3 Non-normal Linear Time Invariant System

Let us continue to develop more intuition on the memory properties of linear "recurrent" neural networks of the form (3). As discussed in Goldman [4], some linear network models can be viewed as feedforward, while others can be feedback. This distinction is very important for the memory properties of the system. The traditional tool for analyzing memory traces in a neural system is to use the eigenvalues (also known as eigenspectrum). In this section, we will still stick with discrete time.

33. **Similarity transform and change of basis**: Let M be any invertible matrix. Consider a linear system over discrete time, $\mathbf{x}_{t+1} = A\mathbf{x}_t$. A change of variable $\mathbf{y}_t = M\mathbf{x}_t$ is a linear trnasformation, or a change of basis. Show that $B = M^{-1}AM$ is the new dynamics matrix for $\mathbf{y}_t$. (B is said to be *"similar"* to A)

34. Consider a square matrix A. Let $B = M^{-1}AM$ be a similarity transform of A. Show that similar matrices A and B have the same eigenvalues.

3.1 Diagonalization

Non-normal matrices

$$W^\top W \neq WW^\top$$

The spectral theorem is nice because it decouples each dimension, but can we extend such analysis to non-normal matrices?

Diagonalization

If a square matrix has n linearly independent (not necessarily orthogonal) eigenvectors, we can diagonalize them using its eigenvectors.

$$A = S\Lambda S^{-1} \qquad \text{(diagonalization)} \tag{32}$$

where Λ is a diagonal matrix of complex numbers.

35. Verify that the columns of S are (right) eigenvectors of A.

As long as the set of eigenvectors span the whole space, we can use (32) in reverse to construct matrices with specific eigenvectors and eigenvalues. For example, take $\mathbf{v}_1 = [1, 0]^\top$, $\mathbf{v}_2 = [1, 1]^\top/\sqrt{2}$, $\lambda_1 = 1$, and $\lambda_2 = 2$. Let $S = [\mathbf{v}_1, \mathbf{v}_2]$ and $\Lambda = \operatorname{diag}(\lambda_1, \lambda_2)$. We get the following non-normal matrix:

$$W_a = S\Lambda S^{-1} = \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix} \tag{33}$$

Let W be diagonizable, and $\mathbf{x}(t+1) = W\mathbf{x}(t)$. Let $\mathbf{v}_i$ be eigenvectors of W.

$$\mathbf{x}(t) = (S\Lambda S^{-1})^k \mathbf{x}(0) = (S\Lambda S^{-1}) \cdots (S\Lambda S^{-1})\mathbf{x}(0) = S \begin{bmatrix} \lambda_1^k & & \\ & \ddots & \\ & & \lambda_n^k \end{bmatrix} S^{-1}\mathbf{x}(0) \tag{34}$$

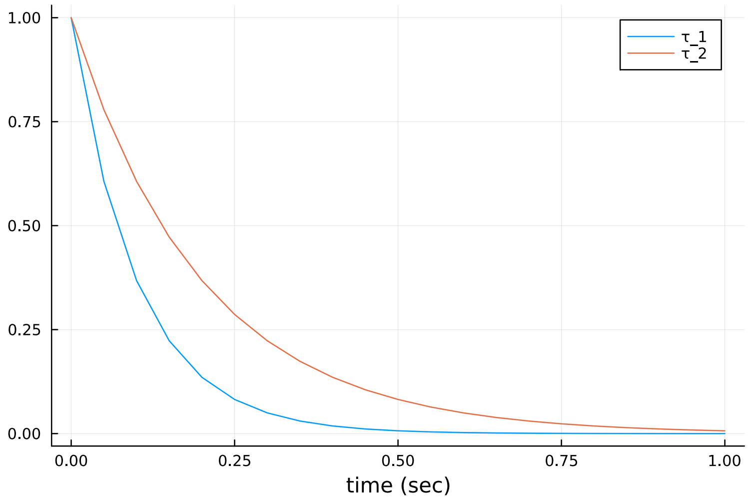
36. Show that $\mathbf{x}(t) = S\lambda_i^t$ if initialized at the eigenvector, i.e., $\mathbf{x}(0) = \mathbf{v}_i$.

This means dynamics decomposed onto the eigenmodes (spanned by $\mathbf{v}_i$) behave independently and geometrically. This is almost as nice as the spectrally decomposed normal matrix case. They just need to be combined back with S which is not an *orthonormal* basis.

3.2 Difference of exponentials is usually not an exponential

37. Let $\tau_1 = 0.1$ and $\tau_2 = 0.2$, sketch the following difference of exponential decay (in continuous time):

$$f(t) = e^{-t/\tau_1} - e^{-t/\tau_2} \tag{35}$$



38. Explain how two eigenmodes that independently exhibit exponential (geometric) decay can manifest in seemingly transient amplification in a different reference frame (e.g. a single neuron activity).

3.3 Purely feedforward networks

If there are no loops in the graph of the neural circuit, the neural activity of a neuron at time t will never influence itself later (at least within the circuit). Let us call such neural circuits **purely feedforward**.

39. Consider a network of 3 neurons where neuron 1 excites neuron 2 and neuron 2 excites neuron 3? Draw the neural circuit graph. What is the corresponding connectivity matrix?

40. Show that the connectivity matrix of any purely feedforward network can be written as a strictly triangular form (by choosing a certain ordering of neurons; recall that reordering neurons is just a multiplication by a permutation matrix).

3.4 Neural computation as a system

Given an input signal $u(t)$, a **system** outputs a corresponding output signal $y(t)$:

$$\{y(s), s = 0, 1, \ldots\} = H\{u(s), s = 0, 1, \ldots\} \tag{36}$$

where $H : (\mathbb{N} \to \mathbb{R}) \to (\mathbb{N} \to \mathbb{R})$ is an operator that denote the action of a system. Note that this is not necessarily an instantaneous mapping, but rather maps a time series to a time series. For simplicity, we will abuse the notation a little bit and simply write,

$$y(t) = H\{u(t)\}. \tag{37}$$

Typically we consider only systems that are *causal*, that is, output is produced through a causal mechanism and does not depend on the future input nor output. A causal system can capture the essence of biophysical computation at various scales–a transformation of signals over time. This transformation may also depend on the internal state of the neural system in general. However, for the simplicity, let us consider systems without an internal state.

A system is said to be **time-invariant** if its response to a certain signal does not depend on (absolute) time.

$$H\{u(t + \tau)\} = y(t + \tau) \tag{38}$$

which shows that time shift of the input of the system is the same as the output shifted by the same amount of time. Note that if a system is changing over time, this is not necessarily true; for example, with time evolution of the unobserved internal state of the system induced by synaptic plasticity or neuromodulators.

A system is said to be **linear**, if linear combination of input leads to corresponding linear combination of outputs,

$$H\{a \cdot u_1(t) + b \cdot u_2(t)\} = a \cdot y_1(t) + b \cdot y_2(t) \tag{39}$$

where $y_i(t) = H\{u_i(t)\}$ for $i = 1, 2$. Most neural systems are nonlinear, but linear analysis often provides a powerful baseline of understanding.

41. Let $y(t) = u(t - 1)$ be a system. It *delays* the input by 1 time step. For example, this delay could represent conduction delay of an axon. Verify that this system is causal, linear, and time-invariant.

42. Let $y(t) = \frac{1}{3}\left(u(t) + u(t - 1) + u(t - 2)\right)$ be a system. It averages the input of the most recent 3 time steps, producing a smoothed output. Verify that this *moving average* system is causal, linear, and time-invariant.

A linear neural network can be seen as a system. For the simple case where it takes a scalar time series $u(t)$ as input and produces a scalar time series $y(t)$ as output, we can use,

$$\mathbf{x}(t+1) = W\mathbf{x}(t) + \mathbf{b}^\top u(t) \tag{40}$$

$$y(t) = \mathbf{c}^\top \mathbf{x}(t) \tag{41}$$

where $\mathbf{b}$ and $\mathbf{c}$ are instantaneous linear maps to and from the vector of neural activities $\mathbf{x}(t)$.

43. Show that a purely linear neural network is a causal LTI system. Draw neural circuit diagrams that corresponds to questions 41 and 42.

A linear time-invariant (LTI) system is fully characterized by its *impuse response*, defined by the output of the system to a unit pulse input at time $t = 0$ (delta function; corresponds to the Dirac delta function in continuous time).

$$h(t) \triangleq H\{\delta(t)\} \qquad \text{(impuse response)} \tag{42}$$

$$\delta(t) = \begin{cases} 1 & t = 0 \\ 0 & t > 0 \end{cases} \tag{43}$$

44. What is the impuse response of the system $y(t) = u(t-2)$? Sketch it out over time.

Convolution

Given two signals $x(t)$ and $y(t)$ for $t \in \mathbb{Z}$, we define convolution as

$$(x * y)(t) = \sum_{k=-\infty}^{\infty} x(k) y(t-k) \tag{44}$$

45. Given an LTI system with impulse response $h(t)$, show that the output of the system is a convolution of the input with the impulse response, i.e.,

$$y(t) = (u * h)(t) \tag{45}$$

When a system's impulse response is restricted to a finite number of non-zero elements, it is said to have a **finite impulse response (FIR)**. If there are infinite number of non-zero elements, then the system has an **infinite impulse response (IIR)**.

46. Show that a purely feedforward linear network produces a finite impulse response.

47. Show that a linear network with a diagonal matrix produces an infinite impulse response.

LTI systems are used as **filters** in signal processing tools for smoothing, denoising, sharpening, removing line noise, isolating signals in a particular frequency bands, and so on. The *frequency response* of an LTI is characterized by the Fourier transform of the impulse response. If you want to study this subject further, we recommend reading a textbook on "digital signal processing" or "signals and systems" (electrical engineering curriculum).

3.5 Coordinate transformed feedforward networks

So purely feedforward system is an FIR LTI, and also the connectivity weights are strictly upper (or lower) triangular.

48. Consider the following 2-neuron feedforward system:

$$\mathbf{x}(t+1) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 & 1 \end{bmatrix} u(t) \tag{46}$$

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}(t) \tag{47}$$

Let us do a change of variables, $\mathbf{z}(t) = M\mathbf{x}(t)$ with an invertible matrix, say a unitary matrix

$$M = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \tag{48}$$

(a) What is the resulting neural network system defined with new neurons $z_1(t)$ and $z_2(t)$?

(b) Is the weight matrix strictly upper triangular for $\mathbf{z}$? Can it be made so through permutation?

(c) Does the impulse response change?

(d) Draw the neural circuit diagram of neurons $\mathbf{z}$ and compare to that of $\mathbf{x}$.

3.6 Schur decomposition

As we saw from 48, systems with finite impulse response do not necessarily have strictly triangular form. Can we reverse the construction of 48 to determine if a system is purely feedforward in disguise? There's another matrix decomposition that can answer this question.

Schur decomposition

Any square matrix (real- or complex-valued) A can be decomposed into the following form,

$$A = UTU^\dagger \qquad \text{(schur decomposition)} \tag{49}$$

where U is a unitary matrix, and T is an upper triangular matrix.

The Schur decomposition can be numerically found cheaply. If the resulting triangular matrix T has no non-zero diagonal element, that is, strictly triangular, then it is a feedforward system in disguise!