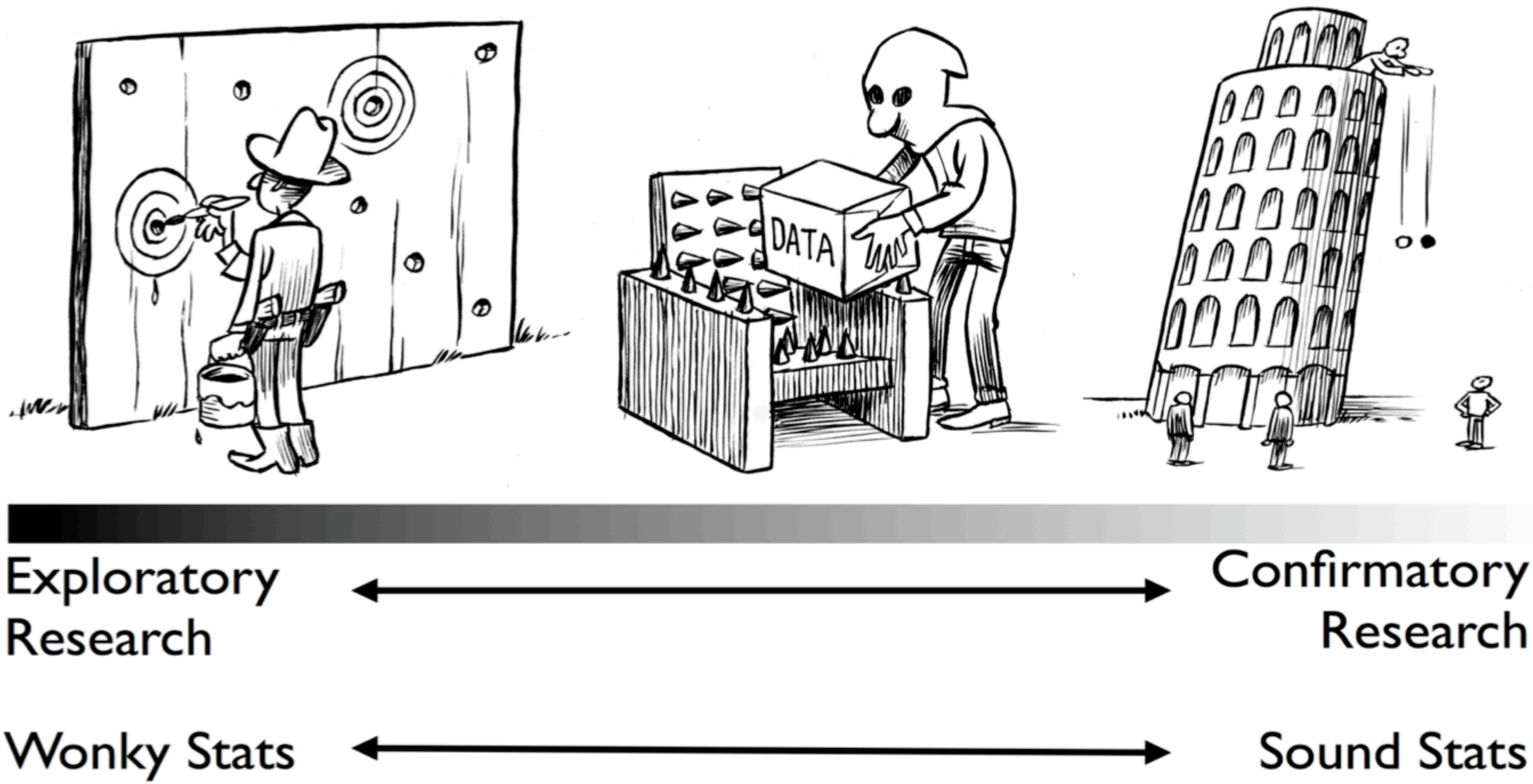
Power analysis

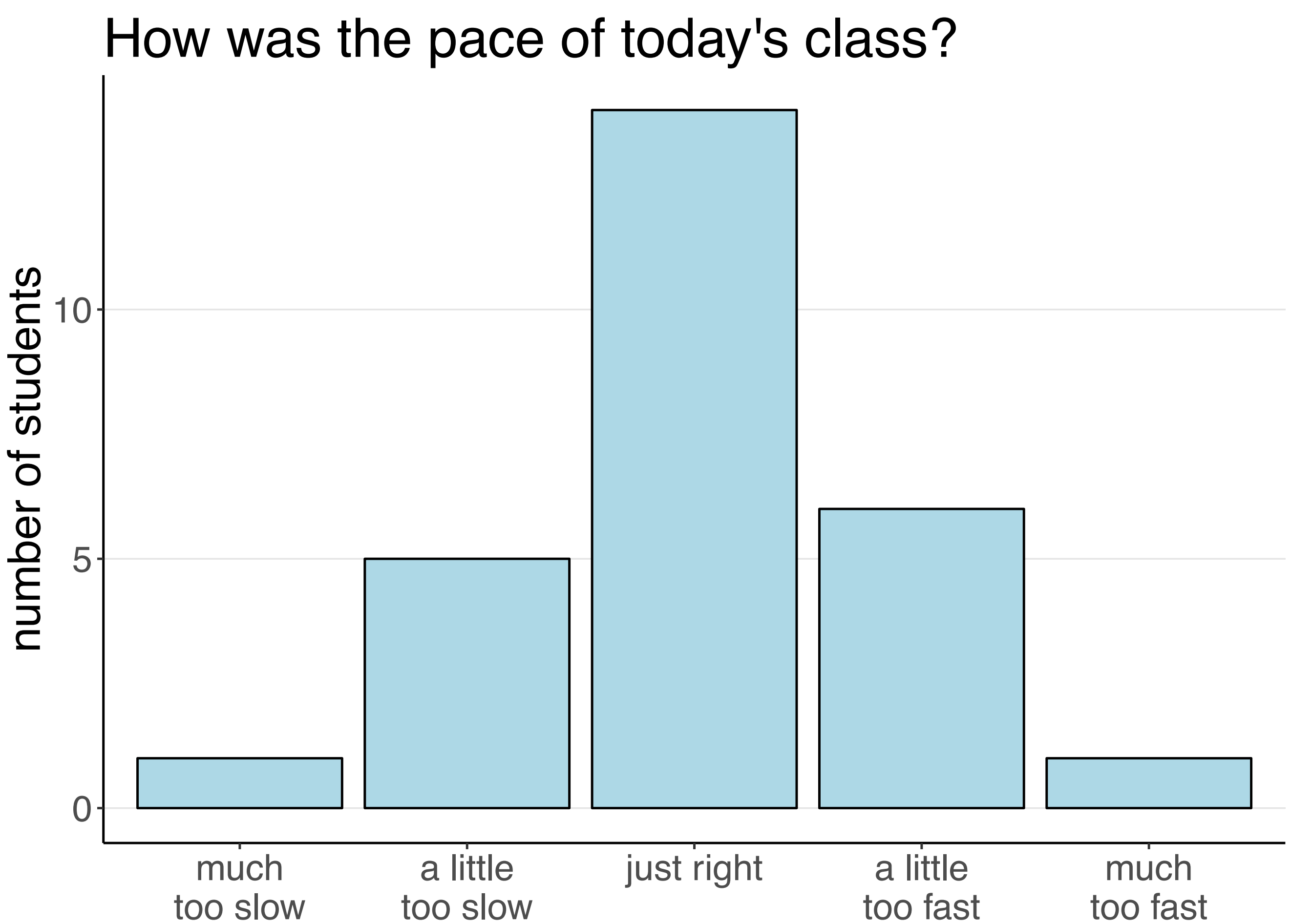


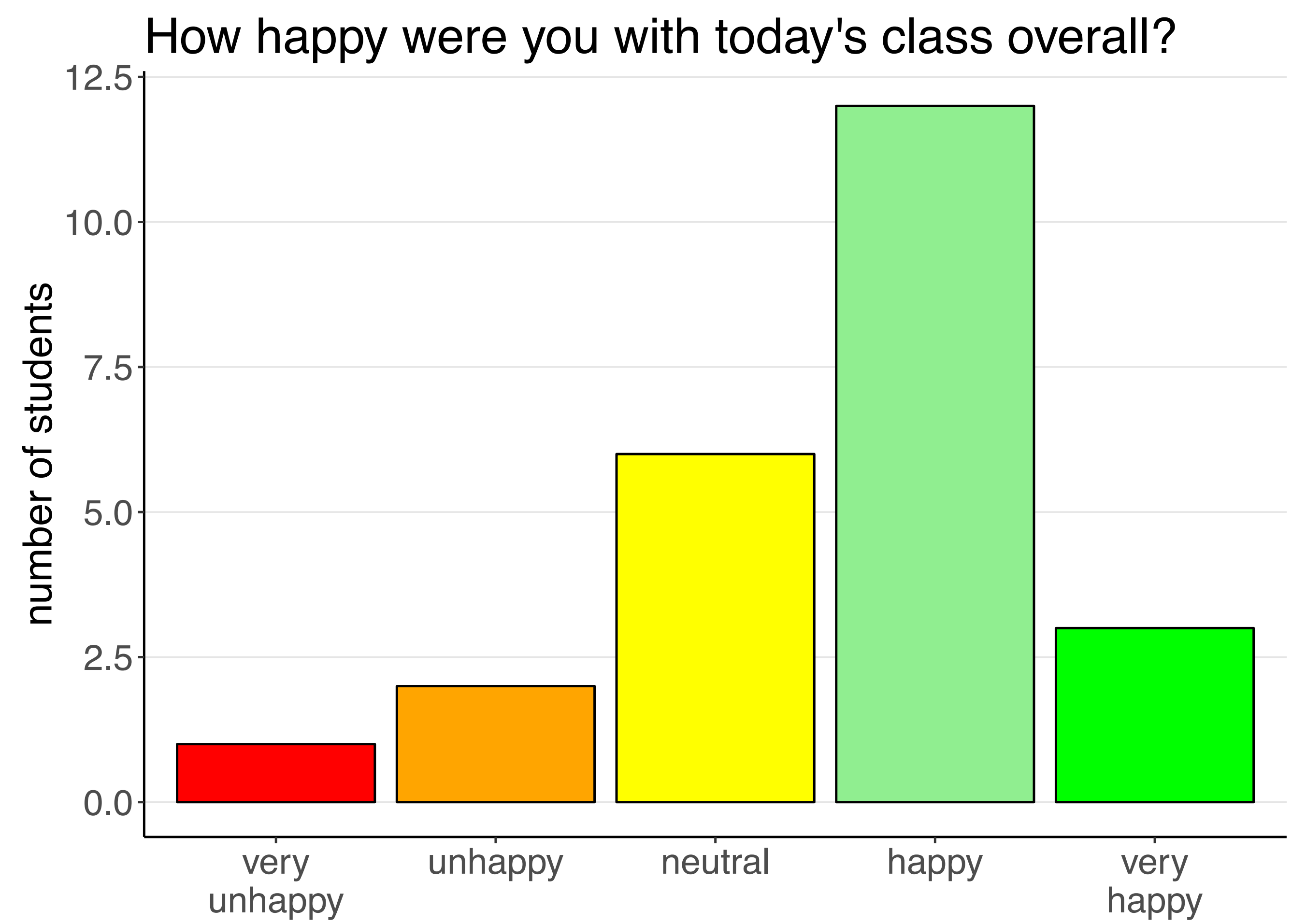
02/08/2019

Logistics

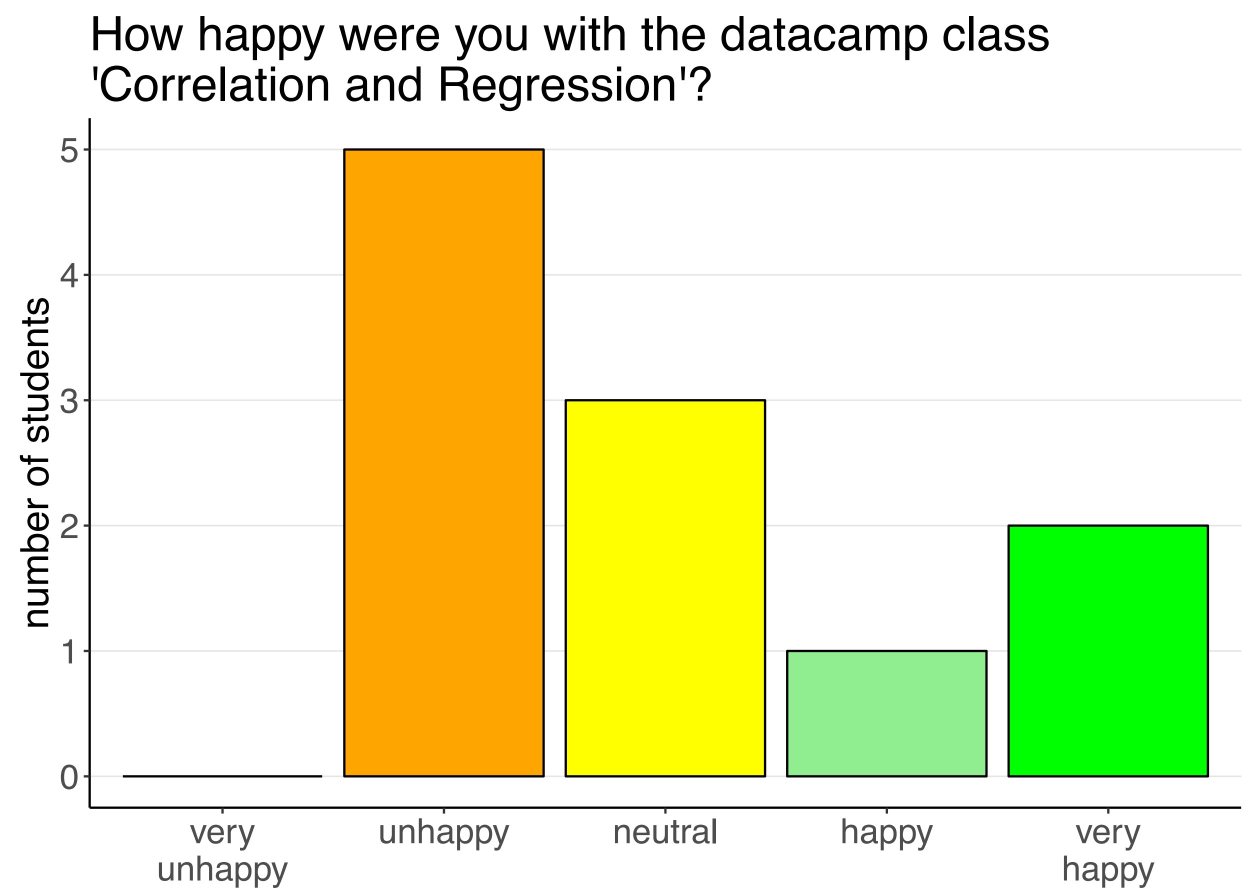
Feedback

Feedback


How was the pace of today's class?
number of students
0
5
10
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?
number of students
0.0
2.5
5.0
7.5
10.0
12.5
very unhappy
unhappy
neutral
happy
very happy

Feedback



we aim for ~6 per week

we won't assign that class again ...

Feedback

I’m having mixed feelings about both the frequency and time spent on questions in class – it makes following the narrative of the lecture that much harder…I’m wondering if there could be a better way for individual questions to be documented and resolved, perhaps via PollEV? Having a text box open throughout the class for people to type in their questions and reserve the last 15 mins of the class to address them? Not sure if this would work, just a thought…

Maybe covering materials and then letting people review it on their own and then come back to class with questions would be helpful. ...

I'll try to find a better balance

Plan for today

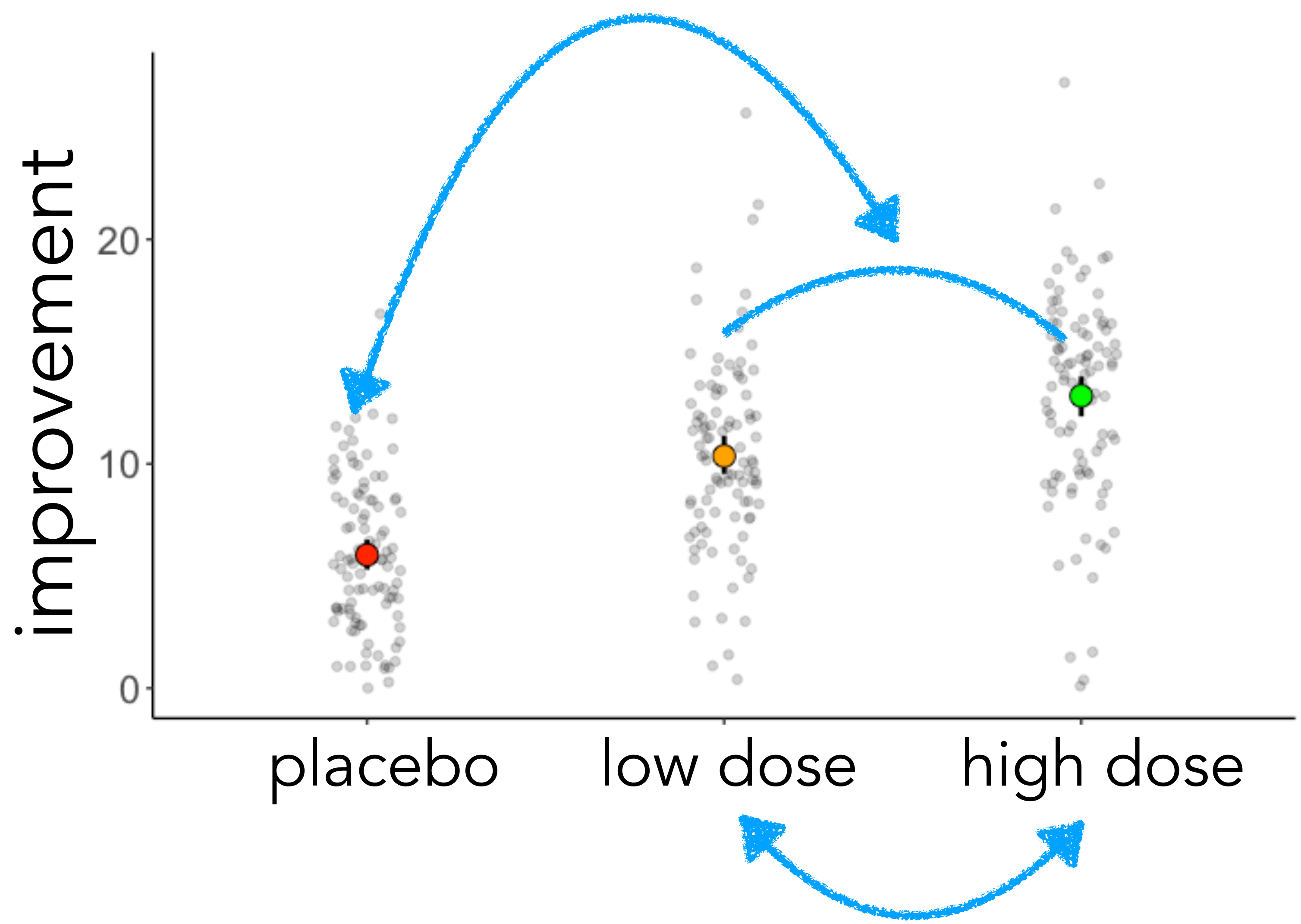
- Contrasts
  - Planned comparisons
- Making decisions based on statistical inference
- Power
- Effect size
- Determining sample size

Planned comparisons

Planned contrasts

must be defined in advance

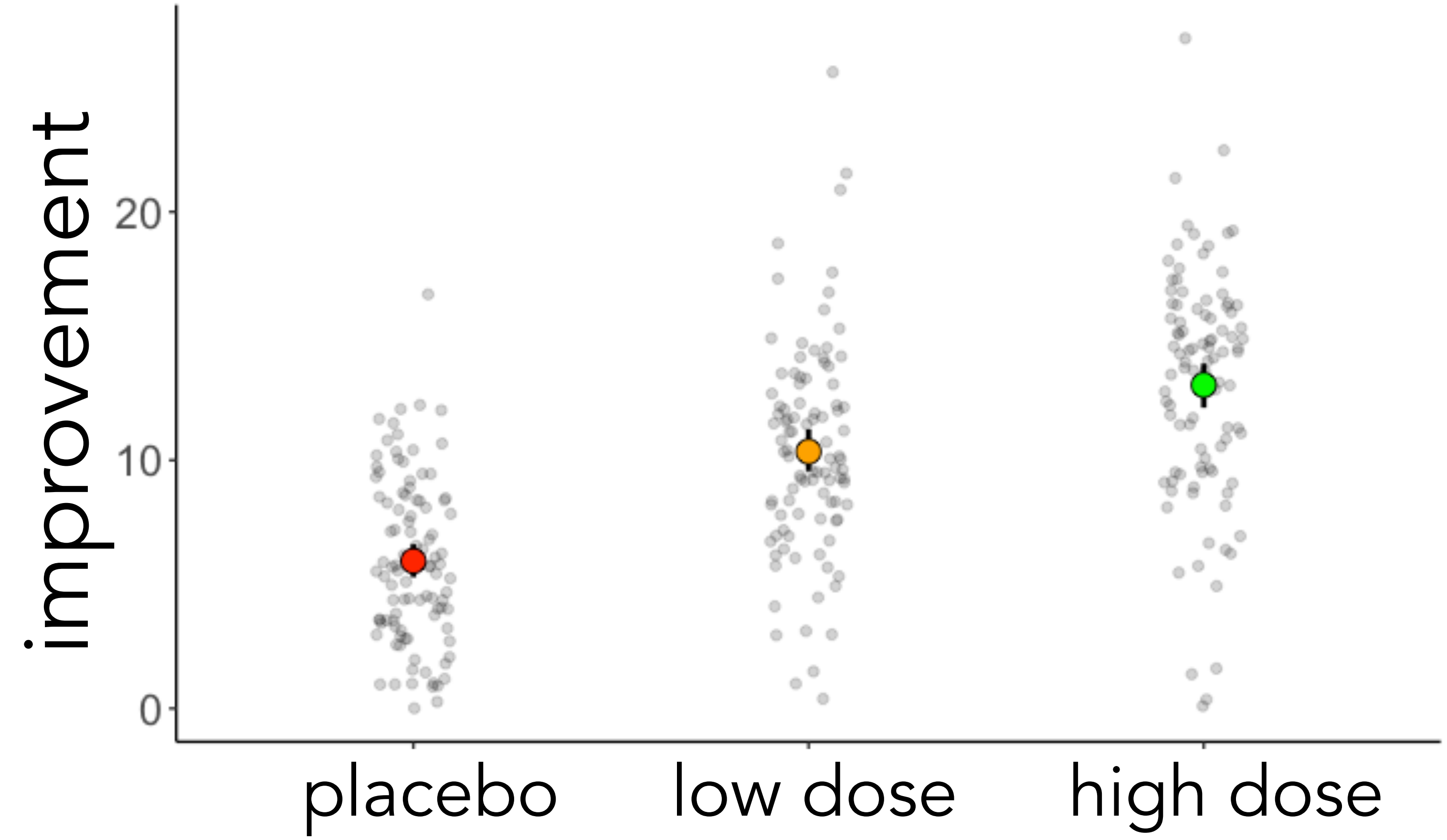
1. Is the treatment different from the placebo?
2. Do the two treatments differ from each other?



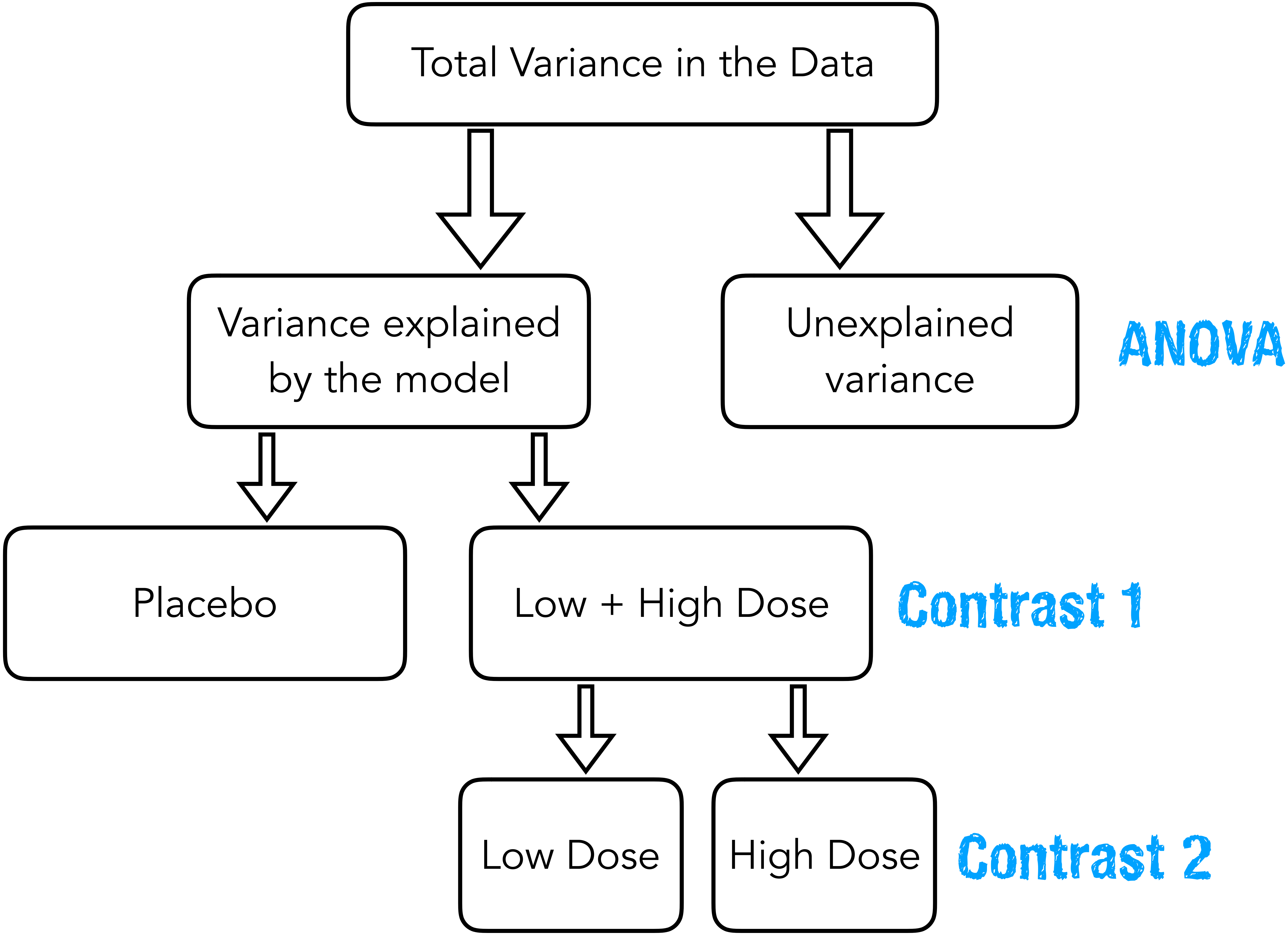
six possible comparisons

if we test all hypotheses, we increase the chance of making a type I error

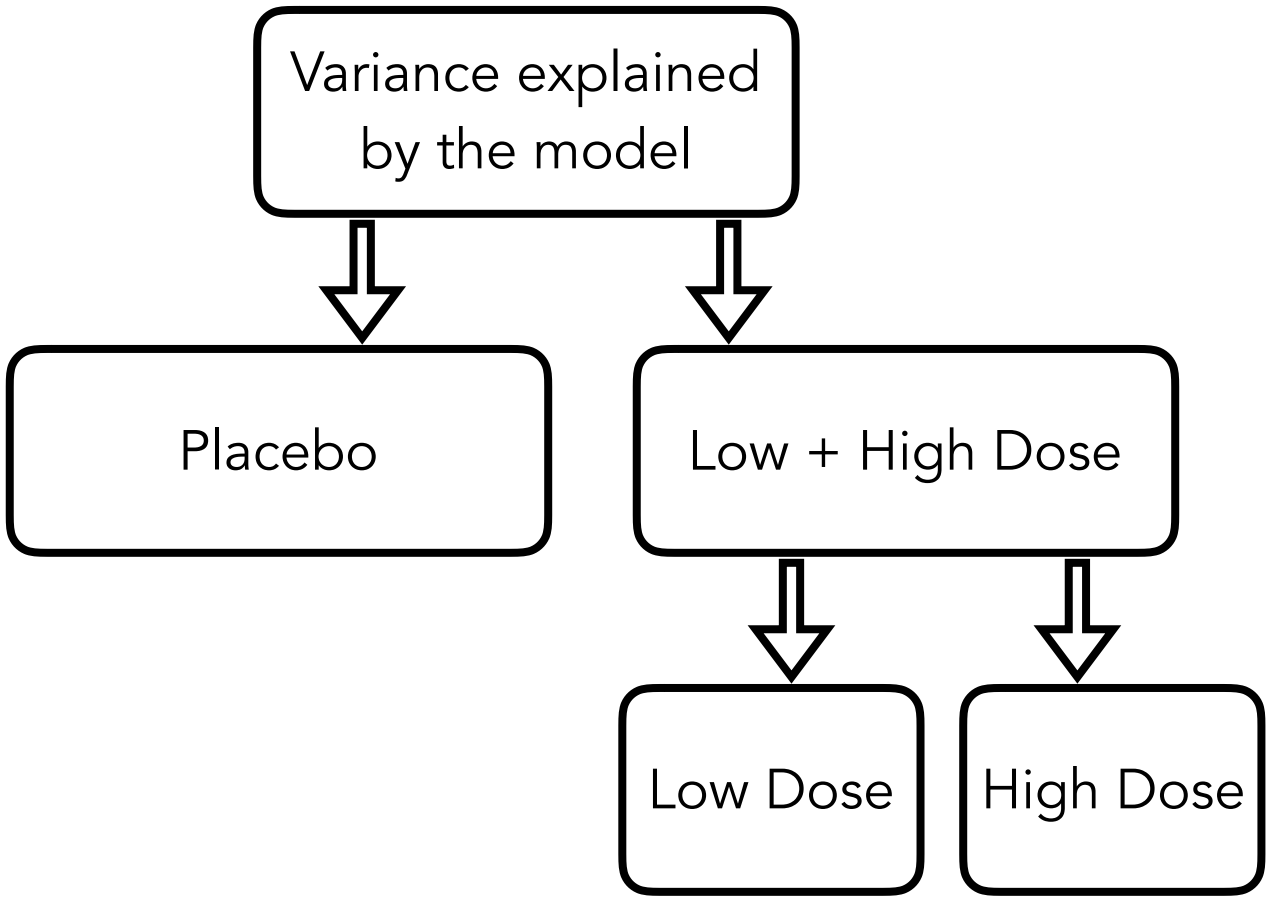
Planned contrasts



Orthogonal contrasts allow us to partition the variance explained by the ANOVA model into a maximum of (# treatment - 1) meaningful and targeted comparisons involving different combinations of means.


Total Variance in the Data
Variance explained by the model
Unexplained variance
ANOVA
Placebo
Low + High Dose
Contrast 1
Low Dose
High Dose
Contrast 2

Planned contrasts

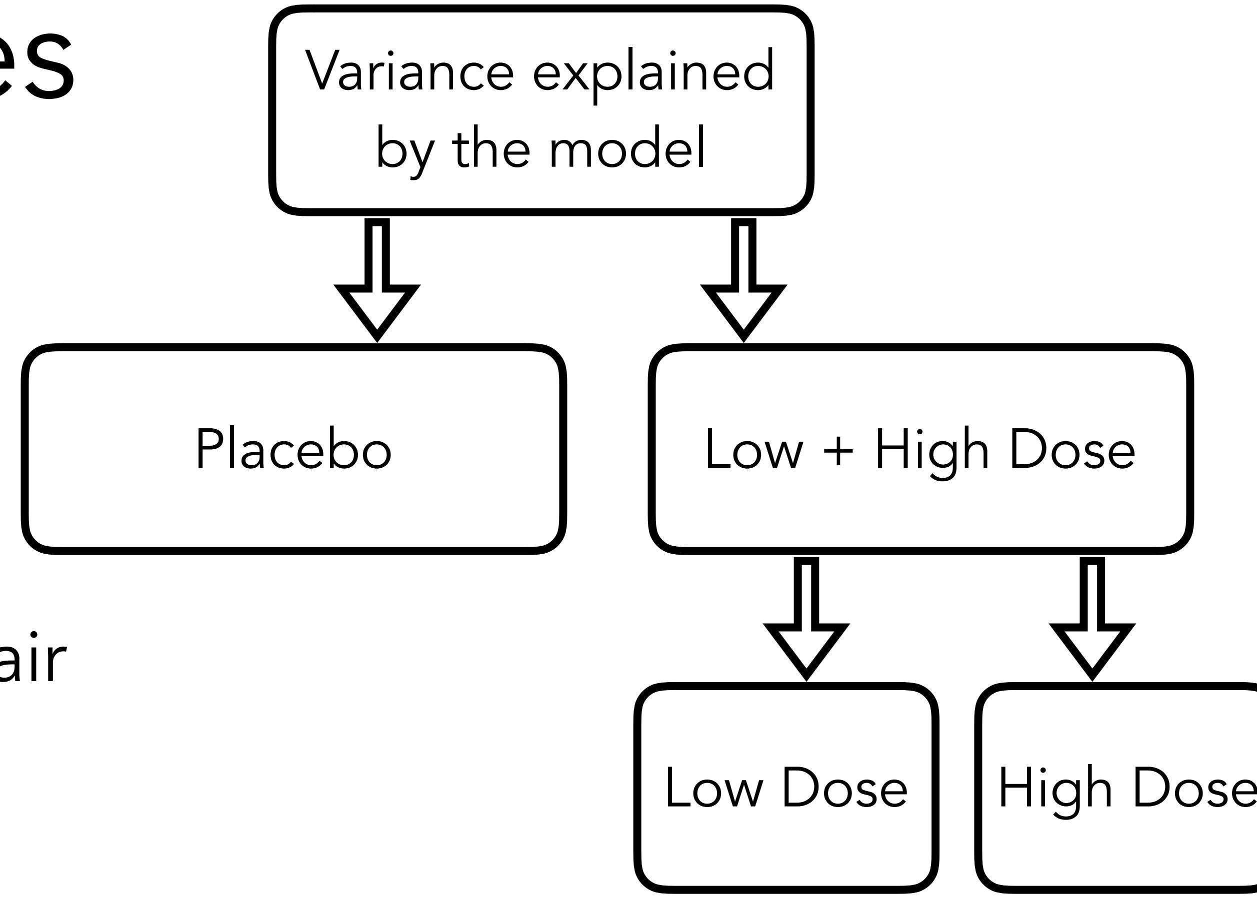


- Each contrast must compare only two "chunks" of variation
- Once a group has been singled out in a contrast, it can't be used in another contrast

Defining orthogonal contrast codes

1. Groups coded with positive weights will be compared against groups with negative weights
2. Assign a 0 to groups who are not involved in a comparison
3. The sum of weights for each comparison should be 0
4. The product of the weights for any pair comparisons should sum to 0

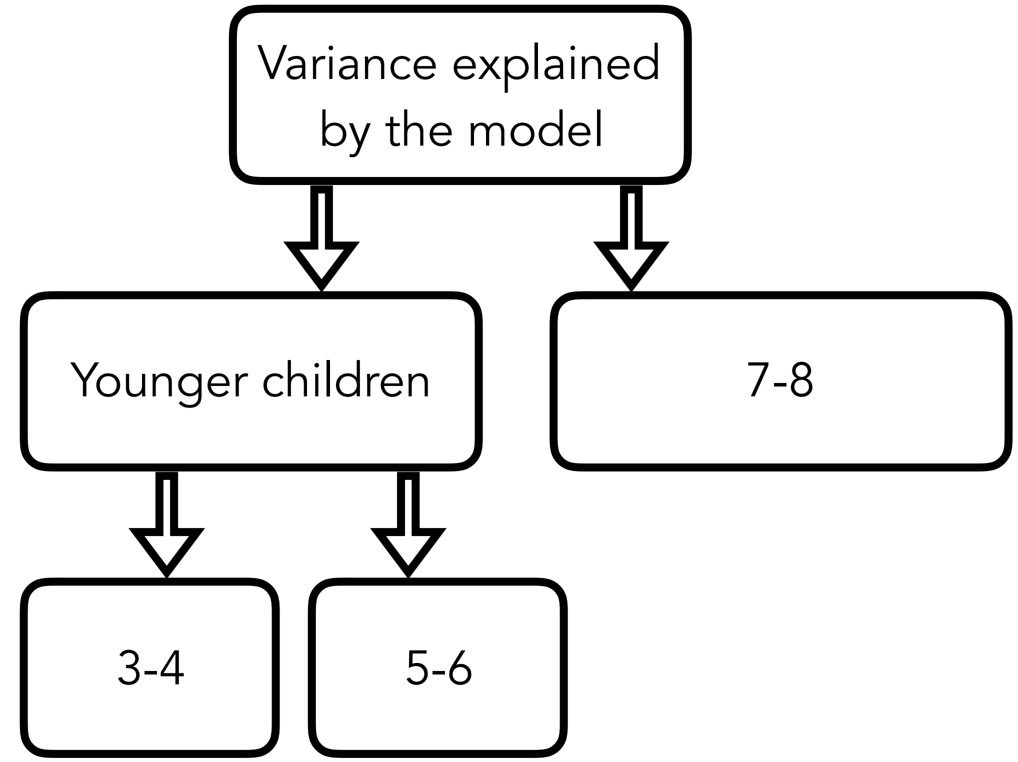
Defining contrast codes



- The sum of weights for each comparison should be 0
- The product of the weights for any pair comparisons should sum to 0

contrast	Placebo	Low	High	sum
Placebo vs. Dose	-1	0.5	0.5	0
Low vs. high	0	-1	1	0
Product	0	-0.5	0.5	0

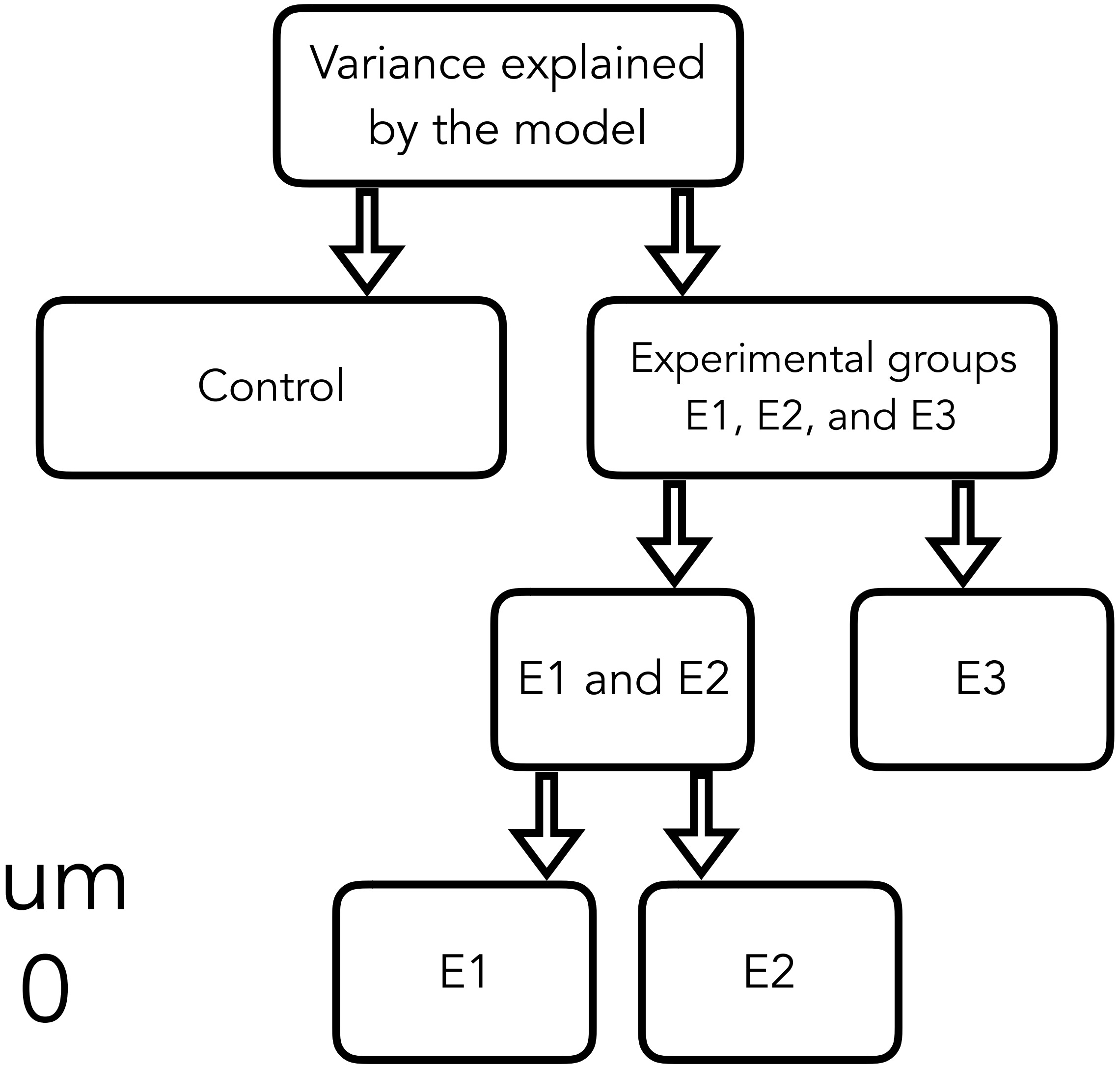
Defining contrast codes



- The sum of weights for each comparison should be 0
- The product of the weights for any pair comparisons should sum to 0

contrast	3-4	5-6	7-8	sum
Young vs. old	-0.5	-0.5	1	0
3-4 vs. 5-6	-1	1	0	0
Product	0.5	-0.5	0	0

Defining contrast codes



- The sum of weights for each comparison should be 0
- The product of the weights for any pair comparisons should sum to 0

contrast	Control	E1	E2	E3
Control vs. E	-3	1	1	1
E1&2 vs. E3	0	-1	-1	2
E1 vs. E2	0	-1	1	0

sum
0
0
0

Check for products

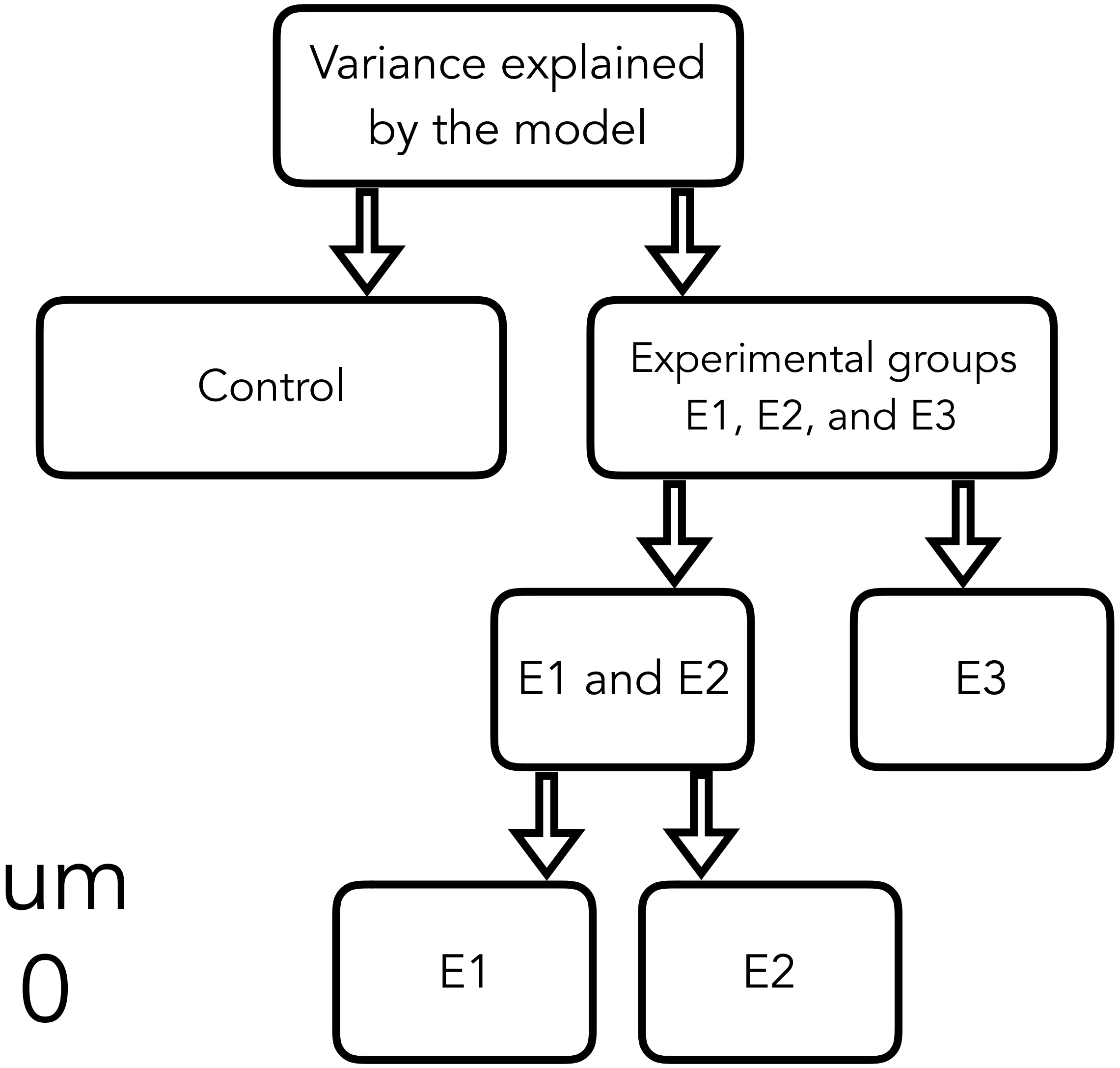
contrast	Control	E1	E2	E3
Control vs. E	-3	1	1	1
E1&2 vs. E3	0	-1	-1	2

product 0 -1 -1 2

sum 0

Defining contrast codes

- The sum of weights for each comparison should be 0
- The product of the weights for any pair comparisons should sum to 0

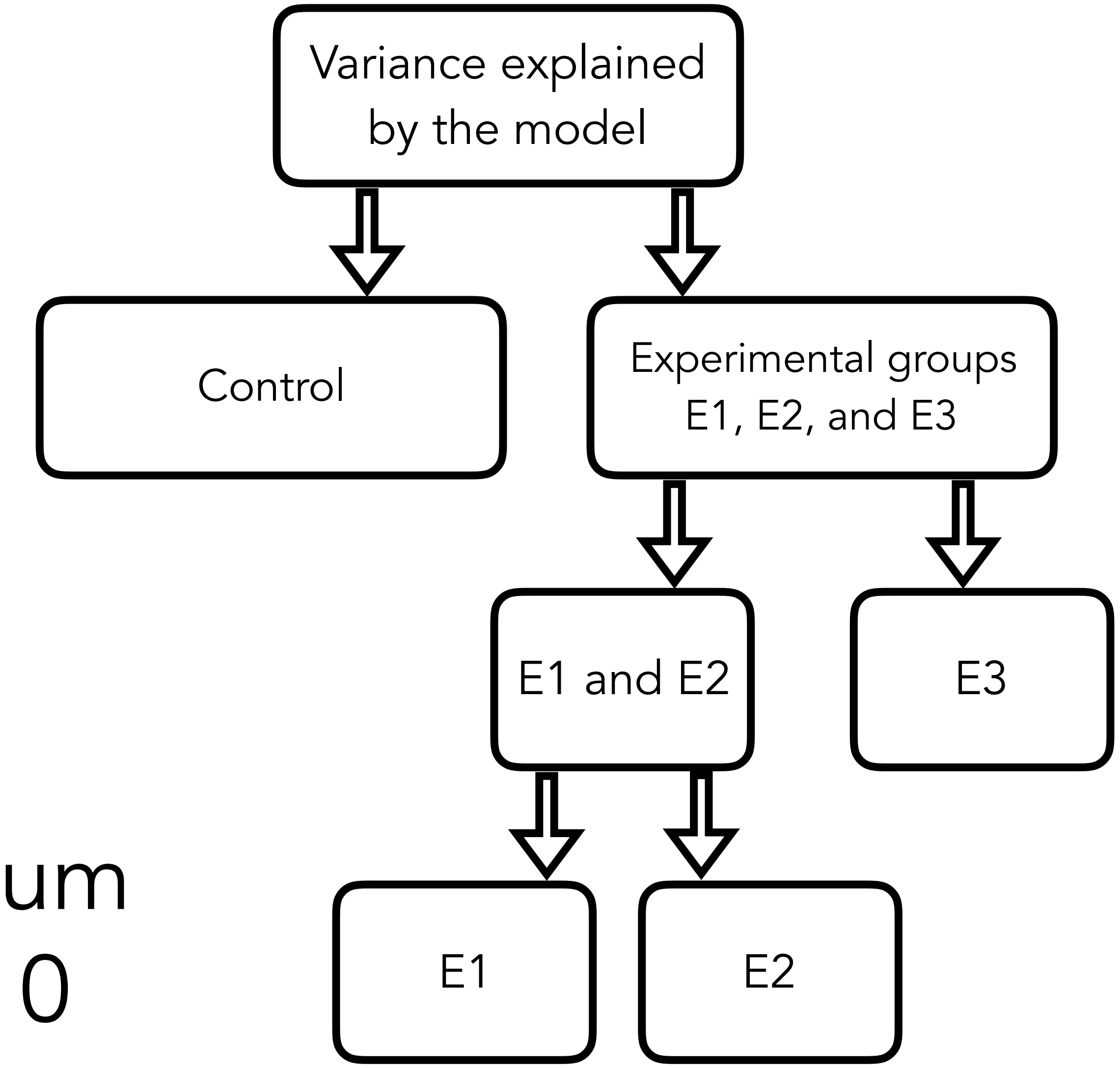


contrast	Control	E1	E2	E3	sum
Control vs. E	-3	1	1	1	0
E1&2 vs. E3	0	-1	-1	2	0
E1 vs. E2	0	-1	1	0	0

Check for products

contrast	Control	E1	E2	E3	sum
Control vs. E	-3	1	1	1	
E1 vs. E2	0	-1	1	0	
product	0	-1	1	0	0

Defining contrast codes



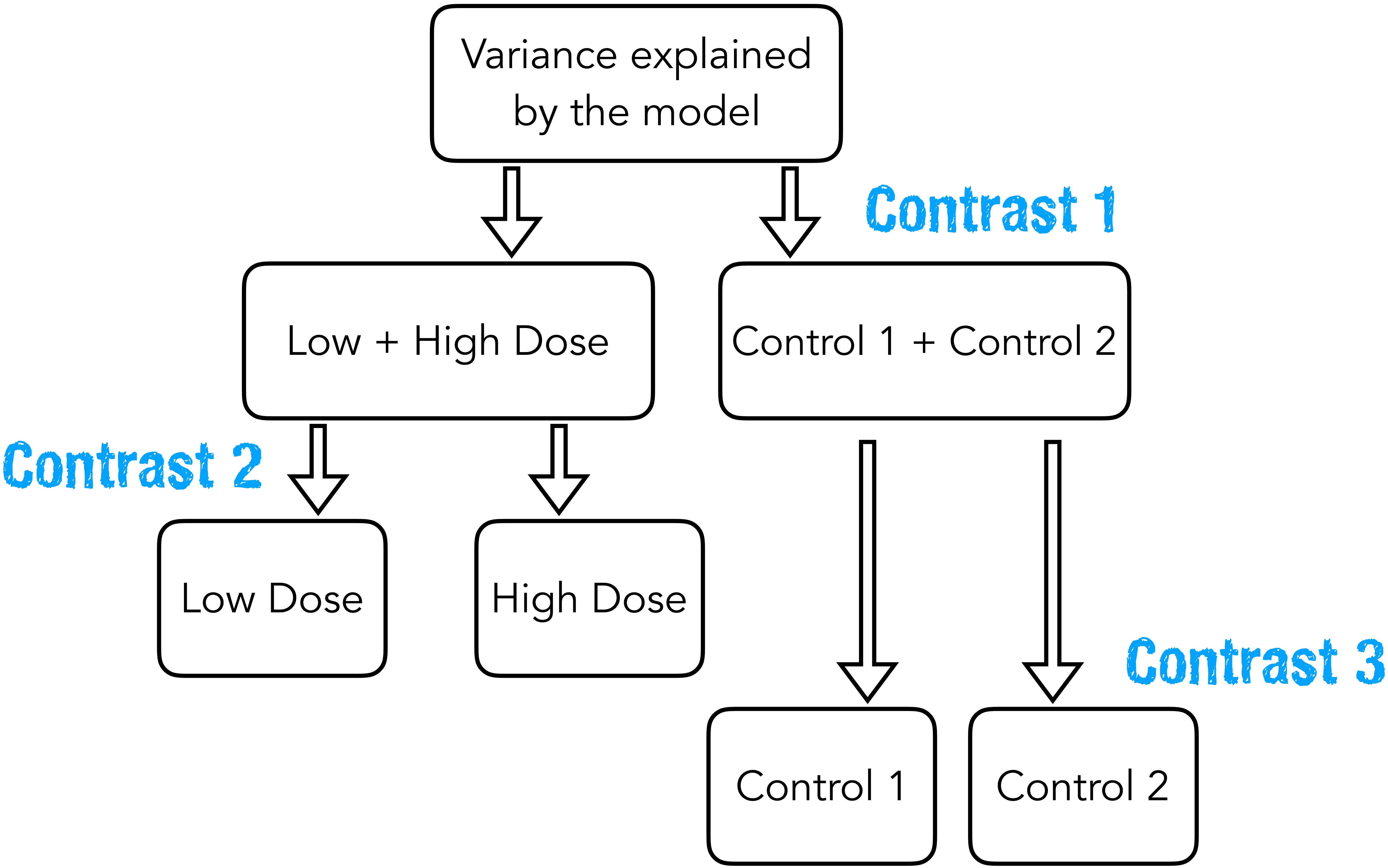
- The sum of weights for each comparison should be 0
- The product of the weights for any pair comparisons should sum to 0

contrast	Control	E1	E2	E3	sum
Control vs. E	-3	1	1	1	0
E1&2 vs. E3	0	-1	-1	2	0
E1 vs. E2	0	-1	1	0	0

Check for products

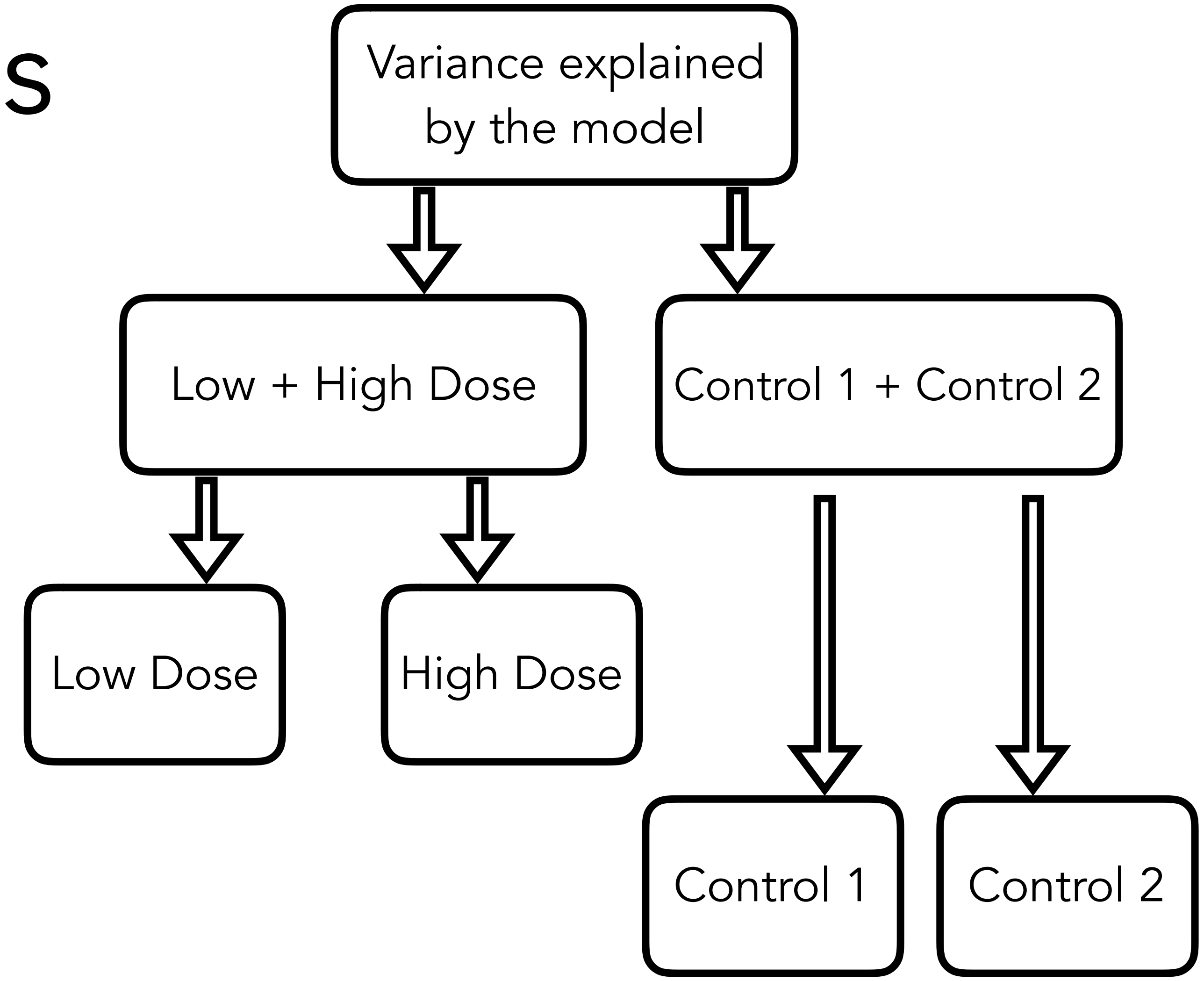
contrast	Control	E1	E2	E3	sum
E1&2 vs. E3	0	-1	-1	2	
E1 vs. E2	0	-1	1	0	
product	0	1	-1	0	0

contrasts are orthogonal


Variance explained by the model
Contrast 1
Low + High Dose
Control 1 + Control 2
Contrast 2
Low Dose
High Dose
Contrast 3
Control 1
Control 2

Defining contrast codes

- The sum of weights for each comparison should be 0
- The product of the weights for any pair comparisons should sum to 0



contrast	low	high	C1	C2
low/high vs. C1/C2				
low vs. high				
C1 vs. C2				

Planned contrasts in R

```r
library("emmeans")

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

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

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

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

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

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

```
[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
```

Post hoc tests

```r
library("emmeans")

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

# post hoc tests
leastsquare = emmeans(fit, "group")
pairs(leastsquare,
      adjust = "bonferroni")
```

```
contrast    estimate         SE df t.ratio p.value
 3-4 - 5-6  -1.606009 0.5475962 57  -2.933  0.0145
 3-4 - 10+ -16.896546 0.5475962 57 -30.856  <.0001
 5-6 - 10+ -15.290537 0.5475962 57 -27.923  <.0001

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

all pairwise tests between groups

Post hoc tests

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

# post hoc tests
leastsquare = emmeans(fit, c("hand", "skill"))
pairs(leastsquare,
      adjust = "bonferroni")
```

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

Dummy coding

Beware of misinterpretation

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

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   5.9415     0.4111  14.451  < 2e-16 ***
handneutral   4.4051     0.5815   7.576 4.55e-13 ***
handgood      7.0849     0.5815  12.185  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

reference category

bad	neutral	good
5.94	10.35	13.03

```
lm(formula = balance ~ hand * skill, data = df.poker)
```

```
Coefficients:
                        Estimate Std. Error t value Pr(>|t|)
(Intercept)               4.5866     0.5686   8.067 1.85e-14 ***
handneutral               5.2572     0.8041   6.538 2.75e-10 ***
handgood                  9.2110     0.8041  11.455  < 2e-16 ***
skillexpert               2.7098     0.8041   3.370 0.000852 ***
handneutral:skillexpert  -1.7042     1.1372  -1.499 0.135038
handgood:skillexpert     -4.2522     1.1372  -3.739 0.000222 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

reference category

skill	bad	neutral	good
average	4.59	9.84	13.80
expert	7.30	10.85	12.26

Effect coding

Effect coding

```r
lm(formula = balance ~ hand, data = df.poker,
   contrasts = list(hand = "contr.sum"))
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   9.7715     0.2374  41.165   <2e-16 ***
hand1        -3.8300     0.3357 -11.409   <2e-16 ***
hand2         0.5751     0.3357   1.713   0.0877 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

reference

grand mean = 9.77

```r
lm(formula = balance ~ hand * skill, data = df.poker,
   contrasts = list(hand = "contr.sum", skill = "contr.sum"))
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)     9.7715     0.2321  42.096  < 2e-16 ***
hand1          -3.8300     0.3283 -11.667  < 2e-16 ***
hand2           0.5751     0.3283   1.752  0.08083 .
skill1         -0.3622     0.2321  -1.560  0.11978
hand1:skill1   -0.9927     0.3283  -3.024  0.00271 **
hand2:skill1   -0.1406     0.3283  -0.428  0.66867
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

reference

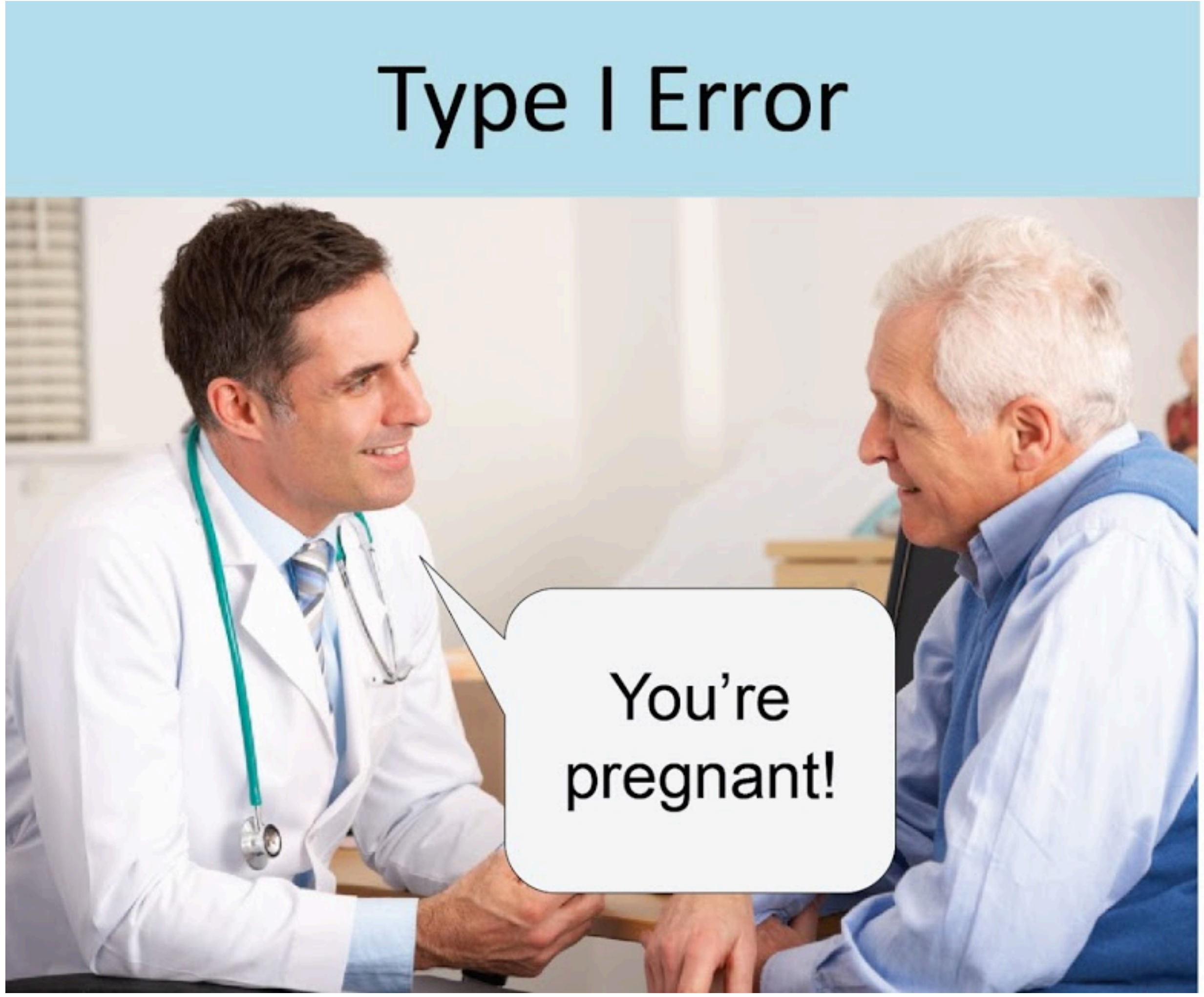
grand mean = 9.77

Note: The last level in each factor is dropped.

Coding schemes

- **Dummy coding:**
  - the reference category is coded as 0
  - represented by the intercept
  - all other categories are compared to this reference category
- **Effect coding:**
  - the intercept is the grand mean
  - all other categories are compared to the grand mean

Making decisions



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

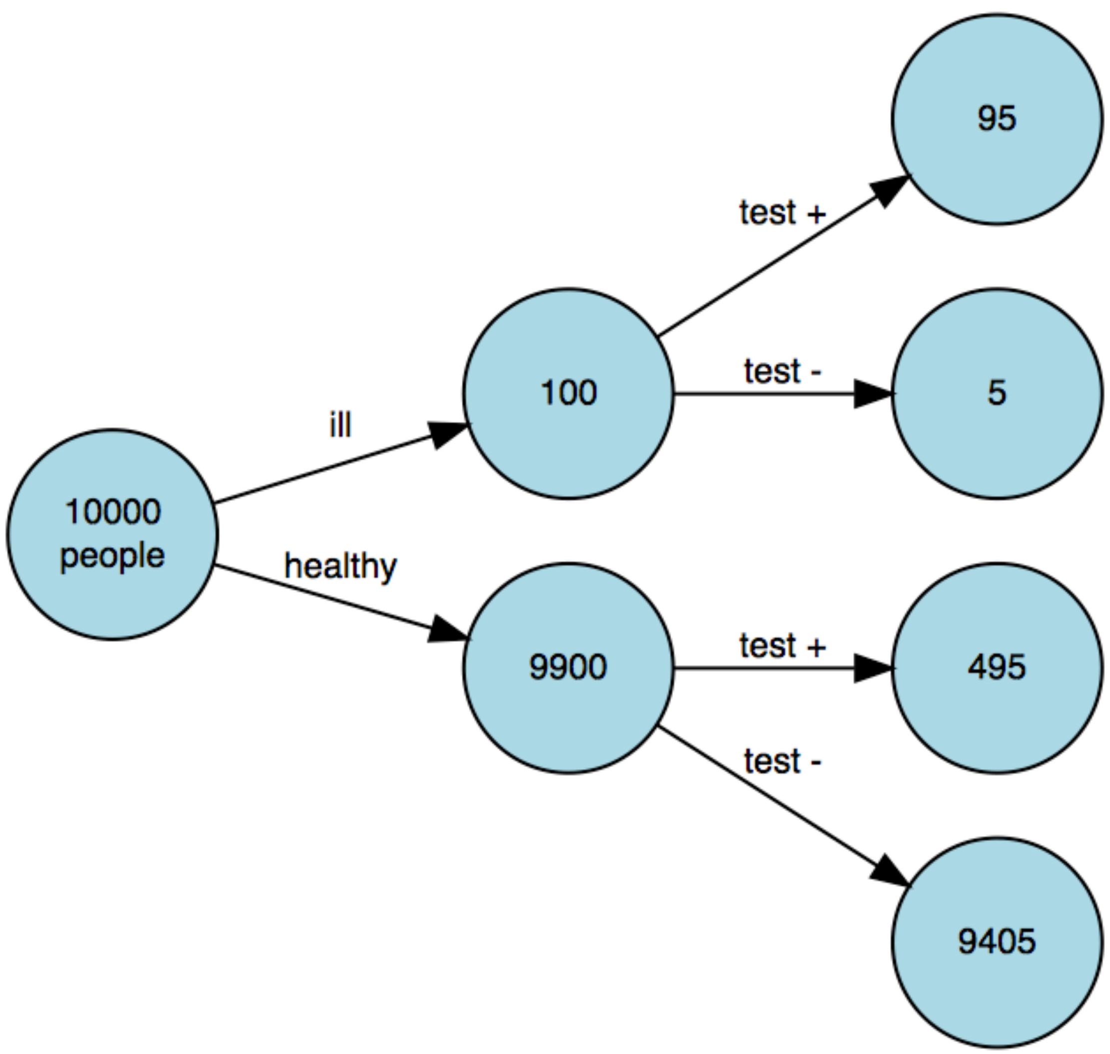
Type I Error: Falsely rejecting the null hypothesis (even though it is true).

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.



Power

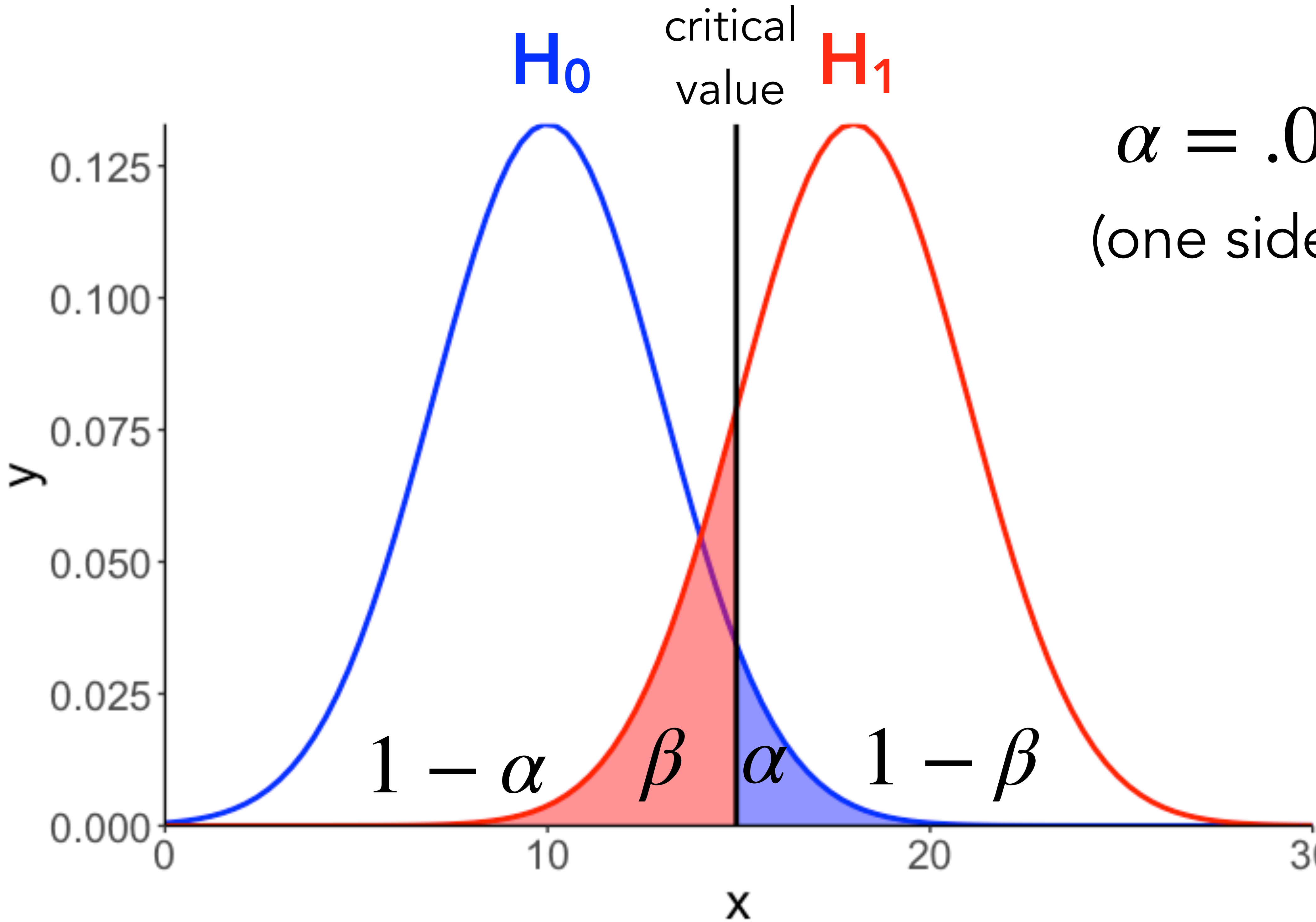
true positive — Sensitivity — $p(\text{reject } H_0 | H_1 \text{ is true})$ — $\boxed{1-\beta}$

false negative — Type II error — $p(\text{not reject } H_0 | H_1 \text{ is true})$ — β

false positive — Type I error — $p(\text{reject } H_0 | H_0 \text{ is true})$ — α

true negative — $p(\text{not reject } H_0 | H_0 \text{ is true})$ — Specificity — $1-\alpha$

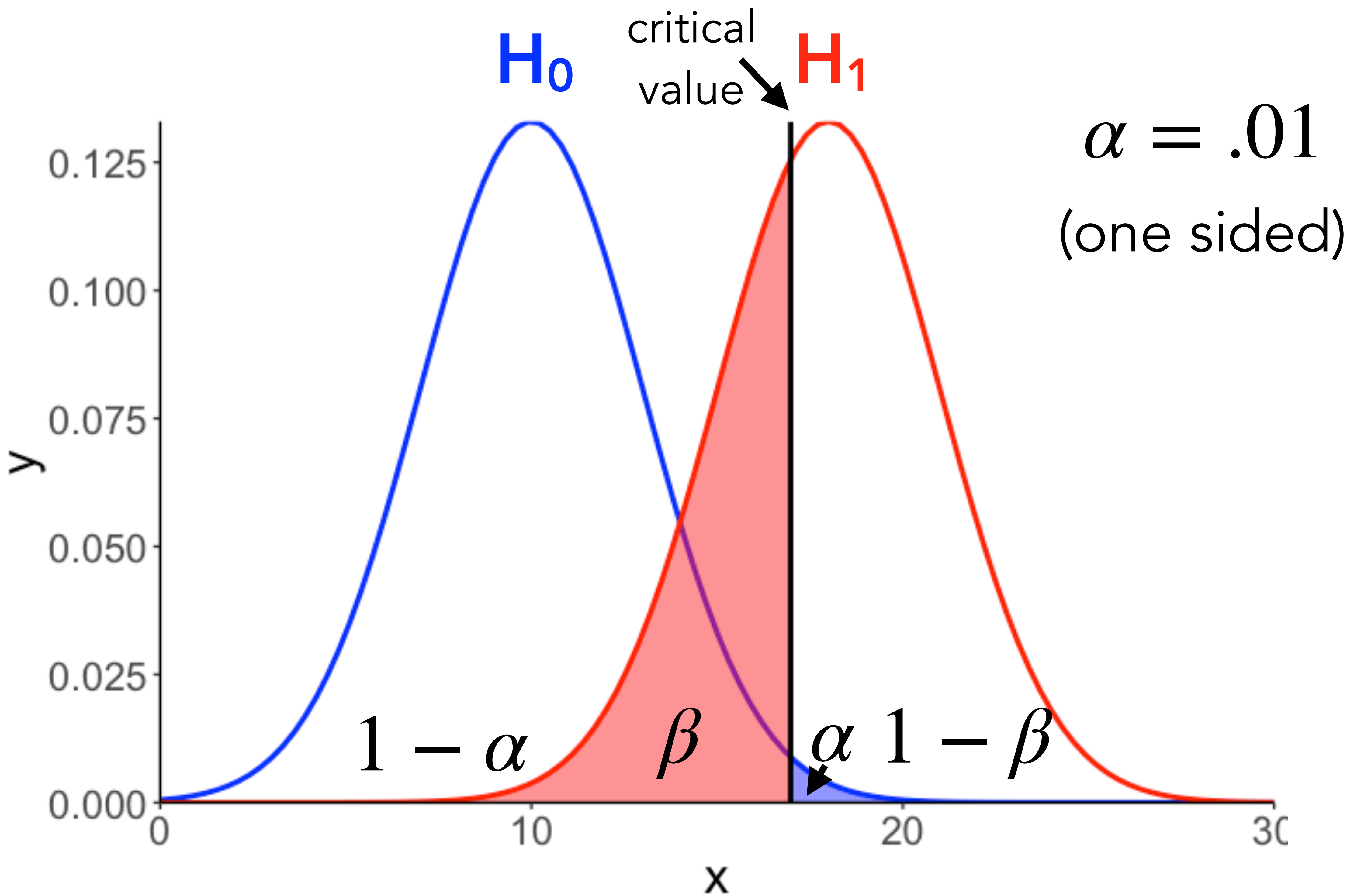
What affects power?



$\alpha = .05$

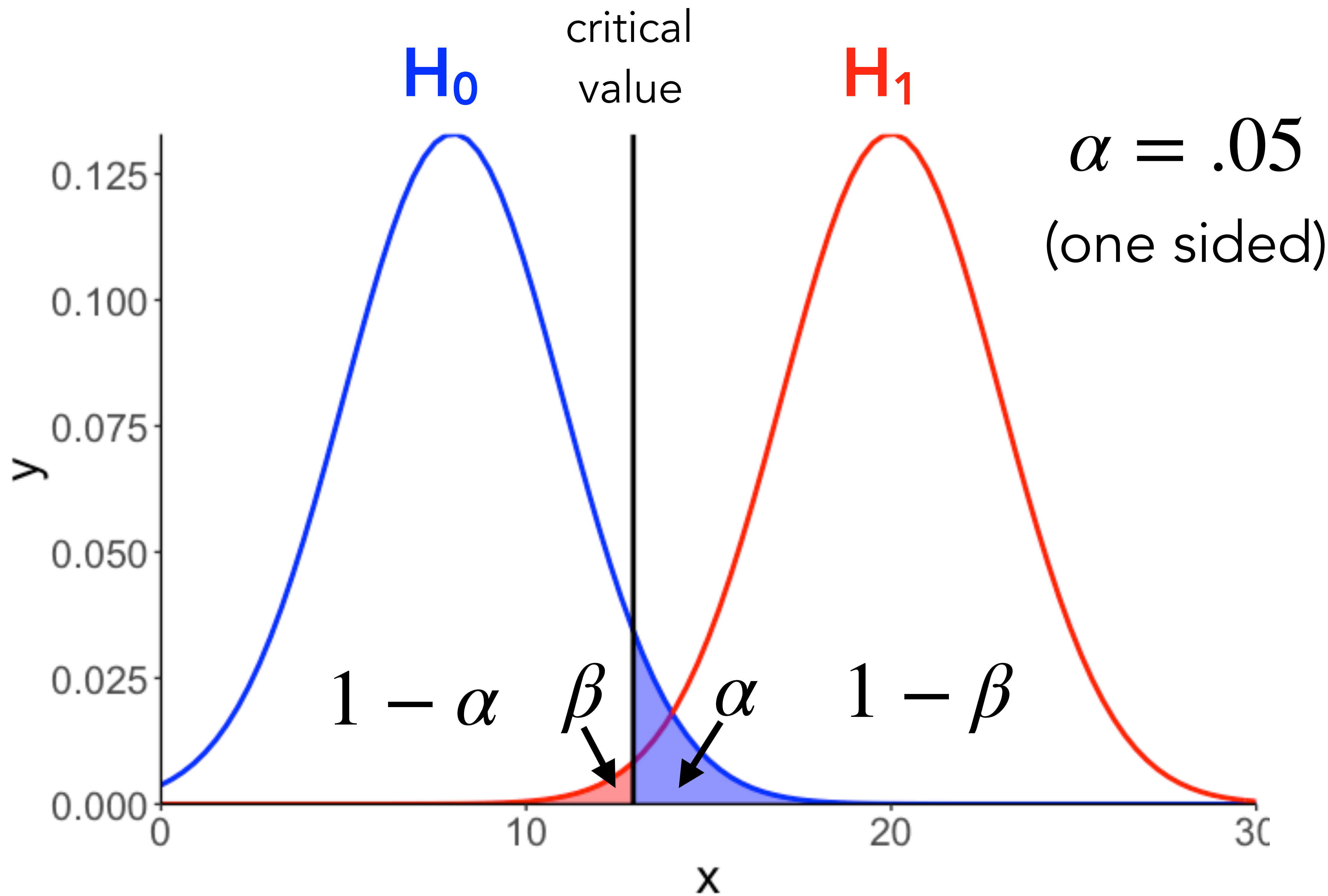
(one sided)

What affects power?



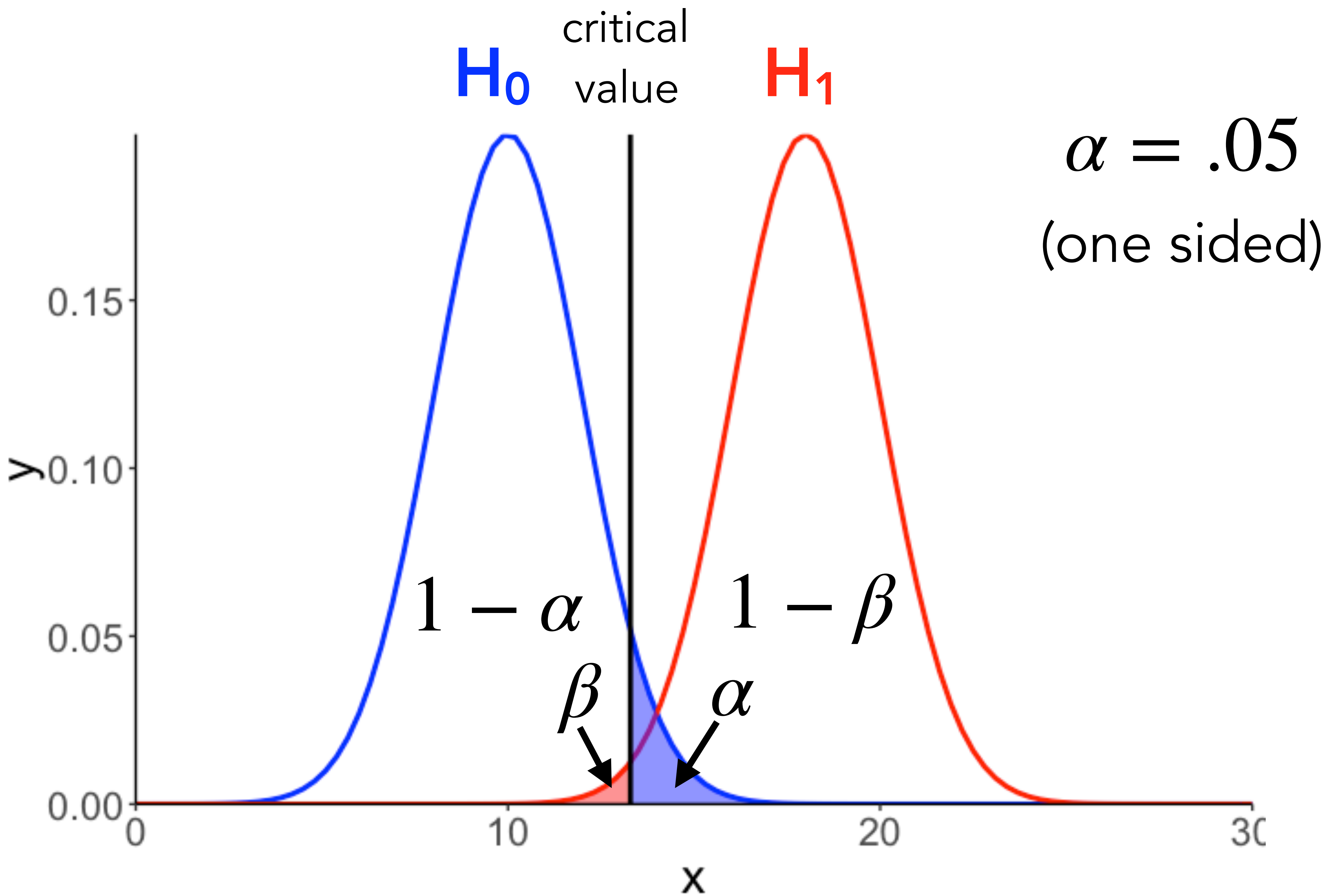
α affects power

What affects power?



distance between means affects power

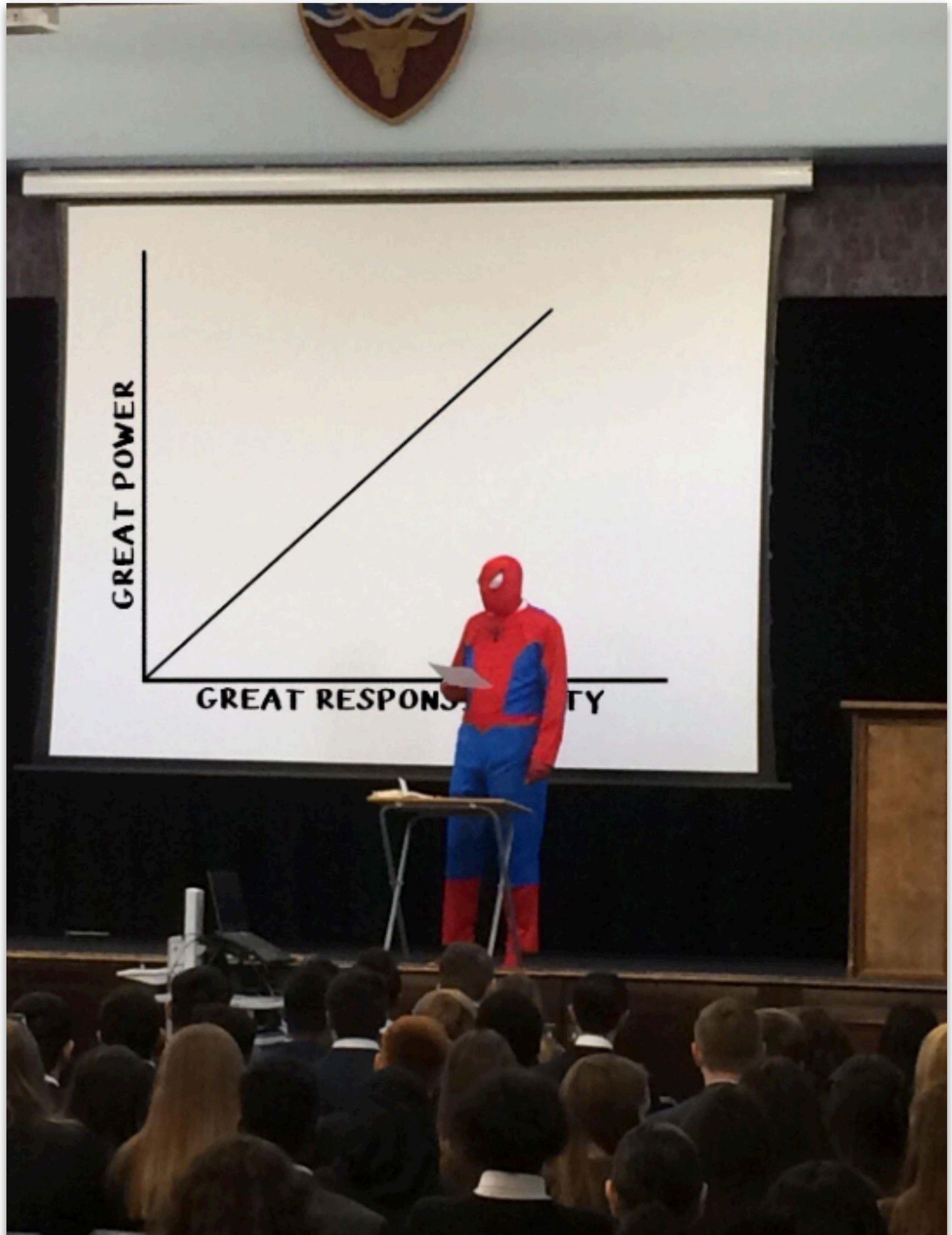
What affects power?



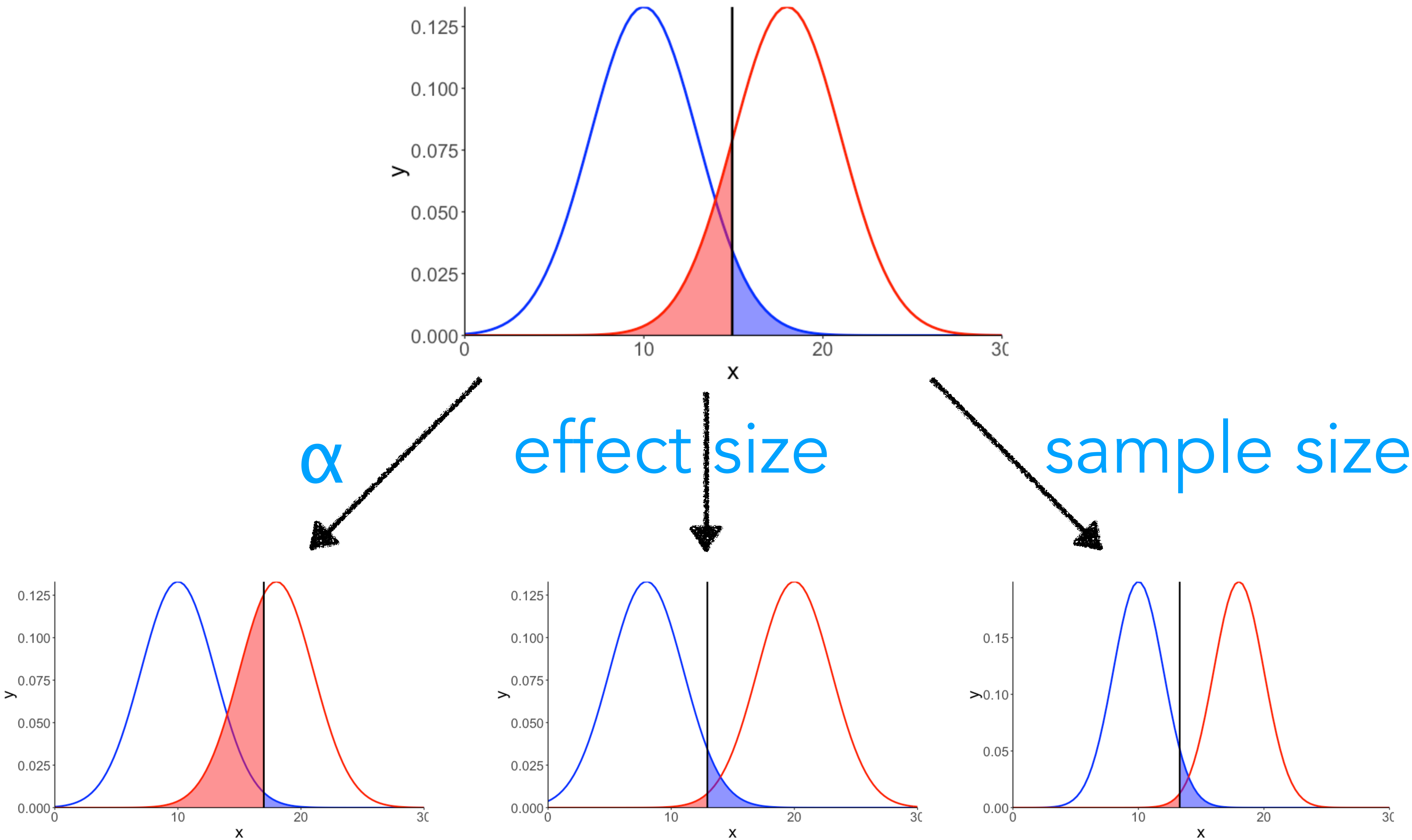
variance affects power

Calculating power

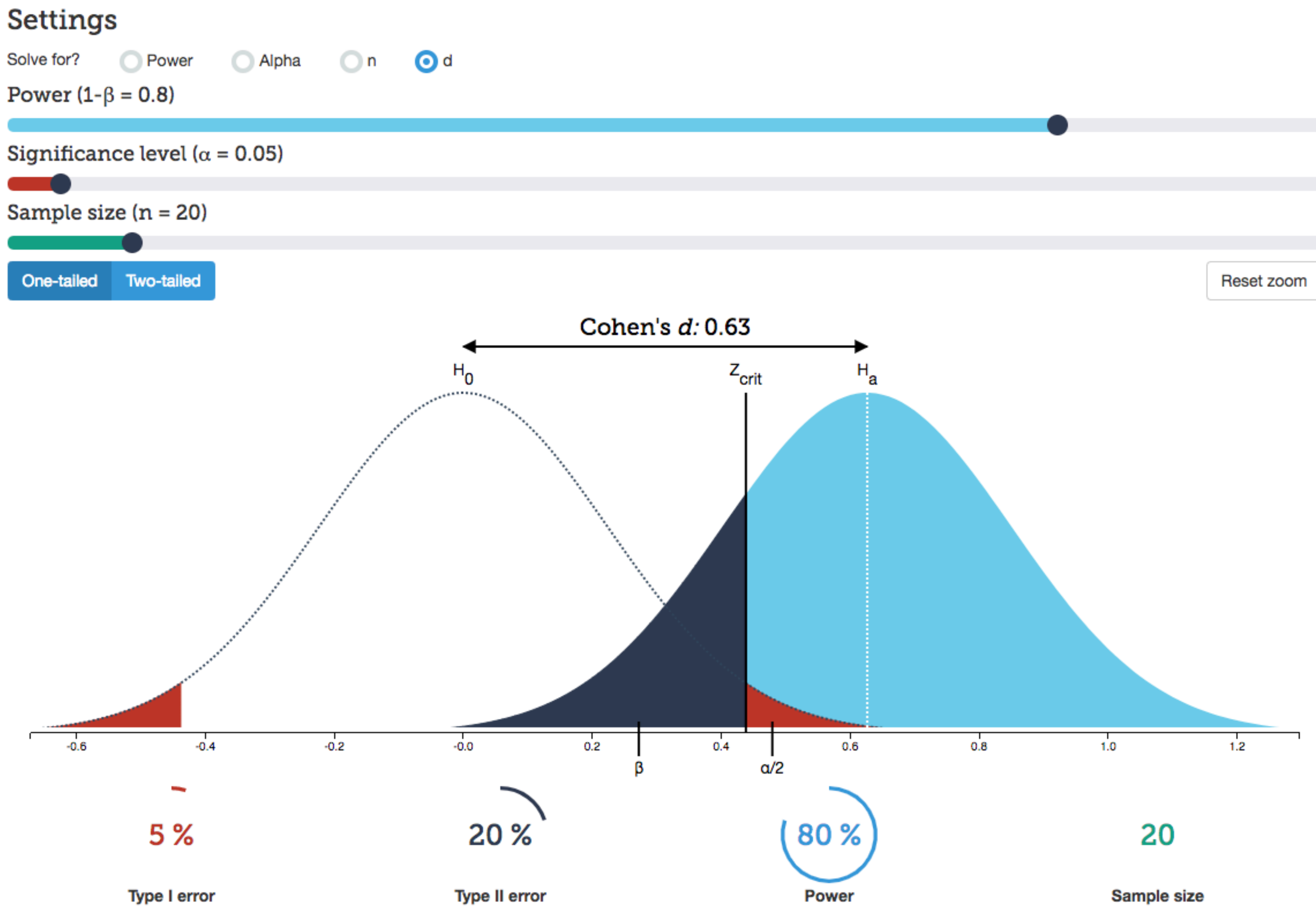
With great power comes …



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

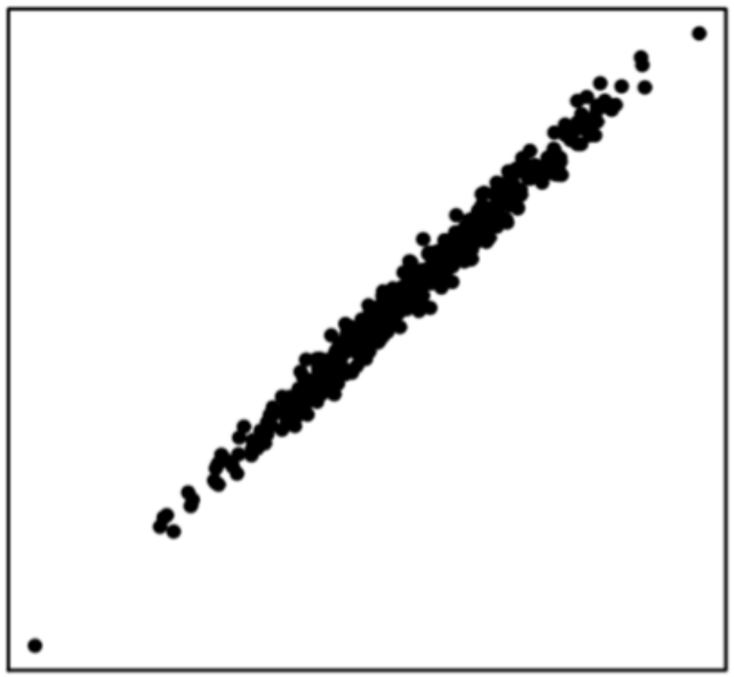
SSE(A)

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

SSE = sum of squared errors

Common effect sizes

Relationships between variables



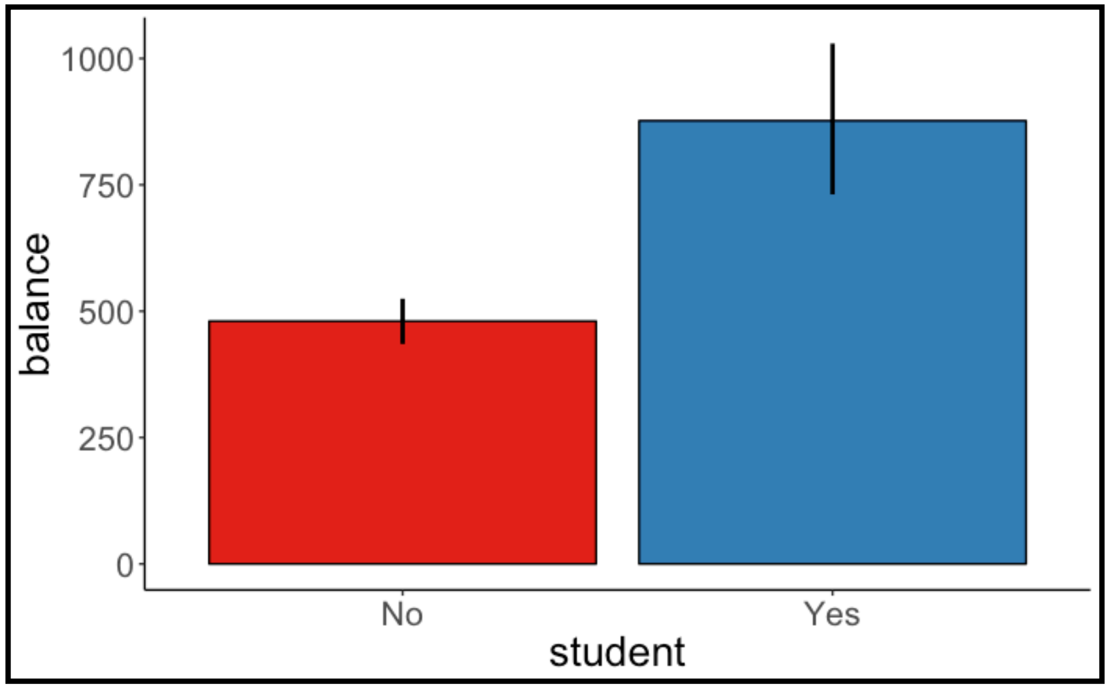
r correlation

R^2 variance explained

η^2 eta squared

η_p^2 partial eta squared

Differences between groups



Cohen's d

Correlation

Pearson correlation

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

standardized covariation

(dividing by the standard deviations)

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

Coefficient of determination R^2

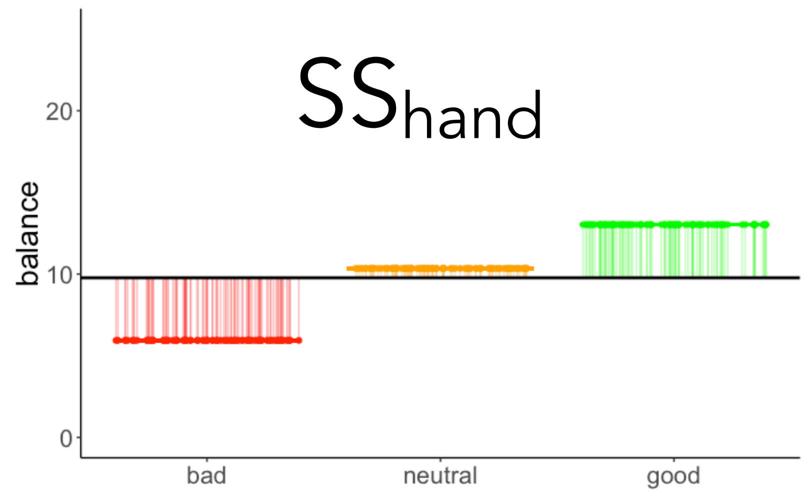
$$R^2 = 1 - \frac{\text{SS}_{\text{residual}}}{\text{SS}_{\text{total}}}$$

equivalent to PRE for the special case in which

$$\text{PRE} = 1 - \frac{\text{SSE}(A)}{\text{SSE}(C)}$$ **Model C**: $Y_i = \beta_0 + \epsilon_i$

ANOVA effect size measures

Hand variance



Skill variance

eta squared

$$\eta^2 = \frac{SS_{treatment}}{SS_{total}}$$

$SS_{treatment}$ depends on what other treatments are included in the ANOVA

as more variables are added, the $SS_{treatment}$ for each variable will decrease (as the variance to be explained is now shared)

ANOVA effect size measures

partial eta squared

$$\eta_p^2 = \frac{SS_{treatment}}{SS_{treatment} + SS_{residual}}$$

less dependent on whether other variables are added to the model

has a less straightforward interpretation ...

η^2 and η_p^2 are identical for a one-way ANOVA

ANOVA effect size measures in R

```r
library("lsr")

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

etaSquared(fit)
```

```
        eta.sq eta.sq.part
hand 0.3311076   0.3311076
```

identical for one-way ANOVA

ANOVA effect size measures in R

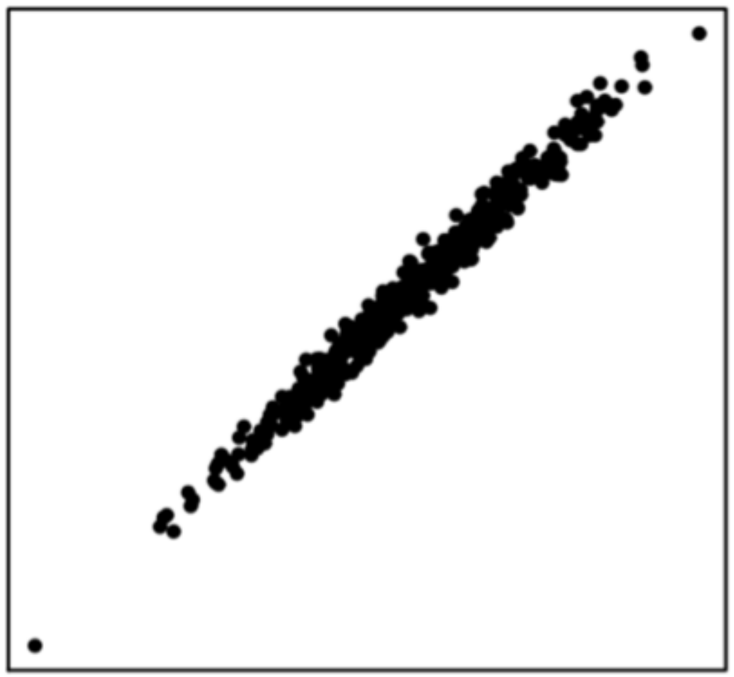
```r
library("lsr")

fit = lm(formula = balance ~ hand * skill,
         data = df.poker)

etaSquared(fit)
```

```
                eta.sq eta.sq.part
hand       0.331107585 0.343119717
skill      0.005191225 0.008123029
hand:skill 0.029817351 0.044925866
```

different for two-way ANOVA

Common effect sizes

Relationships between variables



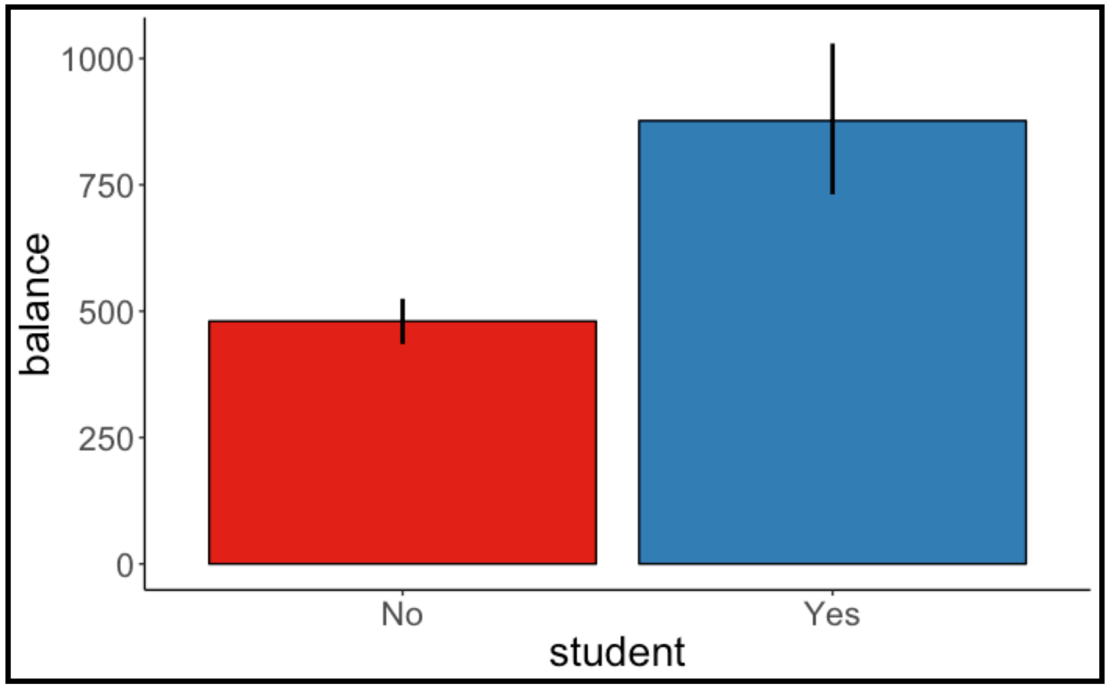
r correlation

R^2 variance explained

η^2 eta squared

η_p^2 partial eta squared

Differences between groups



Cohen's d

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

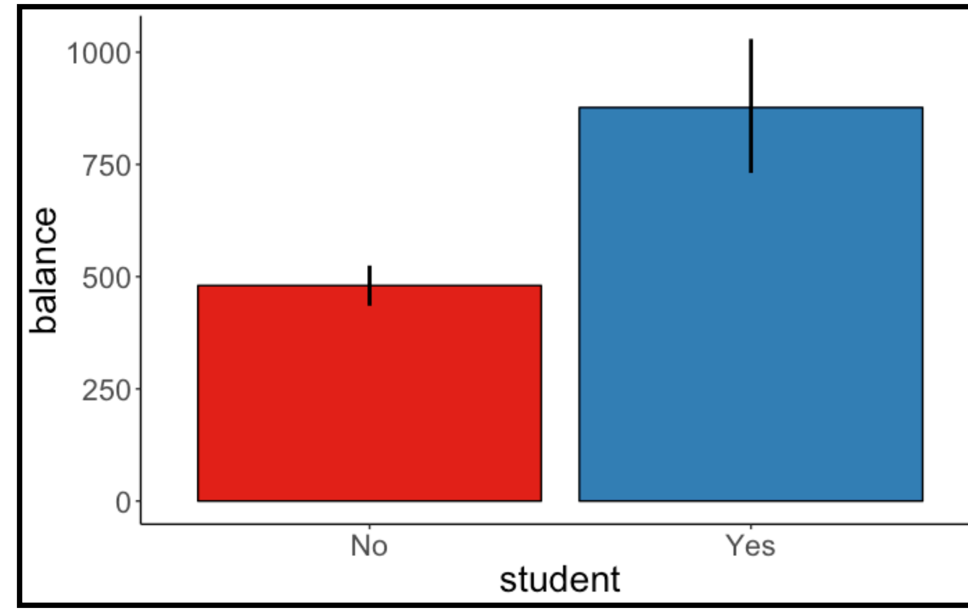
Difference between two means in pooled standard deviation

Cohen's *d* in R

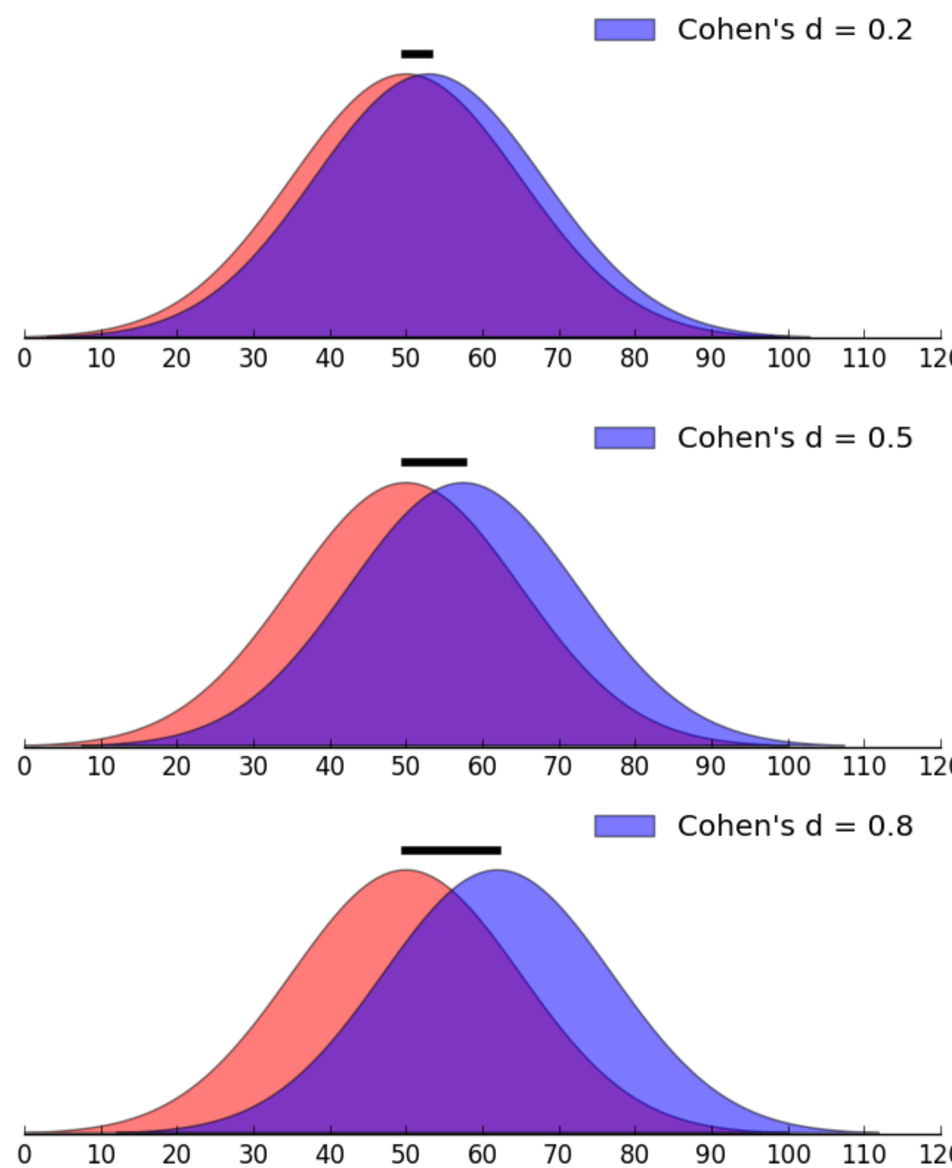
formula

```
library("lsr")
cohensD(x = balance ~ student,
        data = df.credit)
```

d = 0.89



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



Thank you!