Gatsby Computational Neuroscience Unit
Theoretical Neuroscience

Final examination, theoretical neuroscience
10 May 2022

Part II – long questions

There are four questions, one from each main section of the course. Please answer three out of the four, starting the answers for each question on a new page. Don't forget to write your name at the top of the answer to each question.

Good luck!

1 Biophysics

Consider a neuron with a single active channel,

$$\tau \frac{dv}{dt} = -v - gm(v - \mathcal{E}) \tag{1a}$$

$$\tau_m(v) \frac{dm}{dt} = -\big(m - m_1(v)\big)\big(m - m_2(v)\big)\big(m - m_3(v)\big)\,. \tag{1b}$$

Here v is the membrane potential relative to rest and $\mathcal{E}$ is the sodium channel reversal potential relative to rest, so it's about 80 mV. Assume $g \approx 40$, although that doesn't really matter much. The functions $m_1(v)$, $m_2(v)$ and $m_3(v)$ have the following properties:

- At $v = 40$ mV, $m_1(v) < m_2(v) < m_3(v)$.
- Generally, $m_1(v)$ and $m_3(v)$ are decreasing functions of v and $m_2(v)$ is an increasing function of v.
- But not always: when the m's collide, they annihilate, leaving only one fixed point. There are lot of ways to do that; a simple one is as follows: assuming m_1 and m_2 collide at $v = v_a$ (meaning $m_1(v_a) = m_2(v_a) \equiv m_a$) and m_2 and m_3 collide at v_b (meaning $m_3(v_b) = m_2(v_b) \equiv m_b$), then

$$v < v_a\text{: } \big(m - m_1(v)\big)\big(m - m_2(v)\big) \to (m - m_a)^2 + \alpha(v_a - v)^2 \tag{2a}$$

$$v > v_b\text{: } \big(m - m_3(v)\big)\big(m - m_2(v)\big) \to (m - m_b)^2 + \alpha(v - v_b)^2 \tag{2b}$$

  with $\alpha > 0$.

  The above formulation doesn't tell what happens when $v < v_a$ or $v > v_b$. What I should have written was

$$v \approx v_a\text{: } \big(m - m_1(v)\big)\big(m - m_2(v)\big) \to (m - m_a)^2 + \alpha(v_a - v) \tag{3a}$$

$$v \approx v_b\text{: } \big(m - m_3(v)\big)\big(m - m_2(v)\big) \to (m - m_b)^2 + \alpha(v - v_b) \tag{3b}$$

  with $\alpha > 0$.

1. The standard treatment of channels is to assume there are a large number of them, they are either open or closed, and they make random, and independent, transitions between those two states. In this problem we're assuming the channels can take on a continuum of values. Given that assumption, for the above equations to make sense we also have to assume: (1) all channels start in the same state, and (2) they obey identical dynamics. Why do we have to make those two assumptions?

   (5 marks)

Solution

Assume there are n channels, each obeying the equation

$$\tau_{mi}(v)\,\frac{dm_i}{dt} = -\big(m - m_{1i}(v)\big)\big(m - m_{2i}(v)\big)\big(m - m_{3i}(v)\big) \tag{4}$$

and the total channel conductance is

$$m(t) = \sum_i m_i(t)\,. \tag{5}$$

Because the system is bistable, the m_i can end up at different equilibria – either because they have different initial conditions, or because they obey different equations. This makes the equations effectively stochastic, in the sense that if we know $m(t)$ we don't know $m(t+dt)$. Which might not be a bad model of channels, but it's different than Eq. (1b).

2. Draw the nullclines (with V on the x-axis and m on the y-axis) for this setup, in a regime where the neuron is type II (meaning there is one fixed point which may or may not be stable, and the transition between stability and instability is via a Hopf bifurcation).

   (10 marks)

Solution

The v-nullcline is easy; it's

$$m = \frac{1}{g}\,\frac{v}{\mathcal{E} - v}\,. \tag{6}$$

It's shown in blue in Fig. 1. Given the description of the parameters $m_1(v)$, $m_2(v)$ and $m_3(v)$, the m-nullcline is z-shaped. To make the neuron type II, it has the shape shown in Fig. 1 (red curve).

To see that this is type II, write down the linearized dynamics around the fixed point, which occurs at (m_0, v_0). Using the fact that $m_0 = m_2(v_0)$, and letting $m = m_0 + \delta m$ and $v = v_0 + \delta v$, the linearized dynamics are

$$\frac{d}{dt}\begin{pmatrix}\delta v\\ \delta m\end{pmatrix} = \begin{pmatrix} -\frac{1+gm_2}{\tau} & \frac{g(\mathcal{E}-v_0)}{\tau} \\ -\frac{(m_3-m_2)(m_2-m_1)m_2'}{\tau_m} & \frac{(m_3-m_2)(m_2-m_1)}{\tau_m}\end{pmatrix}\begin{pmatrix}\delta v\\ \delta m\end{pmatrix} \tag{7}$$

where all v-dependent quantities should be evaluated at $v = v_0$ and a prime denotes a derivative. The fixed point is stable if the trace of the above matrix is negative and the determinant is positive. As is relatively easy to show, the fact that the m-nullcline intersects the v-nullcline from below, on the unstable branch of the m-nullcline, means the determinant is positive. Thus, stability is determined by the trace, which is given by

$$\text{trace} = \frac{(m_3 - m_2)(m_2 - m_1)}{\tau_m} - \frac{1 + gm_2}{\tau}\,. \tag{8}$$

The first term is positive and the second negative, and we can make either of them larger, and thus the trace either positive or negative, just by adjusting the time constants. Since the determinant is positive, a transition from negative (stable) to positive (unstable) occurs when the real part of the eigenvalue is zero. This is (one of the) definitions of a type II neuron.

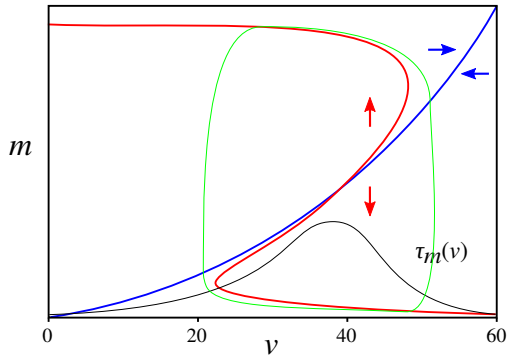

Figure 1: Nullclines in the type II regime. Blue is v-nullcline and red is m-nullcline. The black curve is the shape $\tau_m(v)$ must have to ensure bistability: a fixed point and a stable limit cycle (the latter shown in green).

3. Sketch, qualitatively, how $\tau_m(v)$ should depend on v to ensure bistability; that is, ensure a stable fixed point and a stable limit cycle.

   (10 marks).

   Solution

   As indicated in Eq. (8), the fixed point is stable if $\tau_m(v_0)$ is sufficiently large. If in addition $\tau_m(v)$ is small for small and large v, then at large and small v trajectories will be pulled quickly toward the m-nullcline and slowly toward the v-nullcline, yielding the green trajectory shown in Fig. 1. The resulting shape of $\tau_m(v)$ is sketched, qualitatively, in black in Fig. 1.

4. Draw the nullclines in a regime where the neuron is type I (meaning it goes unstable via a Saddle-node bifurcation), and exhibits all-or-none action potentials (meaning that when the neuron spikes, it's maximum voltage is above some minimum).

   (10 marks)

   Solution

   The nullclines are shown in Fig. 2. See the figure caption for details.

5. If you're in the type I regime (from question 4 above), which is more effective at getting the neuron to spike: excitatory input or inhibitory input? Justify your answer.

   (5 marks)

   Solution

   As shown in Fig. 2, you need inhibitory input.

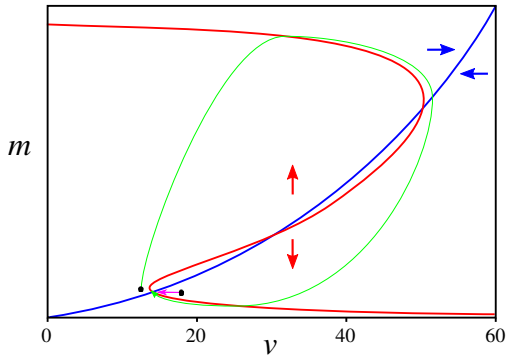

Figure 2: Nullclines in the type I regime. Blue is v-nullcline and red is m-nullcline. To ensure an all-or-none action potential, the right and left "knees" of the m nullcline have to be above and below, respectively, the v-nullcline, $\tau_m(v)$ has to be sufficiently small that the trajectories cross the upper branch of the m nullcline from below (as shown in the green trajectory), and the middle intersection has to be unstable. Black dots are initial conditions associated with brief excitatory input (right) and inhibitory input (left). To get the neuron to spike, it should receive inhibitory, rather than excitatory, input.

2 Networks

Consider a classical Hopfield network,

$$S_i(t+1) = \text{sign}\left(\sum_{j=1}^{n} J_{ij} S_j(t)\right) \tag{1}$$

with a standard-looking weight matrix,

$$J_{ij} \equiv \frac{1}{n}\sum_{\mu=1}^{p} \xi_i^\mu \xi_j^\mu \tag{2}$$

but slightly non-standard "memories"; the ξ_i^μ are not binary, but instead are drawn from a Gaussian distribution,

$$\xi_i^\mu \sim \mathcal{N}(0,1)\,. \tag{3}$$

1. Define the overlaps via

$$m_\mu(t) \equiv \frac{1}{n}\sum_j \xi_j^\mu S_j(t)\,. \tag{4}$$

Show that if $S_i(t)$ is set to one of the memories, $S_i(t) = \text{sign}(\xi_i^\mu)$, then, to leading order in n,

$$m_\mu(t) \equiv m_0 = \sqrt{\frac{2}{\pi}}\,. \tag{5}$$

(5 marks)

Solution

With $S_i(t) = \xi_i^\mu$, the overlap is given by

$$m_\mu(t) = \frac{1}{n}\sum_i \xi_i^\mu \text{sign}\big(\xi_i^\mu\big)\,. \tag{6}$$

In the large n limit, the sum over ξ_i^μ can be turned into an integral over the distribution of ξ_i^μ,

$$m_\mu(t) \approx \int_{-\infty}^{\infty} d\xi\, \frac{e^{-\xi^2/2}}{\sqrt{2\pi}}\, \xi\, \text{sign}(\xi_i) \equiv m_0\,. \tag{7}$$

The corrections are $\mathcal{O}(1/\sqrt{n})$, but we can ignore them in the large n limit (at least for now). The integral is straightforward,

$$m_0 = 2\int_0^\infty \frac{d\xi}{\sqrt{2\pi}}\, \xi e^{-\xi^2/s} = \sqrt{\frac{2}{\pi}}\left(-\,e^{-\xi^2/2}\Big|_{\xi=0}^{\xi=\infty}\right) = \sqrt{\frac{2}{\pi}}\,. \tag{8}$$

2. Again at time t we set the activity to one of the memories,

$$S_i(t) = \text{sign}(\xi_i^\mu)\,. \tag{9}$$

Show that the probability of a bit flip is given by (in the large p and n limit)

$$P\big(S_i(t+1) \neq S_i(t)\big) \equiv P_{\text{flip}} = 2\int_0^\infty d\xi\, \frac{e^{-\xi^2/2}}{\sqrt{2\pi}}\, H\left(-\frac{m_0}{\sigma_0}\xi\right) \tag{10}$$

where

$$\sigma_0^2 \equiv \frac{p}{n} \tag{11}$$

and $H(\cdot)$ is the cumulative normal function,

$$H(x) \equiv \int_{-\infty}^{x} d\xi\, \frac{e^{-\xi^2/2}}{\sqrt{2\pi}}\,. \tag{12}$$

(10 marks)

Solution

We'll start with an expression for $S_i(t+1)$ given that $S_i(t) = \xi_i^\mu$,

$$S_i(t+1) = \text{sign}\left(\xi_i^\mu \frac{1}{n}\sum_j \xi_j^\mu \text{sign}(\xi_j^\mu) + \frac{1}{n}\sum_{\nu\neq\mu,j} \xi_i^\nu \xi_j^\nu \text{sign}(\xi_j^\mu)\right]\,. \tag{13}$$

The first sum on j we've already computed: it's m_0, at least in the large n limit. The second is a Gaussian random variable. Since everything is independent, it's variance is just given by (number of terms) $\times$ (variance or each term) $= n(p-1)/n^2 = (p-1)/n \approx p/n$. Which, recall, is just σ_0^2. We thus have,

$$S_i(t+1) = \text{sign}\left(m_0 \xi_i^\mu + \sigma_0 \eta_i^\mu\right) \tag{14}$$

where η_i^μ is a zero mean, unit variance Gaussian random variable. The probability of a bit flip is thus given by

$$P_{\text{flip}} = 1 - \left[\int_0^\infty d\xi \, \frac{e^{-\xi^2/2}}{\sqrt{2\pi}} P\left(\eta_i^\mu > -m_0\xi/\sigma_0\right) + \int_{-\infty}^0 d\xi \, \frac{e^{-\xi^2/2}}{\sqrt{2\pi}} P\left(\eta_i^\mu < -m_0\xi/\sigma_0\right) \right] . \tag{15}$$

It's not hard to show that the two integrals are the same, and the probability is an integral over η, so we have

$$P_{\text{flip}} = 1 - 2\int_0^\infty d\xi \, \frac{e^{-\xi^2/2}}{\sqrt{2\pi}} \int_{-m_0\xi/\sigma_0}^\infty d\eta \, \frac{e^{-\eta^2/2}}{\sqrt{2\pi}} . \tag{16}$$

The second integral is $1 - H(-m_0\xi/\sigma_0)$, which leads (after a very small amount of algebra) to Eq. (10).

Extra credit (5 marks): do the integral, to show that

$$P_{\text{flip}} = \frac{\text{arccot}(m_0/\sigma_0)}{\pi} . \tag{17}$$

Solution

This is a good integral to know how to do; it comes up a lot.

Inserting the definition of the cumulative normal function into Eq. (10) gives

$$P_{\text{flip}} = 2\int_0^\infty d\xi \, \frac{e^{-\xi^2/2}}{\sqrt{2\pi}} \int_{-\infty}^{-m_0\xi/\sigma_0} d\eta \, \frac{e^{-\eta^2/2}}{\sqrt{2\pi}} . \tag{18}$$

This corresponds to a pizza-shaped region in $\xi - \eta$ space that subtends an angle whose cotangent is m_0/σ_0. To make this formal you should probably change to polar coordinates, but just sketching the region of integration in $\xi - \eta$ space (and taking into account the factor of 2 in front of the integral) will make it all clear.

3. Write down the limiting behavior of P_{flip} as $\sigma_0 \to 0$ and as $\sigma_0 \to \infty$. Explain why this behavior, which should be pretty simple, and make sense.

   (5 marks)

Solution

When $\sigma_0 \to 0$, $H(-m_0/\sigma_0) \to 0$, and the probability of a bit flip goes to zero. This make sense: $\sigma_0 \to 0$ means the number of memories, p, is small and the number of neurons, n, is large. In this regime there's almost no noise, and thus perfect recall.

When $\sigma_0 \to \infty$, $H(-m_0/\sigma_0) \to 1/2$, and the probability of a bit flip goes to 1/2. Again, this make sense: $\sigma_0 \to \infty$ means the number of memories, p, is much larger than the number of neurons, n – a very high noise regime.

Note that $\text{arccot}(\infty) = 0$ and $\text{arccot}(0) = \pi/2$, so Eq. (17) is correct in these limits.

4. Knowing the 1-step error doesn't tell much about the error after a long time. To assess that, we need to consider how the overlaps, m_μ (Eq. (4)) behave. Show that they obey the update rule

$$m_\mu(t+1) = \frac{1}{n}\sum_j \xi_i^\mu \text{sign}\left(\xi_i^\mu m_\mu(t) + \sum_{\nu\neq\mu} \xi_i^\nu m_\nu(t)\right) . \tag{19}$$

(5 marks)

Solution

Using the definition of the weight matrix, Eq. (2), we have

$$\sum_j J_{ij}S_j(t) = \sum_\nu \xi_i^\mu \frac{1}{n}\sum_j \xi_j^\mu S_j(t) = \sum_\nu \xi_i^\nu m_\nu(t) . \tag{20}$$

Inserting this into the update rule, Eq. (1), and operating on both sides with $n^{-1}\sum_i \xi_i^\mu$ yields the desired expression.

5. Assume the second term inside the sign in Eq. (19) is a Gaussian random variable with variance $\sigma^2(t)$, it's uncorrelated with ξ_i^μ, and that m_μ is $\mathcal{O}(1)$. Show that in the large n limit,

$$m_\mu(t+1) = \sqrt{\frac{2}{\pi}} \frac{m_\mu(t)}{\left[\sigma^2(t) + m_\mu^2(t)\right]^{1/2}} . \tag{21}$$

Thus, the asymptotic value of the overlap is

$$m_\mu(\infty) = \sqrt{\frac{2}{\pi}} - \sigma^2(\infty) . \tag{22}$$

Thus, to find the final overlap, we just need to compute the asymptotic value of $\sigma^2(t)$.

(10 marks)

Solution

With σ known, the update rule for m_μ is

$$m_\mu(t+1) = \frac{1}{n}\sum_j \xi_i^\mu \text{sign}\left(\xi_i^\mu m_\mu(t) + \sigma(t)\eta_i\right) \tag{23}$$

where η_i is a zero mean, unit variance Gaussian random variable, and η_i and ξ_i^μ are independent. Thus, in the large n limit, we may write

$$m_\mu(t+1) = \int d\eta \frac{e^{-\eta^2/2}}{\sqrt{2\pi}} \int d\xi \frac{e^{-\xi^2/2}}{\sqrt{2\pi}} \xi \,\text{sign}(\xi m_\mu(t) + \sigma(t)\eta) . \tag{24}$$

Because of the sign function, the second integral separates into two,

$$m_\mu(t+1) = \int d\eta \, \frac{e^{-\eta^2/2}}{\sqrt{2\pi}} \left(\int_{-\sigma(t)/m_\mu(t)}^{\infty} d\xi \, \frac{e^{-\xi^2/2}}{\sqrt{2\pi}} \, \xi - \int_{-\infty}^{\sigma(t)/m_\mu(t)} d\xi \, \frac{e^{-\xi^2/2}}{\sqrt{2\pi}} \, \xi \right) . \tag{25}$$

Both can be done analytically (and they're the same), so that leaves an integral over η,

$$m_\mu(t+1) = \sqrt{\frac{2}{\pi}} \int d\eta \, \frac{e^{-\eta^2/2}}{\sqrt{2\pi}} e^{-\sigma^2(t)\eta^2/2m_\mu^2(t)} . \tag{26}$$

The last integral is straightforward, and we arrive at

$$m_\mu(t+1) = \sqrt{\frac{2}{\pi}} \, \frac{1}{[1+\sigma^2(t)/m_\mu^2(t)]^{1/2}} . \tag{27}$$

This is the same as Eq. (21), so long as $m_\mu(t)$ is positive. In fact, it's not hard to show that this is also the correct expression when $m_\mu(t)$ is negative. (Our derivation actually assumed that $m_\mu(t) > 0$. However, it's easily extended to the case $m_\mu(t) < 0$.)

6. We now just need to compute $\sigma^2(t)$, and we'll know everything about this system! By definition, this quantity is given by

$$\sigma^2(t) = \mathrm{Var}\left[\sum_{\nu \neq \mu} \xi_i^\nu m_\nu(t) \right] = \mathrm{Var}\left[\sum_{\nu \neq \mu} \xi_i^\nu \frac{1}{n} \sum_j \xi_j^\nu S_j(t) \right] . \tag{28}$$

If ξ_i^ν and $S_j(t)$ were uncorrelated, we could easily compute the variance. But they are correlated. Why?

(5 marks)

Solution

To answer this, we write with the equation for the equilibrium (basically, the update rule, Eq. (1), but without the time dependence) in the form

$$S_i = \mathrm{sign}\left(\xi_i^\mu + \sum_{\nu \neq \mu} \xi_i^\nu \frac{1}{n} \sum_j \xi_j^\nu S_j \right) . \tag{29}$$

Correlations between S_i and ξ_i^ν arise because of the factor of ξ_i^ν that appears on the right hand side, inside the sign function. To quantify the correlations, one can define

$$z_i \equiv \sum_{\nu \neq \mu} \xi_i^\nu \frac{1}{n} \sum_j \xi_j^\nu S_j . \tag{30}$$

Then, z_i obeys the equation

$$z_i = \sum_{\nu \neq \mu} \xi_i^\nu \frac{1}{n} \sum_j \xi_j^\nu \mathrm{sign}\left(\xi_i^\mu m_\mu + z_i \right) . \tag{31}$$

This could be the starting point of a mean field theory that will eventually determine the capacity of this network. See chapter 2 of "Introduction to the theory of neural computation" by Hertz, Krogh, and Palmer (1991) for an analysis of the capacity of the standard Hopfield network using this method.

3 Coding

Consider a population of neurons that encodes a single angular variable θ, whose background distribution is non-uniform with cumulative distribution function $F(\theta)$ [define the CDF as the integral from 0°]. An example might be the orientation of a visual contour, where vertical and horizontal orientations tend to be overrepresented in natural images.

Ganguli and Simoncelli have suggested that, when faced with limited and noisy encoding resources, a population code will have minimal average decoding error when it is homogeneous on a histogram-equalising transform of θ. That is, when the neural tuning functions

$$f_i(\theta) = f(F(\theta) - F(\theta_i))$$

for a fixed (usually symmetric) function f, and with θ_i chosen to make the values $F(\theta_i)$ evenly spaced.

An alternative view is that encoded variable is, in fact, $u = F(\theta)$. As the representation of an internal variable is potentially arbitrary, it is plausible that the representation converges naturally on the histogram-equalised transform.

Let $f_i(u) = r_{\max} e^{\kappa \cos(u - u_i)}$ be von Mises tuning curves on the equalised variable and assume that the ith neuron's spike count, n_i, is Poisson distributed with rate $f_i(u)$.

1. Show that the likelihood function $P(\mathbf{n}|u)$ is von Mises in u ($\mathbf{n}$ is the vector of all spike counts). How do the mean and concentration (i.e. the effective κ) depend on the counts $\mathbf{n}$?

Solution

The von Mises can be written in vector form:

$$f_i(u) = r_{\max} e^{\boldsymbol{\kappa}_i^{\mathsf{T}} \mathbf{u}}$$

where $\boldsymbol{\kappa}_i = \kappa[\cos(u_i), \sin(u_i)]$ and $\mathbf{u} = [\cos(u), \sin(u)]$.

$$P(n_i|u) = e^{n_i(\boldsymbol{\kappa}_i^{\mathsf{T}}\mathbf{u} + \log r_{\max}) - f_i(u)} / n_i!$$

$$\Rightarrow P(\mathbf{n}|u) \propto \exp\left(\left(\sum_i n_i \boldsymbol{\kappa}_i\right)^{\mathsf{T}} \mathbf{u} - \sum_i f_i(u)\right)$$

$$\Rightarrow P(\mathbf{n}|u) \propto e^{\bar{\boldsymbol{\kappa}}^{\mathsf{T}} \mathbf{u}}$$

so mean is n_i-weighted circular mean and $\bar{\kappa} = \kappa \sum_i n_i$.

2. Show that the distribution on θ (i.e. the physical angle in degrees rather than the probability-equalised one) obtained by mapping a normalised version of this likelihood through F^{-1} incorporates the background distribution on θ as a prior.

Consider the case of visual orientation, focusing on angles close to horizontal (i.e. 0°). Locally, we assume a Laplace-like background distribution of angles $p(\theta) \propto e^{-|\theta|/\tau}$ (we won't require an exact form). Suppose that a stimulus is presented rotated very slightly anticlockwise from horizontal. Assume that the behavioural report of visual angle is based on decoding from a population code as above.

3. Suppose first that the report is based on the mean value of u under the (normalised) likelihood $P(\mathbf{n}|u)$. Sketch the resulting distribution of responses in terms of physical angle θ. Explain clearly how you arrived at the sketch.

4. Now assume that the report is based on the mean value of θ under the distribution mapped to physical angle as in part (2). Sketch the distribution of response you expect in this case. Again, explain your reasoning.

5. Discuss whether (and, if so, how) the two response strategies might be distinguishable.

4 Learning

In an acquired equivalence learning paradigm, different stimuli (two different face images, say) are paired with the same outcome (usually reward). Then neural representations are measured to see whether the representations of each stimulus have become more or less similar after this pairing. In an acquired distinctiveness paradigm, different stimuli are shown to lead to different outcomes (face image A might predict the subsequent appearance of fish image X, while face image B might predict fish image Y). Classical findings show that representations become more similar when they predict similar outcomes; and become more dissimilar when they predict distinct outcomes. Here we will investigate these dynamics in a simple setting.

Consider two items, each represented by a three-dimensional feature column vector $x^\mu \in R^3, \mu = 1, 2$. The first feature will be shared across items, and can be interpreted as, for instance, an "is face" feature. The second and third features will be distinctive, encoding exactly which face. Concatenating these vectors into the matrix $X \in R^{3\times 2}$, we have the input dataset

$$X = \begin{bmatrix} a & a \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{1}$$

where the parameter $a \geq 0$ controls the 'salience' of the shared feature relative to the distinctive features (which are set to one to fix the scale).

We will consider two different target output settings that instantiate simple versions of acquired equivalence and acquired distinctiveness paradigms. Each input vector x^μ will be associated with a two-dimensional output column vector $y^\mu \in R^2$, where each element might represent a different image that can be presented subsequently (fish image X and fish image Y in the above example). In the *functional equivalence* setting we take the output dataset Y_e to be

$$Y_e = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}. \tag{2}$$

That is, we'll pair both inputs with the same output (i.e. face image A and face image B predicts fish image X).

In the *functional distinctiveness* setting we take the output dataset Y_d to be

$$Y_d = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{3}$$

That is, we'll pair each input with a distinct output (i.e. face image A predicts fish image X and face image B predicts fish image Y).

1. Recall that for a neuron with output $h = w \cdot x$ where $w \in R^N$ is its weight vector, Oja's rule (with parameter $\alpha = 1$) is

$$\tau \frac{d}{dt} w = hx - h^2 w. \tag{4}$$

What weight vector does Oja's rule converge to as a function of a? At what value of a does Oja's rule place more weight on the shared feature than either distinctive feature?

Solution

Oja's rule converges to the eigenvector associated with the largest eigenvalue of the input correlation matrix

$$\langle xx^T \rangle = \frac{1}{2} \begin{bmatrix} 2a^2 & a & a \\ a & 1 & 0 \\ a & 0 & 1 \end{bmatrix}. \tag{5}$$

From the symmetry of the input vectors, we can guess that one eigenvector will have the form $v = \begin{bmatrix} c & d & d \end{bmatrix}^T$. From the definition of eigenvectors,

$$\langle xx^T \rangle v = \lambda v \tag{6}$$

$$\begin{bmatrix} a^2 c + da \\ \frac{ac+d}{2} \\ \frac{ac+d}{2} \end{bmatrix} = \lambda \begin{bmatrix} c \\ d \\ d \end{bmatrix}. \tag{7}$$

Solving one of the last two equations for c yields

$$c = (2\lambda - 1)d/a. \tag{8}$$

Inserting this into the first equation we have

$$a^2(2\lambda - 1)d/a + da = \lambda(2\lambda - 1)d/a \tag{9}$$

$$0 = 2\lambda^2 - (2a^2 - 1)\lambda \tag{10}$$

$$\lambda = \frac{(2a^2 - 1) \pm \sqrt{(2a^2 - 1)^2}}{4} \tag{11}$$

$$\lambda = a^2 + \frac{1}{2} \text{ or } 0, \tag{12}$$

where in going from Eqn. (9) to Eqn. (10) we assumed that $d \neq 0$. Finally we can find an eigenvector by taking $d = 1$ and substituting $\lambda = a^2 + \frac{1}{2}$,

$$c = (2\lambda - 1)d/a \tag{13}$$

$$= 2a. \tag{14}$$

Hence

$$v_1 = \frac{1}{4a^2 + 2} \begin{bmatrix} 2a \\ 1 \\ 1 \end{bmatrix} \tag{15}$$

is a unit-norm eigenvector with eigenvalue $\lambda_1 = a^2 + 1/2$.

Similarly, by symmetry of the inputs, we can guess that another eigenvector has the form $v_2 = \begin{bmatrix} 0 & 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix}$ which can be verified by direct calculation and shown to have eigenvalue $\lambda_2 = 1/2$. Further, we have $v_1 \cdot v_2 = 0$ so these are orthogonal. Because there are

only two inputs, the input correlation matrix is at most rank 2 and all other eigenvalues must be zero and can be ignored.

Because $\lambda_1 > \lambda_2$ for $a > 0$, we know that under Oja's rule the weights will converge to v_1 for $a > 0$. For $a = 0$, Oja's rule will converge to a linear combination of v_1 and v_2, $w = c_1 v_1 + c_2 v_2$ where $c_1^2 + c_2^2 = 1$. More weight will be placed on the shared component when $a > 1/2$.

Now consider a deep linear network. In response to an input vector $x \in R^3$, the deep linear network produces an output vector $\hat{y} \in R^2$ according to $\hat{y} = W_2 W_1 x$, where the weight matrices are $W_2 \in R^{2 \times N_h}$ and $W_1 \in R^{N_h \times 3}$ and $N_h \geq 3$ is the number of hidden units (our "deep" network has just one hidden layer which Hinton wouldn't count as deep but we will). The weights are updated after each step to minimize the mean squared error over the dataset

$$\mathcal{L} = \frac{1}{2} \left\langle ||y - \hat{y}||_2^2 \right\rangle \tag{16}$$

using continuous time gradient flow

$$\tau \dot{W}_l = -\frac{\partial \mathcal{L}}{\partial W_l} \quad \text{for } l = 1, 2. \tag{17}$$

2. Compute the SVD of the input-output correlation matrix, $\langle y x^T \rangle = U S V^T$ for the functional equivalence and distinctiveness datasets introduced above. You only need to do this for the non-zero singular values. Show that the input correlation matrix is diagonalized by the right singular vectors, that is, $\langle x x^T \rangle = V D V^T$ for a diagonal matrix D.

   Solution

   The correlation matrices are

$$\frac{1}{2} Y_e X^T = \begin{bmatrix} a & 1/2 & 1/2 \end{bmatrix} \tag{18}$$

$$\frac{1}{2} Y_d X^T = \frac{1}{2} X^T \tag{19}$$

   in the functional equivalence and distinctiveness settings, respectively. In the functional equivalence setting, the (incomplete) SVD is thus

$$\frac{1}{2} Y_e X^T = U S V^T \tag{20}$$

   where

$$U = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{21}$$

$$S = \sqrt{a^2 + \frac{1}{2}} \tag{22}$$

$$V^T = \frac{1}{\sqrt{a^2 + \frac{1}{2}}} \begin{bmatrix} a & 1/2 & 1/2 \end{bmatrix}. \tag{23}$$

Comparing back to the previous question, we note that $V^T = v_1^T$, the principal eigenvector of the input correlation matrix.

In the functional distinctiveness setting, by definition of the SVD we need to find the eigenvectors of $1/4X^TX$ and $1/4XX^T$, that is, half the input correlation matrix. Using the results previously from Oja's rule and noting the form of U (this can be seen because X^TX is an ultrametric matrix, which is diagonalized by the Haar wavelets; by guessing and checking; or by directly calculating it from the 2x2 matrix $1/4X^TX$), we have

$$U = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix} \tag{24}$$

$$S = \begin{bmatrix}\sqrt{a^2/2+\frac{1}{4}} & 0\\ 0 & 1/2\end{bmatrix} \tag{25}$$

$$V^T = \begin{bmatrix}v_1^T\\ v_2^T\end{bmatrix} \tag{26}$$

where the singular values can be found by noting that they are the square root of half the eigenvalues found in the Oja's rule section.

Because V contains eigenvectors of the input correlation matrix, it is clear that it will also diagonalize the input correlations. In particular we have

$$\langle xx^T\rangle = V\begin{bmatrix}a^2+\frac{1}{2} & 0\\ 0 & 1/2\end{bmatrix}V^T. \tag{27}$$

3. Using the change of variables $W_1(t) = R\bar{W}_1(t)V^T, W_2(t) = U\bar{W}_2(t)R^T$, where R is an arbitrary orthogonal matrix ($R^TR = I$), show that the gradient descent dynamics can be written as

$$\tau\frac{d}{dt}\bar{W}_1 = \bar{W}_2^T(S - \bar{W}_2\bar{W}_1D) \tag{28}$$

$$\tau\frac{d}{dt}\bar{W}_2 = (S - \bar{W}_2\bar{W}_1D)\bar{W}_1^T \tag{29}$$

Solution

The gradient descent dynamics are

$$\tau\frac{d}{dt}W_1 = W_2^T\left[\langle yx^T\rangle - W_2W_1\langle xx^T\rangle\right] \tag{30}$$

$$\tau\frac{d}{dt}\left(R\bar{W}_1(t)V^T\right) = R\bar{W}_2(t)^TU^T\left[USV^T\right. \tag{31}$$

$$\left.-U\bar{W}_1(t)R^TR\bar{W}_1(t)V^TVDV^T\right] \tag{32}$$

$$R\left(\tau\frac{d}{dt}\bar{W}_1(t)\right)V^T = R\bar{W}_2(t)^T\left[S - \bar{W}_2(t)\bar{W}_1(t)D\right]V^T \tag{33}$$

$$\tau\frac{d}{dt}\bar{W}_1(t) = \bar{W}_2(t)^T\left[S - \bar{W}_2(t)\bar{W}_1(t)D\right] \tag{34}$$

and a similar derivation shows

$$\tau \frac{d}{dt} \bar{W}_2 = \left[S - \bar{W}_2 \bar{W}_1 D \right] \bar{W}_1^T. \tag{35}$$

4. Assume a balanced, decoupled initialization such that $\bar{W}_1(0)$ and $\bar{W}_2(0)$ are diagonal and the diagonal elements are equal, $[\bar{W}_1(0)]_{ii} = [\bar{W}_2(0)]_{ii}, \forall i$. Provided all diagonal elements start non-zero, what will $\bar{W}_1(0)$ and $\bar{W}_2(0)$ converge to? (You don't have to compute the full time-course, just the final fixed point.)

   Solution

   By assumption $\bar{W}_2(0) = \bar{W}_1(0)$ and both are diagonal. Thus the dynamics decouple and each diagonal element $b_\alpha(t) = \left[\bar{W}_1(t)\right]_{\alpha\alpha}$ evolves independently with dynamics

$$\tau \frac{d}{dt} b_\alpha = b_\alpha \left[s_\alpha - b_\alpha^2 d_\alpha \right] \tag{36}$$

   where $s_\alpha = S_{\alpha\alpha}$ is the α^{th} singular value and $d_\alpha = D_{\alpha\alpha}$ is the α^{th} input variance. Provided $b_\alpha(0) \neq 0$, steady state occurs when

$$b_\alpha = \sqrt{s_\alpha / d_\alpha} \tag{37}$$

   or in matrix form

$$\lim_{t\to\infty} \bar{W}_2(t) = \lim_{t\to\infty} \bar{W}_1(t) = S^{1/2} D^{-1/2}. \tag{38}$$

5. What total weight vector $W_{tot} = W_2 W_1$ will the deep linear network converge to in the functional equivalence case and in the functional distinctiveness case in this setting? How much weight is placed on the shared feature compared to the maximally-weighted distinctive feature as a function of a? Compare this to your result from Oja's rule.

   Solution

   The total weight vector converges to

$$\lim_{t\to\infty} W_{tot}(t) = \lim_{t\to\infty} W_2(t) W_1(t) \tag{39}$$

$$= USD^{-1}V^T. \tag{40}$$

   In the functional equivalence case this yields

$$\lim_{t\to\infty} W_{tot}(t) = \frac{1}{\sqrt{a^2 + \frac{1}{2}}} \begin{bmatrix} & v_1^T & \\ 0 & 0 & 0 \end{bmatrix} \tag{41}$$

   such that the weight vector lies in the same direction as Oja's rule, and places more weight on the shared feature when $a > 1/2$.

In the functional distinctiveness case,

$$s_1/d_1 = \frac{\sqrt{a^2/2 + 1/4}}{a^2 + 1/2} \tag{42}$$

$$= \frac{\sqrt{2a^2+1}}{2a^2+1} \tag{43}$$

$$= \frac{1}{\sqrt{2a^2+1}} \tag{44}$$

and so

$$SD^{-1} = \begin{bmatrix} \frac{1}{\sqrt{2a^2+1}} & 0 \\ 0 & 1 \end{bmatrix}. \tag{45}$$

The total weight vector is therefore

$$\lim_{t\to\infty} W_{tot}(t) = USD^{-1}V^T \tag{46}$$

$$= \begin{bmatrix} \frac{1}{\sqrt{2}\sqrt{2a^2+1}} v_1^T + \frac{1}{\sqrt{2}} v_2^T \\ \frac{1}{\sqrt{2}\sqrt{2a^2+1}} v_1^T - \frac{1}{\sqrt{2}} v_2^T \end{bmatrix} \tag{47}$$

$$= \begin{bmatrix} \frac{a}{2a^2+1} & \frac{1}{2}\left(\frac{1}{2a^2+1} + 1\right) & \frac{1}{2}\left(\frac{1}{2a^2-1} - 1\right) \\ \frac{a}{2a^2+1} & \frac{1}{2}\left(\frac{1}{2a^2+1} - 1\right) & \frac{1}{2}\left(\frac{1}{2a^2+1} + 1\right) \end{bmatrix} \tag{48}$$

$$= \frac{1}{2a^2+1} \begin{bmatrix} a & a^2+1 & -a^2 \\ a & -a^2 & a^2+1 \end{bmatrix}. \tag{49}$$

From this we see that the magnitude of the weight on the shared feature will only be larger than the largest distinctive feature weight if $a > a^2 + 1$. This cannot occur because the roots of $a^2 - a + 1$ are imaginary, and testing any point reveals that the inequality does not hold. Hence the shared feature always receives less weight than the maximum distinctive feature, regardless of the choice of a.

Compared to the case with Oja's rule, we see that the functional equivalence setting behaves nearly identically: the asymptotic weight vector is aligned with Oja's rule, and it places greatest weight on the shared feature once $a \geq 1/2$. In contrast, in the functional distinctiveness setting, the weight vector never places more weight on the shared feature.

6. One way to compare representations is using representational similarity analysis (in machine learning this is known as the kernel matrix): we can compute pairwise dot products between all feature vectors. What is the RSA matrix $\Omega_X = X^T X$ of the raw inputs? What is the RSA matrix $\Omega_H = H^T H$ of the hidden representation $H = W_1 X$ for the deep linear network in the functional distinctiveness and equivalence settings? To make the comparison insensitive to the scale of the representation, divide the off diagonal element by the on diagonal elements (so we're comparing cosine angles). Does the deep linear network exhibit acquired equivalence and distinctiveness relative to the pre-existing similarity in the input features?

Solution

The RSA of the input matrix is

$$\Omega_X = X^T X = \begin{bmatrix} a^2+1 & a^2 \\ a^2 & a^2+1 \end{bmatrix} \tag{50}$$

and dividing the off-diagonal by the on-diagonal element, the cosine angle θ_x between examples is

$$\theta_x = \frac{a^2}{a^2+1}.$$

In the functional equivalence setting we have

$$\begin{aligned} \Omega_H &= H^T H && (51)\\ &= X^T W_1^T W_1 X && (52)\\ &= X^T V S^{1/2} D^{-1/2} R^T R S^{1/2} D^{-1/2} V^T X && (53)\\ &= X^T V S D^{-1} V^T X && (54)\\ &= \frac{1}{a^2+1/2} X^T v_1 v_1^T X && (55)\\ &= (a^2+1/2) \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} && (56) \end{aligned}$$

and the cosine angle θ_e is always one.

In the functional distinctiveness setting we have

$$\begin{aligned} \Omega_H &= X^T V S D^{-1} V^T X && (57)\\ &= X^T V \begin{bmatrix} \frac{1}{\sqrt{2a^2+1}} & 0 \\ 0 & 1 \end{bmatrix} V^T X && (58) \end{aligned}$$

and

$$\begin{aligned} V^T X &= \begin{bmatrix} v_1^T \\ v_2^T \end{bmatrix} X && (59)\\ &= \begin{bmatrix} \sqrt{a^2+\frac{1}{2}} & \sqrt{a^2+\frac{1}{2}} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix} && (60) \end{aligned}$$

such that

$$\begin{aligned} \Omega_H &= \begin{bmatrix} \frac{a^2+1/2}{\sqrt{2a^2+1}} + 1/2 & \frac{a^2+1/2}{\sqrt{2a^2+1}} - 1/2 \\ \frac{a^2+1/2}{\sqrt{2a^2+1}} - 1/2 & \frac{a^2+1/2}{\sqrt{2a^2+1}} + 1/2 \end{bmatrix} && (61)\\ &= \frac{1}{2} \begin{bmatrix} \sqrt{2a^2+1}+1 & \sqrt{2a^2+1}-1 \\ \sqrt{2a^2+1}-1 & \sqrt{2a^2+1}+1 \end{bmatrix} && (62) \end{aligned}$$

and the cosine angle θ_d between examples is

$$\theta_d = \frac{\sqrt{2a^2+1}-1}{\sqrt{2a^2+1}+1} \tag{63}$$

Finally we can compare the similarity between examples in each case. The cosine angle in the raw input is $\frac{a^2}{a^2+1}$ which is always less than one. Hence the functional equivalence setting exhibits acquired equivalence, in which internal representations become more similar than the raw inputs are through learning. Next we can divide the cosine angle of the functional distinctiveness setting by that of the raw inputs,

$$\theta_d/\theta_x = \frac{\frac{\sqrt{2a^2+1}-1}{\sqrt{2a^2+1}+1}}{\frac{a^2}{a^2+1}} \tag{64}$$

$$= \frac{a^2+1}{a^2+1+\sqrt{2a^2+1}}. \tag{65}$$

From this it is clear that $\theta_d < \theta_x$ for all a, and the functional distinctiveness setting exhibits acquired distinctiveness, in which internal representations become less similar than the raw inputs are through learning.