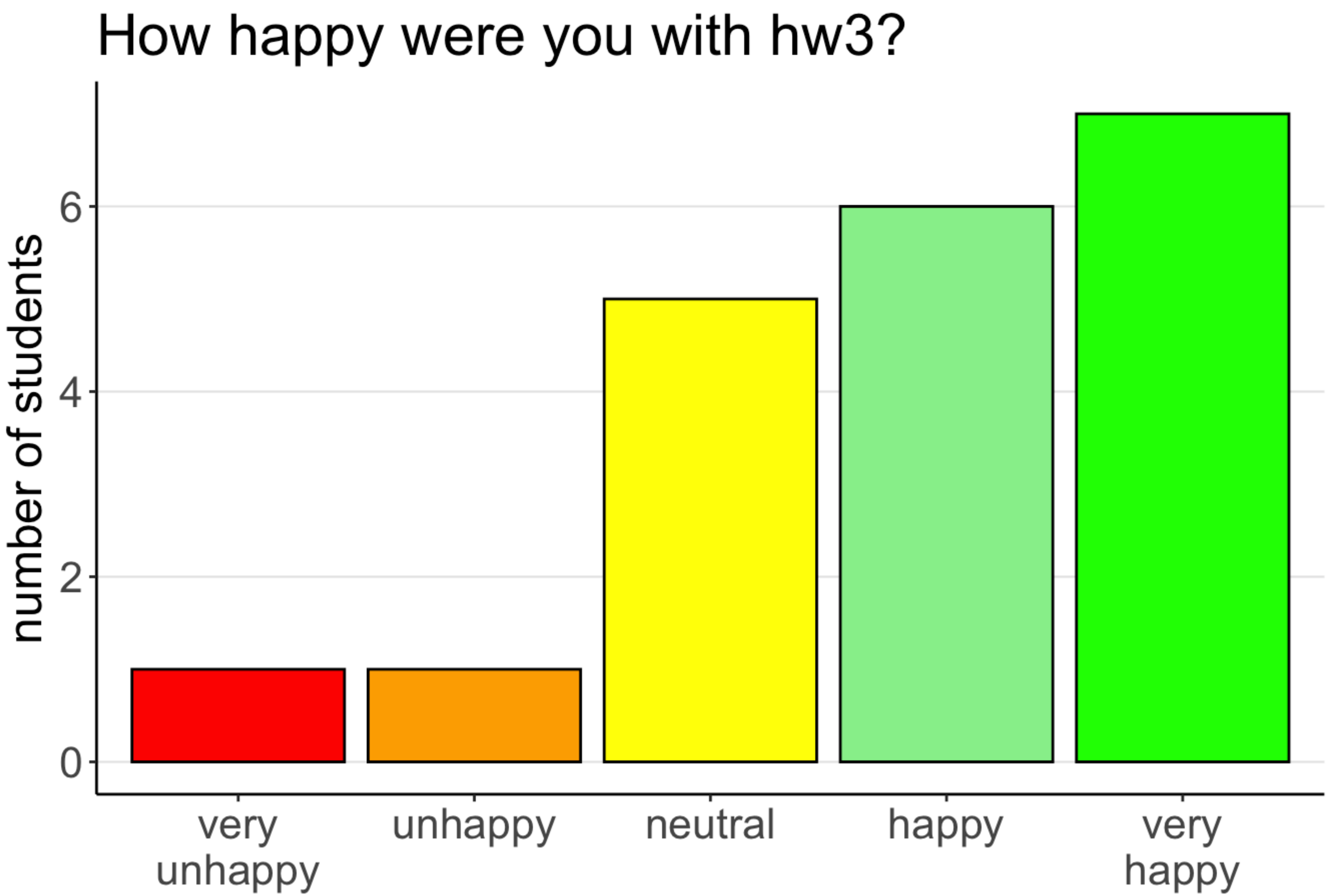
Linear model 3

If she loves you more each and every day,
by linear regression she hated you before you met.

02/03/2020

Logistics

Datacamp homework



I owe you answers to some questions ...

- degrees of freedom
- permutation test vs. t-test
- n-1 in standard deviation
- Likert scale predictor (ordered factor)
- when to standardize predictors and when not
- mean best approximator only if error distribution is normally distributed? or also for other error distributions?

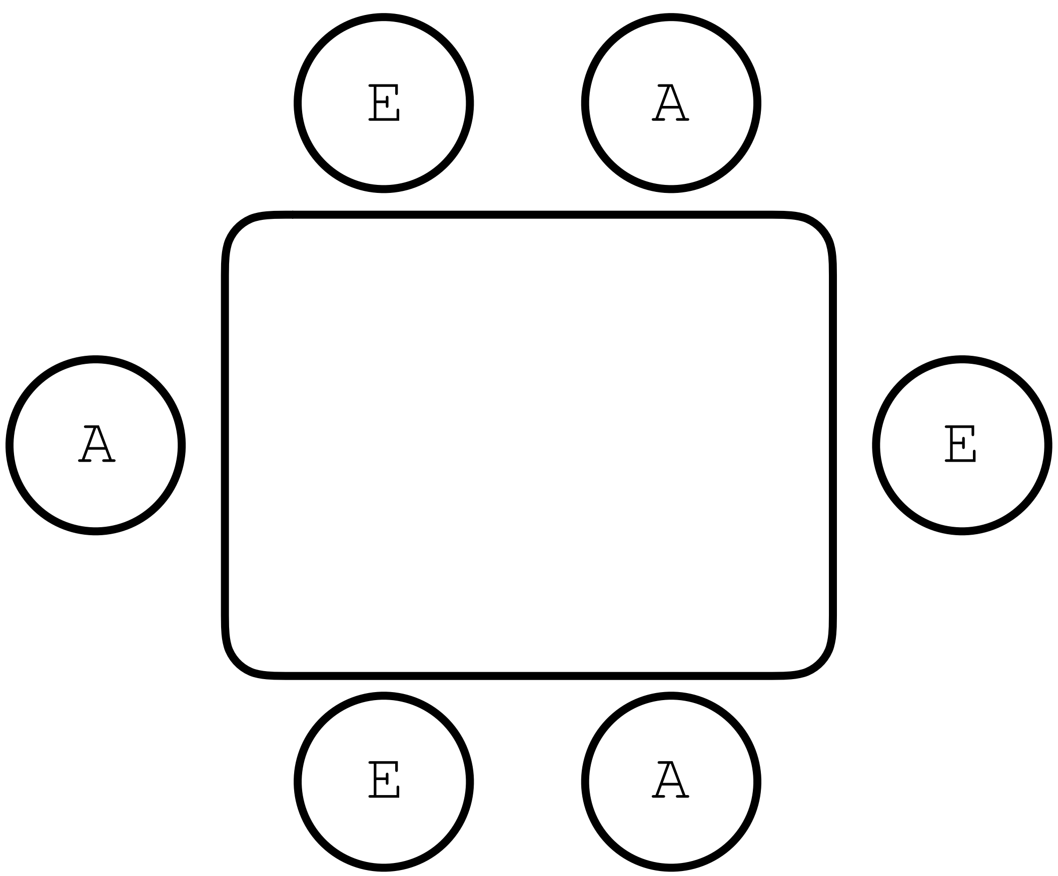
Plan for today

- Linear model with ...
  - categorical predictor that has more than two levels (One-way ANOVA)
  - multiple categorical predictors (N-way ANOVA)
  - linear contrasts

What's the role of skill vs. chance in poker?

Abstract

Adopting a quasi-experimental approach, the present study examined the extent to which the influence of poker playing skill was more important than card distribution. Three average players and three experts sat down at a six-player table and played **60 computer-based** hands of the poker variant ''Texas Hold'em'' for money. In each hand, one of the average players and one expert received (a) better-than-average cards (winner's box), (b) average cards (neutral box) and (c) worse-than-average cards (loser's box). The standardized manipulation of the card distribution controlled the factor of chance to determine differences in performance between the average and expert groups. Overall, 150 individuals participated in a ''fixed- limit'' game variant, and 150 individuals participated in a ''no-limit'' game variant.



During the game, one expert player and one average player received

(a) the winning hand 15 times and the losing hand 5 times (winner's box condition)

(b) the winning hand 10 times and the losing hand 10 times (neutral box condition)

(c) the winning hand 5 times and the losing hand 15 times (loser's box condition)

Data set for today

participant	skill	hand	limit	balance
1	expert	bad	fixed	4.00
2	expert	bad	fixed	5.55
26	expert	bad	none	5.52
27	expert	bad	none	8.28
51	expert	neutral	fixed	11.74
52	expert	neutral	fixed	10.04
76	expert	neutral	none	21.55
77	expert	neutral	none	3.12
101	expert	good	fixed	10.86
102	expert	good	fixed	8.68

skill = expert/average

hand = bad/neutral/good

limit = fixed/none

balance = final balance in Euros

2 (skill) x 3 (hand) x 2 (limit) design

25 participants per condition

n = 300

Meyer, G., von Meduna, M., Brosowski, T., & Hayer, T. (2012). Is poker a game of skill or chance? A quasi-experimental study. *Journal of Gambling Studies*

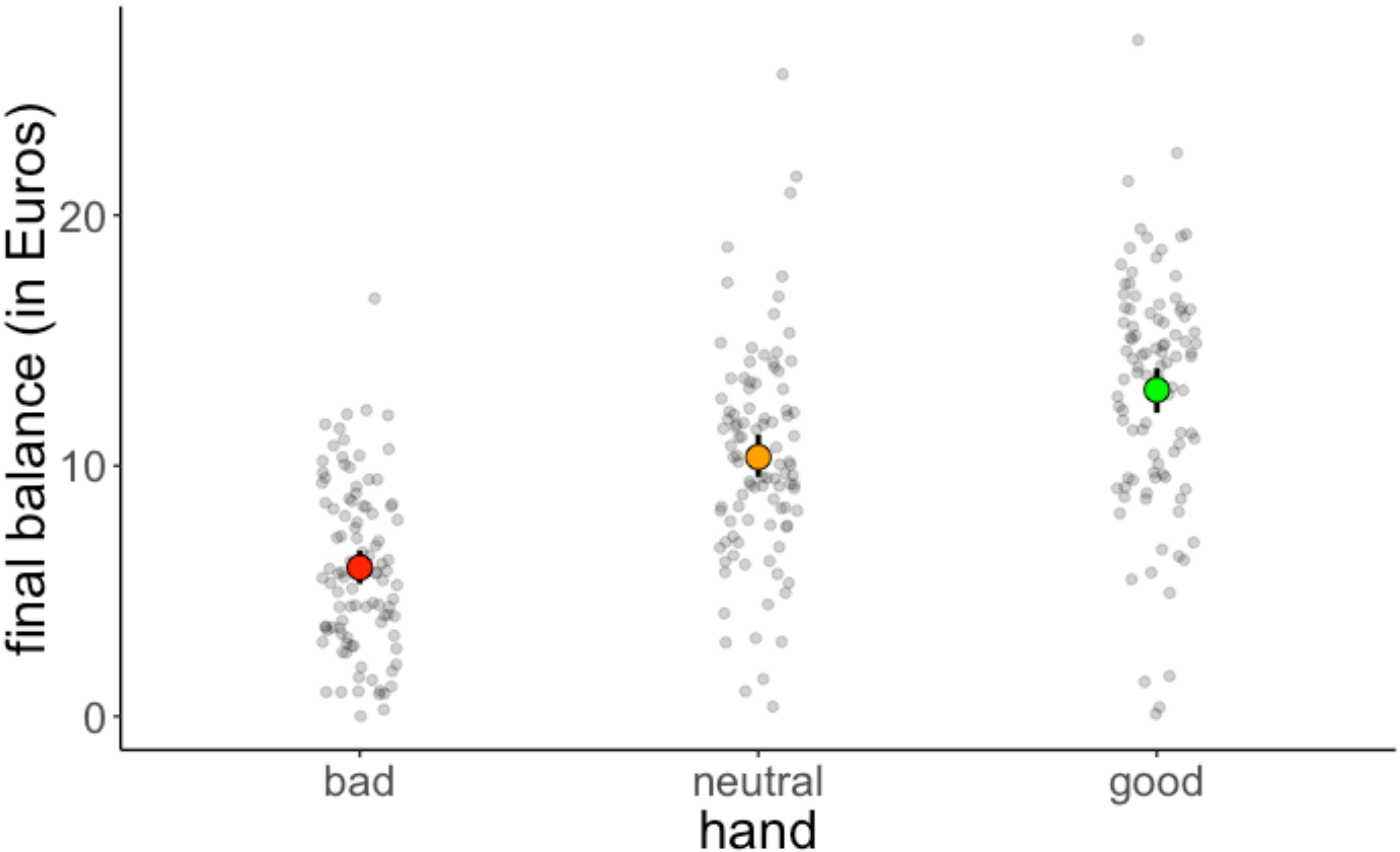
Categorical predictor with more than two levels

Do better hands win more money?

participant	skill	hand	limit	balance
1	expert	bad	fixed	4.00
2	expert	bad	fixed	5.55
26	expert	bad	none	5.52
27	expert	bad	none	8.28
51	expert	neutral	fixed	11.74
52	expert	neutral	fixed	10.04
76	expert	neutral	none	21.55
77	expert	neutral	none	3.12
101	expert	good	fixed	10.86
102	expert	good	fixed	8.68

hand = {bad, neutral, good}

Visualize the data first



H$_0$: Card quality does not affect the final balance.

H$_1$: Card quality affects the final balance.

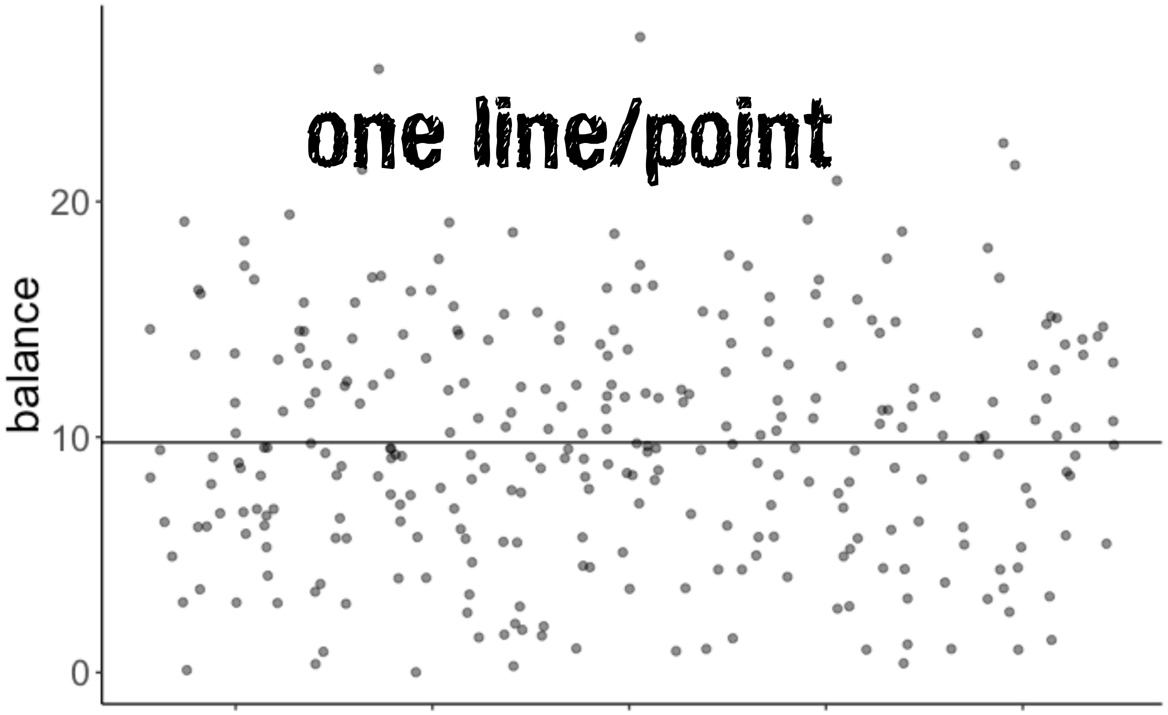
Model C

$$\text{balance}_i = \beta_0 + \epsilon_i$$

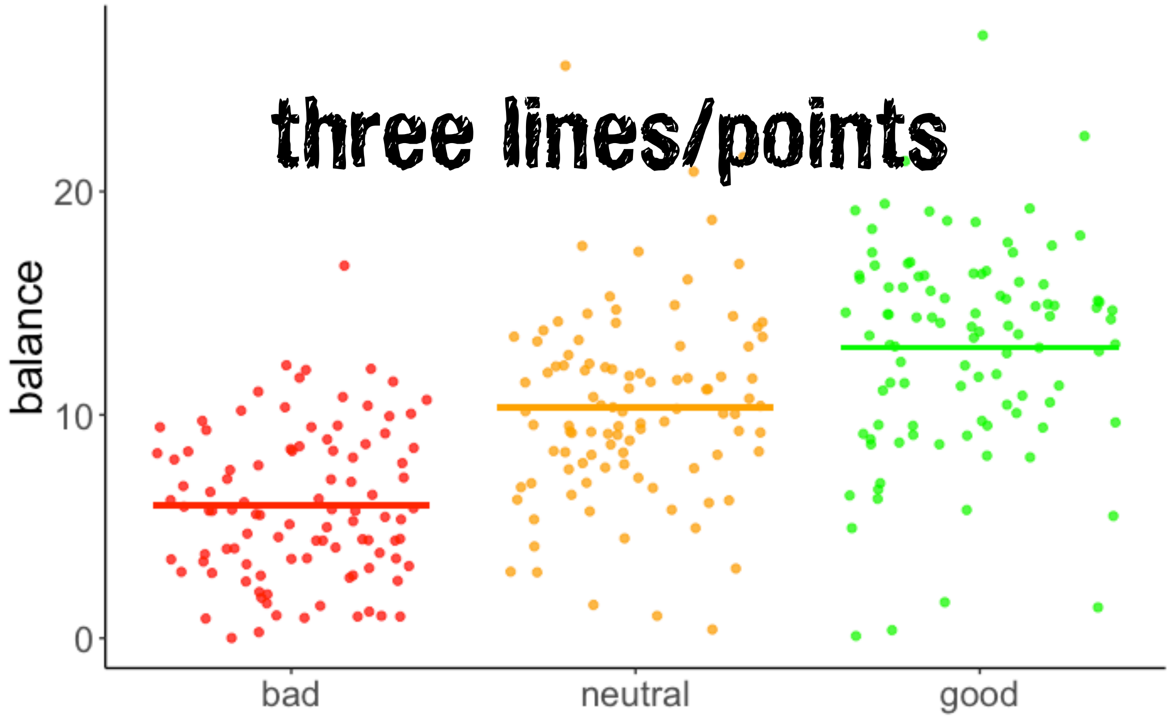
Model A

$$\text{balance}_i = \beta_0 + \beta_1 \text{hand\_neutral}_i + \beta_2 \text{hand\_good}_i + \epsilon$$

Model prediction



Model prediction



Fitted model

$$\widehat{\text{balance}}_i = 9.77$$

Fitted model

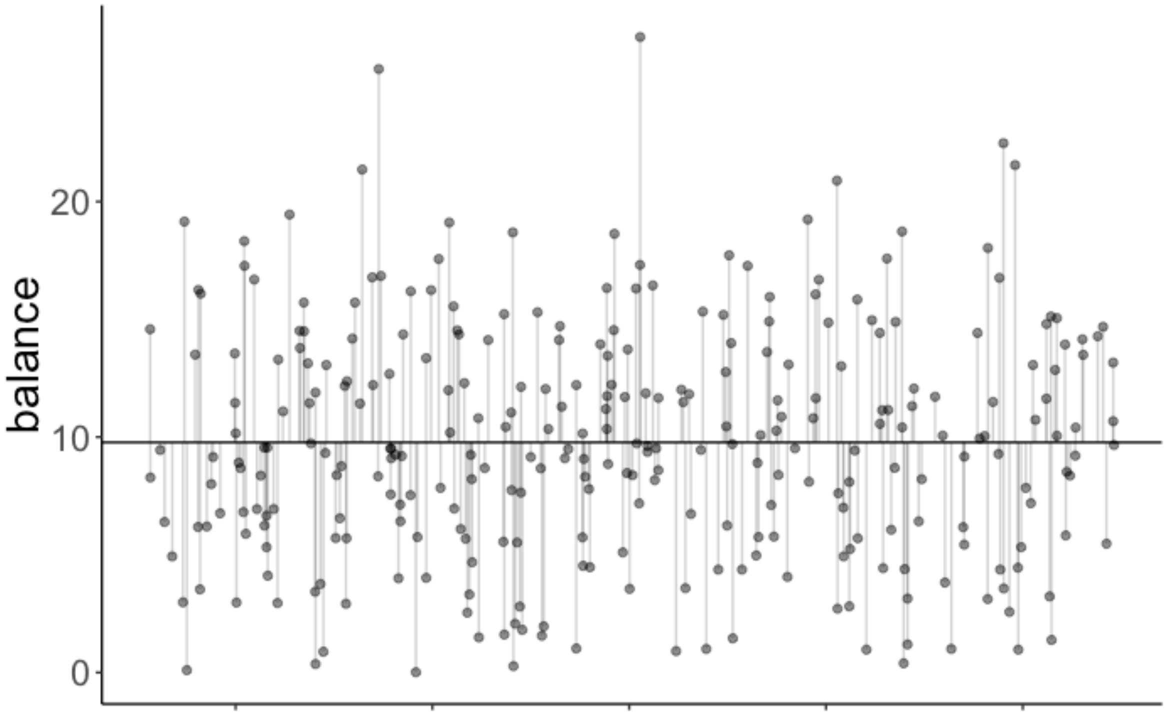
$$\widehat{\text{balance}}_i = 5.94 + 4.41 \cdot \text{hand\_neutral}_i + 7.08 \cdot \text{hand\_good}_i$$

H_0: Card quality does not affect the final balance.

Model C

$$\text{balance}_i = \beta_0 + \epsilon_i$$

Model prediction



SSE(C) = 7580

Fitted model

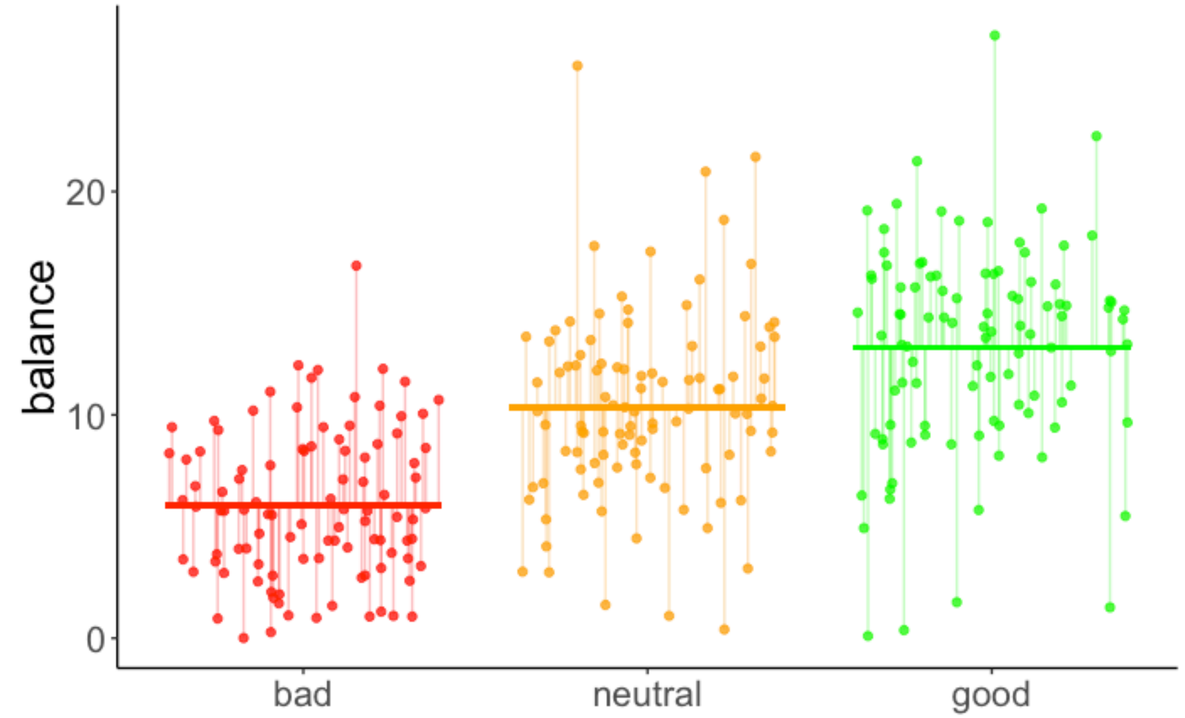
$$\widehat{\text{balance}}_i = 9.77$$

H_1: Card quality affects the final balance.

Model A

$$\text{balance}_i = \beta_0 + \beta_1 \text{hand\_neutral}_i + \beta_2 \text{hand\_good}_i + \epsilon$$

Model prediction



SSE(A) = 5021

Fitted model

$$\widehat{\text{balance}}_i = 5.94 + 4.41 \cdot \text{hand\_neutral}_i + 7.08 \cdot \text{hand\_good}_i$$

Does card quality affect the final balance?

SSE(C) = 7580

SSE(A) = 5021

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

worth it?

$$= 1 - \frac{5021}{7580} \approx 0.34$$

```r
# fit the models
fit_c = lm(formula = balance ~ 1, data = df.poker)
fit_a = lm(formula = balance ~ hand, data = df.poker)

# compare via F-test
anova(fit_c, fit_a)
```

```
Analysis of Variance Table

Model 1: balance ~ 1
Model 2: balance ~ hand
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
1    299 7580.0
2    297 5020.6  2    2559.4 75.703 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Interpreting the results

```
lm(formula = balance ~ 1 + hand, data = df.poker)
```

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

Residuals:
     Min       1Q   Median       3Q      Max 
-12.9264  -2.5902  -0.0115   2.6573  15.2834 

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

Residual standard error: 4.111 on 297 degrees of freedom
Multiple R-squared:  0.3377,	Adjusted R-squared:  0.3332 
F-statistic:  75.7 on 2 and 297 DF,  p-value: < 2.2e-16
```

Dummy coding

```r
df.poker %>%
  mutate(hand_neutral = ifelse(hand == "neutral", 1, 0),
         hand_good = ifelse(hand == "good", 1, 0))
```

participant	hand	hand_neutral	hand_good	balance
31	bad	0	0	12.22
46	bad	0	0	12.06
50	bad	0	0	16.68
76	neutral	1	0	21.55
87	neutral	1	0	20.89
89	neutral	1	0	25.63
127	good	0	1	26.99
129	good	0	1	21.36
283	good	0	1	22.48

for a variable with k levels, we need k-1 dummy variables for encoding

same same, but different

```r
lm(formula = balance ~ 1 + hand_neutral + hand_good + data = df.poker)

lm(formula = balance ~ 1 + hand, data = df.poker)
```

Interpreting the results

regression coefficients encode differences between group means

term	estimate	std.error	statistic	p.value
(Intercept)	5.941	0.411	14.451	0
handneutral	4.405	0.581	7.576	0
handgood	7.085	0.581	12.185	0

$$\widehat{\text{balance}}_i = 5.94 + 4.41 \cdot \text{hand\_neutral}_i + 7.08 \cdot \text{hand\_good}_i$$

participant	hand	hand_neutral	hand_good	balance
31	bad	0	0	12.22
46	bad	0	0	12.06
50	bad	0	0	16.68
76	neutral	1	0	21.55
87	neutral	1	0	20.89
89	neutral	1	0	25.63
127	good	0	1	26.99
129	good	0	1	21.36
283	good	0	1	22.48

if hand == "bad":

$$\widehat{\text{balance}}_i = 5.94$$

if hand == "neutral":

$$\widehat{\text{balance}}_i = 5.94 + 4.41 = 10.35$$

if hand == "good":

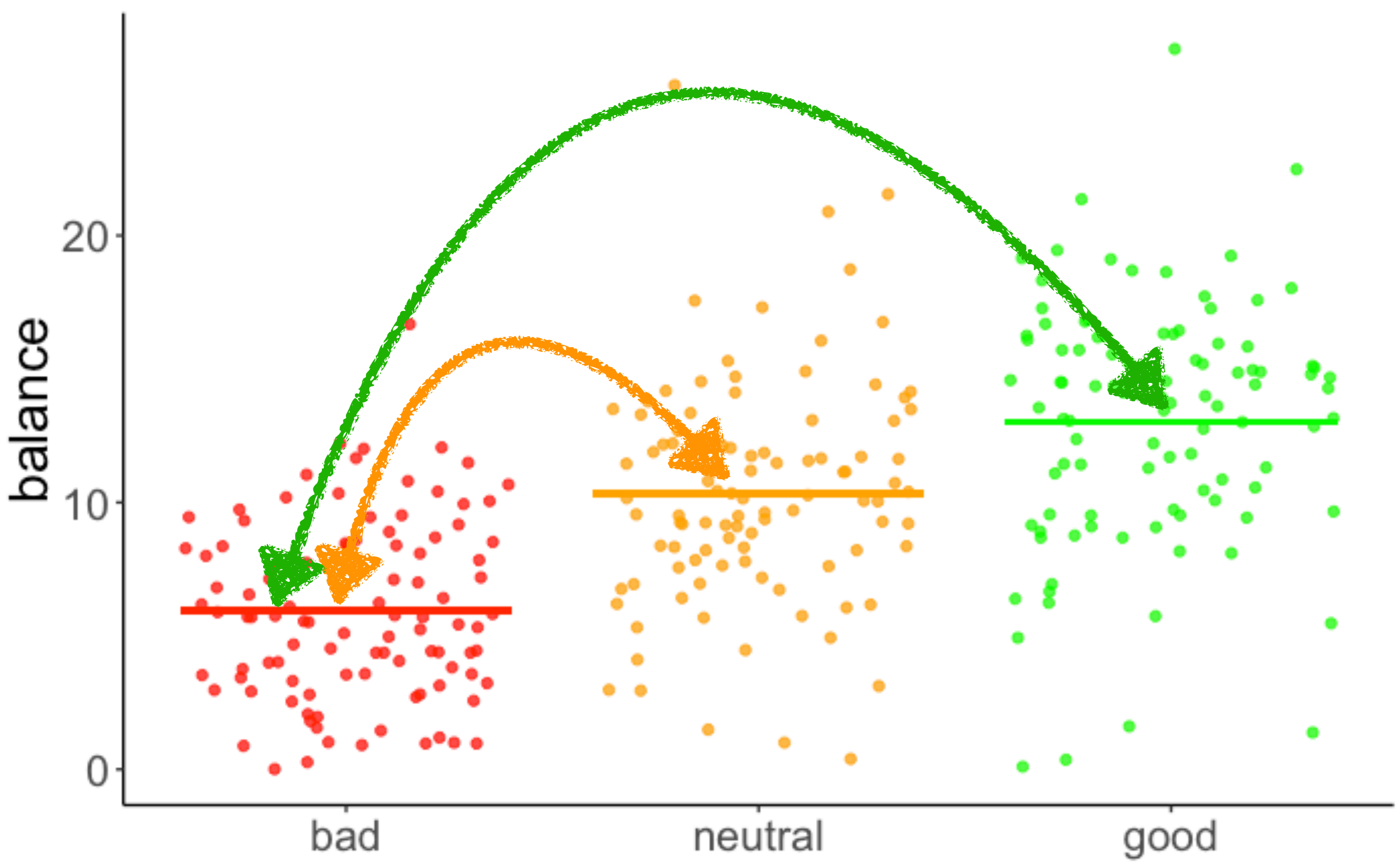
$$\widehat{\text{balance}}_i = 5.94 + 7.08 = 13.02$$

Interpreting the results

term	estimate	std.error	statistic	p.value
(Intercept)	5.941	0.411	14.451	0
handneutral	4.405	0.581	7.576	0
handgood	7.085	0.581	12.185	0

regression coefficients encode differences between group means

$$\widehat{\text{balance}}_i = 5.94 + 4.41 \cdot \text{hand\_neutral}_i + 7.08 \cdot \text{hand\_good}_i$$



if hand == "bad":

$$\widehat{\text{balance}}_i = 5.94$$

if hand == "neutral":

$$\widehat{\text{balance}}_i = 5.94 + 4.41 = 10.35$$

if hand == "good":

$$\widehat{\text{balance}}_i = 5.94 + 7.08 = 13.02$$

One-way ANOVA

```r
lm(formula = balance ~ hand, data = df.poker) %>%
  anova()
```

```
Analysis of Variance Table

Response: balance
           Df Sum Sq Mean Sq F value    Pr(>F)    
hand        2 2559.4  1279.7  75.703 < 2.2e-16 ***
Residuals 297 5020.6    16.9                      
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

What do these mean?

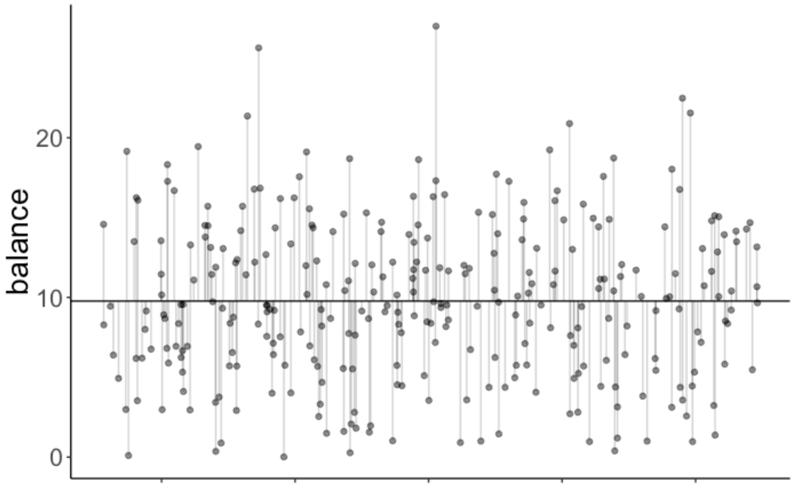
```r
# fit the models
fit_c = lm(formula = balance ~ 1, data = df.poker)
fit_a = lm(formula = balance ~ hand, data = df.poker)

# compare via F-test
anova(fit_c, fit_a)
```

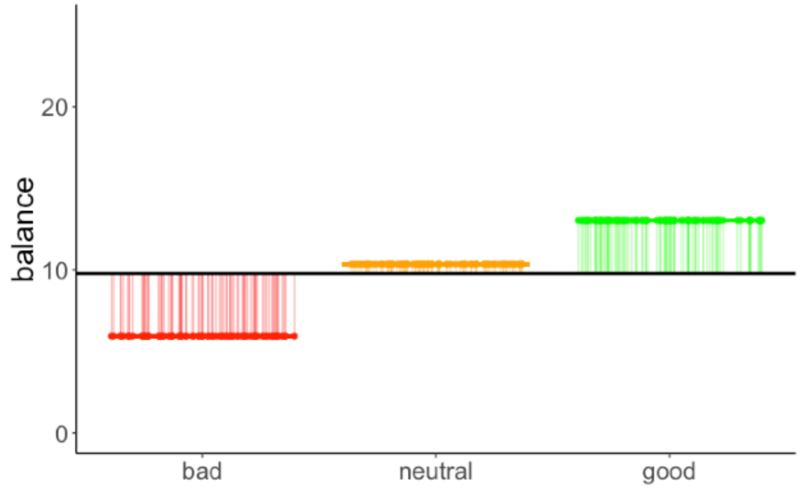
```
Analysis of Variance Table

Model 1: balance ~ 1
Model 2: balance ~ hand
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)    
1    299 7580.0                                  
2    297 5020.6  2    2559.4 75.703 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

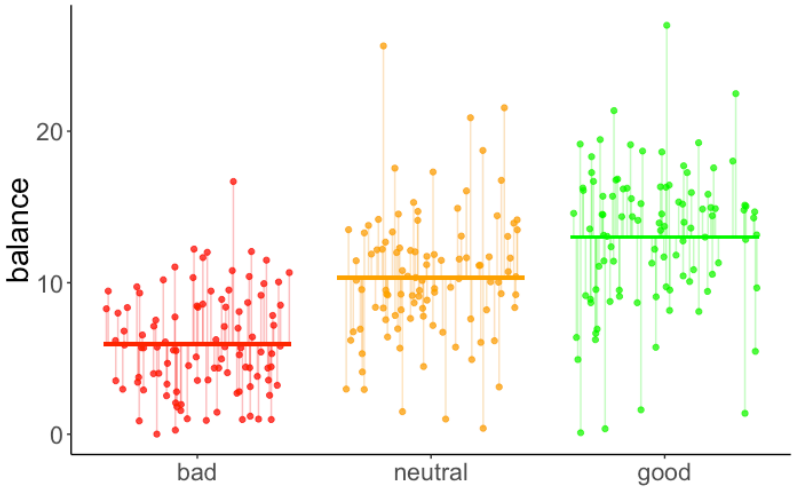
One-way ANOVA

Variance decomposition

Total variance	Model variance	Residual variance







SS_{total} SS_{model} $SS_{residual}$

variance_total	variance_model	variance_residual
7580	2559	5021

One-way ANOVA

```
df.poker %>%
  mutate(mean_grand = mean(balance)) %>%
  group_by(hand) %>%
  mutate(mean_group = mean(balance)) %>%
```

participant	hand	balance	mean_grand	mean_group
1	bad	4.00	9.771	5.941
2	bad	5.55	9.771	5.941
3	bad	9.45	9.771	5.941
51	neutral	11.74	9.771	10.347
52	neutral	10.04	9.771	10.347
53	neutral	9.49	9.771	10.347
101	good	10.86	9.771	13.026
102	good	8.68	9.771	13.026
103	good	14.36	9.771	13.026

variance_total	variance_model	variance_residual
7580	2559	5021

```
Analysis of Variance Table

Response: balance
           Df Sum Sq Mean Sq F value    Pr(>F)
hand        2 2559.4  1279.7  75.703 < 2.2e-16 ***
Residuals 297 5020.6    16.9
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Two-way ANOVA

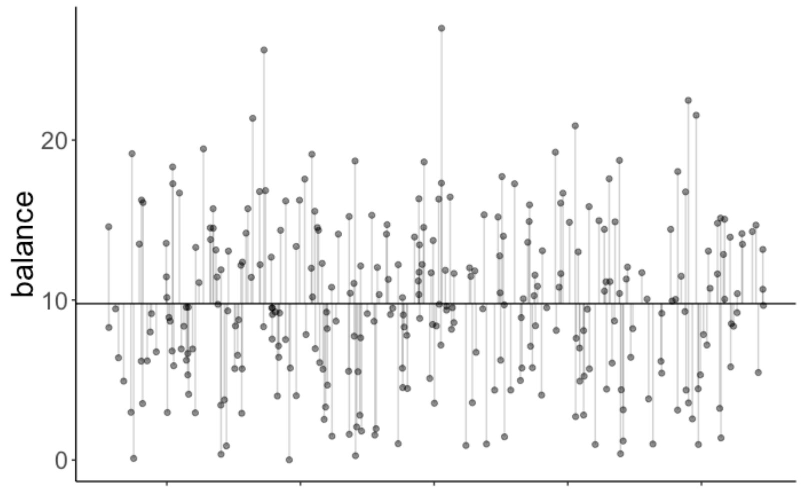
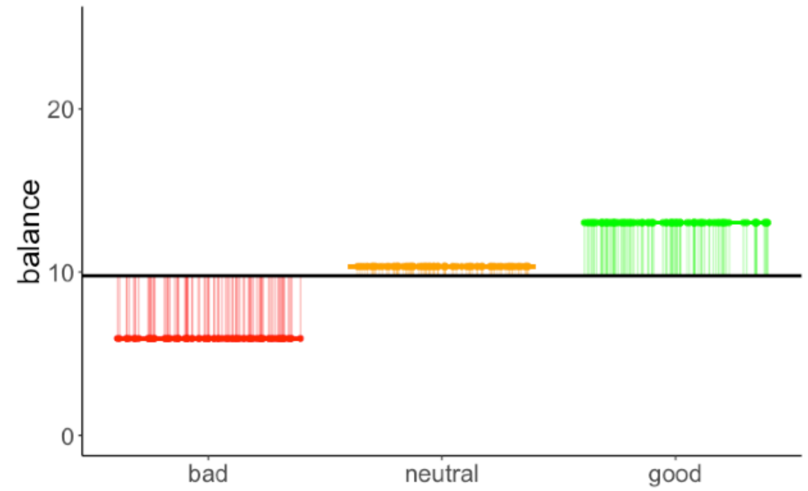
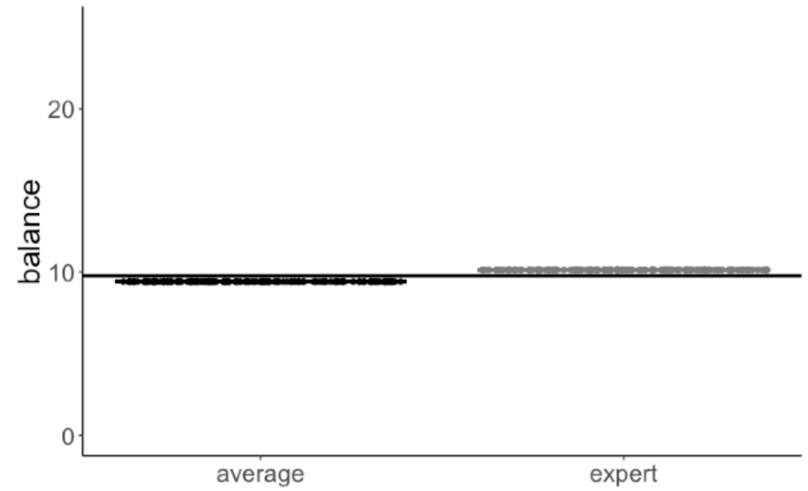
```
lm(formula = balance ~ hand + skill, data = df.poker) %>%
  anova()
```

```
Analysis of Variance Table

Response: balance
           Df Sum Sq Mean Sq F value Pr(>F)
hand        2 2559.4 1279.70 76.0437 <2e-16 ***
skill       1   39.3   39.35  2.3383 0.1273
Residuals 296 4981.2   16.83
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

What do these mean?

Two-way ANOVA

Variance decomposition

Total variance	Hand variance	Skill variance
		
SS_{total}	SS_{hand}	SS_{skill}

$$SS_{residual} = SS_{total} - SS_{hand} - SS_{skill}$$

Two-way ANOVA

```r
df.poker %>%
  mutate(mean_grand = mean(balance)) %>%
  group_by(skill) %>%
  mutate(mean_skill = mean(balance)) %>%
  group_by(hand) %>%
  mutate(mean hand = mean(balance)) %>%
```

participant	skill	hand	balance	mean_grand	mean_skill	mean_hand
1	expert	bad	4.00	9.77	10.13	5.94
2	expert	bad	5.55	9.77	10.13	5.94
51	expert	neutral	11.74	9.77	10.13	10.35
52	expert	neutral	10.04	9.77	10.13	10.35
101	expert	good	10.86	9.77	10.13	13.03
102	expert	good	8.68	9.77	10.13	13.03
151	average	bad	4.37	9.77	9.41	5.94
152	average	bad	3.58	9.77	9.41	5.94
201	average	neutral	6.42	9.77	9.41	10.35
202	average	neutral	14.18	9.77	9.41	10.35

variance_total	variance_skill	variance_hand	variance_residual
7580	39	2559	4981

```
Analysis of Variance Table

Response: balance
           Df Sum Sq Mean Sq F value Pr(>F)
hand        2 2559.4 1279.70 76.0437 <2e-16 ***
skill       1   39.3   39.35  2.3383 0.1273
Residuals 296 4981.2   16.83
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

One-way ANOVA

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

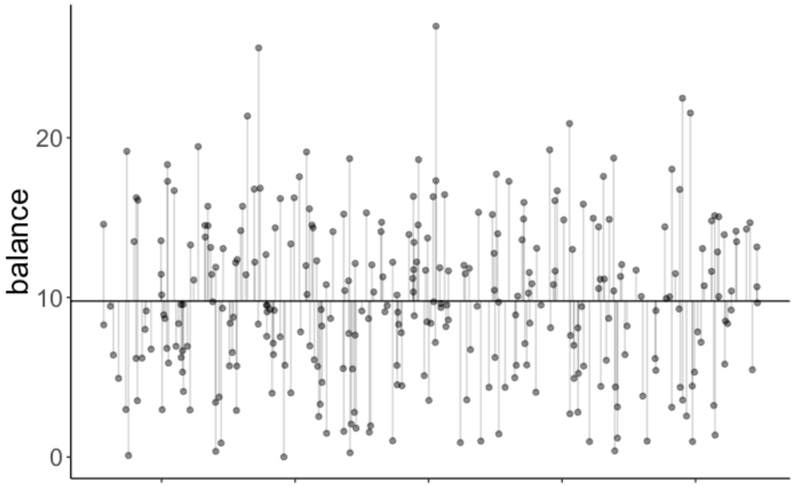
```
Analysis of Variance Table

Response: balance
           Df Sum Sq Mean Sq F value    Pr(>F)
hand        2 2559.4  1279.7  75.703 < 2.2e-16 ***
Residuals 297 5020.6    16.9
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

What do these mean?

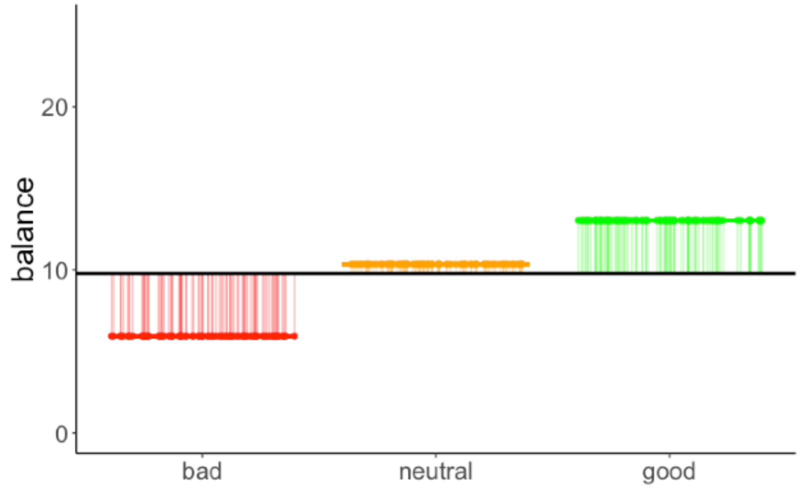
One-way ANOVA

Variance decomposition

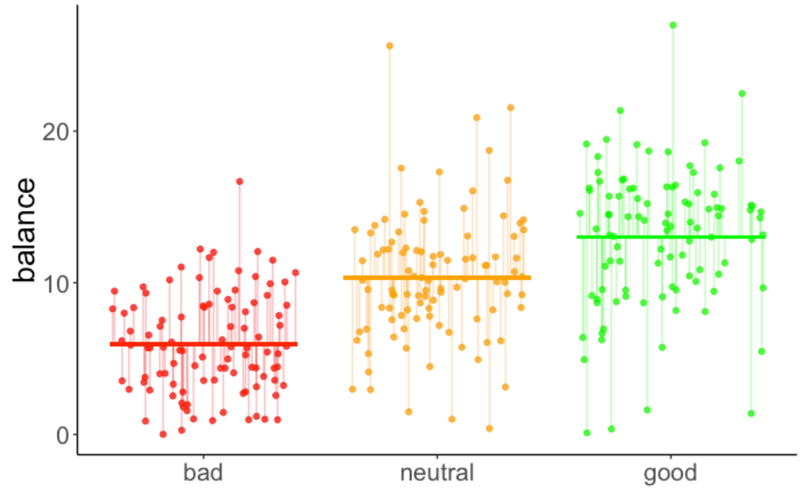
Total variance	Model variance	Residual variance



SS_{total}



SS_{model}



$SS_{residual}$

variance_total	variance_model	variance_residual
7580	2559	5021

One-way ANOVA

```
df.poker %>%
  mutate(mean_grand = mean(balance)) %>%
  group_by(hand) %>%
  mutate(mean_group = mean(balance)) %>%
```

participant	hand	balance	mean_grand	mean_group
1	bad	4.00	9.771	5.941
2	bad	5.55	9.771	5.941
3	bad	9.45	9.771	5.941
51	neutral	11.74	9.771	10.347
52	neutral	10.04	9.771	10.347
53	neutral	9.49	9.771	10.347
101	good	10.86	9.771	13.026
102	good	8.68	9.771	13.026
103	good	14.36	9.771	13.026

variance_total	variance_model	variance_residual
7580	2559	5021

```
Analysis of Variance Table

Response: balance
           Df Sum Sq Mean Sq F value    Pr(>F)
hand        2 2559.4  1279.7  75.703 < 2.2e-16 ***
Residuals 297 5020.6    16.9
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Follow-up tests

Asking more specific questions

Is there a difference in the final balance between bad hands and neutral hands?

```r
df.poker %>%
  filter(hand %in% c("bad", "neutral")) %>%
  lm(formula = balance ~ hand,
     data = .) %>%
  summary()
```

```
Call:
lm(formula = balance ~ hand, data = .)

Residuals:
    Min      1Q  Median      3Q     Max 
-9.9566 -2.5078 -0.2365  2.4410 15.2834 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   5.9415     0.3816  15.570  < 2e-16 ***
handneutral   4.4051     0.5397   8.163 3.76e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.816 on 198 degrees of freedom
Multiple R-squared:  0.2518,	Adjusted R-squared:  0.248 
F-statistic: 66.63 on 1 and 198 DF,  p-value: 3.758e-14
```

Interpreting the results

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

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

Residuals:
     Min       1Q   Median       3Q      Max 
-12.9264  -2.5902  -0.0115   2.6573  15.2834 

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

Residual standard error: 4.111 on 297 degrees of freedom
Multiple R-squared:  0.3377, Adjusted R-squared:  0.3332 
F-statistic:  75.7 on 2 and 297 DF,  p-value: < 2.2e-16
```

What does this summary not tell us?

Model comparison

Is there a difference in the final balance between neutral hands and good hands?

Model C

$$\text{balance}_i = \beta_0 + \epsilon_i$$

Model A

$$\text{balance}_i = \beta_0 + \beta_1 \cdot \text{good\_dummy}_i + \epsilon_i$$

(after having removed bad hands from the data set)

Asking more specific questions

Is there a difference in the final balance between neutral hands and good hands?

```
df.poker %>%
  filter(hand %in% c("neutral", "good")) %>%
  lm(formula = balance ~ hand,
     data = .) %>%
  summary()
```

```
Call:
lm(formula = balance ~ hand, data = .)

Residuals:
     Min       1Q   Median       3Q      Max 
-12.9264  -2.7141   0.2585   2.7184  15.2834 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  10.3466     0.4448   23.26  < 2e-16 ***
handgood      2.6798     0.6291    4.26 3.16e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.448 on 198 degrees of freedom
Multiple R-squared:  0.08396,	Adjusted R-squared:  0.07933 
F-statistic: 18.15 on 1 and 198 DF,  p-value: 3.158e-05
```

Asking more specific questions

Is there a difference in the final balance between neutral hands and good hands?

```
df.poker %>%
  mutate(hand = fct_relevel(hand, "neutral")) %>%
  lm(formula = balance ~ hand,
     data = .) %>%
  summary()
```

same same, but different

```
Call:
lm(formula = balance ~ hand, data = .)

Residuals:
     Min       1Q   Median       3Q      Max 
-12.9264  -2.5902  -0.0115   2.6573  15.2834 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  10.3466     0.4111  25.165  < 2e-16 ***
handbad      -4.4051     0.5815  -7.576 4.55e-13 ***
handgood      2.6798     0.5815   4.609 6.02e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.111 on 297 degrees of freedom
Multiple R-squared:  0.3377,	Adjusted R-squared:  0.3332 
F-statistic:  75.7 on 2 and 297 DF,  p-value: < 2.2e-16
```

Is there a difference between bad hands vs. other hands?

```
df.poker %>%
  mutate(hand_other = ifelse(hand %in% c("neutral", "good"), 1, 0)) %>%
  lm(balance ~ 1 + hand_other,
     data = .) %>%
  summary()
```

```
Call:
lm(formula = balance ~ 1 + hand_other, data = .)

Residuals:
     Min       1Q   Median       3Q      Max 
-11.5865  -2.6203  -0.1815   2.8285  15.3035 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   5.9415     0.4249   13.98   <2e-16 ***
hand_other    5.7450     0.5204   11.04   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.249 on 298 degrees of freedom
Multiple R-squared:  0.2903,	Adjusted R-squared:  0.2879 
F-statistic: 121.9 on 1 and 298 DF,  p-value: < 2.2e-16
```

df.poker

participant	hand	hand_other	balance
31	bad	0	12.22
46	bad	0	12.06
50	bad	0	16.68
76	neutral	1	21.55
87	neutral	1	20.89
89	neutral	1	25.63
127	good	1	26.99
129	good	1	21.36
283	good	1	22.48

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

if hand == bad: $\widehat{\text{balance}}_i = b_0 = 5.94$

if hand != bad: $\widehat{\text{balance}}_i = b_0 + b_1 = 5.94 + 5.75 = 11.69$

group means

bad	neutral	good
5.94	10.35	13.03

Multiple categorical predictors

Do skill level and quality of cards affect the final balance?

participant	skill	hand	limit	balance
1	expert	bad	fixed	4.00
2	expert	bad	fixed	5.55
26	expert	bad	none	5.52
27	expert	bad	none	8.28
51	expert	neutral	fixed	11.74
52	expert	neutral	fixed	10.04
76	expert	neutral	none	21.55
77	expert	neutral	none	3.12
101	expert	good	fixed	10.86
102	expert	good	fixed	8.68

Why not just fit separate models?

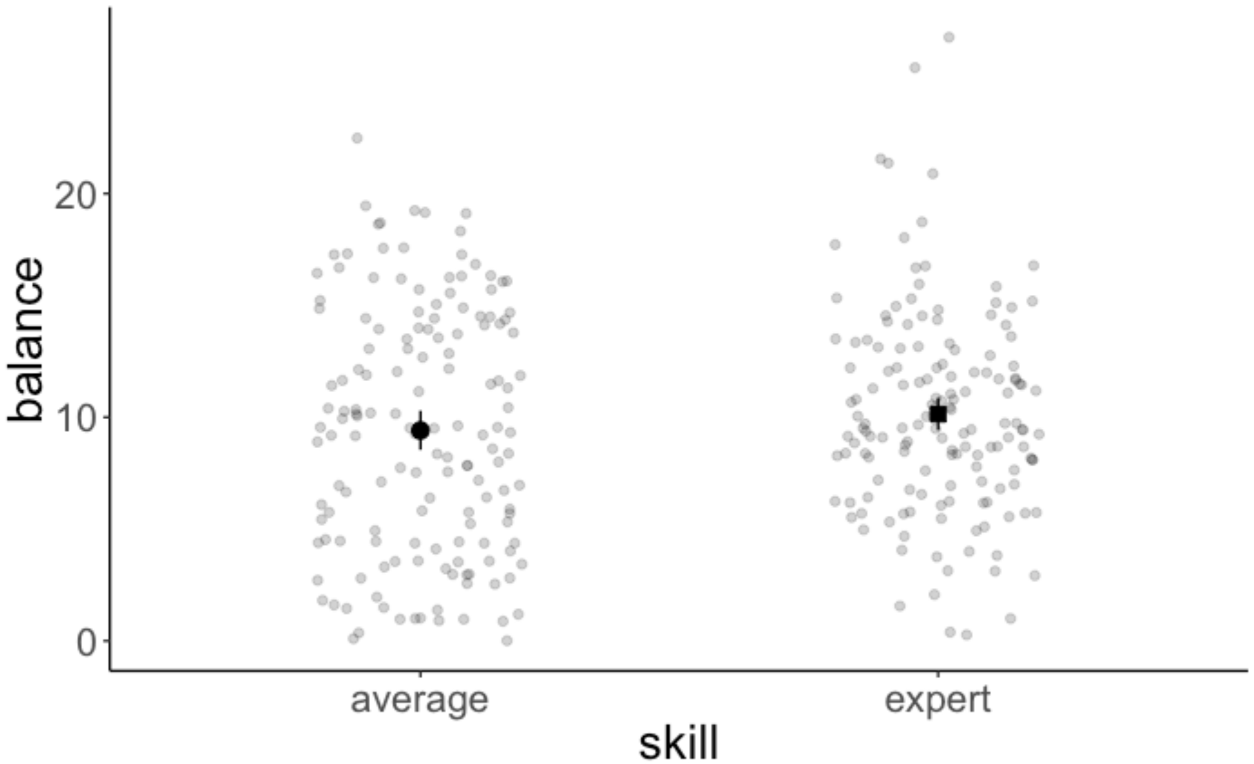
One testing whether skill level affects the final balance, and one testing whether quality of cards affects the final balance?

Interested in interactions!

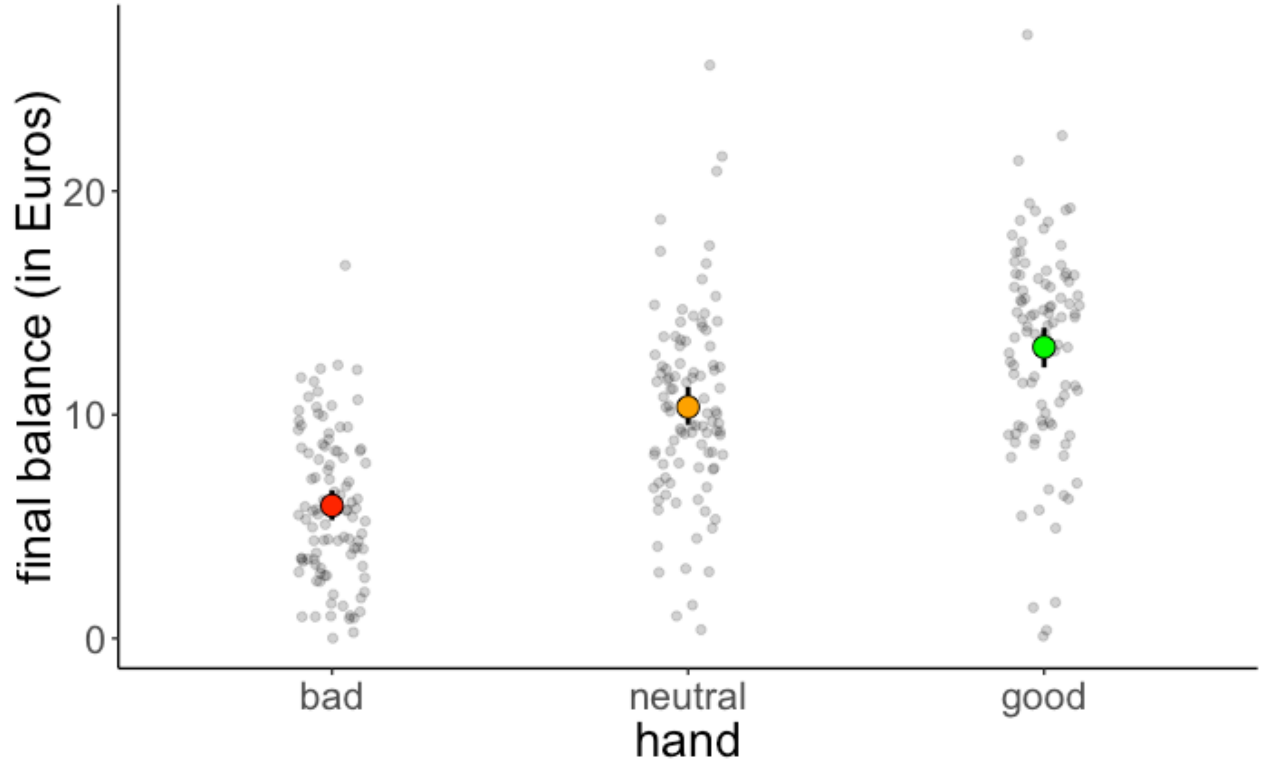
Does the effect of one variable depend on the other?

Visualize the data

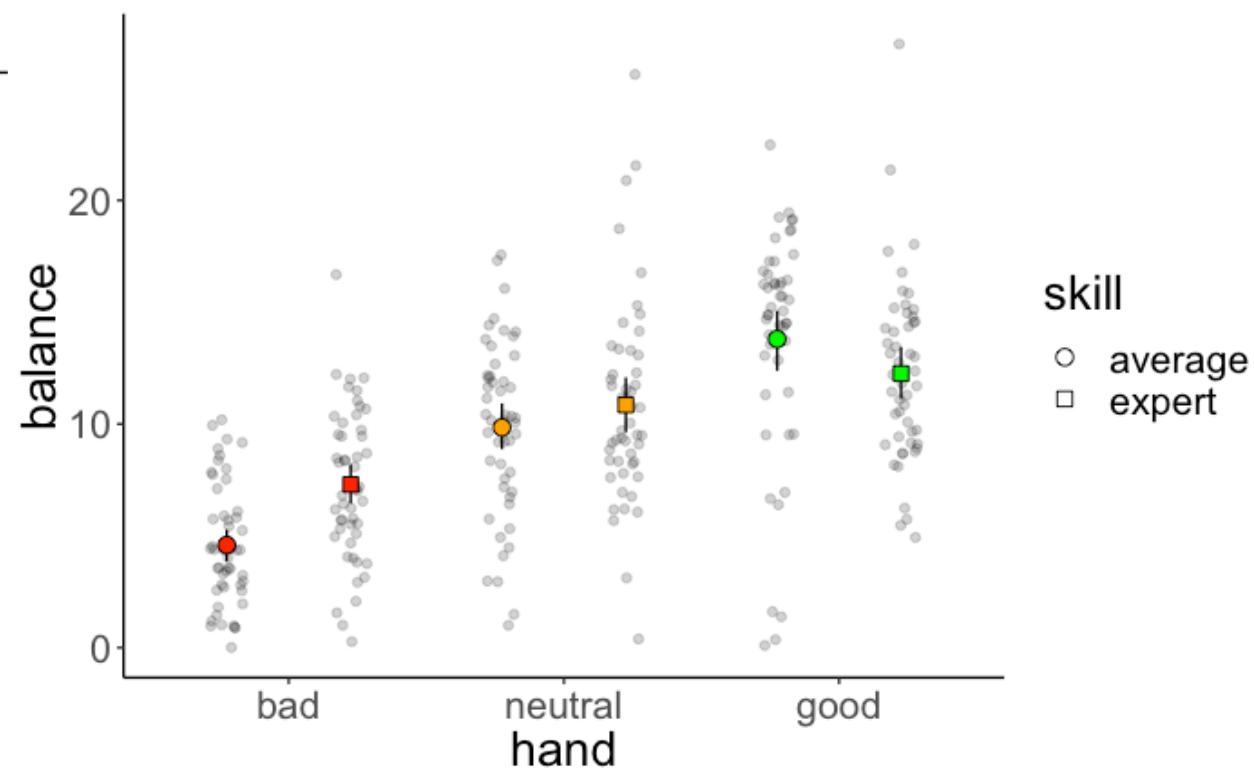
Skill



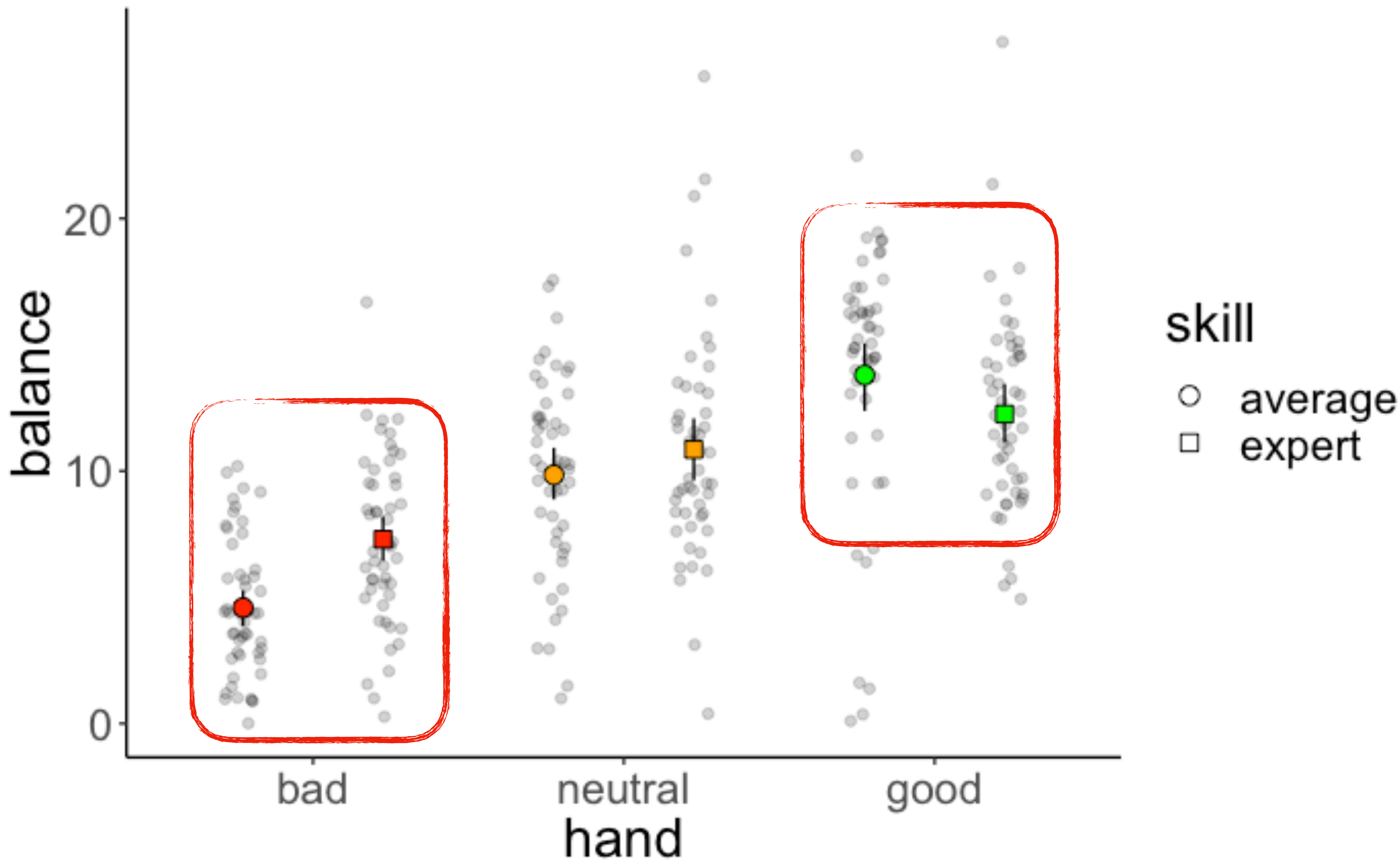
Quality of hand



Quality of hand & Skill



Visualize the data



Fit a model

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

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

Residuals:
     Min       1Q   Median       3Q      Max 
-13.6976  -2.4740   0.0348   2.4644  14.7806 

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

Residual standard error: 4.02 on 294 degrees of freedom
Multiple R-squared:  0.3731,Adjusted R-squared:  0.3624 
F-statistic: 34.99 on 5 and 294 DF,  p-value: < 2.2e-16
```

?

Interpretation

```
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 ***
```

group means

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

$$\widehat{\text{balance}}_i = b_0 + b_1 \cdot \text{hand\_neutral}_i + b_2 \cdot \text{hand\_good}_i + b_3 \cdot \text{skill\_expert}_i + b_4 \cdot \text{hand\_neutral:skill\_expert}_i + b_5 \cdot \text{hand\_good:skill\_expert}_i$$

hand = bad, skill = average

$$\widehat{\text{balance}}_i = b_0 = 4.59$$

hand = neutral, skill = average

$$\widehat{\text{balance}}_i = b_0 + b_1 \cdot \text{hand\_neutral}_i = 4.59 + 5.26 = 9.85$$

hand = good, skill = expert

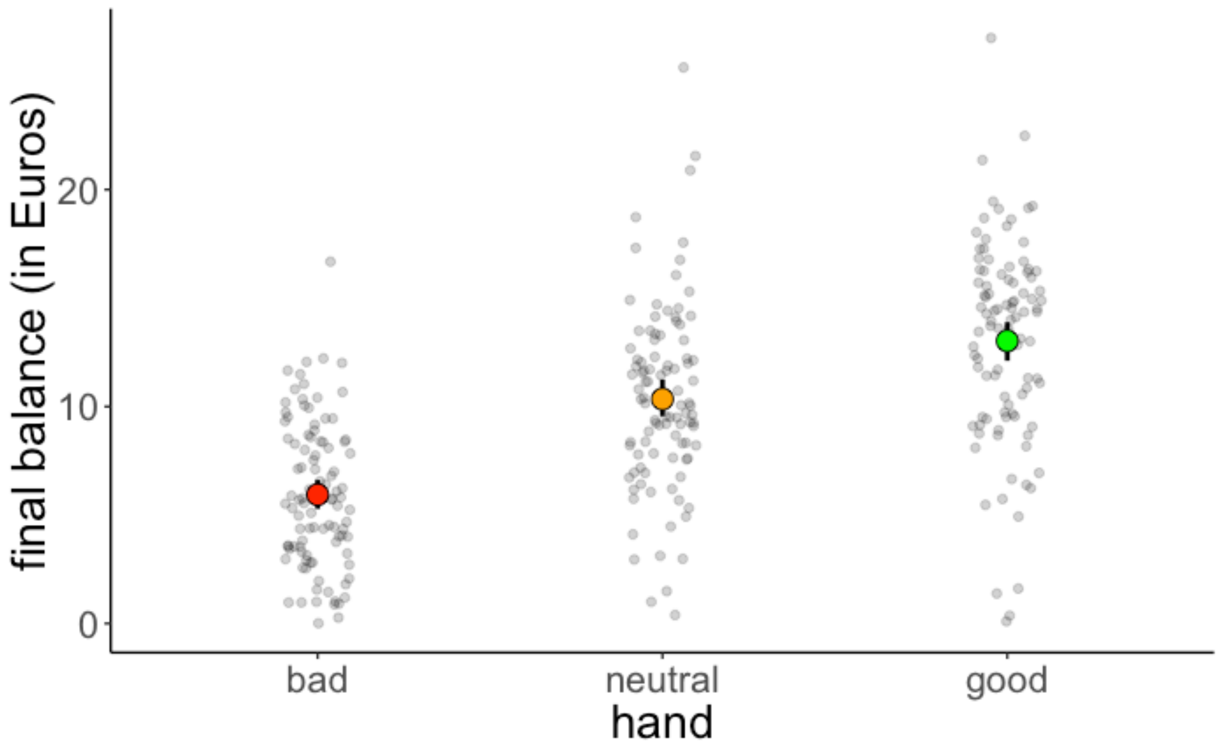
$$\widehat{\text{balance}}_i = b_0 + b_2 \cdot \text{hand\_good}_i + b_3 \cdot \text{skill\_expert}_i + b_5 \cdot \text{hand\_good:skill\_expert}_i = 12.26$$

Analysis of variance

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

```
Analysis of Variance Table

Response: balance
             Df Sum Sq Mean Sq F value    Pr(>F)    
hand          2 2559.4 1279.70 79.1692 < 2.2e-16 ***
skill         1   39.3   39.35  2.4344 0.1197776    
hand:skill    2  229.0  114.49  7.0830 0.0009901 ***
Residuals   294 4752.3   16.16                      
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

main effect of hand



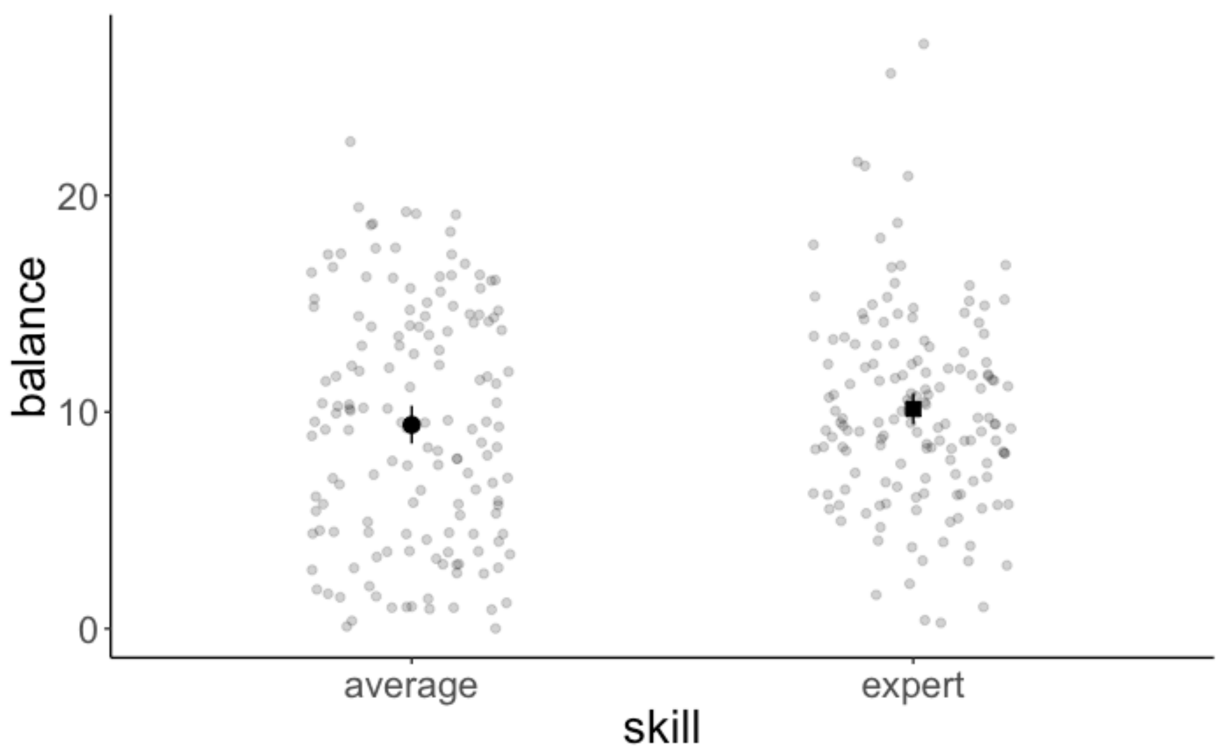
Analysis of variance

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

```
Analysis of Variance Table

Response: balance
             Df Sum Sq Mean Sq F value    Pr(>F)
hand          2 2559.4 1279.70 79.1692 < 2.2e-16 ***
skill         1   39.3   39.35  2.4344 0.1197776
hand:skill    2  229.0  114.49  7.0830 0.0009901 ***
Residuals   294 4752.3   16.16
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

main effect of `hand`

no main effect of `skill`



Analysis of variance

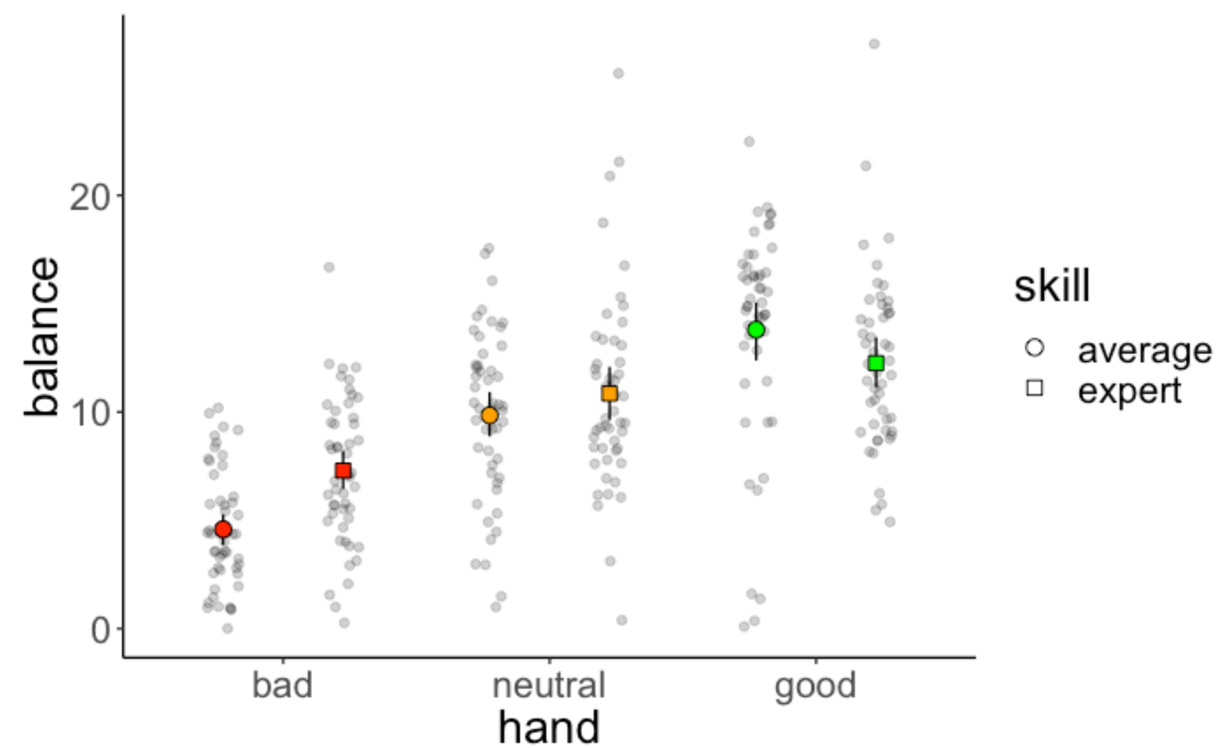
```
lm(formula = balance ~ hand * skill, data = df.poker) %>%
  anova()
```

```
Analysis of Variance Table

Response: balance
             Df Sum Sq Mean Sq F value    Pr(>F)    
hand          2 2559.4 1279.70 79.1692 < 2.2e-16 ***
skill         1   39.3   39.35  2.4344 0.1197776    
hand:skill    2  229.0  114.49  7.0830 0.0009901 ***
Residuals   294 4752.3   16.16                      
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

main effect of `hand`

no main effect of `skill`

interaction between `hand` and `skill`



Two-way ANOVA

```
lm(formula = balance ~ hand + skill, data = df.poker) %>%
  anova()
```

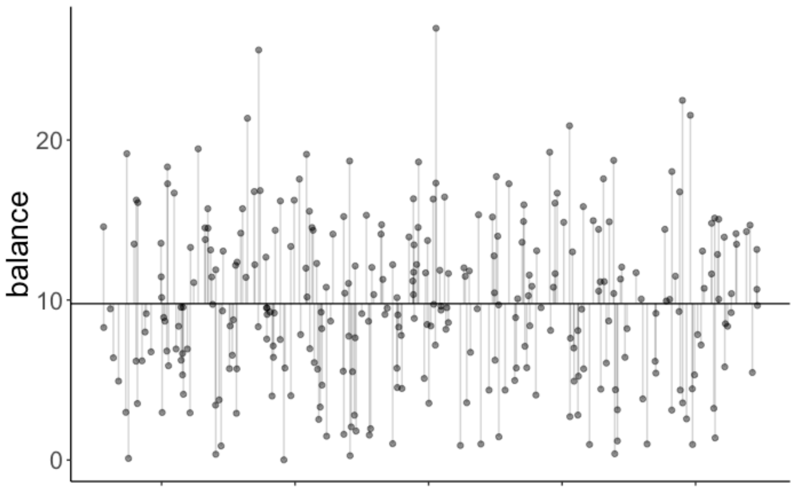
```
Analysis of Variance Table

Response: balance
           Df Sum Sq Mean Sq F value Pr(>F)
hand        2 2559.4 1279.70 76.0437 <2e-16 ***
skill       1   39.3   39.35  2.3383 0.1273
Residuals 296 4981.2   16.83
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

What do these mean?

Two-way ANOVA

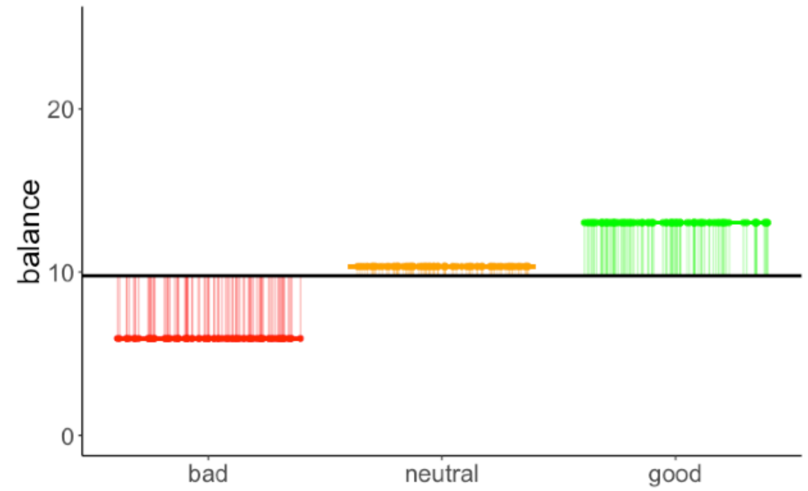
Variance decomposition

Total variance



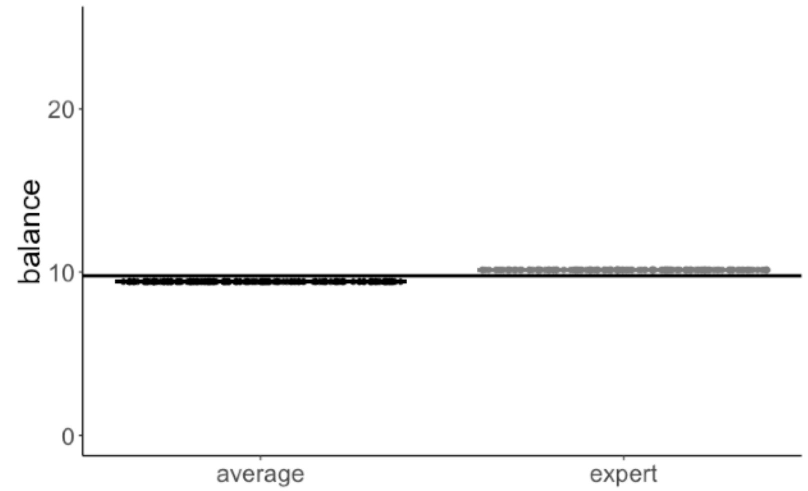
SS_{total}

Hand variance



SS_{hand}

Skill variance



SS_{skill}

$$SS_{residual} = SS_{total} - SS_{hand} - SS_{skill}$$

Two-way ANOVA

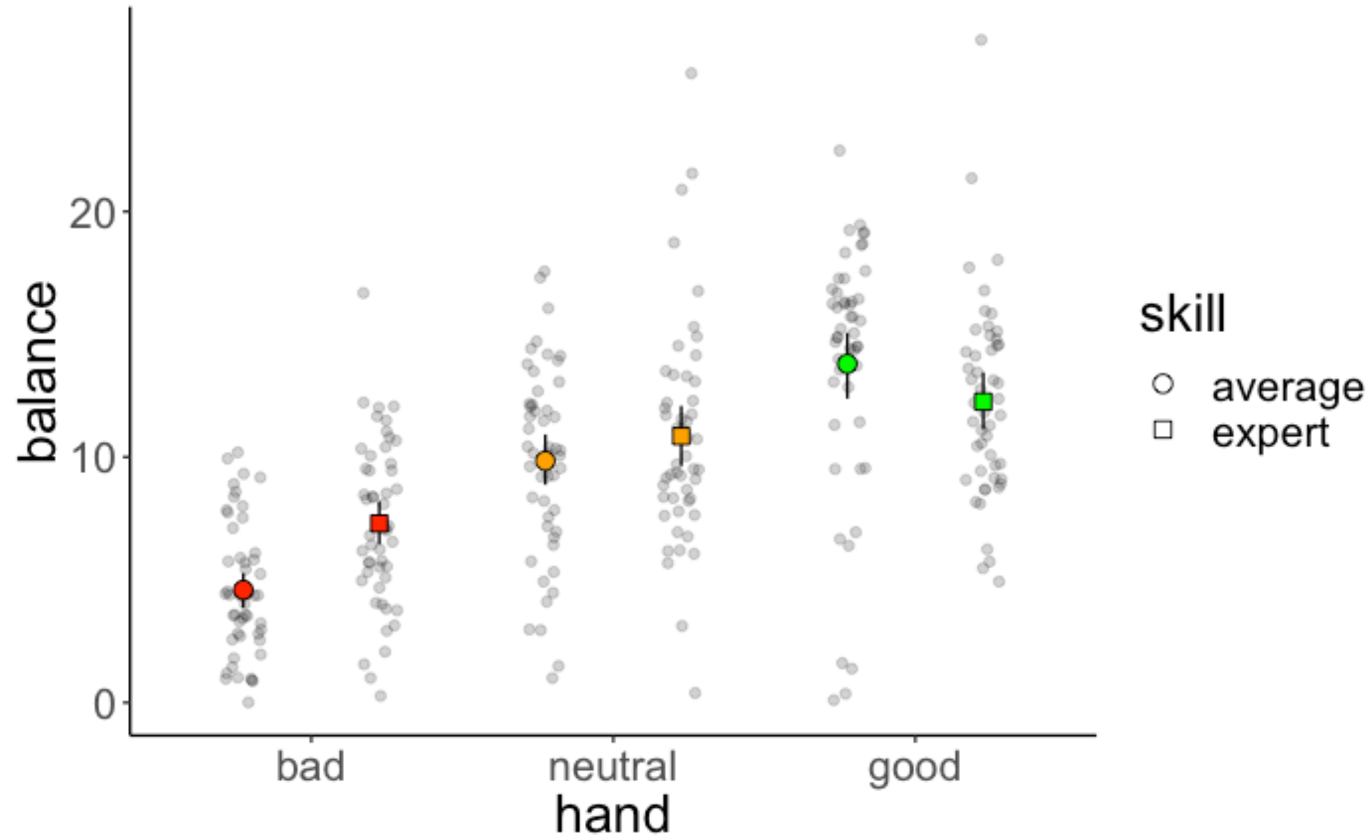
```
df.poker %>%
  mutate(mean_grand = mean(balance)) %>%
  group_by(skill) %>%
  mutate(mean_skill = mean(balance)) %>%
  group_by(hand) %>%
  mutate(mean hand = mean(balance)) %>%
```

participant	skill	hand	balance	mean_grand	mean_skill	mean_hand
1	expert	bad	4.00	9.77	10.13	5.94
2	expert	bad	5.55	9.77	10.13	5.94
51	expert	neutral	11.74	9.77	10.13	10.35
52	expert	neutral	10.04	9.77	10.13	10.35
101	expert	good	10.86	9.77	10.13	13.03
102	expert	good	8.68	9.77	10.13	13.03
151	average	bad	4.37	9.77	9.41	5.94
152	average	bad	3.58	9.77	9.41	5.94
201	average	neutral	6.42	9.77	9.41	10.35
202	average	neutral	14.18	9.77	9.41	10.35

variance_total	variance_skill	variance_hand	variance_residual
7580	39	2559	4981

```
Analysis of Variance Table

Response: balance
           Df Sum Sq Mean Sq F value Pr(>F)
hand        2 2559.4 1279.70 76.0437 <2e-16 ***
skill       1   39.3   39.35  2.3383 0.1273
Residuals 296 4981.2   16.83
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Reporting the results



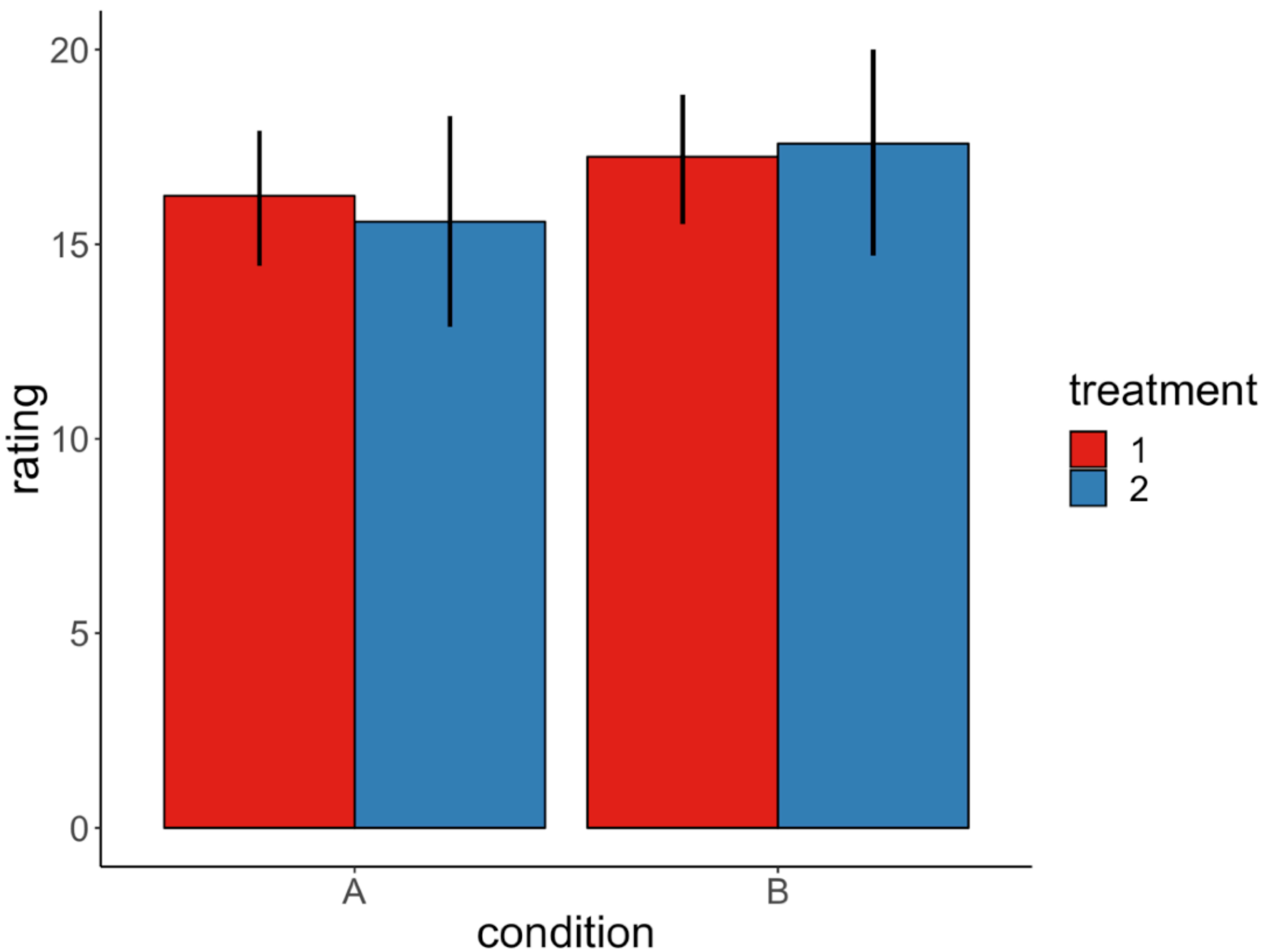
There was no main effect of skill *F*(1, 294) = 2.43, p = .12. The final balance of average (*M* = 9.41, *SD* = 5.51) and expert poker players (*M* = 10.13, *SD* = 4.50) did not differ significantly.

However, the quality of a player's hand significantly affected the final balance F(2, 294) = 79.17, p < .001. The final balance for good hands (M = 13.03, SD = 4.65) was significantly greater than for neutral hands (M = 10.35, SD = 4.24), and the balance for neutral hands was significantly higher than for bad hands (M = 5.94, SD = 3.34).

There was also a significant interaction between the quality of a player's hand and the player's skill level F(2, 294) = 7.08, p < .001. Whereas for bad hands, average players had a lower final balance than experts, for good hands, average players had a higher final balance than experts.

Who is the ANOVA champ?

Which effects are significant?

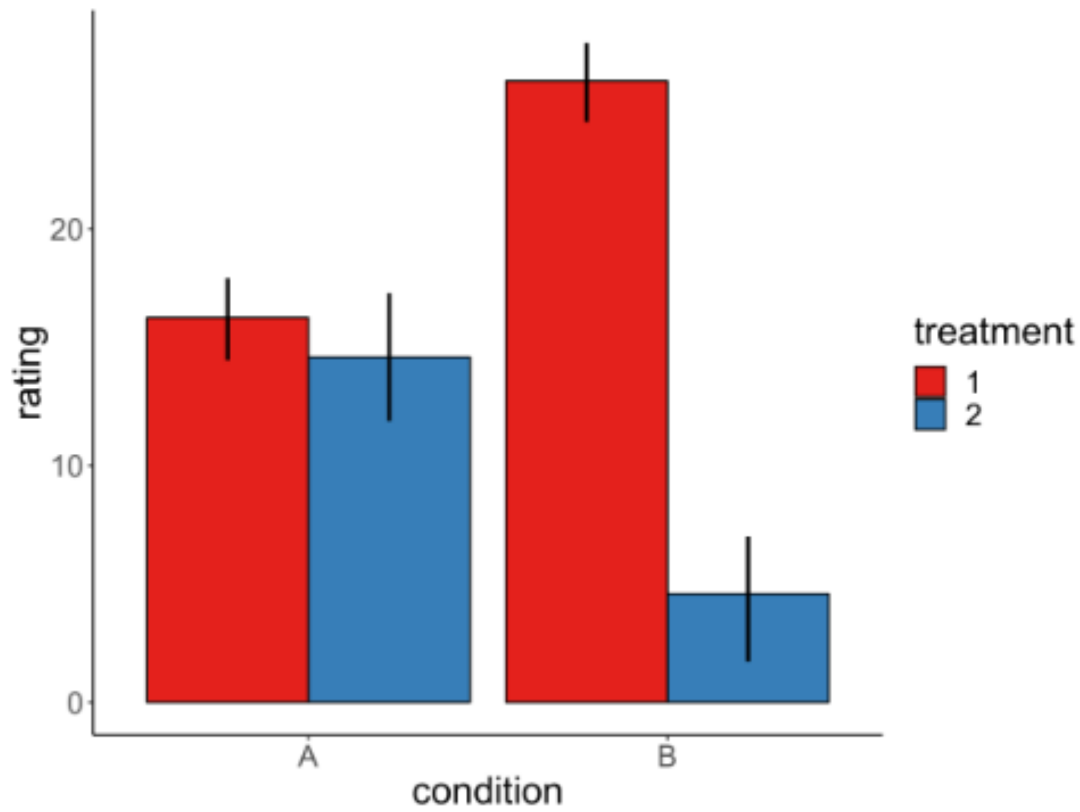


- Condition
- Treatment
- Condition x Treatment **interaction effect**
- Condition, Treatment **two main effects**
- Condition, Condition x Treatment
- Treatment, Condition x Treatment
- Condition, Treatment, Condition x Treatment

Who is the ANOVA champ?

Get ready to compete!

Which effects are significant?



Condition

Treatment

Condition x Treatment

Condition, Treatment

Condition, Condition x Treatment

Treatment, Condition x Treatment

Condition, Treatment, Condition x Treatment

Which effects are significant?

- Condition
- Treatment
- Condition x Treatment
- Condition, Treatment
- Condition, Condition x Treatment
- Treatment, Condition x Treatment
- Condition, Treatment, Condition x Treatment

Which effects are significant?

- Condition
- Treatment
- Condition x Treatment
- Condition, Treatment
- Condition, Condition x Treatment
- Treatment, Condition x Treatment
- Condition, Treatment, Condition x Treatment

Leaderboard

Which effects are significant?

- Condition
- Treatment
- Condition x Treatment
- Condition, Treatment
- Condition, Condition x Treatment
- Treatment, Condition x Treatment
- Condition, Treatment, Condition x Treatment

Which effects are significant?

- Condition
- Treatment
- Condition x Treatment
- Condition, Treatment
- Condition, Condition x Treatment
- Treatment, Condition x Treatment
- Condition, Treatment, Condition x Treatment

Leaderboard

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

```
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

```
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

ANOVA

- for these examples, I've assumed a balanced design (i.e. the same number of observations in each of the different factor levels)
- things get funky when we have an unbalanced design

Beware of unbalanced designs

```r
lm(formula = balance ~ skill + hand, data = df.poker) %>%
  anova()
```

```
Analysis of Variance Table

Response: balance
           Df Sum Sq Mean Sq F value  Pr(>F)
skill       1   74.3   74.28  4.2904 0.03922 *
hand        2 2385.1 1192.57 68.8827 < 2e-16 ***
Residuals 286 4951.5   17.31
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

flipped the order

```r
lm(formula = balance ~ hand + skill, data = df.poker) %>%
  anova()
```

```
Analysis of Variance Table

Response: balance
           Df Sum Sq Mean Sq F value Pr(>F)
hand        2 2419.8 1209.92 69.8845 <2e-16 ***
skill       1   39.6   39.59  2.2867 0.1316
Residuals 286 4951.5   17.31
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

If you want to reproduce SPSS

- There are different kinds of ANOVAs, for which the sums of squares are calculated differently.
- This makes a difference when we have an unbalanced design (i.e. some of the cell sizes are unequal).
- We won't discuss how the different sums of squares are computed.
- If you want to reproduce what SPSS spits out, then do the following ...

If you want to reproduce SPSS

```r
library("car")

lm(formula = balance ~ hand + skill,
   data = df.poker,
   contrasts = list(hand = "contr.sum",
                    skill = "contr.sum")) %>%
  Anova(type = "3")
```

load the "car" package

set the contrasts

run Anova (capital A) with type "3"
for the sum of squares

```
Anova Table (Type III tests)

Response: balance
             Sum Sq  Df   F value Pr(>F)
(Intercept) 27629.1   1 1595.8482 <2e-16 ***
hand         2385.1   2   68.8827 <2e-16 ***
skill          39.6   1    2.2867 0.1316
Residuals    4951.5 286
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

If you want to reproduce SPSS

```r
library("car")

lm(formula = balance ~ skill + hand,
   data = df.poker,
   contrasts = c("contr.sum", "contr.poly")) %>%
  Anova(type = "3")
```

now the order doesn't matter ...

```
Anova Table (Type III tests)

Response: balance
             Sum Sq  Df   F value Pr(>F)
(Intercept) 27629.1   1 1595.8482 <2e-16 ***
skill          39.6   1    2.2867 0.1316
hand         2385.1   2   68.8827 <2e-16 ***
Residuals    4951.5 286
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

If you want to reproduce SPSS

```r
library("afex")

fit = aov_ez(id = "participant",
             dv = "balance",
             data = df.poker,
             between = c("hand", "skill"))
fit$Anova
```

```
Contrasts set to contr.sum for the following variables: hand, skill
Anova Table (Type III tests)

Response: dv
             Sum Sq  Df   F value    Pr(>F)
(Intercept) 27781.3   1 1676.9096 < 2.2e-16 ***
hand         2285.3   2   68.9729 < 2.2e-16 ***
skill          48.9   1    2.9540 0.0867525 .
hand:skill    246.5   2    7.4401 0.0007089 ***
Residuals    4705.0 284
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

ANOVA

- We won't be discussing the different techniques to run ANOVAs in more detail.
- In practice, you'll almost never want to run an ANOVA. You'll want to ask more specific questions using planned contrasts.
- We will learn how to use linear mixed effects models (which are much more powerful and flexible than ANOVAs).
- Linear mixed effects models:
  - have no trouble with unbalanced designs
  - can capture dependencies in your data

Contrasts

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

```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())
```

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
```

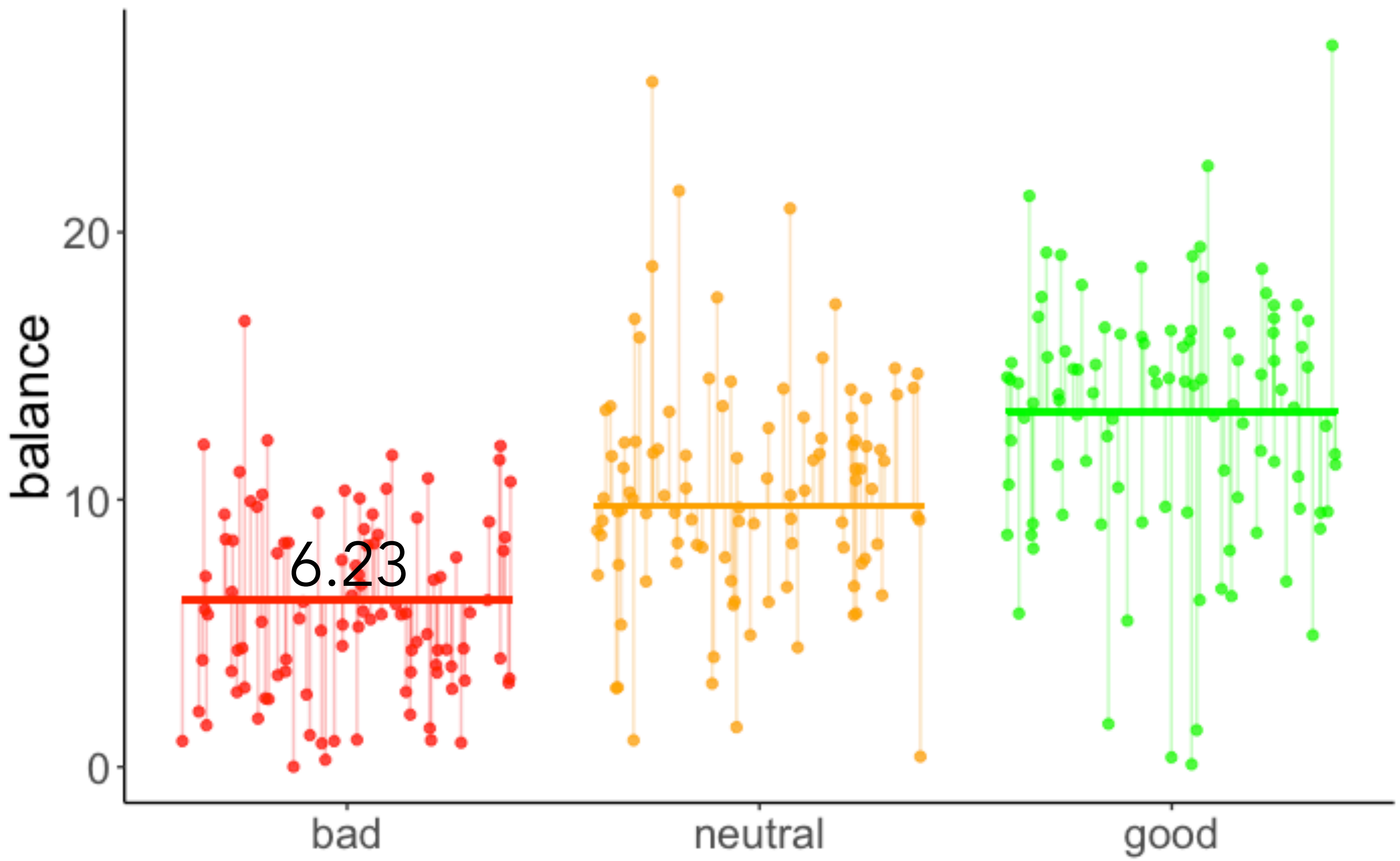
grand mean

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

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



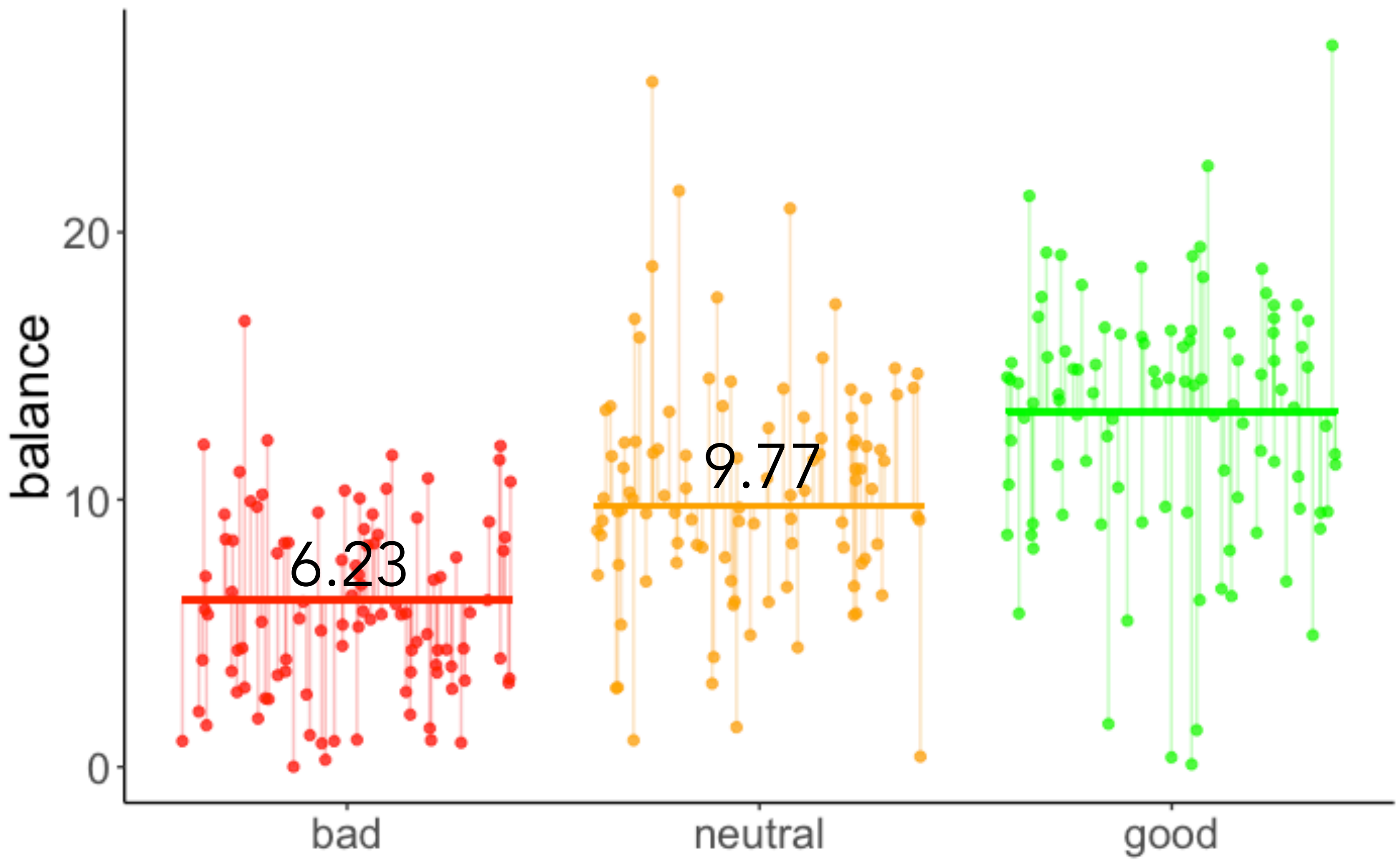
if contrast == -1

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

Contrasts

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

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



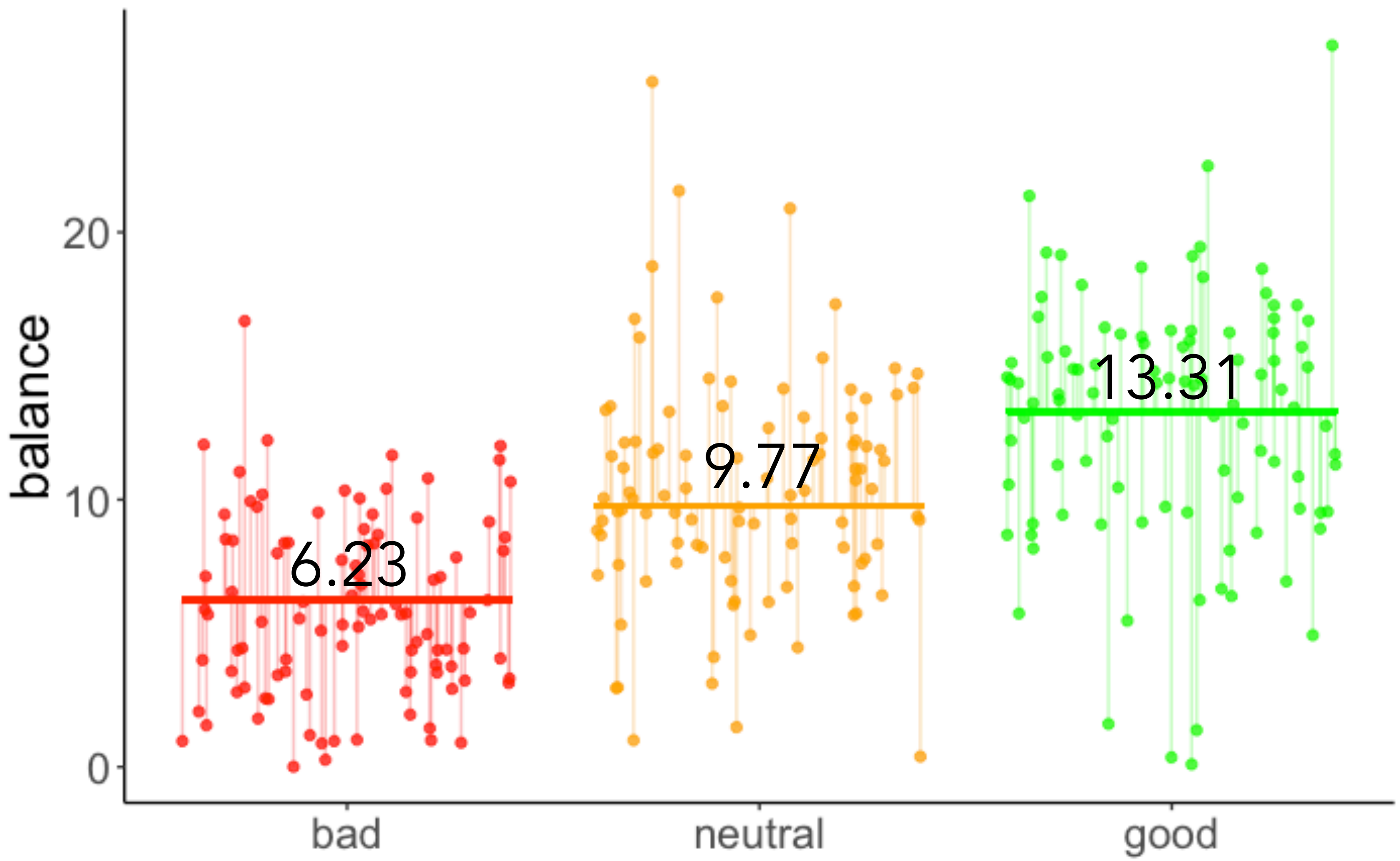
if contrast == 0

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

Contrasts

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

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

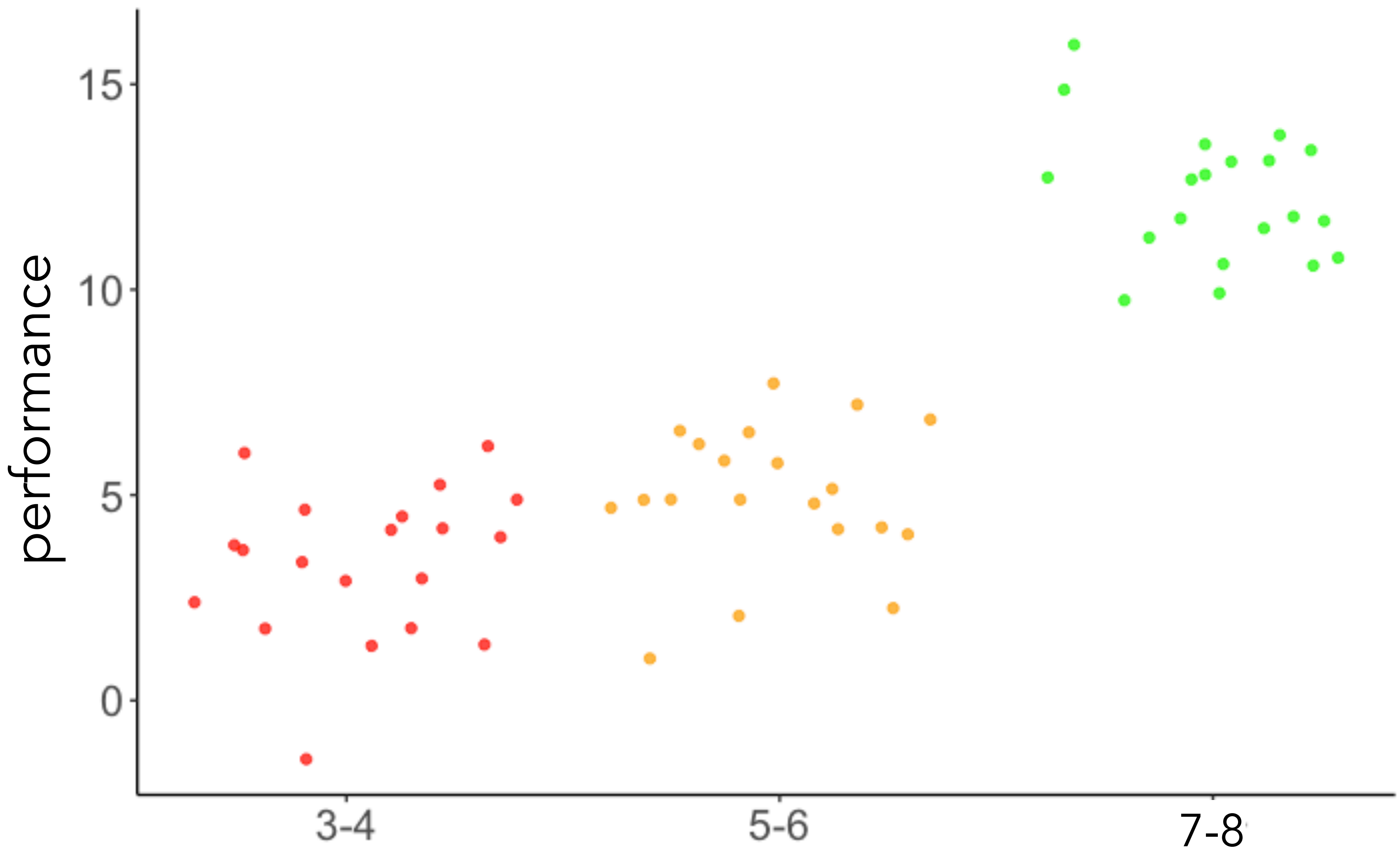


if contrast == 1

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

Contrasts

Does performance increase with age?



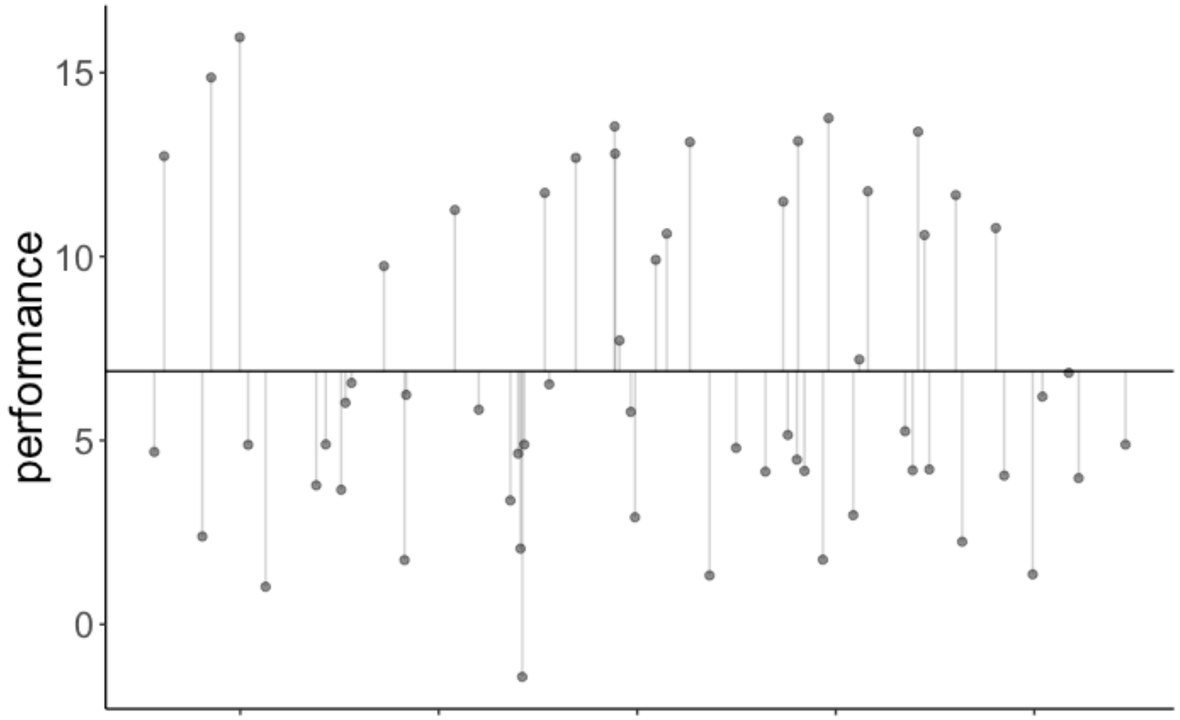
Data from a hypothetical developmental study

Contrasts

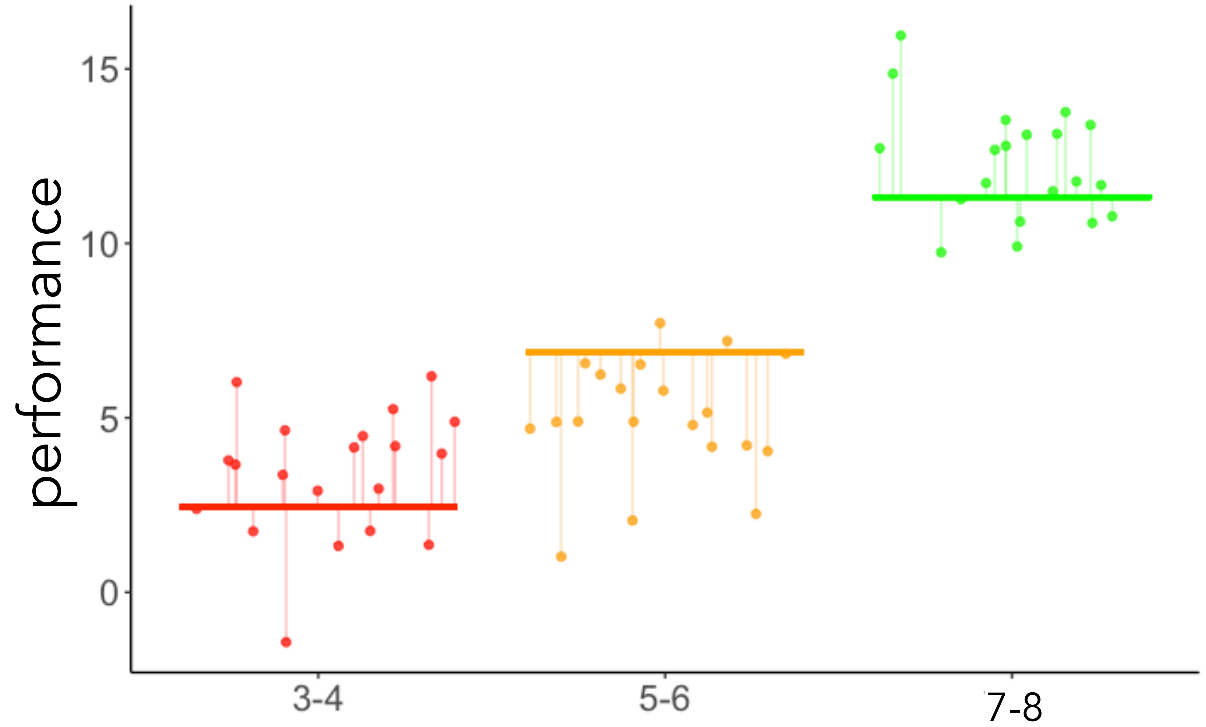
Does performance increase with age?

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

Compact model



Augmented model

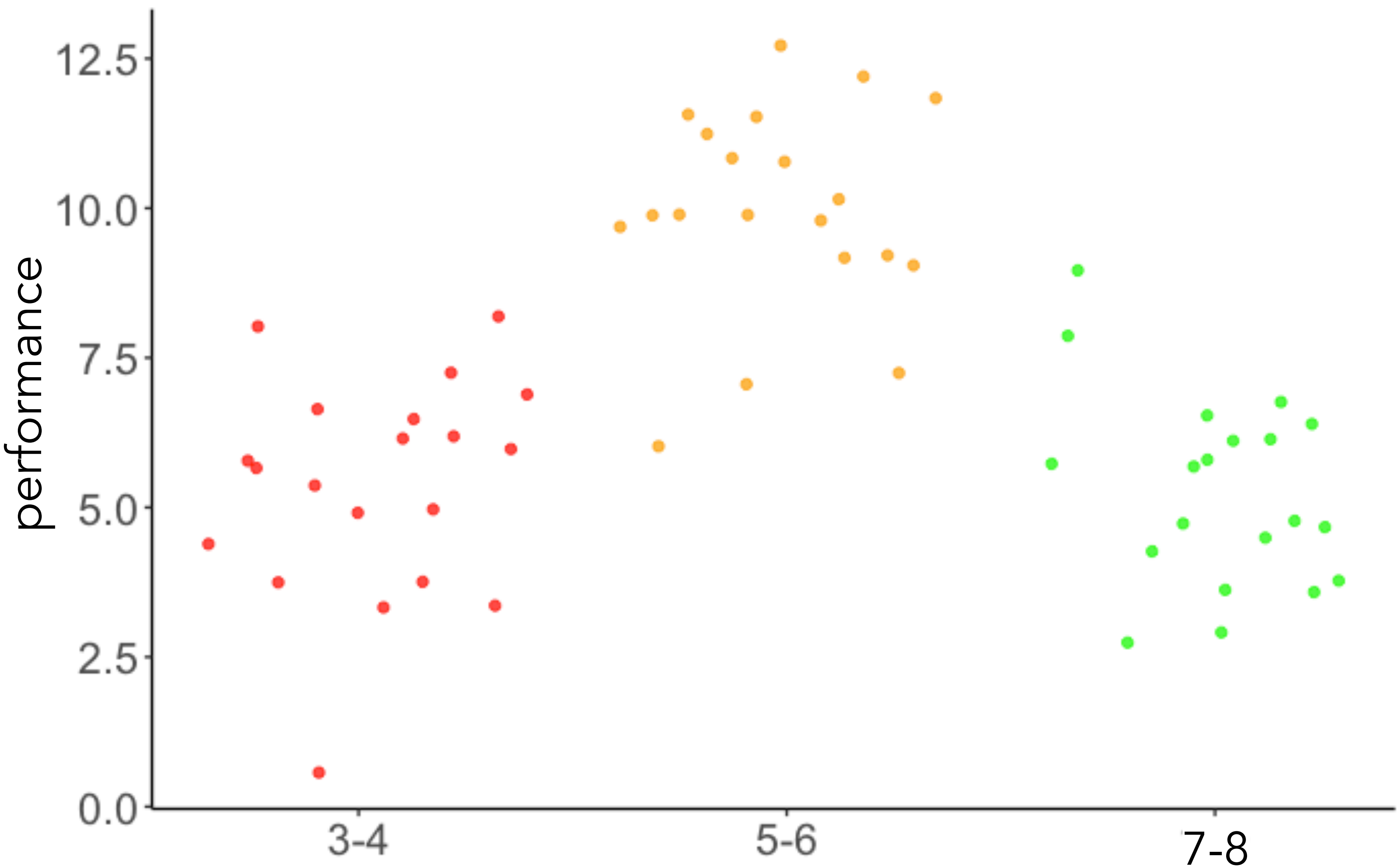


Model comparison

$p < .001$

Contrasts

Does performance increase with age?



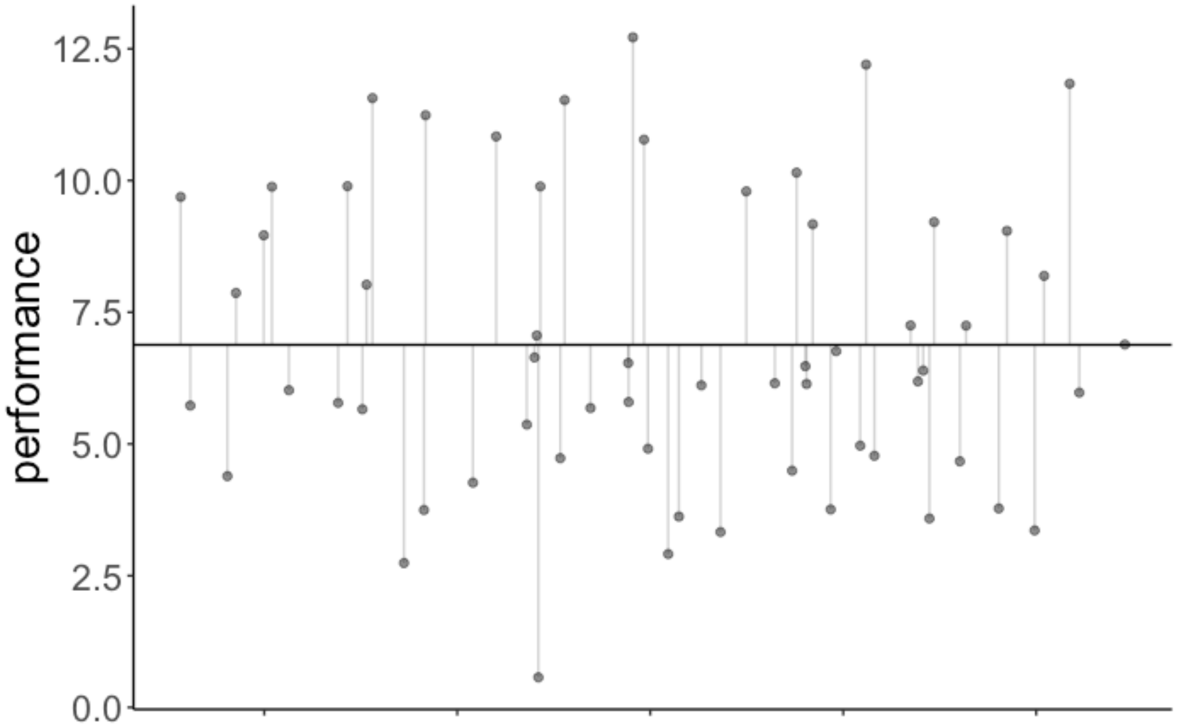
Data from another hypothetical developmental study

Contrasts

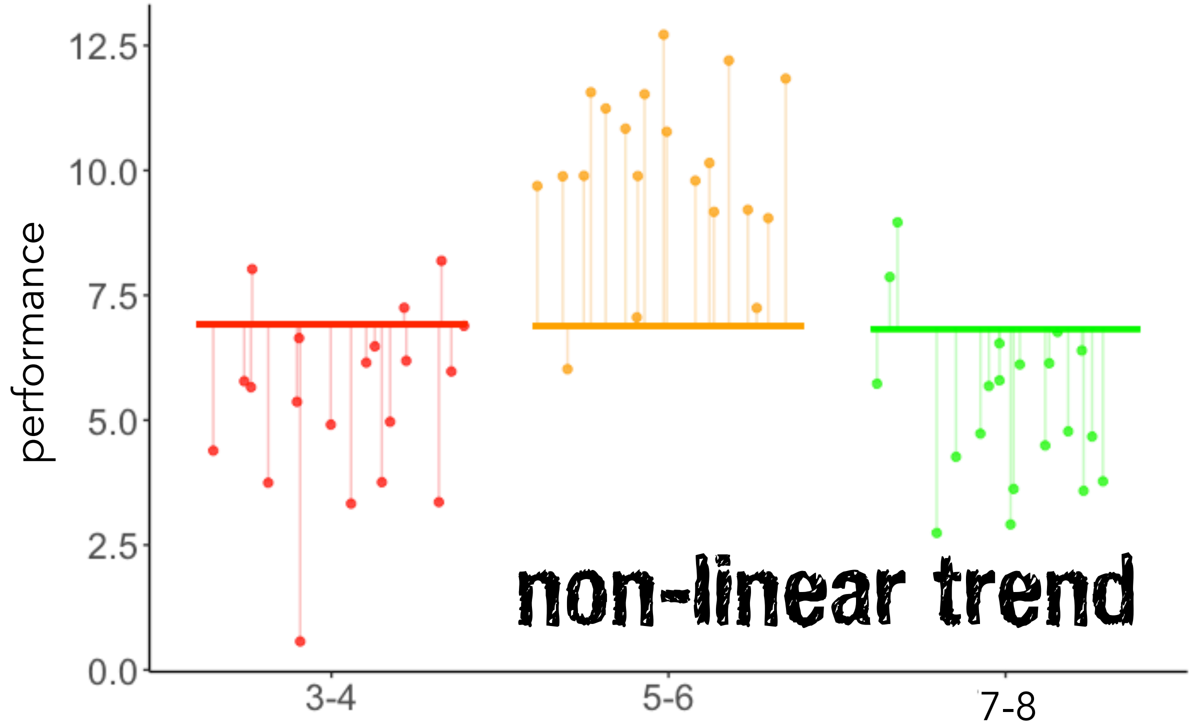
Does performance increase with age?

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

Compact model



Augmented model



Model comparison

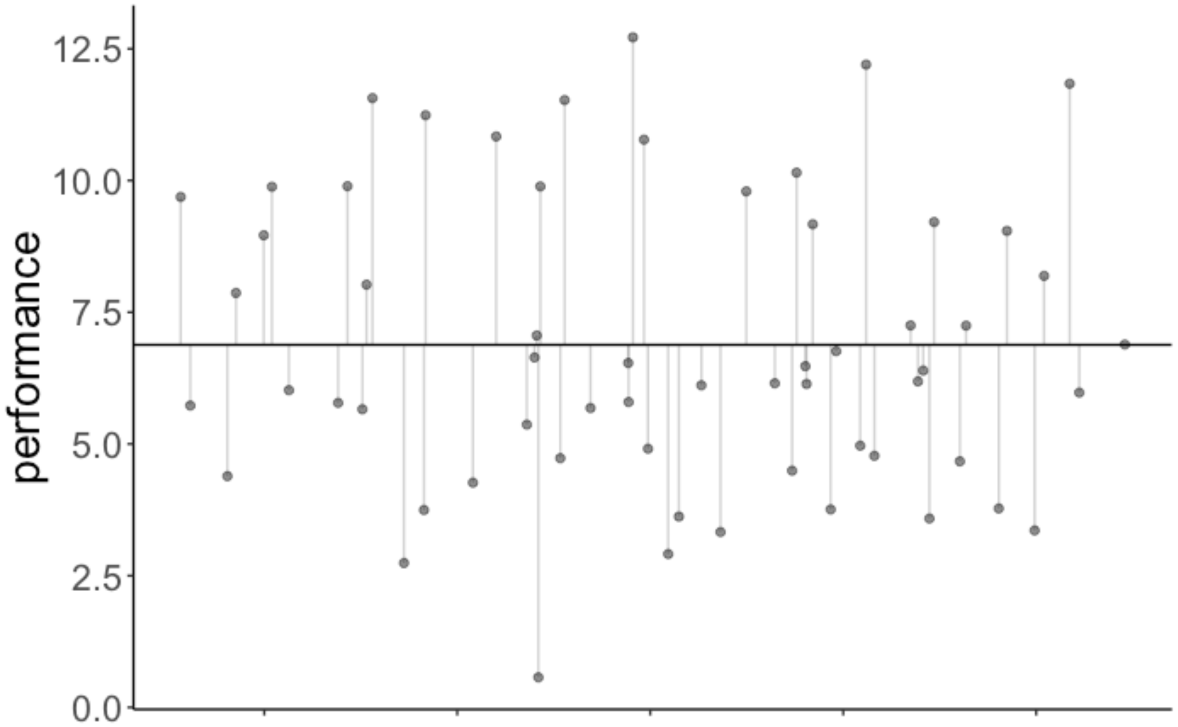
$p = .8508$

Contrasts

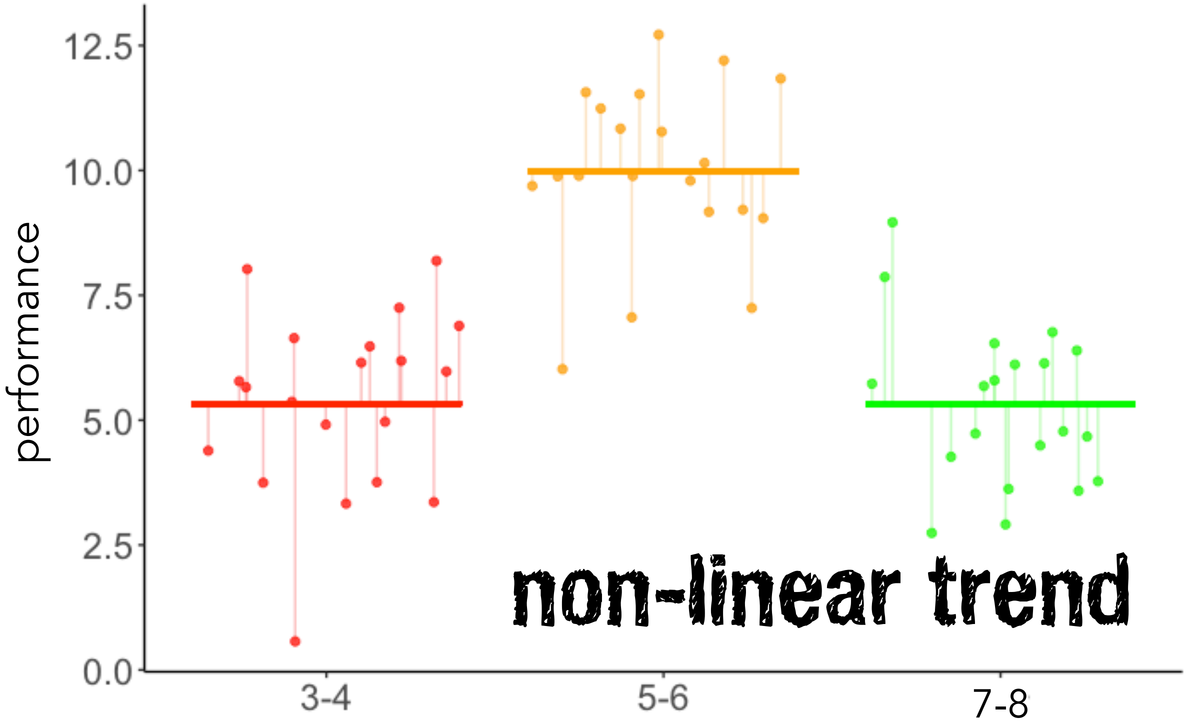
Does performance increase with age?

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

Compact model



Augmented model



Model comparison

$p < .001$

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
```

Summary

- Linear model with ...
  - categorical predictor that has more than two levels (One-way ANOVA)
  - multiple categorical predictors (N-way ANOVA)
  - linear contrasts

Feedback

How was the pace of today's class?

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!