Assignment 1: Neuroscience of Decision Making PSY 307 (Monsoon 2024)

Name: Nimit Panwar

Roll Number: 2022324

Instructions: Please write your own responses and do not copy or lift text/code from any source (including the 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_RollNo. before submitting. Please note that answers in this template only will be evaluated and hand-written or scanned answer sheets will not be evaluated.

[Strict deadline for submission: 21 September, Saturday, 11:00 PM]

---

1. **Fill out the google form: https://forms.gle/a23if8Kbdjs2gRfJ8**

2. **An experimenter recorded a SINGLE cortical neuron's activity from two brain regions- H & A- processing visual information as the organism viewed three distinct visual stimuli on the screen. Each stimulus was presented and its neuronal activity was recorded for 30 times. The collected datasets are attached herewith: 'dataset_H.mat' and 'dataset_A.mat'.**

   dataset_H.mat: Neuronal activity recorded from region H.

   dataset_A.mat: Neuronal activity recorded from region A.

   Each dataset consists of 3 columns/cells:

   Column 1: Timestamps of action potentials for the face stimulus.

   Column 2: Timestamps of action potentials for the text stimulus.

   Column 3: Timestamps of action potentials for the speech stimulus.

   Each column contains 30 trials and the array size for each trial is different. The timestamps are associated to the occurrence of each action potential, measured in milliseconds from stimulus onset. For example: In the dataset_A, the first trial has 17 timestamps meaning, action potential/spike was recorded at these 17 timestamps. A negative timestamp (e.g.: -900) means action potential/spike that occurred 900 ms before stimulus onset time (0 ms). Similarly, a positive timestamp (e.g.: 300) indicates spikes that occurred after stimulus onset (0 ms).

Total neuronal recording duration = 3000 milliseconds; Time = 0 millisecond (stimulus onset); Time = 1000 milliseconds (stimulus offset)

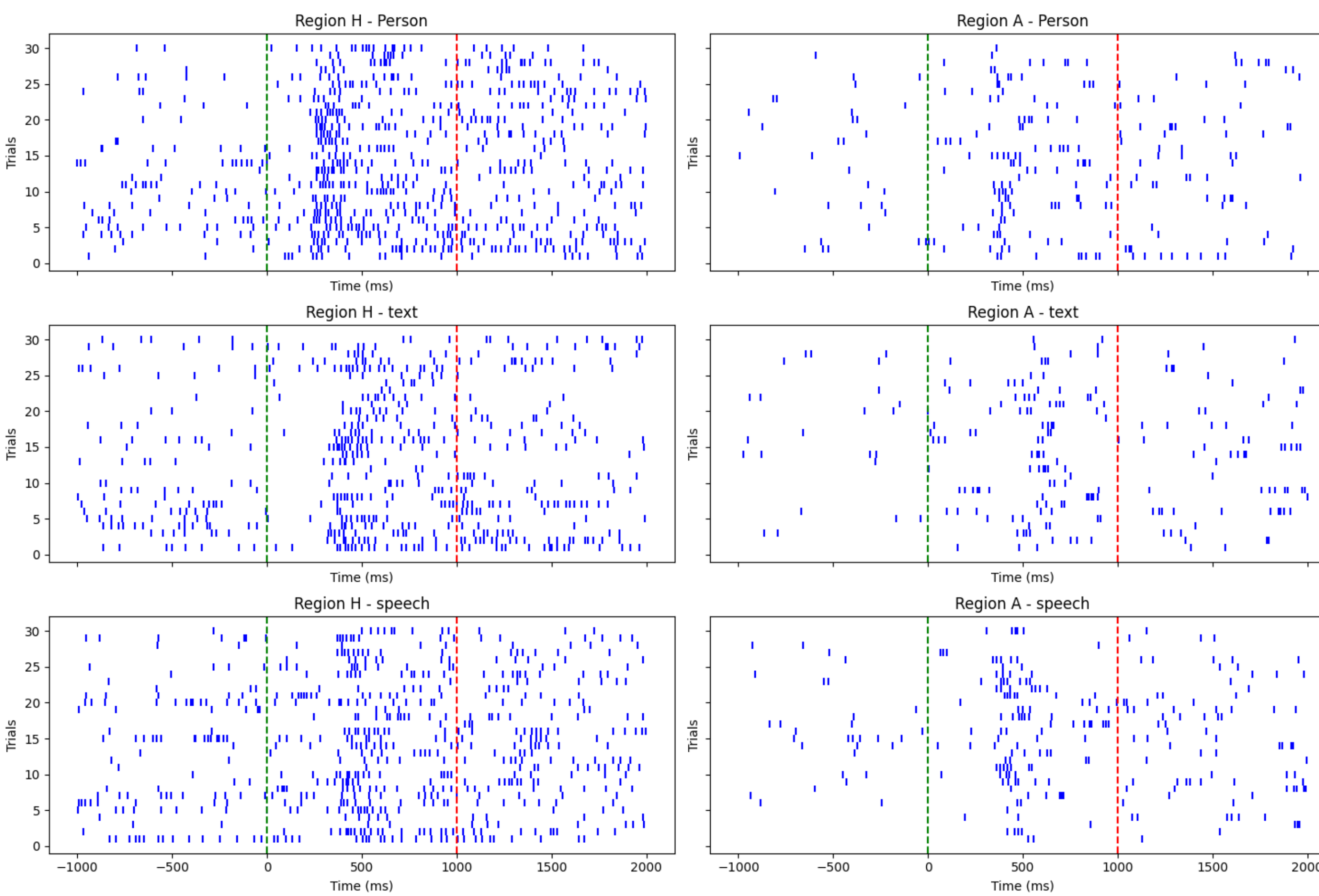
Now solve the following. Insert a figure (wherever required) and paste the MATLAB/ Python/R code for the same. Any figure must provide all information necessary to interpret it including axes labels, captions/legends <u>(see Fig.3 of the attached paper for a sample; simple figure titles as captions are not enough)</u>.

[link for importing MATLAB data arrays into Python and R]

- **https://www.askpython.com/python/examples/mat-files-in-python**

A) Create a Raster plot of the two neurons recorded from two distinct brain regions for ALL SIX of the stimuli (2 X face; 2 X text; 2 X speech) and mark the onset of stimulus onset and offset of the stimulus by a vertical green and red line respectively on the same individual subplots. Mark the spikes (action potentials) with blue colour. [5 marks]

Hint: Create a larger figure with six subplots (positioned as 3 rows x 2 columns). Plot dataset_H in the first column and dataset_A in the second column of the larger figure. Indicate the stimulus type - face, text, speech on top of each subplot as subplot title

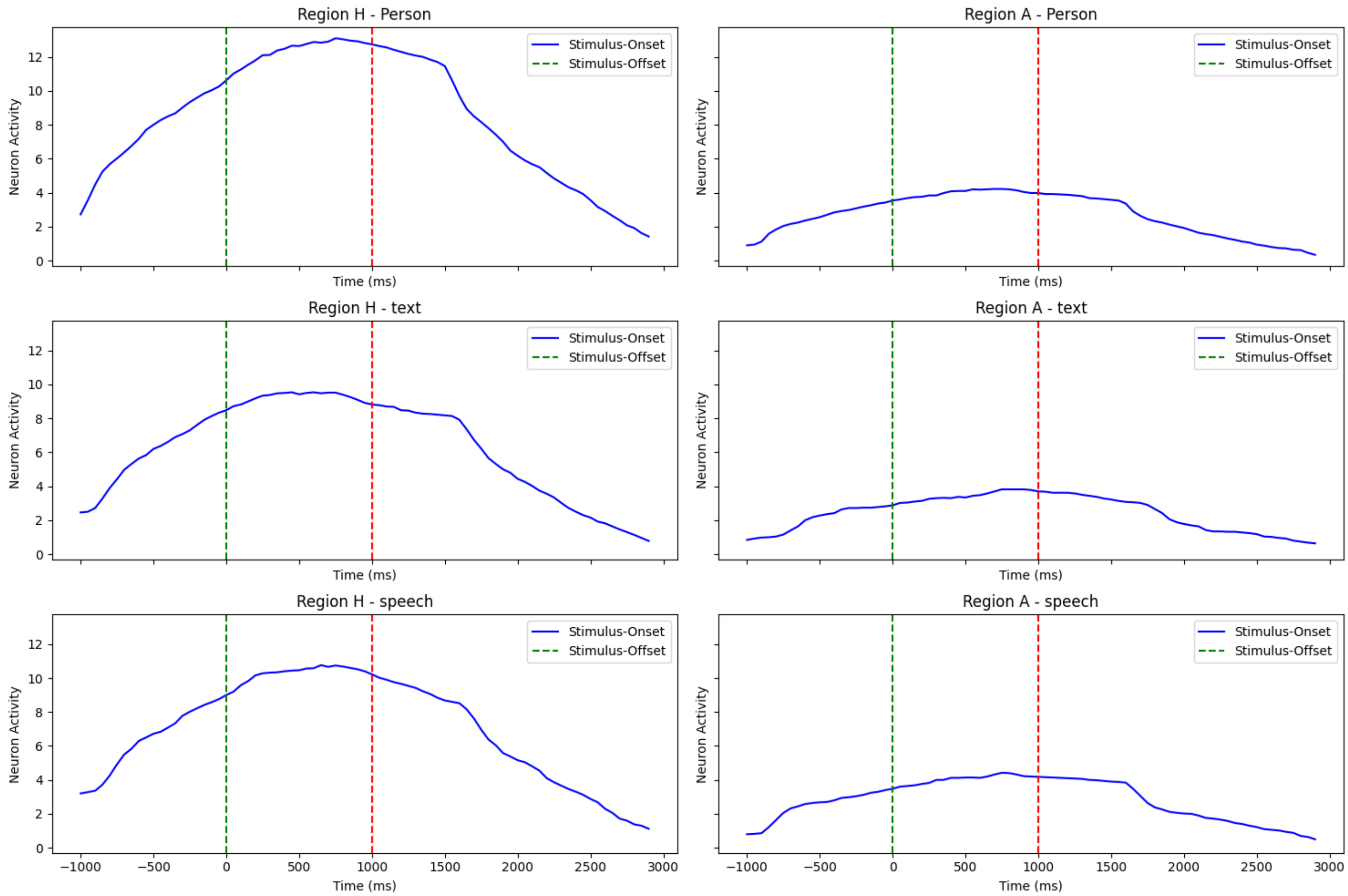


Code for Part - A :

```
import scipy.io
import matplotlib.pyplot as plt

# Load data
H = scipy.io.loadmat('dataset_H.mat')['dataset_H'][0]
A = scipy.io.loadmat('dataset_A.mat')['dataset_A'][0]
stimuli = ['Person', 'text', 'speech']

# Raster plot function
def Rastor(subplot, data, title):
    for i, j in enumerate(data):
        subplot.vlines(j[0], i + 0.5, i + 1.5, color='blue')
    subplot.axvline(0, color='green', linestyle='--')
    subplot.axvline(1000, color='red', linestyle='--')
    subplot.set_title(title)
    subplot.set_ylabel('Trials')
    subplot.set_xlabel('Time (ms)')

# Plot data
fig, subplot = plt.subplots(3, 2, figsize=(15, 10), sharex=True,
sharey=True)
for i, j in enumerate(stimuli):
    Rastor(subplot[i, 0], H[i], f'H - {j}')
    Rastor(subplot[i, 1], A[i], f'A - {j}')
plt.tight_layout()
plt.show()
```

B) Create a Peri Stimulus Time Plot of two neurons recorded from two distinct brain regions for ALL SIX of the stimuli and mark the onset of stimulus and offset of the stimulus by a vertical green and red line respectively on the same individual subplots. Before computing the histogram, smooth the data for each subplot over a time window of 50 ms. [5 marks]

Hint: Create a larger figure with six subplots (positioned as 3 rows x 2 columns); Smooth the data by moving average method; a line plot would suffice and depict the trend. Plot dataset_H in the first column and dataset_A in the second column of the larger figure. Indicate the stimulus type - person, text, speech on top of each subplot as subplot title



Code for Part-B

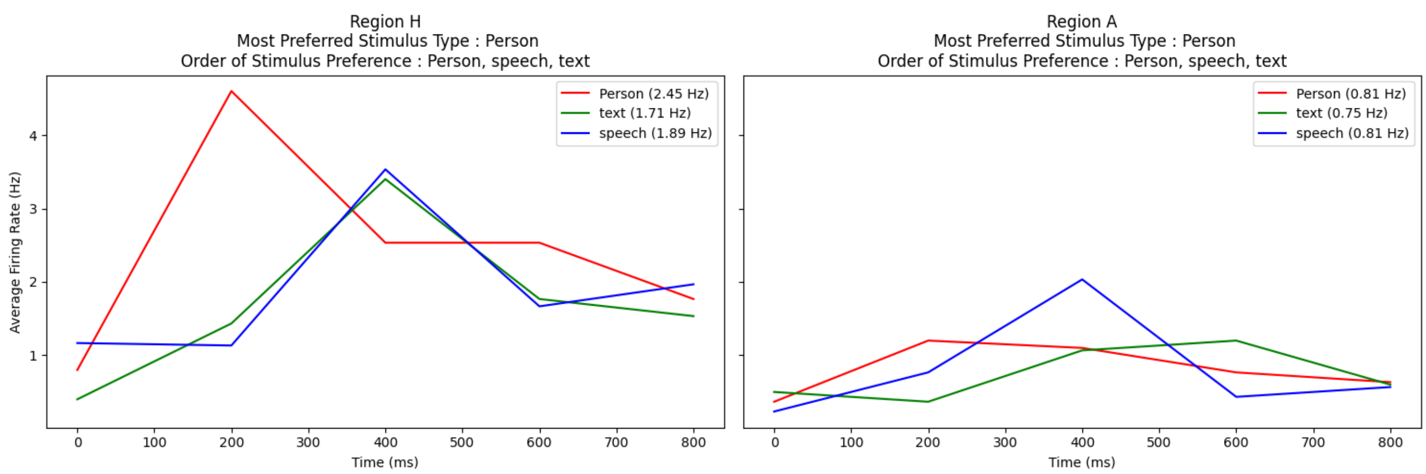
```
import scipy.io
import numpy as np
import matplotlib.pyplot as plt

# Load and extract data
H = [scipy.io.loadmat('dataset_H.mat')['dataset_H'][0][i] for i in
range(3)]
A = [scipy.io.loadmat('dataset_A.mat')['dataset_A'][0][i] for i in
range(3)]
stimuli = ['Person', 'text', 'speech']

# Computation and Smoothing data
def compute(data, size=50, timeWindow=50):
    time = np.arange(-1000, 3000, size)
    histogram = np.sum([np.histogram(i[0], bins=time)[0] for i in data],
axis=0)
    smoothedHistogram = np.convolve(histogram, np.ones(timeWindow) /
timeWindow, mode='same')
    return time[:-1], smoothedHistogram
```

```
# Plot data
fig, subplot = plt.subplots(3, 2, figsize=(15, 10), sharex=True,
sharey=True)
for i, p in enumerate(stimuli):
    for j, (data, area) in enumerate(zip([H, A], ['H', 'A'])):
        time, PSTH = compute(data[i])
        subplot[i, j].plot(time, PSTH, color='blue')
        subplot[i, j].axvline(0, color='green', linestyle='--')
        subplot[i, j].axvline(1000, color='red', linestyle='--')
        subplot[i, j].set_title(f'Region {area} - {p}')
        subplot[i, j].set_ylabel('Neuron Activity')
        subplot[i, j].set_xlabel('Time (ms)')
        subplot[i, j].legend(['Stimulus-Onset', 'Stimulus-Offset'])
plt.tight_layout()
plt.show()
```

C) Create figures representing the preferred stimulus type of two neurons (dataset_H & dataset_A) recorded from two distinct brain regions from the average firing rate of the neuron (between 0 – 1000 ms and all trials) for each stimulus type using a binwidth of 200 ms. Plot the curves of each stimulus type recorded from the same neuron in one graph. There will be two separate graphs for the two neurons/datasets. Mark the curves for stimulus type -person in red colour, text in green colour, and speech in blue colour. Computationally calculate the 'most preferred stimulus type' for each neuron and report it on the title of the plot. Also, report the order of stimulus preference for each neuron. [5 + 2 + 1 marks]

What kind of neuronal information coding is represented by these neurons and why? [2 marks]

Hint: Create a larger figure with two subplots (positioned as 1 row x 2 columns).



The neuronal information coding represented by these neurons is **stimulus selectivity.**

Because different stimuli show different firing rates, also the neurons respond differently in region A and region H. The most preferred stimulus type is computed using the highest average firing rate, which means neurons are more responsive to this particular stimulus therefore the neuronal information coding represented by these neurons is stimulus selectivity.

Code for Part-C

```
import scipy.io
import numpy as np
import matplotlib.pyplot as plt

# Data-Loading
H = [scipy.io.loadmat('dataset_H.mat')['dataset_H'][0][i] for i in
range(3)]
A = [scipy.io.loadmat('dataset_A.mat')['dataset_A'][0][i] for i in
range(3)]
stimuli = ['Person', 'text', 'speech']
color = {'Person': 'red', 'text': 'green', 'speech': 'blue'}

# Computing average firing rate
def compute(data, size=200, timeInterval=1000):
    time = np.arange(0, timeInterval + size, size)
    result = np.sum([np.histogram(i[0], bins=time)[0] for i in data],
axis=0) / len(data)
    return time[:-1], result

# Data-Plotting
fig, subplot = plt.subplots(1, 2, figsize=(15, 5), sharey=True)
for data, area, subplt in zip([H, A], ['H', 'A'], subplot):
    firingRate = {}
    for type, color in color.items():
        time, rate = compute(data[stimuli.index(type)])
        subplt.plot(time, rate, color=color, label=f'{type}
({np.mean(rate):.2f} Hz)')
        firingRate[type] = np.mean(rate)
    preferredType = max(firingRate, key=firingRate.get)
    order = sorted(firingRate, key=firingRate.get, reverse=True)
    subplt.set_title(f'Region {area}\n Most Preferred Stimulus Type :
{preferredType}\nOrder of Stimulus Preference : {", ".join(order)}')
    subplt.set_xlabel('Time (ms)')
    subplt.legend()
subplot[0].set_ylabel('Average Firing Rate (Hz)')
plt.tight_layout()
plt.show()
```