Overview

This coursework exam will have two main parts: data analysis (part 1) and modelling (part 2). The modelling part will also contain two sub-parts: Associative networks (part 2.1) and Integrate and fire neurons (part 2.2).

Write a brief report, including the figures and the comments specified. No need to be verbose, just say what you are asked to say. Use at least 2 cm margins and 10 pt fonts. You can use whatever programming language you like, however we will only support Python or Julia. Python would probably be the easiest. Submit both your report and code. Do not put them together in a zip file or similar. Submit the report as pdf; make sure to set the size of the graphs in your plotting program, not by shrinking the graph to fit in the report, this makes the figure legends and axis numbers tiny and is super annoying. And don't use screenshots as your figures. All plots should have clear x- and y-axis labels, and units. Marks will be docked for every plot that has missing or unclear labels, or looks like a screenshot.

To help you decide which results to include in your report, I will encode meaning in my text colour and formatting. If a sentence is coloured magenta, I expect to see a written response. *For sentences formatted with italics I expect to see a plot.*

Most sections have a basic part and an advanced part. Doing all the basic parts (implementing what you are asked to do correctly, generating the plots, etc) will get you reasonable mark, perhaps up to $\sim 60\%$. To get a higher mark you will also need to answer the advanced parts well, which require some thinking and further analysis.

Other logistic information is included at the end of this document.

1 Data analysis [30 marks]

This part will focus on analysing some fictional spike train data and the random-dot motion task.

Question 1 [10 marks]

In the data folder associated with the coursework, you will find a file called `neuron_A_data.csv`. Load this file, for example using the python command `np.loadtxt('neuron_A.csv', delimiter=',')`. This contains a set of spike times from a fictional neuron, in milliseconds.

`https://cs-uob.github.io/COMS30017`

Compute the coefficient of variation (CV) of this spike train. Also compute the Fano factor for time bins of 100 ms, 300 ms, 600ms and 1 s. State the values of both the CV and Fano factor in your report.

Now load the file `trial_ID.csv`, which contains an array of zeros and ones. Assume that the spike times above came from a neuron recorded from the visual cortex of a monkey during a series of sequential trials, each lasting one second long. So trial 4 lasts from $t = 3000$ to $t = 3999$, trial 18 lasts from $t = 17000$ to $t = 17999$, and so on. The corresponding trial IDs are given by the values in the `trial_ID` array. trial_id $= 0$ corresponds to a random-dot visual stimulus where there was no net motion. trial_id $= 1$ corresponds to a moving random-dot stimulus. Now separate the spike times corresponding to trials where trial_id $= 0$ and trials where trial_id $= 1$. Compute the CV and Fano factors again separately on the data corresponding to each trial type. For Fano factor, use the same time bins as before. For the CV, just include spike intervals for pairs within the same time bin (exclude spike pairs that span across bins). State the values you find in your report. Do these values differ from what you found above on the total data? Offer an explanation for why or why not.

Question 2 [10 marks]

Now you will decode the stimulus from the spike data using the procedure described in week 3 video 5. You will work with the spike counts. Count the number of spikes on each trial, and *plot histograms of the spike count for the stimulus off and stimulus on trials.* Compute the d' value for these data.

Now write a function to do the decoding. It should take as input a spike count from a single trial, and the decision boundary. If the spike count is greater than the decision boundary, predict the motion stimulus (1), if it is less than the boundary, predict the no-motion stimulus (0). Then run the decoder on all the trials in the dataset, multiple times for various values of the decision boundary between 0 and 40 spikes/trial. Call a correct detection of a motion stimulus a 'true positive', a correct detection of no-motion stimulus a 'true negative'. *Make a plot overlaying three curves: the true positive rate, the true negative rate, and the total correct rate, all as a function of the decision boundary value.* Note that random guessing would correspond to a total correct rate of 50%. Comment on what you observe in the plots. If we wanted to maximise the total fraction of correct responses, what would you estimate the optimal decision boundary to be? Explain how this relates to

the count histograms you calculated above.

Question 3 [10 marks]

Now load the other data file `neuron_B_data`. This contains spike times from another neuron simultaneously recorded during the same task. Compute the d' value for these data. Recompute the true-positive rate, true-negative rate, and total correct rate for neuron B as a function of the decision boundary and *plot the three together on a graph*, as you did before for neuron A.

Now let's try to decode the stimulus from the joint activity from the pair of neurons. Let x_A and x_B correspond to single-trial spike counts from neurons A and B respectively. Write a function to decode the stimulus value from a single trial spike count pair (x_A, x_B), using the following rule:

$$\text{if } x_A - x_B - 6 > 0, \text{ then trial\_id} = 1 \tag{1}$$
$$\text{if } x_A - x_B - 6 \leq 0, \text{ then trial\_id} = 0 \tag{2}$$

Apply this function to the spike data to decode the stimulus state on each trial. Report the total correct rate you find using this joint rule. Comment on how this compares with the best decoding accuracy you found for the individual neurons above. Offer an explanation for what you observe, *including additional plots if necessary to support your explanation.*

2 Modelling [70 marks]

2.1 Auto-associative networks [30 marks]

This part is on Hopfield networks.

Question 1 [10 marks]

Begin by loading the partial Fashion MNIST dataset (taken from
`https://github.com/zalandoresearch/fashion-mnist`) stored in the files
`train_images.csv` and `test_images.csv`. The total dataset consists of three
training patterns and two test patterns. Each individual image is actually a
784-element array of +1s and -1s, corresponding to pixels in a 28×28 image.
You can visualise the images using a command like
`plt.imshow(np.reshape(x,(28,28)))`.

   The values of the weights in a Hopfield network are given by

$$w_{ij} = \frac{1}{N} \sum_a x_i^a x_j^a \tag{3}$$

when $i \neq j$. a labels patterns, x_i^a is the activiation of the ith node in the ath
pattern and N is the number of patterns. $w_{ii} = 0$.

   Now you should:

- Create a weight matrix of w_{ij}s.

- Fix the weight values to store the three training patterns. *Plot the
  weight matrix as an image.*

- Write a function to evolve the network according to the McCulloch-
  Pitts formula; this should be done asynchronously, so one random node
  is update each timestep. Assume the threshold $\theta = 0$.

- For each of the two test patterns, evolve the patterns until they stop
  changing. *For each case make plots of the network state at timesteps
  0, 800, and 4000.*

Question 2 [10 marks]

Now write a function to calculate the energy of a configuration

$$E = -\frac{1}{2} \sum_{ij} x_i w_{ij} x_j \tag{4}$$

Re-run the two test plot simulations, recording the energy value at each iteration. *For each case, make plots of the energy as a function of timestep number.* Describe what the plots show. Explain in your own words why this happens, referring to the network update equation or energy equation if needed.

Question 3 [10 marks]

Now we will examine the network's 'basins of attraction'. Write a function that when given a pair of images A and B, can generate a mixed image which has $x\%$ of the pixels from image A and the remaining $(100 - x)\%$ pixels from image B. For each of three pairs of images in the training set, compute such mixed images, varying x in increments between 0 and 100%, and compute the energy of the mixed image at each increment. *Plot the energy as a function of the mixing fraction x, overlaid for each of the three pairs of images.* Comment on the relative shapes of the three curves. List some predictions for the model dynamics based on these curves, explaining your reasoning in each case.

2.2 Integrate-and-fire neurons [40 marks]

This part will look at integrate-and-fire neurons and short-term plasticity.

Question 1 [7 marks]

Make a simulation of a single leaky-integrate-and-fire neuron. Set the resting potential $V_{rest} = E_L = -72$ mV, spike threshold $V_{th} = -52$ mV, reset voltage $V_{reset} = -62$ mV, a passive membrane leak conductance that gives an input resistance of $R_m = 100$ MΩ, and $\tau_m = 30$ ms. Set the injected current $I_e = 0.21$ nA. Use Euler's method with timestep $\delta t = 0.1$ ms to solve the differential equations numerically. Make sure to use the differential equations, do not just plot the analytical solution for V(t) given in the lectures – this will be useless for later questions anyway.

For simplicity assume that the neuron does not have a refractory period after producing a spike. You do not need to plot spikes – once the membrane potential exceeds threshold, simply set the membrane potential to V_{reset}. To keep track of the units I recommend converting everything to S.I. units inside your simulation code – you don't strictly have to do this but it often makes things easier.

Set the initial voltage value $V(t = 0)$ to a random number between V_{rest} and V_{th}. Run the simulation for one second and *plot the voltage as a function of time*.

Now run the simulation for ten seconds, for various levels of injected input current I_e between 0 nA and 0.5 nA, and computing the spike frequency in each case. *Plot the spike frequency as a function of the input current* (the '*f-I*' curve). Analytically calculate the threshold input current and report its value.

Question 2 [8 marks]

Now turn off the background input current, $I_e = 0$. Add $N = 100$ total incoming synapses, split into two equal sized groups, ($N_1 = 50$, $N_2 = 50$). We will make these groups have different firing rates. You should just model the synaptic responses to spike trains, do not explicitly simulate the somatic voltages of these presynaptic neurons.

Model all synapses as a conductance-based, following a single-exponential timecourse with decay time constant $\tau_s = 2$ ms, the peak conductances (strengths) all $\bar{g}_i = 0.5$ nanoSiemens for all synapses, $E_s = 0$, and $\Delta s = 1$.

https://cs-uob.github.io/COMS30017

Let $s_i(t = 0) = 0$ for all 100 synapses. It's probably easier to store them all in arrays and update them together, rather than code each one by hand. Again use the Euler method to solve the system of ODEs (100 s variables plus 1 voltage variable), with the same timestep $\delta t = 0.1$ milliseconds.

Simulate each input spike train as an independent homogeneous Poisson process with the same average firing rate r. A homogeneous Poisson process implies that average firing rate r is fixed in time, but that the spikes themselves arrive randomly. Initially set the rate for group 1 $r_1 = 10$ Hz and the rate for group 2 $r_2 = 100$ Hz. There are multiple ways to simulate Poisson spike trains. One simple method is to draw a random number on the unit interval for each synapse at every timestep, then if that number is less than $r\delta t$, assume a spike has occurred at that synapse. This method is valid as long as $r\delta t \ll 1$.

Simulating your model at this stage should result in the neuron firing irregularly. *Plot the neuron's voltage for one second of simulation time.*

Now run the simulation for ten seconds, for various levels of r_1 and r_2 separately between 0 Hz and 150 Hz, and compute the spike frequency in each case. When varying r_1, keep r_2 fixed at 100 Hz. When varying r_2, keep r_1 fixed at 10 Hz. *Plot the output spike frequency as a function of the input spike frequency, overlaid for both cases.* Describe what is shown in the plots and offer an explanation for the shapes of the curves.

Question 3 [10 marks]

Now we will make the mean firing rate for each group vary sinusoidally in time:

$$r(t) = \bar{r} + B \sin(2\pi f t)$$

where $\bar{r}$ is the average firing rate. The amplitude of the oscillation depends on the parameter B, while the timescale of oscillations depends on f. Fix $f = 2$ Hz, $\bar{r}_1 = 10$ Hz, $\bar{r}_2 = 100$ Hz.

Running the simulation for 1 s in each case, *make plots of the voltage as a function of time*, for two different scenarios (visualised in the left and centre panels in the figure below):

- $B_1 = 0$ Hz and $B_2 = 50$ Hz.

- $B_1 = 5$ Hz and $B_2 = 0$ Hz.

Describe what you see in the plots. What was the mean firing rate of the neuron in each case, as averaged over a 30 second simulation?. Note that the mean total input firing rate (averaged over time) is the same in both cases, and also identical to that used in question 2. Explain why the three different cases lead to different output firing rates.
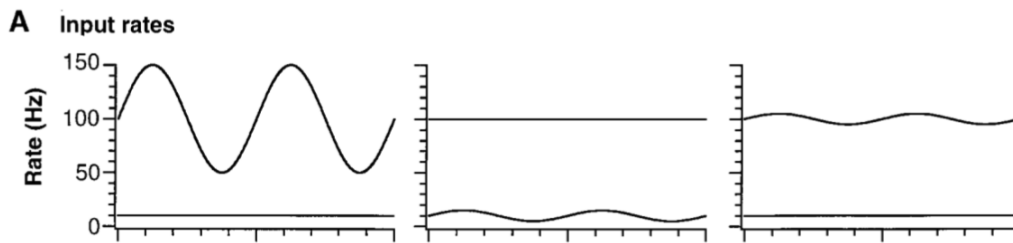
Question 4 [15 marks]

Now implement short-term depression at each synapse. Introduce an additional variable A for each synapse, so that the total synaptic conductance will be $g_i(t) = A_i(t)s_i(t)\bar{g}_i$. A is going to represent the effects of short-term depression. Every time a spike arrives at the ith synapse, set $A_i \to \alpha \times A_i$, where $\alpha = 0.75$. In between spikes, let A_i recover back towards 1 by letting it follow the differential equation $\frac{dA_i}{dt} = \frac{1-A_i}{\tau_A}$, with $\tau_A = 300$ ms. Also change all the synaptic strengths to $\bar{g} = 3$ nS (to counter the effects of depression).

Now repeat the two simulations from Q3 but with synaptic depression turned on. In this case run the simulation for 1 second first to let it 'settle down', then run it for a further 3 seconds to observe the steady-state dynamics. Make plots of the voltage as a function of time in each case. Describe what you observe, and compare it to the plots obtained in Question 3. Offer an explanation for what is going on. You can make extra plots if they support your argument.

Finally, run the simulation with synaptic depression still on, but with $B_1 = 0$ Hz and $B_2 = 5$ Hz (visualised in the right panel of the above figure). Again run the simulation for 1 second warmup plus a further 3 seconds. Make a plot of the voltage vs time. Describe the results and and offer an explanation for what you observe, commenting on how it compares with the previous two plots.

Logistic information

Deadline

The deadline for submission of all optional unit assignments is 13:00 on Monday 16th of August 2021. Students should submit all required materials to the "Assessment, submission and feedback" section of Blackboard - it is essential that this is done on the Blackboard page related to the "With Coursework" variant of the unit.

Time commitment

The effort spent on the assignment for each unit should be approximately equivalent to 1.5 working weeks.

Academic Offences

Academic offences (including submission of work that is not your own, falsification of data/evidence or the use of materials without appropriate referencing) are all taken very seriously by the University. Suspected offences will be dealt with in accordance with the University's policies and procedures. If an academic offence is suspected in your work, you will be asked to attend an interview with senior members of the school, where you will be given the opportunity to defend your work. The plagiarism panel are able to apply a range of penalties, depending the severity of the offence. These include: requirement to resubmit work, capping of grades and the award of no mark for an element of assessment.

Extenuating circumstances

If the completion of your assignment has been significantly disrupted by serious health conditions, personal problems, periods of quarantine, or other similar issues, you may be able to apply for consideration of extenuating circumstances (in accordance with the normal university policy and processes). Students should apply for consideration of extenuating circumstances as soon as possible when the problem occurs, using the following online form: https://apps.powerapps.com/play/3172b943-0956-4b88-bf3d-3f37871d1170?tenantId=b2e47f30-cd7d-4a4e-a5da-b18cf1a4151b

https://cs-uob.github.io/COMS30017