**Due: Midnight December 10, 2023**

- All files should contain your name-lastname in the file name and also as a comment written inside the file.
- Upload written answers in a single file for both questions
- Upload programs AND outputs in separate files for both questions
- You can exchange thoughts and methods with other students BUT NOT codes and results
- You can use freely available software as long as you provide a reference in your report

Q1. (20 pts) Facial EMG classification.
You can perform the computations using Excel, Matlab, Python. Upload the program(s) and a single result for the program(s) after running them.

Facial expressions are important in our daily evaluations and can be captured visually through a camera or by muscle contractions (EMG) through surface electrodes as seen below.
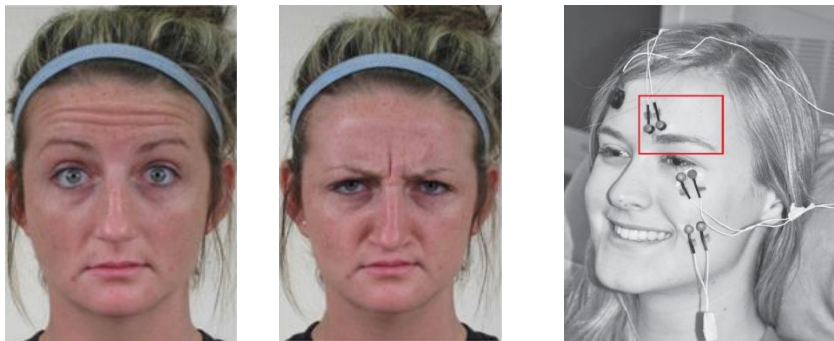
Details:
We have a relatively small set of EMG samples collected from 9 subjects for 2 facial expressions: The first expression: Raise eyebrow (left), the second expression: Frown (middle). Each subject repeated each expression twice. So in total, we have 36 sample recordings (9x2x2). There are only 18 recordings for each expression to classify. The EMG data collected from the probes is preprocessed to remove noise and high frequency components and therefore the resulting data looks like an ERP waveform (but remember it is NOT ERP it is EMG).

The format of the data is as follows:
Every row contains the next sample measured through the trial time, where the recorded value is the measured muscle contraction under the electrode. The first row is the recording at t=0, when the subject heard the command: raise or frown. The worksheets are named as follows: **s:e:r** where **s is subject number**, **e is expression number**, **r is repetition number**. For example 3:2:1 means the first repetition of expression 2 of subject 3.

The meaningful parts of a single recording is as follows:
Below is a sample expression from the data (a single column with EMG (y) plotted through time (x). The magnitude of the peak (y), the location (x) of the peak, the onset latency (x) of the peak,

the width of the waveform at half value of the peak are a few features. Some other features that aggregate the entire data can be mean and entropy.
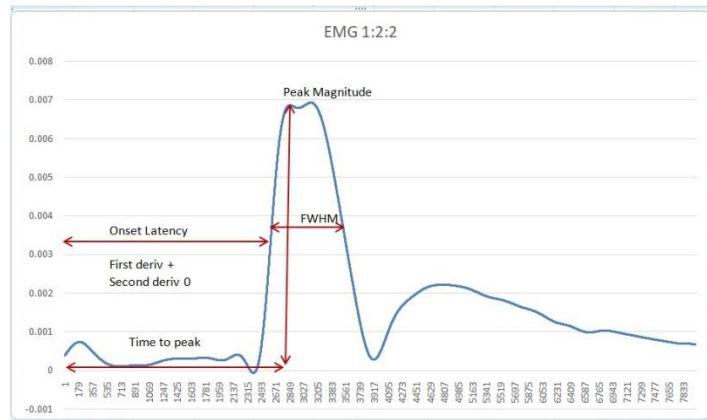
Complications are as follows:

Sometimes there are double peaks in the data, due to muscle twitch. We may decide to ignore the second.

Coding assignment:

Part 1: Exploration and evaluation: Plot sample EMG recordings across time, determine complications (code)

Part 2: Selection of features: Choose 2-3 features and write code to calculate them (upload code)

Part 3: Classification: (eg. t-tests, ANOVA, KNN, Kmeans**)** Classify if the recorded expression is 1 (raise) or 2 (frown) using statistics or ML with leave one out or 10 fold cross validation (upload code)

Questions to answer in writing

a) Attach a few EMG graphs per person per expression per repetition. Also specify the method you implemented for dealing with complications.

b) Add a single 2d or 3d graph showing the distribution of the selected features across the x,y (and z) axes with colored classes of the two expressions (raise and frown). Provide detailed explanation of your rationale and computation method for the chosen features.

c) Show the classification results of the tests using a table

d) Explain why EMG is useful rather than camera recordings and also discuss barriers in using EMG

Q2. (20 pts) Recognition memory test for false memory
Perform the experiment using Psychopy. Upload the programs and a single result for the programs after running them.
Hint1: page 292 of the textbook , experiment in the pink box.
Hint2: For 6 demos, go to https://www.testable.org/ward and click on: memory for lists.

Recognition memory tests have 3 parts. You are required to **program AND test 2 of these parts**.
Part 1: Encoding (programming needed)
Part 2: Short break (no programming) a brief stop period between part 1 and part 3 (eg. 5 min)
Part 3: Recall (programming needed)

Details:
During the **encoding**, participants are shown several words and they are asked to remember them for later. In this part, words come to the screen one after the other for a fixed interval (let's say for 1 sec). During the **break**, the participants are asked to do something else to keep their mind busy so that they do not try to recall the words themselves (eg 3-5 minutes of simple math addition or subtraction, but not anything to do with words). During **recall**, participants are shown both words from part 1 and words that are new, and they are asked to press Y, if they have seen the word on the screen in part 1 and N if they have not seen the word on the screen in part 1.

Actually, both parts are quite similar in terms of programming. One part requires no response from the participant, just displays the next word after a fixed delay. The other part requires Y or N response from the participant and switches to the next word after a response is obtained.

List of words for part 1 (26)

		List of words for part 2 (50) All the words in part 1, plus the 24 new words below	
bayır	ağaç	odun	mayo
dağ	selvi	çadır	midye
kamp	çınar	kene	balık
ateş	nehir	sıçan	duba
çadır	dere	çam	okyanus
duman	vadi	orman	yelken
kömür	ova	dal	yüzme
tente	şafak	tepe	meltem
sinek	gece	ırmak	çıpa
böcek	siyah	mola	gemi
fare	yeşil	akşam	olta
konaklama	koyu	lacivert	martı
tempo	koru	(column1)	(column2)

Questions to answer in writing
a) What do you expect in part 2? Explain your hypothesis in detail based on consolidation theory.
b) Review the words carefully. Can we do this same experiment using artificial neural networks? With what kind of input? Would we obtain similar results between artificial neural networks and humans? Justify your answers carefully for <u>all</u> of words in the lists.