Gatsby Computational Neuroscience Unit
Theoretical Neuroscience

Final examination, theoretical neuroscience
6-8 May 2021

Part I – short questions

There are four sections with four questions each. Please answer three out of each four, starting the answers for each section on a new page. Don't forget to write your name at the top of each block of answers.

Good luck!

1 Biophysics

1. Consider a Hodgkin Huxley neuron that has lost most of its channels,

$$\tau \frac{dV}{dt} = -(V - \mathcal{E}_L) - g_{Na} m V + V_{\text{ext}} \tag{1a}$$

$$\tau_m \frac{dm}{dt} = m_\infty(V) - m \tag{1b}$$

where V_{ext} is external voltage. For simplicity, we have set the sodium reversal potential to zero. Assume standard values for $\mathcal{E}_L$ and a standard shape for $m_\infty(V)$.

Show that when g_{Na} is sufficiently small, there is only one fixed point, and it's stable.

Show that when g_{Na} is sufficiently large, there can be one, two or three fixed points, depending on the value of V_{ext}. Indicate the stability of the fixed points in these three regimes.

Solution The m and V nullclines are given, respectively, by

$$m = m_\infty(V) \tag{2a}$$

$$m = \frac{1}{g_{Na}} \left[\frac{\mathcal{E}_L + V_{\text{ext}}}{V} - 1 \right] . \tag{2b}$$

For $m_\infty(V)$ I used

$$m_\infty(V) = \frac{1}{1 + \exp(-(V + 50)/6)} , \tag{3}$$

but pretty much any sigmoidal-ish function that's not too steep or too shallow will work. I used $\mathcal{E}_L = -65$ mV. The nullclines are plotted in Fig. 1, using the parameters given in the caption. It's not hard to show, just by drawing trajectories, that an equilibrium is stable if the m-nullcline has smaller slope than the V-nullcline, and it's a saddle point otherwise.

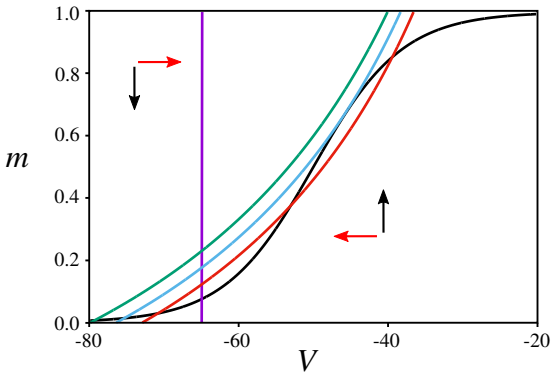

Figure 1: Nullclines. Black arrows indicate the sign of dm/dt and red arrows indicate the sign of dV/dt. Black curve: m-nullcline. Colored curves: V-nullclines. The purple line has $g_{Na} = 0$ and $V_{\text{ext}} = 0$; the (single) equilibrium is clearly stable, as can be seen from the black and red arrows. The green, blue and red curves have $g_{Na} = 0$ and V_{ext} = -15, -11.5 and -8 mV, respectively. For the reversal potential I used $\mathcal{E}_L = -65$ for all the V-nullclines.

2. The channel dynamics in the Hodgkin Huxley equations arise from a two-state model. Assume instead that there are n states. Let α_{ij} be the transition probability from state j to state i – that is, the probability of making a transition from state j to state i in time dt is $\alpha_{ij} dt$, for $i \neq j$. Show that the probability, $P_i(t)$, of being in state i at time t evolves according to

$$\frac{dP_i(t)}{dt} = \sum_{j \neq i} (\alpha_{ij} P_j - \alpha_{ji} P_i) . \tag{4}$$

Consider a three-state model: state 1 is closed, state 2 is open, and state 3 is inactivated. Assume that the only possible transitions are:

$$\text{closed} \to \text{open} \tag{5a}$$
$$\text{open} \to \text{inactivated} \tag{5b}$$
$$\text{inactivated} \to \text{closed}. \tag{5c}$$

What's the equilibrium probability of being open in terms of the α_{ij}?

Solution To derive the update rules for the probabilities, use the fact that the probability at time $t + dt$ is determined by

$$P_i(t + dt) = \sum_{j \neq i} \alpha_{ij} dt P_j(t) + \left(1 - \sum_{j \neq i} \alpha_{ji} dt\right) P_i(t)\,. \tag{6}$$

The first term is the increase in probability due to other states transitioning into state i; the second term is the decrease in probability due to transitions from state i to other states. Using

$$P(+dt) = P(t) + dt\,\frac{dP(t)}{dt} \tag{7}$$

(strictly valid as $dt \to 0$), and performing a small amount of algebra, we arrive at the answer to the first part of the question.

For the second part, we're considering a three state model in which only α_{21}, α_{32} and α_{13} are nonzero. Consequently, the equations become

$$\frac{dP_1}{dt} = \alpha_{13} P_3 - \alpha_{21} P_1 \tag{8a}$$
$$\frac{dP_2}{dt} = \alpha_{21} P_1 - \alpha_{32} P_2 \tag{8b}$$
$$\frac{dP_1}{dt} = \alpha_{32} P_2 - \alpha_{13} P_3\,. \tag{8c}$$

To find the equilibrium values, we set dP_i/dt to zero, and solve the resulting algebraic equations. However, we have to also include the fact that probabilities sum to 1, $P_1 + P_2 + P_3 = 1$. When we do that, we arrive at

$$P_1 = \frac{\alpha_{13}\alpha_{32}}{\alpha_{21}\alpha_{13} + \alpha_{32}\alpha_{21} + \alpha_{13}\alpha_{32}} \tag{9a}$$
$$P_2 = \frac{\alpha_{21}\alpha_{13}}{\alpha_{21}\alpha_{13} + \alpha_{32}\alpha_{21} + \alpha_{13}\alpha_{32}} \tag{9b}$$
$$P_1 = \frac{\alpha_{32}\alpha_{21}}{\alpha_{21}\alpha_{13} + \alpha_{32}\alpha_{21} + \alpha_{13}\alpha_{32}}\,. \tag{9c}$$

The relevant one is P_2, the open probability.

3. Consider a current-based linear integrate and fire neuron receiving synaptic input,

$$\tau \frac{dV_i}{dt} = -(V_i - \mathcal{E}_L) + \sum_j W_{ij} \sum_k g(t - t_j^k) \tag{10}$$

where t_j^k is the k^{th} spike on neuron j. Assume all neurons are firing with Poisson statistics, and that the firing rate of neuron j is ν_j. The quantity $g(t)$, which determines the shape of the PSPs, and is given by

$$g(t) = \frac{t^2 e^{-t/\tau_s}}{\tau_s^3}\,\Theta(t) \tag{11}$$

where, as usual, $\Theta(t)$ is the Heaviside step function: it's 1 if $t > 0$ and 0 otherwise.

What's the time-averaged synaptic drive to the neuron i as a function of the weights, W_{ij}, the firing rates, ν_j, and the synaptic time constant, τ_s?

Solution The time averages synaptic drive associated with neuron j, which we'll call $\overline{g}_j$, is given by

$$\overline{g}_j = \lim_{T \to \infty} \int_0^T \frac{dt}{T} \sum_k \langle g(t - t_j^k) \rangle \tag{12}$$

where the angle brackets refer to an average over spike times. We can do the integral over time first. Ignoring edge effects, this yields

$$\overline{g}_j = \lim_{T\to\infty} \frac{\langle n_j(T)\rangle}{T} \int_0^\infty dt\, g(t) \tag{13}$$

where $n_j(T)$ is the number of spikes in time T for a particular realization of a spike train, and the angle brackets represents an average over that number. The ratio $\langle n_j(T)\rangle/T$ is, by definition, the firing rate of neuron j, denoted ν_j. So all we need is the integral. Inserting the expression for $g(t)$, this is

$$\begin{aligned}\int_0^\infty dt\, g(t) &= \int_0^\infty dt\, \frac{t^2 e^{-t/\tau_s}}{\tau_s^3} \\ &= \int_0^\infty dx\, x^2 e^{-x} \\ &= -x^2e^{-x} - 2xe^{-x} - 2e^{-x}\big|_0^\infty \\ &= 2\,.\end{aligned} \tag{14}$$

For the second line we made the change of variables $t = \tau_s x$; the third is easy to verify. (Alternatively, you could recognize the integral as the Gamma function.) Putting everything together, we have

$$\text{time averaged synaptic drive } = 2\sum_j W_{ij}\nu_j\,. \tag{15}$$

4. You accidentally take a drug that blocks the active potassium channels. Sketch the shape of an action potential before and after you take the drug. Assume the neuron obeys the Hodgkin-Huxley equation.

   Solution See Fig. 2 below. These aren't the best action potentials in the world, but the point is that after taking the drug, the action potentials lose the after-hyperpolarization associated with opening of the active potassium channels (the n-current).

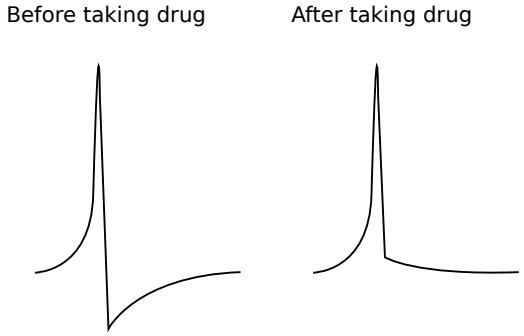

Figure 2: Action potential before (left) and after (right) taking a drug that blocks the active potassium channel in a Hodgkin-Huxley neuron.

2 Networks

1. Assume that a recurrent network evolves according to

$$\tau_x \frac{dx_i}{dt} = \sum_{\mu=1}^{K} A_{i\mu} z_\mu - x_i \tag{1a}$$

$$\tau_z \frac{dz_\mu}{dt} = \sum_{i=1}^{n} B_{\mu i} \phi(x_i) - z_\mu \tag{1b}$$

where ϕ is some nonlinear function (typically sigmoidal). Show that the z_μ can be found from the set of equations

$$\left(\tau_z \frac{d}{dt} + 1\right)\left(\tau_x \frac{d}{dt} + 1\right) \tilde{z}_\mu = \sum_i B_{\mu i} \phi\left(\sum_\nu A_{i\nu} \tilde{z}_\nu\right) \tag{2a}$$

$$z_\mu = \left(\tau_x \frac{d}{dt} + 1\right) \tilde{z}_\mu \,. \tag{2b}$$

Solution Start by defining $\tilde{\mathbf{z}}$ via

$$\left(\tau_x \frac{d}{dt} + 1\right) \tilde{\mathbf{z}} = \mathbf{z} \tag{3}$$

(we have switched to vector notation, to make this as painless as possible). In terms of $\tilde{\mathbf{z}}$, the equation for $\mathbf{x}$ becomes

$$\left(\tau_x \frac{d}{dt} + 1\right) \mathbf{x} = \left(\tau_x \frac{d}{dt} + 1\right) \mathbf{A} \cdot \tilde{\mathbf{z}} \,.$$

The operator on the left hand side isn't exactly invertible, but at long times (after transients have died away), it is. We thus have

$$\mathbf{x} = \mathbf{A} \cdot \tilde{\mathbf{z}} \,.$$

Inserting this into the equation for $\mathbf{z}$ yields

$$\left(\tau_z \frac{d}{dt} + 1\right) \mathbf{z} = \mathbf{B} \cdot \phi(\mathbf{A} \cdot \tilde{\mathbf{z}})$$

where ϕ is taken to be a pointwise nonlinearity. Using Eq. (3), we arrive at the desired result.

2. Consider a randomly connected network of excitatory and inhibitory neurons. Assume the network operates on the unstable branch of the excitatory nullcline. You increase the drive to the inhibitory neurons. Plot the firing rate of the inhibitory neurons as a function of the inhibitory drive, up to very large values.

Solution The standard nullclines are shown in Fig. 3 below. Increasing the drive to the inhibitory neurons raises the inhibitory nullcline. This initially causes the inhibitory firing rate to decrease. However, eventually the inhibitory nullcline will intersect at the minimum of the excitatory nullcline; after that, the firing rate of the inhibitory neurons will start to increase.

3. Consider an almost standard Hopfield network,

$$S_i(t+1) = \text{sign}\left[\frac{1}{n}\sum_{j=1}^{n} J_{ij} S_j(t)\right] \,. \tag{4}$$

The connectivity matrix is given, as usual, by

$$J_{ij} = \sum_{\mu=1}^{p} \xi_i^\mu \xi_j^\mu \tag{5}$$

where

$$\xi_i^\mu = \begin{cases} +1 & \text{probability } 1/2 \\ -1 & \text{probability } 1/2 \,. \end{cases} \tag{6}$$

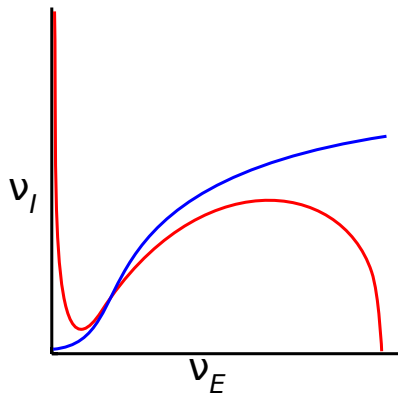

Figure 3: Standard nullclines for a randomly connected network of excitatory and inhibitory neurons, operating on the unstable branch of the excitatory nullcline. Red: excitatory nullcline; Blue: inhibitory nullcline.

Now the almost standard part: the ξ_i^μ are correlated; for $\mu \neq \nu$,

$$\text{probability}(\xi_i^\mu = \xi_i^\nu) = q \tag{7}$$

with $q > 1/2$.

Show that in the large n limit, it's possible to embed memories (fixed points of the update rule, Eq. (4)) so long as

$$p \ll 1 + \frac{1}{(2q-1)^2}\,. \tag{8}$$

Solution As usual, we'll let $S_i = \xi_i^\mu$, and see if that's a fixed point. This yields

$$\xi_i^\mu \stackrel{?}{=} \text{sign}\left[\xi_i^\mu + \sum_{\nu\neq\mu} \xi_i^\nu \frac{1}{n}\sum_j \xi_j^\nu \xi_j^\mu\right]$$

In the large n limit,

$$\frac{1}{n}\sum_j \xi_j^\nu \xi_j^\mu = q\times(+1) + (1-q)\times(-1) = 2q-1\,,$$

giving us

$$\xi_i^\mu \stackrel{?}{=} \text{sign}\left[\xi_i^\mu + (2q-1)\sum_{\nu\neq\mu}\xi_i^\nu\right]$$

The variance of the sum over ν is $p-1$, and so we may write

$$\xi_i^\mu \stackrel{?}{=} \text{sign}\left[\xi_i^\mu + (2q-1)(p-1)^{1/2}\zeta_i\right]$$

where ζ_i is a zero mean, unit variance random variable. Assuming ζ_i isn't heavy tailed (it isn't), we have equality if

$$(2q-1)(p-1)^{1/2} \ll 1$$

which we may write

$$p \ll 1 + \frac{1}{(2q-1)^2}\,.$$

4. Consider a network of randomly connected linear integrate and fire neurons,

$$\tau\frac{dV_i^E}{dt} = -(V_i^E - \mathcal{E}_L) + \sum_{j=1}^{N_E} W_{ij}^{EE} g_j^E(t) - \sum_{j=1}^{N_I} W_{ij}^{EI} g_j^I(t) + h_i^E(t) \tag{9a}$$

$$\tau\frac{dV_i^I}{dt} = -(V_i^I - \mathcal{E}_L) + \sum_{j=1}^{N_E} W_{ij}^{IE} g_j^E(t) - \sum_{j=1}^{N_I} W_{ij}^{II} g_j^I(t) + h_i^I(t) \tag{9b}$$

where N_E and N_I are the number of excitatory and inhibitory neurons, respectively, $g_j^E(t)$ and $g_j^I(t)$ are the excitatory and inhibitory drives from the j^{th} excitatory and inhibitory neurons, respectively, and $h_i^E(t)$ and $h_i^I(t)$ are excitatory and inhibitory external drive, respectively.

We'll choose weights so that the network is perfectly balanced, which means

$$\langle W_{ij}^{EE} \rangle = \langle W_{ij}^{IE} \rangle \equiv W_E \tag{10a}$$
$$\langle W_{ij}^{EI} \rangle = \langle W_{ij}^{II} \rangle \equiv W_I \tag{10b}$$

with $N_I W_I > N_E W_E$ and, of course, both W_I and W_E positive, and the external drives are also the same on average,

$$\langle h_i^E(t) \rangle = h \tag{11a}$$
$$\langle h_i^I(t) \rangle = h\,, \tag{11b}$$

and the excitatory and inhibitory conductances have the same shape,

$$g_j^E(t) = \sum_k g(t - t_j^k) \tag{12a}$$
$$g_j^I(t) = \sum_k g(t - t_j^k) \tag{12b}$$

where the g_j^k are spike times. For simplicity, assume that $g(t)$ integrates to 1.

These networks tend to be very stable; in particular, they don't oscillate. Explain why.

Solution The key observation is that the average drive to both excitatory and inhibitory neurons is the same,

$$\tau \frac{dV_i^E}{dt} = -(V_i^E - \mathcal{E}_L) + N_E W_E \langle \nu_E \rangle - N_I W_I \langle \nu_I \rangle + h + \text{fluctuations}$$
$$\tau \frac{dV_i^I}{dt} = -(V_i^I - \mathcal{E}_L) + N_E W_E \langle \nu_E \rangle - N_I W_I \langle \nu_I \rangle + h + \text{fluctuations}\,.$$

The fluctuations are, of course, different, but they have the same statistical structure. Therefore, the average excitatory and inhibitory firing rates are the same, so we'll let

$$\langle \nu_E \rangle = \langle \nu_I \rangle = \langle \nu \rangle$$

and the equations for the membrane potential become

$$\tau \frac{dV_i^E}{dt} = -(V_i^E - \mathcal{E}_L) + (N_E W_E - N_I W_I) \langle \nu \rangle + h + \text{fluctuations}$$
$$\tau \frac{dV_i^I}{dt} = -(V_i^I - \mathcal{E}_L) + (N_E W_E - N_I W_I) \langle \nu \rangle + h + \text{fluctuations}\,.$$

There is, then, really only one firing rate, and it obeys the (very approximate) equation

$$\tau \frac{d\langle \nu \rangle}{dt} = \phi\big((N_E W_E - N_I W_I) \langle \nu \rangle + h\big)$$

where ϕ is a monotonic increasing gain function. Because $N_E W_I < N_I W_I$, the network experiences negative feedback, and so can't oscillate.

Another way of seeing this is to note that oscillations occur when the excitatory firing rate rises and causes the inhibitory firing rate to rise with a delay. However, since the excitatory and inhibitory populations are essentially the same, the delay is zero.

3 Coding

1. The spiking of two neurons is well-modeled by a Hawkes-type process with intensity functions:

$$\lambda_1(t) = \rho_1(t)$$
$$\lambda_2(t) = \rho_2(t) + \sum_{i=1}^{N_1(t)} h(t - t_i^1)$$

where t_i^k and $N_k(t)$ are the spike times and counting process respectively for neuron k. Suppose that ρ_1 and ρ_2 are fixed functions.

   (a) Find $\bar{\lambda}_2(t)$ – the trial-averaged mean intensity.

   (b) Find the cross-covariance function between the two neurons in terms of the auto-correlaton function of ρ_1.

   Now suppose that ρ_1 and ρ_2 are themselves stochastic, and possibly correlated.

   (c) Give the new cross-covariance in terms of the cross-covariance of the ρs. Discuss the impact on the identifiability of h from data.

2. You read a paper describing the following experimental results: Similar-sized population recordings with overlapping receptive fields were made in both LGN and V1 while an animal viewed a natural movie. The experimenters applied a set of Gabor-like filters to the movie images at the common RF location to obtain an estimate of instantaneous orientation energy in the movie. Using a decoding approach, they computed the mutual information rate between this time-series and the LGN and V1 populations, finding that the measured information in V1 is higher. *Prima facie*, this would seem to contradict the data-processing inequality. Suggest at least two possible explanations. You may consider:

   - known properties of the two areas
   - limitations of the decoder
   - the natural movie context

   or take a different approach.

3. A neuron's response in discretised time-bins (labelled by t) is well modelled as Linear-Nonlinear-Poisson:

$$n_t \sim \text{Poisson}\big(f(\mathbf{k}^\mathsf{T}\mathbf{x}_t)\big)$$

   for count n_t, stimulus vector $\mathbf{x}_t$, linear filter $\mathbf{k}$ and non-linearity f.

   - Find the Fisher information matrix $F(\mathbf{x}) = \left\langle \nabla_\mathbf{x} \log p(n|\mathbf{x}) (\nabla_\mathbf{x} \log p(n|\mathbf{x}))^\mathsf{T} \right\rangle_{n|\mathbf{x}}$.
   - Suppose f is constrained to be monotonically increasing, has output bounded by a maximal rate $r_{\max}$; and that $\mathbf{x}$ is normally distributed with zero mean. Sketch (and justify) the form of f that will maximise the (trace of) the average Fisher information.

4. We reviewed the argument that if a population of neurons encodes a single quantity of dimension 3 or more with radially symmetric receptive fields, then it is optimal (at least in a Fisher info sense) for those fields to be as broad as possible.

   (a) Briefly recall the intuition that underlies this result – you don't need to give a full derivation.

   (b) Discuss how you would expect this result to change if the same population instead represented an inferred belief about the 3+-dimensional quantity. Would dimensionality (of the quantity, or of the belief) still play a role?

4 Learning

1. Consider a linear neuron whose output, y is related to its input $\mathbf{x}$, via

$$y = \mathbf{w} \cdot \mathbf{x} \tag{1}$$

Assume an update rule of the form

$$\mathbf{w}_{n+1} = \frac{\mathbf{w}_n + \eta y_n \mathbf{x}_n}{|\mathbf{w}_n + \eta y_n \mathbf{x}_n|} \tag{2}$$

where $\mathbf{x}_n$ and y_n are the input and output on trial n. Show that in the limit of small η, this reduces to Oja's rule.

Solution We need to Taylor expand the denominator in the small η limit. For that we'll use

$$|\mathbf{w}_n + \eta y_n \mathbf{x}_n| \approx \sqrt{\mathbf{w}_n \cdot \mathbf{w}_n + 2\eta y_n \mathbf{w}_n \cdot \mathbf{x}_n} \approx 1 + \eta y_n \mathbf{w}_n \cdot \mathbf{x}_n \tag{3}$$

where for the second line we used the fact that $\mathbf{w}_n \cdot \mathbf{w}_n = 1$. Consequently,

$$\frac{\mathbf{w}_n + \eta y_n \mathbf{x}_n}{|\mathbf{w}_n + \eta y_n \mathbf{x}_n|} \approx (\mathbf{w}_n + \eta y_n \mathbf{x}_n)(1 - \eta y_n \mathbf{w}_n \cdot \mathbf{x}_n) \approx \mathbf{w}_n + \eta(y_n \mathbf{x}_n - y_n \mathbf{w}_n \mathbf{w}_n \cdot \mathbf{x}_n)\,. \tag{4}$$

Finally, replacing $\mathbf{w}_n \cdot \mathbf{x}_n$ with y_n, we arrive at the update rule is

$$\Delta \mathbf{w} = \eta y_n (\mathbf{x}_n - y_n \mathbf{w})\,. \tag{5}$$

This is Oja's rule.

2. In class we derived backprop through time using Lagrange multipliers. But there's an easier, if perhaps less rigorous, way: start with a recurrent network, discretize time and treat it as a feedforward network, pretend that the weights are different at each timestep and run standard backprop, then fix all the weights to the same value and average the gradient. Easier than it sounds, actually. We'll start with

$$\tau \frac{d\mathbf{x}}{dt} = \phi(\mathbf{w} \cdot \mathbf{x}) - \mathbf{x}\,. \tag{6}$$

The first step is to discretize time, yielding

$$\mathbf{x}(t + dt) = \left(1 - \frac{dt}{\tau}\right)\mathbf{x}(t) + \frac{dt}{\tau}\phi\big(\mathbf{w}(t) \cdot \mathbf{x}(t)\big) \tag{7}$$

where equality really holds only in the $dt \to 0$ limit. We have written $\mathbf{w}(t)$ rather than just $\mathbf{w}$, because we're going to pretend that t labels layer.

Now compute the backprop error. However, for reasons that I don't totally understand, we need to use a slightly nonstandard method: using $\mathcal{L}$ for the loss, write

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}(t)} = \frac{\partial \mathcal{L}}{\partial \mathbf{x}(t + dt)} \cdot \frac{\partial \mathbf{x}(t + dt)}{\partial \mathbf{w}(t)} = \frac{dt}{\tau} \frac{\partial \mathcal{L}}{\partial \mathbf{x}(t + dt)} \odot \phi'\big(\mathbf{w}(t) \cdot \mathbf{x}(t)\big)\mathbf{x}(t) \tag{8}$$

where $\odot$ indicates element-wise multiplication. Consequently, we just need to write down an update rule for $\partial \mathcal{L}/\partial \mathbf{x}(t)$ in terms of $\partial \mathcal{L}/\partial \mathbf{x}(t + dt)$ and then add up all the terms using the above equation.

Your job is to complete the calculation, and then take the $dt \to 0$ limit to cast everything in terms of ODEs and integrals. If all goes well, you'll end up with the standard backprop through time rule.

Solution Our goal, first of all, is to express $\partial \mathcal{L}/\partial \mathbf{x}(t)$ in terms of $\partial \mathcal{L}/\partial \mathbf{x}(t + dt)$. To reduce clutter, define

$$\mathbf{z}(t) \equiv \frac{\partial \mathcal{L}}{\partial \mathbf{x}(t)}\,. \tag{9}$$

We then have

$$\mathbf{z}(t) = \mathbf{z}(t + dt) \cdot \frac{\partial \mathbf{x}(t + dt)}{\partial \mathbf{x}(t)} = \mathbf{z}(t + dt)\left(1 - \frac{dt}{\tau}\right) + \frac{dt}{\tau}\mathbf{z}(t + dt) \cdot \phi'\big(\mathbf{w}(t) \odot \mathbf{x}(t)\big) \cdot \mathbf{w}(t) \tag{10}$$

where, as above, $\odot$ denotes element-wise multiplication and the second equality came from Eq. (7). Then taking the limit $dt \to 0$, we can write a differential equation for $\mathbf{z}$,

$$\tau \frac{d\mathbf{z}(t)}{dt} = \mathbf{z}(t) \odot \phi'\big(\mathbf{w} \cdot \mathbf{x}(t)\big) \cdot \mathbf{w} - \mathbf{z}(t) \tag{11}$$

with the proviso that this must be integrated backwards in time. Note that $\mathbf{w}$ no longer depends on time, as we're reverting to our recurrent network.

Combining this with Eq. (8), averaging the gradient over time, and setting $\mathbf{w}(t) = \mathbf{w}$ for all t, we arrive at

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = \int \frac{dt}{\tau}\, \mathbf{z}(t) \odot \phi'\big(\mathbf{w}(t) \cdot \mathbf{x}(t)\big)\mathbf{x}(t)\,, \tag{12}$$

which leads to the standard backprop through time update rule.

3. The time-dependent firing rate of a neuron, $\nu(t)$, is given by

$$\nu(t) = \phi\left(\sum_i w_i s_i(t)\right)\,. \tag{13}$$

Here $s_i(t)$ is the filtered spike train of presynaptic neuron i,

$$s_i(t) = \sum_k g(t - t_i^k) \tag{14}$$

where t_i^k is the k^{th} spike on neuron i and $g(t)$ is a PSP (e.g., $g(t) = \tau^{-1} e^{-t/\tau}\Theta(t)$). Assume that the neuron fires with Poisson statistics, so the probability of a spike between times t and $t + dt$ is $\nu(t)dt$.

The neuron receives a teacher spike train; that is, a set of times t_{post}^k at which the neuron *should* fire. You want to adjust the rates to maximize the log probability of the teacher spike train given the weights. Show that a reasonable online learning rule for the weights is

$$\frac{dw_j}{dt} = \eta \left[\frac{\phi'\big(\sum_i w_i s_i(t)\big)}{\phi\big(\sum_i w_i s_i(t)\big)} \sum_k \delta(t - t_{\text{post}}^k) - \phi'\left(\sum_i w_i s_i(t)\right)\right] s_j(t)\,. \tag{15}$$

Solution The log probability of the spike train, L, is given by the usual expression,

$$L = \sum_k \log\big(\nu(t_{\text{post}}^k)\big) - \int dt\, \nu(t)\,. \tag{16}$$

Defining

$$\ell(t) \equiv \log \nu(t) \sum_k \delta(t - t_{\text{post}}^k) - \nu(t)\,, \tag{17}$$

this can be written

$$L = \int dt\, \ell(t)\,. \tag{18}$$

An online update rule for the weights, which tends to increase L, is,

$$\frac{dw_j}{dt} = \eta \frac{\partial \ell(t)}{\partial w_j} \tag{19}$$

This leads directly to Eq. (15).

4. Consider a linear feedforward network,

$$\mathbf{x}^l = \mathbf{W}^l \cdot \mathbf{x}^{l-1} \tag{20}$$

where l refers to layer. You want to add feedback weights,

$$\mathbf{x}^{l-1} = \mathbf{B}^l \cdot \mathbf{x}^l\,. \tag{21}$$

You start with random weights, and train them with a Hebb rule, plus weight decay,

$$\Delta \mathbf{B}^l = \eta(\mathbf{x}^{l-1}\mathbf{x}^l - \mathbf{B}^l) \tag{22}$$

where, as usual, two vectors next to each other denotes an outer product: $(\mathbf{x}^{l-1}\mathbf{x}^l)_{ij} = x_i^{l-1}x_j^l$. During learning, the input to each layer is white noise; so for layer $l-1$,

$$\langle \mathbf{x}^{l-1}\mathbf{x}^{l-1} \rangle = \mathbf{I} \tag{23}$$

where $\mathbf{I}$ is the identity matrix.

Show that under this learning rule, for small learning rate and a large number of update steps,

$$\mathbf{B}^l \approx (\mathbf{W}^l)^T \tag{24}$$

where T denotes transpose.

Why is it a good idea to use this learning rule?

Solution Inserting Eq. (20) into (22) yields

$$\Delta \mathbf{B}^l = \eta(\mathbf{x}^{l-1}\mathbf{x}^{l-1} \cdot {\mathbf{W}^l}^T - \mathbf{B}^l)\,. \tag{25}$$

In the limit of slow learning, we can average over the x^{l-1}, yielding

$$\Delta \mathbf{B}^l = \eta(\langle \mathbf{x}^{l-1}\mathbf{x}^{l-1} \rangle \cdot {\mathbf{W}^l}^T - \mathbf{B}^l) = \eta({\mathbf{W}^l}^T - \mathbf{B}^l)\,. \tag{26}$$

After a large number of training steps, $\mathbf{B}^l$ converges to ${\mathbf{W}^l}^T$. This exactly what we need for backprop.