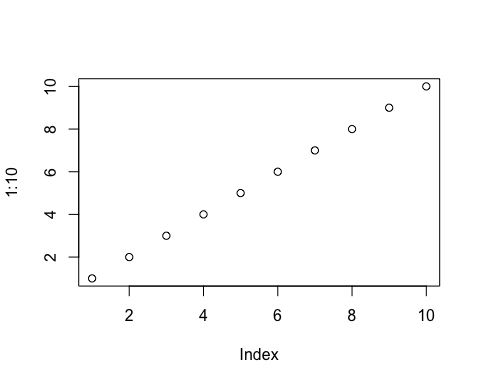
The Impact of Autism Spectrum Disorder On A Restless Bandit Task

Evan Knep

2024-05-04

## [1] 2



# **Reproducible Image and Online Resources**

An interactive web app is available by [clicking this link](https://knepx001.shinyapps.io/psy8712-final/) instead.

# **Script Settings and Resources**

setwd(dirname(rstudioapi::getActiveDocumentContext()$path))  
library(dplyr)  
library(stringr)  
library(readr)

# **Data Import and Cleaning**

file\_list <- list.files("../data") #my raw data is output as one .csv file per participant. I need to combine all ind files into a mega-file, so first I create a list of all files in the data directory  
file\_list <- paste("../data/", file\_list, sep = "") #pasting the path information onto the start of the filenames to avoid issues when reading all the files in  
spark\_detect <- str\_detect(file\_list, "SPARK") #the raw data was from two different rounds of collection, one through the psyREP program here at the U, and one through the SPARK for autism foundation. I can separate these groups by checking for the presence of the word spark in the filename.   
  
spark\_files <- file\_list[spark\_detect] #creating a list of all filenames that contain spark  
rep\_files <- file\_list[!spark\_detect] #creating a list of all filenames that do not contain spark (meaning they were collected through psyREP)  
  
spark\_master <- lapply(spark\_files, read\_csv) %>% #reading in each individual .csv in our spark files  
 bind\_rows() %>% #binding rows of each file to make one superfile  
 mutate(group = "spark") #I intend to merge the spark and rep master files, so before doing that I will label all the spark data by creating a new "group" column  
   
rep\_master <- lapply(rep\_files, read\_csv) %>% #reading in each individual .csv in our rep files  
 bind\_rows() %>% #binding rows of each file to make one superfile  
 mutate(group = "rep") #labeling this group as rep prior to merging with the spark data  
  
combined\_master <- bind\_rows(spark\_master, rep\_master) %>% #combining rep and spark datasets now that they've been labeled  
 filter(test\_part == "restless") %>% #each individual trial has a "test\_part" value associated with it, the rows that contain NA here are only saving server info in between trials so we can filter them out. In the spark sample we only used the restless bandit task, so we can filter out all the NA by only including rows that contain a restless value in the test\_part column.  
 select(-c(trial\_type, trial\_index, time\_elapsed, internal\_node\_id, participantCode, success, test\_part, generatedNum)) #removing irrelevant columns. Some of it is server info from Pavlovia.org, some it is simply redundant information that we've used for organizing files  
  
combined\_master <- combined\_master %>% #cleaning up the columns and ensuring the datatypes are correct  
 mutate(group = factor(group, ordered = FALSE)) %>% #factorizing my group variable, initially it was being treated as just a string   
 mutate(rt = as.numeric(rt)) %>% #changing my reaction time values to numeric  
 mutate(key\_press = case\_when( #for readability, changing the key\_press values from javascript keycode values to one that will make more sense to someone looking at the data. Each keypress (37,38,39) corresponds to the left,center,right arrow keys  
 key\_press == "37" ~ "left",  
 key\_press == "38" ~ "center",   
 key\_press == "39" ~ "right"  
 )) %>%  
 mutate(stimulus = sub(".\*/(.\*?)\\.jpg$", "\\1", stimulus)) %>% #pulling out specific stimulus identity from the path to the image file, grabbing the text in between the final / and the .jpg  
 mutate(chosen\_image = case\_when( # the stimulus file name grabbed in the previous mutate is named by the combination of the 3 images shown to the participant. So the first two characters refer to the left image, the middle two characters refer to the middle image, and the last two characters refer to the right image. I can look at image chosen by pulling those specific characters based on what the keypress was  
 key\_press == "left" ~ substring(stimulus, 1,2),  
 key\_press == "center" ~ substring(stimulus, 3,4),  
 key\_press == "right" ~ substring(stimulus, 5,6)  
 )) %>%  
 mutate(walkNumber = str\_remove(walkNumber, ".csv")) %>% #participants were given random "walks", that are the potential jumps in reward probability for each option. While they were generated to have the same environmental richness over the course of the task, the different walks could theoretically influence behaviors.  
 mutate(z\_score = scale(rt)) %>% #exploratory data analysis revealed some significant outliers in reaction time, because there was no limited hold on each trial. Calculating the z-score of the column and filtering out values above 3 or below -3 allows us to remove rts that are more than 3 standard deviations away from the mean  
 filter(abs(z\_score) <= 3) %>% # ^  
 rowwise() %>%  
 mutate(chance = sum(leftProb, midProb, rightProb) / 3) # looking at probability of reward of each participant can provide a skewed sense of performance, because the random component of the task means that different participants will be exposed to difference levels of chance. By calculating the chance each trial and averaging those values per participant (in analysis), we can determine each participants performance above chance