R Markdown Basics

Stephanie Simpson

November 08, 2019

Contents

What is R Markdown?

This is a chunk. This chunk tells us that want to show the output of all the other chunks, unless otherwise specified. You can name your chunks whatever you want! It becomes helpful during knitting so you can track your process.

Click on the mini green "play button" to run the chunk individually.

```r
knitr::opts_chunk$set(echo = TRUE)
```

R Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. This allows you to easily share your work with others! They don't have to run your entire code in order to the see the analysis or data analysis that you conducted.

For more details on using R Markdown see http://rmarkdown.rstudio.com or download the "cheat sheet" at: https://rstudio.com/resources/cheatsheets/

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document.

Set up

To make a chunk quickly, us the short cut: command, option, i.

```r
stroop<-read.csv('/Users/Stephanie/Dropbox/Baycrest_Rworkshop/PART 2/stroop_data.csv')
```

```r
stroop_clean <- stroop %>% filter (rt < 1500)

stroop_clean %>%
  group_by(participant) %>%
  summarise(numTrials=length(trial_number))
```
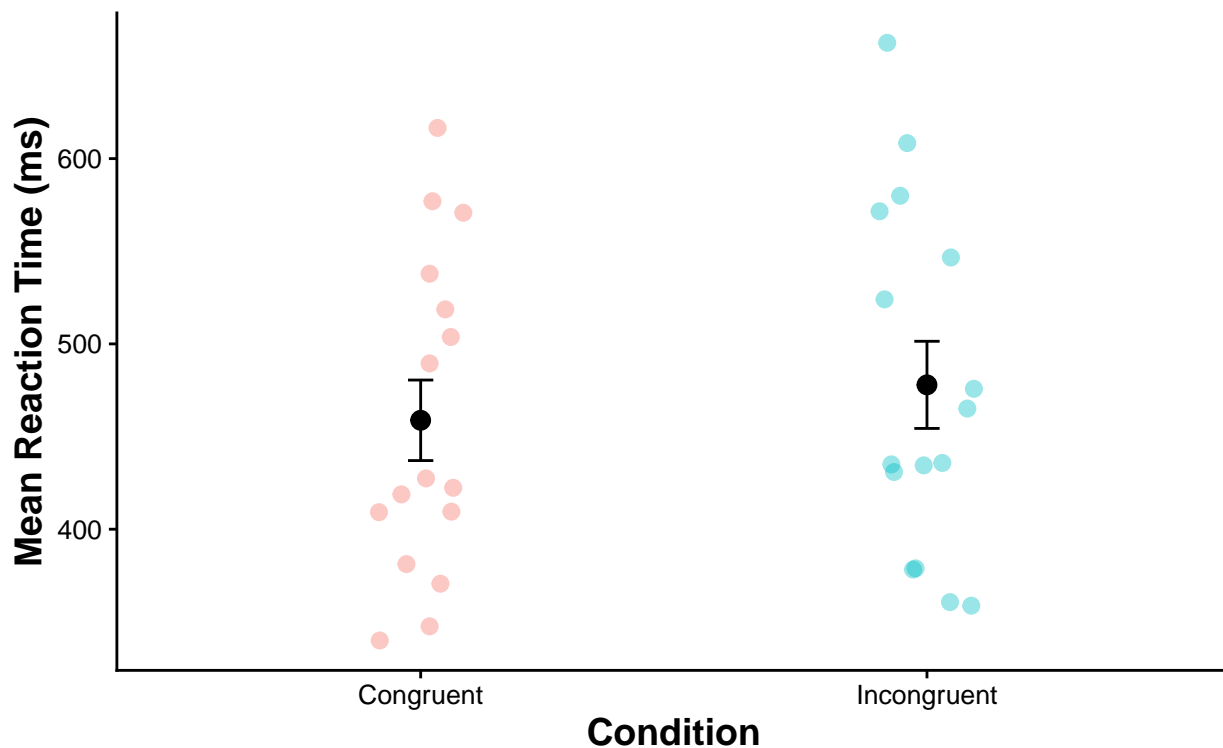
```
## # A tibble: 16 x 2
##    participant numTrials
##          <int>     <int>
## 1        6001       511
## 2        6004       511
## 3        6006       504
## 4        6008       511
## 5        6010       512
```

```
##  6         6011        506
##  7         6013        505
##  8         6015        510
##  9         6017        494
## 10         6018        512
## 11         6020        512
## 12         6023        489
## 13         6024        478
## 14         6026        509
## 15         6029        511
## 16         6031        509
```

Plots

- You can change the echo paramaters to either include or exclude the code in your final output! This is useful depending on which audience you are trying to cater to (i.e., coders vs. non-coders)
- `echo = FALSE` prevents printing of the R code that generated the plot
- `include = FALSE` prevents code and output from being seen in the final file

Analysis

Question: Controlling for the random variation across participants (and tasks), how does the congruency of the trial and accuracy on the trial predict reaction time?

```r
lmer(rt ~ stimulus_congruency*accuracy+(1|participant), data=stroop_clean) %>% summary
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: rt ~ stimulus_congruency * accuracy + (1 | participant)
##    Data: stroop_clean
##
## REML criterion at convergence: 106450.1
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.9765 -0.5793 -0.1840  0.3082  6.3278
##
## Random effects:
##  Groups      Name        Variance Std.Dev.
##  participant (Intercept) 7988     89.37
##  Residual                30466    174.54
## Number of obs: 8084, groups:  participant, 16
##
## Fixed effects:
##                            Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)                  371.51      28.72    40.31  12.934 6.22e-16
## stimulus_congruencyI         103.20      23.85  8065.19   4.328 1.53e-05
## accuracy                      89.43      18.26  8065.65   4.896 9.95e-07
## stimulus_congruencyI:accuracy -86.22     24.17  8065.19  -3.567 0.000363
##
## (Intercept)                   ***
## stimulus_congruencyI          ***
## accuracy                      ***
## stimulus_congruencyI:accuracy ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##            (Intr) stml_I accrcy
## stmls_cngrI -0.474
## accuracy    -0.621  0.745
## stmls_cngI:  0.468 -0.987 -0.753
```

Pros and Cons

- "stream of consciousness" while coding
- easy to share your work (open science)
- easier to run chunks than highlight lines of code
- can sometimes be "buggy"
- might not be great for large datasets because it takes too long to knit