

## Part 3


Ordinal logistic regression

# Ordinal outcome

---

Use when the outcome is categorical, there are 3 or more levels, and there is an ordering to the levels

For instance:

- Mild, moderate or severe disease
  - Below, at or above expected performance
  - Strongly disagree, disagree, neither agree nor disagree, agree, strongly agree
- 

# Ordinal logistic regression

---

An extension of binary logistic regression used when the outcome is ordinal

We will focus on **the proportional odds model**

# Assumptions

---

1. Independence of errors
2. Linearity of the logit (to be checked for **every** continuous predictor)
3. No multicollinearity: Predictor variables should not be highly correlated (only an assumption for multiple ordinal logistic regression)
4. The proportional odds assumption....

# The proportional odds model

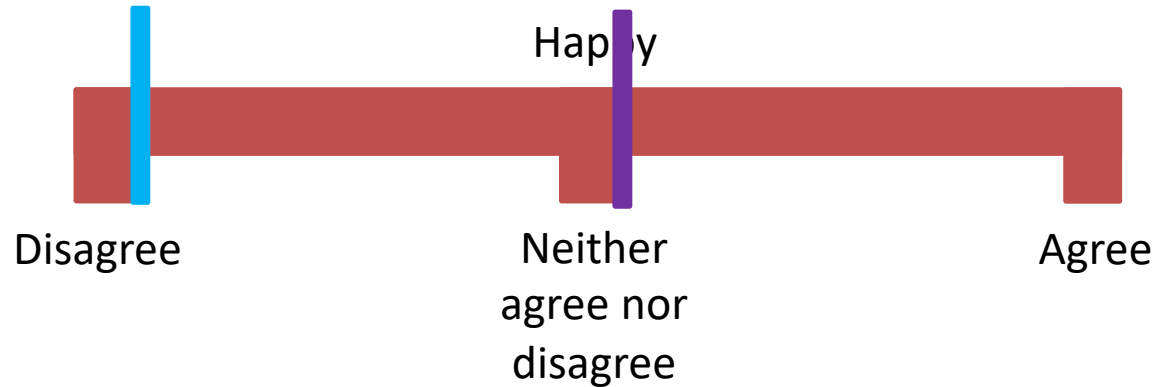
---

- When we use a proportional odds model, we make a key assumption about the data:
- The predictor variable has the identical effect at each cumulative split
- As proportional odds models make this assumption, we only get one odds ratio for each continuous predictor/one odds ratio for each comparison of a categorical variable (e.g. marital\_statusSingle – marital\_statusCohabiting)

# The proportional odds assumption

HamsterYes = 1.88

Does hamster ownership (yes/no) predict happiness (agree, neither agree nor disagree, disagree)?



Disagree vs Neither agree nor disagree or agree: OR = 1.88

Disagree or Neither agree nor disagree vs agree: OR = 1.88

# Example

## What factors predict happiness?

Does hamster ownership, marital status, and number of hours free time an individual has predict response to the following survey question:

I am happy:

- Agree
- Neither agree nor disagree
- Disagree



- Predictors: Hamster ownership (yes/no), marital status (single, cohabiting, married, divorced), and hours free time (continuous)
- Outcome: Happy (Agree/Neither agree nor disagree/Disagree)

# 1. Prepare dataset:

## Variable types

---

- Outcome: ordered factor
- Predictors:
  - Categorical: factors with first level as the reference category:
    - Hamster ownership = no
    - Marital status = single
  - Continuous: numeric/integer variable

Check using the “str”  
function and adjust  
variables as required



# 1. Prepare dataset: Changing variable types

---

- Outcome: needs to be an ordered factor

```
happiness_order$happiness <- ordered(happiness_order$happiness, levels = c("Disagree", "Neither agree nor disagree", "Agree"))
```

# 1. Prepare dataset: Check the structure

```
> str(happiness_order)
'data.frame': 53 obs. of 5 variables:
 $ Participant_ID : int 1 2 3 4 5 6 7 8 9 10 ...
 $ Hours_free_time: int 19 6 8 6 3 2 18 19 12 10 ...
 $ marital_status : Factor w/ 4 levels "Single","Cohabiting",...: 1 1 1 1 1 1 1 1 1 1 ...
 $ Hamster        : Factor w/ 2 levels "No","Yes": 2 1 2 1 2 1 2 1 2 1 ...
 $ Happy          : Ord.factor w/ 3 levels "Disagree"<"Neither agree nor disagree"<...: 3 1
```

- Hours\_free\_time is an integer
- Marital\_status is a factor, with single as the first factor level
- Hamster is a factor, with "No" as the first factor level
- Happy is an ordered factor

## 2. Explore the data and check for separation

### Categorical variables: use 'table'

Hamster ownership

```
> table(happiness_order$Hamster, happiness_order$Happy)
```

|     | Disagree | Neither agree nor disagree | Agree |
|-----|----------|----------------------------|-------|
| No  | 12       | 4                          | 8     |
| Yes | 8        | 9                          | 12    |

Marital status

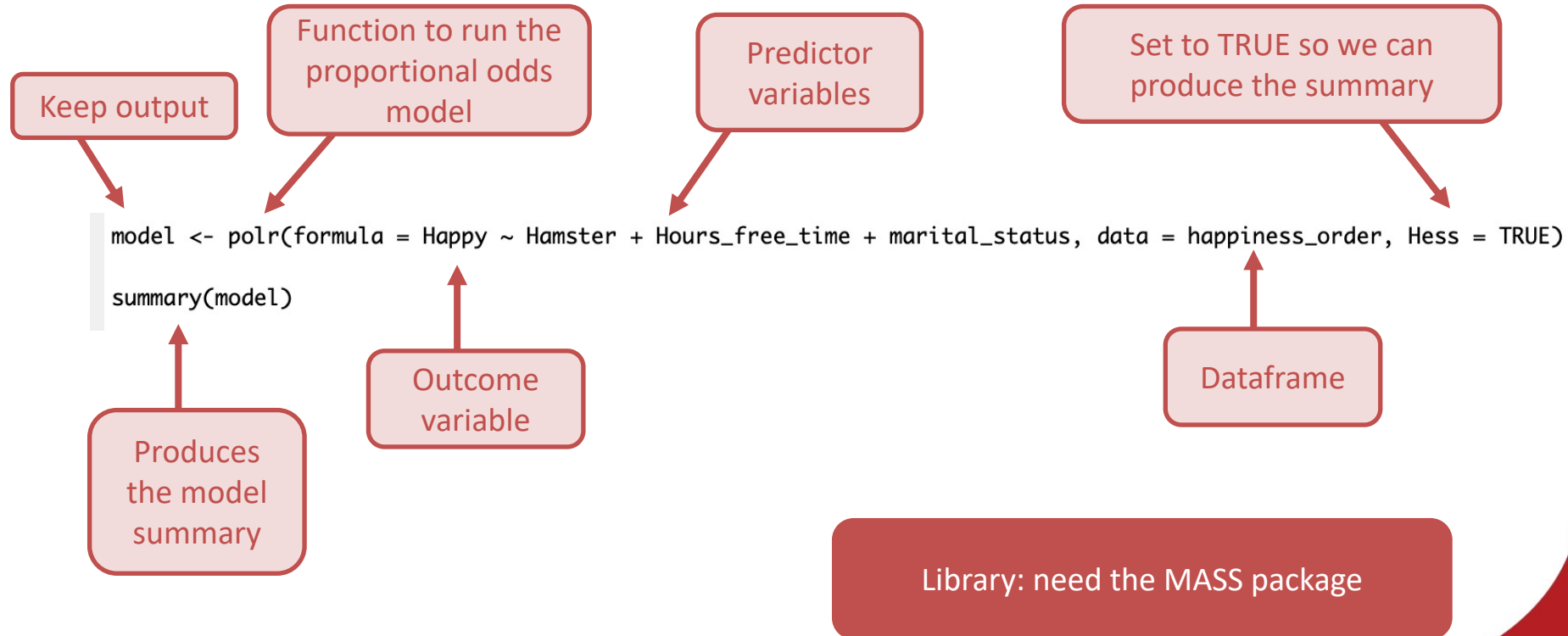
```
> table(happiness_order$marital_status, happiness_order$Happy)
```

|            | Disagree | Neither agree nor disagree | Agree |
|------------|----------|----------------------------|-------|
| Single     | 6        | 6                          | 3     |
| Cohabiting | 3        | 2                          | 3     |
| Married    | 3        | 2                          | 10    |
| Divorced   | 8        | 3                          | 4     |

No evidence of complete separation or quasi-complete separation for either variable

### 3. Running the model

#### Code to run the model



# 3. Running the model

## Model output

```
> summary(model)
Call:
polr(formula = Happy ~ Hamster + Hours_free_time + marital_status,
      data = happiness_order, Hess = TRUE)
```

Coefficients:

|                          | Value  | Std. Error | t value |
|--------------------------|--------|------------|---------|
| HamsterYes               | 0.5722 | 0.56493    | 1.0129  |
| Hours_free_time          | 0.1463 | 0.05349    | 2.7351  |
| marital_statusCohabiting | 0.2783 | 0.83629    | 0.3328  |
| marital_statusMarried    | 1.7319 | 0.77863    | 2.2243  |
| marital_statusDivorced   | 0.2895 | 0.73277    | 0.3951  |

Intercepts:

|                                     | Value  | Std. Error | t value |
|-------------------------------------|--------|------------|---------|
| Disagree Neither agree nor disagree | 1.7566 | 0.8260     | 2.1267  |
| Neither agree nor disagree Agree    | 3.0782 | 0.8992     | 3.4234  |

Residual Deviance: 96.99787

AIC: 110.9979

No warning messages – no evidence of complete separation or quasi-complete separation

## 4. Evaluating the model

### Comparing to the intercept-only model

- In binary logistic regression, the intercept-only model was calculated automatically alongside the specified model, allowing us to use output from the model to evaluate the model

Binary logistic  
regression:

```
Null deviance: 70.252 on 52 degrees of freedom  
Residual deviance: 63.475 on 51 degrees of freedom  
AIC: 67.475
```

```
Number of Fisher Scoring iterations: 4
```

Null deviance =  
deviance for the  
intercept-only  
model

Residual deviance  
= deviance for  
specified model

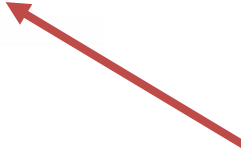
## 4. Evaluating the model

### Comparing to the intercept-only model

---

- When running a proportional odds model, we only get the residual deviance (deviance for specified model)
- Proportional odds model:

```
Residual Deviance: 96.99787  
AIC: 110.9979
```



Residual deviance  
= deviance for  
specified model

## 4. Evaluating the model

### Comparing to the intercept-only model

---

- We therefore need to create a second model including only the intercept

```
intercept_model <- polr(formula = Happy ~ 1, data = happiness_order, Hess = TRUE)
```

- We can then use the 'anova' function to compare the specified model to the intercept-only model

```
anova(model, intercept_model)
```



## 4. Evaluating the model

### Comparing to the intercept-only model

Which models  
are being  
compared?

Likelihood ratio tests of ordinal regression models

Response: Happy

|   | Model                                      | Resid. | df | Resid. Dev | Test   | Df | LR stat. | Pr(Chi)     |
|---|--|--------|----|------------|--------|----|----------|-------------|
| 1 | 1  | 51     |    | 114.50368  |        |    |          |             |
| 2 | Hamster + Hours_free_time + marital_status | 46     |    | 96.99787   | 1 vs 2 | 5  | 17.5058  | 0.003634013 |

Chi square = LR stat  
Df = Df  
p = Pr(Chi)

- $X^2(5) = 17.51, p = .004$

## 4. Pseudo R<sup>2</sup>s

```
PseudoR2(model, which = "all")
```

| McFadden    | CoxSnell    | Nagelkerke  | AldrichNelson | VeallZimmermann | Efron | McKelveyZavoina | Tjur | AIC         |
|-------------|-------------|-------------|---------------|-----------------|-------|-----------------|------|-------------|
| 0.1528842   | 0.2812906   | 0.3179408   | NA            | NA              | NA    | NA              | NA   | 110.9978746 |
| BIC         | logLik      | logLik      | G2            |                 |       |                 |      |             |
| 124.7899180 | -48.4989373 | -57.2518388 | 17.5058031    |                 |       |                 |      |             |

- McFadden = 0.15
- CoxSnell = 0.28
- Nagelkerke = 0.32

# 5. Interpreting the individual predictors

## The intercepts

---

Intercepts:

|                                     | Value  | Std. Error | t value |
|-------------------------------------|--------|------------|---------|
| Disagree Neither agree nor disagree | 1.7566 | 0.8260     | 2.1267  |
| Neither agree nor disagree Agree    | 3.0782 | 0.8992     | 3.4234  |

- Two intercepts?!
- When running a proportional odds model, we have outcome\_levels – 1 intercepts
- Here, three possible outcomes (disagree, neither agree nor disagree, and agree) so two intercepts

# 5. Interpreting the individual predictors

## The intercepts

- What do the intercepts mean?

| denotes where the cumulative split is:

- Disagree | Neither agree nor disagree =  
Disagree vs Neither agree nor disagree  
OR agree
- Neither agree nor disagree | Agree =  
Disagree or Neither agree nor disagree  
vs agree

Intercepts:

|                                     | Value  | Std. Error | t value |
|-------------------------------------|--------|------------|---------|
| Disagree Neither agree nor disagree | 1.7566 | 0.8260     | 2.1267  |
| Neither agree nor disagree Agree    | 3.0782 | 0.8992     | 3.4234  |

# 5. Interpreting the individual predictors

## The intercepts

- What do the intercepts mean?

The intercept displays the log odds of having the category (or categories) before | when:

- Each categorical variables = reference category
- Each continuous variables = 0

Intercepts:

|                                     | Value  | Std. Error | t value |
|-------------------------------------|--------|------------|---------|
| Disagree Neither agree nor disagree | 1.7566 | 0.8260     | 2.1267  |
| Neither agree nor disagree Agree    | 3.0782 | 0.8992     | 3.4234  |

# 5. Interpreting the individual predictors

## The intercepts

- What do the intercepts mean?

The log odds that happiness = disagree when:

- Hamster = No
- Marital status = Single
- Hours\_free\_time = 0

The log odds that happiness = disagree OR neither agree nor disagree when:

- Hamster = No
- Marital status = Single
- Hours\_free\_time = 0

Intercepts:

|                                     | Value  | Std. Error | t value |
|-------------------------------------|--------|------------|---------|
| Disagree Neither agree nor disagree | 1.7566 | 0.8260     | 2.1267  |
| Neither agree nor disagree Agree    | 3.0782 | 0.8992     | 3.4234  |

## 5. Interpreting the individual predictors

### The predictors

---

- One Estimate for each predictor – explains each cumulative split.
- For instance, imagine we get an odds ratio of 3.55 for HamsterYes:
  - Individuals who have a hamster have  $\sim 3.55x$  higher odds of responding neither agree nor disagree or agree (as opposed to 'disagree') relative to individuals who do not have a hamster
  - Individuals who have a hamster have  $\sim 3.55x$  higher odds of responding agree (as opposed to 'disagree' or 'neither agree nor disagree') relative to individuals who do not have a hamster

## 5. Interpreting the individual predictors

### The predictors

---

- We can summarise this by saying:
- Individuals who have a hamster have 3.55x higher odds of being more happy (e.g. agree vs neither agree nor disagree or disagree) relatively to individuals who do not have a hamster



## 5. Interpreting the individual predictors

### The predictors: Hamster

```
> summary(model)
```

Call:

```
polr(formula = Happy ~ Hamster + Hours_free_time + marital_status,  
      data = happiness_order, Hess = TRUE)
```

Coefficients:

|                          | Value  | Std. Error | t value |
|--------------------------|--------|------------|---------|
| HamsterYes               | 0.5722 | 0.56493    | 1.0129  |
| Hours_free_time          | 0.1463 | 0.05349    | 2.7351  |
| marital_statusCohabiting | 0.2783 | 0.83629    | 0.3328  |
| marital_statusMarried    | 1.7319 | 0.77863    | 2.2243  |
| marital_statusDivorced   | 0.2895 | 0.73277    | 0.3951  |

Intercepts:

|                                     | Value  | Std. Error | t value |
|-------------------------------------|--------|------------|---------|
| Disagree Neither agree nor disagree | 1.7566 | 0.8260     | 2.1267  |
| Neither agree nor disagree Agree    | 3.0782 | 0.8992     | 3.4234  |

Residual Deviance: 96.99787

AIC: 110.9979

.

The change in the log odds of being more happy when moving from HamsterNo to HamsterYes, when holding the other variables constant

# 5. Interpreting the individual predictors

## The predictors: Hours\_free\_time

```
> summary(model)
```

Call:

```
polr(formula = Happy ~ Hamster + Hours_free_time + marital_status,  
      data = happiness_order, Hess = TRUE)
```

Coefficients:

|                          | Value  | Std. Error | t value |
|--------------------------|--------|------------|---------|
| HamsterYes               | 0.5722 | 0.56493    | 1.0129  |
| Hours_free_time          | 0.1463 | 0.05349    | 2.7351  |
| marital_statusCohabiting | 0.2783 | 0.83629    | 0.3328  |
| marital_statusMarried    | 1.7319 | 0.77863    | 2.2243  |
| marital_statusDivorced   | 0.2895 | 0.73277    | 0.3951  |

Intercepts:

|                                     | Value  | Std. Error | t value |
|-------------------------------------|--------|------------|---------|
| Disagree Neither agree nor disagree | 1.7566 | 0.8260     | 2.1267  |
| Neither agree nor disagree Agree    | 3.0782 | 0.8992     | 3.4234  |

Residual Deviance: 96.99787

AIC: 110.9979

The change in the log odds of being more happy with a one unit increase in Hours\_free\_time when holding the other variables constant

## 5. Interpreting the individual predictors

### The predictors: marital\_statusCohabiting

```
> summary(model)
```

Call:

```
polr(formula = Happy ~ Hamster + Hours_free_time + marital_status,
      data = happiness_order, Hess = TRUE)
```

Coefficients:

|                          | Value  | Std. Error | t value |
|--------------------------|--------|------------|---------|
| HamsterYes               | 0.5722 | 0.56493    | 1.0129  |
| Hours_free_time          | 0.1463 | 0.05349    | 2.7351  |
| marital_statusCohabiting | 0.2783 | 0.83629    | 0.3328  |
| marital_statusMarried    | 1.7319 | 0.77863    | 2.2243  |
| marital_statusDivorced   | 0.2895 | 0.73277    | 0.3951  |

Intercepts:

|                                     | Value  | Std. Error | t value |
|-------------------------------------|--------|------------|---------|
| Disagree Neither agree nor disagree | 1.7566 | 0.8260     | 2.1267  |
| Neither agree nor disagree Agree    | 3.0782 | 0.8992     | 3.4234  |

Residual Deviance: 96.99787

AIC: 110.9979

The change in the log odds of being more happy when moving from marital\_statusSingle to marital\_statusCohabiting, when holding the other variables constant

Repeat for the rest...

## 5. Evaluating individual predictors

### Exponentiated values

---

```
odds_ratio <- exp(model$coefficients)
odds_ratio
```

## 5. Evaluating individual predictors

### Exponentiated values

> odds\_ratio

|            |                 |                          |                       |                        |
|------------|-----------------|--------------------------|-----------------------|------------------------|
| HamsterYes | Hours_free_time | marital_statusCohabiting | marital_statusMarried | marital_statusDivorced |
| 1.772220   | 1.157526        | 1.320918                 | 5.651540              | 1.335801               |

> |

- HamsterYes: Odds ratio: the change in odds of being more happy (e.g. “agree” vs “neither agree nor disagree” or “disagree”), when holding the other variables constant
- Hours\_free\_time: Odds ratio: the change in odds of being more happy (e.g. “agree” vs “neither agree nor disagree” or “disagree”) with a one unit change in hours\_free\_time, when holding the other variables constant
- marital\_statusCohabiting: Odds ratio: the change in odds of being more happy (e.g. “agree” vs “neither agree nor disagree” or “disagree”), when holding the other variables constant

## 5. Odds ratio confidence intervals

```
exp(confint(model))
```

```
> exp(confint(model))
Waiting for profiling to be done...
              2.5 %      97.5 %
HamsterYes      0.5886499  5.484066
Hours_free_time  1.0479302  1.294931
marital_statusCohabiting 0.2502179  6.976327
marital_statusMarried  1.2955234 28.406087
marital_statusDivorced 0.3168994  5.775195
```

## 5. P-values

```
> summary(model)
```

Call:

```
polr(formula = Happy ~ Hamster + Hours_free_time + marital_status,  
      data = happiness_order, Hess = TRUE)
```

Coefficients:

|                          | Value  | Std. Error | t value |
|--------------------------|--------|------------|---------|
| HamsterYes               | 0.5722 | 0.56493    | 1.0129  |
| Hours_free_time          | 0.1463 | 0.05349    | 2.7351  |
| marital_statusCohabiting | 0.2783 | 0.83629    | 0.3328  |
| marital_statusMarried    | 1.7319 | 0.77863    | 2.2243  |
| marital_statusDivorced   | 0.2895 | 0.73277    | 0.3951  |

Intercepts:

|                                     | Value  | Std. Error | t value |
|-------------------------------------|--------|------------|---------|
| Disagree Neither agree nor disagree | 1.7566 | 0.8260     | 2.1267  |
| Neither agree nor disagree Agree    | 3.0782 | 0.8992     | 3.4234  |

Residual Deviance: 96.99787

AIC: 110.9979

- Wait... where are the p-values for the individual predictors?!
- R does not output them, but these can be calculated

## 5. P-values

---

```
coefficients <- summary(model)$coefficients  
p_value <- (1 - pnorm(abs(coefficients[, "t value"]), 0, 1))*2  
coefficients_with_p <- cbind(coefficients, p_value)  
coefficients_with_p
```



# P-values

```
> coefficients_with_p
```

|                                     | Value     | Std. Error | t value   | p_value     |
|-------------------------------------|-----------|------------|-----------|-------------|
| HamsterYes                          | 0.5722328 | 0.56493092 | 1.0129253 | 0.311095844 |
| Hours_free_time                     | 0.1462847 | 0.05348519 | 2.7350501 | 0.006237078 |
| marital_statusCohabiting            | 0.2783272 | 0.83629455 | 0.3328100 | 0.739277698 |
| marital_statusMarried               | 1.7319281 | 0.77863020 | 2.2243269 | 0.026126456 |
| marital_statusDivorced              | 0.2895310 | 0.73277181 | 0.3951175 | 0.692756149 |
| Disagree Neither agree nor disagree | 1.7565738 | 0.82595475 | 2.1267192 | 0.033443423 |
| Neither agree nor disagree Agree    | 3.0781902 | 0.89916033 | 3.4234053 | 0.000618418 |

- No stars this time, need to carefully examine the p-values yourself
- Hours\_per\_day is significant ( $p = .006$ )
- marital\_statusMarried is significant ( $p = .026$ )
- Ignore significance of intercepts

## 6. Predicted probabilities

---

- Predicted probabilities are a little more complex when we have 3+ levels of the outcomes variable
  - Need to know the predicted probability for each individual within each outcome category
- Can use same ‘fitted’ function, but...
- We shouldn’t make this a new variable in our existing dataframe, as it will only display the values for one of the outcome levels (e.g. disagree)

## 6. Predicted probabilities

- Instead, save this to it's own object name:

```
predicted_probabilities <- fitted(model)
```

```
> predicted_probabilities
```

|    | Disagree   | Neither agree nor disagree | Agree      |
|----|------------|----------------------------|------------|
| 1  | 0.16867207 | 0.26338788                 | 0.56794005 |
| 2  | 0.70658712 | 0.19370568                 | 0.09970720 |
| 3  | 0.50351583 | 0.28826279                 | 0.20822138 |
| 4  | 0.70658712 | 0.19370568                 | 0.09970720 |
| 5  | 0.67819453 | 0.20946991                 | 0.11233556 |
| 6  | 0.81214438 | 0.12974946                 | 0.05810616 |
| 7  | 0.19018890 | 0.27806217                 | 0.53174893 |
| 8  | 0.26447553 | 0.30966905                 | 0.42585543 |
| 9  | 0.36098877 | 0.31830785                 | 0.32070338 |
| 10 | 0.57290832 | 0.26124361                 | 0.16584807 |
| 11 | 0.36098877 | 0.31830785                 | 0.32070338 |
| 12 | 0.32513642 | 0.31853858                 | 0.35632501 |

- Predicted probability each individual is in each level of the outcome
- This is useful, but it would be good to have this information linked to our predictors

## 6. Predicted probabilities

```
happiness_order_with_pp <- cbind(happiness_order, predicted_probabilities)
```

Binds our original  
dataframe and the  
predicted  
probabilities

|   | Participant_ID | Hours_free_time | marital_status | Hamster | Happy                      | Disagree   | Neither agree nor disagree | Agree      |
|---|----------------|-----------------|----------------|---------|----------------------------|------------|----------------------------|------------|
| 1 | 1              | 19              | Single         | Yes     | Agree                      | 0.16867207 | 0.26338788                 | 0.56794005 |
| 2 | 2              | 6               | Single         | No      | Disagree                   | 0.70658712 | 0.19370568                 | 0.09970720 |
| 3 | 3              | 8               | Single         | Yes     | Neither agree nor disagree | 0.50351583 | 0.28826279                 | 0.20822138 |
| 4 | 4              | 6               | Single         | No      | Disagree                   | 0.70658712 | 0.19370568                 | 0.09970720 |
| 5 | 5              | 3               | Single         | Yes     | Disagree                   | 0.67819453 | 0.20946991                 | 0.11233556 |
| 6 | 6              | 2               | Single         | No      | Disagree                   | 0.81214438 | 0.12974946                 | 0.05810616 |
| 7 | 7              | 18              | Single         | Yes     | Neither agree nor disagree | 0.19018890 | 0.27806217                 | 0.53174893 |
| 8 | 8              | 19              | Single         | No      | Agree                      | 0.26447553 | 0.30966905                 | 0.42585543 |

# Checking assumptions

---

- Linearity of the logit
- No multicollinearity
- Same as in part 2

# 7. Proportional odds assumption Very important!!!

```
library(brant)
```

```
brant(model)
```

```
> brant(model)
```

| Test for                 | X2   | df | probability |
|--------------------------|------|----|-------------|
| Omnibus                  | 3.97 | 5  | 0.55        |
| HamsterYes               | 1.5  | 1  | 0.22        |
| Hours_free_time          | 0.16 | 1  | 0.69        |
| marital_statusCohabiting | 0.7  | 1  | 0.4         |
| marital_statusMarried    | 1.64 | 1  | 0.2         |
| marital_statusDivorced   | 1.27 | 1  | 0.26        |

H0: Parallel Regression Assumption holds

P-value

Omnibus = model

Also value for each comparison (e.g. continuous predictor or comparison for categorical predictors)

Sometimes goes off centre – last number is the p-value (e.g. 0.69 is the p-value for Hours\_free\_time)

# 7. Proportional odds assumption

## Very important!!!

```
library(brant)
```

```
brant(model)
```

```
> brant(model)
```

| Test for                 | X2   | df | probability |
|--------------------------|------|----|-------------|
| Omnibus                  | 3.97 | 5  | 0.55        |
| HamsterYes               | 1.5  | 1  | 0.22        |
| Hours_free_time          | 0.16 | 1  | 0.69        |
| marital_statusCohabiting | 0.7  | 1  | 0.4         |
| marital_statusMarried    | 1.64 | 1  | 0.2         |
| marital_statusDivorced   | 1.27 | 1  | 0.26        |

H0: Parallel Regression Assumption holds

P-value

Sometimes goes off centre – last number is the p-value (e.g. 0.69 is the p-value for Hours\_free\_time)

If  $p > .05$  for all, no violation of the proportional odds assumption

# Thank you for listening!

Please post any questions on the relevant Qualtrics link on Moodle.