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

Name: Shagun Mohta

Roll Number: 2020468

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 labeled and should have accompanying captions/legends to provide all information necessary to interpret the figures...

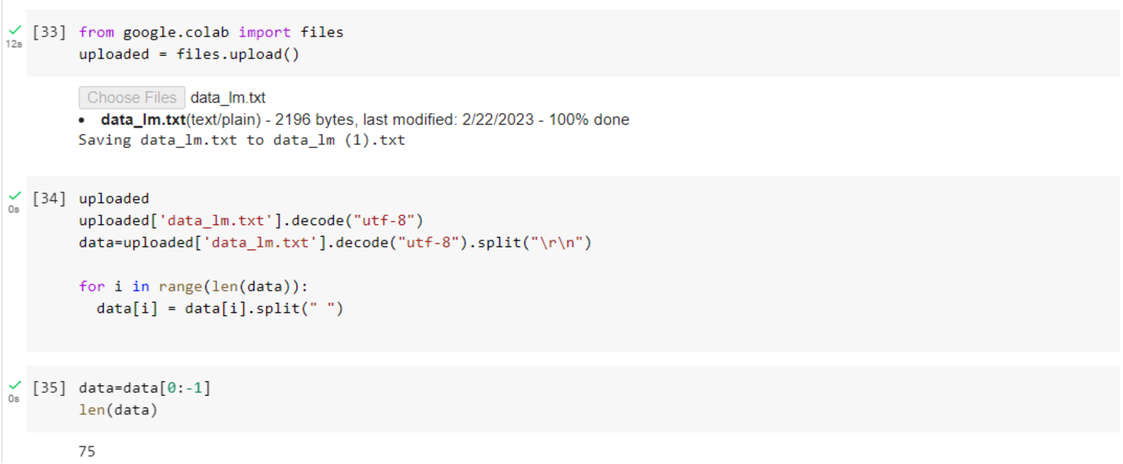
A) Which cognitive process does the test measure. Briefly explain how? [illegible]

ANS:

Individuals' **short term memory capacity** is effectively tested using the n(n=2) back test (*N-back - Wikipedia*, 2018) . In the 2 back task, participants must report whether the current letter on the screen, which is the stimulus, matches the one presented two letters back. Participants have to constantly engage, attend, concentrate, recall and act while the stimulus is presented for a limited duration of time. While **Active Recall** and **Attention** are the most prominent cognitive processes being tested ,this is also a good way to assess **cognitive flexibility**.

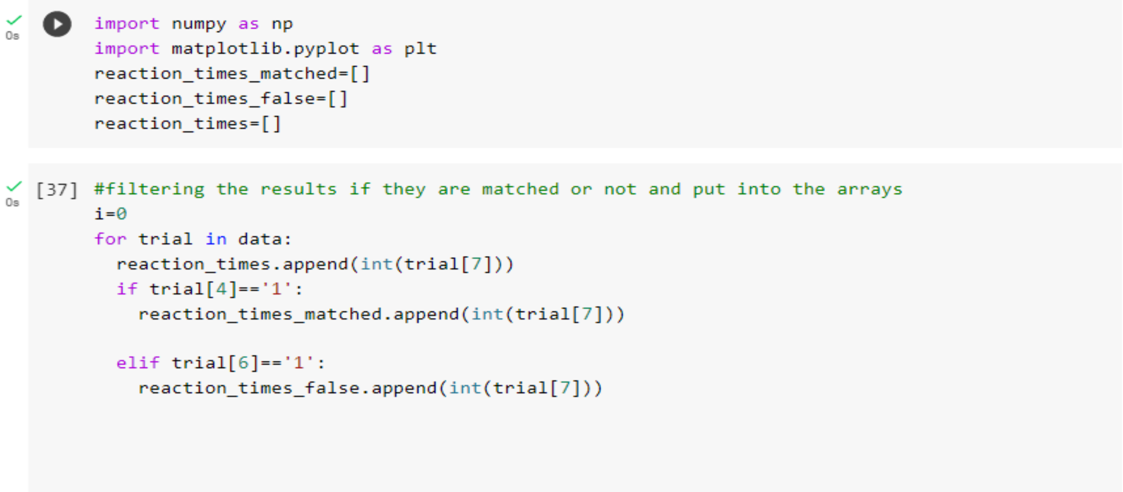
B) Plot two simple bar diagrams showing the average reaction times with their standard deviations (error bars) of the total ‘match’ trials and the ‘false alarm’ trials respectively.[3]

ANS:



```
[33] from google.colab import files
     uploaded = files.upload()

     Choose Files data_lm.txt
     • data_lm.txt(text/plain) - 2196 bytes, last modified: 2/22/2023 - 100% done
     Saving data_lm.txt to data_lm (1).txt

[34] uploaded
     uploaded['data_lm.txt'].decode("utf-8")
     data=uploaded['data_lm.txt'].decode("utf-8").split("\r\n")

     for i in range(len(data)):
       data[i] = data[i].split(" ")

[35] data=data[0:-1]
     len(data)

     75
```

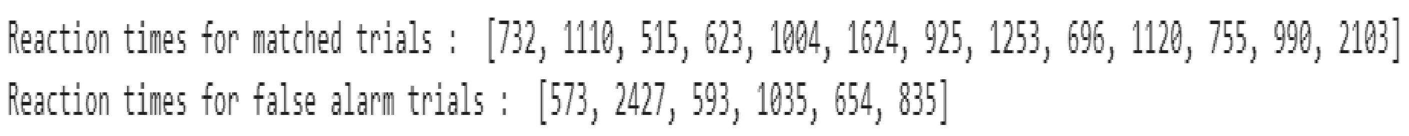
Step1: The Dataset is uploaded in the form of a text file and converted into an array of all trial details.



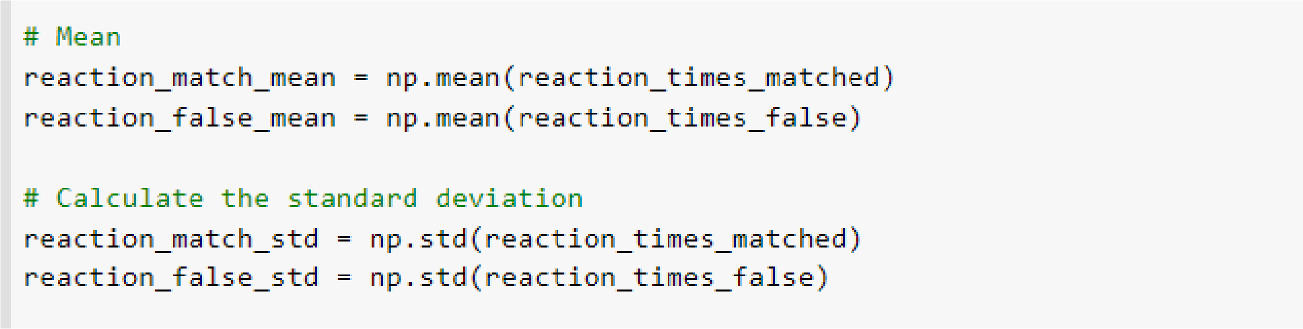
```
import numpy as np
import matplotlib.pyplot as plt
reaction_times_matched=[]
reaction_times_false=[]
reaction_times=[]

[37] #filtering the results if they are matched or not and put into the arrays
     i=0
     for trial in data:
       reaction_times.append(int(trial[7]))
       if trial[4]=='1':
         reaction_times_matched.append(int(trial[7]))

       elif trial[6]=='1':
         reaction_times_false.append(int(trial[7]))
```

Step 2: Each trial is parsed to extract the reaction times of the trials into trials with Matched responses and trials with False alarm responses by checking the column values for MAtched and False Alarms if they are 1.

The image captures the 2 lists that were created, ie reaction_times_matched and reaction_times_false



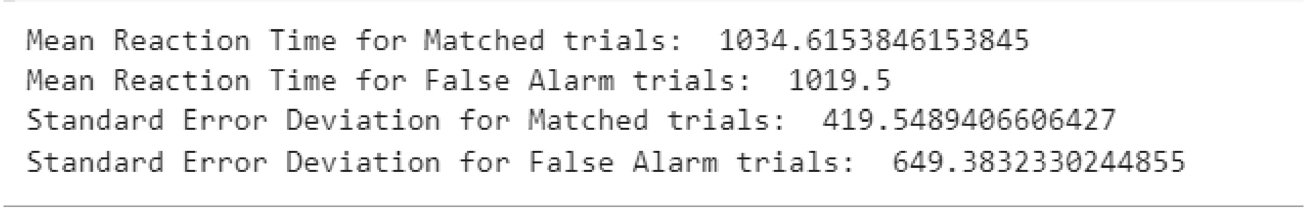
```
Reaction times for matched trials :  [732, 1110, 515, 623, 1004, 1624, 925, 1253, 696, 1120, 755, 990, 2103]
Reaction times for false alarm trials :  [573, 2427, 593, 1035, 654, 835]
```



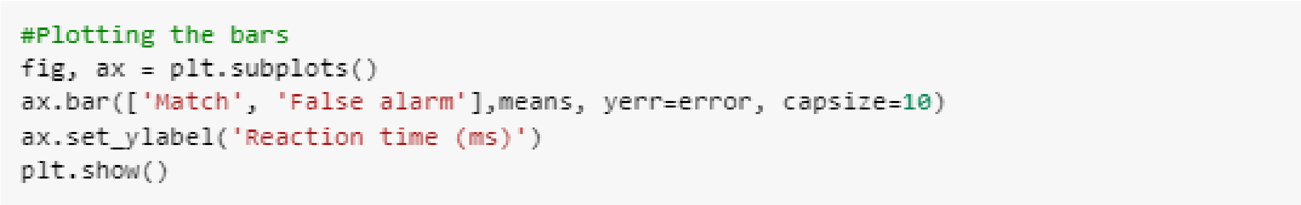
```
# Mean
reaction_match_mean = np.mean(reaction_times_matched)
reaction_false_mean = np.mean(reaction_times_false)

# Calculate the standard deviation
reaction_match_std = np.std(reaction_times_matched)
reaction_false_std = np.std(reaction_times_false)
```

Step 3: Using numpy library in python, means of the lists and standard deviations are calculated.

The following results were obtained for the trials:



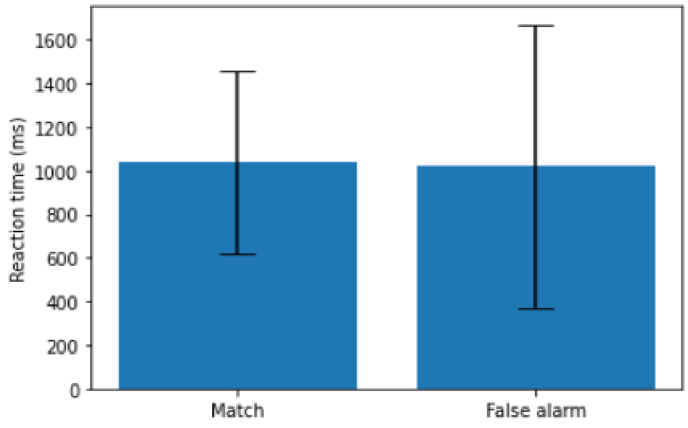
```
Mean Reaction Time for Matched trials:  1034.6153846153845
Mean Reaction Time for False Alarm trials:  1019.5
Standard Error Deviation for Matched trials:  419.5489406606427
Standard Error Deviation for False Alarm trials:  649.3832330244855
```



```
#Plotting the bars
fig, ax = plt.subplots()
ax.bar(['Match', 'False alarm'],means, yerr=error, capsize=10)
ax.set_ylabel('Reaction time (ms)')
plt.show()
```

Step4: Using python's matplotlib library, the bar graph was plotted.

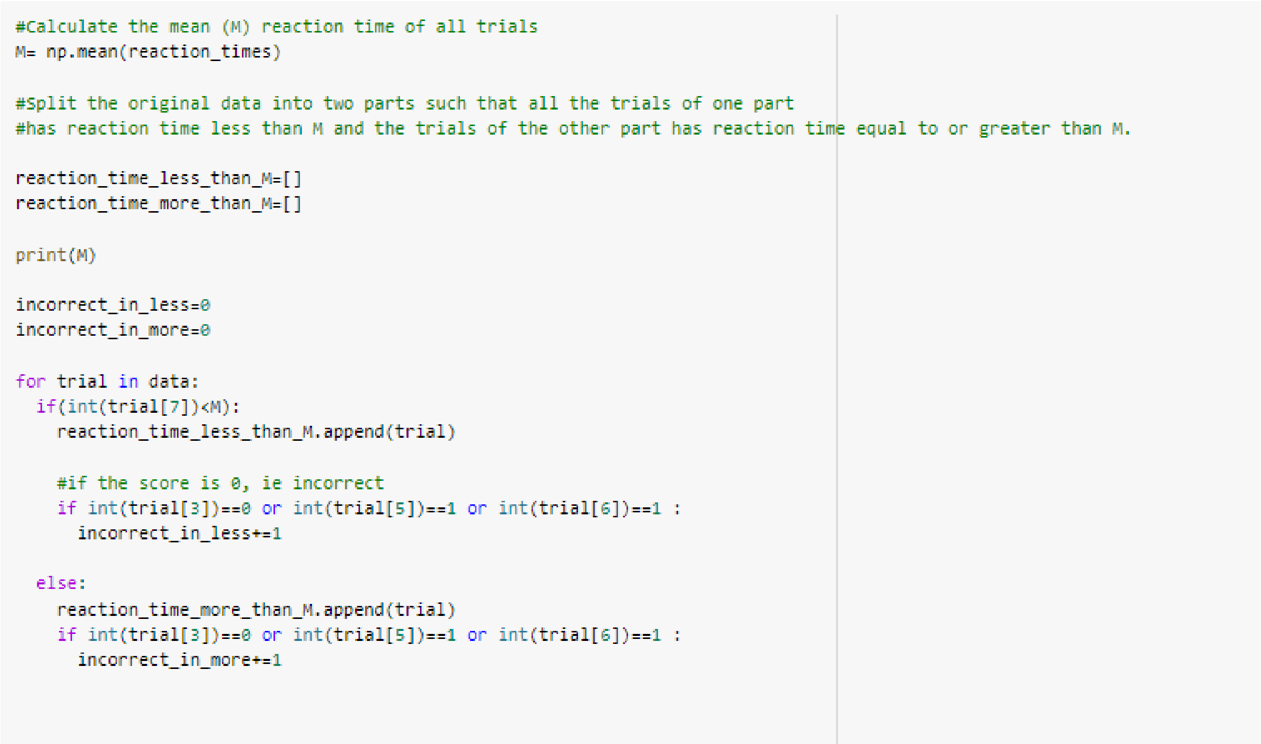
The following bar plot was obtained for the Matched and False Alarm trials:



The vertical bars represent the Error in both the cases.
The heights of the error bars and standard deviations are determined by yerr=error.

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?

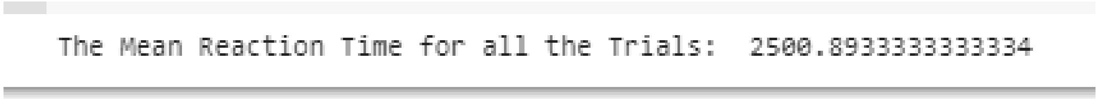
ANS:



```
#Calculate the mean (M) reaction time of all trials
M= np.mean(reaction_times)

#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.

reaction_time_less_than_M=[]
reaction_time_more_than_M=[]

print(M)

incorrect_in_less=0
incorrect_in_more=0

for trial in data:
  if(int(trial[7])<M):
    reaction_time_less_than_M.append(trial)

    #if the score is 0, ie incorrect
    if int(trial[3])==0 or int(trial[5])==1 or int(trial[6])==1 :
      incorrect_in_less+=1

  else:
    reaction_time_more_than_M.append(trial)
    if int(trial[3])==0 or int(trial[5])==1 or int(trial[6])==1 :
      incorrect_in_more+=1
```

Step1: reaction_times is the list of reaction times for all the trials. Using numpy library, the mean reaction time has been calculated.

The Mean Reaction Time for the trials :

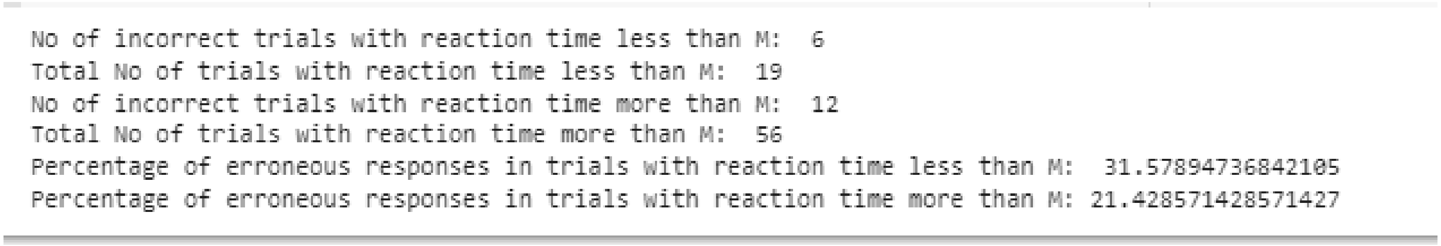


```
The Mean Reaction Time for all the Trials:  2500.8933333333334
```

Next, two lists are created that contain trials with reaction times less than the mean and trials with reaction times greater than the means separately.

Step2: Two counters are used to measure the number of erroneous responses in both the lists that have been split from the original data. In both the lists, incorrect or miss or false alarms in both the match and unmatched trials are measured by checking the conditions.

The final percentages are calculated by dividing the number of incorrect responses in one split divided by the total number of trials in that split and multiplying the result by 100.



```
No of incorrect trials with reaction time less than M:  6
Total No of trials with reaction time less than M:  19
No of incorrect trials with reaction time more than M:  12
Total No of trials with reaction time more than M:  56
Percentage of erroneous responses in trials with reaction time less than M:  31.57894736842105
Percentage of erroneous responses in trials with reaction time more than M: 21.428571428571427
```

Observation:
When the reaction time is lower than the average M, we observe that the incorrect or erroneous trial percentage is more than for the trials where reaction time is less than the average.

Conclusion
From the observation,it can be interpreted that as the response time increases, the ratio of erroneous trials decreases, hence Response accuracy increases. Thus, Response time is directly proportional to Response Accuracy.

References

1. *Ref-1, Stoet, G. (2010). PsyToolkit - A software package for programming psychological experiments using Linux. Behavior Research Methods, 42(4), 1096-1104.*

2. *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.*

3. *Ref-3, n-back - Wikipedia*. (2018, November 10). N-back - Wikipedia. https://en.wikipedia.org/wiki/N-back#:~:text=The%20n%20%2Dback%20task%20is,memory%20and%20working%20memory%20capacity.

4. Ref-4, Kazarinoff, P. D. (2018, January 7). *Bar charts with error bars using Python and matplotlib*. Python for Undergraduate Engineers. https://./python-matplotlib-error-bars.html