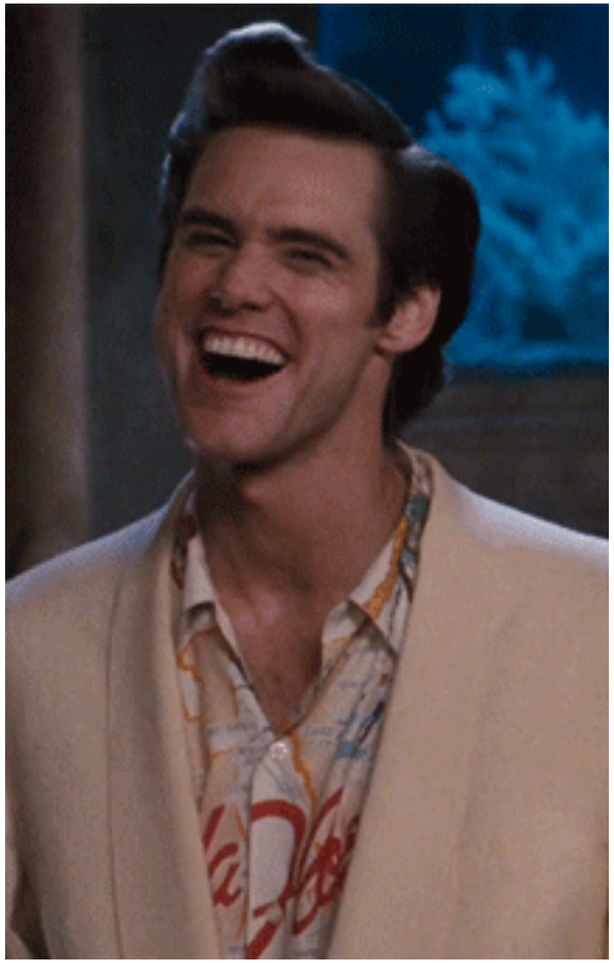
PSYCH 252: Statistical Methods for Behavioral and Social Sciences

Generalized linear model

02/07/2025

Logistics

Homework 3 grades released



Midterm

- will be released shortly after today's class
- midterm is take-home, open exam (you can use all the course notes, all of the internet, etc.)
- is due on **Friday, February 14th at 8pm** (stretch days can't be used for the midterm)
- midterm is like the other homeworks but:
  - it's longer
  - you'll need to work on it on your own
- there won't be office hours or sections in the week that the midterm is due
- **no class on Wednesday** next week
- ask questions about the midterm on Ed Discussion (please **ask private first** -- we will then respond and, if appropriate, check with you whether we can share with the rest of the class)

Midterm

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 **Friday, February 14th, 8pm**.

> Upload both the rendered pdf file (`252_midterm.pdf`) and the Rmarkdown code file (`252_midterm.rmd`) to Canvas.

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 Friday but before 8pm on Saturday, February 15th.
- No points will be granted if you submit later than 8pm on Saturday, February 15th.

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 lectures, as well as in **'Reporting Results.pdf'** on Canvas under `Files > papers`).

- 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.
- **Pro tip**: Read each question carefully so you don't miss any instructions. There are sometimes multiple sub-tasks or questions, and try to answer them all.

If you have any questions about the midterm, please post them on Ed Discussion addressed to the instructors only. We will answer your question and check with you whether we can share both your question and our answer with the rest of the group.

Best of luck with the midterm!

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.

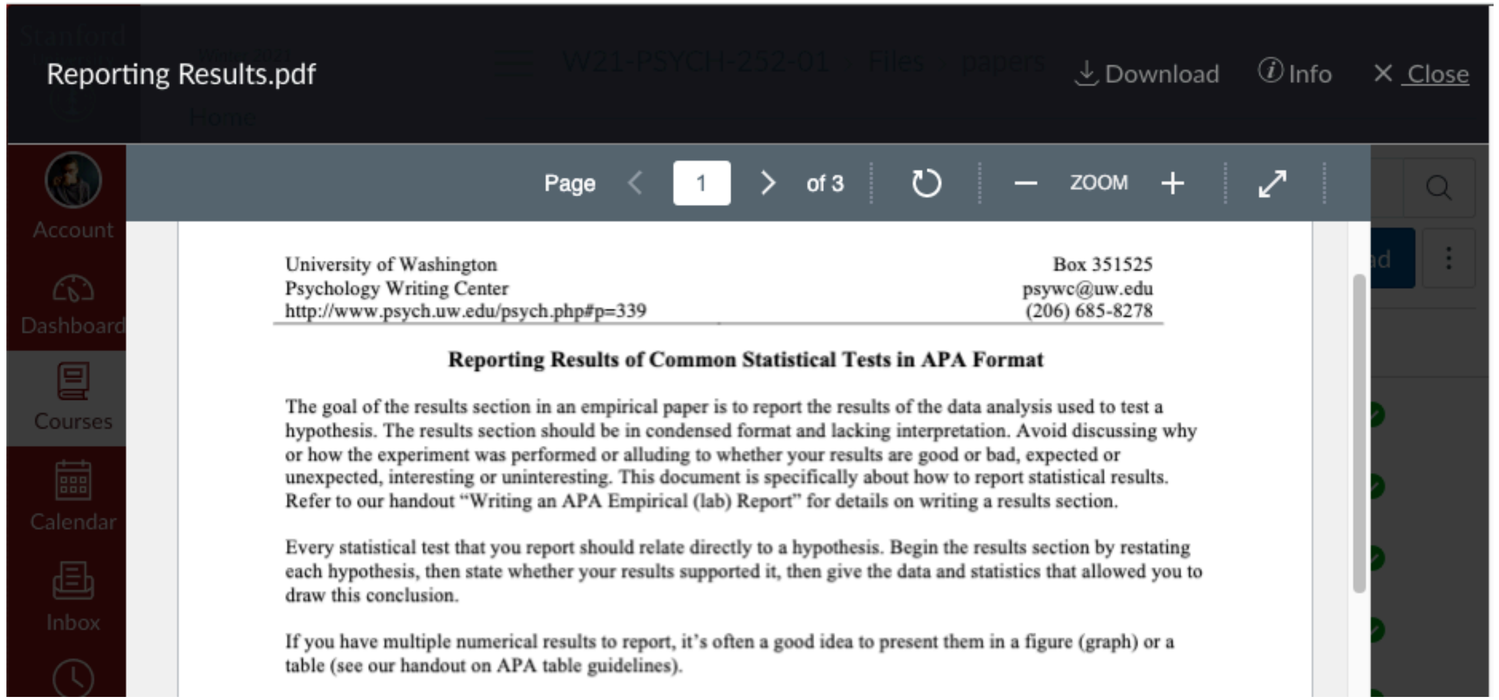
2. The faculty on its part manifests its confidence in the honor of its students by refraining from proctoring examinations and from taking unusual and unreasonable precautions to prevent the forms of dishonesty mentioned above. The faculty will also avoid, as far as practicable, academic procedures that create temptations to violate the Honor Code.

3. While the faculty alone has the right and obligation to set academic requirements, the students and faculty will work together to establish optimal conditions for honorable academic work.

3 parts, 40 points each

Reporting results

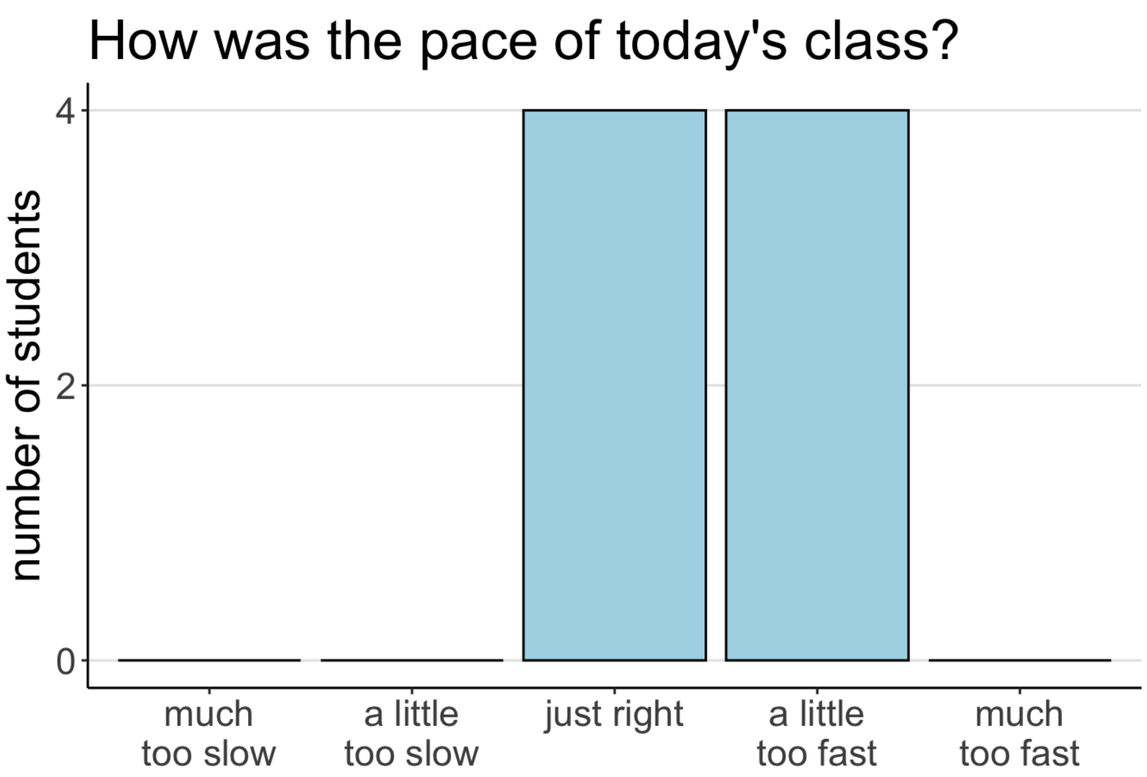
- When we ask you to report results in the homework, try and do this like you would do in a scientific paper (reporting the statistical test, test-statistic, p-value, effect size).

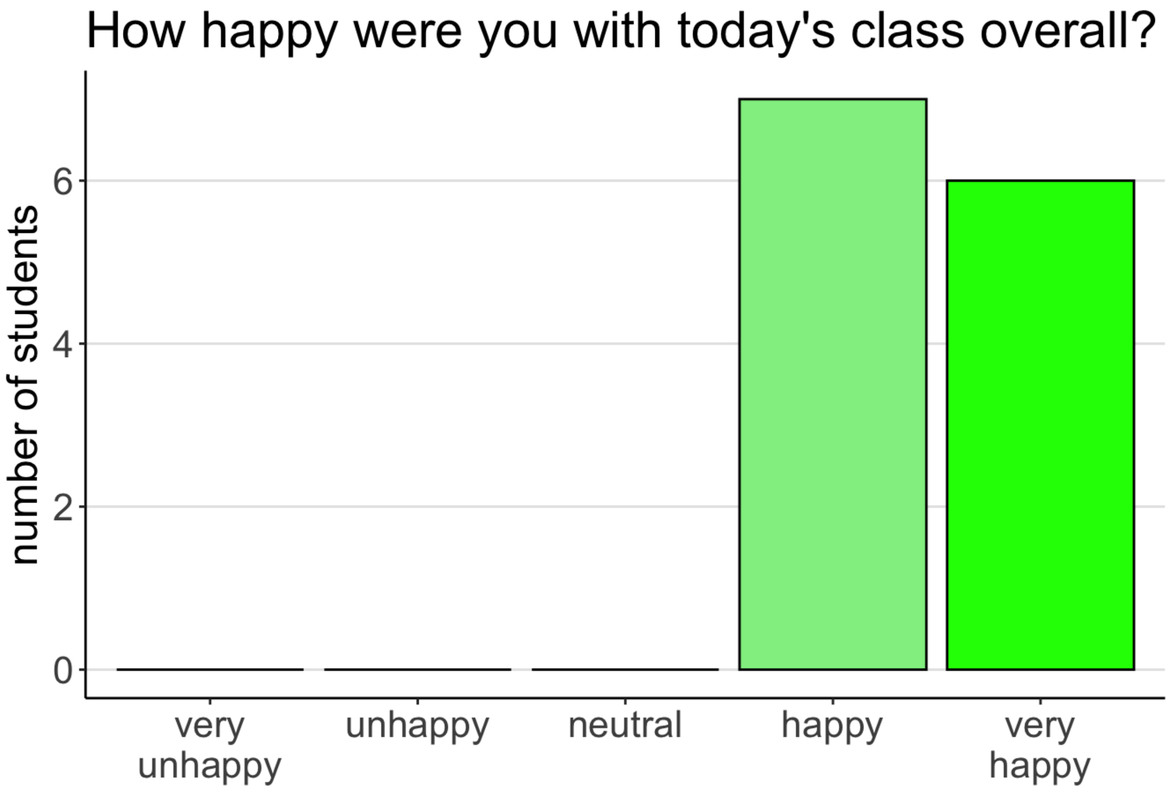


Files > papers > Reporting results.pdf

Feedback

Feedback


How was the pace of today's class?
number of students
4
2
0
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
6
4
2
0
very unhappy
unhappy
neutral
happy
very happy

Plan for today

- Quick Recap
- Interpreting parameters
- Who is the ANOVA champ?
- Unbalanced designs
- Linear contrasts
- Generalized linear model
  - Logistic regression
    - interpreting the model output
    - fitting and reporting models

Quick recap

Quick recap: Interactions

H_0: The relationship between income and balance is the same for students and non-students.

H_1: The relationship between income and balance differs between students and non-students.

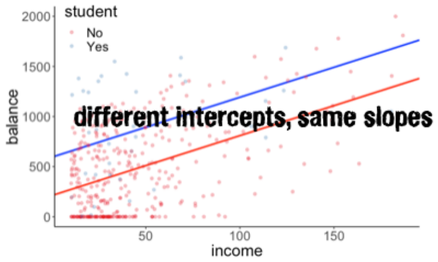
Model C

$$\text{balance}_i = \beta_0 + \beta_1 \text{income}_i + \beta_2 \text{student}_i + \epsilon_i$$

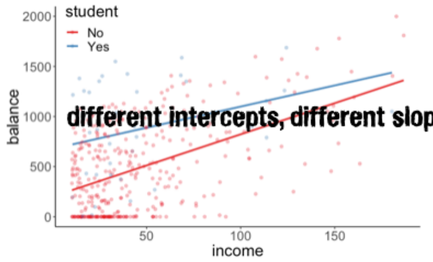
Model A

$$\widehat{\text{balance}}_i = b_0 + b_1 \text{income}_i + b_2 \text{student}_i + b_3(\text{income}_i \times \text{student}_i) + \epsilon_i$$

Model prediction



Model prediction



Fitted model

$$\widehat{\text{balance}}_i = 211.14 + 5.98 \cdot \text{income}_i + 382.67 \cdot \text{student}_i$$

Fitted model

$$\widehat{\text{balance}}_i = 200.62 + 6.22 \cdot \text{income}_i + 476.68 \cdot \text{student}_i - 2.00 \cdot (\text{income}_i \times \text{student}_i)$$

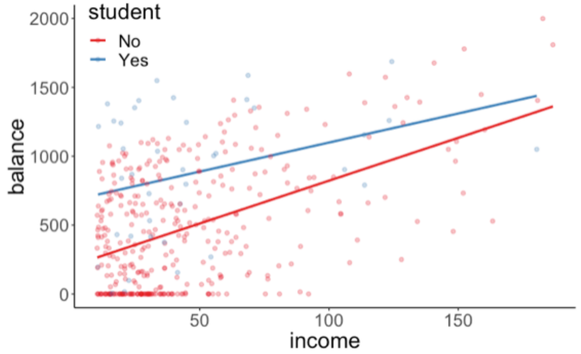
```
Call:
lm(formula = balance ~ 1 + income + student, data = df.credit)

Residuals:
    Min      1Q  Median      3Q     Max
-762.37 -331.38  -45.04  323.60  818.28

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 211.1430    32.4572   6.505 2.34e-10 ***
income        5.9843     0.5566  10.751  < 2e-16 ***
studentYes  382.6705    65.3108   5.859 9.78e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 391.8 on 397 degrees of freedom
Multiple R-squared:  0.2775, Adjusted R-squared:  0.2738
F-statistic: 76.22 on 2 and 397 DF,  p-value: < 2.2e-16
```

effect of income for non-students

```
Call:
lm(formula = balance ~ income + student + income:student, data = df.credit)

Residuals:
    Min      1Q  Median      3Q     Max
-773.39 -325.70  -41.13  321.65  814.04

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       200.6232    33.6984   5.953 5.79e-09 ***
income              6.2182     0.5921  10.502  < 2e-16 ***
studentYes        476.6758   104.3512   4.568 6.59e-06 ***
income:studentYes  -1.9992     1.7313  -1.155    0.249
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 391.6 on 396 degrees of freedom
Multiple R-squared:  0.2799, Adjusted R-squared:  0.2744
F-statistic:  51.3 on 3 and 396 DF,  p-value: < 2.2e-16
```

effect of income for non-students

Interpretation



$$\widehat{\text{balance}}_i = 200.62 + 6.22 \cdot \text{income}_i + 476.68 \cdot \text{student}_i - 2.00 \cdot (\text{income}_i \times \text{student}_i) + 0$$

if student = "No" $\widehat{\text{balance}}_i = 200.62 + 6.22 \cdot \text{income}_i + (476.68 \cdot 0) - 2.00 \cdot (\text{income}_i \times 0)$

if student = "Yes"

$$\begin{aligned}\widehat{\text{balance}}_i &= 200.62 + 6.22 \cdot \text{income}_i + 476.68 \cdot 1 - 2.00 \cdot (\text{income}_i \times 1) \\ &= 677.3 + 6.22 \cdot \text{income}_i - 2.00 \cdot \text{income}_i \\ &= 677.3 + 4.22 \cdot \text{income}_i\end{aligned}$$

22

Quick recap: `lm()` output

```
1 lm(formula = balance ~ income + student + income:student, data = df.credit) %>%
2   summary()
```

```
Call:
lm(formula = balance ~ income + student + income:student,
data = df.credit)

Residuals:
    Min      1Q  Median      3Q     Max
-773.39 -325.70  -41.13  321.65  814.04

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       200.6232    33.6984   5.953 5.79e-09 ***
income              6.2182     0.5921  10.502  < 2e-16 ***
studentYes        476.6758   104.3512   4.568 6.59e-06 ***
income:studentYes  -1.9992     1.7313  -1.155    0.249
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’
1

Residual standard error: 391.6 on 396 degrees of freedom
Multiple R-squared:  0.2799,  Adjusted R-squared:  0.2744
F-statistic:  51.3 on 3 and 396 DF,  p-value: < 2.2e-16
```

assesses a specific predictor

```
1 fit_c = lm(formula = balance ~ student + income:student, data = df.credit)
2 fit_a = lm(formula = balance ~ income + student + income:student, data = df.credit)
3
4 anova(fit_c, fit_a)
```

assesses the full model

```
1 fit_c = lm(formula = balance ~ 1, data = df.credit)
2 fit_a = lm(formula = balance ~ 1 + income + student + income:student, data = df.credit
3
4 anova(fit_c, fit_a)
```

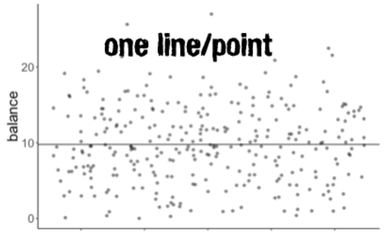
Quick recap: Categorical variables (single predictor)

H_0: Card quality does not affect the final balance.

Model C

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

Model prediction



Fitted model

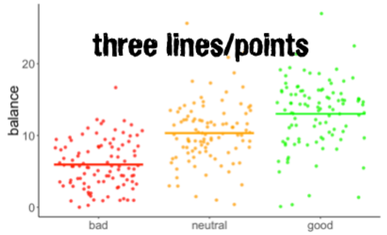
$\widehat{\text{balance}}_i = 9.77$

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



Fitted model

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

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

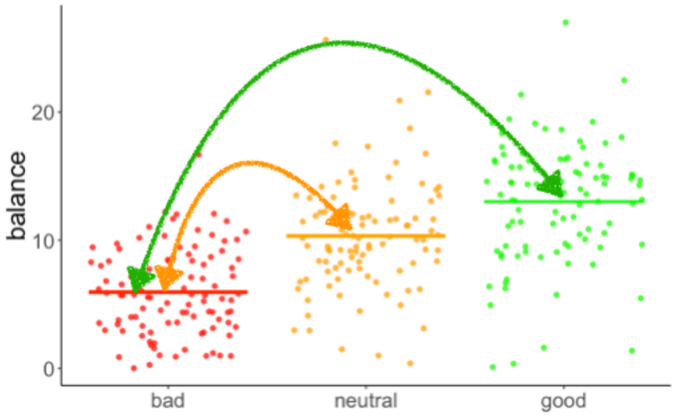
```
Call:
lm(formula = balance ~ hand, data = df.poker)

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
```

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

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



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 parameters

Parameter interpretation

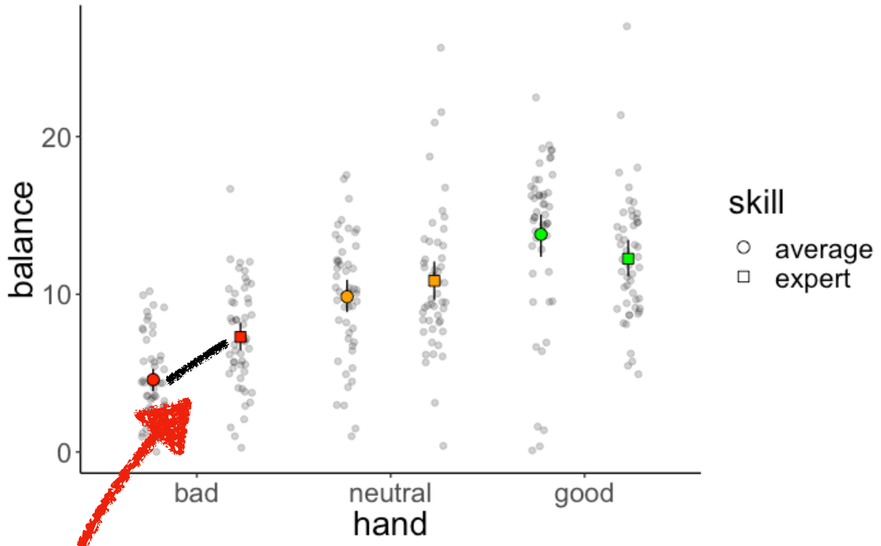
```
lm(formula = balance ~ 1 + hand * skill, data = df.poker) %>%
  summary()
```

```
Call:
lm(formula = balance ~ 1 + 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 ***
---
Sign                                                          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
```

acute danger of misinterpretation!!

~~there was a significant effect of skill~~

Parameter interpretation

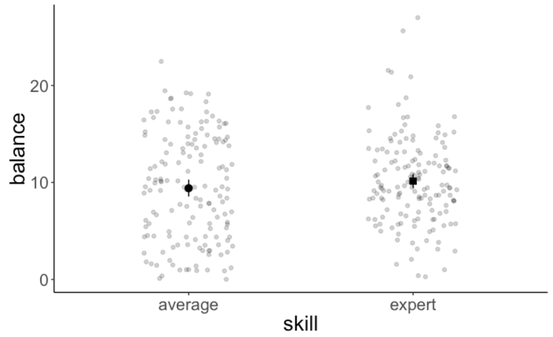



```
Call:
lm(formula = balance ~ hand * skill, data = df.poker)

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
```

```
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
```

there was no main effect of skill!



is this difference significantly different from 0?

hand	average	expert	difference
bad	4.59	7.3	2.71

Different effect terms

- **main effect**: effect of one independent variable on the dependent variable
- **interaction effect**: when the effect of one independent variable depends on the level of another
- **simple effect**: comparison between two specific cell means

https://opentext.wsu.edu/carriecuttler/chapter/9-2-main-effects-and-interactions/

Interpreting parameters

`lm()` gives **simple effects**

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

```
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 0.135058
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
```

`anova()` gives **main effects**, and **interactions**

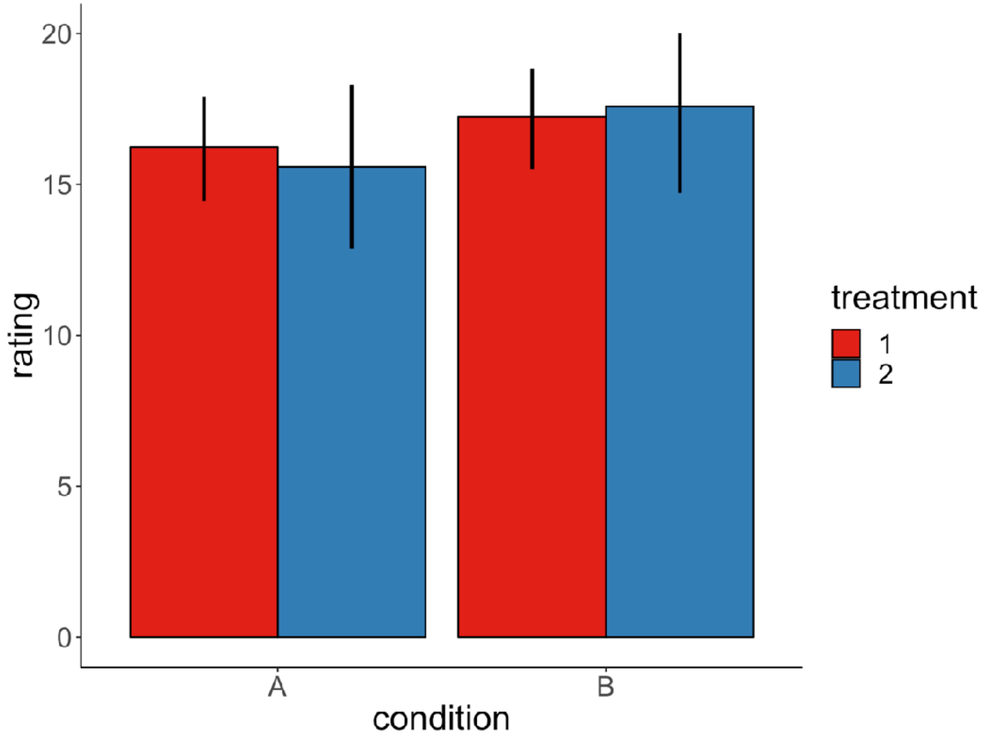
```
lm(formula = balance ~ hand * skill,
   data = df.poker) %>%
  anova()
```

```
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
```

Who is the ANOVA champ?

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

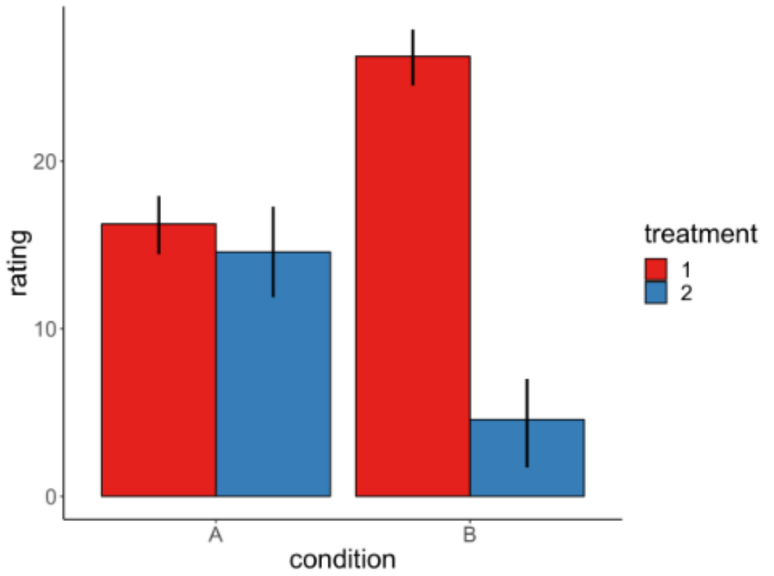
The winner gets chocolate!

Who is the ANOVA champ?

Get ready to compete!

Start the presentation to see live content. Still no live content? Install the app or get help at **PollEv.com/app**

Which effects are significant?



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

Start the presentation to see live content. Still no live content? Install the app or get help at **PollEv.com/app**

Total Results

Which effects are significant?

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

Start the presentation to see live content. Still no live content? Install the app or get help at **PollEv.com/app**

Total Results

Which effects are significant?

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

Start the presentation to see live content. Still no live content? Install the app or get help at **PollEv.com/app**

Total Results

Leaderboard

Start the presentation to see live content. Still no live content? Install the app or get help at **PollEv.com/app**

Which effects are significant?

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

Start the presentation to see live content. Still no live content? Install the app or get help at **PollEv.com/app**

Total Results

Which effects are significant?

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

Start the presentation to see live content. Still no live content? Install the app or get help at **PollEv.com/app**

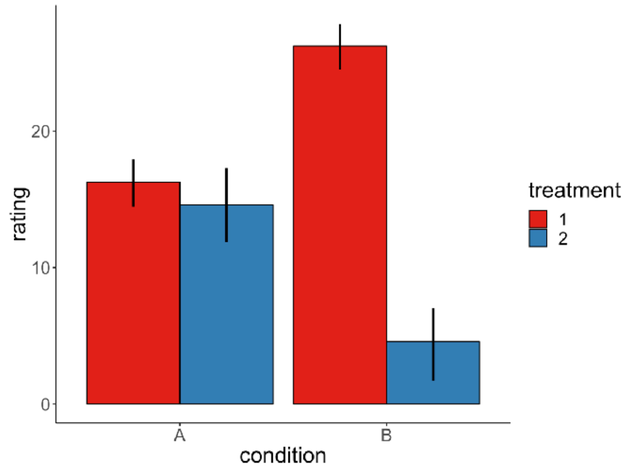
Total Results

Leaderboard

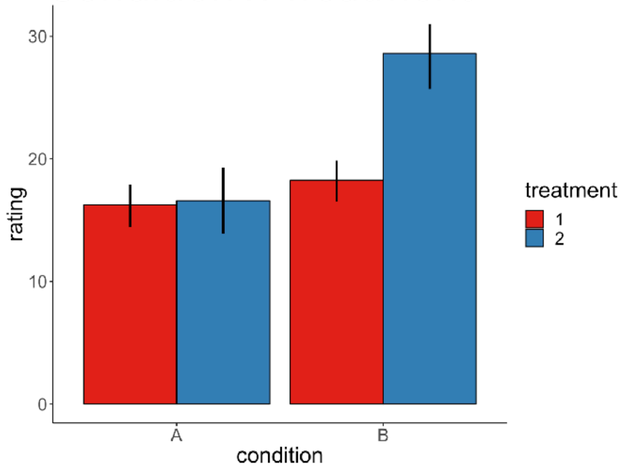
Start the presentation to see live content. Still no live content? Install the app or get help at **PollEv.com/app**

Solution

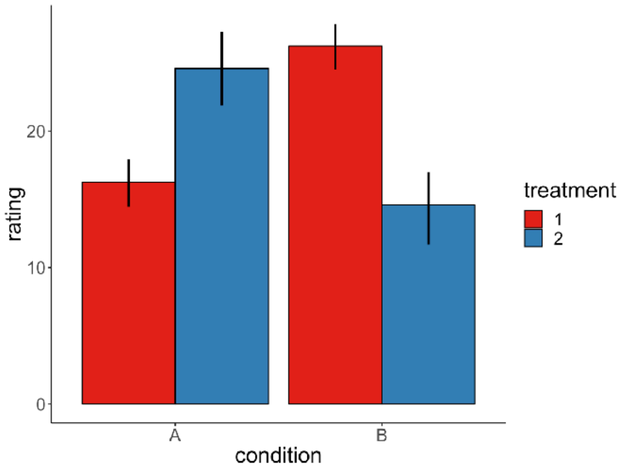
Treatment
Condition x Treatment



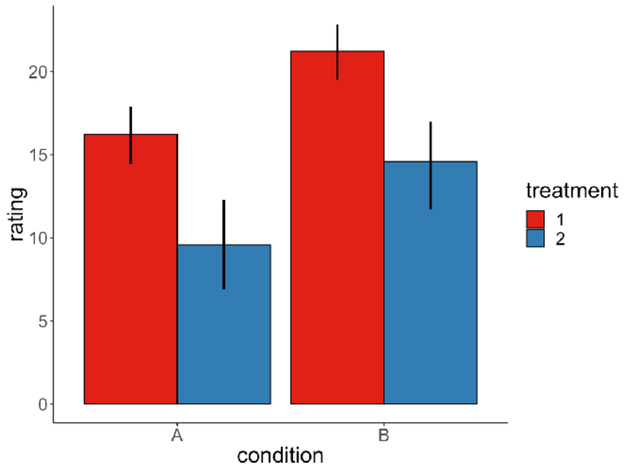
Condition,
Treatment,
Condition x Treatment



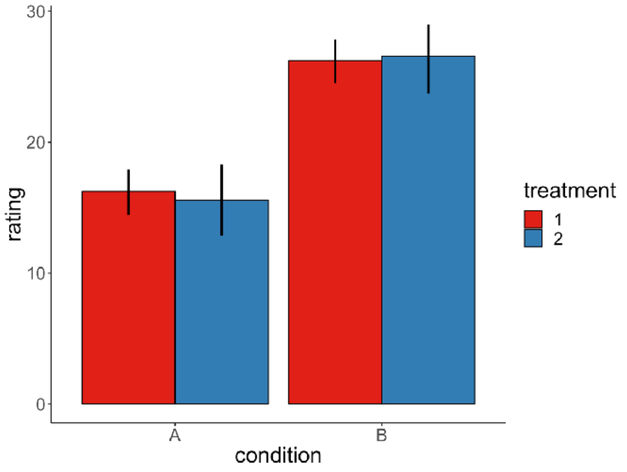
Condition x Treatment



Condition
Treatment

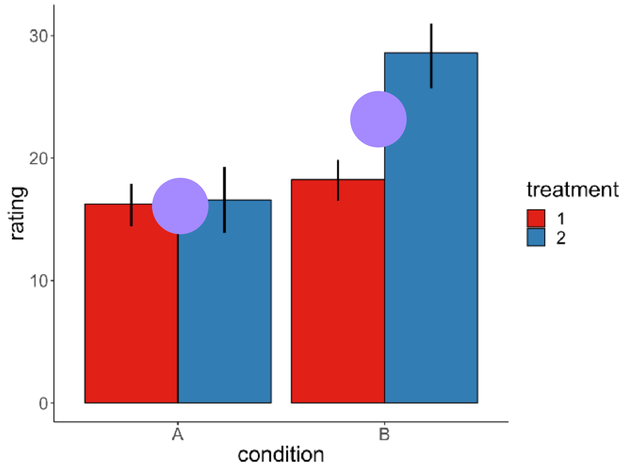


Condition



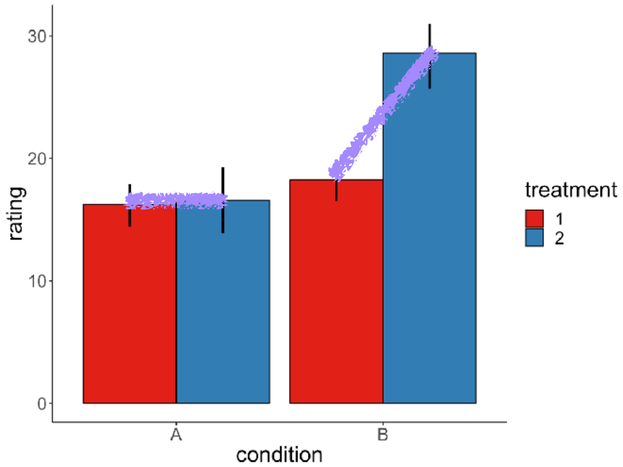
Solution

to detect main effects, try to visualize what the averaged group means would look like



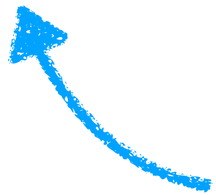
main effect of condition

to detect interaction effects, try to visualize whether the slopes are different from each other



interaction effect

Unbalanced designs



not the same number of participants in each cell

ANOVA

- for all the examples so far, 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

https://towardsdatascience.com/anovas-three-types-of-estimating-sums-of-squares-don-t-make-the-wrong-choice-91107c77a27a

Beware of unbalanced designs

```
1 lm(formula = balance ~ skill + hand, data = df.poker.unbalanced) %>%
2   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

```
1 lm(formula = balance ~ hand + skill, data = df.poker.unbalanced) %>%
2   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
```

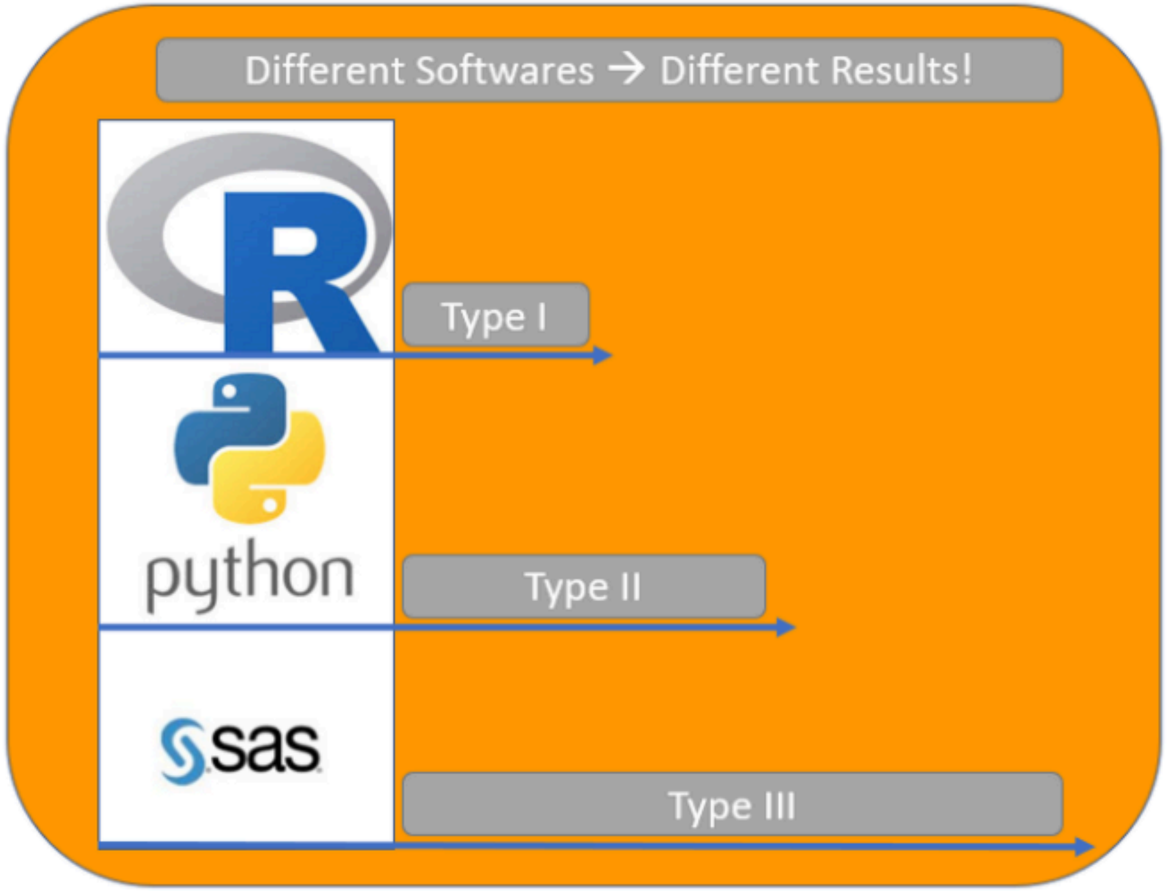
The different sums of squares

Three different methodologies for splitting variation exist: Type I, Type II and Type III Sums of Squares. They do not give the same result in case of unbalanced data.

Type I, Type II and Type III ANOVA have different outcomes!

https://towardsdatascience.com/anovas-three-types-of-estimating-sums-of-squares-don-t-make-the-wrong-choice-91107c77a27a

Default sums of squares ...



Default Types of Sums of Squares for different programming languages

not great for reproducibility ...

Type I Sums of Squares

Type I Sums of Squares are Sequential, so the order of variables in the models makes a difference. This is rarely what we want in practice!

Sums of Squares are Mathematically defined as:

- SS(A) for independent variable A
- SS(B | A) for independent variable B
- SS(AB | B, A) for the interaction effect

caution: this is what `anova()` uses by default

https://towardsdatascience.com/anovas-three-types-of-estimating-sums-of-squares-don-t-make-the-wrong-choice-91107c77a27a

Type III Sums of Squares

The Type III Sums of Squares are also called partial sums of squares again another way of computing Sums of Squares:

- Like Type II, the Type III Sums of Squares are not sequential, so the order of specification does not matter.
- Unlike Type II, the Type III Sums of Squares do specify an interaction effect.

Sums of Squares are Mathematically defined as:

- SS(A | B, AB) for independent variable A
- SS(B | A, AB) for independent variable B

this is the default in the literature (e.g. SPSS, SAS, Stata etc use it)

Route I: Using "afex"

```r
library("afex")

fit = aov_ez(id = "participant",
             dv = "balance",
             data = df.poker.unbalanced,
             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
```

Route II: Using "emmeans"

preferred route!!

```
1 library("emmeans")
2
3 lm(formula = balance ~ hand * skill,
4    data = df.poker.unbalanced) %>%
5   joint_tests()
```

very handy function

```
model term df1 df2 F.ratio p.value
 hand          2 284  68.973  <.0001
 skill         1 284   2.954  0.0868
 hand:skill    2 284   7.440  0.0007
```

Unbalanced design

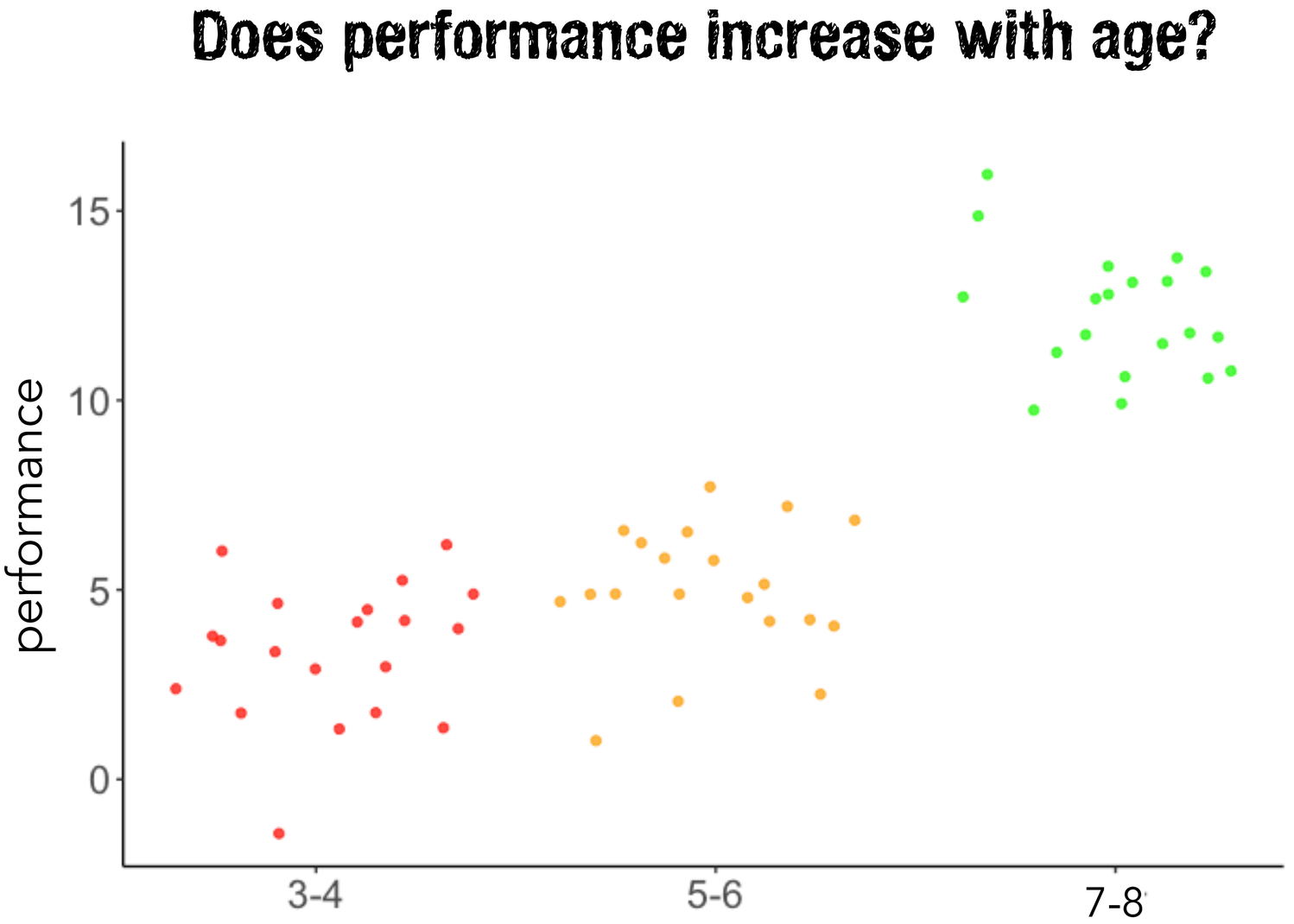
- There are different kinds of ANOVAs, for which the sums of squares are calculated differently.
- This makes a difference when we have an unbalanced design (i.e. the number of participants is not the same for each cell in our design).

`joint_tests()` is your friend!

Linear contrasts

Testing (more) specific hypotheses with linear contrasts

Contrasts


Does performance increase with age?
performance
15
10
5
0
3-4
5-6
7-8
Data from a hypothetical developmental study

Does performance increase with age?

ANOVA

Does performance differ between age groups?

post-hoc tests

3-4 vs. 5-6

5-6 vs. 7-8

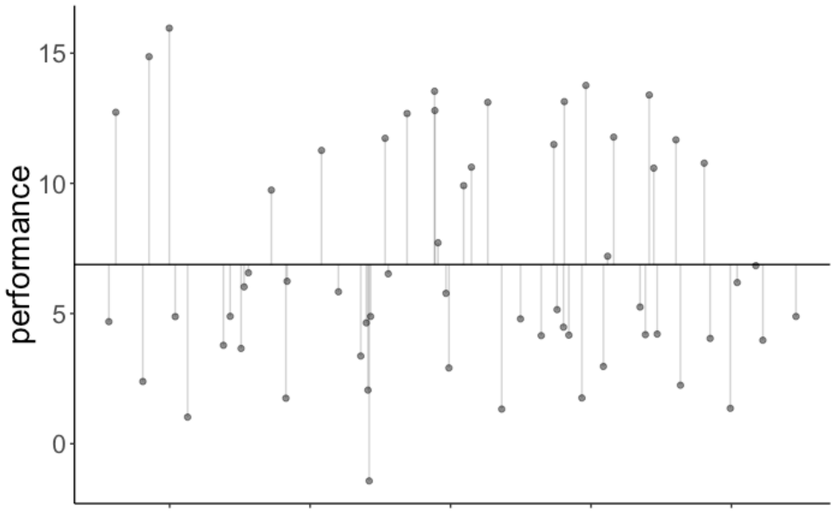
Is there are more direct way of asking this question with a statistical model?

Contrasts

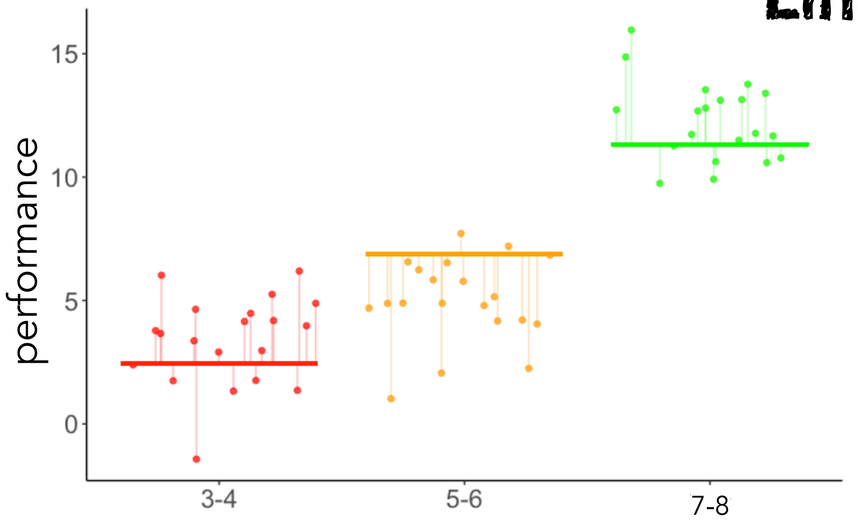
Does performance increase with age?

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

Compact model



Augmented model



Linear contrast

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.

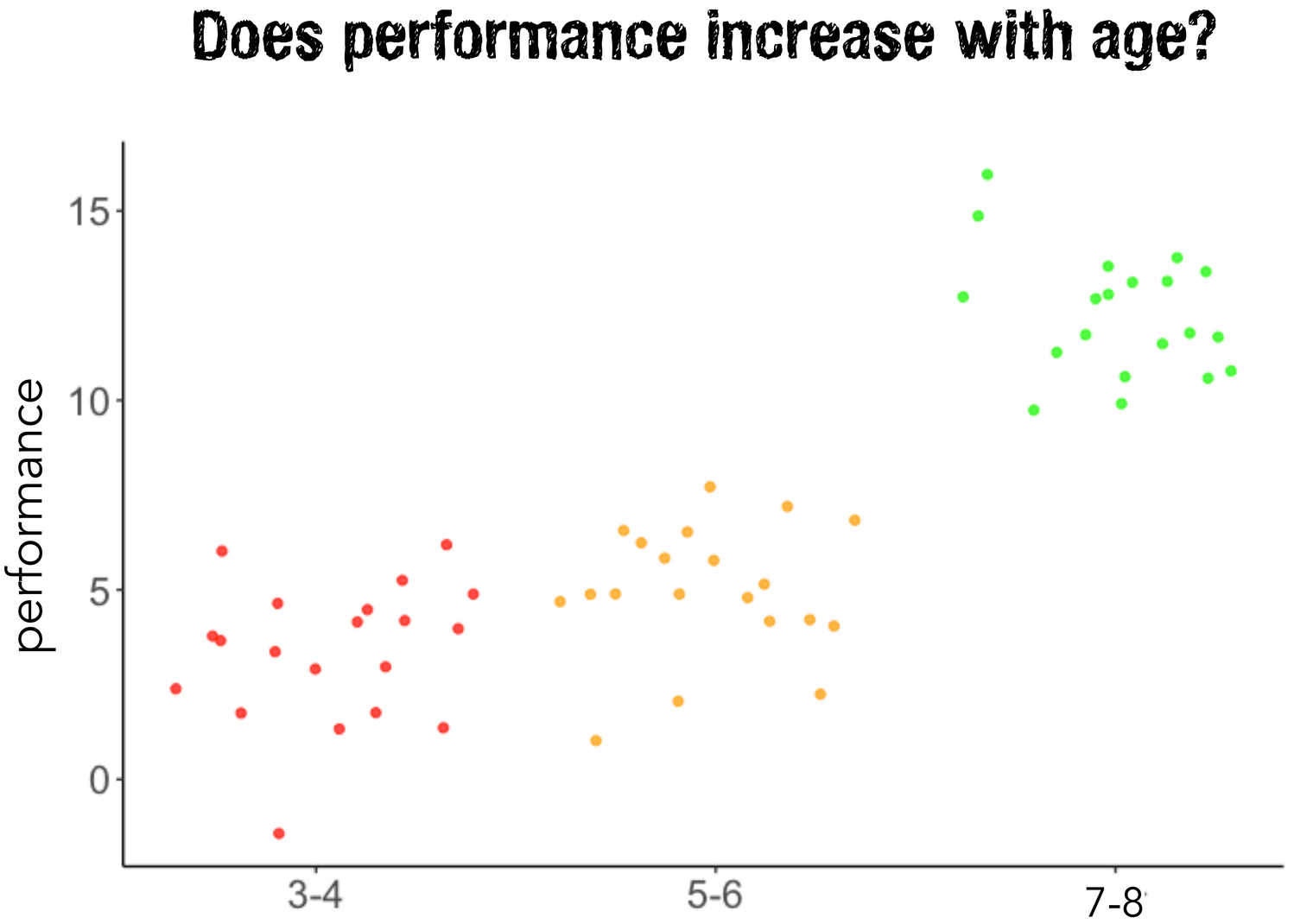
Luckily for me, someone came along and fixed the situation: `emmeans`. `emmeans` frames contrasts as a question you pose to a model: you can ask for all pairwise comparisons and get back that. `lm` and `summary` treat the same problem as fitting abstract coefficients, and you are left to answer your 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?
performance
15
10
5
0
3-4
5-6
7-8
Data from a hypothetical developmental study

Linear contrasts in R

```
1 library("emmeans") # for calculating contrasts
2
3 #  fit the linear model
4 fit = lm(formula = performance ~ 1 + group,
5           data = df.development)
```

fit linear model

Linear contrasts in R

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

#  fit the linear model
fit = lm(formula = performance ~ 1 + 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

```
1  library("emmeans") # for calculating contrasts
2
3  #  fit the linear model
4  fit = lm(formula = performance ~ group,
5           data = df.development)
6
7  # check factor levels
8  levels(df.development$group)
9                                          [1] "3-4" "5-6" "7-8"
10 # define the contrasts of interest
11 contrasts = list(young_vs_old = c(-0.5, -0.5, 1),
12                   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")
```

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

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")
```

```r
# 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))
```

```
[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
```

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
```

Defining contrasts

- groups that we don't want to include in the comparison get a 0
- groups that we want to compare with one another should sum to 0
- this also means that all the contrasts together should sum to 0

Example:

```
contrasts = list(young_vs_old = c(-1, -1, 2),
                 three_vs_five = c(-1, 1, 0))
```

Post hoc tests

```
1 fit = lm(formula = performance ~ 1 + group,
2           data = df.development)
3
4 # pairwise differences between all the groups
5 fit %>%
6   emmeans(pairwise ~ group) %>%
7   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 ~ 1 + 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

all pairwise tests between groups

that's a lot of tests!

... not

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?

Generalized linear model


Titanic dataset

Titanic data set

891 passengers

passenger_id	survived	pclass	name	sex	age	sib_sp	parch	ticket	fare	cabin	embarked
1	0	3	Braund, Mr. Owen Harris	male	22	1	0	A/5 21171	7.25		S
2	1	1	Cumings, Mrs. John Bradley (Florence Briggs	female	38	1	0	PC 17599	71.28	C85	C
3	1	3	Heikkinen, Miss. Laina	female	26	0	0	STON/O2. 3101282	7.92		S
4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35	1	0	113803	53.10	C123	S
5	0	3	Allen, Mr. William Henry	male	35	0	0	373450	8.05		S
6	0	3	Moran, Mr. James	male	NA	0	0	330877	8.46		Q
7	0	1	McCarthy, Mr. Timothy J	male	54	0	0	17463	51.86	E46	S
8	0	3	Palsson, Master. Gosta Leonard	male	2	3	1	349909	21.07		S
9	1	3	Johnson, Mrs. Oscar W (Elisabeth	female	27	0	2	347742	11.13		S
10	1	2	Nasser, Mrs. Nicholas (Adele Achem)	female	14	1	0	237736	30.07		C

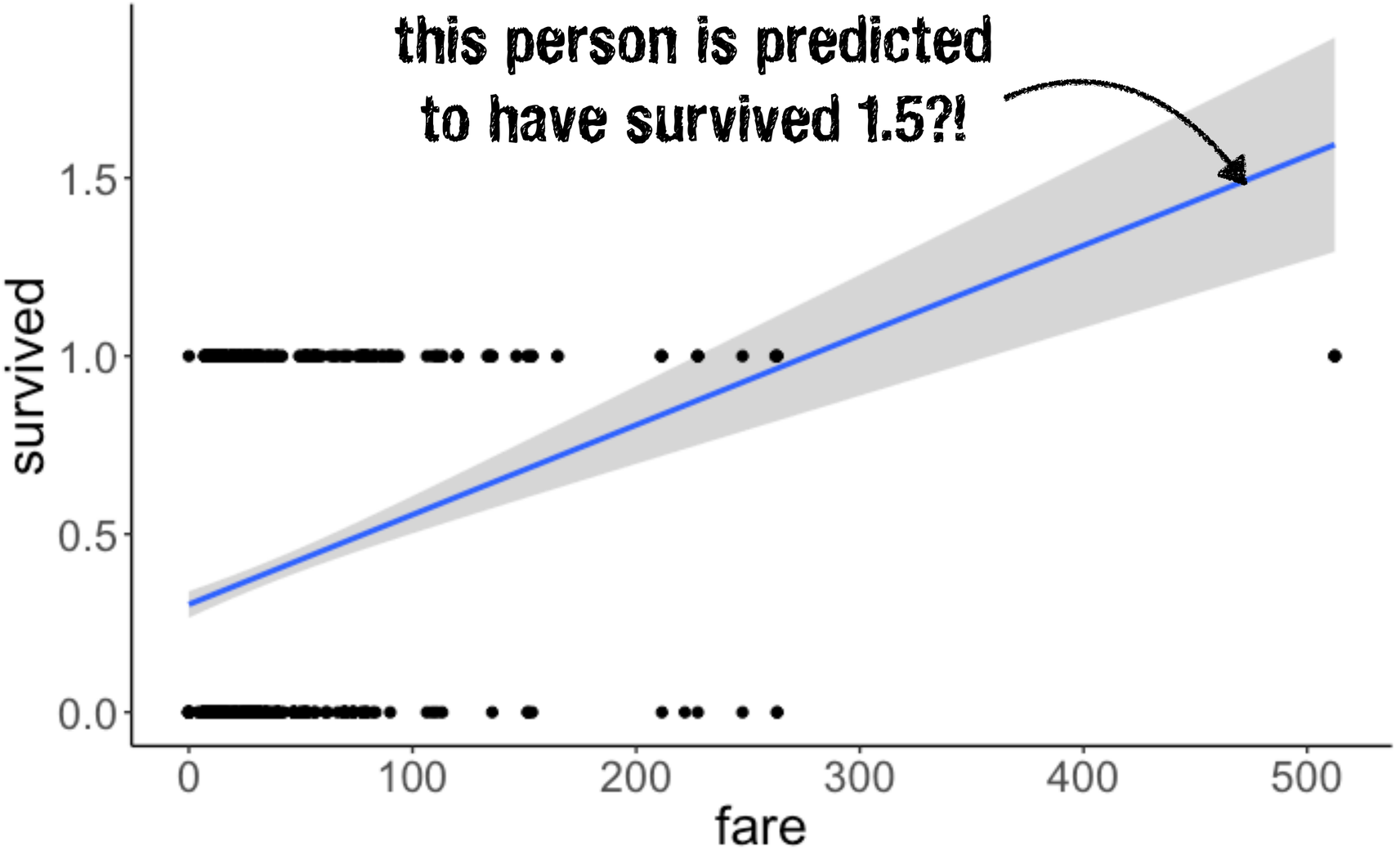
Is there a relationship between `fare` and `survived`?

```
1 fit.lm = lm(formula = survived ~ 1 + fare,
2               data = df.titanic)
3
4 fit.lm %>% summary()
```

```
Call:
lm(formula = survived ~ 1 + fare, data = df.titanic)

Residuals:
    Min      1Q  Median      3Q     Max
-0.9653 -0.3391 -0.3222  0.6044  0.6973

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.3026994  0.0187849  16.114  < 2e-16 ***
fare        0.0025195  0.0003174   7.939 6.12e-15 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4705 on 889 degrees of freedom
Multiple R-squared:  0.06621,  Adjusted R-squared:  0.06516
F-statistic: 63.03 on 1 and 889 DF,  p-value: 6.12e-15
```

How should we interpret this parameter?

Is there a relationship between `fare` and `survived`?



Generalized linear model

- so far, we have only looked at situations where our dependent variable was continuous
- what about situations in which we have a binary dependent variable?
  - survived vs. died
  - correct vs. incorrect
  - benign vs. malignent
  - yes vs. no
  - ...

Logistic regression

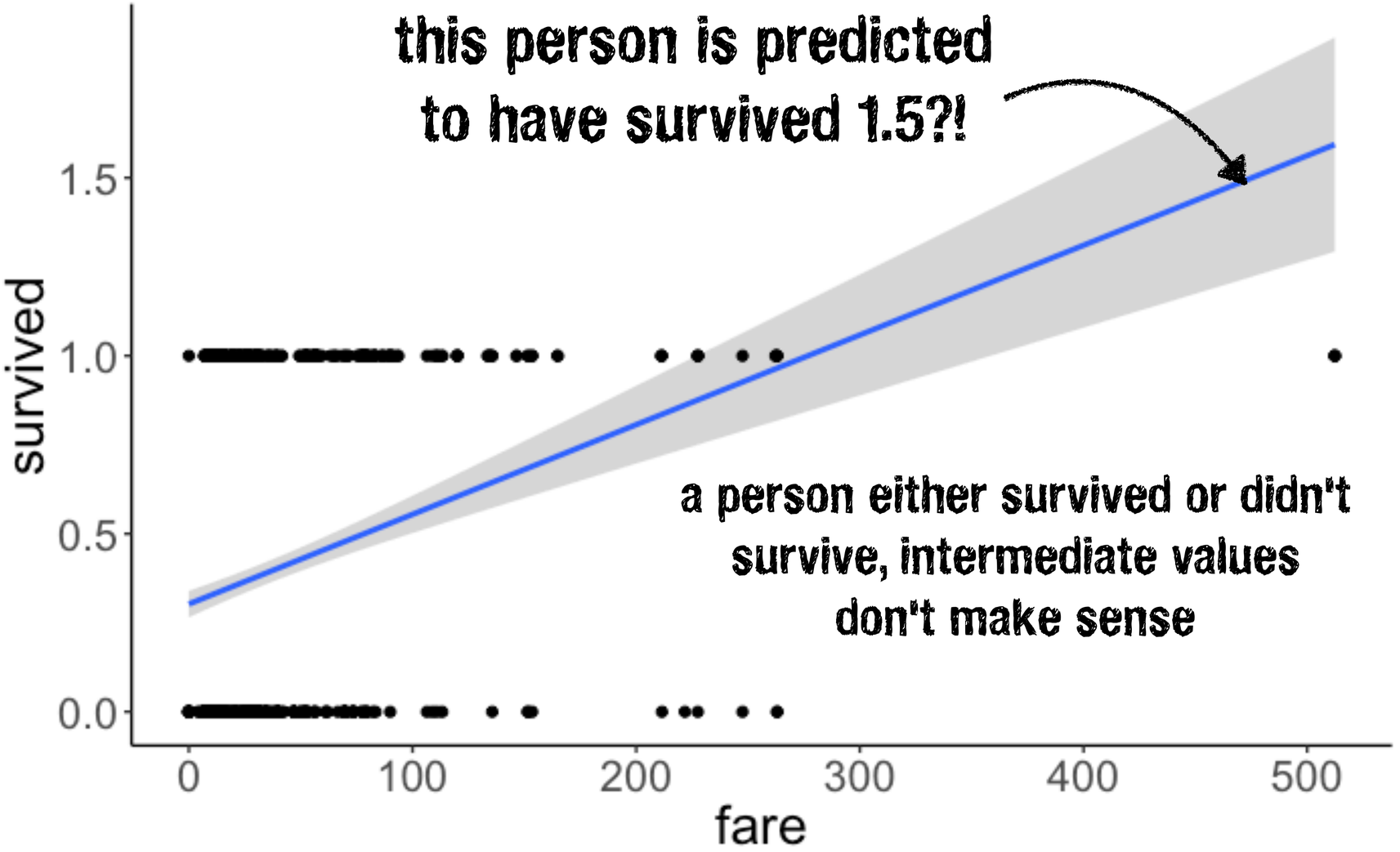
Is there a relationship between `fare` and `survived`?

Can we still use a linear model to make predictions about a binary outcome variable?

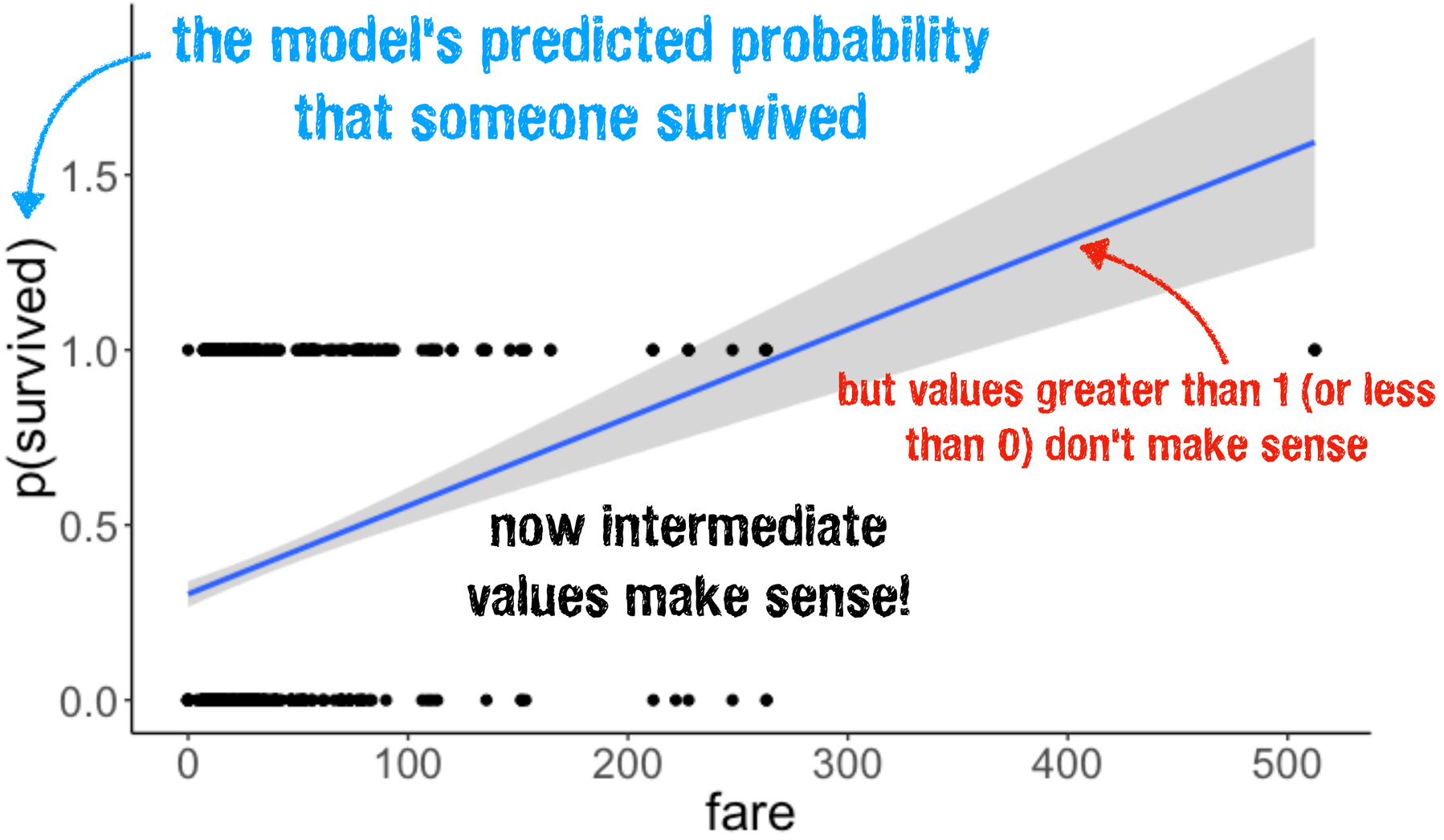
The fact that this class is called "**Generalized linear model**" suggests we can!

Is there a relationship between `fare` and `survived`?

```
fit.lm = lm(formula = survived ~ 1 + fare, data = df.titanic)
```



Is there a relationship between `fare` and `survived`?



From linear regression to logistic regression

$$Y_i = b_0 + b_1 \cdot X_i + e_i$$ predict the value of Y

$$\pi_i = b_0 + b_1 \cdot X_i + e_i$$ predict the probability of Y

$$\pi_i = P(Y_i = 1)$$

let's just do a **logit** transform

we need to map from **[-∞, +∞]** to **[0, 1]**

Logit transform

$$\pi_i = b_0 + b_1 \cdot X_i + e_i$$

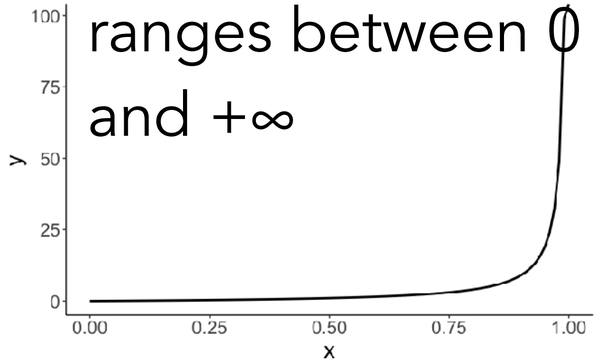
predict the probability of Y

$$\pi_i = P(Y_i = 1)$$

we need to transform the dependent variable so that it can take any value between -∞ and +∞ (we can then transform it back into a probability later)

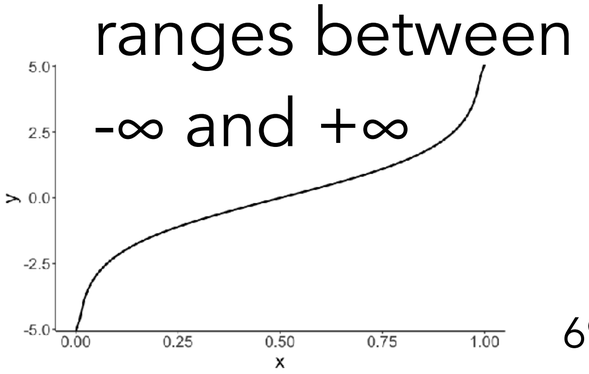
Step 1: Calculate the "odds"

$$\frac{P(Y_i = 1)}{P(Y_i = 0)} = \frac{\pi_i}{1 - \pi_i}$$



Step 2: Take the (natural) log

$$\ln\left(\frac{\pi_i}{1 - \pi_i}\right) = b_0 + b_1 \cdot X_i + e_i$$



Logit transform

after transforming from a binary variable, to a probability, to odds, to log odds, the model looks like a normal linear model

log odds

$$\ln\left(\frac{\pi_i}{1-\pi_i}\right) = b_0 + b_1 \cdot X_i + e_i$$

$$\pi_i = P(Y_i = 1)$$

if log odds == 0: $P(Y_i = 1) = P(Y_i = 0)$

if log odds > 0: $P(Y_i = 1) > P(Y_i = 0)$

if log odds < 0: $P(Y_i = 1) < P(Y_i = 0)$

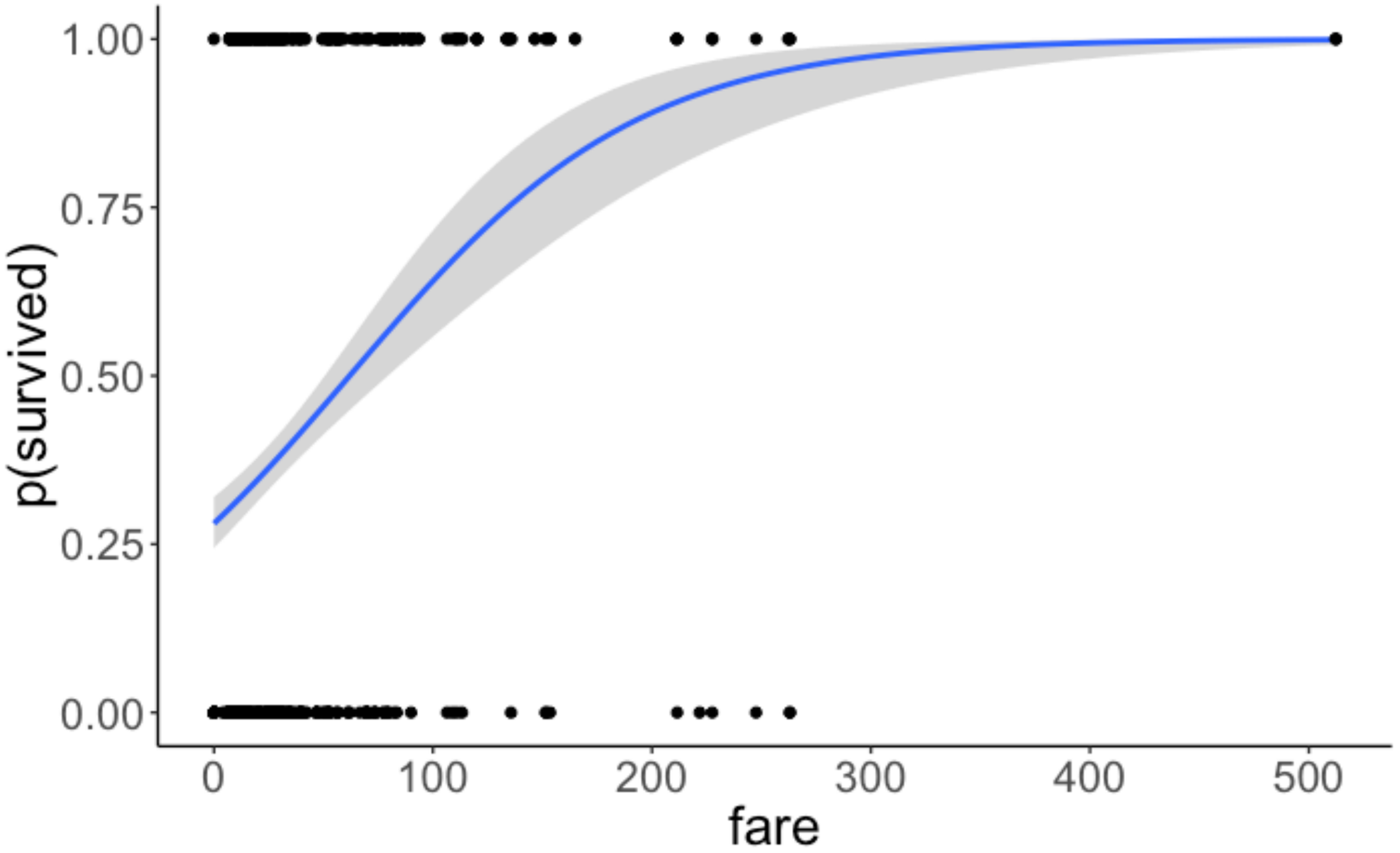
Fitting a logistic regression in R

```r
fit.glm = glm(formula = survived ~ 1 + fare,
              family = "binomial",
              data = df.titanic)

fit.glm %>% summary()
```

```
Call:
glm(formula = survived ~ 1 + fare, family = "binomial", data = df.titanic)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.4906  -0.8878  -0.8531   1.3429   1.5942

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.941330   0.095129  -9.895  < 2e-16 ***
fare         0.015197   0.002232   6.810 9.79e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1186.7  on 890  degrees of freedom
Residual deviance: 1117.6  on 889  degrees of freedom
AIC: 1121.6

Number of Fisher Scoring iterations: 4
```

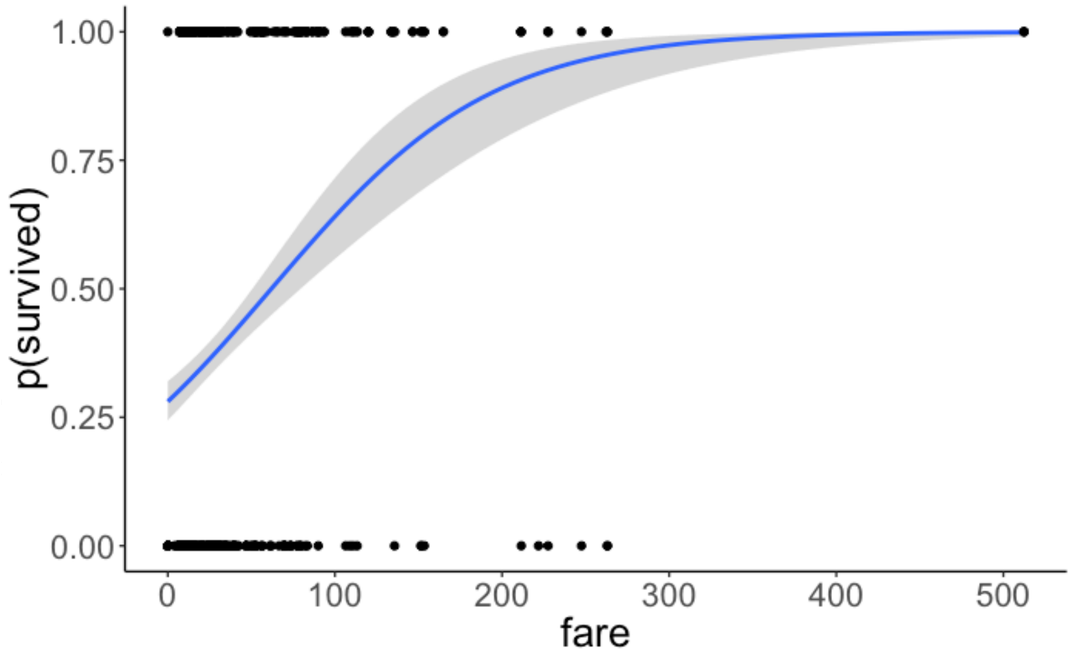
Visualize the model's predictions

```
ggplot(data = df.titanic,
       mapping = aes(x = fare,
                     y = survived)) +
  geom_smooth(method = "glm",
              method.args = list(family = "binomial")) +
  geom_point() +
  labs(y = "p(survived)")
```



Interpreting the model output

Interpreting the model output




```
Call:
glm(formula = survived ~ 1 + fare, fa

Deviance Residuals:
    Min       1Q   Median       3Q
-2.4906  -0.8878  -0.8534    1.3429

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.941330   0.095129  -9.895  < 2e-16 ***
fare         0.015197   0.002232   6.810 9.79e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1186.7  on 890  degrees of freedom
Residual deviance: 1117.6  on 889  degrees of freedom
AIC: 1121.6

Number of Fisher Scoring iterations: 4
```

log odds ?

Transform log odds into probability

$$\pi = P(Y = 1)$$

just a placeholder

$$\ln\left(\frac{\pi}{1-\pi}\right) = V$$

logit transformation

$$\pi = \frac{e^V}{1 + e^V}$$

inverse logit

gives us back the probability
(which is much easier to interpret)

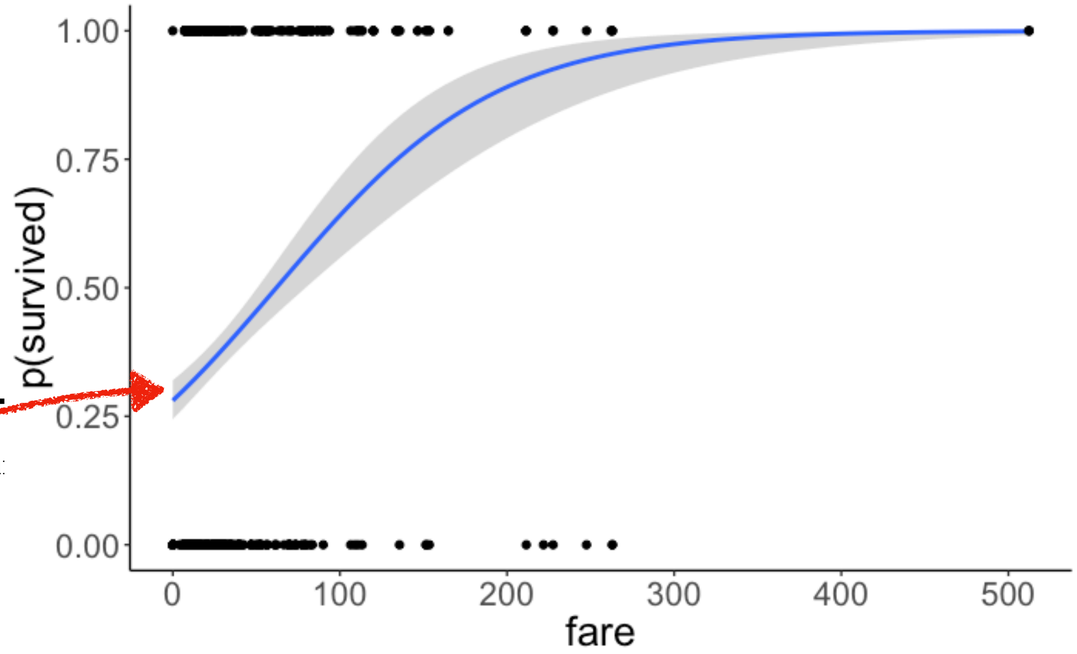
$$\pi_i = \frac{e^{b_0 + b_1 \cdot X_i}}{1 + e^{b_0 + b_1 \cdot X_i}}$$

another way to
specify the model

Interpreting the model output

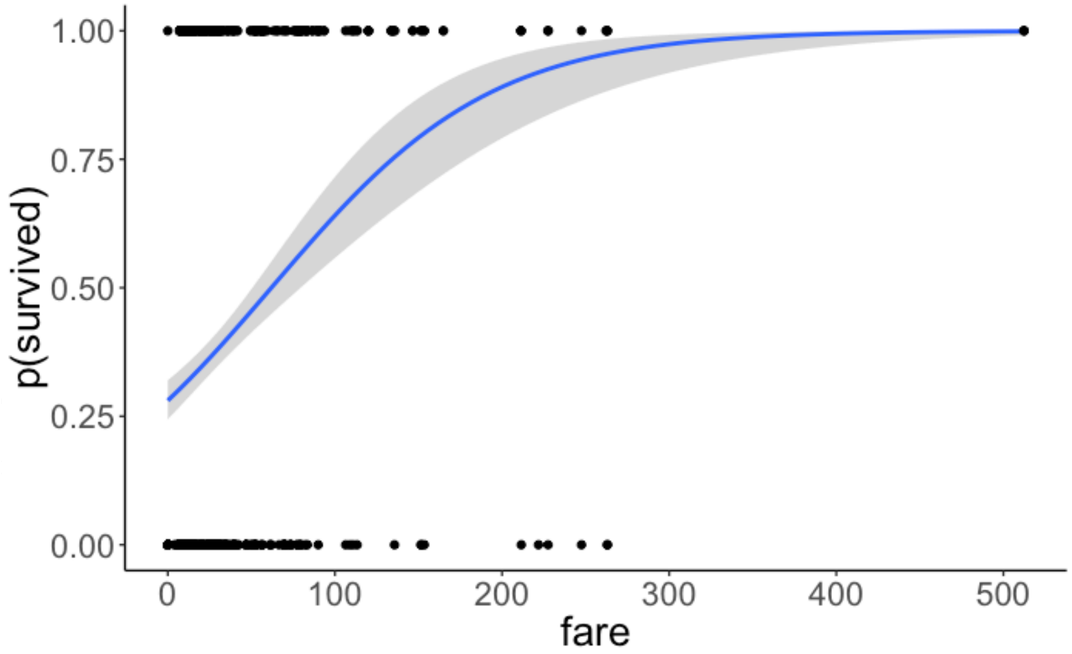
inverse logit

$$\pi = \frac{e^{-0.94}}{1 + e^{-0.94}} \approx 0.28$$




```
Call:
glm(formula = survived ~ 1 + fare, fa

Deviance Residuals:
    Min       1Q   Median       3Q
-2.4906  -0.8878  -0.8531   1.3429

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.941330   0.095129  -9.895  < 2e-16 ***
fare         0.015197   0.002232   6.810 9.79e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1186.7  on 890  degrees of freedom
Residual deviance: 1117.6  on 889  degrees of freedom
AIC: 1121.6

Number of Fisher Scoring iterations: 4
```

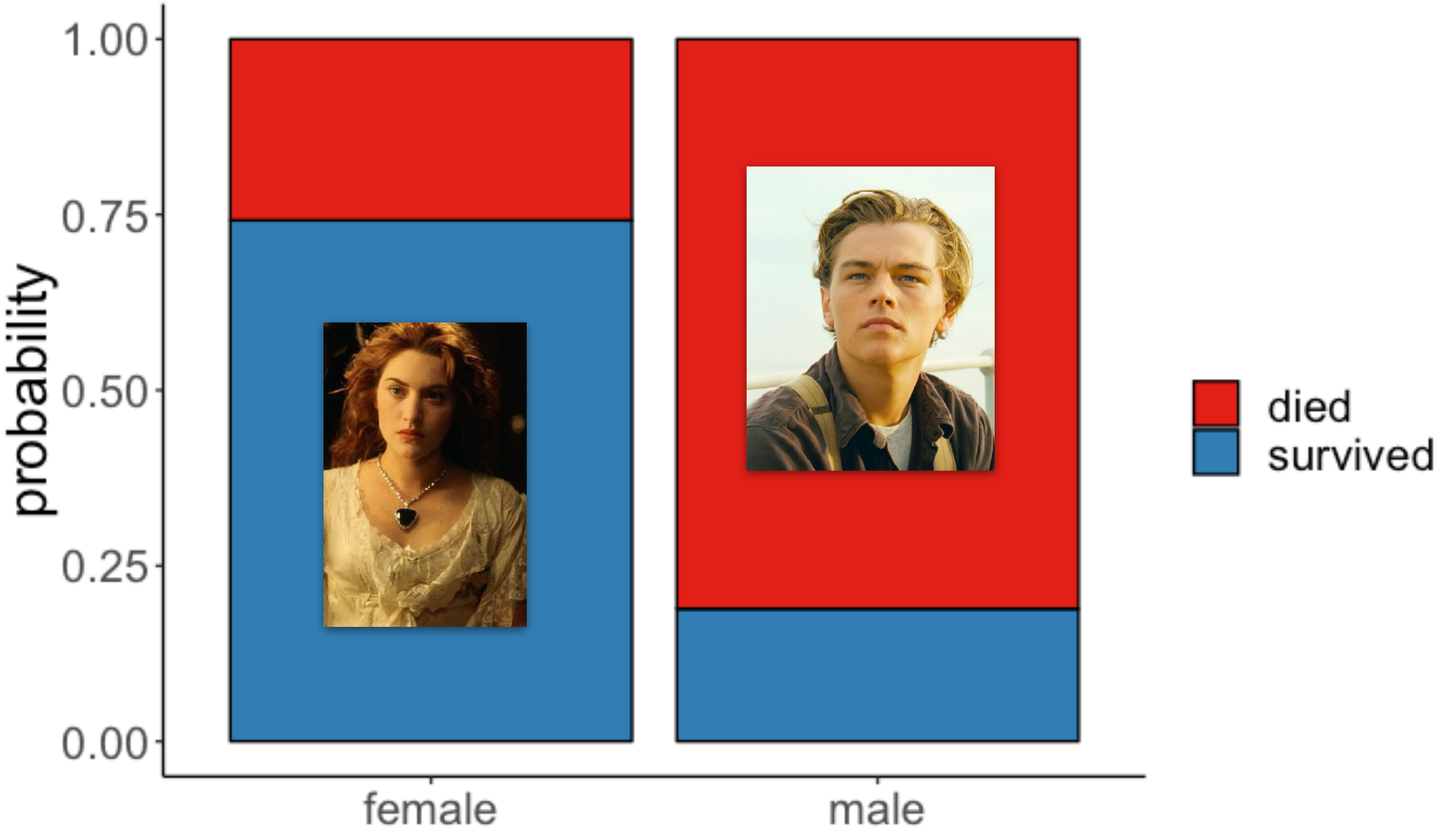
Interpreting the model output




```
Call:
glm(formula = survived ~ 1 + fare, fa

Deviance Residuals:
    Min       1Q   Median       3Q
-2.4906  -0.8878  -0.8521    1.3429

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.941330   0.095129  -9.895  < 2e-16 ***
fare         0.015197   0.002232   6.810 9.79e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1186.7  on 890  degrees of freedom
Residual deviance: 1117.6  on 889  degrees of freedom
AIC: 1121.6

Number of Fisher Scoring iterations: 4
```

Let's consider a binary predictor

Was the probability of survival different between female and male passengers on the Titanic?



Let's consider a binary predictor

```r
fit.glm2 = glm(formula = survived ~ sex,
               family = "binomial",
               data = df.titanic)

fit.glm2 %>% summary()
```

```
Call:
glm(formula = survived ~ sex, family = "binomial", data = df.titanic)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6462  -0.6471  -0.6471   0.7725   1.8256

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)    1.0566     0.1290   8.191 2.58e-16 ***
sexmale       -2.5137     0.1672 -15.036  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1186.7  on 890  degrees of freedom
Residual deviance:  917.8  on 889  degrees of freedom
AIC: 921.8

Number of Fisher Scoring iterations: 4
```

sex was significantly associated with survival

Let's consider a binary predictor

$$\ln\left(\frac{\text{p(survived)}_i}{1-\text{p(survived)}_i}\right) = b_0 + b_1 \cdot \text{sex}_i$$

```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)    1.0566     0.1290   8.191 2.58e-16 ***
sexmale       -2.5137     0.1672 -15.036  < 2e-16 ***
```

sex	survived	n	p	p(survived | sex)
female	0	81	0.09	0.26
female	1	233	0.26	0.74
male	0	468	0.53	0.81
male	1	109	0.12	0.19

if sex == 0:

$$\ln\left(\frac{\widehat{\text{p(survived)}}_i}{1-\text{p(survived)}_i}\right) = b_0$$

$$\text{p(survived)}_i = \frac{e^{b_0}}{1+e^{b_0}} = 0.74$$

Let's consider a binary predictor

$$\ln\left(\frac{\text{p(survived)}_i}{1-\text{p(survived)}_i}\right) = b_0 + b_1 \cdot \text{sex}_i$$

```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)   1.0566     0.1290   8.191 2.58e-16 ***
sexmale      -2.5137     0.1672 -15.036  < 2e-16 ***
```

sex	survived	n	p	p(survived | sex)
female	0	81	0.09	0.26
female	1	233	0.26	0.74
male	0	468	0.53	0.81
male	1	109	0.12	0.19

if sex == 1:

$$\ln\left(\frac{\widehat{\text{p(survived)}}_i}{1-\text{p(survived)}_i}\right) = b_0 + b_1$$

$$\text{p(survived)}_i = \frac{e^{b_0+b_1}}{1+e^{b_0+b_1}} = 0.19$$

Now let's go back to a continuous predictor

$$\ln\left(\frac{\text{p(survived)}_i}{1-\text{p(survived)}_i}\right) = b_0 + b_1 \cdot \text{fare}_i$$

```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.941330   0.095129  -9.895  < 2e-16 ***
fare         0.015197   0.002232   6.810 9.79e-12 ***
```

fare	prediction	p(survival)
0	-0.94	0.28
10	-0.79	0.31
50	-0.18	0.45
100	0.58	0.64
500	6.66	1.00

$$\ln\left(\frac{\widehat{\text{p(survived)}}}{1-\text{p(survived)}}\right) = -0.94 + 0.015 \cdot 10$$

$$\text{p(survived)}_i = \frac{e^{-0.94+0.015\cdot 10}}{1+e^{-0.94+0.015\cdot 10}} = 0.31$$

Now let's go back to a continuous predictor

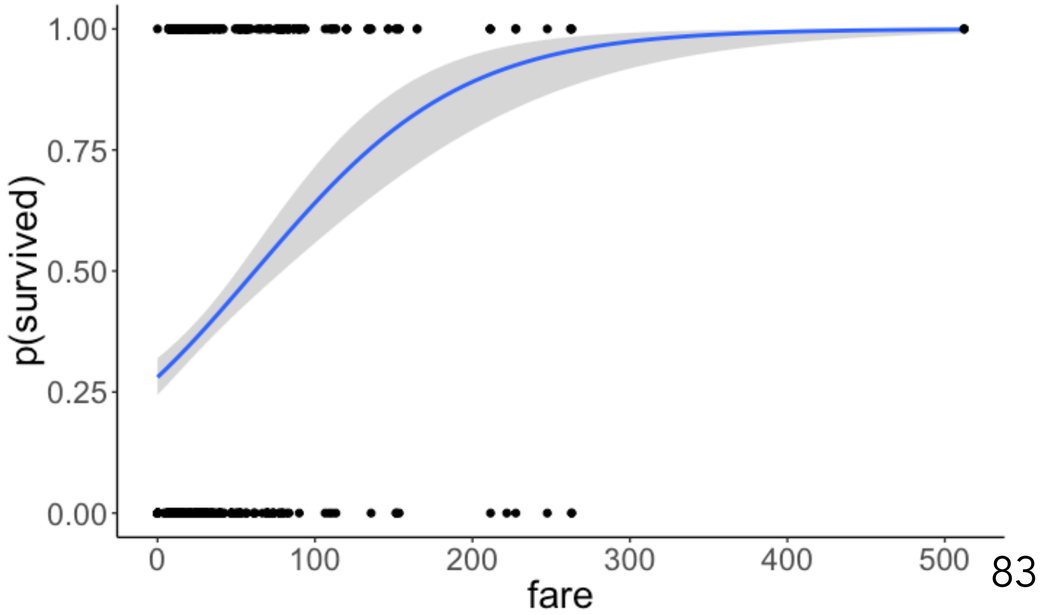
$$\ln\left(\frac{\text{p(survived)}_i}{1-\text{p(survived)}_i}\right) = b_0 + b_1 \cdot \text{fare}_i$$

```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.941330   0.095129  -9.895  < 2e-16 ***
fare         0.015197   0.002232   6.810 9.79e-12 ***
```

For a one-unit increase in the fare, the expected increase in the odds of survival is 16%.

$$e^{0.015} \approx 1.16$$

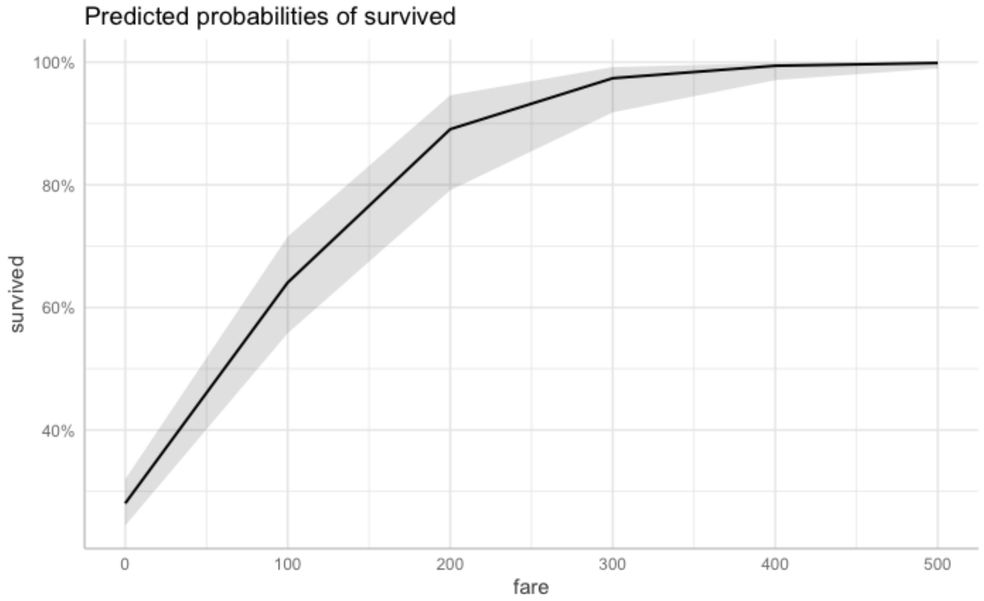
fare	prediction	p(survival)
0	-0.94	0.28
10	-0.79	0.31
50	-0.18	0.45
100	0.58	0.64
500	6.66	1.00



Do we have to do this by hand?


```
1 ggpredict(model = fit.glm,
2            terms = "fare [0, 100, 200, 300, 400, 500]")
```

```
# Predicted probabilities of survived
# x = fare

  x | Predicted |       95% CI
------------------------------
  0 |      0.28 | [0.24, 0.32]
100 |      0.64 | [0.56, 0.72]
200 |      0.89 | [0.79, 0.95]
300 |      0.97 | [0.92, 0.99]
400 |      0.99 | [0.97, 1.00]
500 |      1.00 | [0.99, 1.00]
```



Models with several predictors

$$\ln\left(\frac{\text{p(survived)}_i}{1-\text{p(survived)}_i}\right) = b_0 + b_1 \cdot \text{sex}_i + b_2 \cdot \text{fare}_i$$

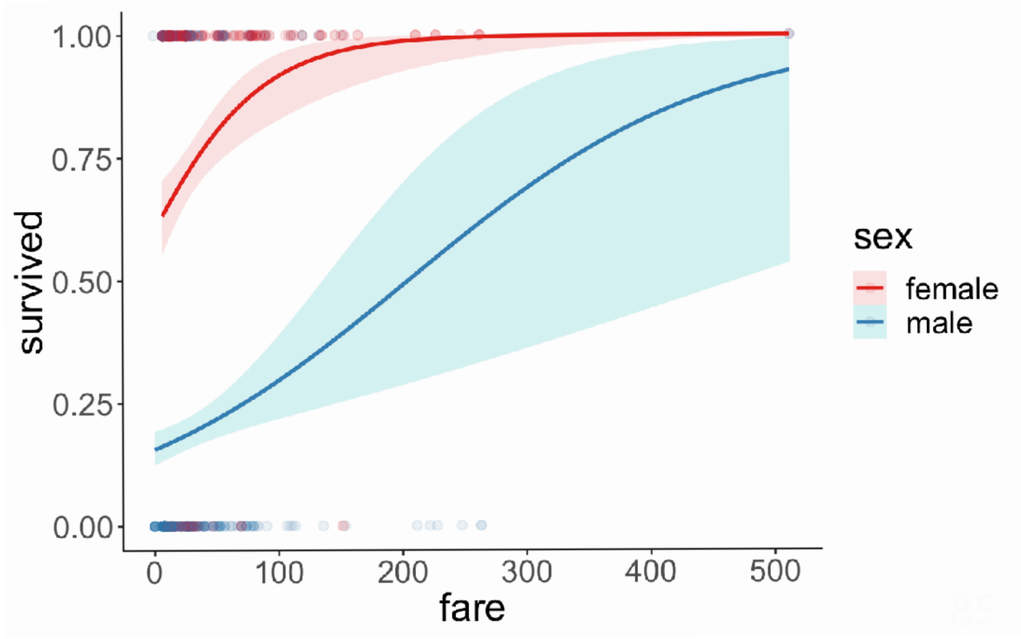
```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.647100   0.148502   4.358 1.32e-05 ***
sexmale     -2.422760   0.170515 -14.208  < 2e-16 ***
fare         0.011214   0.002295   4.886 1.03e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

controlling for "fare" there is still a significant difference between female and male passengers

```
ggpredict(fit.glm,
          terms = c("sex"))
```

```
# Predicted values of survived
# x = sex

x      | Predicted |   SE |       95% CI
---------------------------------------
female |      0.73 | 0.13 | [0.68, 0.78]
male   |      0.20 | 0.11 | [0.16, 0.23]

Adjusted for:
* fare = 32.20
```

```
df.titanic %>%
  mutate(sex = as.factor(sex)) %>%
  ggplot(data = .,
         mapping = aes(x = fare,
                       y = survived,
                       color = sex)) +
  geom_point(alpha = 0.1, size = 2) +
  geom_smooth(method = "glm",
              method.args = list(family = "binomial"),
              alpha = 0.2,
              aes(fill = sex)) +
  scale_color_brewer(palette = "Set1")
```



Fitting and reporting models

Simulating a logistic regression

```
# make example reproducible
set.seed(1)

# set parameters
sample_size = 1000
b0 = 0
b1 = 1

# generate data
df.data = tibble(
  x = rnorm(n = sample_size),
  y = b0 + b1 * x,
  p = inv.logit(y)) %>%
  mutate(response = rbinom(n(), size = 1, p = p))

# fit model
fit = glm(formula = response ~ 1 + x,
          family = "binomial",
          data = df.data)

# model summary
fit %>% summary()
```

set some parameters

linear model (y is in log odds)

transform into probability

randomly draw response

fit a logistic regression

summarize the result

Simulating a logistic regression

```r
# make example reproducible
set.seed(1)

# set parameters
sample_size = 1000
b0 = 0
b1 = 1

# generate data
df.data = tibble(
  x = rnorm(n = sample_size),
  y = b0 + b1 * x,
  p = inv.logit(y)) %>%
  mutate(response = rbinom(n(), size = 1, p = p))

# fit model
fit = glm(formula = response ~ 1 + x,
          family = "binomial",
          data = df.data)

# model summary
fit %>% summary()
```

```
Call:
glm(formula = response ~ 1 + x, family = "binomial", data = df.data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1137  -1.0118  -0.4591   1.0287   2.2591

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.06214    0.06918  -0.898    0.369
x            0.92905    0.07937  11.705   <2e-16 ***

Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1385.4  on 999  degrees of freedom
Residual deviance: 1209.6  on 998  degrees of freedom
AIC: 1213.6

Number of Fisher Scoring iterations: 3
```

Assessing the model fit

actual value

predicted value

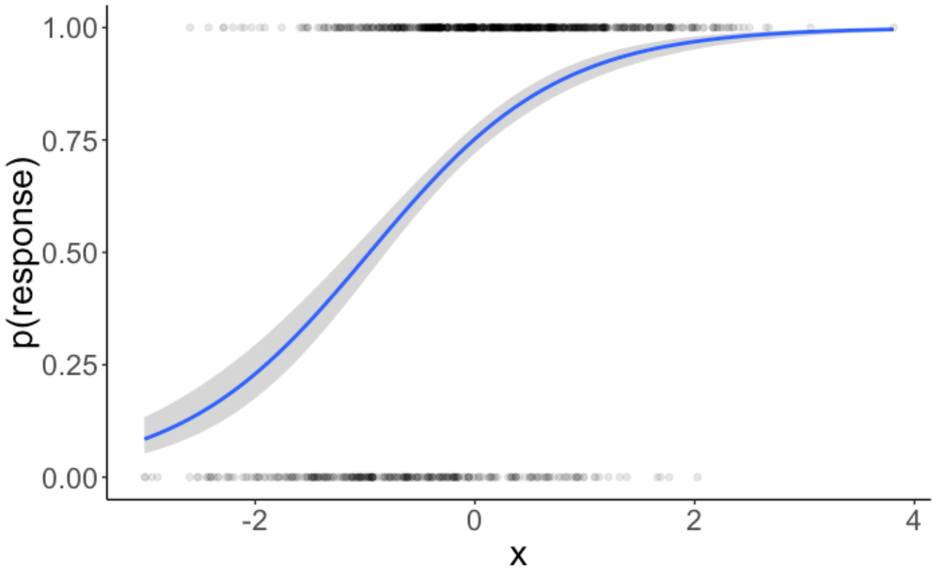
$$\text{log-likelihood} = \sum_{i=1}^{n} [Y_i \cdot \ln(P(Y_i)) + (1 - Y_i) \cdot \ln(1 - P(Y_i))]$$

- calculate the probability of the observed response
- take the log of these probabilities
- sum them up to get the log-likelihood of the data (given the model)

response	p(Y = 1)	p(Y = response)	log(p(Y = response))
1	0.34	0.34	-1.07
0	0.53	0.47	-0.75
1	0.30	0.30	-1.20
1	0.81	0.81	-0.22
1	0.56	0.56	-0.58
0	0.30	0.70	-0.36
1	0.60	0.60	-0.52
1	0.65	0.65	-0.43
1	0.62	0.62	-0.48
0	0.41	0.59	-0.54

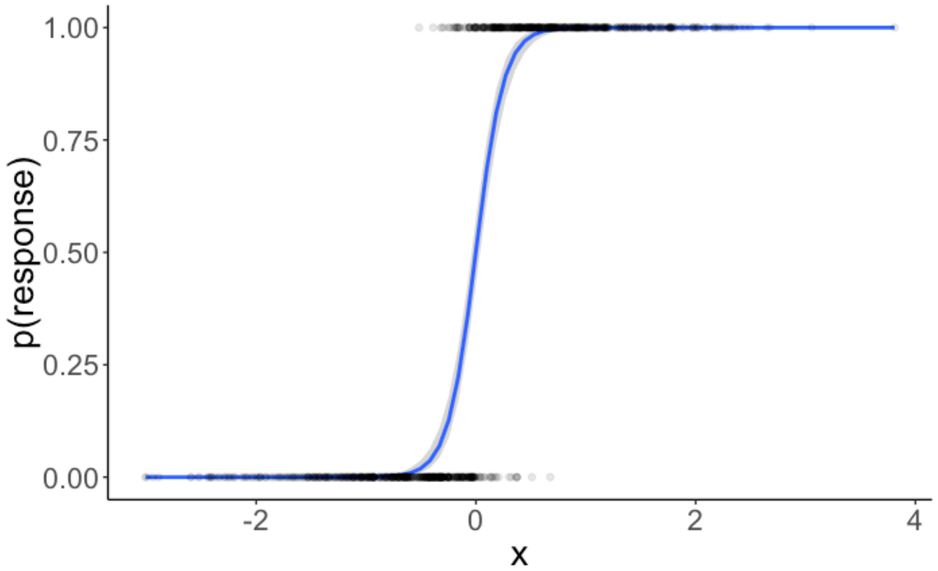
Assessing the model fit

doesn't predict the response very well



logLik	AIC	BIC
-501.65	1007.3	1017.12

predicts the response much better



logLik	AIC	BIC
-156.37	316.74	326.55

Testing hypotheses

aka checking whether it's **worth it**

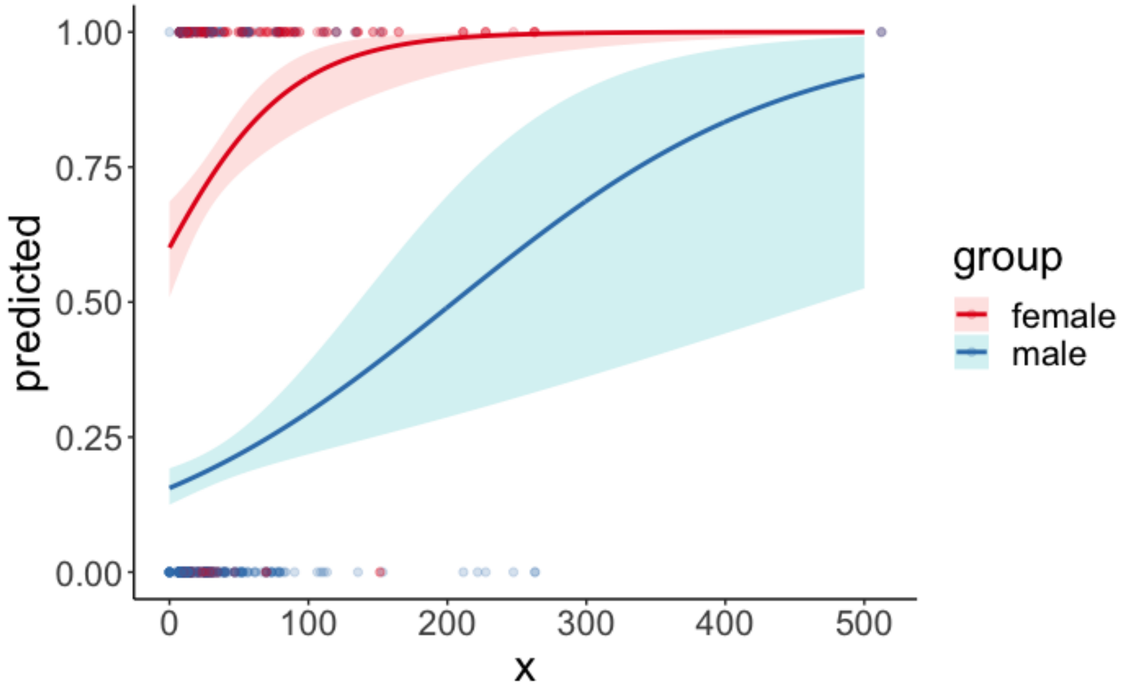
```r
# fit compact model
fit.compact = glm(formula = survived ~ 1 + fare,
                  family = "binomial",
                  data = df.titanic)

# fit augmented model
fit.augmented = glm(formula = survived ~ 1 + sex + fare,
                    family = "binomial",
                    data = df.titanic)

# likelihood ratio test
anova(fit.compact, fit.augmented, test = "LRT")
```

we need to specify that we want a likelihood ratio test

```
Analysis of Deviance Table

Model 1: survived ~ 1 + fare
Model 2: survived ~ 1 + sex + fare
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1       889    1117.57
2       888     884.31  1   233.26 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Reporting results

- Visualize the data
- Show a table with the regression results
- Report significance of different factors
- Interpreting parameter estimates is tricky -- probably best to report probabilities for a few example cases




```
# Predicted values of survived
# x = fare

# sex = female

  x | Predicted |   SE |       95% CI
-------------------------------------
  0 |      0.60 | 0.19 | [0.51, 0.69]
100 |      0.92 | 0.42 | [0.83, 0.96]
200 |      0.99 | 0.95 | [0.93, 1.00]
300 |      1.00 | 1.48 | [0.97, 1.00]
400 |      1.00 | 2.02 | [0.99, 1.00]
500 |      1.00 | 2.55 | [1.00, 1.00]

# sex = male

  x | Predicted |   SE |       95% CI
-------------------------------------
  0 |      0.16 | 0.13 | [0.12, 0.19]
100 |      0.30 | 0.21 | [0.22, 0.39]
200 |      0.49 | 0.44 | [0.29, 0.70]
300 |      0.69 | 0.69 | [0.36, 0.90]
400 |      0.83 | 0.94 | [0.44, 0.97]
500 |      0.92 | 1.19 | [0.53, 0.99]
```

Assumptions

- linearity (between predictors and log odds)
- independence
- no multi-collinearity

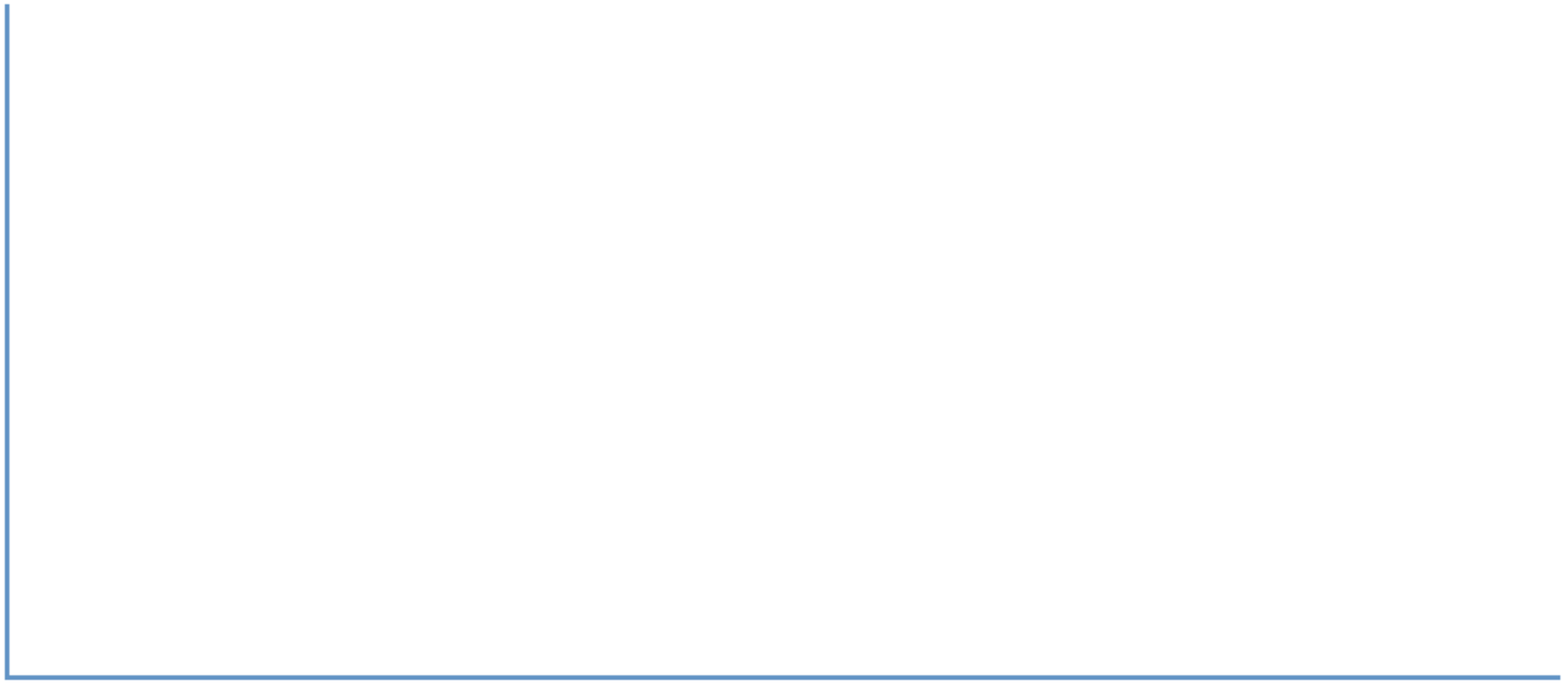
- model fails to converge when there is **complete separation**:
  - if outcome variable can be perfectly predicted by a (combination of) predictor(s)

Plan for today

- Quick Recap
- Interpreting parameters
- Who is the ANOVA champ?
- Unbalanced designs
- Linear contrasts
- Generalized linear model
  - Logistic regression
    - interpreting the model output
    - fitting and reporting models

Feedback

How was the pace of today's class?



Start the presentation to see live content. Still no live content? Install the app or get help at **PollEv.com/app**

How happy were you with today's class overall?

Answers to this poll are anonymous

Start the presentation to see live content. Still no live content? Install the app or get help at **PollEv.com/app**

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

Answers to this poll are anonymous

Start the presentation to see live content. Still no live content? Install the app or get help at **PollEv.com/app**

Thank you!