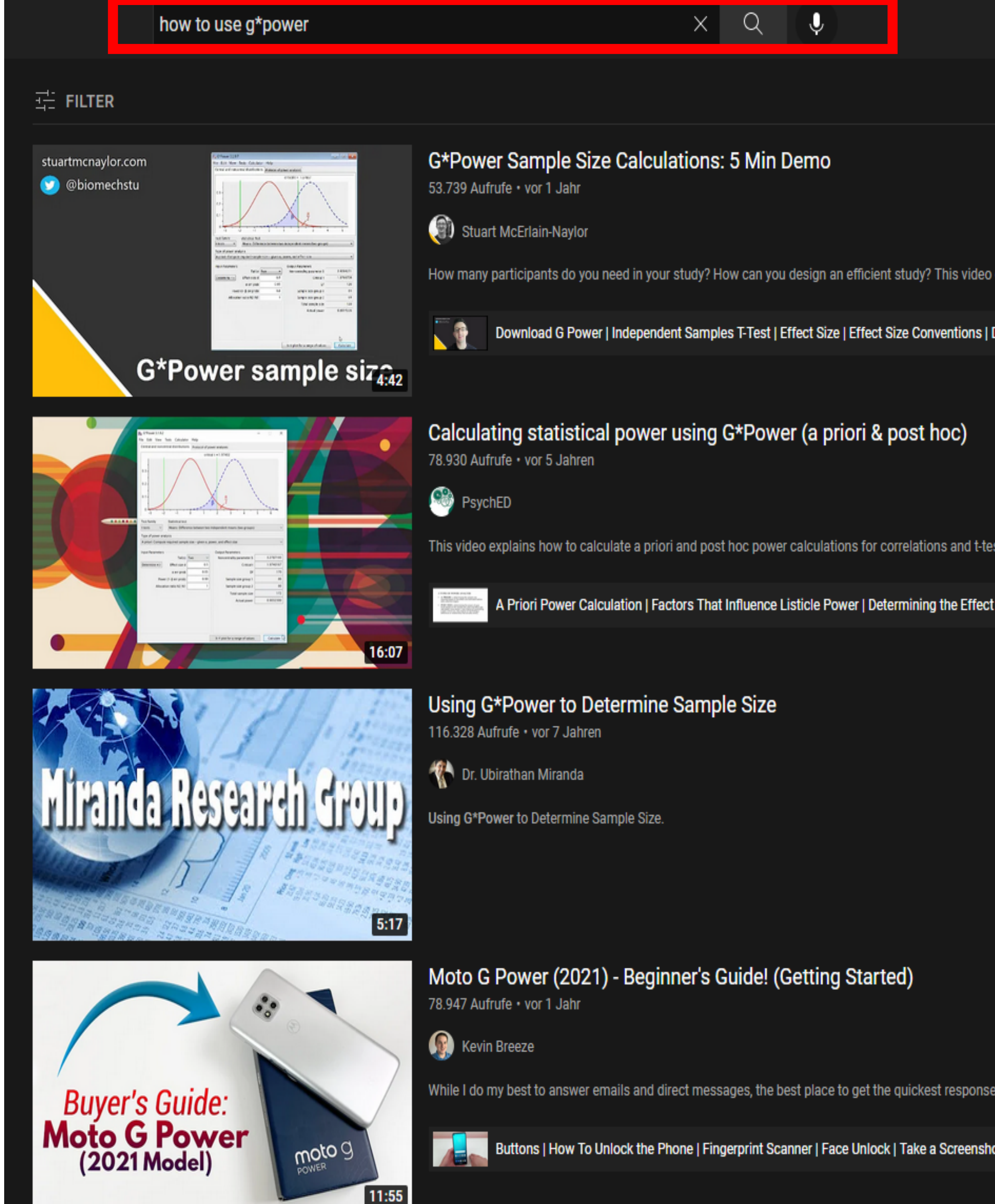
A second application is testing polygonal contrasts in a **trend analysis**. In this case it is normally assumed that the factor represents a quantitative variable and that the levels of the factor that correspond to specific values of this quantitative variable are equally spaced (for more details, see e.g. Hays (1988, p. 706ff)). In a factor with k levels $k - 1$ orthogonal polynomial trends can be tested.

In planned comparisons the null hypothesis is: $H_0 : C = 0$, and the alternative hypothesis $H_1 : C \neq 0$.

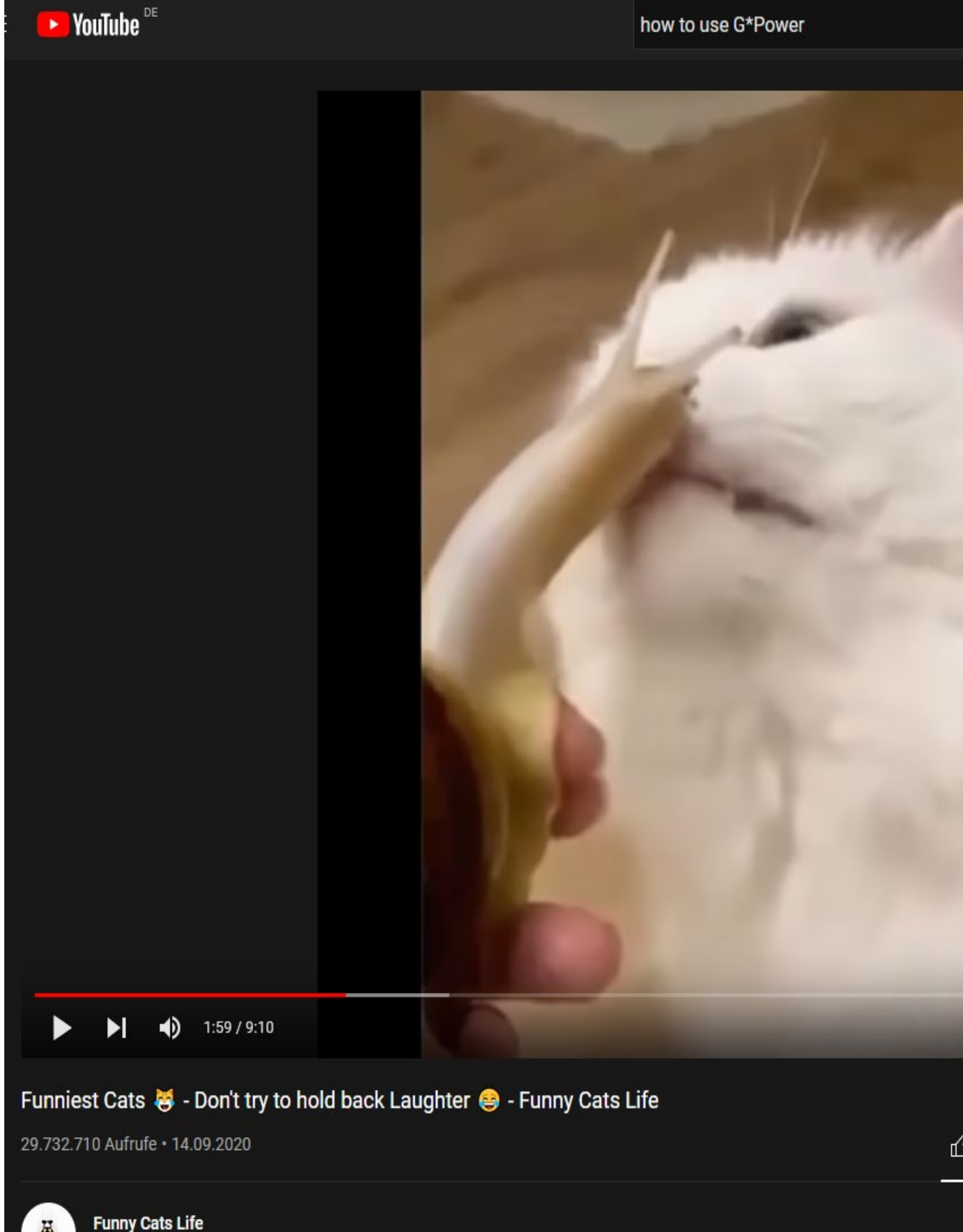
11.1 Effect size index

The effect size f is defined as: $f = \sigma_m / \sigma$. In this equation σ_m is the standard deviation of the effects that we want to test and σ the common standard deviation within each of the groups in the design. The total variance is then $\sigma_t^2 = \sigma_m^2 + \sigma^2$. A different but equivalent way to specify the effect size

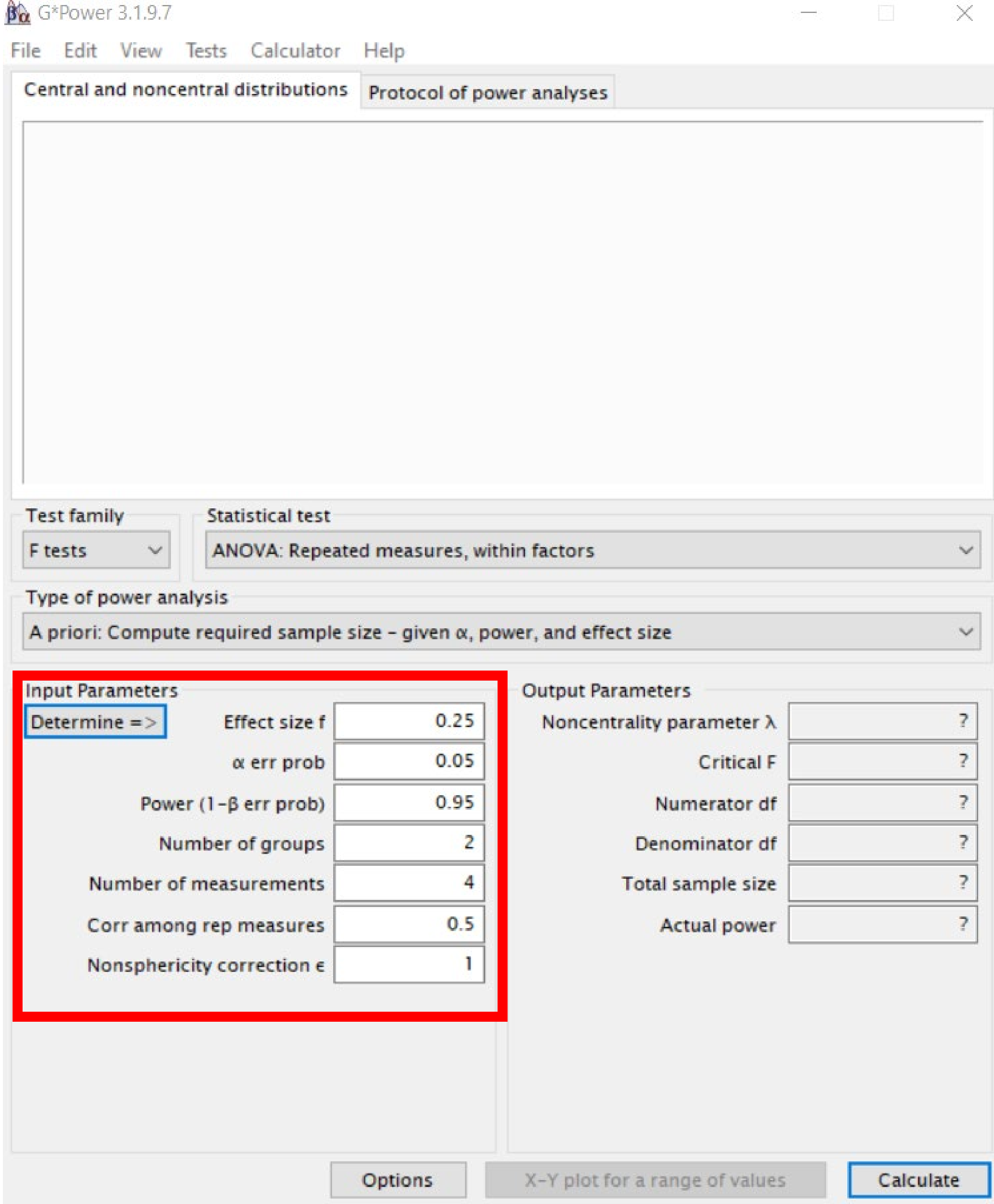
- Open G*Power
- ~~Select the right test family~~
- Randomly click through everything it says “ANOVA” somewhere
- We are interested in an interaction so select “ANOVA: Fixed effects, main effects and interactions”
- Realize being unsure about what to use for effect size f so click on “Determine”
- Ok no idea what to do with this either…
- Also realize we are interested in interactions but also have a “within-subject” design so probably we have to change that to “ANOVA: Repeated measures, within factors”
- Ok.. Now there are even more things to specify now so look up all of this is in the [G*Power manual]
- The selected analysis is nowhere to be found so go to one of the other ANOVA’s instead
- Try reading this for a while, realize it is still not entirely clear
- Search for “How to use G*Power” on youtube



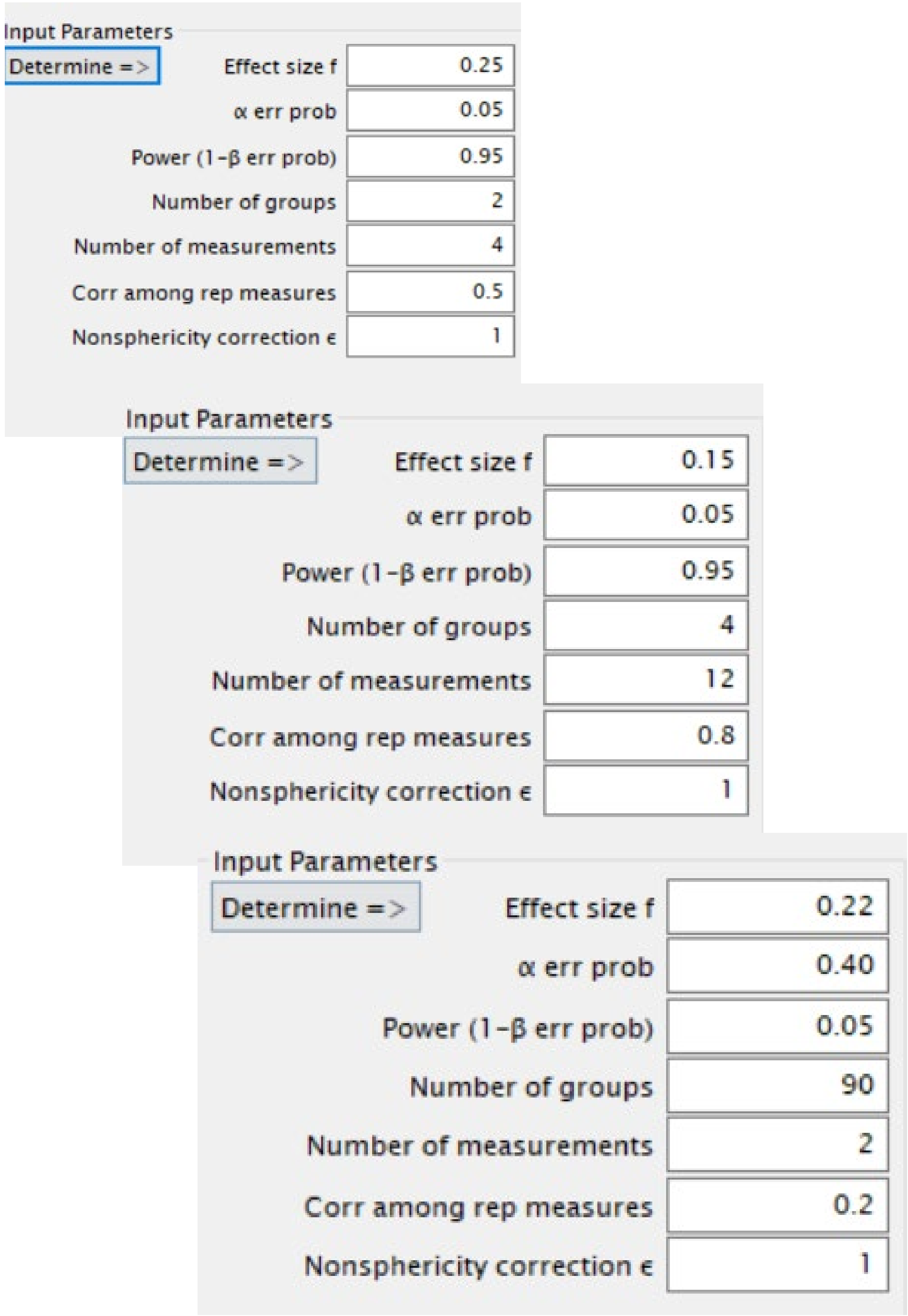
- Open G*Power
- ~~Select the right test family~~
- Randomly click through everything it says “ANOVA” somewhere
- We are interested in an interaction so select “ANOVA: Fixed effects, main effects and interactions”
- Realize being unsure about what to use for effect size f so click on “Determine”
- Ok no idea what to do with this either…
- Also realize we are interested in interactions but also have a “within-subject” design so probably we have to change that to “ANOVA: Repeated measures, within factors”
- Ok.. Now there are even more things to specify now so look up all of this is in the [G*Power manual]
- The selected analysis is nowhere to be found so go to one of the other ANOVA’s instead
- Try reading this for a while, realize it is still not entirely clear
- Search for “How to use G*Power” on youtube
- Watch 3 videos until the algorithm suggests watching “**Funniest Cats 😹 - Don't try to hold back Laughter 😂 - Funny Cats Life**” instead



- Open G*Power
- ~~Select the right test family~~
- Randomly click through everything it says “ANOVA” somewhere
- We are interested in an interaction so select “ANOVA: Fixed effects, main effects and interactions”
- Realize being unsure about what to use for effect size f so click on “Determine”
- Ok no idea what to do with this either…
- Also realize we are interested in interactions but also have a “within-subject” design so probably we have to change that to “ANOVA: Repeated measures, within factors”
- Ok.. Now there are even more things to specify now so look up all of this is in the [G*Power manual]
- The selected analysis is nowhere to be found so go to one of the other ANOVA’s instead
- Try reading this for a while, realize it is still not entirely clear
- Search for “How to use G*Power” on youtube
- Watch 3 videos until the algorithm suggests watching “**Funniest Cats 😹 - Don't try to hold back Laughter 😂 - Funny Cats Life”** instead
- After watching 10 more cat videos, go back to G*Power… still not sure what to do, but also really needing to finish that preregistration



- Open G*Power
- ~~Select the right test family~~
- Randomly click through everything it says “ANOVA” somewhere
- We are interested in an interaction so select “ANOVA: Fixed effects, main effects and interactions”
- Realize being unsure about what to use for effect size f so click on “Determine”
- Ok no idea what to do with this either…
- Also realize we are interested in interactions but also have a “within-subject” design so probably we have to change that to “ANOVA: Repeated measures, within factors”
- Ok.. Now there are even more things to specify now so look up all of this is in the G*Power manual
- The selected analysis is nowhere to be found so go to one of the other ANOVA’s instead
- Try reading this for a while, realize it is still not entirely clear
- Search for “How to use G*Power” on youtube
- Watch 3 videos until the algorithm suggests watching “**Funniest Cats 😹 - Don't try to hold back Laughter 😂 - Funny Cats Life**” instead
- After watching 10 more cat videos, go back to G*Power… still not sure what to do, but also really needing to finish that preregistration
- Eventually just decide to try 20 different combinations of the parameters until it makes some intuitive sense and the sample size seems reasonable enough to collect



- Open G*Power
- ~~Select the right test family~~
- Randomly click through everything it says “ANOVA” somewhere
- We are interested in an interaction so select “ANOVA: Fixed effects, main effects and interactions”
- Realize being unsure about what to use for effect size f so click on “Determine”
- Ok no idea what to do with this either…
- Also realize we are interested in interactions but also have a “within-subject” design so probably we have to change that to “ANOVA: Repeated measures, within factors”
- Ok.. Now there are even more things to specify now so look up all of this is in the G*Power manual
- The selected analysis is nowhere to be found so go to one of the other ANOVA’s instead
- Try reading this for a while, realize it is still not entirely clear
- Search for “How to use G*Power” on youtube
- Watch 3 videos until the algorithm suggests watching “**Funniest Cats 😹 - Don't try to hold back Laughter 😂 - Funny Cats Life”** instead
- After watching 10 more cat videos, go back to G*Power… still not sure what to do, but also really needing to finish that preregistration
- Eventually just decide to try 20 different combinations of the parameters until it makes some intuitive sense and the sample size seems reasonable enough to collect
- **Preregister running 30 participants per condition**

This is of course exaggerated, but the point here is:

- Power analysis software is often unnecessarily complicated

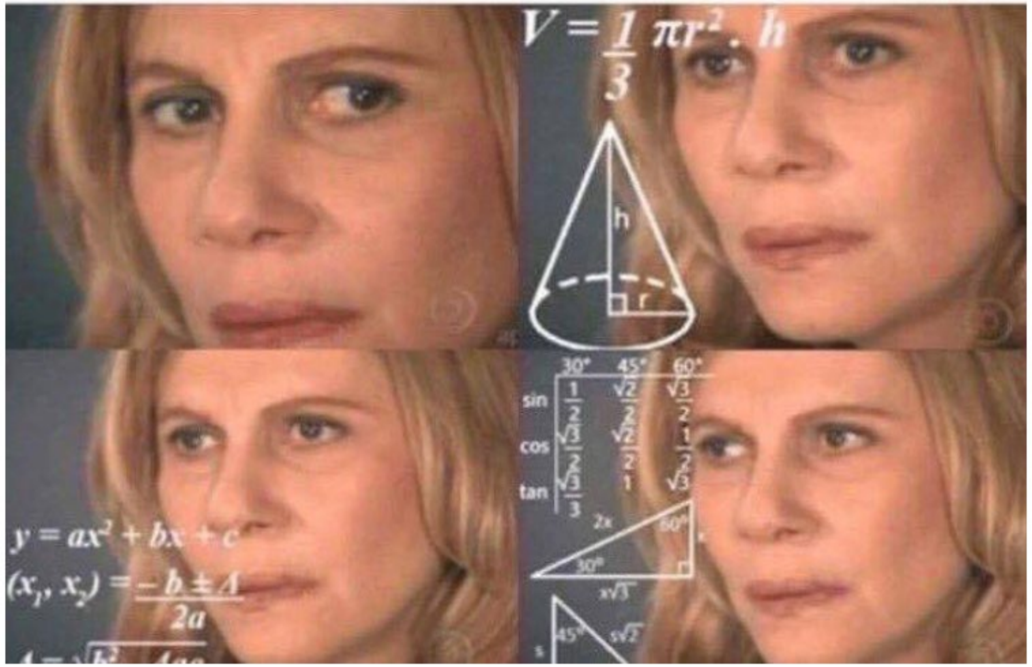


Figure 1. Me trying to figure out how to use a standard power-analysis software.

This is of course exaggerated, but the point here is:

- Power analysis software is often unnecessarily complicated

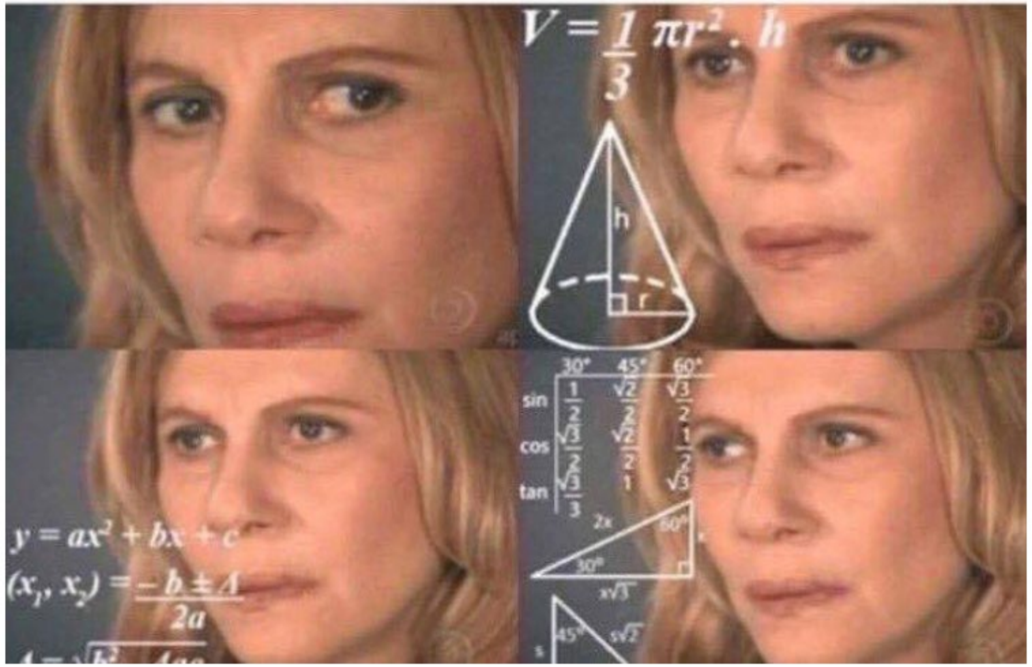


Figure 1. Me trying to figure out how to use a standard power-analysis software.

- The split into different model families, statistical procedures and arbitrary parameters is artificial and unnecessary, as we will discuss soon.

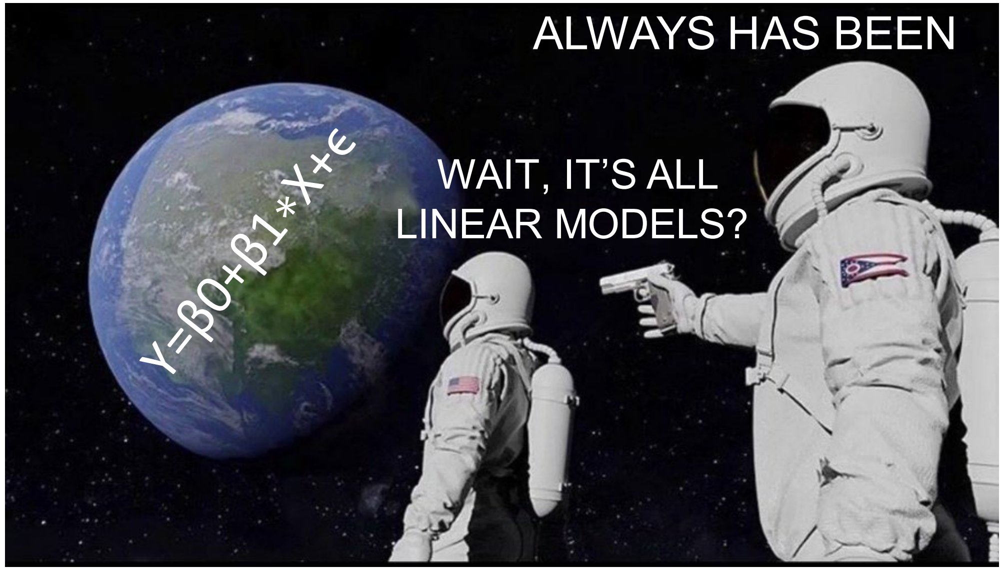


Table of Content

Part I

- ~~Why are we here?~~
- **Statistical testing and the counting of possibilities**
- Counting possibilities in R with probability functions and calculating power
- Simulating data and introduction to Power Simulation

Part II

- Practical One: Making use of Probability functions and Simulation functions
- Simulating Linear models
- Practical Two: Simulating linear models
- Mixed-effects models: What else is there to consider
- Practical Three: Simulating mixed-effects models

Statistical tests as counting possibilities

Before talking about data simulation it can be very helpful to develop some intuition for how simulations relate to statistical testing, so let’s start with the easiest possible example of a statistical test that we could think of…

Example

Imagine we are tossing a coin 3 times and it lands on HEADS 3 times in a row

Question: **Is this coin fair or should we consider it to be unfair?**

In NHST scenario: *Assuming the coin is fair (null-hypothesis), what number of heads would we consider too improbable to keep believing that the coin is fair?*

Statistical tests as counting possibilities

How do we find out how likely 3 heads are?

We can write down all possible outcomes and count the ones that produce 3 HEADS to see how often that would happen.

We see that there are 8 (2^3) possible outcomes only 1 of which would produce 3 HEADS

If we repeated this 3-toss experiment very often, 1 in 8 (12.5%) would produce 3 HEADS in 3 tosses.

→**<u>Who thinks that this coin is unfair?</u>**

	Toss #1	Toss #2	Toss #3	number of heads
Possibility #1	HEAD	HEAD	HEAD	3
Possibility #2	HEAD	HEAD	TAIL	2
Possibility #3	HEAD	TAIL	TAIL	1
Possibility #4	TAIL	TAIL	TAIL	0
Possibility #5	TAIL	TAIL	HEAD	1
Possibility #6	TAIL	HEAD	HEAD	2
Possibility #7	TAIL	HEAD	TAIL	1
Possibility #8	HEAD	TAIL	HEAD	2

Statistical tests as counting possibilities

10 out of 10 Heads: possibility table would have 1024 rows (2^{10}), one of which produces 10 Heads (i.e. about 0.1%)

→ **Who thinks that this coin is unfair?**

→**If we write this result up as a statistical test, can you guess what the p-value is and with a commonly used alpha level of .05 would we consider the coin unfair?**

→ The people who consider this coin unfair have internally used a higher **alpha level** compared to people who still believe this coin is fair.

Interim Summary

Statistical testing under NHST can be expressed in terms of counting possibilities.

In the coin tossing example, we can make a table with all possible sequences of results that we could get.

If we count how many of these possibilities result in 10 heads, we can divide this number by the total possibilities (1 / 1024 ≈ .001) to get the p-value.

If we define a proportion beforehand that we consider as “too unlikely to still believe the coin is fair”, (e.g. .05) we can see if the p-value is smaller than that number and draw conclusions based on this.

Table of Content

Part I

- ~~Why are we here?~~
- ~~Statistical testing and the counting of possibilities~~
- **Counting possibilities in R with probability functions and calculating power**
- Simulating data and introduction to Power Simulation

Part II

- Practical One: Making use of Probability functions and Simulation functions
- Simulating Linear models
- Practical Two: Simulating linear models
- Mixed-effects models: What else is there to consider
- Practical Three: Simulating mixed-effects models

Counting possibilities in R with probability functions

We have now seen how to do a significance test for a coin-flipping experiment. But how do we calculate the power of such an experiment?

To talk about power, we need to talk about **unfairness**.

→Power is the chance that **if any unfairness in the coin exists**, we would detect it with a given number of coin tosses… Thus, we need to define what we mean by **unfairness**.

For example: What is the chance that we would **detect an unfairness of 10%** (a coin landing on heads 60% instead of 50% of the times) **when we toss it 10 times**.

Counting possibilities in R with probability functions

If we follow in the steps of our scientific foremothers and forefathers we would stop believing in the fairness of the coin if the observed number of heads (or more heads) happens in **less than 5%** of all possibilities

The power question: Assuming that a coin is fair, when tossing it 10 times, what is the chance of detecting a 10% unfairness if we would only stop believing in a fair coin when the observed number of heads would happen in less than 5% of all possibilities?

In other words we answer the following two questions:

1.) What number of heads has a less than 5% chance of occurring with a fair coin?

2.) What is the chance of observing at least that number of heads with a coin that would land on heads 60% of the time?

→How do we answer the power question by writing down and counting possibilities?

Counting possibilities in R with probability functions

1) What number of heads has a less than 5% chance of occurring with a fair coin?

→Make a table with 1024 sequences and mark the 5% of them (1024*0.05 = 52) with the highest number of heads. Anything **larger than** the lowest number in this sequence has less than a 5% chance of occurring when tossing a coin 10 times.

Slide notes: line 9 to 50.

2) What is the chance of observing more than that number of heads with a coin that would land on heads 60% of the time?

→ We need to write down possibilities of the unfair coin and count how many sequences out of the total number of sequences have more than that number of heads

Slide notes: line 50 to 100.

Counting possibilities in R with probability functions

Fortunately, there is an easier way to do this in R by using *probability functions*. **Slide notes lines 102 - 140**

1) What number of heads has a less than 5% chance of occurring with a fair coin?

```
qbinom(p = .05, size = 10, prob = .5, lower.tail = FALSE)
```

2) What is the chance of observing more than that number of heads with a coin that would land on heads 60% of the time?

```
pbinom(q = 8, size = 10, prob = .6, lower.tail = FALSE)
```

Counting possibilities in R with probability functions

In a power analysis, we normally want to see when power reaches a certain threshold, for example 80%

In other words, we want to know how often we would have to toss a coin until 80% of the sequences with the unfair coin have more than 60% heads.

→ **How can we use the `qbinom()` and `pbinom()` functions to answer this question in R?**

→ We put this in a loop and try it repeatedly until we reach the desired power. **Slide notes lines 140 - 180**

Counting possibilities in R with probability functions

<u>Interim Summary</u>

Power analysis can be achieved merely by writing up all possibilities and counting how often events of interest (e.g. more than 8 heads) happen if we work with a hypothesized unfair coin.

R has an array of probability functions like `qbinom()` and `pbinom()` that help us with this job.

The `qbinom()` function answers the question: *"At what number of heads would the probability to get that many heads be smaller than our alpha level?"*

The `pbinom()` function answers the question: "*What are the chances of getting x or more heads with a coin that lands on heads with a certain probability*"

By repeatedly using these functions in a loop, we can perform power analysis.

Table of Content

Simulating data and introduction to Power Simulation

What we have done so far were **not yet power simulations**. We just made use of the entire table of possibilities, to look for those possibilities that resulted in more than the critical number of heads with our unfair coin.

→**How do we make use of the possibility table in a simulation-based approach**

In a simulation, we make use of the fact that randomly drawing rows from the table of possibilities, will eventually resemble the actual table of possibilities very closely.

To simulate coin tosses in R we make use of the `rbinom()` function. **Slide notes lines 183 - 255**

Interim Summary

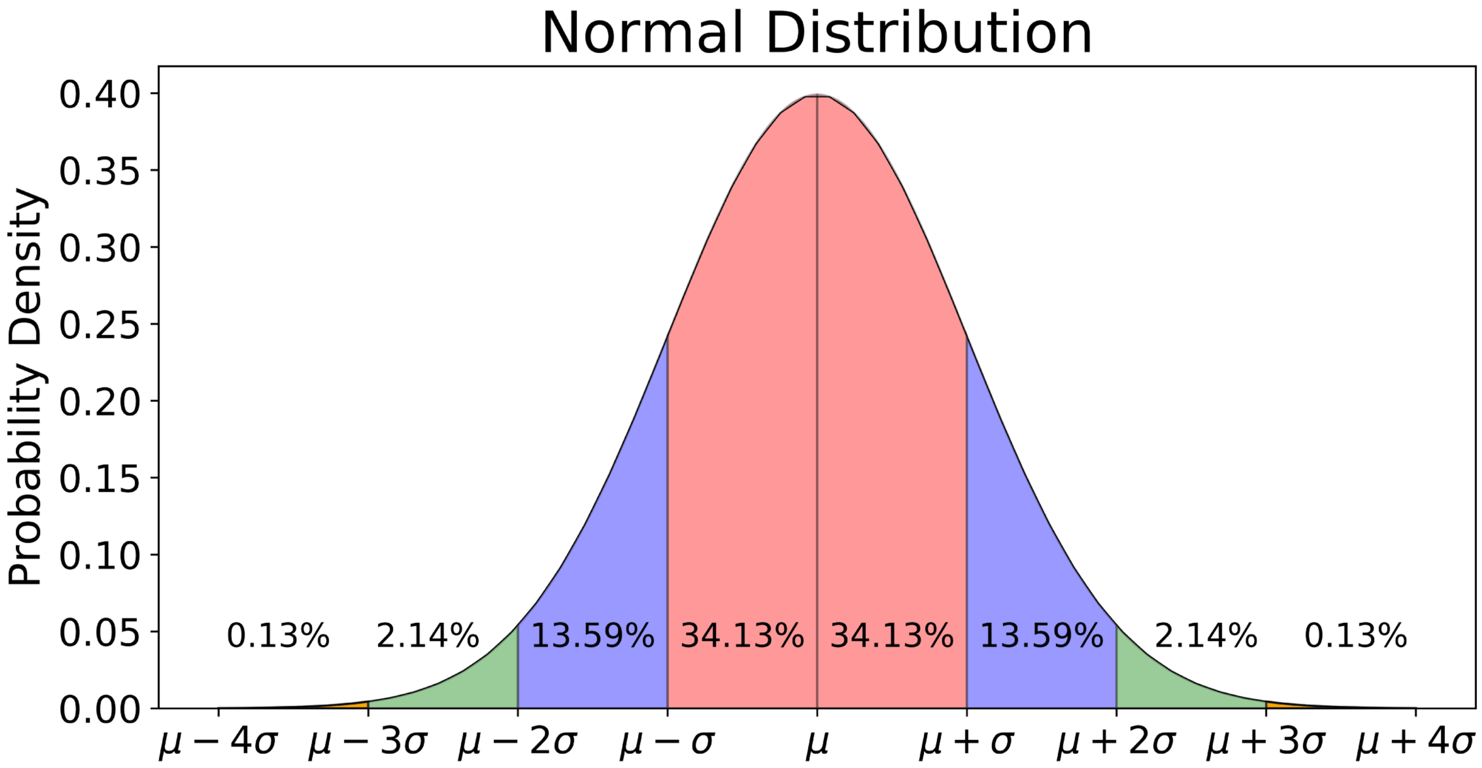
With `rbinom()` we can tell R to toss coins for us and thereby simulate rows from the possibility table instead of drawing the entire table. By doing this very often we will resemble the actual possibility table very closely.

→ A huge advantage is that we can even do this when we would not know exactly how to write down the possibility table!

Ok enough with the coin tossing! Let us look at more interesting examples.

Simulating data and introduction to Power Simulation

Similar to **`rbinom()`** there is a function called **`rnorm()`** that lets us simulate observations from a Normal Distribution.



Simulating data and introduction to Power Simulation

Let us **assume we want to compare the following two groups using a t-test**

Group 1 : expected population mean = 0, population SD = 5

Group 2: expected population mean = 2, population SD = 5

Imagine that Group 1 is a control group and Group 2 is a treatment group.

The scores for each person represent self-reported happiness following an intervention.

→ Let's see how we can simulate this situation. **Slide notes lines 258 - 302**

Table of Content

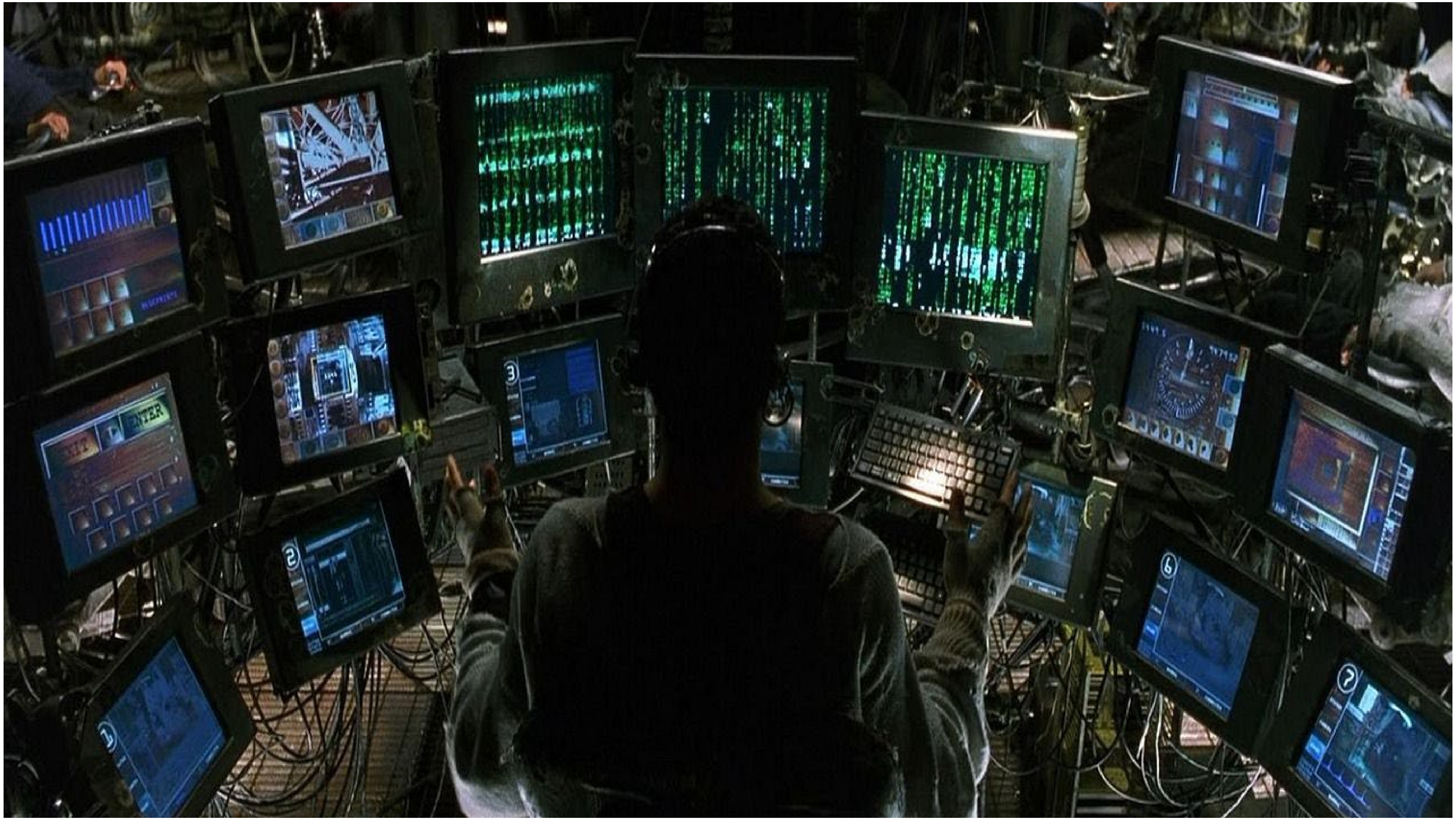
Part I

- ~~Why are we here?~~
- ~~Statistical testing and the counting of possibilities~~
- ~~Counting possibilities in R with probability functions and calculating power~~
- ~~Simulating data and introduction to Power Simulation~~

Part II

- **Practical One: Making use of Probability functions and Simulation functions**
- Simulating Linear models
- Practical Two: Simulating linear models
- Mixed-effects models: What else is there to consider
- Practical Three: Simulating mixed-effects models

Time for exercise 1



So far so good

- ☑ Binomial proportions
- ☑ Two-sample t-test

… 29 analyses to go!

This manual is not yet complete. We will be adding help on more tests in the future. If you cannot find help for your test in this version of the manual, then please check the G*Power website to see if a more up-to-date version of the manual has been made available.

Contents


ALWAYS HAS BEEN
WAIT, IT'S ALL
LINEAR MODELS?
Y=β0+β1*X+ϵ

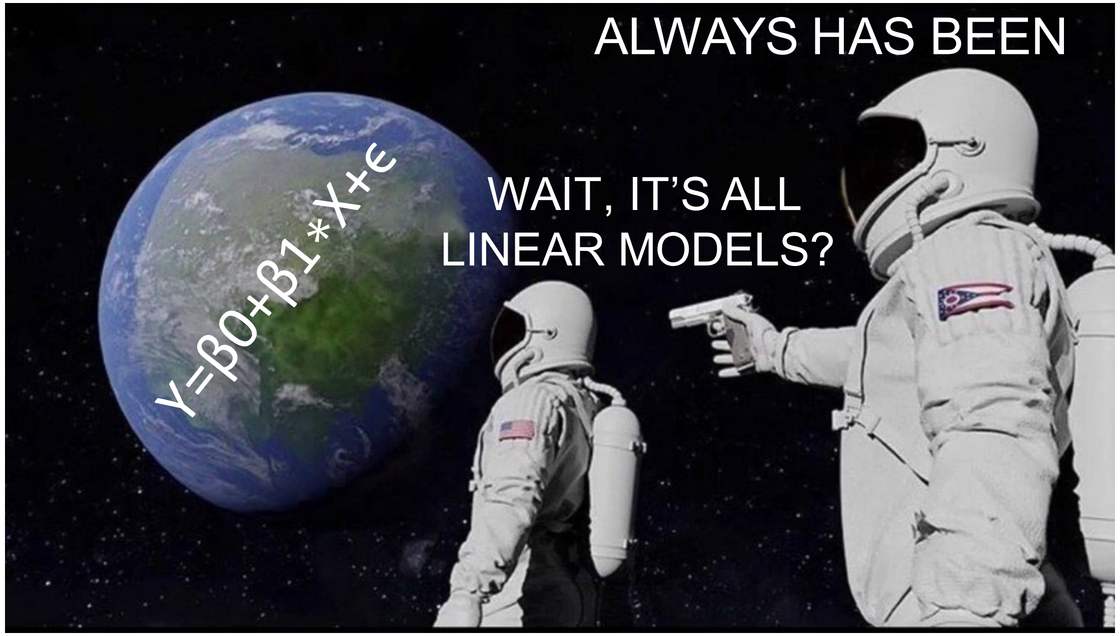
Simulating linear models

For instance, we could rewrite the t-test we just did as:

$$happiness_i = \beta_0 + \beta_1\, group_i + \epsilon_i$$

(happiness is a function of some average happiness and the group that you were assigned to in the study + an error)

→ This allows us to simulate most common research designs as linear (mixed models) in the same way.



Simulating linear models

Before we had two groups that we simulated separately

Group 1: m = 0, SD = 5 `rnorm(size = 30, mean = 0, sd = 5)`

Group 2: m = 2, SD = 5 `rnorm(size = 30, mean = 2, sd = 5)`

What we can do instead is to calculate the happiness score in each row based on the linear model formula

$$happiness_i = \boxed{\beta_0} + \boxed{\beta_1}\boxed{group_i} + \boxed{\epsilon_i}$$

If someone is in G2:
Happiness is calculated as the happiness of G1 + the difference between the groups

Slide notes lines 302 - 377

Time for exercise 2

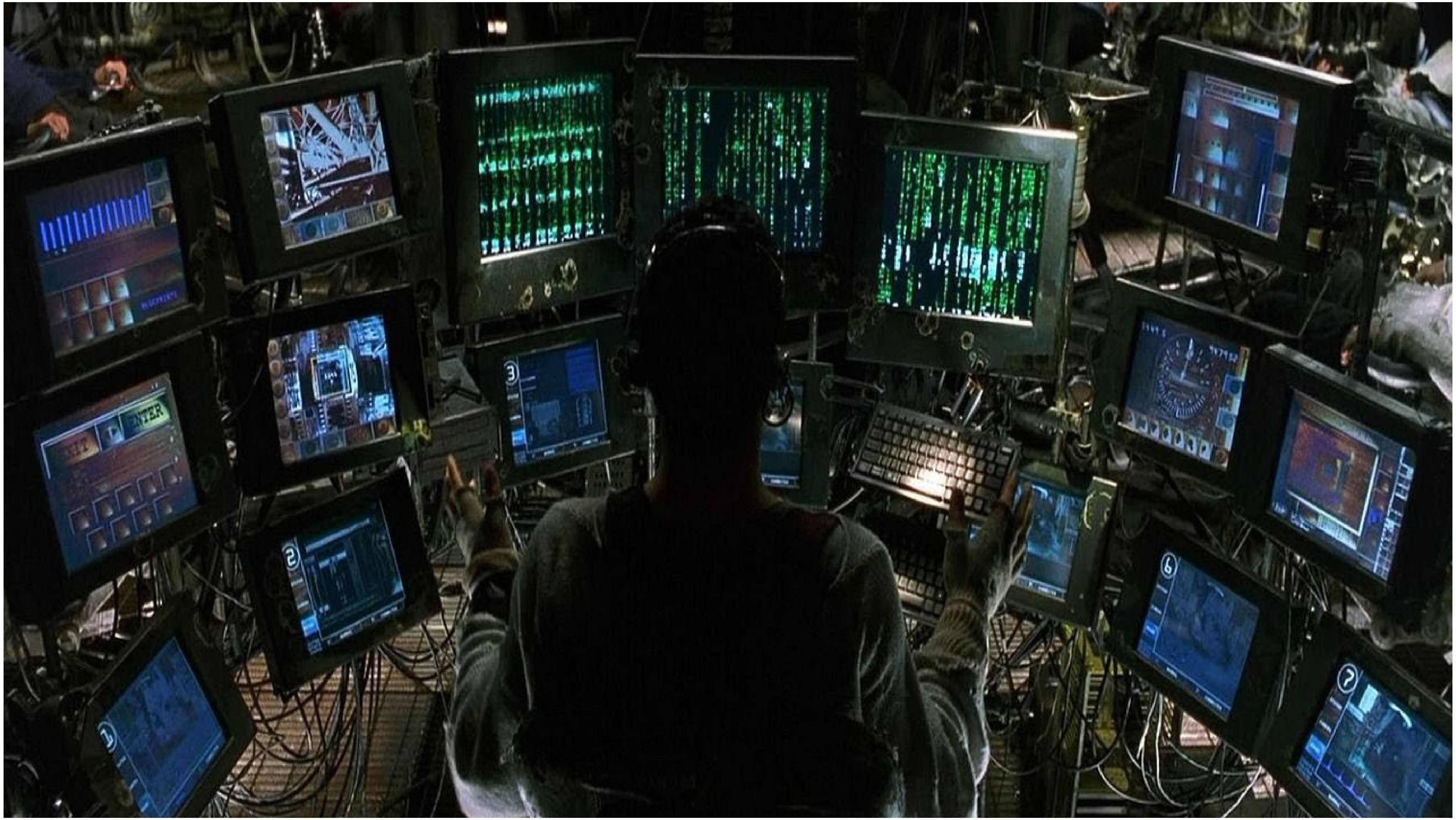


Table of Content

Part I

- ~~Why are we here?~~
- ~~Statistical testing and the counting of possibilities~~
- ~~Counting possibilities in R with probability functions and calculating power~~
- ~~Simulating data and introduction to Power Simulation~~

Part II

- ~~Practical One: Making use of Probability functions and Simulation functions~~
- ~~Simulating Linear models~~
- ~~Practical Two: Simulating linear models~~
- Mixed-effects models: What else is there to consider
- Practical Three: Simulating mixed-effects models

Interim Summary

We have seen that we can use the linear model formula to simulate all kinds of analyses like t-tests and even mixed ANOVA / mixed models.

However, we have not yet discussed simulating random effects – we will do this next…

Simulating Random Effects

In Exercise 2D, we tried to fit a mixed-model on the data and got a singular fit warning.

<u>→ Any ideas why we might git this warning?</u>

We tried to estimate random intercept for participants while not simulating one. Thus, it is 0. Random intercept parameters are **standard deviations** which cannot be smaller than 0. Thus, it is at the **boundary** of possible values.

Using the regression equation, random intercepts are relatively easy to add to a simulation

$$y_{ij} = \beta_0 + \boxed{u_{0j}} + \beta_1 \, group_i + \epsilon_i$$

For each participant 1 to j, add an offset (i.e. participant-specific number) to the simulation

Slide notes lines 384 - 483

Simulating Random Effects

$$y_{ij} = \beta_0 + \boxed{u_{0j}} + \beta_1 \, group_i + \epsilon_i$$

For each participant 1 to j, add an offset (i.e. participant-specific number) to the simulation

$$y_{ij} = \beta_0 + u_{0j} + (\beta_1 + \boxed{u_{1j}}) * group_i + \epsilon_i$$

For each participant 1 to j, add an offset (i.e. participant-specific number) to the simulation depending on the timepoint! You can see this as participants being of varying susceptibility to the effect of time

Slide notes lines 386 - 528

Last but not least, let’s have a look at a full mixed-effects model power simulation.

Slide notes lines 528 - 617

Time for exercise 3

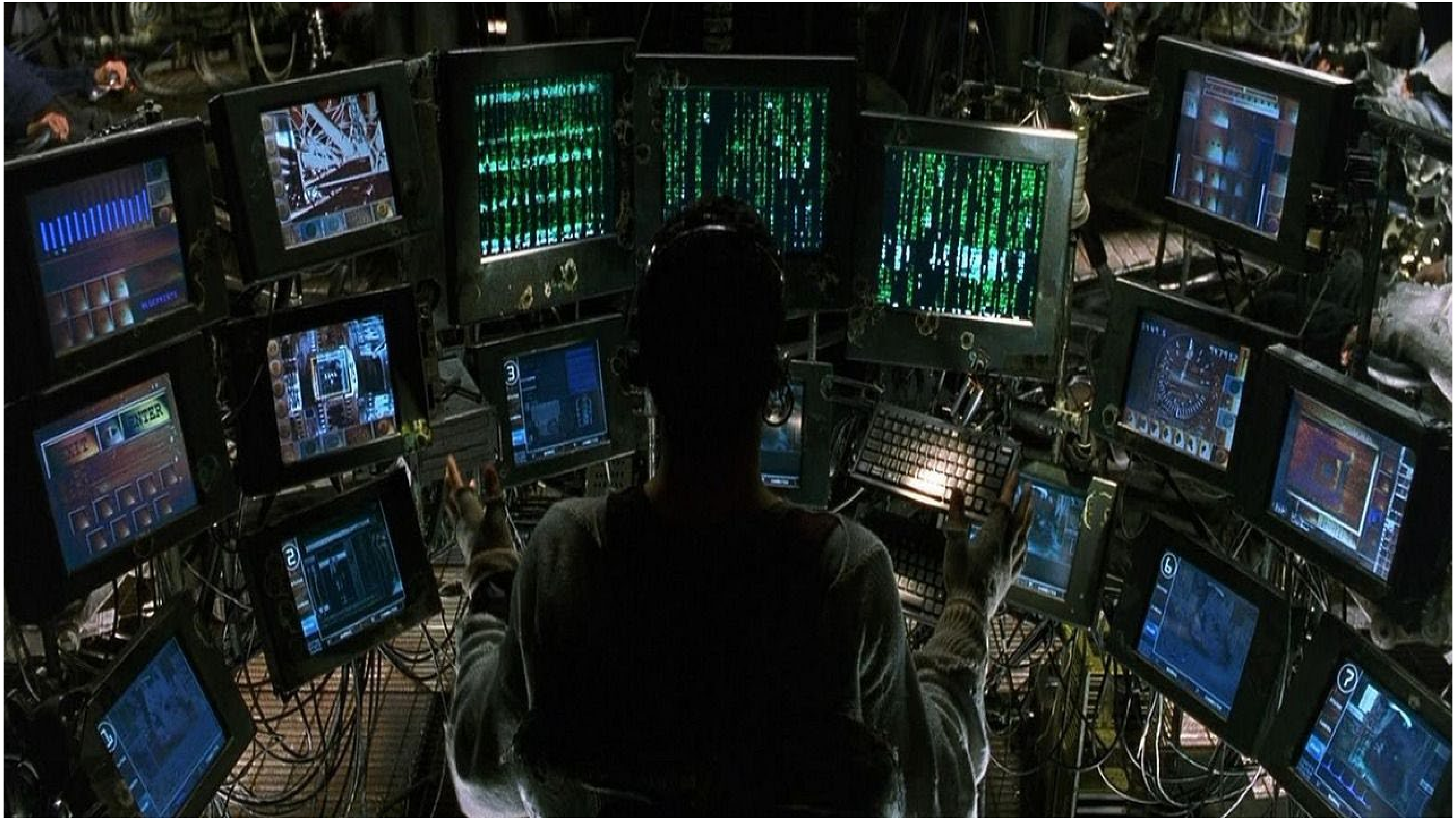


Table of Content

Part I

- ~~Why are we here?~~
- ~~Statistical testing and the counting of possibilities~~
- ~~Counting possibilities in R with probability functions and calculating power~~
- ~~Simulating data and introduction to Power Simulation~~

Part II

- ~~Practical One: Making use of Probability functions and Simulation functions~~
- ~~Simulating Linear models~~
- ~~Practical Two: Simulating linear models~~
- ~~Mixed-effects models: What else is there to consider~~
- ~~Practical Three: Simulating mixed-effects models~~

Table of Content

Words of advice

There is a LOT that we did not cover yet.

1. **Correlations**

→Random effects, and everything really, are **correlated** (see the random correlations in the output) but we did not put this in our simulations yet. This is something you should definitely look into before simulating mixed-model data.

2. **Contrast coding**

→For now, we coded time as 0 to 10 for example, but normally we would center it, which would have impact on intercept interpretation. Moreover, groups should be sum-to-zero coded if interactions are present, or everything should be coded directly according to hypotheses and expectations.

3. Interactions

→Partly related to the contrast coding point, but few other things to consider.

4. More complex models: Crossed random effects, nested random effects

5. Model families (though we did simulate binomial data so you can do that!)

6. Distributional properties

→What if we have bounded scales (e.g. response times that cannot be smaller than zero) or if we expect skewed data etc.

Words of advice

→I discuss most of these things in detail in my online tutorial.

If, after this workshop, you think data simulation can be helpful to you, I strongly encourage reading it (total reading time without running code about 300 minutes). You will know many things already but also learn more details about most things we discussed and learn the points on the previous slide (mostly in part IV).

Last but not least:

There is already other R packages to make simulations easier, e.g. faux or simR.

These are very good, and make many things much easier than writing everything up by hand.