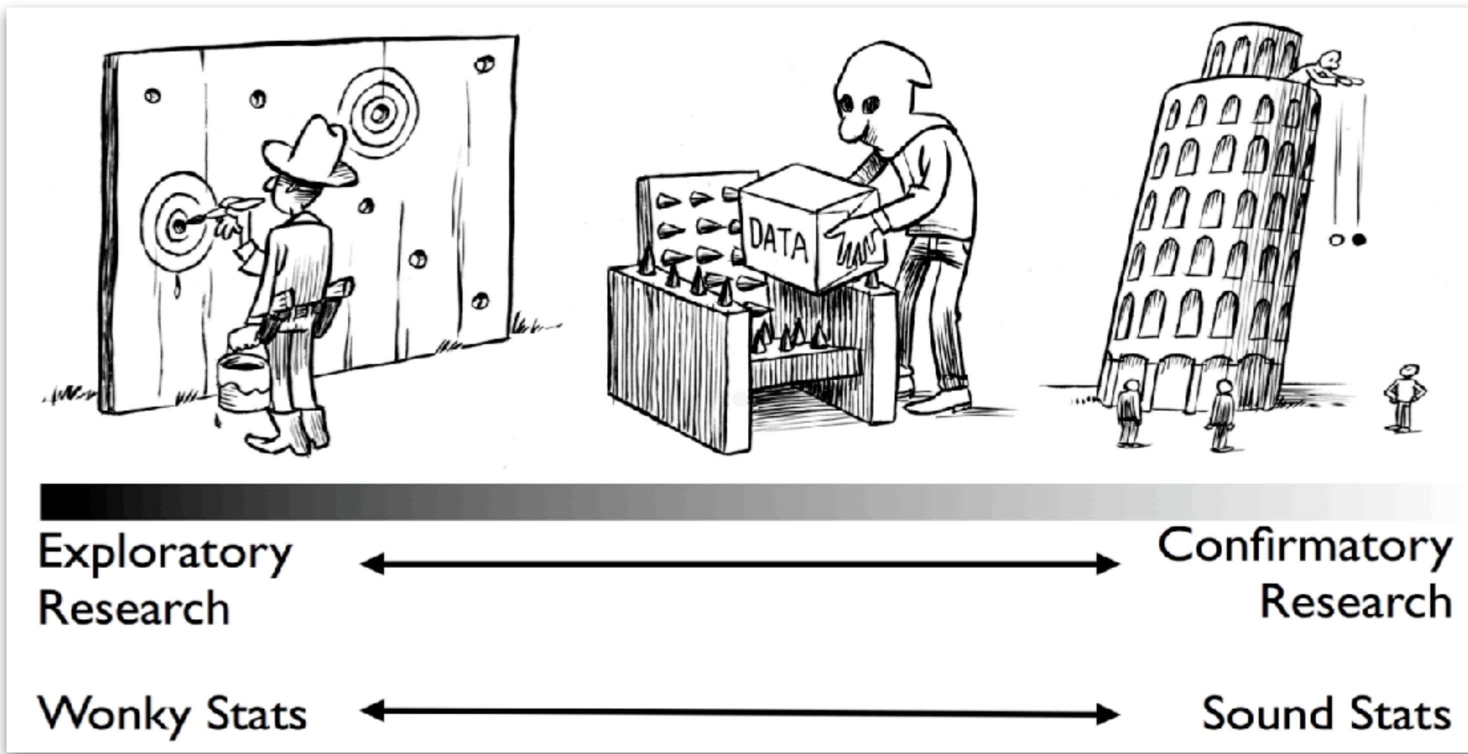
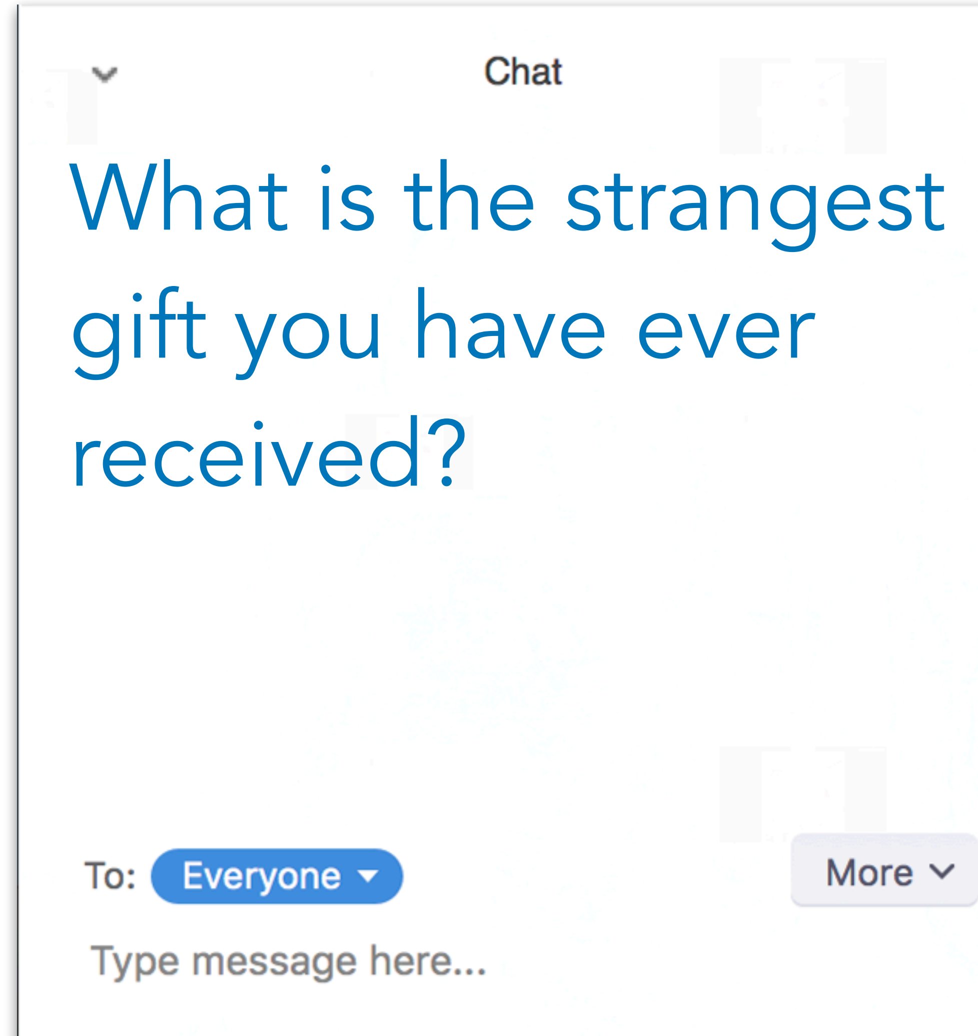
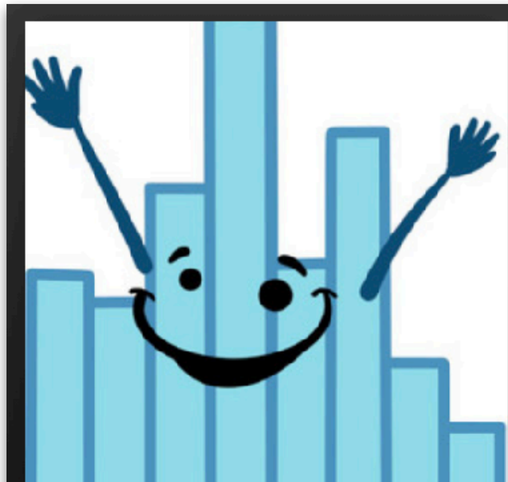
PSYCH 252: Statistical Methods for Behavioral and Social Sciences

Power analysis







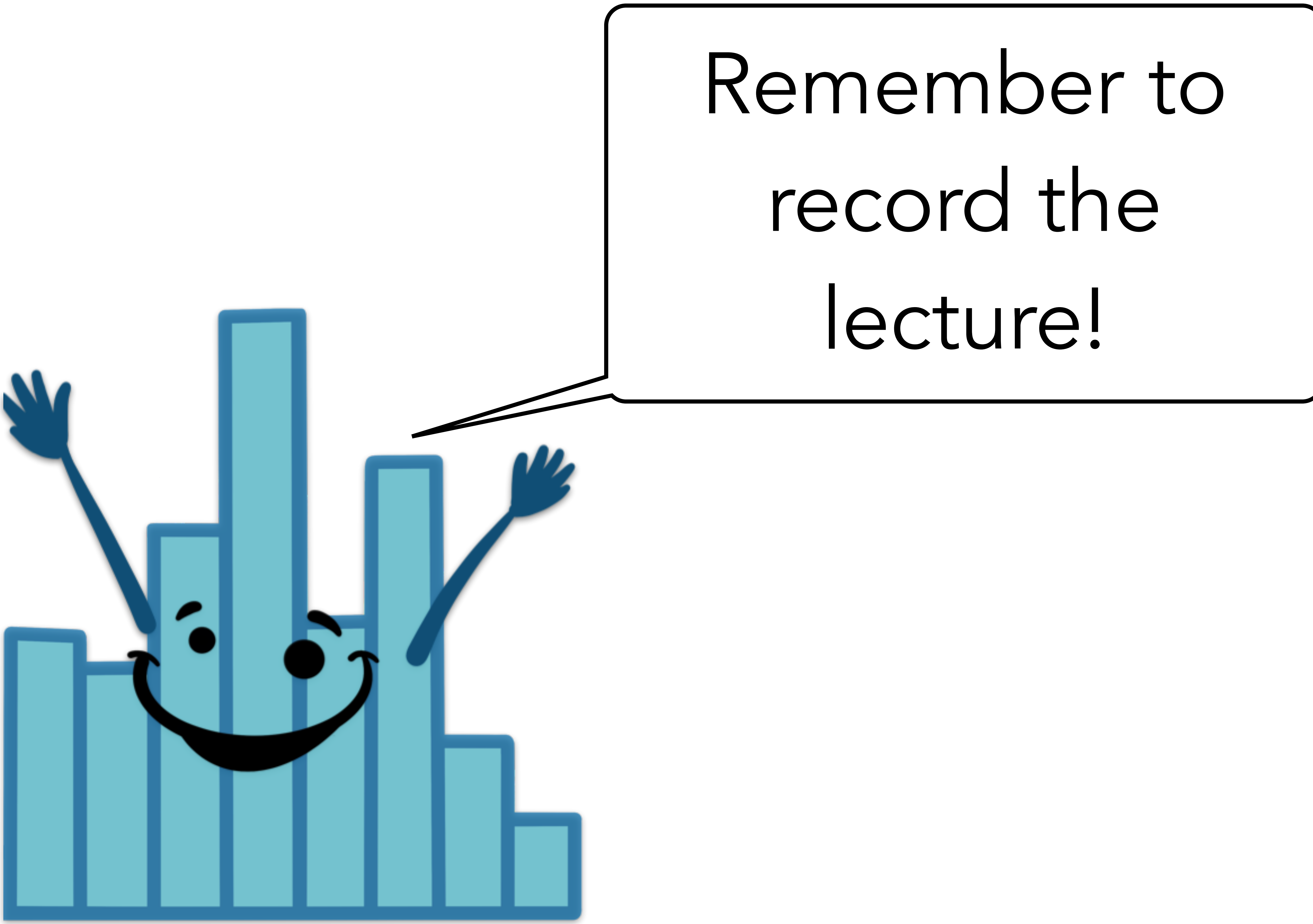
COLLABORATIVE PLAYLIST

psych252

https://tinyurl.com/psych252spotify21

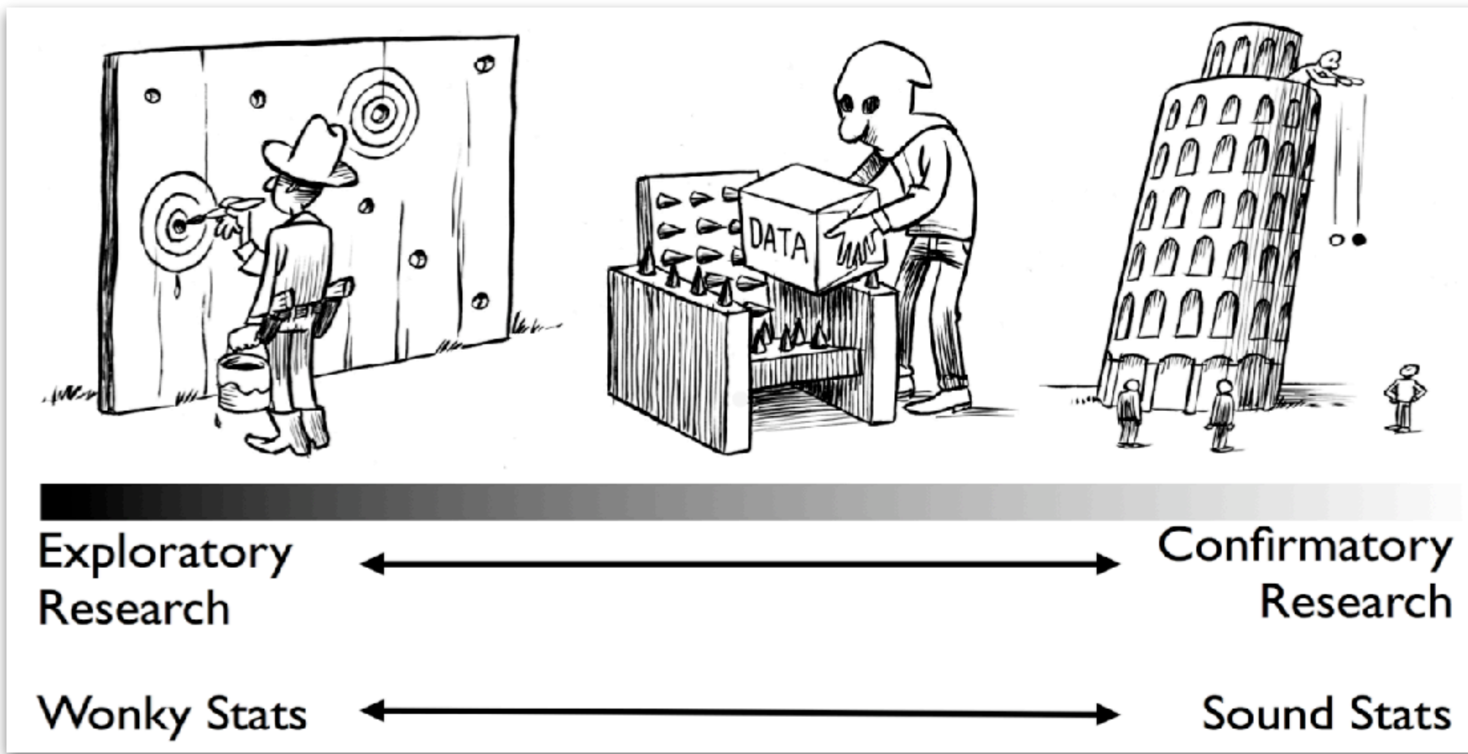
PLAY

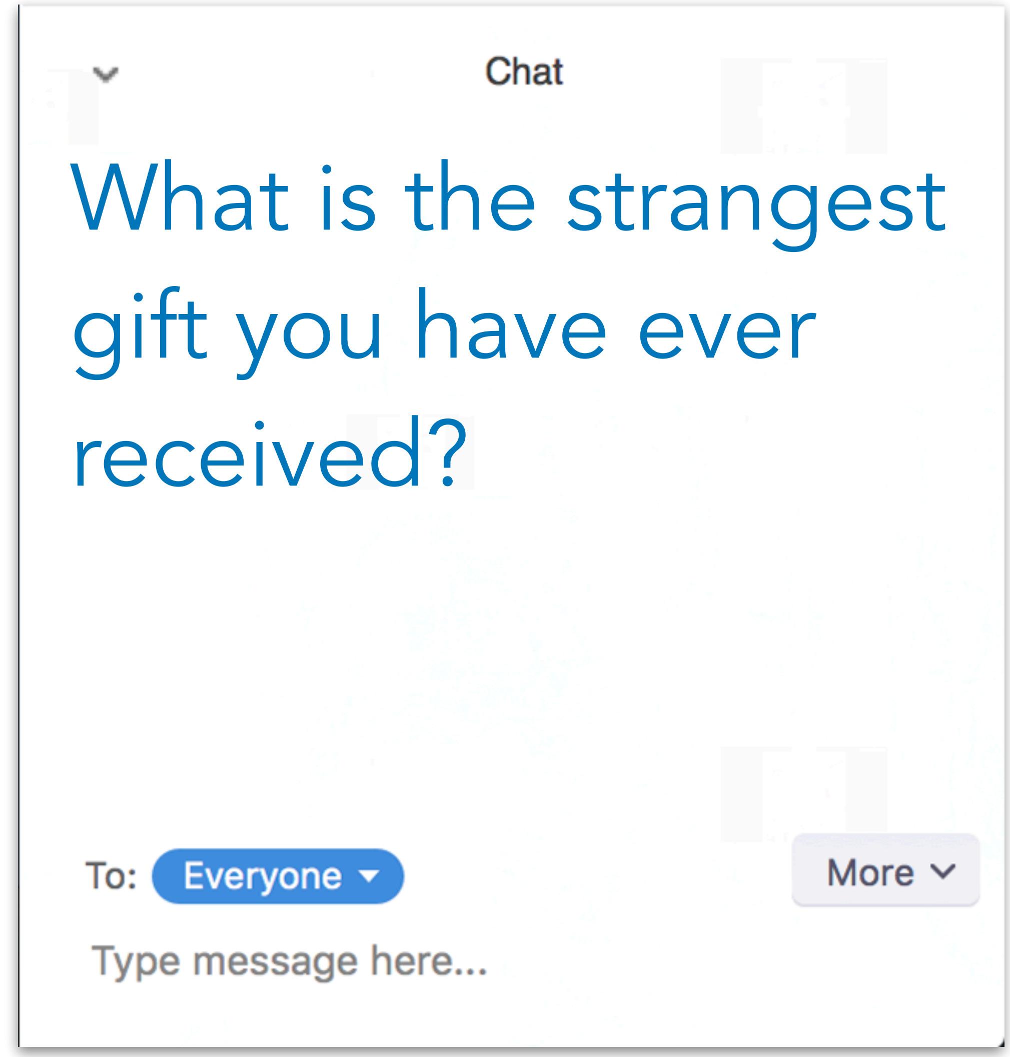
02/12/2021

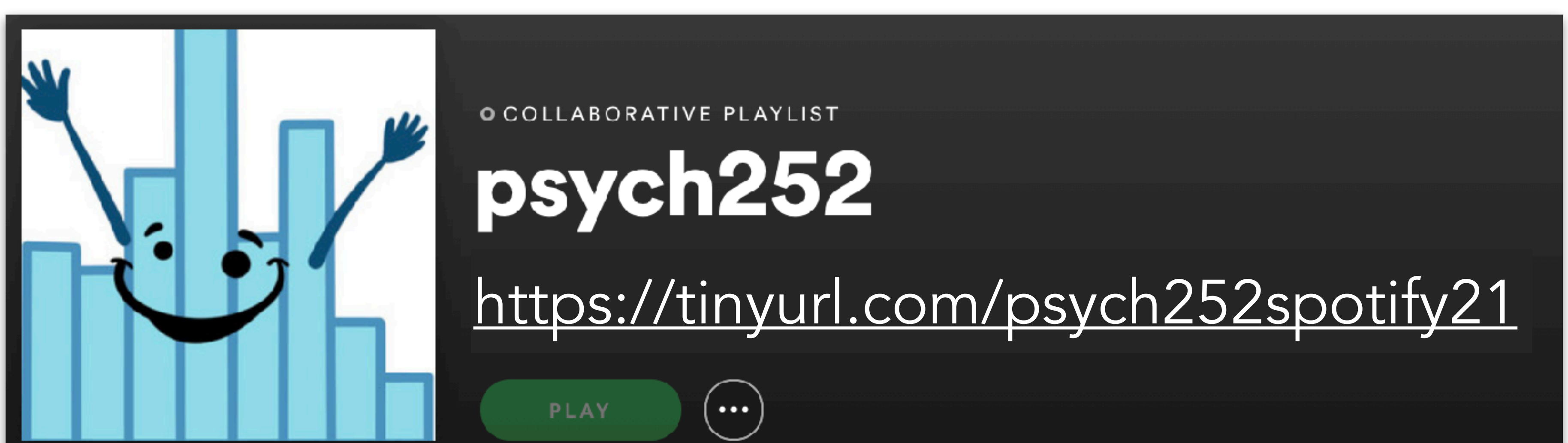

Remember to record the lecture!

PSYCH 252: Statistical Methods for Behavioral and Social Sciences

Power analysis







02/12/2021

Things that came up

Go! Go! Linear model

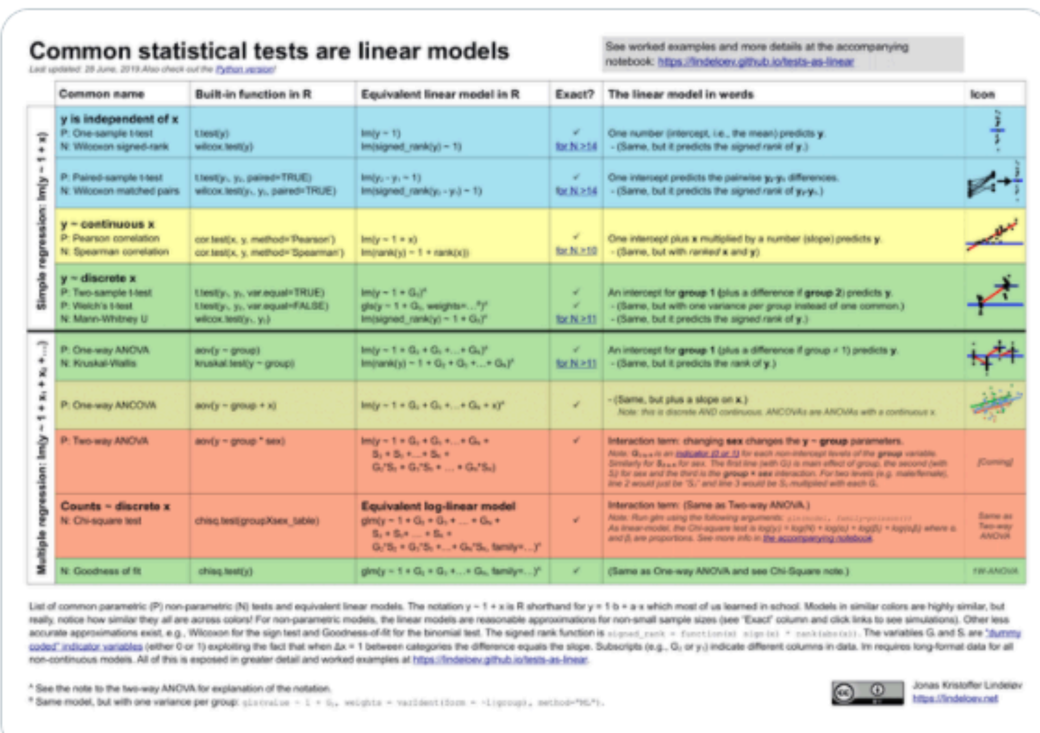
Indrajeet Patil
@patilindrajeets

Your periodic reminder that the most common statistical tests are nothing but special cases of linear models 🧐

lindeloev.github.io/tests-as-linea...

I wish someone had pointed out this beautiful simplicity and parsimony when I was learning statistics 🤷‍♂️

#rstats



Jonas K. Lindeløv

12:04 AM · Feb 12, 2021 · Twitter Web App

102 Retweets **6** Quote Tweets **373** Likes

Common statistical tests are linear models

Last updated: 28 June, 2019.Also check out the Python version!

See worked examples and r
notebook: https://lindeloev.g

	Common name	Built-in function in R	Equivalent linear model in R	Exact?	The linear model in word
Simple regression: lm(y ~ 1 + x)	**y is independent of x** P: One-sample t-test N: Wilcoxon signed-rank	t.test(y) wilcox.test(y)	lm(y ~ 1) lm(signed_rank(y) ~ 1)	✓ for N >14	One number (intercept, i.e., the - (Same, but it predicts the *sig*
	P: Paired-sample t-test N: Wilcoxon matched pairs	t.test(y_1, y_2, paired=TRUE) wilcox.test(y_1, y_2, paired=TRUE)	lm(y_2 - y_1 ~ 1) lm(signed_rank(y_2 - y_1) ~ 1)	✓ for N >14	One intercept predicts the pairv - (Same, but it predicts the *sig*
	y ~ continuous x P: Pearson correlation N: Spearman correlation	cor.test(x, y, method='Pearson') cor.test(x, y, method='Spearman')	lm(y ~ 1 + x) lm(rank(y) ~ 1 + rank(x))	✓ for N >10	One intercept plus **x** multiplied - (Same, but with *ranked* **x** and
	y ~ discrete x P: Two-sample t-test P: Welch's t-test N: Mann-Whitney U	t.test(y_1, y_2, var.equal=TRUE) t.test(y_1, y_2, var.equal=FALSE) wilcox.test(y_1, y_2)	lm(y ~ 1 + G_2)[A] gls(y ~ 1 + G_2, weights=...[B])[A] lm(signed_rank(y) ~ 1 + G_2)[A]	✓ ✓ for N >11	An intercept for **group 1** (plus a - (Same, but with one variance - (Same, but it predicts the *sig*
Multiple regression: lm(y ~ 1 + x_1 + x_2 +...)	P: One-way ANOVA N: Kruskal-Wallis	aov(y ~ group) kruskal.test(y ~ group)	lm(y ~ 1 + G_2 + G_3 +...+ G_N)[A] lm(rank(y) ~ 1 + G_2 + G_3 +...+ G_N)[A]	✓ for N >11	An intercept for **group 1** (plus a - (Same, but it predicts the *ran*
	P: One-way ANCOVA	aov(y ~ group + x)	lm(y ~ 1 + G_2 + G_3 +...+ G_N + x)[A]	✓	- (Same, but plus a slope on **x**. *Note: this is discrete AND continu*
	P: Two-way ANOVA	aov(y ~ group * sex)	lm(y ~ 1 + G_2 + G_3 +...+ G_N + S_2 + S_3 +...+ S_K + G_2*S_2 + G_3*S_3 + ... + G_N*S_K)	✓	Interaction term: changing **sex** *Note: $G_{2 \text{ to } N}$ is an indicator (0 or 1) f* *Similarly for $S_{2 \text{ to } K}$ for sex. The first l* *S_j) for sex and the third is the* ***group*** *line 2 would just be "S_2" and line 3 w*
	Counts ~ discrete x N: Chi-square test	chisq.test(groupXsex_table)	**Equivalent log-linear model** glm(y ~ 1 + G_2 + G_3 + ... + G_N + S_2 + S_3+ ... + S_K + G_2*S_2 + G_3*S_3 +...+ G_N*S_K, family=...)[A]	✓	Interaction term: (Same as Two *Note: Run glm using the following a* *As linear-model, the Chi-square tes* *and β_j are proportions. See more in*
	N: Goodness of fit	chisq.test(y)	glm(y ~ 1 + G_2 + G_3 +...+ G_N, family=...)[A]	✓	(Same as One-way ANOVA and

List of common parametric (P) non-parametric (N) tests and equivalent linear models. The notation y ~ 1 + x is R shorthand for y = 1·b + a·x which most of us learn
really, notice how similar they *all* are across colors! For non-parametric models, the linear models are reasonable approximations for non-small sample sizes (see "
accurate approximations exist, e.g., Wilcoxon for the sign test and Goodness-of-fit for the binomial test. The signed rank function is `signed_rank = function(x)`
coded" indicator variables (either 0 or 1) exploiting the fact that when Δx = 1 between categories the difference equals the slope. Subscripts (e.g., G_2 or y_1) indicate
non-continuous models. All of this is exposed in greater detail and worked examples at https://lindeloev.github.io/tests-as-linear.

[A] See the note to the two-way ANOVA for explanation of the notation.
[B] Same model, but with one variance per group: `gls(value ~ 1 + G2, weights = varIdent(form = ~1|group), method="ML")`.

https://lindeloev.github.io/tests-as-linear/

Excellent online book!

An Introduction to Data Analysis

Michael Franke

last rendered at: 2021-01-26 08:56:45

Preface

This book provides basic reading material for an introduction to data analysis. It uses R to handle, plot and analyze data. After covering the use of R for data wrangling and plotting, the book introduces key concepts of data analysis from a Bayesian and a frequentist tradition. This text is intended for use as a first introduction to statistics for an audience with some affinity towards programming, but no prior exposition to R.

Many people have supported this project actively by providing text, examples, code or technical support. Many thanks for their support to (in alphabetic order): Tobias Anton, Florence Bockting, Noa Kallioinen, Minseok Kang, Marcel Klehr, Özge Özenoglu, Maria Pershina, Timo Roettger, Polina Tsvilodub and Inga Wohlert.

https://michael-franke.github.io/intro-data-analysis/index.html

Logistics

Midterm

will be available shortly after class today

Psych 252 Midterm

My name goes here

2021-02-12 12:04:21

Introduction

This is a take-home exam. The exam is open notes and open book (in short, you can use any source of information you like as long as you work on the exam by yourself). The maximum score is 120 points. Please adhere to the honor code. Submit the midterm as a PDF on the canvas 'midterm' assignment by **Thursday, February 18th, 8pm**.

The late policy submission policy is:

- We will subtract 2% from your points for each hour that the midterm is submitted late but before midnight. For example, 2% will be subtracted if you submit between 8pm and 9pm, or 8% if you submit between 11pm and midnight.
- 20% will be subtracted if you submit after midnight on Thursday but before 8pm on Friday, February 19th.
- No points will be granted if you submit later than 8pm on Friday, February 19th.

For questions that require written responses, please make sure to show any relevant tables, summaries (e.g. from `lm()` or `anova()`), or visualizations. Some of the code chunks have existing code that you can use to build your code around.

When asked to report results, please do so like you would in a scientific article (see examples from class).

- Please leave the `\clearpage` commands where they are. This makes sure that each question is printed on a separate page in the pdf.
- Some code chunks are set to `eval=F`, make sure to set these to `eval=T` before knitting the final version.
- We note for each question how many points you can get. You can get up to 120 points in total.
- Good coding style matters! We will add or subtract up to 5 points depending on style.

If you have any questions about the midterm, please post them on Piazza addressed to the instructors only. We will answer your question and may choose to share both your question and our answer with the rest of the group.

Best of luck with the midterm!

Honor Code

The Honor Code is the University's statement on academic integrity written by students in 1921. It articulates University expectations of students and faculty in establishing and maintaining the highest standards in academic work:

1. The Honor Code is an undertaking of the students, individually and collectively:

a. that they will not give or receive aid in examinations; that they will not give or receive unpermitted aid in class work, in the preparation of reports, or in any other work that is to be used by the instructor as the basis of grading;
b. that they will do their share and take an active part in seeing to it that others as well as themselves uphold the spirit and letter of the Honor Code.

Part 1: Wash your hands!

A student investigated how effective different methods are for eliminating bacteria. She tested four different methods: (1) washing her hands with water only, (2) with regular soap, (3) with antibacterial soap (ABS), or (4) using an antibacterial spray (AS). She suspected that the number of bacteria on her hands might vary considerably from day to day. To account for this, she generated random numbers to determine on what day she would use which treatment. After each treatment, she placed her right hand on a sterile media plate to measure bacteria growth. She incubated each plate for 2 days after which she counted the bacteria colonies. She replicated this procedure 9 times for each of the four treatments.

> Note: For statistical analysis purposes, we make the assumption that the individual measurements are independent from each other.

Part 2: Life satisfaction

In this exercise, we are interested in seeing what affects life satisfaction. We have a (fake) data set with the following variables:

Table 1: Variables in the satisfaction data set.

variable	description
`id`	participant id
`age`	age in years
`kids`	number of kids
`jobsatis`	job satisfaction (1 = not at all, 7 = very much)
`marsatis`	marital satisfaction (1 = not at all, 7 = very much)
`lifsatis`	life satisfaction (1 = not at all, 7 = very much)

Part 3: You've got the power!

In this exercise, we'll take a look at determining what group sample size we would need in order to achieve adequate statistical power to test our research hypothesis of interest. We will be using the data from "data/power.csv" and you can see its visualization in "figure/df_power.png".

> Tip: We will provide saved checkpoints for each problem. Feel free to look ahead to compare your results with ours.

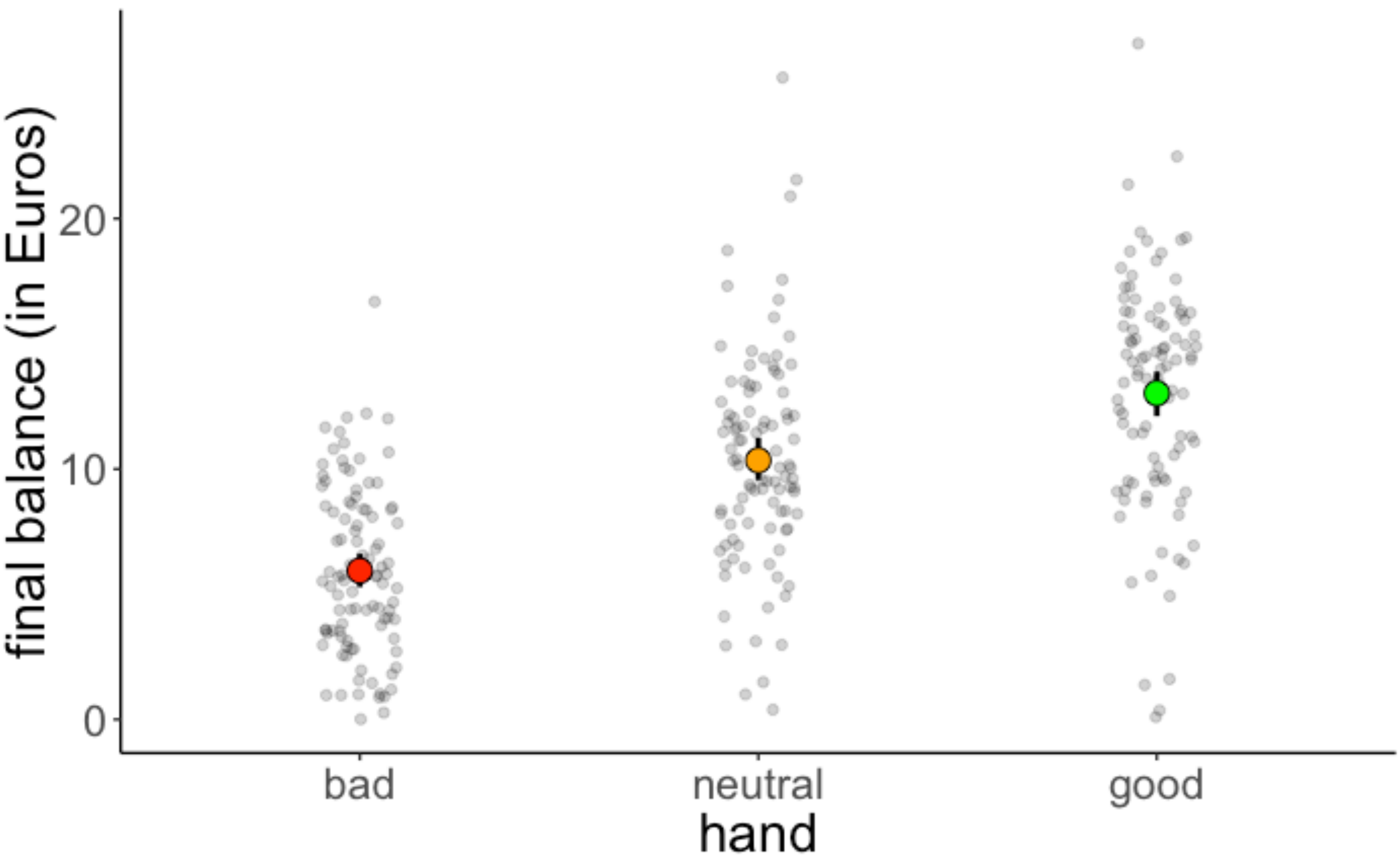
Plan for today

- Linear contrasts
  - Testing specific hypotheses with linear contrasts
  - `emmeans` for handling linear contrasts in R
- Power analysis
  - Making decisions
  - Calculating power
  - Effect sizes
  - Determining sample size
- Learn about more advanced simulation techniques in R
  - `map()`
  - list columns: `nest()`, `unnest()`

Linear contrasts

Testing (more) specific hypotheses with linear contrasts

Do better hands win more money?



Do better hands win more money?

ANOVA

Does card quality affect the final balance?

post-hoc tests

bad vs. **neutral**

neutral vs. **good**

Is there are more direct way of asking this question with a statistical model?

Contrasts

```
df.poker = df.poker %>%
  mutate(hand_contrast = factor(hand,
                                levels = c("bad", "neutral", "good"),
                                labels = c(-1, 0, 1)),
         hand_contrast = hand_contrast %>% as.character() %>% as.numeric())
```

participant	hand	balance	hand_contrast
1	bad	4.00	-1
2	bad	5.55	-1
3	bad	9.45	-1
51	neutral	11.74	0
52	neutral	10.04	0
53	neutral	9.49	0
101	good	10.86	1
102	good	8.68	1
103	good	14.36	1

Contrasts

```r
df.poker = df.poker %>%
  mutate(hand_contrast = factor(hand,
                                levels = c("bad", "neutral", "good"),
                                labels = c(-1, 0, 1)),
         hand_contrast = hand_contrast %>% as.character() %>% as.numeric())

fit = lm(formula = balance ~ hand_contrast,
         data = df.poker)
```

```
Call:
lm(formula = balance ~ hand_contrast, data = df.fit)

Residuals:
    Min      1Q  Median      3Q     Max
-13.214  -2.684  -0.019   2.444  15.858

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)     9.7715     0.2381   41.03   <2e-16 ***
hand_contrast   3.5424     0.2917   12.14   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.125 on 298 degrees of freedom
Multiple R-squared:  0.3311,	Adjusted R-squared:  0.3289
F-statistic: 147.5 on 1 and 298 DF,  p-value: < 2.2e-16
```

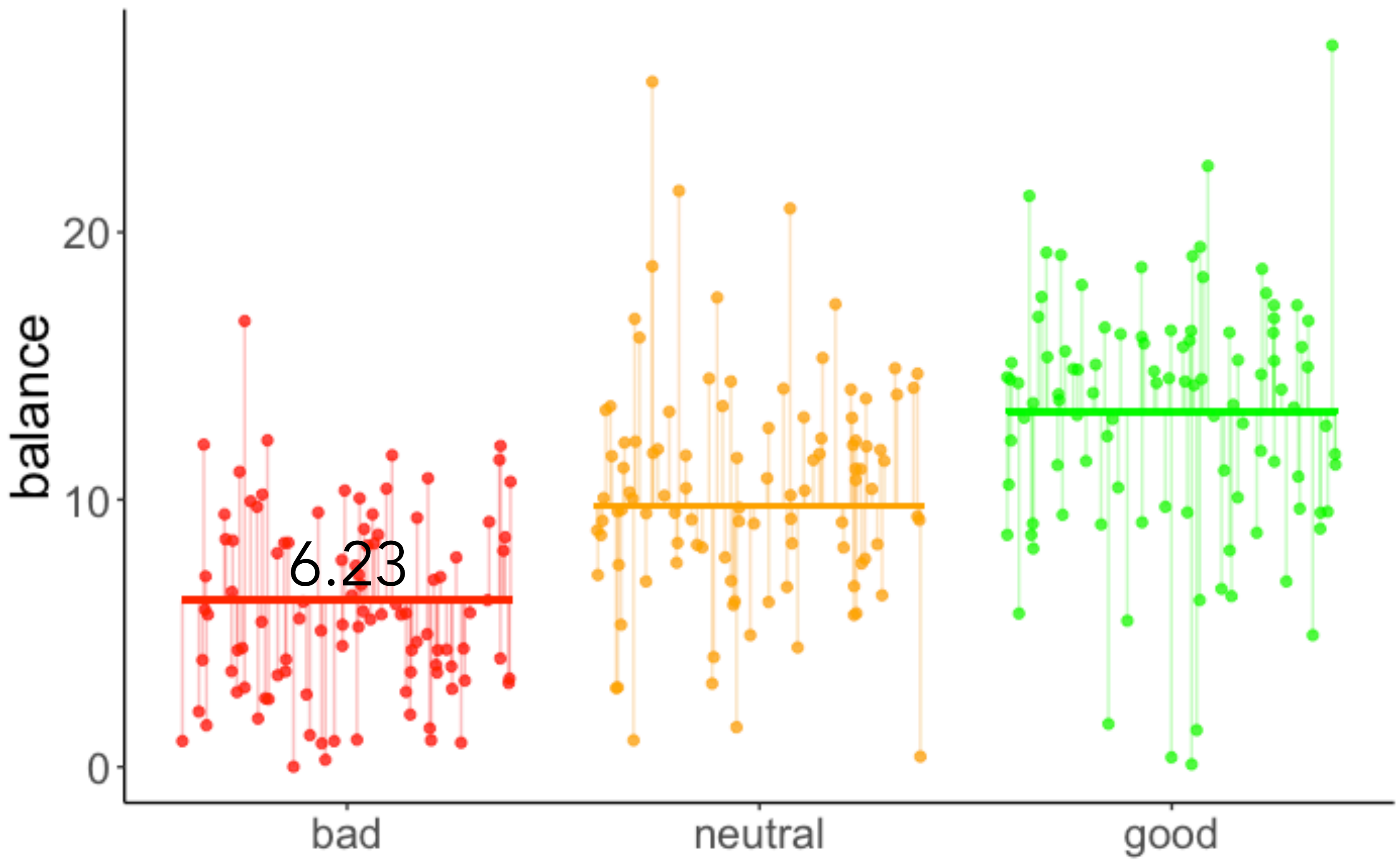
mean in neutral condition

significant contrast

Contrasts

name	estimate	std.error	statistic	p.value
intercept	9.77	0.24	41.03	0
contrast	3.54	0.29	12.15	0

$$\widehat{\text{balance}}_i = b_0 + b_1 \cdot \text{hand\_contrast}$$



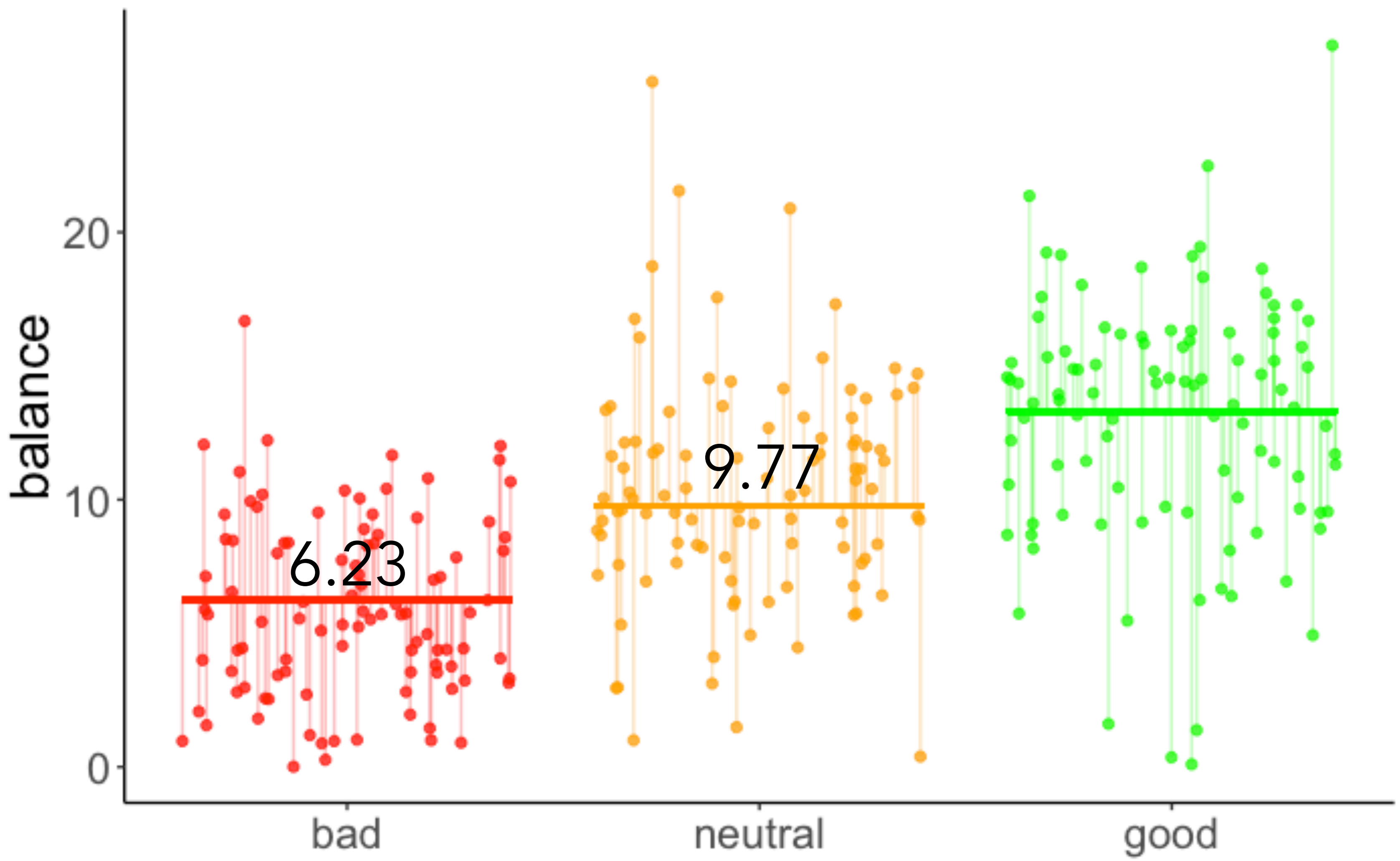
if contrast == -1

$$\widehat{\text{balance}}_i = b_0 + b_1 \cdot \text{hand\_contrast}_i$$
$$= 9.77 + (-1) \cdot 3.54 = 6.23$$

Contrasts

name	estimate	std.error	statistic	p.value
intercept	9.77	0.24	41.03	0
contrast	3.54	0.29	12.15	0

$$\widehat{\text{balance}}_i = b_0 + b_1 \cdot \text{hand\_contrast}$$



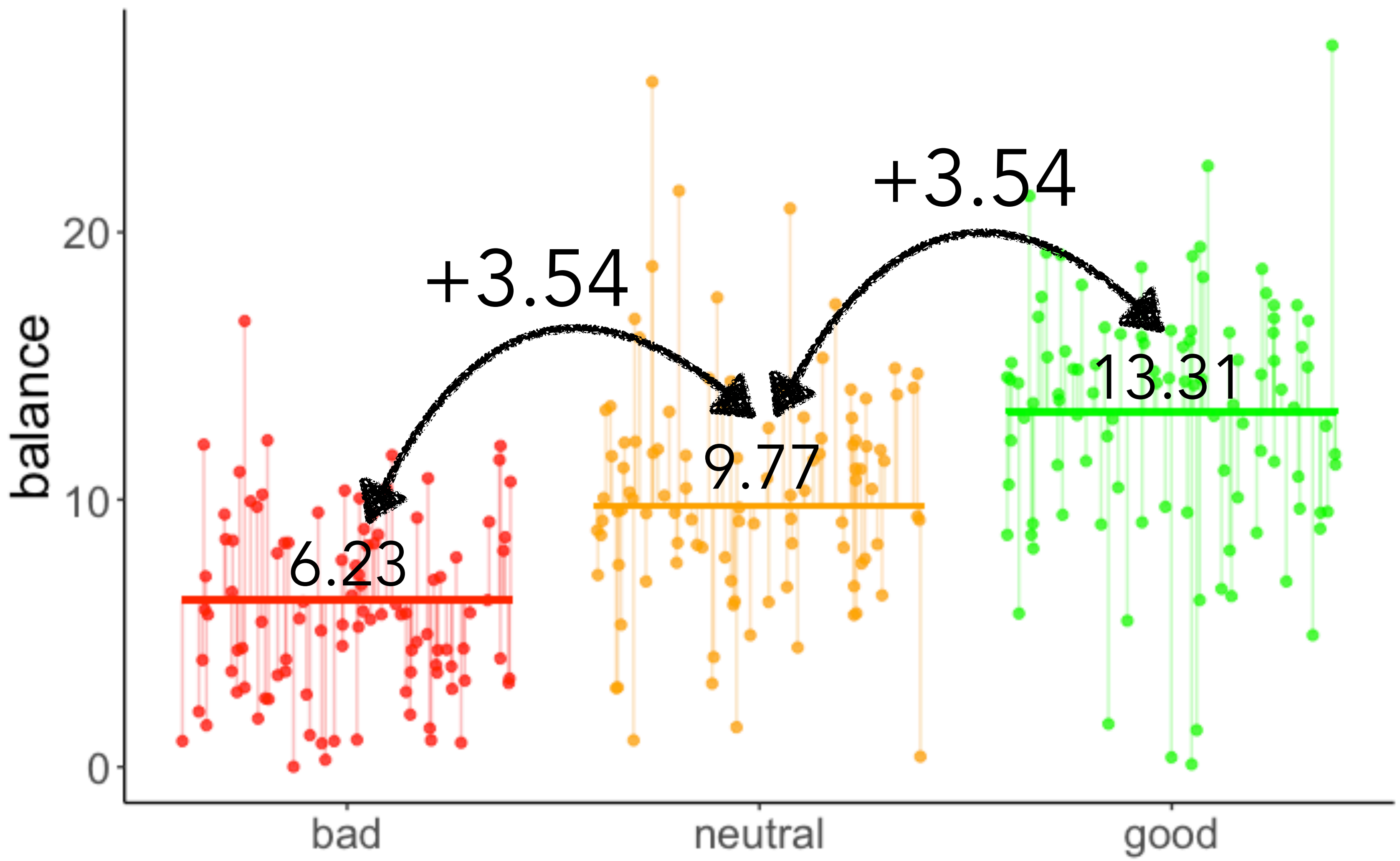
if contrast == 0

$$\widehat{\text{balance}}_i = b_0 + b_1 \cdot \text{hand\_contrast}_i$$
$$= 9.77 + 0 \cdot 3.54 = 9.77$$

Contrasts

name	estimate	std.error	statistic	p.value
intercept	9.77	0.24	41.03	0
contrast	3.54	0.29	12.15	0

$$\widehat{\text{balance}}_i = b_0 + b_1 \cdot \text{hand\_contrast}$$

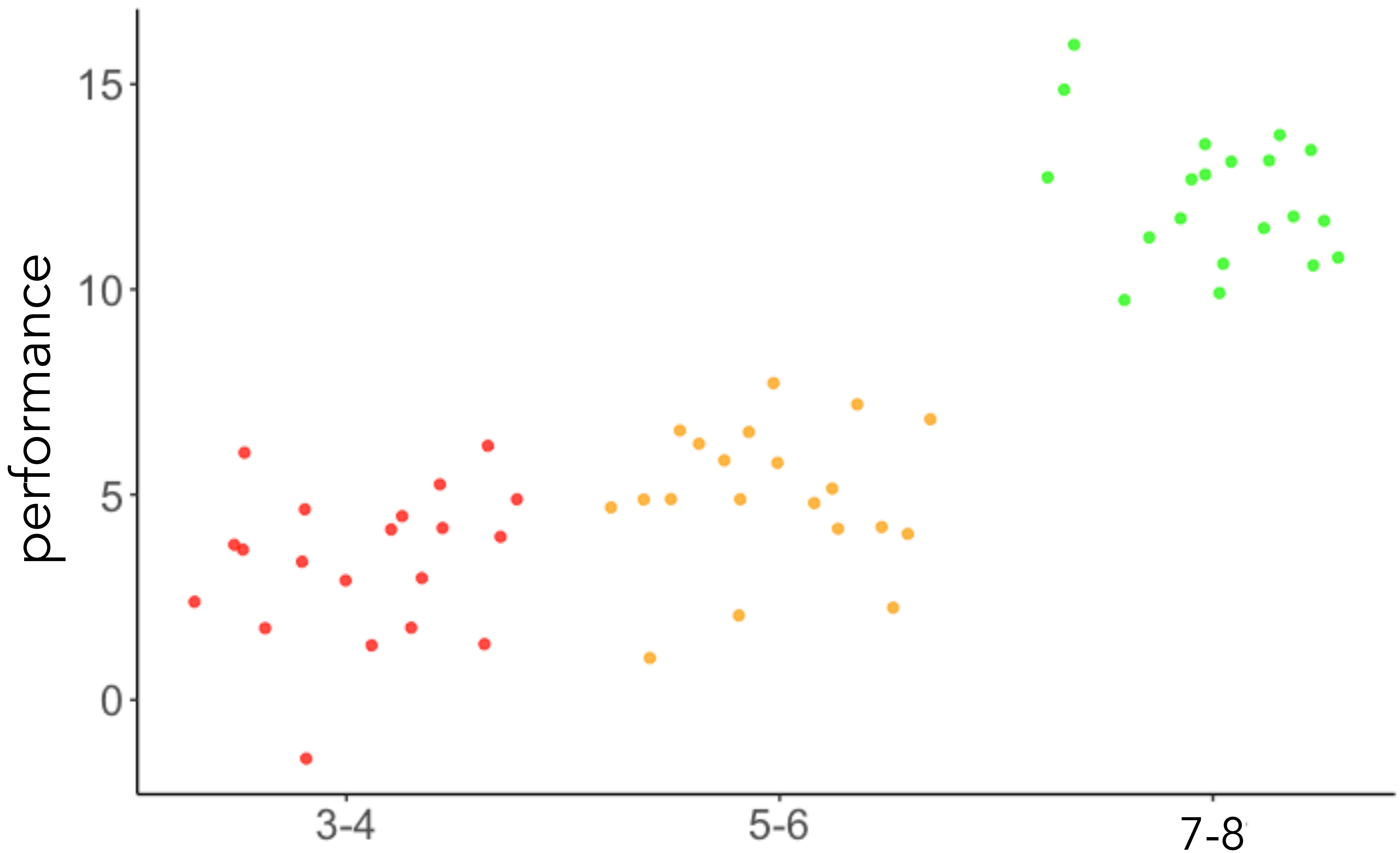


if contrast == 1

$$\widehat{\text{balance}}_i = b_0 + b_1 \cdot \text{hand\_contrast}_i$$
$$= 9.77 + 1 \cdot 3.54 = 13.31$$

Contrasts

Does performance increase with age?



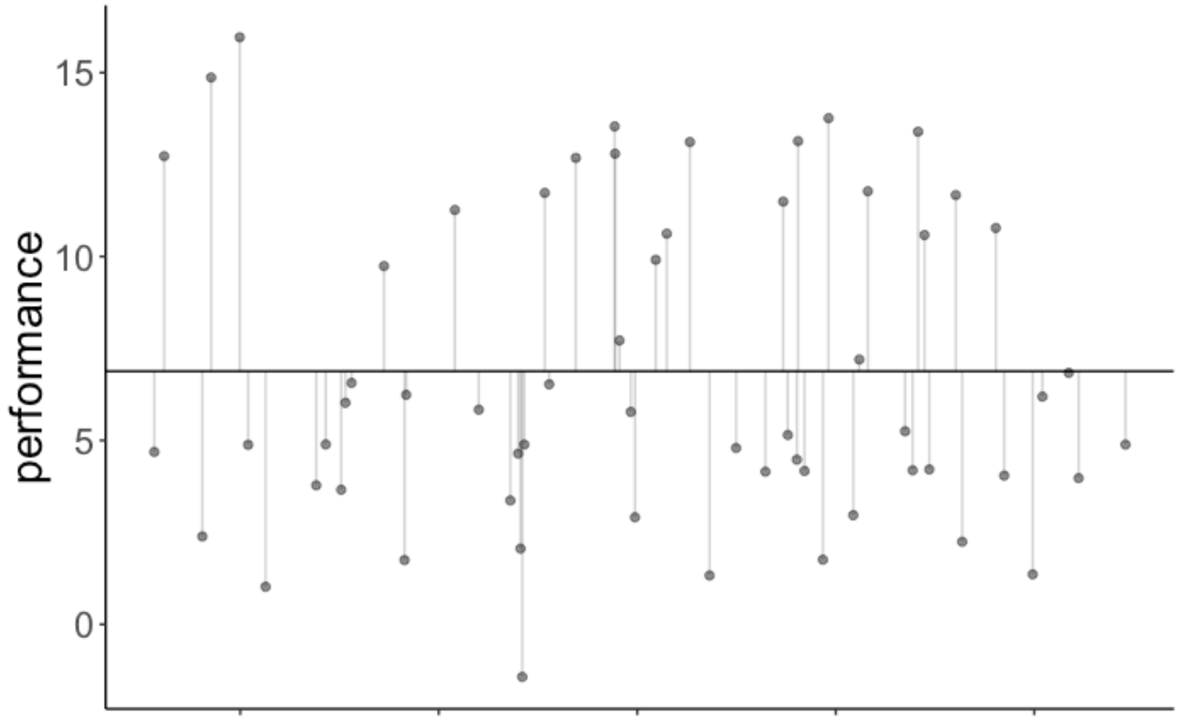
Data from a hypothetical developmental study

Contrasts

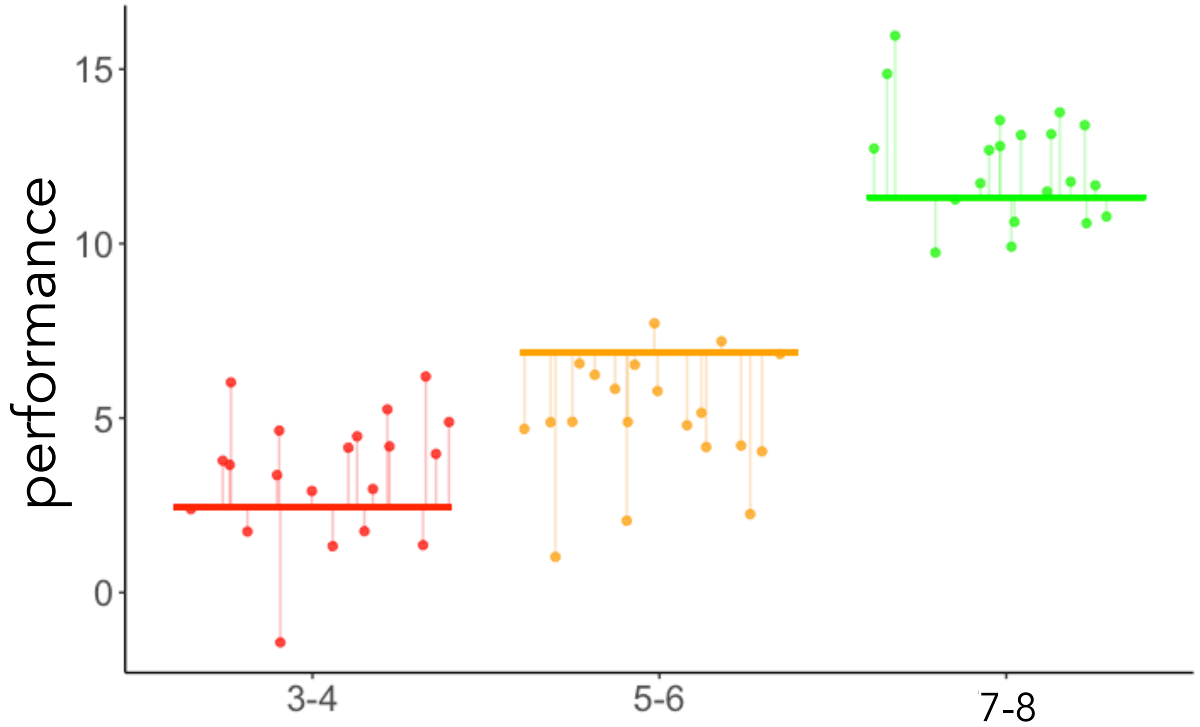
Does performance increase with age?

`contrasts = c(-1, 0, 1)`

Compact model



Augmented model

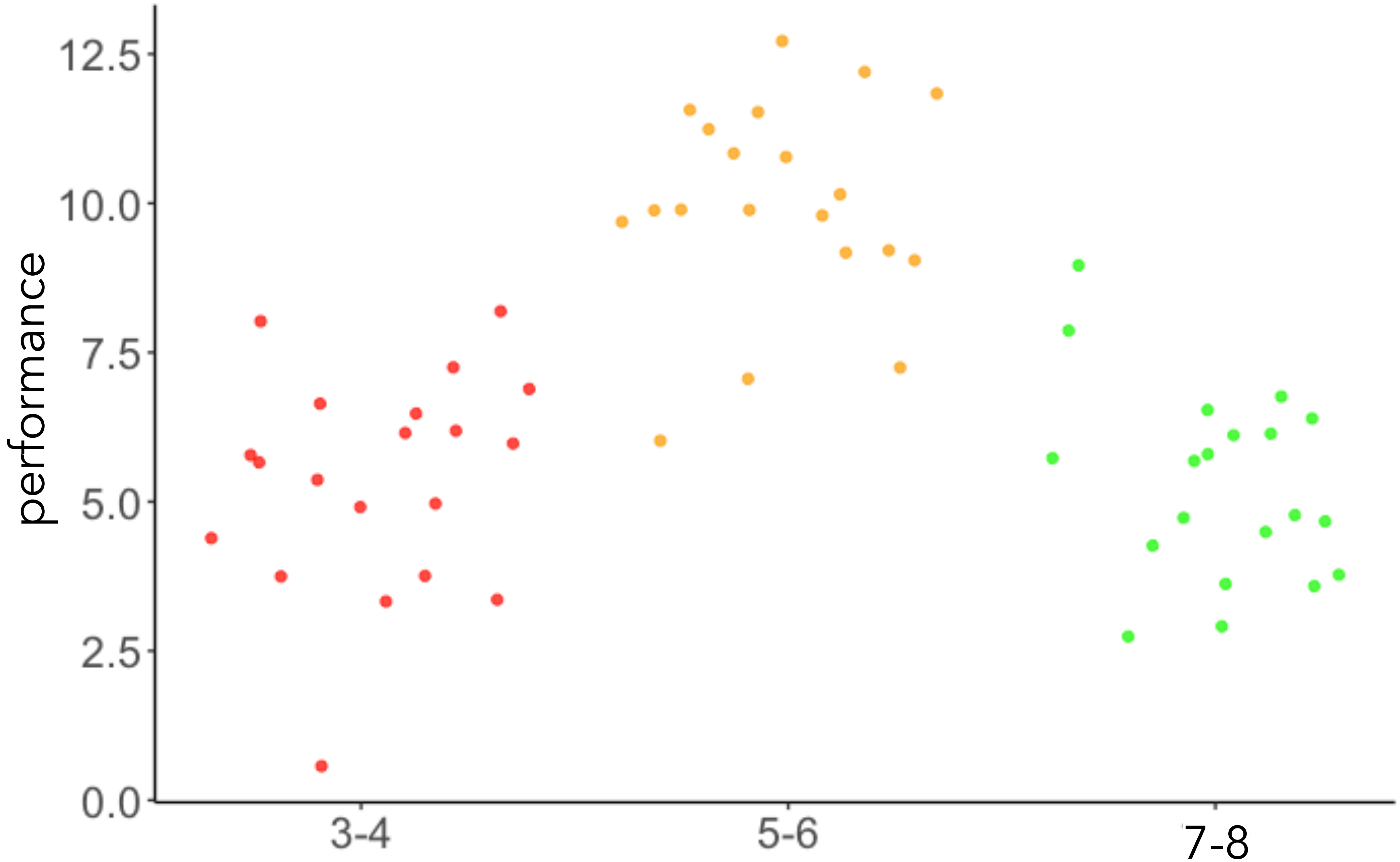


Model comparison

$p < .001$

Contrasts

Does performance increase with age?



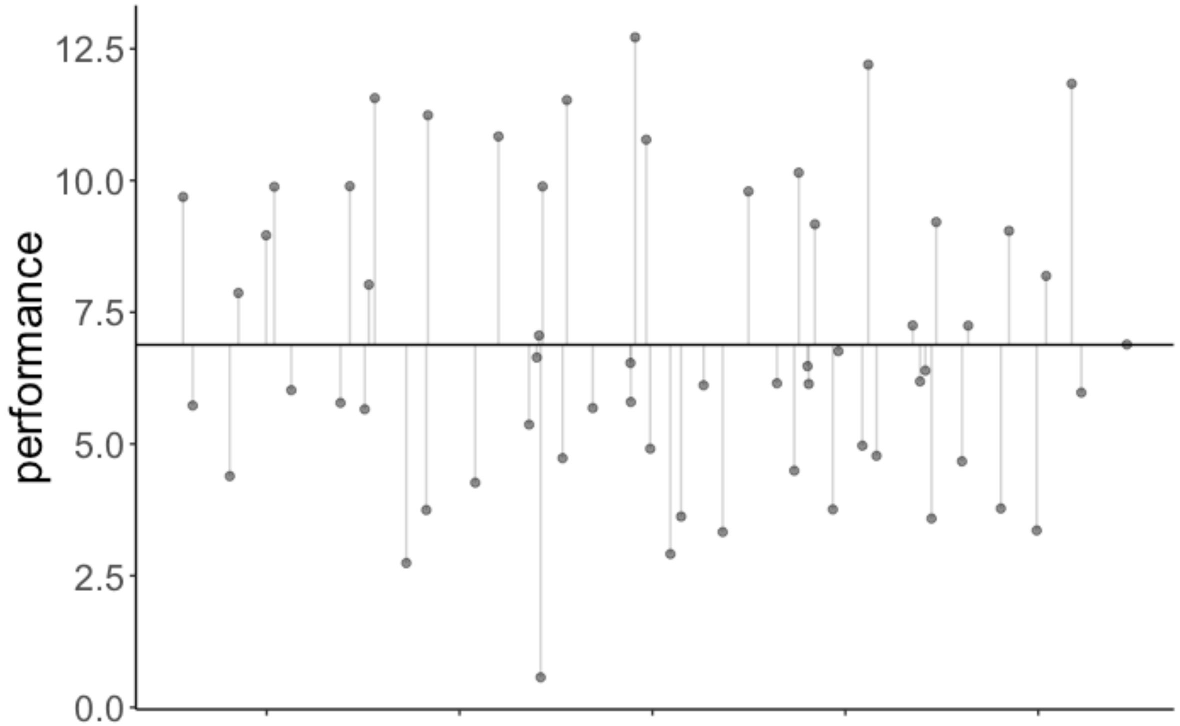
Data from another hypothetical developmental study

Contrasts

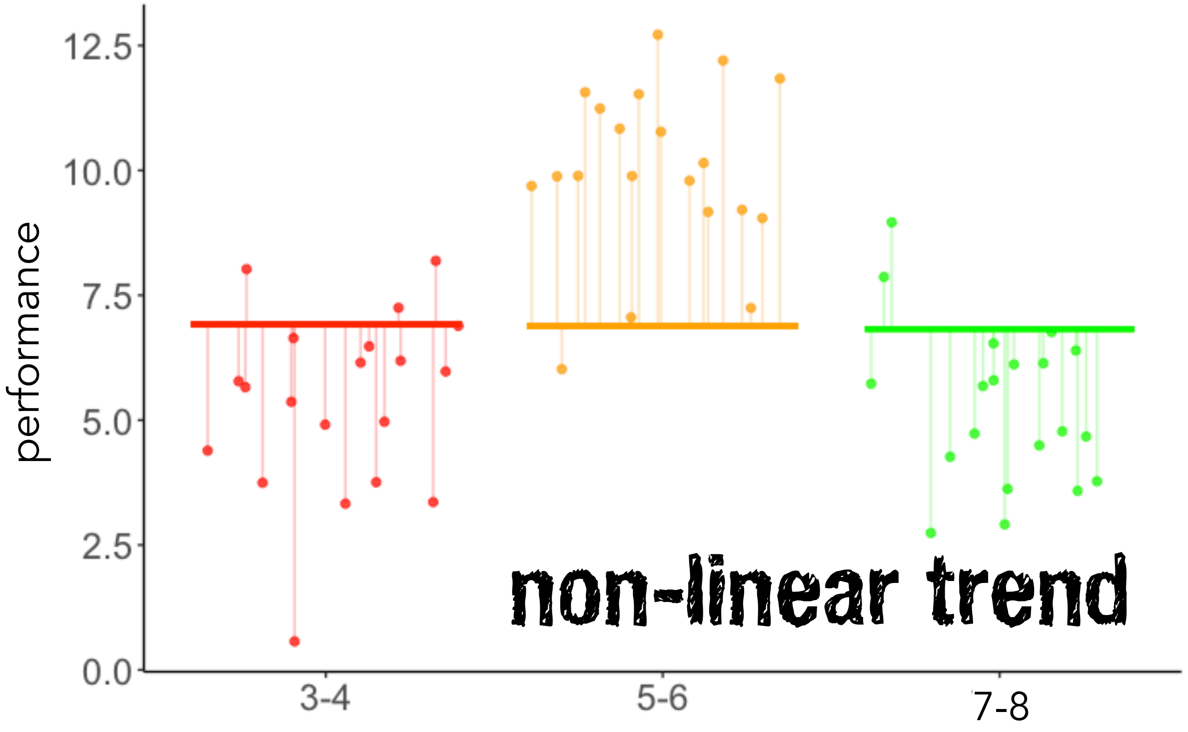
Does performance increase with age?

`contrasts = c(-1, 0, 1)`

Compact model



Augmented model



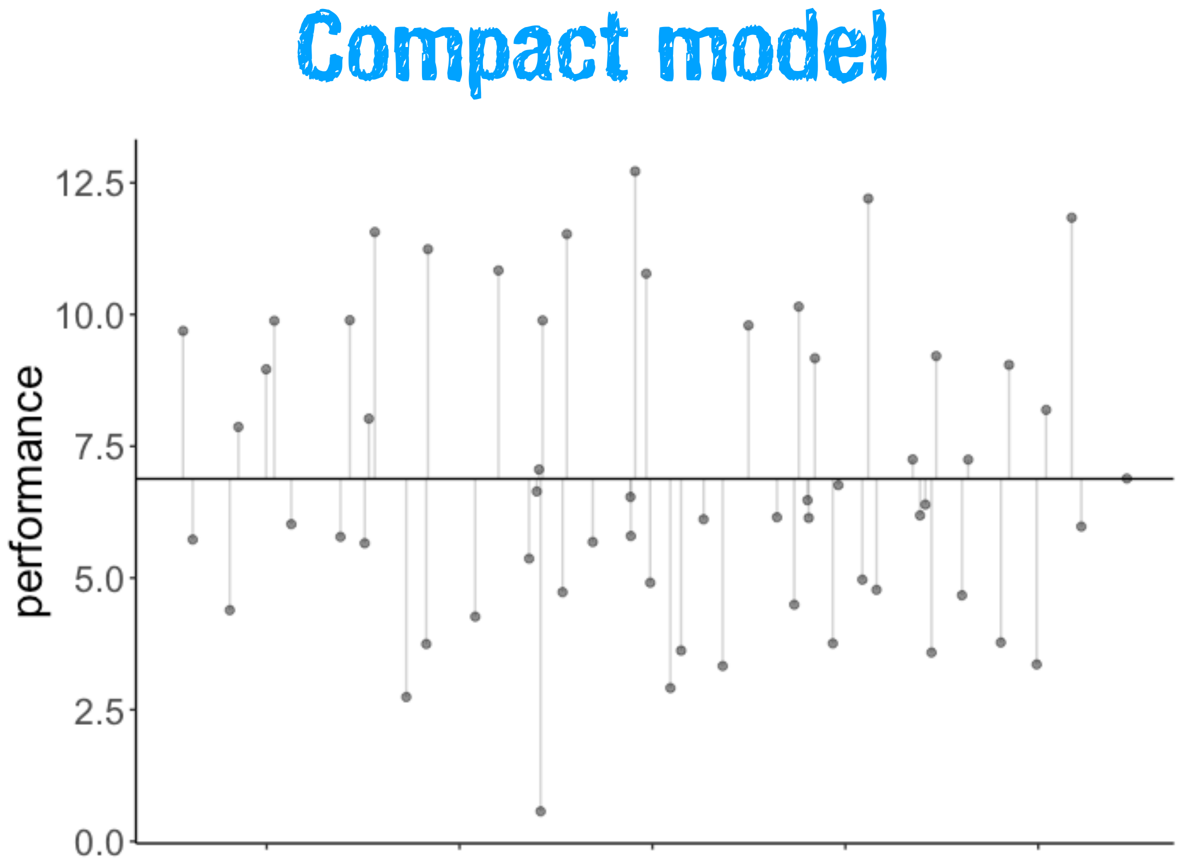
Model comparison

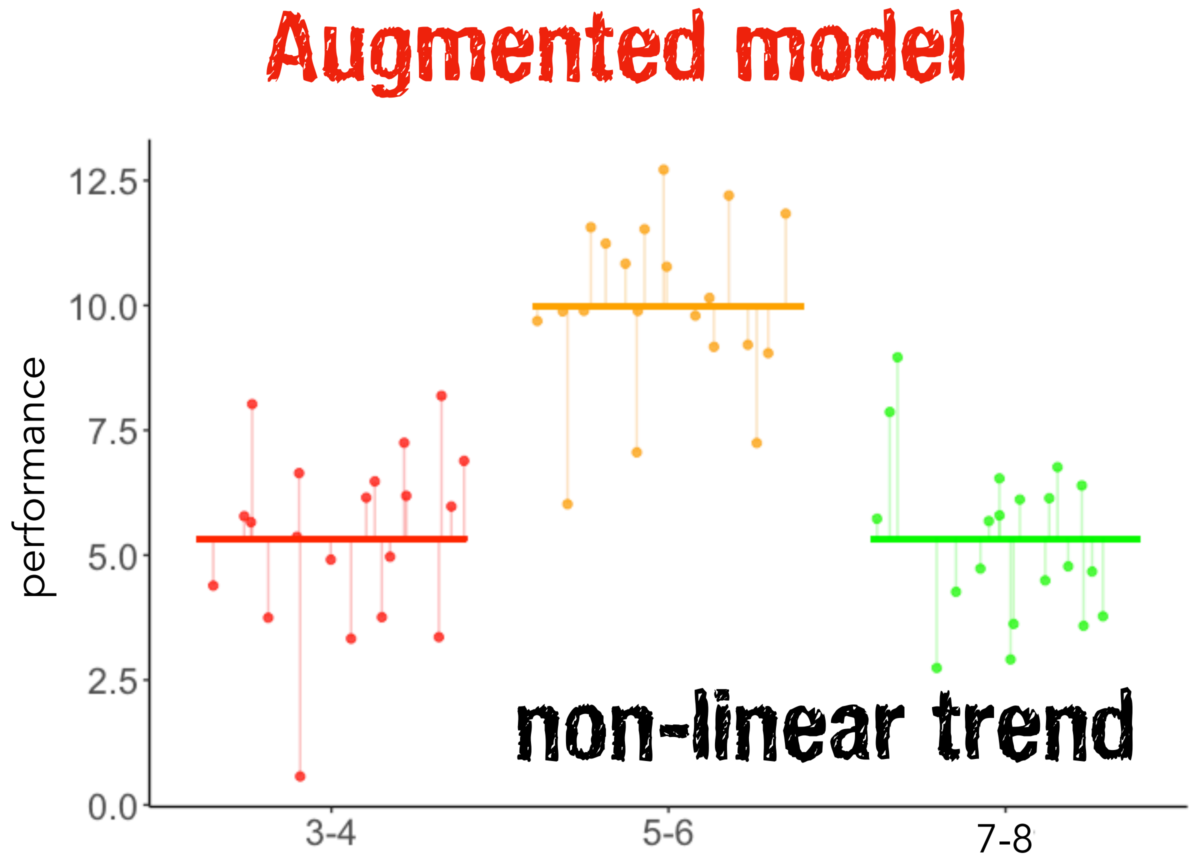
$p = .8508$

Contrasts

Does performance increase with age?

```
contrasts = c(-1, 2, -1)
```





Model comparison

p < .001

`emmeans` for handling linear contrasts in R

Linear contrasts

~~How to use contrasts in R~~

In short: don't bother.[1]

Like many before me, one of my stats classes technically "taught" me contrasts. But I didn't get the point and using them was cumbersome, so I promptly ignored them for years.

Luckily for me, someone came along and fixed the situation: `emmeans`. `emmeans` frames contrasts as a question you pose to a model: you can ask for all pairwise comparisons and get back that. `lm` and `summary` treat the same problem as fitting abstract coefficients, and you are left to answer your own question.

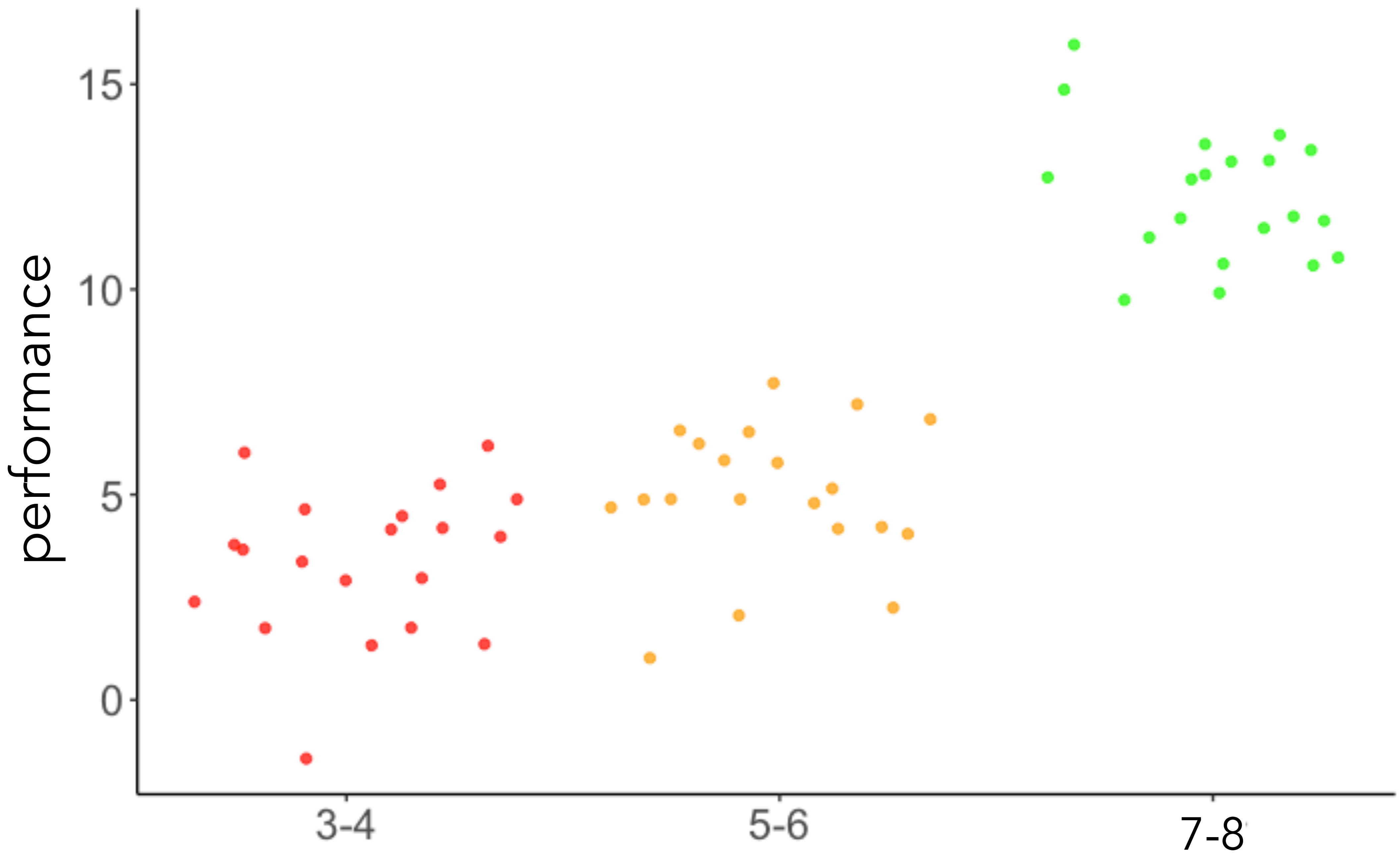
`emmeans` works with `lm`, `glm`, and the Bayesian friends in `brms` and `rstanarm`, so the process is applicable no matter the tool.

And you don't have to learn (much) about contrasts to take advantage of it.

https://timmastny.rbind.io/blog/tests-pairwise-categorical-mean-emmeans-contrast/

Contrasts

Does performance increase with age?



Data from a hypothetical developmental study

Linear contrasts in R

```r
library("emmeans") # for calculating contrasts

#  fit the linear model
fit = lm(formula = performance ~ group,
         data = df.development)
```

fit linear model

Linear contrasts in R

```
library("emmeans") # for calculating contrasts

#  fit the linear model
fit = lm(formula = performance ~ group,
         data = df.development)

# check factor levels
levels(df.development$group)
```

```
[1] "3-4" "5-6" "7-8"
```

check factor levels before defining contrasts

Linear contrasts in R

```
library("emmeans") # for calculating contrasts

#  fit the linear model
fit = lm(formula = performance ~ group,
         data = df.development)

# check factor levels
levels(df.development$group) [1] "3-4" "5-6" "7-8"

# define the contrasts of interest
contrasts = list(young_vs_old = c(-0.5, -0.5, 1),
                 three_vs_five = c(-0.5, 0.5, 0))
```

set up linear contrasts

Linear contrasts in R

```r
library("emmeans") # for calculating contrasts

#  fit the linear model
fit = lm(formula = performance ~ group,
         data = df.development)

# check factor levels
levels(df.development$group)

# define the contrasts of interest
contrasts = list(young_vs_old = c(-0.5, -0.5, 1),
                 three_vs_five = c(-0.5, 0.5, 0))

# compute significance test on contrasts
fit %>%
  emmeans("group",
          contr = contrasts,
          adjust = "bonferroni") %>%
  pluck("contrasts")
```

compute the results

```
[1] "3-4" "5-6" "7-8"
 contrast       estimate        SE df t.ratio p.value
 young_vs_old  16.093541 0.4742322 57  33.936  <.0001
 three_vs_five  1.606009 0.5475962 57   2.933  0.0097

P value adjustment: bonferroni method for 2 tests
```

Linear contrasts in R

```r
library("emmeans") # for calculating contrasts

#  fit the linear model
fit = lm(formula = performance ~ group,
         data = df.development)

# check factor levels
levels(df.development$group)

# define the contrasts of interest
contrasts = list(young_vs_old = c(-1, -1, 2),
                 three_vs_five = c(-1, 1, 0))

# compute significance test on contrasts
fit %>%
  emmeans("group",
          contr = contrasts,
          adjust = "bonferroni") %>%
  pluck("contrasts")
```

```
[1] "3-4" "5-6" "7-8"
 contrast      estimate    SE df t.ratio p.value
 young_vs_old    32.187 0.948 57  33.936 <.0001
 three_vs_five    0.803 0.274 57   2.933 0.0097

P value adjustment: bonferroni method for 2 tests
```

hypothesis tests are the same!

Interpreting the coefficients

```r
fit = lm(formula = performance ~ group,
         data = df.development)

# check factor levels
levels(df.development$group)

# define the contrasts of interest
contrasts = list(young_vs_old = c(-1, -1, 2),
                 three_vs_five = c(-0.5, 0.5, 0))

# compute estimated marginal means
leastsquare = emmeans(fit, "group")

# run analyses
contrast(leastsquare,
         contrasts,
         adjust = "bonferroni")
```

hypothesis tests are the same!

```
[1] "3-4" "5-6" "7-8"
 contrast      estimate    SE df t.ratio p.value
 young_vs_old    32.187 0.948 57  33.936 <.0001
 three_vs_five    0.803 0.274 57   2.933 0.0097

P value adjustment: bonferroni method for 2 tests
```

Post hoc tests

```r
fit = lm(formula = performance ~ group,
         data = df.development)

# pairwise differences between all the groups
fit %>%
  emmeans(pairwise ~ group) %>%
  pluck("contrasts")
```

all pairwise tests between groups

```
contrast    estimate        SE df t.ratio p.value
 3-4 - 5-6   -1.606009 0.5475962 57  -2.933  0.0145
 3-4 - 7-8 -16.896546 0.5475962 57 -30.856  <.0001
 5-6 - 7-8 -15.290537 0.5475962 57 -27.923  <.0001

P value adjustment: bonferroni method for 3 tests
```

Post hoc tests

```r
# fit the model
fit = lm(formula = balance ~ hand + skill,
         data = df.poker)

# post hoc tests
fit %>%
  emmeans(pairwise ~ hand + skill,
          adjust = "bonferroni") %>%
  pluck("contrasts")
```

the poker data

```
contrast                             estimate        SE  df t.ratio p.value
bad,average - neutral,average       -4.381023 0.6051766 286  -7.239  <.0001
bad,average - good,average          -7.060823 0.6051766 286 -11.667  <.0001
bad,average - bad,expert            -0.740385 0.4896119 286  -1.512  1.0000
bad,average - neutral,expert        -5.121408 0.7611327 286  -6.729  <.0001
bad,average - good,expert           -7.801208 0.7611327 286 -10.249  <.0001
neutral,average - good,average      -2.679800 0.5884403 286  -4.554  0.0001
neutral,average - bad,expert         3.640638 0.7953578 286   4.577  0.0001
neutral,average - neutral,expert    -0.740385 0.4896119 286  -1.512  1.0000
neutral,average - good,expert       -3.420185 0.7654945 286  -4.468  0.0002
good,average - bad,expert            6.320438 0.7953578 286   7.947  <.0001
good,average - neutral,expert        1.939415 0.7654945 286   2.534  0.1774
good,average - good,expert          -0.740385 0.4896119 286  -1.512  1.0000
bad,expert - neutral,expert         -4.381023 0.6051766 286  -7.239  <.0001
bad,expert - good,expert            -7.060823 0.6051766 286 -11.667  <.0001
neutral,expert - good,expert        -2.679800 0.5884403 286  -4.554  0.0001

P value adjustment: bonferroni method for 15 tests
```

that's a lot of tests!

... not

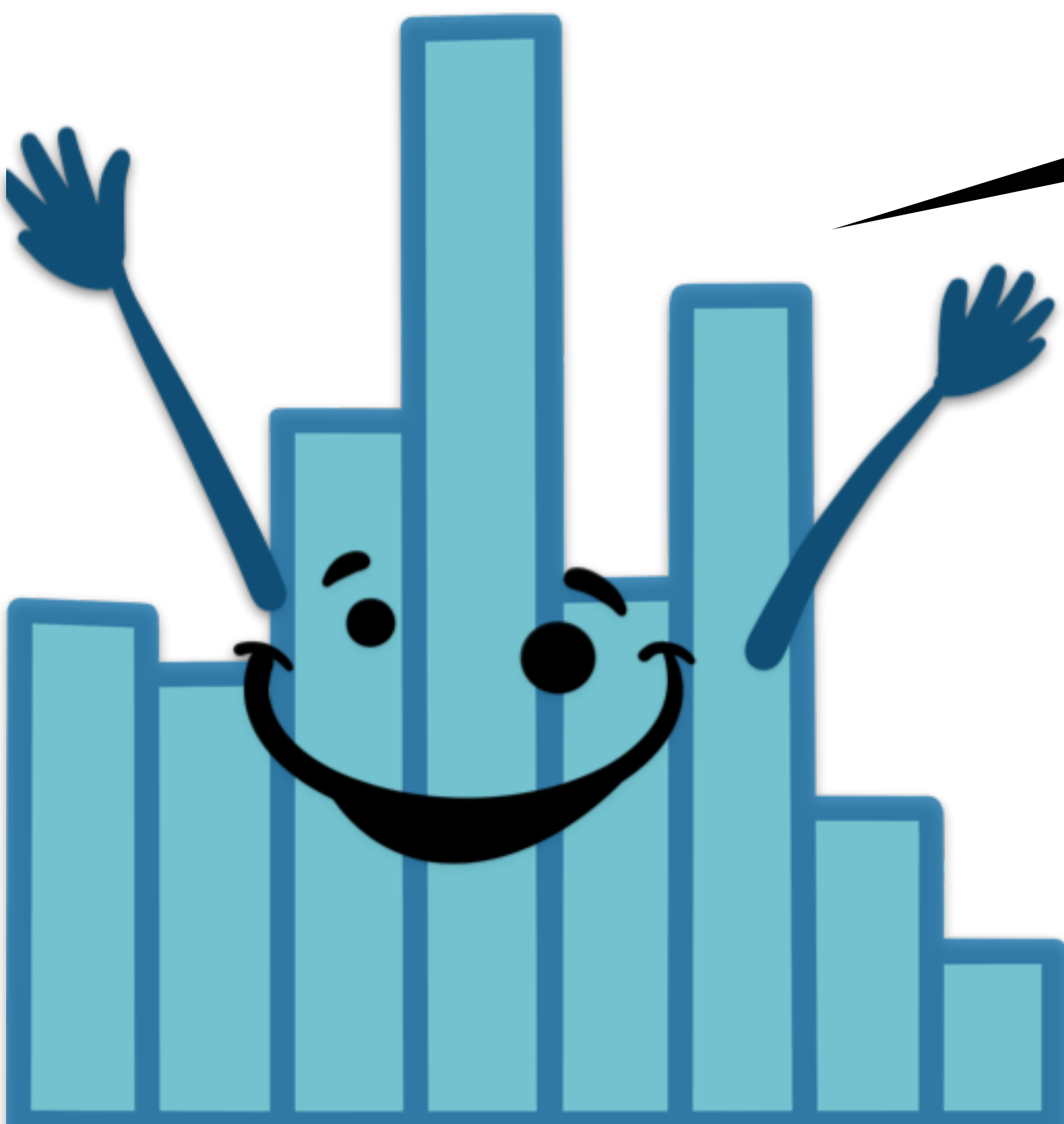
all pairwise tests between groups

Contrasts

- linear contrasts allow us to ask more specific questions of our data
- rather than asking whether any of the group means are significantly different from each other (ANOVA), we can ask questions such as:
  - Does performance increase with age?
  - Is the overall performance in Condition B and C better from the performance in Condition A?

01:00

stretch break!



Power analysis

Making decisions

Type I Error

Type II Error

$\mathbf{H_0}$: Not pregnant. $\mathbf{H_1}$: Pregnant.

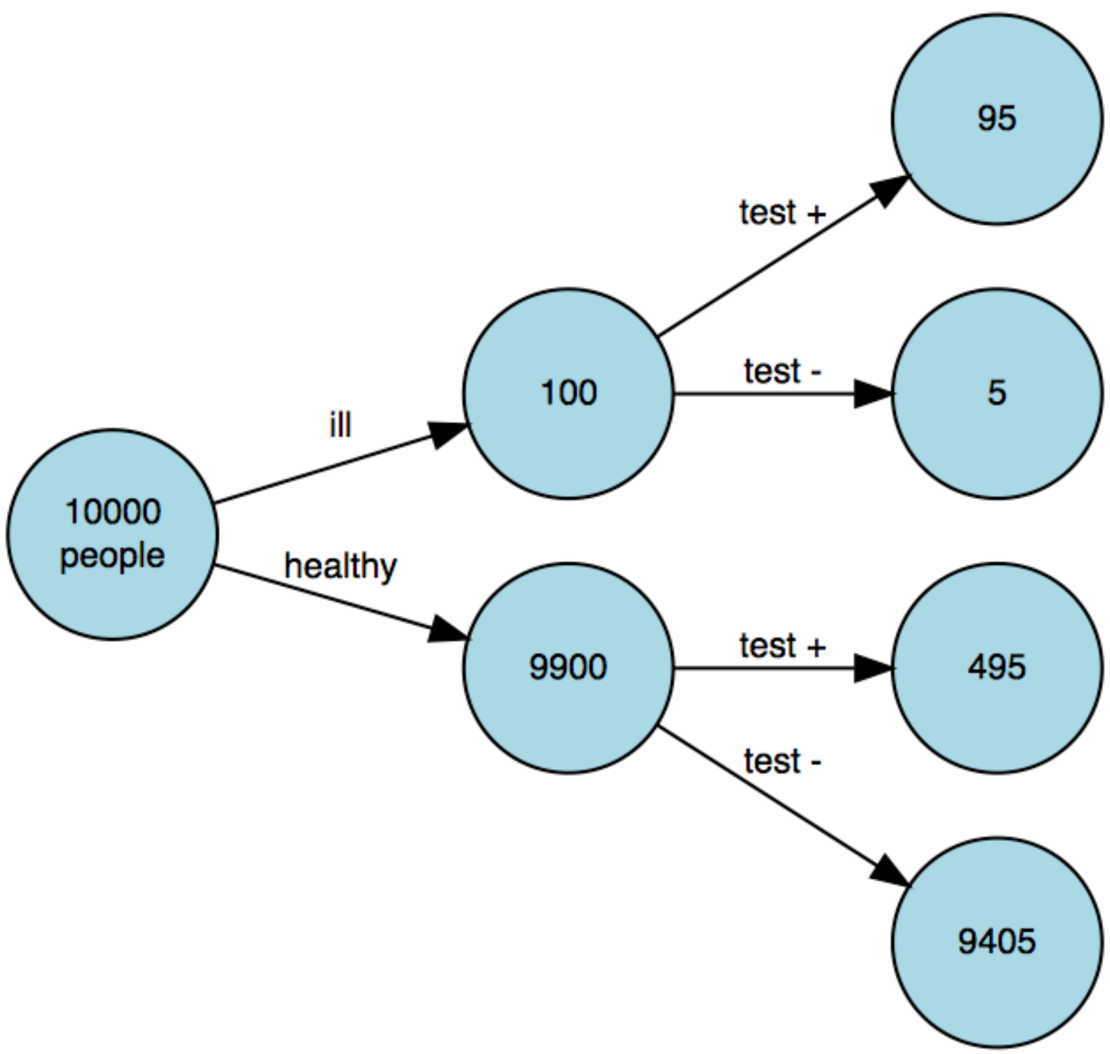
Type I Error: Falsely rejecting the null hypothesis (even though it is true).

Type II Error: Failing to reject the null hypothesis (even though it is false).

Clue guide to probability

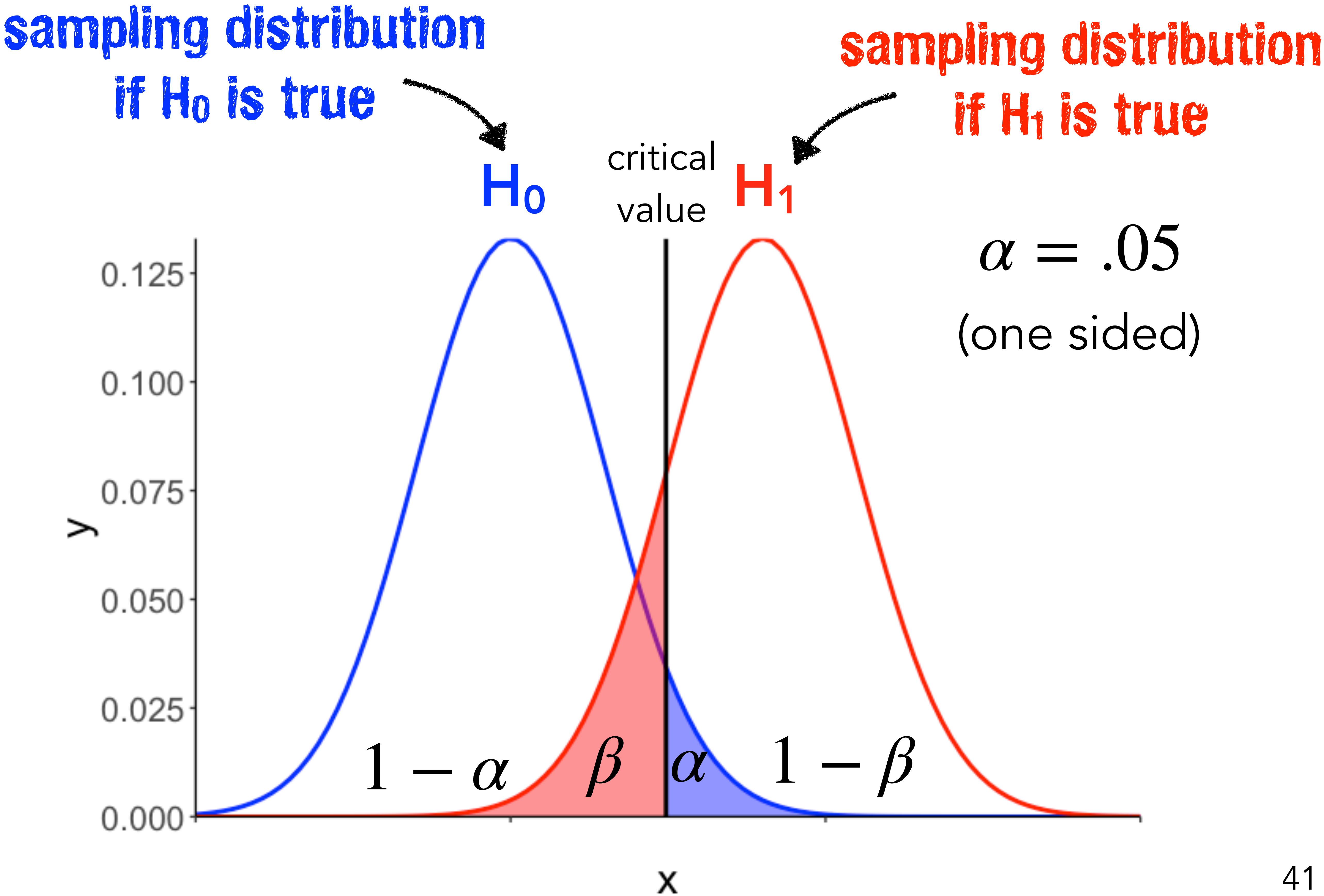
H_0: The person is healthy.

H_1: The person is ill.

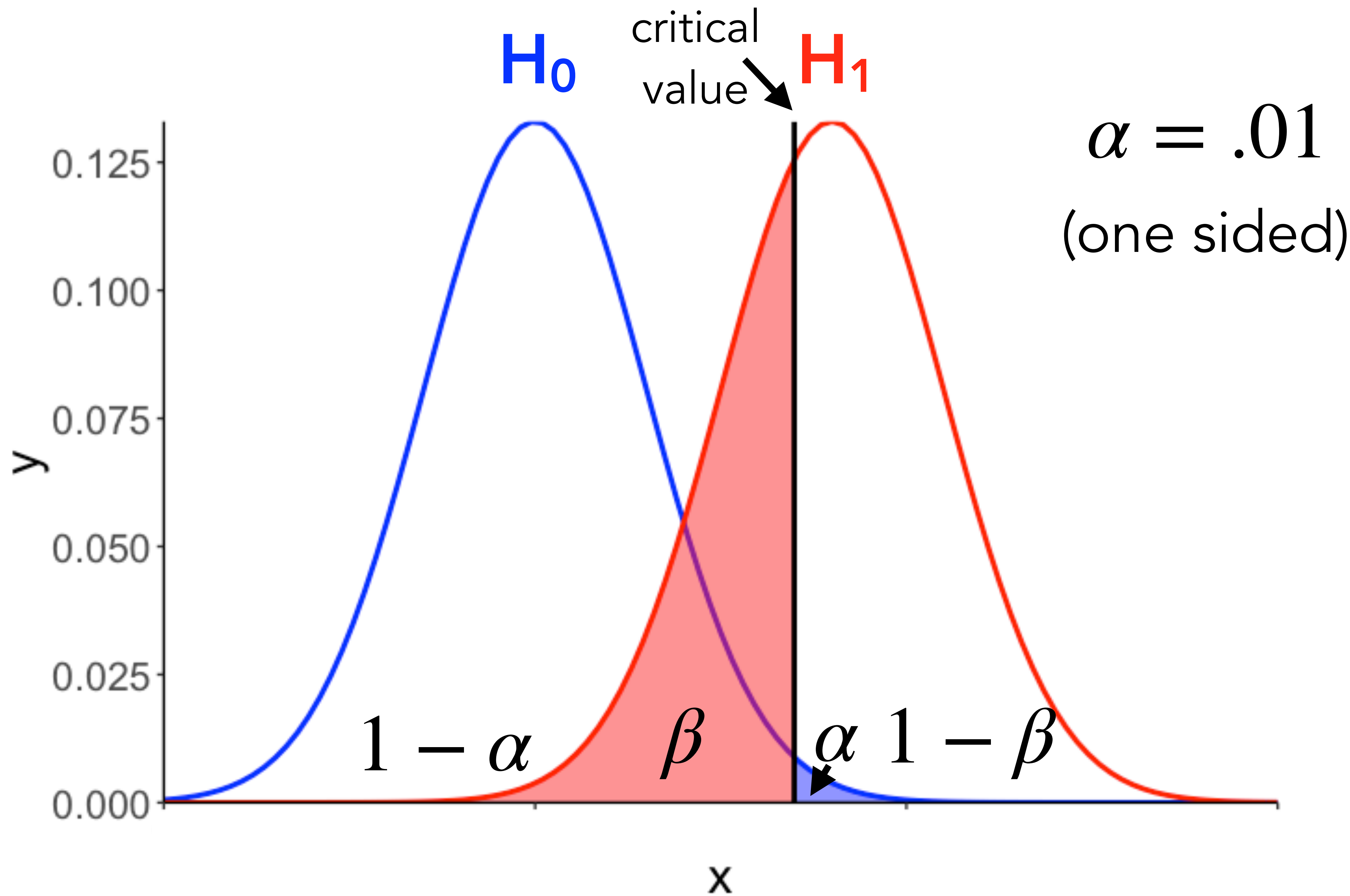


true positive	Sensitivity $p(\text{reject } H_0 \mid H_1 \text{ is true})$	Power	$\boxed{1-\beta}$
false negative	Type II error $p(\text{not reject } H_0 \mid H_1 \text{ is true})$		β
false positive	Type I error $p(\text{reject } H_0 \mid H_0 \text{ is true})$		α
true negative	$p(\text{not reject } H_0 \mid H_0 \text{ is true})$ Specificity		$1-\alpha$

What affects power?

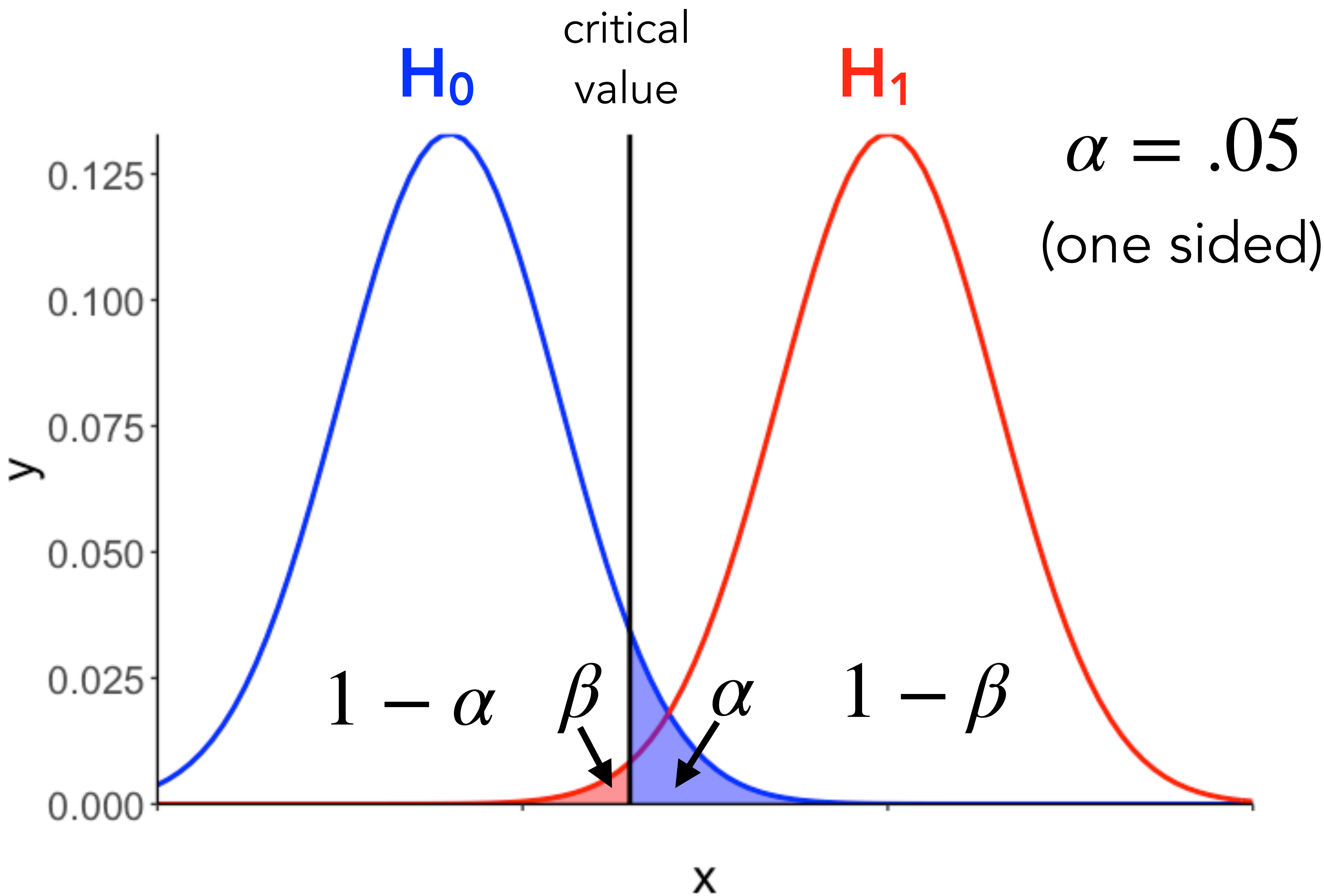


What affects power?



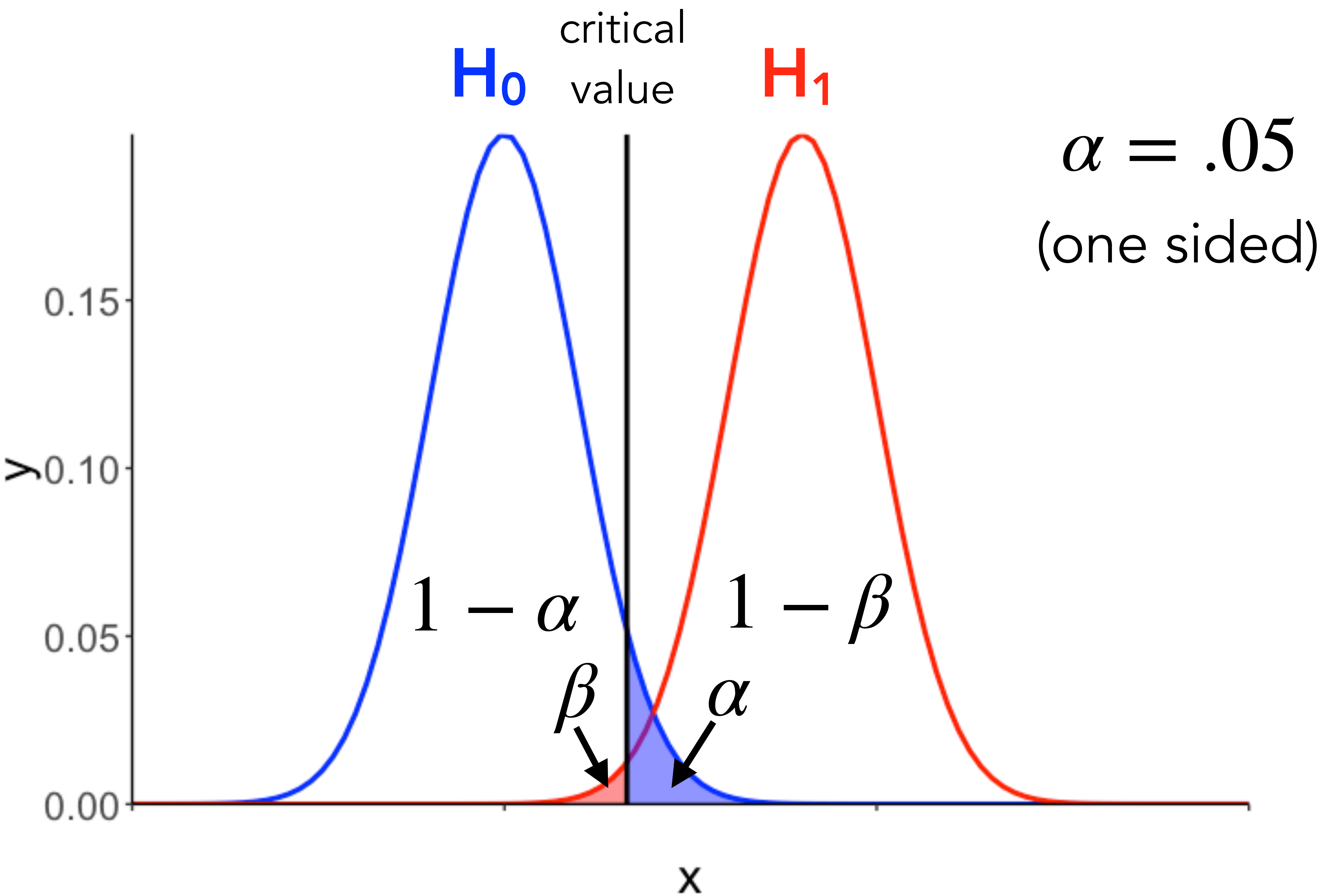
α affects power

What affects power?



distance between means affects power

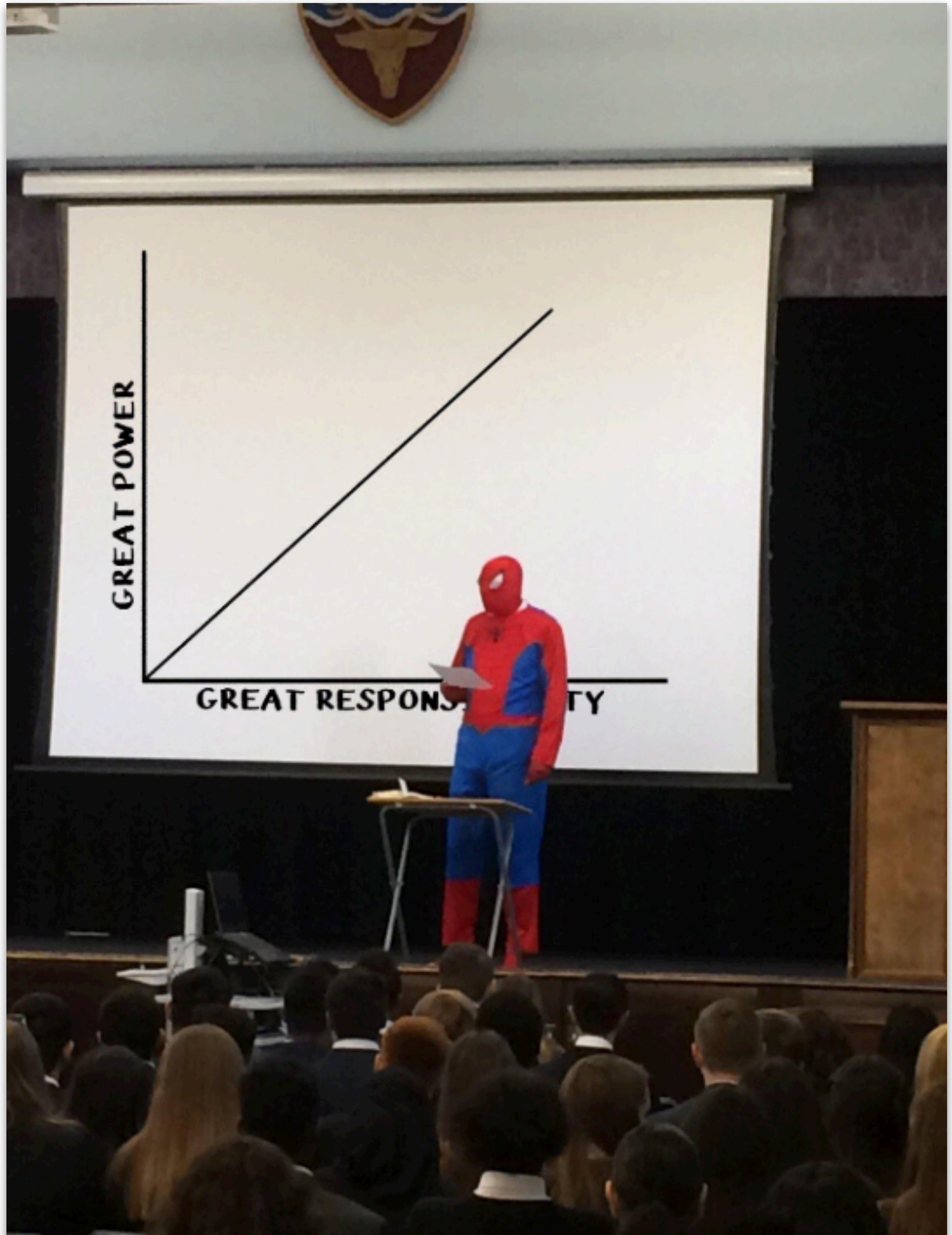
What affects power?



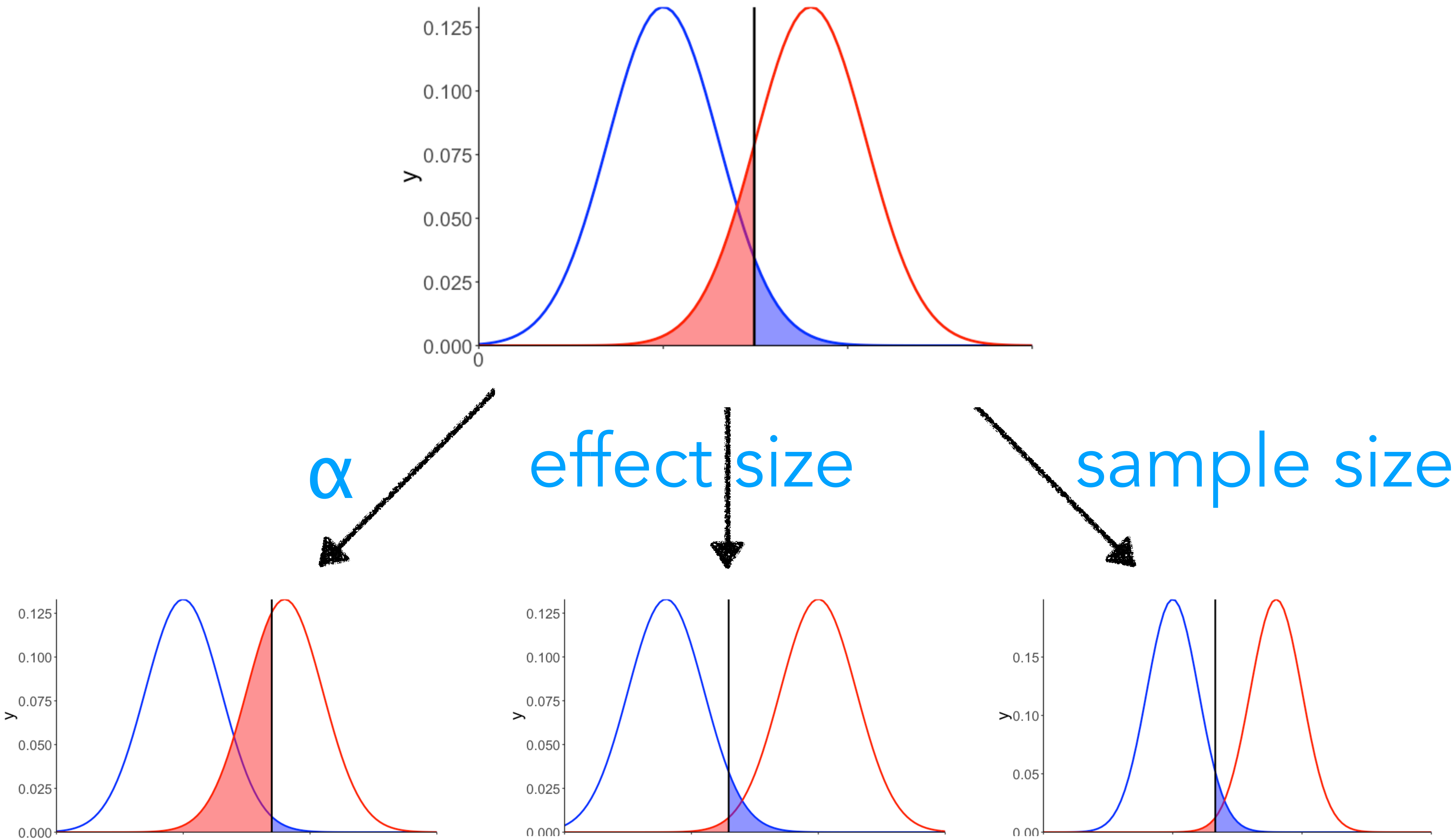
variance affects power

Calculating power

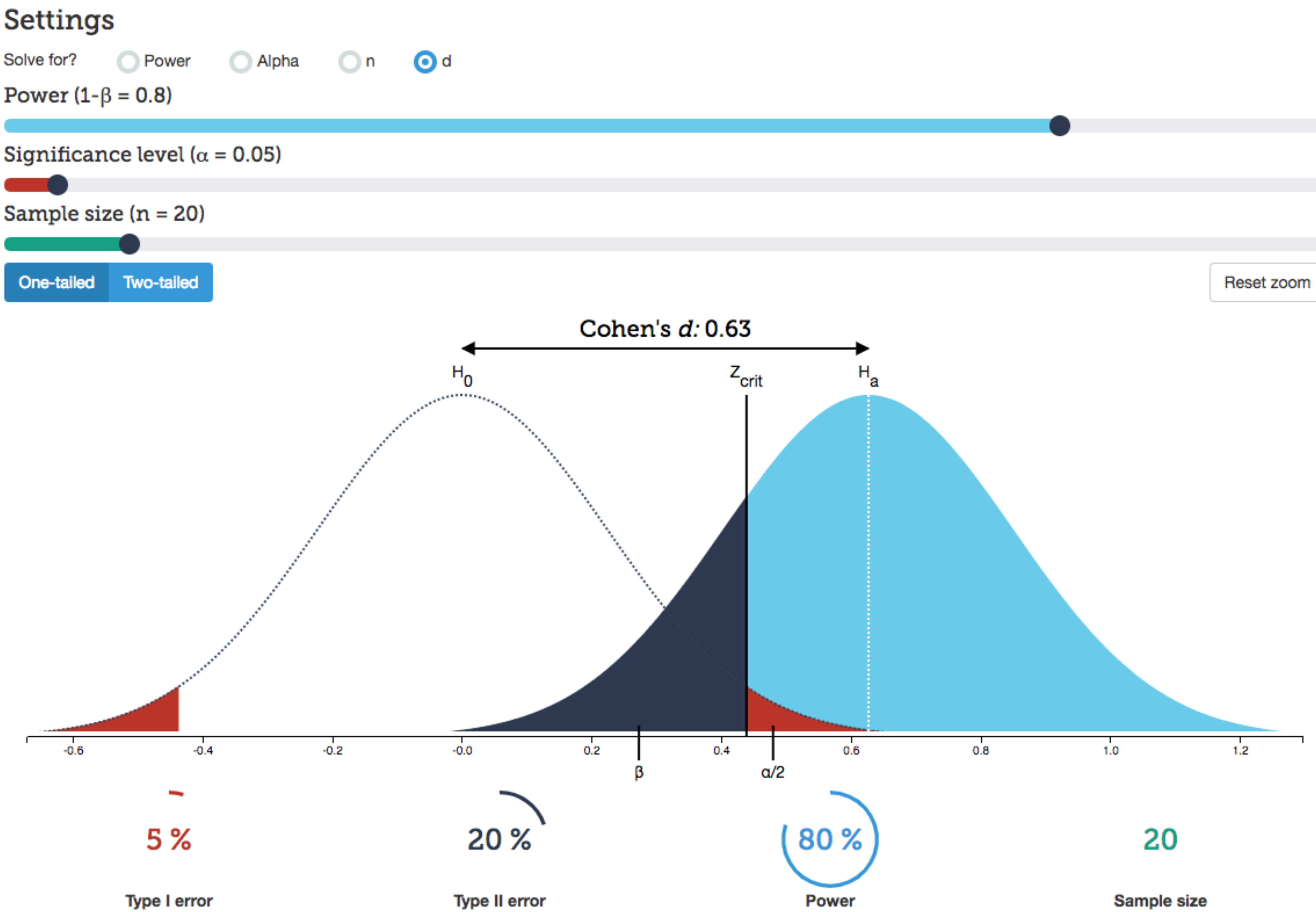
With great power comes ...



The knobs we can turn to affect power



Visualization demo



https://rpsychologist.com/d3/NHST/

The **power** of a binary hypothesis test is the probability that the test rejects the null hypothesis (H_0) when a **specific** alternative hypothesis (H_1) is true.

H_0: Students and non-students have the same balance.

Model C

$$Y_i = \beta_0 + \epsilon_i$$

$$\beta_1 = 0$$

H_1: Students and non-students have different balances.

Model A

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$$

$$\beta_1 \neq 0$$

We cannot calculate power in this case.
We need a specific alternative hypothesis!

The **power** of a binary hypothesis test is the probability that the test rejects the null hypothesis (H_0) when a **specific** alternative hypothesis (H_1) is true.

H_0: Students and non-students have the same balance.

Model C

$$Y_i = \beta_0 + \epsilon_i$$

$$\beta_1 = 0$$

H_1: Students and non-students have different balances.

Model A

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$$

$$\beta_1 = 300$$

We can calculate power in this case (since we have a specific alternative hypothesis)!

Effect sizes

Effect sizes

- a p-value tells us whether we can reject the H_0
- effect sizes is a measure of the strength of the actual effect

Why can't we just use p-values as a measure of the effect size?

$$F = \frac{\mathrm{PRE}/(\mathrm{PA} - \mathrm{PC})}{(1 - \mathrm{PRE})/(n - \mathrm{PA})}$$

any PRE will become significant if *n* gets large enough

PRE = proportional reduction in error
PA = # parameters in the augmented model
PC = # parameters in the compact model
n = sample size

statistical vs. practical significance

Effect sizes

PRE = **p**roportional **r**eduction in **e**rror

Compact model

SSE(C)

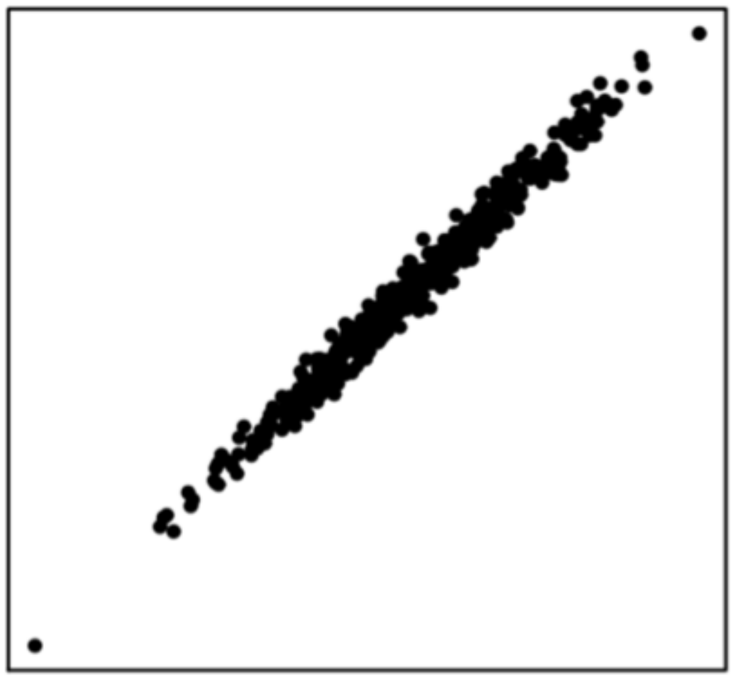
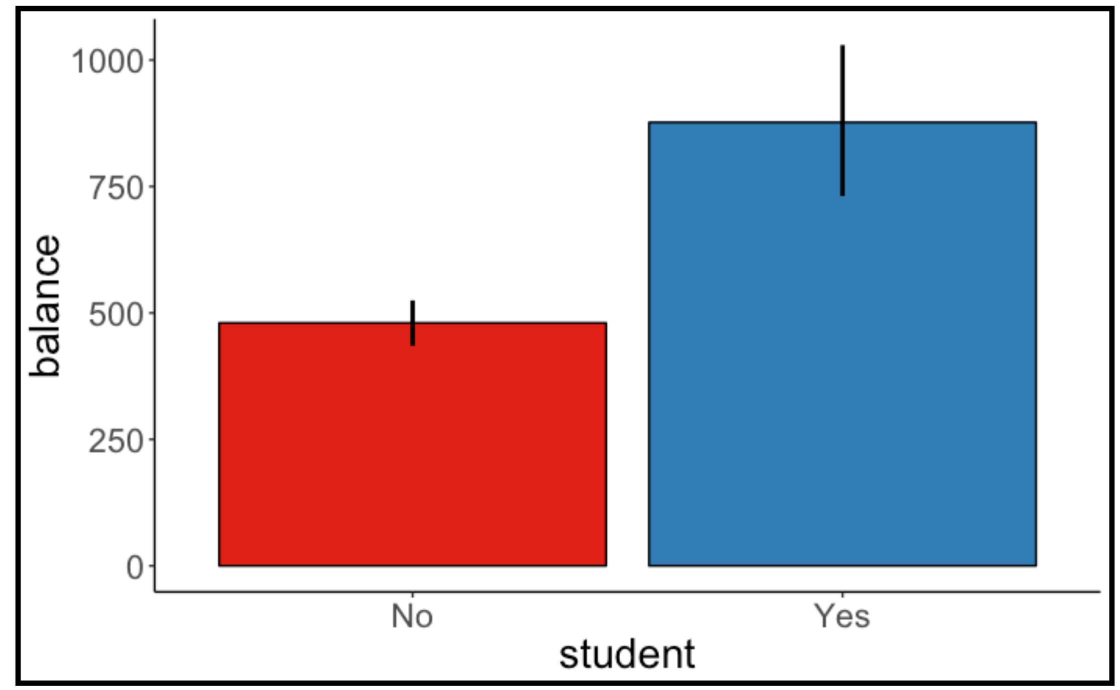
Augmented model

SSE(A)

$$\text{PRE} = 1 - \frac{\text{SSE}(A)}{\text{SSE}(C)}$$

SSE = sum of squared errors

Common effect sizes

Relationships between variables	Differences between groups
	
r correlation	Cohen's d

Correlation

Pearson correlation

$$r(X,Y) = \frac{\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \overline{X})^2} \cdot \sqrt{\sum_{i=1}^{n}(Y_i - \overline{Y})^2}}$$

Cohen's guidelines for the social sciences

Effect size	*r*
Small	0.1
Medium	0.3
Large	0.5

depends very much on the domain

Cohen's *d*

- standardized difference between two means

absolute
difference
between means

$$d = \frac{|\bar{y}_1 - \bar{y}_2|}{s_p}$$

pooled standard variation

$$s_p = \sqrt{\frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}{n_1 + n_2 - 2}}$$

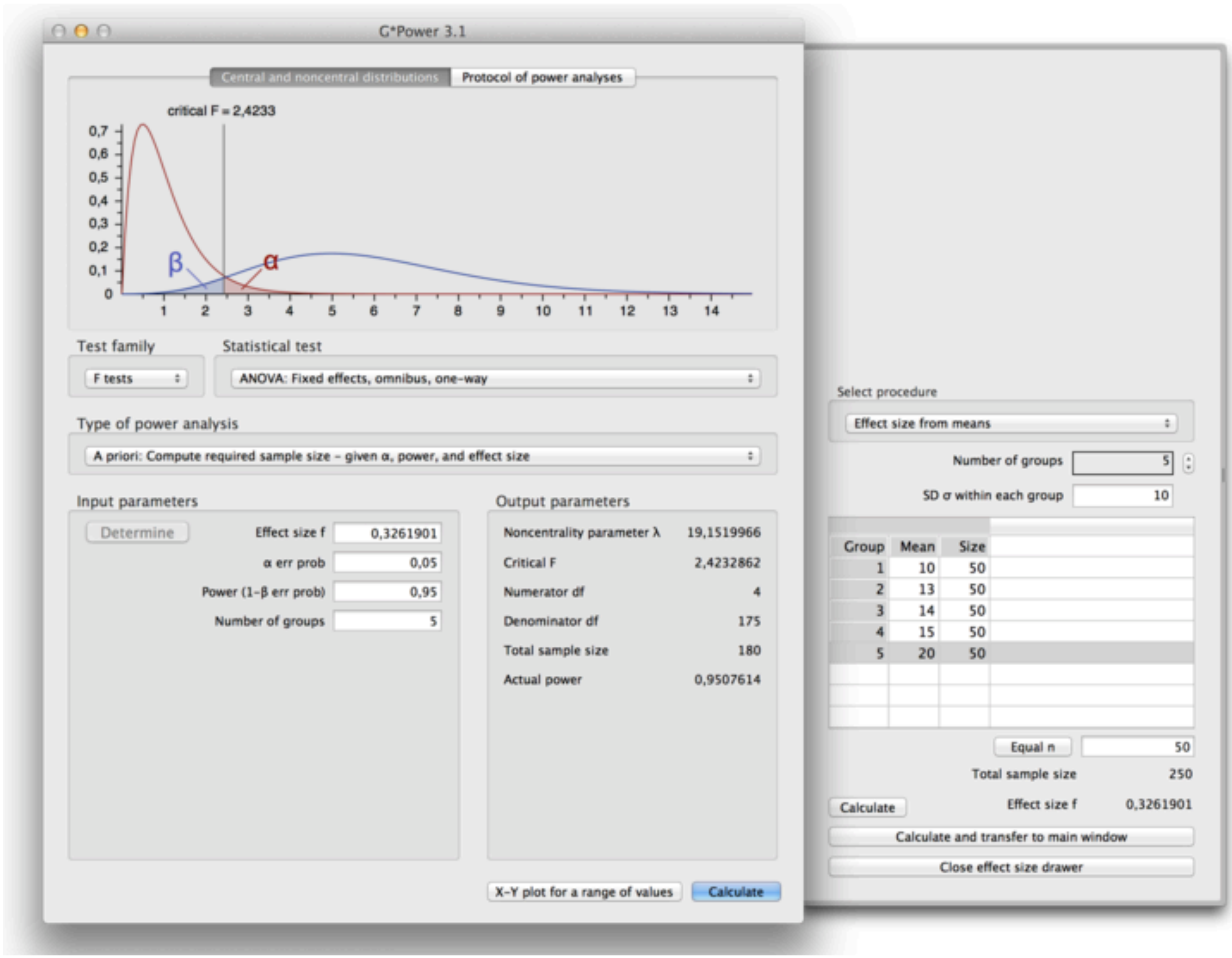
Effect size	***d***
Very small	0.01
Small	0.20
Medium	0.50
Large	0.80
Very large	1.20
Huge	2.0

Difference between two means in pooled standard deviation

Determining sample size

How many participants do I need to run to have a good chance of detecting a true effect?

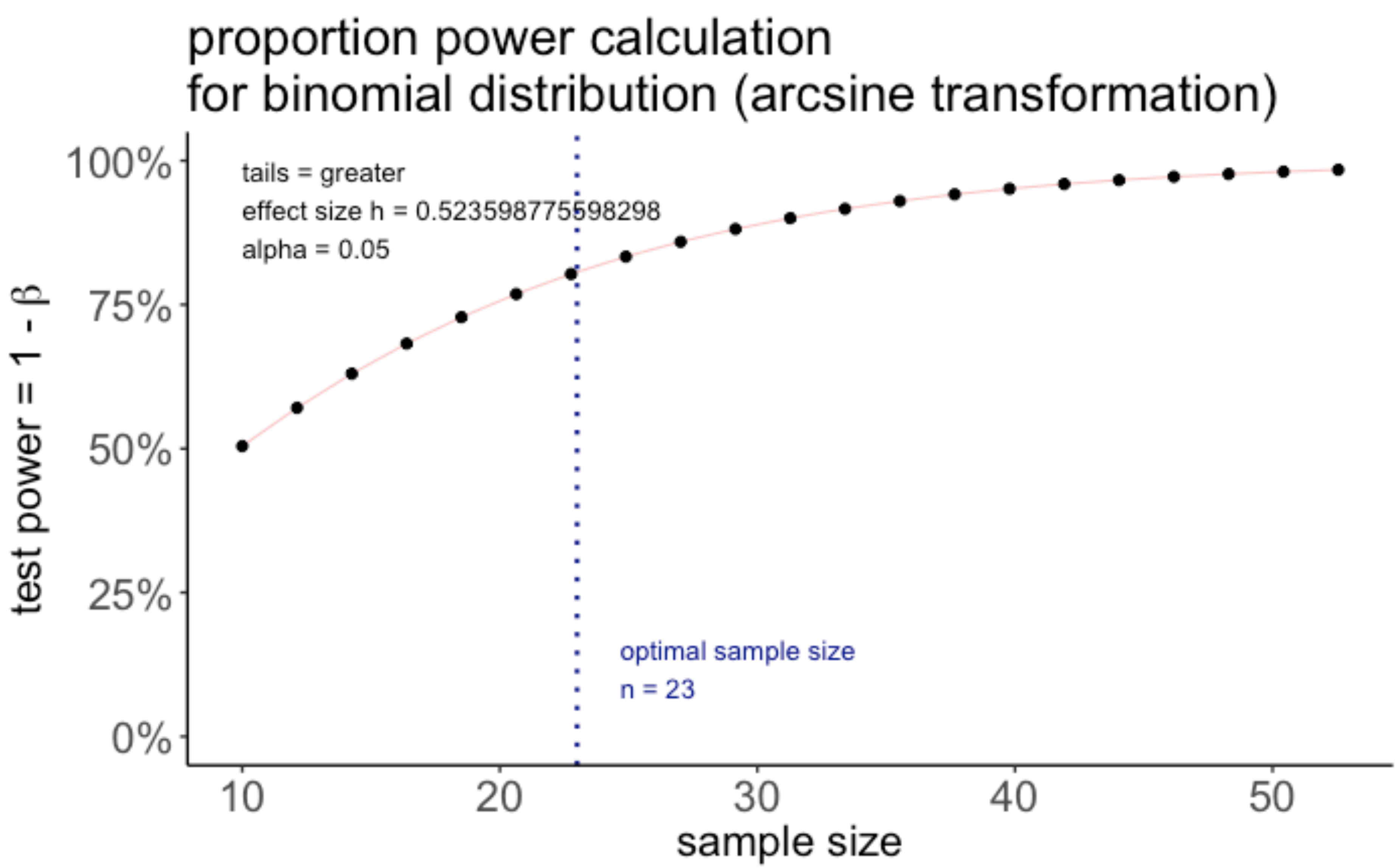
G*Power 3.1: Alternative software for power calculations



http://www.gpower.hhu.de/

Example

```r
library("pwr")
pwr.p.test(h = ES.h(p1 = 0.75, p2 = 0.50),
               sig.level = 0.05,
               power = 0.80,
               alternative = "greater") %>%
 plot()
```



Power simulation recipe

- assume:
  - α, *n*, effect size
- simulate a large number of data sets of size *n* with the specified effect size
- for each data set, run a statistical test to calculate the p-value
- determine the probability of rejecting the H_0 (given that H_1 is true)

Learn about more advanced simulation techniques in R

Let's simulate

```
library("purrr")
```

automatically loaded with
`library("tidyverse")`

`map()`

`map()`

vector or list

function

```r
map(.x = 1:3, .f = ~ .x^2)
```

outputs a list

```
[[1]]
[1] 1

[[2]]
[1] 4

[[3]]
[1] 9
```

- **`map`**`(list, function)` applies a function to each element of the list
- it's a unified version of the many different **`apply`**`()` functions in base R
- you already know a cousin of **`map`**`()`: **`replicate`**`()`
- use **`map`**`()`, don't write `for(){}` loops!
- it's extremely powerful in combination with data frames

map()

same same but different

```r
map(.x = 1:3, .f = ~ .x^2)

map(1:3, ~ .x^2)

map(1:3, ~ .^2)

map(.x = 1:3, .f = function(.x) .x^2)

# using a function
square = function(x){x^2}
map(1:3, square)
```

time

Plan for today

- Linear contrasts
  - Testing specific hypotheses with linear contrasts
  - `emmeans` for handling linear contrasts in R
- Power analysis
  - Making decisions
  - Calculating power
  - Effect sizes
  - Determining sample size
- Learn about more advanced simulation techniques in R
  - `map()`
  - list columns: `nest()`, `unnest()`

Feedback

How was the pace of today's class?

much too slow	a little too slow	just right	a little too fast	much too fast

How happy were you with today's class overall?

What did you like about today's class? What could be improved next time?

Thank you!