Assignment 1: Learning and Memory PSY 306 (Winter 2023)

Name: Utkarsh Arora

Roll Number: 2020143

Instructions: Please write your own responses and DO NOT copy or lift text/code from any source, including the attached paper. If you are referring to credible external sources other than the attached paper for your answers, please cite those sources (within the body of text and the provide a reference list at the end) in the APA citation format (https://www.mendeley.com/guides/apa-citation-guide). Word limits given are indicative and less than the indicated numbers may also be used.

Please download this MS word question-cum-response template to TYPE your answers and feel free to add sheets as required. Convert this document to a PDF and rename the file: name_roll no. before submitting. Please note that answers in this template only will be evaluated and hand-written or scanned answer sheets will not be evaluated. Verbatim copying of any extent and total percent similarity with other sources exceeding 20% will be deemed plagiarized and dealt as per IIITD policies.

[Strict deadline for submission: 22 Feb, 11 PM]

Q2) Please do the following for this question:

- **Register on PsyToolkit (https://www.psytoolkit.org/) & log in. References**
  Ref-1, Stoet, G. (2010). PsyToolkit - A software package for programming psychological experiments using Linux. Behavior Research Methods, 42(4), 1096-1104. Ref-2,Stoet, G. (2017). PsyToolkit: A novel web-based method for running online questionnaires and reaction-time experiments. Teaching of Psychology, 44(1), 24-31.
- **Click on ' Get from Library' on the left-hand side panel of the screen.**
- **On the central panel of the screen click on ' Official PsyToolkit experiment library' (https://www.psytoolkit.org/experiment-library/)**
- **Scroll down and click on 'N-back Task (2 back)'.**
- **Scroll down on this page (https://www.psytoolkit.org/experiment-library/nback2.html) & click under Download on 'The PsyToolkit code zip file'.**
- **Follow the instructions in this video to compile the experiment from the downloaded zip file in the previous step : https://www.youtube.com/watch?v=VIf-UuLbi3Y&feature=youtu.be .**
- **Read the documentation of the experiment and the detailed instructions of running the experiment and the output data structure (https://www.psytoolkit.org/experiment-library/nback2.html).**
- **Run the experiment either 'in the browser' or download the compiled experiment offline by clicking on 'Download for Running Offline'**
- **PLEASE CARRY OUT ALL 75 TEST TRIALS given in three blocks of 25 trials each .**
- **At the end, a table of results will be displayed and the column headers of the results table are here** https://www.psytoolkit.org/experiment-library/nback2.html **under 'Data output file'.**
- **Download the results table (text file ('.txt')) and then answer the following…**

Insert a figure (wherever required) and paste the MATLAB/Python code for the same. The datasheet generated from the test trials may also be pasted on this sheet at appropriate places. All figures must be properly labelled and should have accompanying captions/legends to provide all information necessary to interpret the figures…

A) Which cognitive process does the test measure. Briefly explain how? [1+3]

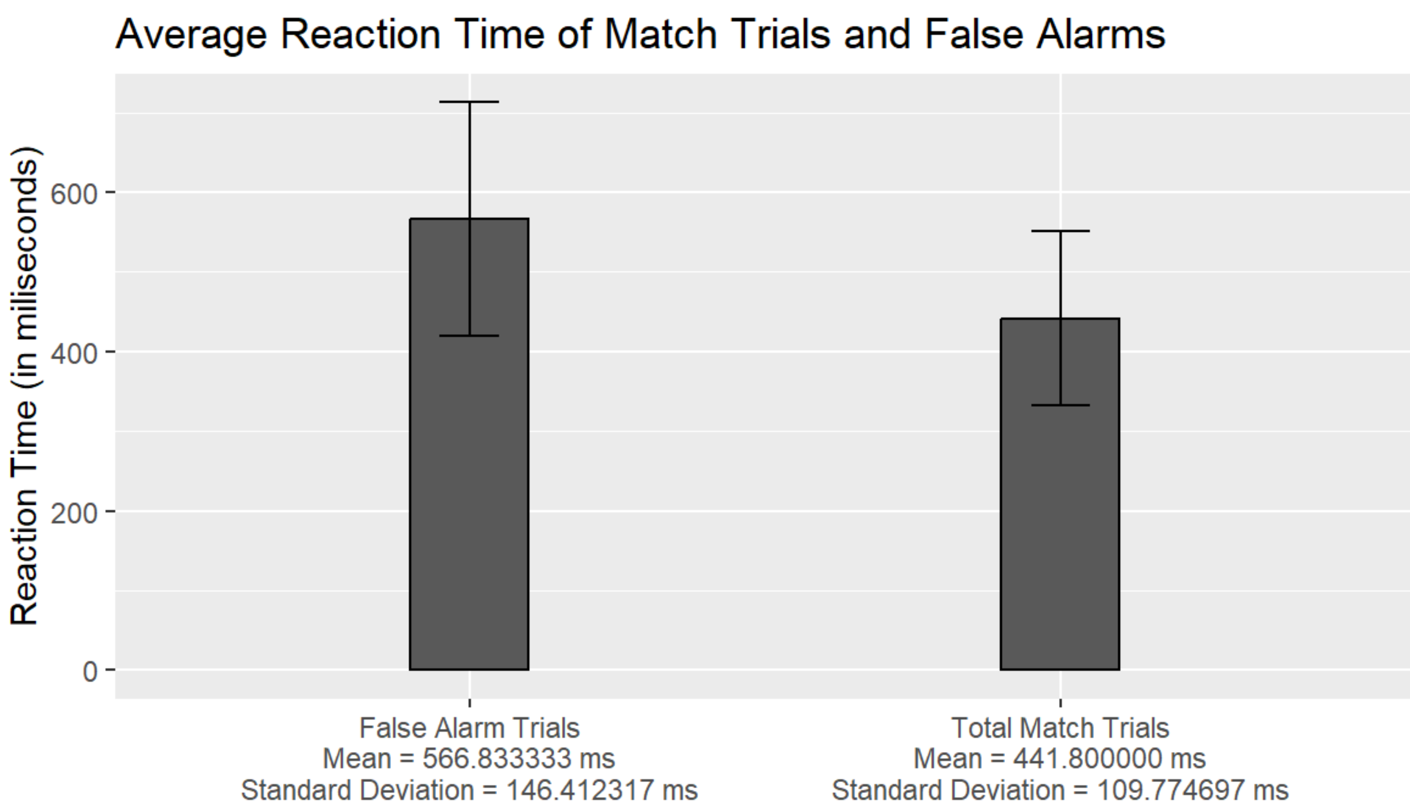
The N-back test is a cognitive task that measures working memory, specifically, it evaluates an individual's ability to hold information in short-term memory and manipulate it.

During the N-back task, the participant is presented with a sequence of stimuli, typically letters or numbers, and is asked to respond whether the current stimulus matches the one that was presented N trials ago. For example, in a 2-back task, the participant would respond "yes" if the current stimulus matches the one presented two trials ago, and "no" otherwise.

The task measures working memory by challenging the participant to hold a sequence of items in memory and updating it with new information, while also inhibiting irrelevant stimuli. The task becomes more difficult as N increases because it requires the individual to hold a longer sequence of items in memory and manipulate it, which places greater demands on working memory.

Overall, the N-back task provides a measure of working memory capacity, updating ability, and inhibitory control.

B) Plot two simple bar diagrams showing the average reaction times with their standard deviations (error bars) of the total 'match' trials and the 'false alarm' trials respectively. [3+3]



<u>R code:</u>

```
library(ggplot2)


mydata <- read.table("C://Users//Utkarsh//Documents//homework//Learning and
Memory//data.ssv",header=FALSE)
```

```r
#Question 2 Part B

match_trials <- subset(mydata, mydata$V5 == 1)
false_alarm_trials <- subset(mydata, mydata$V7 == 1)

new_df <- data.frame(
  Category = c(sprintf("Total Match Trials\nMean = %f ms\nStandard Deviation = %f
ms",mean(match_trials$V8),sd(match_trials$V8)),
               sprintf("False Alarm Trials\nMean = %f ms\nStandard Deviation = %f
ms",mean(false_alarm_trials$V8),sd(false_alarm_trials$V8))),
  Means = c(mean(match_trials$V8),mean(false_alarm_trials$V8)),
  SDs = c(sd(match_trials$V8),sd(false_alarm_trials$V8))
)

ggplot(new_df,
        aes(x = Category, y = Means)) +
  geom_bar(position = position_dodge(), stat = "identity", colour = "black", width = 0.2) +
  geom_errorbar(aes(ymin = Means - SDs, ymax = Means + SDs), width = .1) +
  labs(title = "Average Reaction Time of Match Trials and False Alarms") +
  ylab("Reaction Time (in miliseconds)")
```

C) Calculate the mean (M) reaction time of all trials. Split the original data into two parts such that all the trials of one part has reaction time less than M and the trials of the other part has reaction time equal to or greater than M. Next, calculate the total number of erroneous responses (i.e., incorrect responses, false alarms, misses, in match and non-match trials as relevant) and express that as a percentage of the total trial number for both parts of the split data. Report both percentages.

Based on a comparison of the above two percentages, what can be concluded about the relationship between response accuracy and reaction time in your experimental data? [9+1]

(Hint: report all steps; conclude based on your own data + analysis)

The mean reaction of all trials is 2123.16 ms

The dataset has been split in two parts, one where the reaction time is less than 2123.16 ms, and the other where the reaction time is greater than 2123.16 ms.

The error rate is calculated as (1-(total number of successful trials/total number of trials))*100

The error rate in the dataset where the reaction time is lesser is 23.07%
The error rate in the dataset where the reaction time is greater is 6.12%

On running a linear regression we can see that there is a positive correlation between the score and the reaction time.

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 7.494e-01  7.497e-02   9.996 2.59e-15 ***
mydata$V8   6.149e-05  3.071e-05   2.003   0.0489 *  
```

Therefore, this would mean that the higher the reaction time, there is a higher success rate.

However, there is one issue with our data. There is an overrepresentation of non-matching trials. To be successful in a non-matching trial, the user must not press any button. This means that the reaction time for success in that trial is 3000 ms. Additionally, there is an overrepresentation of non-matching trials, which takes the average reaction time for success above.

To account for this overrepresentation, we are going to consider only trials where the user has pressed the button. This means that the trials where the reaction time is 3000 ms is excluded from our further analysis. This means that the misses are excluded as well. Only correct hits, and false alarms are included i.e., only trials where the user has pressed the button.

Let us run the analysis on this dataset now.
The mean of all hits is 470.6538 ms.

The error rate when the reaction time is lower than the local mean is 11.76%
The error rate when the reaction time is higher than the local mean is 44.44%

```
> summary(lm(hits$V4~hits$V8))

Call:
lm(formula = hits$V4 ~ hits$V8)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.91432 -0.00124  0.11253  0.20299  0.52100 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.4344431  0.3032908    4.73 8.26e-05 ***
hits$V8     -0.0014134  0.0006227   -2.27   0.0325 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3979 on 24 degrees of freedom
Multiple R-squared:  0.1767,	Adjusted R-squared:  0.1424 
F-statistic: 5.152 on 1 and 24 DF,  p-value: 0.03249
```

On running linear regression, we can see that there is a negative correlation between the success rate and reaction time.

Hence, we can see that with a higher response time, there is a lower success rate.

This analysis would imply that the user reacts faster on correct hits, and slower on false positives. Psychologically, this would imply that I responded faster when I was sure of the answer, and slower when I was unsure of the answer, and I made more errors when I was unsure of the answer.
This phenomenon can be explained by cognitive processes such as uncertainty and decision-making. When a participant is unsure about the correct answer, they may take longer to respond because they need more time to process the information and make a decision. In the context of a 2-back task, when a participant has to determine whether the current stimulus matches the one presented two trials ago, they may not always be certain about the correct answer. If they are unsure, they may take longer to evaluate their options and make a decision, which can lead to slower reaction times. Additionally, if they are unsure and take longer to respond, they may be more likely to make an error due to lapses in attention or working memory.

This is the R code I have run for this part. It also explains the steps:

```
#Question 2 Part C


#Calculating the mean reaction time of the entire dataset
```

```r
M <- mean(mydata$V8)

#Subsetting the entire dataset into two parts, one where the reaction time is
#lesser than the mean, and the other where the reaction time is greater
lesser <- subset(mydata, mydata$V8<M)
greater <- subset(mydata, mydata$V8>M)

#Calculating the error rate of both the subset datasets
errorness_lesser <- (1-(sum(lesser$V4)/nrow(lesser))) * 100
errorness_greater <- (1-(sum(greater$V4)/nrow(greater))) * 100


#Subsetting the dataset into a dataset which only has reaction times not 3000ms
hits <- subset(mydata, mydata$V8 < 3000)

#Calculating the mean of this new dataset
M2 <- mean(hits$V8)

#Subsetting this new dataset into two datasets based on the mean
lesser_hits <- subset(hits, hits$V8<M2)
greater_hits <- subset(hits, hits$V8>M2)

#Calculating the error rate of these two datasets
errorness_lesser_hits <- (1-(sum(lesser_hits$V4)/nrow(lesser_hits))) * 100
errorness_greater_hits <- (1-(sum(greater_hits$V4)/nrow(greater_hits))) * 100

#Running linear regression
summary(lm(hits$V4~hits$V8))
```