BCS 511 Minilab 2 - Psychophysics: Sex-related differences in motion processing?

T. Florian Jaeger

11/1/2019

1 Goal of minilabs

The goal of the BCS 511 minilabs is to improve our coding, data analysis skills, and communications of scientific findings. Please use this opportunity to think about how you would communicate results effectively to your collaborators, PIs, and to your future advisees. For this reason, you are asked to submit a preliminary report each week of the minilab, and the final report at the last day of the minilab (see below for details). Our scientific reports—even those we 'just' use to communicate our approach and findings within our own lab—should be self-contained and accessible, so that you avoid unnecessary burden on our audience.

1.1 Requirements and due dates

Each minilab will come with an overarching goal and a set of specific questions to be addressed for each week of the lab. Please answer these questions in an R markdown report submitted *each weak of the minilab* no later than 8am the day of the class. Additionally, the *final report* is due 8am the day of the last class meeting related to the lab. The final report is simply the accumulation of the weekly reports, potentially revised to incorporate feedback you have received.

It is strongly recommended that you start the minilab absolutely no later than at least 3 days in advance—like in your everyday research, unexpected problems are bound to come up and you want there to be enough time to address those problems. **Finally, be prepared to present your solutions each week in class.** We will go through the minilabs in class, review your answers, and help each other where we got stuck.

1.2 What should go into the report?

The goal of the report is to answer the research questions described in the minilab brief. Provide visual and table summaries where adequate, provide captions where required to understand these summaries, provide a clear written answer. Just as you might in a collaboration, please make sure to describe your analysis plan.

Your report should only contain parts that you need to answer the questions you are asked to research. (So if you copy parts of the assignment Rmd file, please make sure to remove those parts). If your report had any dependencies on external files, figures etc. please make sure that all of these materials are included and the file names used to refer to them are relative, rather than absolute (i.e., not "~/MyStuff/ProjectX/BCS511/scripts/" but something like "../scripts/").

While we want the report to show the code (echo=T), you should make sure that the interpretation of the figures, tables, etc. does not depend on reading the code. I.e., make sure that the report is readable even if the code prompts are not shown (i.e., if the global knitr option was to be changed to echo=F).

1.3 What if I get stuck? Can I ask for help?

You can ask other class members and the instructor for help at any point. Please us the slack channel #general for that. This ensures that everyone benefits from whatever answers are provided, and that everyone has access to the same resources. There is not limit to the collaboration you can engage in, as long as it is through the slack channel.

Try to get as far as you can. If you get stuck in R, try to describe in your own words what you're trying to achieve. You can also add hand-drown figures to the report, as long as you import them (e.g., take a photo of a plot with your phone and add it to the report.) First of all, the minilabs are a **learning experience**, so make sure to use them as such. Don't be embarassed to ask, help each other.

2 Goal of Minilab 2

The overarching goals for minilab 2 is learn how to analyze psychophysics data, and think about researchers' degrees of freedom in those analyses: how do the choices we make in setting up our analyses affect the conclusions we draw?

In Week 1, we focus on the output of the Naka-Rushton model discussed in class, and examine to what extent the parameters of this model differ between female and male participants. We also aim to:

- increase our familiarity with data visualization
- increase out fluency in *multiple* linear regression, in order to determine the *partial* effects of multiple predictors.

In Week 2, we aim to will revisit assumptions about the analyses that went into the Naka-Rushton parameters analyzed in Week 1. We'll think about what assumptions we are implicitly making when we analyze such meta-parameters, and explore alternative approaches to analyze gender-related differences in the *raw* data.

2.1 Background

Murray et al. (2018) report sex-related differences in visual motion perception. Their paradigm "utilizes briefly presented drifting gratings. After each stimulus presentation, participants are asked to classify the stimulus as moving leftward or rightward." (p. 2794). The paper is available as part of this lab.

The data you will be working with was collected by three undergraduate researchers (Penny, Kailin and Sunny) with three goals in mind. First, to see whether the sex-related effect observed in Murray et al. replicates (again). Second, to narrow down under what tasks the sex-related effect is observed. Specifically, the sex-related differences in the original motion task could be restricted to the sex-related differences in motion processing (only motion stimuli were used), differences in processing brief visual stimuli (stimuli were brief), or generalize to a wide range of stimuli. The present data thus comes from three different tasks. The *Motion* task is (more or less) the same as Murray et al. The static *Mask* task estimates the minimum presentation duration required to do a coarse orientation discrimination task (-45 deg vs 45 deg tilt). These stimuli are similarly brief as the motion stimuli. The static *Angular* task estimates the the ability to do a fine orientation discrimination (e.g., -1 deg vs 1 deg) for longer stimuli (400 ms). In all tasks, the stimuli were presented at a wide range of stimulus contrasts.

The third goal is to understand whether the sex-related differences observed in Murray et al. (2018) might originate in individual differences that are correlated with sex. For example, Murray et al already noted a striking similarity between the sex-related differences in motion processing they observed, and differences observed between neurotypical subjects and subjects with autism spectrum disorder. Penny, Kailin and Sunny thus collected five individual difference measures for each of their subjects (*EmphatisingQuotient*, *SystemizingQuotient*, *AutismQuotient*, *ADHD*, and the hours/day spent *VideoGaming*). The Emphatising and Systemising quotients measure the corresponding component of the social cognition (Wheelwright et al., 2006). The Autism quotient measures autism-like tendencies in the general population.

2.2 The Naka-Rushton model

The Nake-Rushton model can model s-shaped functional relations. In the present work, the inversion of that model is used. Note that this can make some of the parameter names of the model somewhat confusing. For example, *Rmax* is now the *minimum* duration threshold estimate against which the model converges.

$$DurationThreshold = f(contrast) = \frac{1}{R_{max}\frac{contrast^{Slope}}{contrast^{Slope}+C50^{Slope}}+R_0}$$

2.3 Data Description

The raw psychophysics data consists of individual trials with various stimulus durations (log-transformed to base 10, *LogDuration*) and contrast values (between 0-100, *Contrast*). Each trial was either answered correctly (*Correct* == 1) or not (0). Each *Subject* participated in 3 *Task*s ("Motion", "Mask", "Angular"), completing multiple *Block*s for each task.

A total of 43 subjects were run, with the goal of balancing the sample across *Sex* (1 = "Female" vs. 0 = "Male). This resulted in 51600 observations. Additionally, five individual difference variables were collected for each subject: *EmphatizingQuotient*, *SystemizingQuotient*, *AutismQuotient*, *ADHD*, and the hours/day spent *VideoGaming*. These and the three parameters of the Naka-Rushton model (*Rmax*, *Slope*, and *C50*) are attached to each row (i.e., trial-level observation) in the data—that is, the data is in long format. Additionally, the researchers provided three transformations of the two parameters of the Naka-Rushton model that they consider most important. The asymptotic performance (*Asymp* = 1 / *Rmax*) and its log-transform (*LogAsymp* = log_{10}(*Asymp*) = -log_{10}(*Rmax*)), as well as the log-transformed of the semi-saturation transform (*LogC50* = log_{10}(C50)). It is the log-transformed variants of Rmax and C50 that the team analyzed.

Of the individual difference variables, the three "Quotient" measures range from 0 to whateva, with higher values indicating higher scoring on that trait. *ADHD* ranges from 1-18, with higher values indicating more attention deficits and hyperactivity. Finally, the *VideoGaming* variable is coded as three-level factor (0 = no significant action video game playing; 2 = significant action video game playing; 1 = between 0 and 2).

2.4 Week 1

For the first week of Minilab 2, our goal is to see whether the data for Minilab 2 replicates the effect observed by Murray et al. (2018). We will introduce a few methodological advances.

HINT: All analyses in Week 1 analyze summary data, with one observation per participant. For Week 1, you thus will want to summarize the data down to one row per participant (rather than trial-level data). Since all the columns your interested in for Week 1 are constant across all trials of the subject, one efficient way to get the data you want is simply to select the columns of interest and then use *distinct* (look it up).

1. Analyze whether the present data replicates the effect of gender on motion perception observed in previous work. Start by visualizing and analyzing the distribution of the three parameters of the Naka-Rushton model (*Rmax*, *Slope*, *C50*) for the *Motion* task. Visualize and write-up your analysis including adequate description of the statistics in text form (e.g., "There was a significant main effect of participant's sex on Rmax ($\hat{\beta}$ = XX.X, t = XX.X, p < .XX), so that female participants had a higher/lower minimum duration threshold."). HINTS:

    a. Visualization is often helpful. In this case, you might want to visualize the differences in duration threshold and/or the four parameters of the Naka-Rushton model. Consider histograms or violin plots to visualize the distribution, or use points and point ranges.
    b. To analyze sex-related effects you can use *t*-test or (equivalently) linear model analysis. If you're using a linear model, don't forget to sum code the factor *Sex* if you you want to assess main effects. The function *prepVars* in this document shows one way to do that.
    c. You can take advantage of *gather*, *spread*, and *broom* to create the various plots and models you need.

2. Would you answer to question 1 change in any way if either of the two alternative measures for Rmax (i.e., *Asymp* or *LogAsymp*) or the alternative measure for C50 (i.e., *LogC50*) are examined instead? Would your answer change if you excluded from your analysis any parameter value that is more than +/-3 SDs from the cross-subject mean of that parameter? If the answer to any of these question is “yes”, what do you make out of that?

3. In your analysis of the four Naka-Rushton parameters, you might find significant effects of *Sex* on some, but not all, parameters. Does this mean that we can safely conclude that the different parameters differ qualitatively in how they are affected by participants’ sex? How could you use linear models to test whether the sex-related effects differ significantly between the four parameters of the Naka-Rushton model? HINTS:

   a. Think about what interactions allow you to test.
   b. The coefficient report that is part of the standard output of a linear model provides us with the effect of each individual parameter. If we want to know whether a multi-level factor and/or all of its interactions has a significant effect (i.e., whether all parameters that result from the coding of the factor *together* add significantly to the model fit), we can do so via nested model comparison. The R command for nest model comparison is *anova(smaller nested model, larger nesting model, test = “Chisq”)*. This conducts a χ^2-test over the difference in deviance between the models. The DFs of the χ^2-test correspond to the difference in the number of parameters between the smaller and larger model (for sufficiently large data sets, the difference in deviance between nested models approximates a χ^2 distribution).
   c. You might have to wrangle the data into a shape that allows you to analyze all parameters in one model. *gather* might come in handy.

4. Extend the analyses of Steps 1-2 to the two other tasks (“Angular”, “Mask”). Do the results differ for these tasks? Are those differences significant?

5. Next, explore to what extent the sex-related effects might be mediated through any of the five individual difference variables. For this analysis, you can limit yourself to the *Rmax* parameter and the motion *Task*. Begin by visualize the effects of the individual difference variables on *Rmax*. Then assess the correlations between the different individual difference variables. If they are correlated with each other and/or *Sex*, then try to tease apart the effects of the different variables (incl. *Sex*) on *Rmax*. HINTS:

   a. You can add additional predictors to the linear model to see whether they are significant.
   b. Note that *collinearity* between multiple predictors can affect the standard error estimates (and thus p-values) of the coefficients. Nested model comparison can be useful in this context.
   c. Note, however, that there is no free lunch. Explorative model building—either “forward” by adding predictors to a model, or “backwards model reduction” by removing predictors from the full model—inevitably increases the number of tests we conduct, and thus the number of false significances (Type I error) we will get just by chance (Harrell, 2001 provides a succinct summary of this problem). It’s thus best to first state your hypothesis and then conduct as few tests as possible.

2.5 Week 2

TBA

3 References

- Wheelwright, S., Baron-Cohen, S., Goldenfeld, N., Delaney, J., Fine, D., Smith, R., … & Wakabayashi, A., 2006). Predicting autism spectrum quotient (AQ) from the systemizing quotient-revised (SQ-R) and empathy quotient (EQ). Brain research, 1079(1), 47-56

4 Session info

```
## - Session info -----------------------------------------------------------------------------------
##  setting  value
##  version  R version 3.6.0 (2019-04-26)
##  os       macOS High Sierra 10.13.6
##  system   x86_64, darwin15.6.0
##  ui       X11
##  language (EN)
##  collate  en_US.UTF-8
##  ctype    en_US.UTF-8
##  tz       America/New_York
##  date     2019-11-04
##
## - Packages ---------------------------------------------------------------------------------------
##  package        * version    date       lib source
##  abind            1.4-5      2016-07-21 [1] CRAN (R 3.6.0)
##  assertthat       0.2.1      2019-03-21 [1] CRAN (R 3.6.0)
##  backports        1.1.5      2019-10-02 [1] CRAN (R 3.6.0)
##  base64enc        0.1-3      2015-07-28 [1] CRAN (R 3.6.0)
##  bayesplot        1.7.0      2019-05-23 [1] CRAN (R 3.6.0)
##  boot             1.3-23     2019-07-05 [1] CRAN (R 3.6.0)
##  bridgesampling   0.7-2      2019-07-21 [1] CRAN (R 3.6.0)
##  brms           * 2.10.0     2019-08-29 [1] CRAN (R 3.6.0)
##  Brobdingnag      1.2-6      2018-08-13 [1] CRAN (R 3.6.0)
##  broom          * 0.5.2      2019-04-07 [1] CRAN (R 3.6.0)
##  callr            3.3.2      2019-09-22 [1] CRAN (R 3.6.0)
##  cellranger       1.1.0      2016-07-27 [1] CRAN (R 3.6.0)
##  cli              1.1.0      2019-03-19 [1] CRAN (R 3.6.0)
##  coda             0.19-3     2019-07-05 [1] CRAN (R 3.6.0)
##  colorspace       1.4-1      2019-03-18 [1] CRAN (R 3.6.0)
##  colourpicker     1.0        2017-09-27 [1] CRAN (R 3.6.0)
##  crayon           1.3.4      2017-09-16 [1] CRAN (R 3.6.0)
##  crosstalk        1.0.0      2016-12-21 [1] CRAN (R 3.6.0)
##  desc             1.2.0      2018-05-01 [1] CRAN (R 3.6.0)
##  devtools         2.2.1      2019-09-24 [1] CRAN (R 3.6.0)
##  digest           0.6.22     2019-10-21 [1] CRAN (R 3.6.0)
##  dplyr          * 0.8.3      2019-07-04 [1] CRAN (R 3.6.0)
##  DT               0.9        2019-09-17 [1] CRAN (R 3.6.0)
##  dygraphs         1.1.1.6    2018-07-11 [1] CRAN (R 3.6.0)
##  ellipsis         0.3.0      2019-09-20 [1] CRAN (R 3.6.0)
##  evaluate         0.14       2019-05-28 [1] CRAN (R 3.6.0)
##  fastmap          1.0.1      2019-10-08 [1] CRAN (R 3.6.0)
##  forcats        * 0.4.0      2019-02-17 [1] CRAN (R 3.6.0)
##  fs               1.3.1      2019-05-06 [1] CRAN (R 3.6.0)
##  generics         0.0.2      2018-11-29 [1] CRAN (R 3.6.0)
##  ggeffects      * 0.12.0     2019-09-03 [1] CRAN (R 3.6.0)
##  ggplot2        * 3.2.1      2019-08-10 [1] CRAN (R 3.6.0)
##  ggridges         0.5.1      2018-09-27 [1] CRAN (R 3.6.0)
##  glue             1.3.1      2019-03-12 [1] CRAN (R 3.6.0)
##  gridExtra        2.3        2017-09-09 [1] CRAN (R 3.6.0)
##  gtable           0.3.0      2019-03-25 [1] CRAN (R 3.6.0)
##  gtools           3.8.1      2018-06-26 [1] CRAN (R 3.6.0)
##  haven            2.1.1      2019-07-04 [1] CRAN (R 3.6.0)
##  hms               0.5.2      2019-10-30 [1] CRAN (R 3.6.0)
##  htmltools         0.4.0      2019-10-04 [1] CRAN (R 3.6.0)
##  htmlwidgets       1.5.1      2019-10-08 [1] CRAN (R 3.6.0)
##  httpuv            1.5.2      2019-09-11 [1] CRAN (R 3.6.0)
##  httr              1.4.1      2019-08-05 [1] CRAN (R 3.6.0)
##  igraph            1.2.4.1    2019-04-22 [1] CRAN (R 3.6.0)
##  inline            0.3.15     2018-05-18 [1] CRAN (R 3.6.0)
##  insight           0.6.0      2019-10-17 [1] CRAN (R 3.6.0)
##  jsonlite          1.6        2018-12-07 [1] CRAN (R 3.6.0)
##  knitr             1.25       2019-09-18 [1] CRAN (R 3.6.0)
##  later             1.0.0      2019-10-04 [1] CRAN (R 3.6.0)
##  lattice           0.20-38    2018-11-04 [1] CRAN (R 3.6.0)
##  lazyeval          0.2.2      2019-03-15 [1] CRAN (R 3.6.0)
##  lifecycle         0.1.0      2019-08-01 [1] CRAN (R 3.6.0)
##  lme4            * 1.1-21     2019-03-05 [1] CRAN (R 3.6.0)
##  lmerTest        * 3.1-0      2019-02-11 [1] CRAN (R 3.6.0)
##  loo               2.1.0      2019-03-13 [1] CRAN (R 3.6.0)
##  lubridate         1.7.4      2018-04-11 [1] CRAN (R 3.6.0)
##  magrittr        * 1.5        2014-11-22 [1] CRAN (R 3.6.0)
##  markdown          1.1        2019-08-07 [1] CRAN (R 3.6.0)
##  MASS              7.3-51.4   2019-03-31 [1] CRAN (R 3.6.0)
##  Matrix          * 1.2-17     2019-03-22 [1] CRAN (R 3.6.0)
##  matrixStats       0.55.0     2019-09-07 [1] CRAN (R 3.6.0)
##  memoise           1.1.0      2017-04-21 [1] CRAN (R 3.6.0)
##  mime              0.7        2019-06-11 [1] CRAN (R 3.6.0)
##  miniUI            0.1.1.1    2018-05-18 [1] CRAN (R 3.6.0)
##  minqa             1.2.4      2014-10-09 [1] CRAN (R 3.6.0)
##  modelr            0.1.5      2019-08-08 [1] CRAN (R 3.6.0)
##  munsell           0.5.0      2018-06-12 [1] CRAN (R 3.6.0)
##  nlme              3.1-141    2019-08-01 [1] CRAN (R 3.6.0)
##  nloptr            1.2.1      2018-10-03 [1] CRAN (R 3.6.0)
##  numDeriv          2016.8-1.1 2019-06-06 [1] CRAN (R 3.6.0)
##  pillar            1.4.2      2019-06-29 [1] CRAN (R 3.6.0)
##  pkgbuild          1.0.6      2019-10-09 [1] CRAN (R 3.6.0)
##  pkgconfig         2.0.3      2019-09-22 [1] CRAN (R 3.6.0)
##  pkgload           1.0.2      2018-10-29 [1] CRAN (R 3.6.0)
##  plyr              1.8.4      2016-06-08 [1] CRAN (R 3.6.0)
##  prettyunits       1.0.2      2015-07-13 [1] CRAN (R 3.6.0)
##  processx          3.4.1      2019-07-18 [1] CRAN (R 3.6.0)
##  promises          1.1.0      2019-10-04 [1] CRAN (R 3.6.0)
##  ps                1.3.0      2018-12-21 [1] CRAN (R 3.6.0)
##  purrr           * 0.3.3      2019-10-18 [1] CRAN (R 3.6.0)
##  R6                2.4.0      2019-02-14 [1] CRAN (R 3.6.0)
##  Rcpp            * 1.0.2      2019-07-25 [1] CRAN (R 3.6.0)
##  readr           * 1.3.1      2018-12-21 [1] CRAN (R 3.6.0)
##  readxl            1.3.1      2019-03-13 [1] CRAN (R 3.6.0)
##  remotes           2.1.0      2019-06-24 [1] CRAN (R 3.6.0)
##  reshape2          1.4.3      2017-12-11 [1] CRAN (R 3.6.0)
##  rlang             0.4.1      2019-10-24 [1] CRAN (R 3.6.0)
##  rmarkdown         1.16       2019-10-01 [1] CRAN (R 3.6.0)
##  rprojroot         1.3-2      2018-01-03 [1] CRAN (R 3.6.0)
##  rsconnect         0.8.15     2019-07-22 [1] CRAN (R 3.6.0)
##  rstan             2.19.2     2019-07-09 [1] CRAN (R 3.6.0)
##  rstantools        2.0.0      2019-09-15 [1] CRAN (R 3.6.0)
##  rstudioapi       0.10       2019-03-19 [1] CRAN (R 3.6.0)
##  rvest            0.3.4      2019-05-15 [1] CRAN (R 3.6.0)
##  scales           1.0.0      2018-08-09 [1] CRAN (R 3.6.0)
##  sessioninfo      1.1.1      2018-11-05 [1] CRAN (R 3.6.0)
##  shiny            1.4.0      2019-10-10 [1] CRAN (R 3.6.0)
##  shinyjs          1.0        2018-01-08 [1] CRAN (R 3.6.0)
##  shinystan        2.5.0      2018-05-01 [1] CRAN (R 3.6.0)
##  shinythemes      1.1.2      2018-11-06 [1] CRAN (R 3.6.0)
##  sjlabelled       1.1.1      2019-09-13 [1] CRAN (R 3.6.0)
##  sjmisc           2.8.2      2019-09-24 [1] CRAN (R 3.6.0)
##  StanHeaders      2.19.0     2019-09-07 [1] CRAN (R 3.6.0)
##  stringi          1.4.3      2019-03-12 [1] CRAN (R 3.6.0)
##  stringr        * 1.4.0      2019-02-10 [1] CRAN (R 3.6.0)
##  testthat         2.2.1      2019-07-25 [1] CRAN (R 3.6.0)
##  threejs          0.3.1      2017-08-13 [1] CRAN (R 3.6.0)
##  tibble         * 2.1.3      2019-06-06 [1] CRAN (R 3.6.0)
##  tidyr          * 1.0.0      2019-09-11 [1] CRAN (R 3.6.0)
##  tidyselect       0.2.5      2018-10-11 [1] CRAN (R 3.6.0)
##  tidyverse      * 1.2.1      2017-11-14 [1] CRAN (R 3.6.0)
##  usethis          1.5.1      2019-07-04 [1] CRAN (R 3.6.0)
##  vctrs            0.2.0      2019-07-05 [1] CRAN (R 3.6.0)
##  withr            2.1.2      2018-03-15 [1] CRAN (R 3.6.0)
##  xfun             0.10       2019-10-01 [1] CRAN (R 3.6.0)
##  xml2             1.2.2      2019-08-09 [1] CRAN (R 3.6.0)
##  xtable           1.8-4      2019-04-21 [1] CRAN (R 3.6.0)
##  xts              0.11-2     2018-11-05 [1] CRAN (R 3.6.0)
##  yaml             2.2.0      2018-07-25 [1] CRAN (R 3.6.0)
##  zeallot          0.1.0      2018-01-28 [1] CRAN (R 3.6.0)
##  zoo              1.8-6      2019-05-28 [1] CRAN (R 3.6.0)
##
## [1] /Library/Frameworks/R.framework/Versions/3.6/Resources/library
```