Final Exam of Computational Neuroscience

Due at the beginning of spring semester, 2023

Backpropagation rule

Derive the backpropagation rule for 3-layer networks. Suppose for the neurons, we use the following notations:

$$I_j^i = b_j + \sum_k r_k^{i-1} w_{jk} \tag{1}$$

$$r = \frac{1}{1 + e^{-I}} \tag{2}$$

The superscript denotes which layer and subscript denotes which neuron in the layer. Please write out the equations for the value on the final third layer and take the gradient and show why backpropagation algorithm works.

Hopfield model

In the Hopfield network, we make even a more drastic assumption that the activity of a neuron is a binary variable, $s_i \in [+1, -1]$. In the deterministic version of this model, the state of a cell is set according to the following equation

$$s_i(t + \Delta t) = \mathbf{sgn}(\sum_j w_{ij} s_j), \tag{3}$$

We consider an asynchronous update rule for each neuron (choosing either a random or fixed update order and update s_i according to that order at each clock cycle).

(1) Check my lecture note and proof that any mixture of an odd number of memory patterns, such as

$$\xi^{mix} = \text{sign}(\xi^{\mu_1} + \xi^{\mu_2} + \xi^{\mu_3}) \tag{4}$$

are fixed point of the Hopfield model.

(2) Numerically simulate the dynamics of Hopfield network with random unbiased memory patterns, using 500 binary neurons. We want to draw some statistical conclusions, so in the tasks that follow, you should run many simulations using different sets of random stored patterns and, for each, start from many different initial conditions (so I recommend writing reasonably efficient routines, e.g. minimize the number of for-loops in MATLAB). For simplicity, you can choose the order of update of the neurons by going through neuron 1 to N all the time.

i) Probability of perfect recall: Store 10 patterns, run simulations to find the probability of perfect recall. That is, start with a pattern, corrupt some fraction of its bits, input the corrupted pattern to the network, and wait until the network settles into a fixed point. Plot the fraction of times you get perfect recall of the original state against the fraction of corrupted bits.

ii) Memory retrieval with errors: Start the dynamics with one of the memory states. Plot the fraction of errors (fraction of incorrectly recalled bits) in the final state as a function of the number of stored patterns P. Change P from 10 to 100.

Simulating large spiking neural network with excitatory and inhibitory neurons

Let us analyze the dynamics of a network composed of N integrate-and-fire (IF) neurons, from which N_E are excitatory and N_I inhibitory. Each neuron receives a fraction of randomly chosen connections from other neurons in the network, from which $C_E = \epsilon N_E$ from excitatory neurons and $C_I = \epsilon N_I$ from inhibitory neurons. We consider a sparsely connected network with $\epsilon \ll 1$. Each neuron also receives $C_e xt$ connections from excitatory neurons outside the network.

The membrane potential of neuron i obeys the equation

$$\tau \dot{V}_i(t) = -V_i(t) + \tau \sum_j J_{ij} \sum_k \delta(t - t_j^k - D) \tag{5}$$

where the first sum on the right hand side is a sum on different synapses ($j = 1, ..., C_E + C_I + C_{ext}$), with synaptic weight J_{ij} , while the second sum represents a sum on different spikes arriving at synapse j, at time $t = t_j^k + D$, where t_j^k is the emission time of kth spike at neuron j, and D is the transmission delay. Note that in this model a single synaptic time D is present. For simplicity, we take weights equal at each synapse—that is, $J_{ij} = J > 0$ for excitatory external synapses, J for excitatory recurrent synapses (note the strength of external synapses is taken to be equal to the recurrent ones), and $-gJ$ for inhibitory ones. External synapses are activated by independent Poisson processes with rate ν_{ext}. When $V_i(t)$ reaches the firing threshold θ, an action potential is emitted by neuron i, and the depolarization is reset to the reset potential V_r after a refractory period τ_{rp} during which the potential is insensitive to stimulation. The external frequency ν_{ext} will be compared in the following to the frequency that is needed for a neuron to reach threshold in absence of feedback, $\nu_{thr} = \theta/(JC_E\tau)$.

We study the case in which inhibitory and excitatory neurons have identical properties, and let $J_{EE} = J$ (excitatory to excitatory); $J_{EI} = gJ$ (inhibitory to excitatory), $J_{IE} = J$ (excitatory to inhibitory); $J_{II} = gJ$ (inhibitory to inhibitory).

Here are some of the parameters we will be using. $N_E = 0.8N$, $N_I = 0.2N$ (80 percent of excitatory neurons). The number of connections from outside the network is taken to be equal to the number of recurrent excitatory ones, $C_{ext} = C_E$, and we set the sparseness factor $\epsilon = 0.1$. We also choose $\tau = 20\ ms$; $\theta = 20\ mV$; $V_r = 10\ mV$; $\tau_{rp} = 2\ ms$, $D = 1.8\ ms$, $J = 0.2\ mV$. Note that the value for θ and V_r are a bit arbitrary, but you can always shift these values by the same amount without changing much of the conclusion.

Write a code to simulate a neural network with $N = 1000$ neurons over time. There are two parameters that remain to be determined, g, the ratio of inhibition to excitation, and ν_{ext}, the rate of poisson spike from external inputs. Here are a few things you are asked to do.

1. plot mean firing rate of a neuron as a function of g, for different $\nu_{ext}/\nu_{thr} = 0, 1, 2$, and explain what you see. Please at least have some idea the range of g you need to explore.

2. plot coefficient of variation (CV) of inter spike interval as a function of g for for different ratios of ν_{ext}/ν_{thr} and explain what you see. It would also be nice to have a raster plot of spikes, where the x-axis is time, y-axis is neuron index, and color indicates whether there is a spike or not.

3. (*) With the above insights you have, explore the phase diagram of the population firing patterns, where the y-axis would be ν_{ext}/ν_{thr}, and x-axis is g.