Proportional reasoning across formats

Madison Chin[1]

[1] Rutgers University

Proportional reasoning across formats

Introduction

Comparing proportions is sometimes very hard! But, even infants seem to be able to do it a little bit. The purpose of this science project was to better understand how well people compare proportions when the proportions are presented in different formats. The purpose of this class assignment is to take the R-code and plots we've been generating over the last several weeks and put it all together into one poster format.

Research Objectives

1.Does average performance vary across format type?

2.Does average performance vary across numerator congruency status?

3.Does numerator congruency vary across format type?(ie., is there an interaction)

Methods

Participants

A total of 99 adults participated in the study.

First participants were introduced to a story about a magic ball and that the outcome(ie., blue or orange) depended on the proportions. They were then asked to compare the proportions of different images.

In other words, participants were shown two images of the same kind at the same time and asked to decided which had a higher proportion of the shape (or dots) colored in blue.

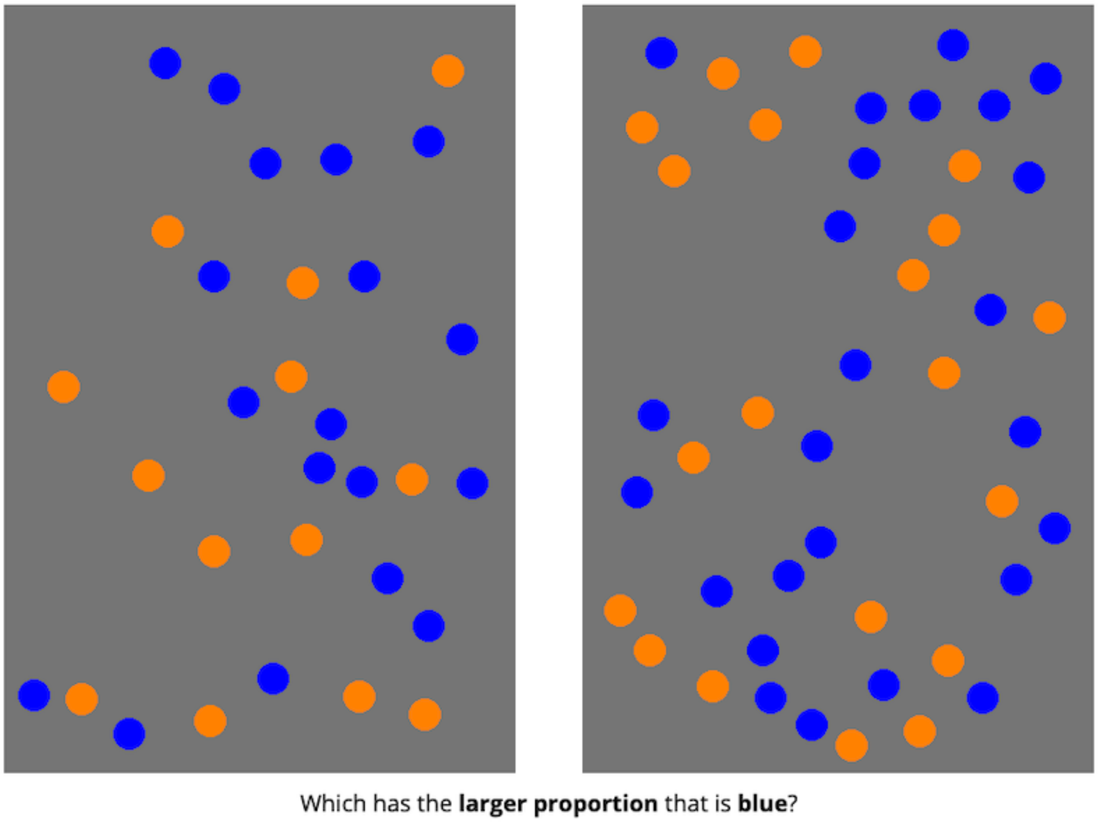

Figure 1. An example of integrated blobs

Figure 1 shows a clear example of what the “mixed” example of dots would look like to a participant.

Conditions. There were four different conditions that changed what kinds of images the participants saw:

- divided blobs: blur and orange were entirely separate.
- integrated blob: one blob, divided to be part blue and part orange.
- separated dots: blue and orange dots were on opposite sides of the image.
- integrated dots: blue and orange dots were intermixed.

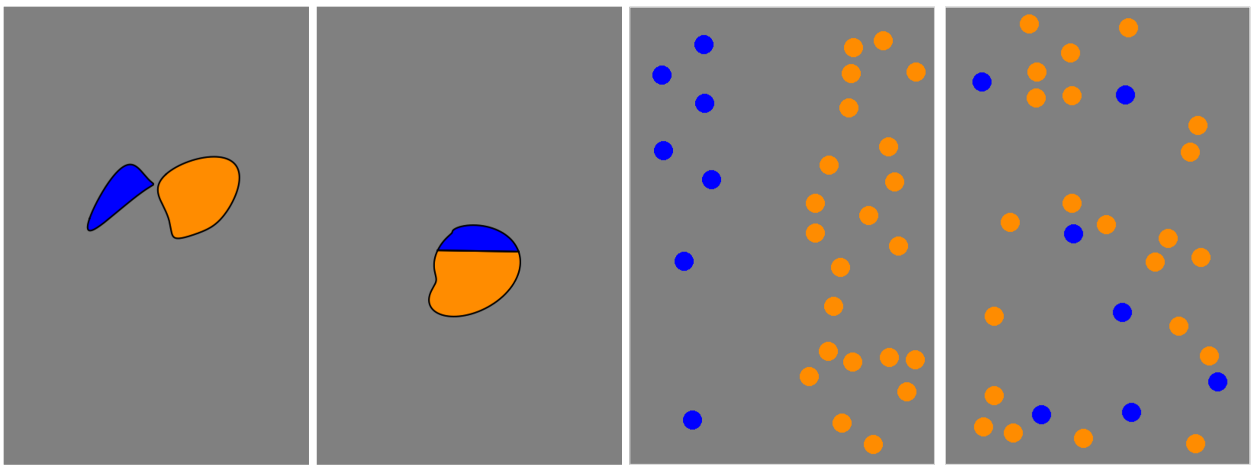

Figure 2. A formats in the study

Figure 2 shows the following from left to right : divided blobs, integrated blobs, separated dots, integrated dots.

Results

1. Does average performance vary across format type, ignoring all other aspects of the stimuli?

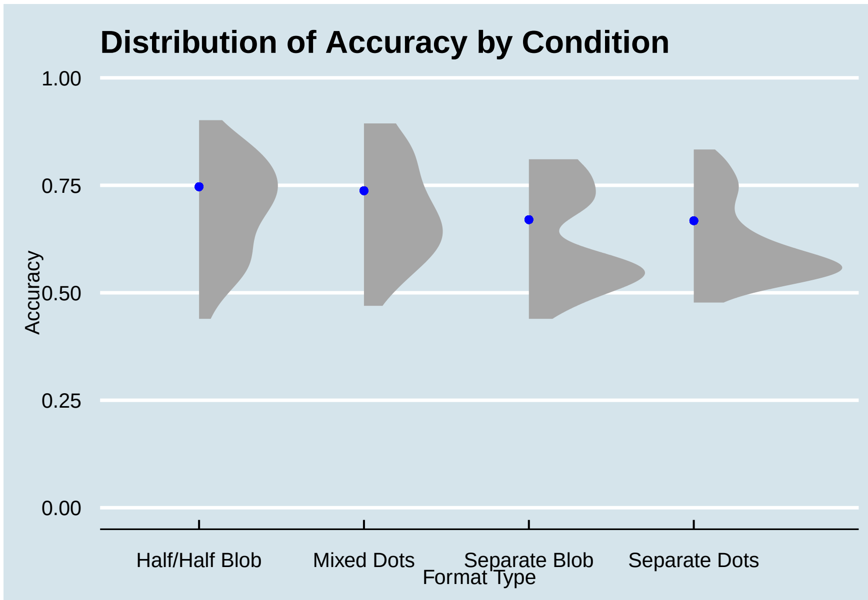

Figure 3. Plot of Distribution of Accuracy by Condition

Yes, the blue dots in Figure 3 prove that the average performance varies across formatting types.

2. How are reaction time and accuracy related?

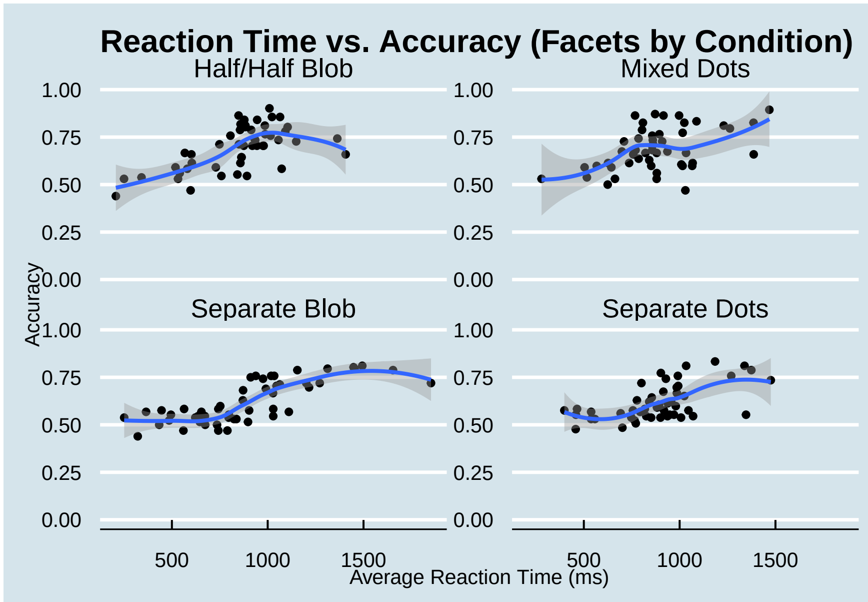

Figure 4. Plot of Distribution of Accuracy by Condition

As time increases along the x axis in Figure 4, so does the accuracy for each of the format types, until the time reaches about 1250 milliseconds.

3. How does numerator congruency interact with format type?

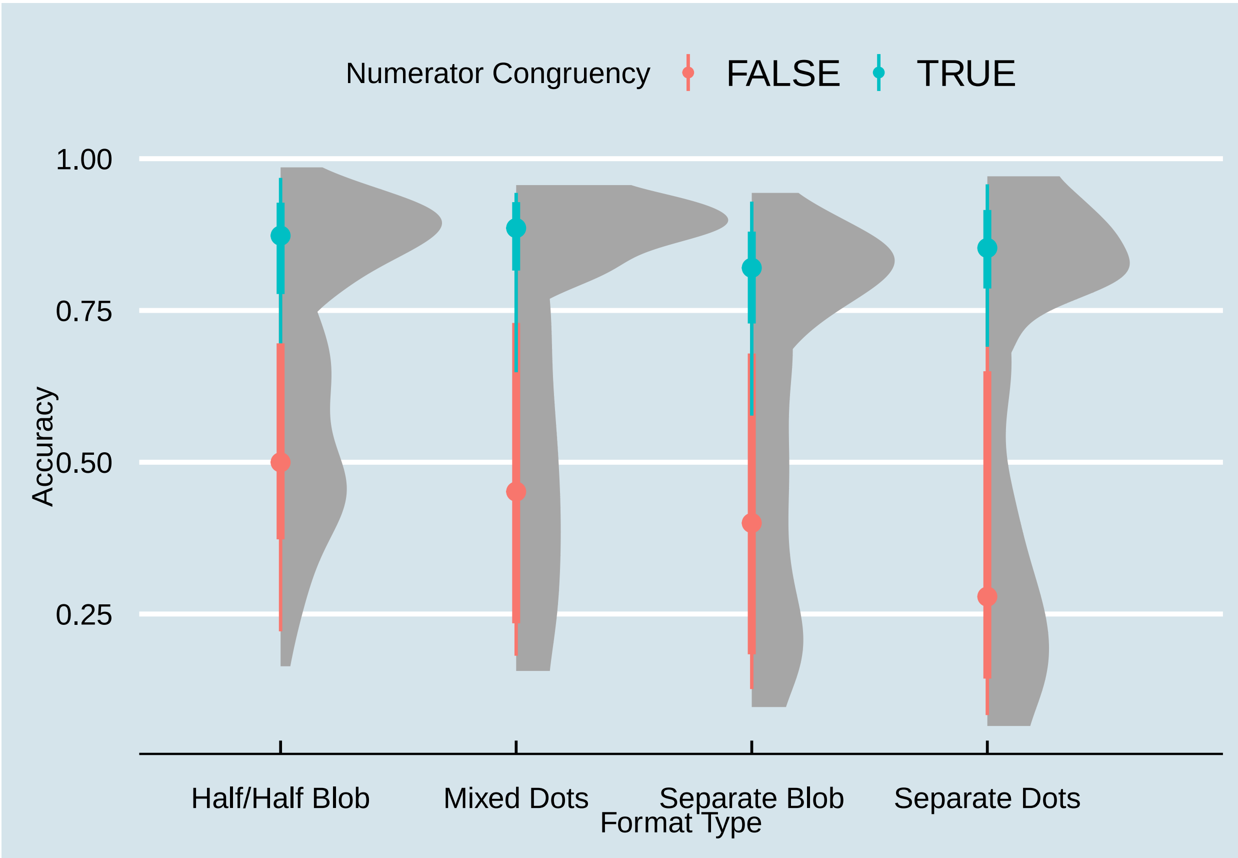

Figure 5

Numerator Congruency as shown in Figure 5 is only marked TRUE for Accuracy rates above about .60 according to the graph, across all format types. The largest range of TRUE numerator congruency is in blob_stacked format type than any other formatting.

Interpretation

Out of the three plots that we took a look at, here are the main ideas.Average performance does vary across different formats, when we ignore all aspects of stimuli. But, we analyze how stimuli was affected by formatting type in the Numerator Congruency Plot. Where, we concluded that higher accuracy rates all follow a similar pattern where Numerator

Congruency is TRUE above Accuracy rates of about .60. Finally, in our second plot, it is clear to see that accuracy follows a steady increase up until about 1250-1500 millisecond mark where it then dips.

Conclusion

What was the most annoying or hardest thing about this assignment?

- I had previously forgot to finish re-fixing my rstudio code from previous assignments, so debugging and then reproducing them here in this poster was a bit annoying. But, I will say that although it was annoying, I was able to learn more about the data by playing around withe the plot types.

What was the most satisfying or fun thing about the assignment?

- The most satisfying part was probably the first couple of questions of the assignments, as I was flying through them without having to flip constantly back and forth between the rendering and R-Studio to see if everything was working, because it was so straight-forward. I also had fun using Figure Labeling and Captioning, as it made referencing, and naming (for readers) a lot clearer, so that they can understand what was going on in my plots/figures.

Data analysis

I used R (Version 4.4.1; R Core Team, 2024) and the R-packages *dplyr* (Version 1.1.4; Wickham, François, Henry, Müller, & Vaughan, 2023), *forcats* (Version 1.0.0; Wickham, 2023a), *ggdist* (Version 3.3.2; Kay, 2024), *ggplot2* (Version 3.5.1; Wickham, 2016), *ggthemes* (Version 5.1.0; Arnold, 2024), *lubridate* (Version 1.9.3; Grolemund & Wickham, 2011), *papaja* (Version 0.1.3; Aust & Barth, 2024), *purrr* (Version 1.0.2; Wickham & Henry, 2023), *readr*

(Version 2.1.5; Wickham, Hester, & Bryan, 2024), *stringr* (Version 1.5.1; Wickham, 2023b), *tibble* (Version 3.2.1; Müller & Wickham, 2023), *tidyr* (Version 1.3.1; Wickham, Vaughan, & Girlich, 2024), *tidyverse* (Version 2.0.0; Wickham et al., 2019) and *tinylabels* (Version 0.2.4; Barth, 2023) for the entirety of this assignment.

Arnold, J. B. (2024). *Ggthemes: Extra themes, scales and geoms for 'ggplot2'.* Retrieved from https://CRAN.R-project.org/package=ggthemes

Aust, F., & Barth, M. (2024). *papaja: Prepare reproducible APA journal articles with R Markdown.* https://doi.org/10.32614/CRAN.package.papaja

Barth, M. (2023). *tinylabels: Lightweight variable labels.* Retrieved from https://cran.r-project.org/package=tinylabels

Grolemund, G., & Wickham, H. (2011). Dates and times made easy with lubridate. *Journal of Statistical Software*, *40*(3), 1–25. Retrieved from https://www.jstatsoft.org/v40/i03/

Kay, M. (2024). ggdist: Visualizations of distributions and uncertainty in the grammar of graphics. *IEEE Transactions on Visualization and Computer Graphics*, *30*(1), 414–424. https://doi.org/10.1109/TVCG.2023.3327195

Müller, K., & Wickham, H. (2023). *Tibble: Simple data frames.* Retrieved from https://CRAN.R-project.org/package=tibble

R Core Team. (2024). *R: A language and environment for statistical computing.* Vienna, Austria: R Foundation for Statistical Computing. Retrieved from https://www.R-project.org/

Wickham, H. (2016). *ggplot2: Elegant graphics for data analysis.* Springer-Verlag New York. Retrieved from https://ggplot2.tidyverse.org

Wickham, H. (2023a). *Forcats: Tools for working with categorical variables (factors).* Retrieved from https://CRAN.R-project.org/package=forcats

Wickham, H. (2023b). *Stringr: Simple, consistent wrappers for common string operations.* Retrieved from https://CRAN.R-project.org/package=stringr

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., … Yutani,

H. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, *4*(43), 1686. https://doi.org/10.21105/joss.01686

Wickham, H., François, R., Henry, L., Müller, K., & Vaughan, D. (2023). *Dplyr: A grammar of data manipulation.* Retrieved from https://CRAN.R-project.org/package=dplyr

Wickham, H., & Henry, L. (2023). *Purrr: Functional programming tools.* Retrieved from https://CRAN.R-project.org/package=purrr

Wickham, H., Hester, J., & Bryan, J. (2024). *Readr: Read rectangular text data.* Retrieved from https://CRAN.R-project.org/package=readr

Wickham, H., Vaughan, D., & Girlich, M. (2024). *Tidyr: Tidy messy data.* Retrieved from https://CRAN.R-project.org/package=tidyr