Response opts further test

2023-01-31

Contents

Intro

We produced morphed faces of different levels of femininity and masculinity. There were 18 continua, where gender varied in seven increments, for a total of 126 faces.

There were five response options conditions:

1. binary categories - man/woman
2. multiple categories - man/woman/other/don't know
3. Freetext - a free text box
4. binary dimension - woman ——- man on a slider
5. multiple dimensions - woman / man on separate sliders.

At the previous meeting, we decided that i would look at answering the question "do people use non-binary options when they have them?"

The way I've gone about doing that is by primarily comparing option the results from the multiple categories (option2) and freetext conditions (option 3).

Do people use the beyond-binary options?

The way I went about testing that is by comparing the amount of beyond-binary (i.e. "other" and "I don't know" responses) across the free text and multiple categories conditions. Here's snippet of the data for some context:

```
## # A tibble: 6 x 4
##   categorization condition fem   bbcat
##   <chr>          <chr>     <fct> <dbl>
## 1 m              ft        0         0
## 2 m              ft        16.67     0
## 3 o              ft        33.33     1
## 4 f              ft        50        0
## 5 f              ft        66.67     0
## 6 f              ft        83.33     0
```

Simple explanation So how do we actually test the question "do people use the non-binary option when they have them"? We can construct several models which gets increasingly more complicated and compare how well they predict the data.

In short, the idea is simple. We make three models:

- m0: a null model with no predictors
- m1: main effects model, with the predictors “morph level” and response option condition
- m2: interaction model, which is the the same as m1, with an additional interaction effect

Then we see which model best explains the data using leave-one-out cross validation. (if you want to go into the weeds on the models and what the models actually consist of, click the “complex model tabs”)

Complex models So we start with a null model, m_0, which just has an intercept, and includes neither morph level or condition. For reference, it looks like this, but feel free to let your eyes glaze over it. (but in case your curious, the model includes random effects for subjects and faces, but nothing else)

$$\begin{aligned}
\text{beyond-binary}_i &\sim \text{Binomial}(1, p) \\
\text{logit}(p_i) &= \alpha_{subject[i]} + \alpha_{face[i]} \\
\alpha_{face} &\sim \text{Normal}(0, \sigma_{\text{subject}}) \\
\alpha_{subject} &\sim \text{Normal}(0, \sigma_{subject}) \\
\sigma_{face} &\sim \text{HalfCauchy}(3) \\
\sigma_{subject} &\sim \text{HalfCauchy}(3)
\end{aligned} \tag{1}$$

Then m_1 adds fixed effect of condition and morph level. We can think about this as basically testing just the main effect of both condition (written out as γ_{cid}) and morph level ($\beta_1 Morph_i$), but not the interaction (which we’ll get to later). *cid* is short for condition id, which means that this model calculates separate values for each condition, as opposed to a dummy variable, which would calculate the value (intercept) for one condition and then the difference of another condition. Here we just get values for each condition, and if we want to know the difference, we have to calculate it ourselves.

To add another level of complexity, we can make a second model m_1 which includes the effects of both response option condition and morph level.

$$\begin{aligned}
\text{beyond-binary}_i &\sim \text{Binomial}(1, p) \\
\text{logit}(p_i) &= \gamma_{cid[i]} + \alpha_{subject[i]} + \beta_{[1]} Morph_i + \alpha_{face[i]} \\
\gamma_{cid} &\sim \text{Normal}(0, 3) \\
\alpha_{subject} &\sim \text{Normal}(0, \sigma_{subject}) \\
\beta_1 &\sim \text{Normal}(0, 3) \\
\sigma_{face} &\sim \text{HalfCauchy}(3) \\
\alpha_{subject} &\sim \text{HalfCauchy}(3) \\
\sigma_{\gamma_{pronoun}} &\sim \text{HalfCauchy}(3) \\
\mathbf{R} &\sim \text{LKJcorr}(2)
\end{aligned} \tag{2}$$

Lastly, m_2 includes the full interaction effect. Note that all we’ve really done is changed β by also adding the *cid* subscript. What that means is that we’re calculating a unique effect of morph level for each condition. Again, the model doesn’t include an *explicit* interaction term (which, in standard notation, be the difference between the a reference condition and another condition), but we can calculate it later. And we will, don’t worry!

Table 1: Relative predictive power of models describing the outcome on the categorization task

	LOO difference	St. Error diff	LOO	St. Error LOO
m2	0.00	0.00	-234.17	23.23
m1	-2.46	2.71	-236.63	23.07
m0	-18.83	6.02	-253.00	24.51

Note. LOO diff refers to the difference in loo between the model and the most predictive model. The first row describes the most predictive model, which is why the difference is 0

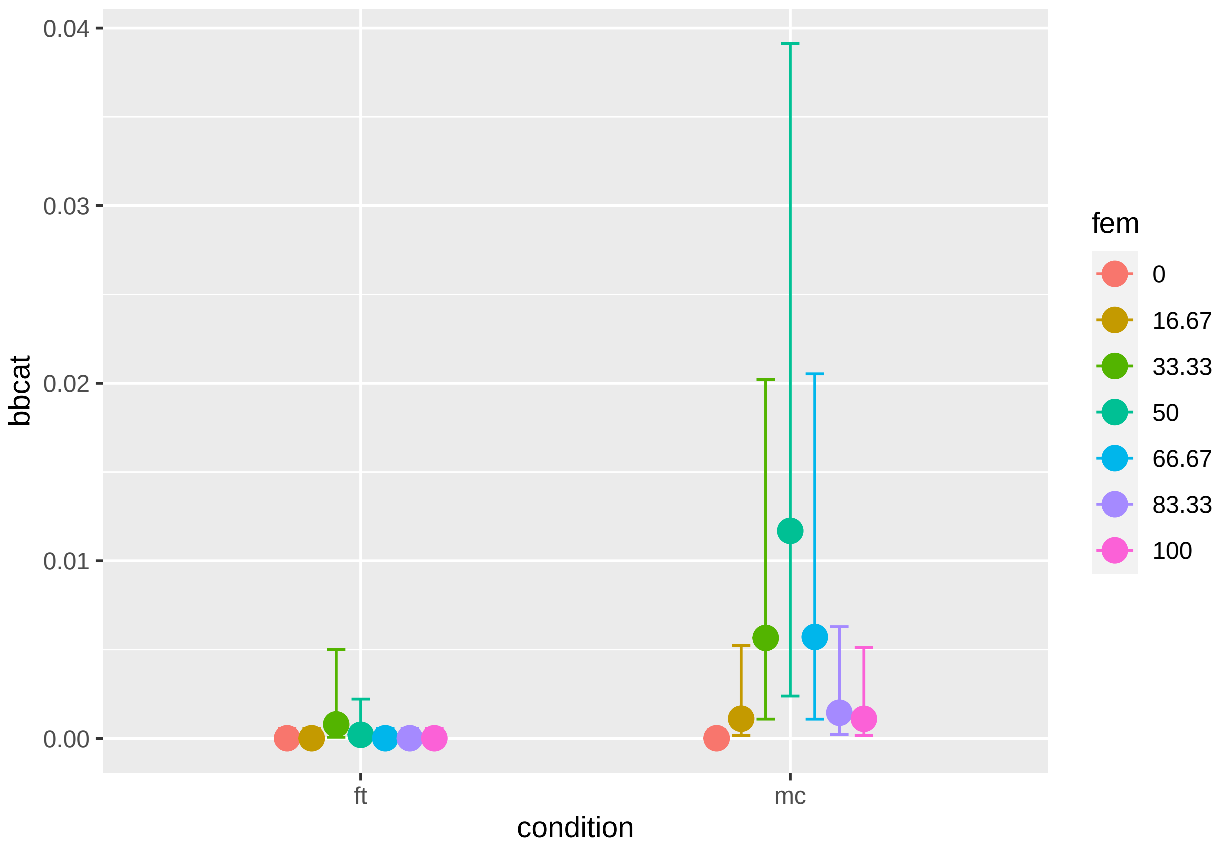
$$\begin{aligned}
\text{beyond-binary}_i &\sim \text{Binomial}(1, p) \\
\text{logit}(p_i) &= \gamma_{cid[i]} + \alpha_{subject[i]} + \beta_{cid[i]} Morph + \alpha_{face[i]} \\
\gamma_{cid} &\sim \text{Normal}(0, 3), \text{ for } cid = \text{ft, mc} \\
\alpha_{subject} &\sim \text{Normal}(0, \sigma_{subject}) \\
\beta_{cid} &\sim \text{Normal}(0, 3), \text{ for } cid = \text{ft, mc} \\
\sigma_{face} &\sim \text{HalfCauchy}(3) \\
\sigma_{subject} &\sim \text{HalfCauchy}(3) \\
\sigma_{\gamma_{pronoun}} &\sim \text{HalfCauchy}(3) \\
\mathbf{R} &\sim \text{LKJcorr}(2)
\end{aligned} \tag{3}$$

Results

Having fit all of these models I then compare them to see which best predicts the data. We do this using a method call leave-one-out cross validation. This tells us which of our five models best predict the data on “new” or out of sample data points? If I can get the machine to work, this should show up in table 1

Table 1. suggests that the interaction models was the best at predicting data, suggesting that the interaction between condition and morph is an important determinant of the outcome. However, the comparatively large standard errors suggests that this effect is only inconclusively supported by the data.

Pairwise comparisons I’m going to forge ahead anyway, and do what we might think of as contrast analyses, focusing on m_3 only. Here we have several specific questions that we want answered. First, is it really the case that people generally make more beyond-binary categorizations in the mc condition? And secondly, is this effect concentrated at 50/50 morph level. First, let’s take a look at the data visualised.



First, if we just look at the number of beyond-binary categorizations.

H1. Do particpants make beyond-binary categorizations in the multiple categories condition compared to the free text condition? It seems they do (Estimate = -4.11, CI =[-6.77], [-1.56], BF$_{10}$= 61.58).

H2. If we focus on the 50/50 faces, do we still see an effect? It seems like we do (Estimate = -4.14, CI =[-7.43], [-1.4], BF$_{10}$= 16.98).

H3. And if we focus on the faces at the end of the spectrum (i.e. 0 and 100 faces)? Here the evidence is only inconclusive (Estimate = -2.6, CI = [-7.96] - [2.49], BF$_{10}$= 0.8). What does this mean? Well, there is an overall difference between the two conditions, and it is concentrated at the 50/50 faces

How does this effect the relative distribution of m/f scores

Great, let's move on to the step 2 that we talked about. So having answered the questions of "do people actually use the extra response options" with a resounding "probably, but not so much", I'm going to move on to the second question: is it the case that the answers shift the distribution of m/f scores? What I mean is, are people replacing their ratings of "man" with ratings of "non-binary" for example. If that were to be the case, then we would expect the relative distribution of "man" scores to be somewhat lower in the multiple categories condition

Spoilers, it seems maybe not. Anyway, the first thing I did was make a version of the data where all the beyond-binary responses were taken out. Sort of the opposite of what I did last. I named the relevant variable f_cat because naming things is hard.

```
## # A tibble: 6 x 3
##   condition categorization f_cat
```

Table 2: Relative predictive power of models describing the outcome on the categorization task

	LOO difference	St. Error diff	LOO	St. Error LOO
morph_only	0.00	0.00	-1343.72	43.10
Null	-5.35	4.92	-1349.07	44.12
main_effects	-37.60	11.96	-1381.31	44.77
interaction	-42.10	12.09	-1385.81	45.14
condition_only	-4398.47	44.23	-5742.19	11.18

Note. LOO diff refers to the difference in loo between the model and the most predictive model. The first row describes the most predictive model, which is why the difference is 0

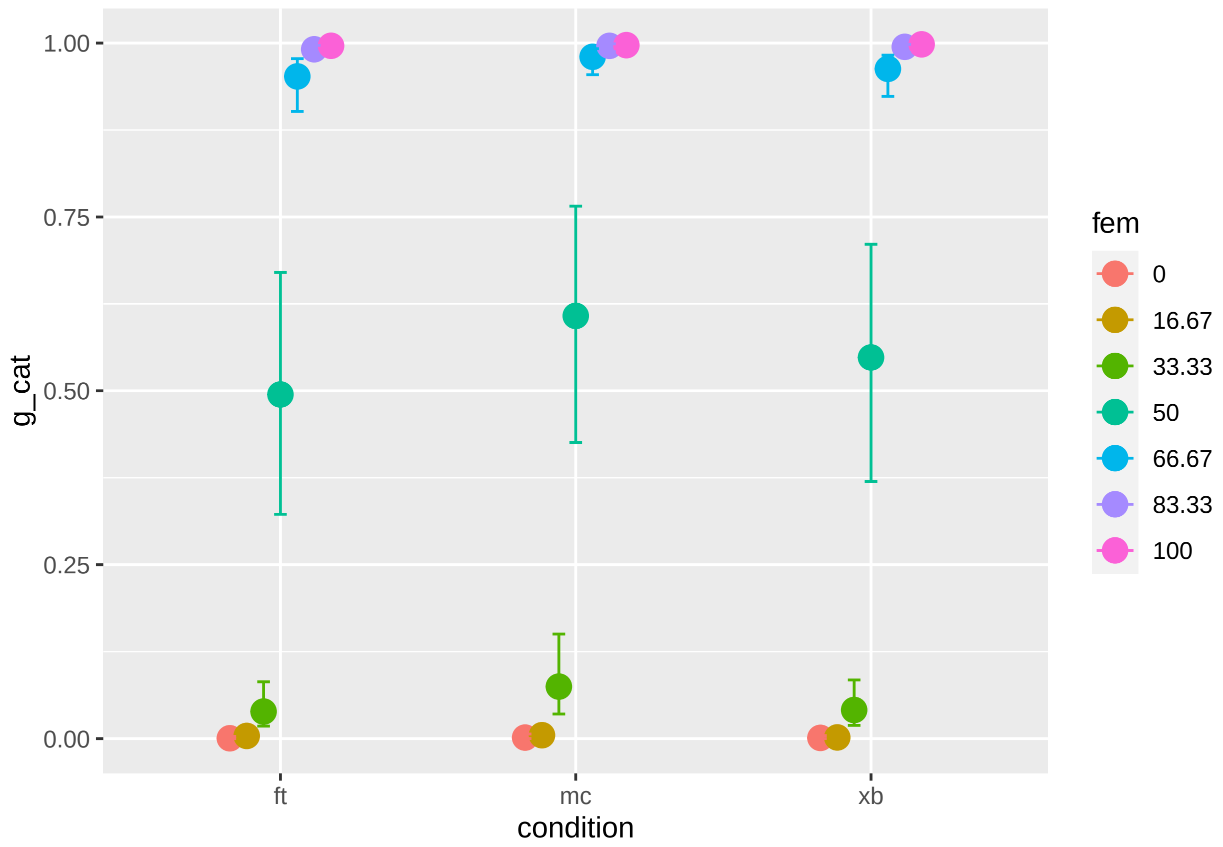
```
##   <chr>     <chr>          <dbl>
## 1 ft        m                  0
## 2 ft        m                  0
## 3 ft        o                 NA
## 4 ft        f                  1
## 5 ft        f                  1
## 6 ft        f                  1
```

I'm tired, your tired, so I'm going to move through this a little more quickly than the previous stuff. But basically, I went through the same steps as for the previous dataset, note that now I included the binary category condition this time.

We're going to do the same thing again, we're going to start buy looking at whether the interaction model does better than the simple effects model, and then we're going to look at the actual effects within the model. Buuut. It doesn't seem lik it does much here. I'm going to show all that in table 2

If we start by looking at which model is the best at predicting the data, it's clear that only the morph only model really does a good job at predicting the outcomes. What does that mean? It means that morph level is the only variable that's really useful for predicting the outcome. Adding condition

Just to give a sense of what the data actually look like, I'm also going to put up a figure for the interaction between condition and morph level.



In this figure, we can see that the probability of any person answering “woman” on any face depends mostly on that face’s femininity. We can see that for 50/50 faces, this probability is veery slightly higher in the mc condition, but the results from model comparison and the fact that the confidence intervals overlap so widely make me think that this difference is pretty small.

```
##  Family: bernoulli
##   Links: mu = logit
## Formula: g_cat ~ 0 + condition:fem + (1 | id) + (1 | face)
##    Data: tmp (Number of observations: 8368)
##   Draws: 4 chains, each with iter = 6000; warmup = 2000; thin = 1;
##          total post-warmup draws = 16000
##
## Group-Level Effects:
## ~face (Number of levels: 18)
##               Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)     1.40      0.26     0.99     2.01 1.00     3492     6509
##
## ~id (Number of levels: 68)
##               Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)     1.01      0.11     0.82     1.25 1.00     4595     6529
##
## Population-Level Effects:
##                      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS
## conditionft:fem0        -7.75      1.25   -10.58    -5.72 1.00    10589
## conditionmc:fem0        -6.58      0.91    -8.60    -5.02 1.00     7859
## conditionxb:fem0        -6.94      0.87    -8.88    -5.43 1.00     7240
## conditionft:fem16.67    -5.58      0.61    -6.86    -4.45 1.00     4441
## conditionmc:fem16.67    -5.34      0.62    -6.63    -4.20 1.00     4987
## conditionxb:fem16.67    -6.42      0.74    -8.02    -5.09 1.00     7025
## conditionft:fem33.33    -3.21      0.40    -4.00    -2.42 1.00     2313
## conditionmc:fem33.33    -2.52      0.40    -3.31    -1.73 1.00     2415
## conditionxb:fem33.33    -3.15      0.39    -3.94    -2.39 1.00     2492
## conditionft:fem50       -0.02      0.37    -0.74     0.71 1.00     1979
## conditionmc:fem50        0.44      0.38    -0.30     1.18 1.00     2167
## conditionxb:fem50        0.19      0.36    -0.53     0.90 1.00     2156
## conditionft:fem66.67     2.99      0.40     2.22     3.77 1.00     2179
## conditionmc:fem66.67     3.91      0.45     3.04     4.81 1.00     2806
## conditionxb:fem66.67     3.26      0.40     2.49     4.03 1.00     2426
## conditionft:fem83.33     4.72      0.48     3.80     5.68 1.00     3046
## conditionmc:fem83.33     5.53      0.61     4.38     6.78 1.00     4463
## conditionxb:fem83.33     5.22      0.50     4.27     6.23 1.00     3523
## conditionft:fem100       5.55      0.58     4.46     6.75 1.00     4201
## conditionmc:fem100       5.83      0.66     4.62     7.22 1.00     5159
## conditionxb:fem100       6.33      0.65     5.12     7.69 1.00     5410
##                      Tail_ESS
## conditionft:fem0         8175
## conditionmc:fem0         9024
## conditionxb:fem0         9364
## conditionft:fem16.67     8135
## conditionmc:fem16.67     9344
## conditionxb:fem16.67     9280
## conditionft:fem33.33     5002
## conditionmc:fem33.33     5287
## conditionxb:fem33.33     5081
## conditionft:fem50        4139
## conditionmc:fem50        4164
## conditionxb:fem50        4147
## conditionft:fem66.67     4792
## conditionmc:fem66.67     5454
## conditionxb:fem66.67     5237
## conditionft:fem83.33     5990
## conditionmc:fem83.33     7544
## conditionxb:fem83.33     6886
## conditionft:fem100       7602
## conditionmc:fem100       7760
## conditionxb:fem100       8877
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

```
## Hypothesis Tests for class b:
##                 Hypothesis Estimate Est.Error CI.Lower CI.Upper Evid.Ratio
## 1 (conditionft:fem5... = 0    -0.21      0.34    -0.87     0.45      10.28
##   Post.Prob Star
## 1      0.91
## ---
## 'CI': 90%-CI for one-sided and 95%-CI for two-sided hypotheses.
## '*': For one-sided hypotheses, the posterior probability exceeds 95%;
## for two-sided hypotheses, the value tested against lies outside the 95%-CI.
## Posterior probabilities of point hypotheses assume equal prior probabilities.
## Warning in regularize.values(x, y, ties, missing(ties)): collapsing to unique
## 'x' values

## Warning in regularize.values(x, y, ties, missing(ties)): collapsing to unique
## 'x' values
```

```
## Hypothesis Tests for class b:
##                  Hypothesis Estimate Est.Error CI.Lower CI.Upper Evid.Ratio
## 1 (conditionft:fem5... = 0        0         0        0        0          1
##   Post.Prob Star
## 1       0.5
## ---
## 'CI': 90%-CI for one-sided and 95%-CI for two-sided hypotheses.
## '*': For one-sided hypotheses, the posterior probability exceeds 95%;
## for two-sided hypotheses, the value tested against lies outside the 95%-CI.
## Posterior probabilities of point hypotheses assume equal prior probabilities.
```