Flexible Memory Networks - Curto, Degeratu, Itskov

Viktor Giordano

CSE 5339, TDA in Neuroscience

April 12, 2022

Motivation

1. The Importance of Forgetting was shown to us by Brantley.
2. In a similar light, we want to understand how neurons can quickly encode new memories and quickly forget.
3. It is common to model neurons and their activity in a network.

Assumption

Memories can be thought of as activity patterns within this network.

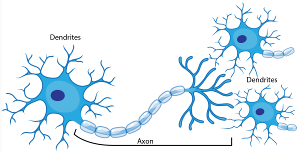

Figure: Neurons

Neural Plasticity

Definition

Neural plasticity is the nervous system's ability to adapt its function or structure in response to external stimulus.

1. It allows for new functions to be created, for old functions to be repurposed, and for maintenance of old connections.
2. We have seen that forgetting can actually allow for better memory, neural plasticity provides an explanation.
3. "Memories" can adapt to more efficiently remember.

Brain as a Network

1. To begin modeling changes in the brain, we must identify how neurons work together.
2. This paper follows a common neural network, a standard firing rate model with a handful of stipulations placed upon it.
3. If we can accommodate slight changes in a network pattern, this could potentially explain how rapid learning can take place.
4. We will call a network of neurons that can slightly change a flexible network. A formal definition of this concept will be given later.

Going Forward

Can a combination of network dynamics and topology be used to explain flexible networks?

Starting in network dynamics:

1. Want to establish an "architecture" of neuron connections.
2. Want to determine the "flexibility" of our architecture.

Moving to graph theory and topology:

1. Want to constrain our network architectures to a graph.
2. Want to identify the stability and then flexibility of cliques of our graph.
3. Apply simplicial homology to these cliques to identify our maximally flexible networks.

Standard Firing Rate Model

1. We will model the brain using a standard firing rate model for the i-th neuron of the network:
$$\frac{dx_i}{dt} = -\frac{1}{\tau_i}x_i + \phi(\Sigma_{j=1}^{n} W_{ij}x_j + b_i) \tag{1}$$
2. x is a real-valued function of the firing rate, W_ij is the strength of connection between the j-th to i-th neuron, τ is a timescale for the rate of recovery of the neuron, b is an external input, and a nonlinear function ϕ makes sure that x is nonnegative.

Standard Firing Rate Model

$$\dot{x} = -Dx + \phi(Wx + b) \tag{2}$$

This equation more compactly represents our first system using matrices and vectors.

1. D is a diagonal matrix of $\frac{1}{\tau_i}$.
2. W is our connectivity matrix.
3. $-D + W$ must have negative diagonal entries, otherwise a neuron could have "run-away" activity.

The network and its dynamics are denoted (W, D), as these matrices can be used to determine all pertinent information moving forward.

Constraints on the Model

To simplify the theory, we impose key assumptions about the model:

1. This model will only cover fast-timescale dynamics. Probabilistic firing of neurons is ignored.
2. This model is of a recurrent neural network. We can restrict the patterns which it takes and stored data can be used to influence future inputs and outputs.

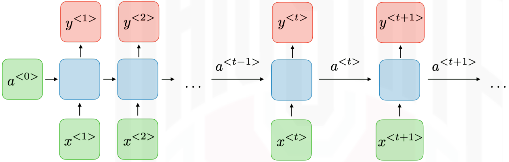

Figure: Recurrent Network

Fast Timescale Dynamics

Fast timescale dynamics refers to analyzing the system in a time frame that is "too short" to visualize the system changing. We want to understand stability for short periods of time.

In the case of a neural network, we can understand this as one spike of neurons. There is no excess time allowed for the system to change neuron structure.

Dissecting Matrix W

We will divide our connectivity matrix, W, into two components.

$$W = J + A$$

1. J is fixed as the "architecture", the set weights of each neuron connection. This does not change with time.
2. A is a matrix of perturbations about J, A is constantly changing as memories/experiences occur in the brain.

Architecture J

The physical arrangement of neurons is not as important as the connection weights of J in our model.
This constrains the patterns that we can observe.

Fixed Points

A fixed point, x^*, of our system is achieved whenever when the firing rate is not changing, $\dot{x} = 0$.

Fixed points can either be:

1. Stable -solution trajectories remain very close to x^*
2. Unstable -solution trajectories can stray from x^*

Definition

A fixed point x^* is *asymptotically stable* if there exists an open neighborhood U such that $\lim_{t \to \inf} x(t) = x^*$ for every $x(t)$ with $x(0) \in U$.

Definition

The subset of active neurons is called the *support* of x,

$$supp(x) = \{i | x_i > 0\}$$

Stable, Marginal, and Unstable Sets

Definition

A nonempty subset of neurons, σ, is a *stable set* of (W,D) if there exists an asymptotically stable fixed point x^* such that $supp(x^*) = \sigma$, for at least one external input vector b. A *marginal set* of (W,D) is a nonempty subset of neurons for which there exists a stable fixed point of the dynamics (but no asymptotically stable fixed point), and an *unstable set* of (W,D) is a non-empty subset of neurons that is neither stable nor marginal.

"Memories" correspond to a stable set of neurons.
Our constrained recurrent network model only performs meaningful computations since only a small number of sets are stable.

Threshold-Linear Networks

We had a nonlinear function, ϕ, in our standard firing rate model. However, finding fixed points of high dimensional nonlinear equations is difficult.
To work-around this, we introduce a threshold-linear constraint.

Assumption

We choose the nonlinearity of a network (W, D) *to be,*

$$\phi(y) = \begin{cases} y & y > 0 \\ 0 & y \leq 0 \end{cases}$$

Now we can choose our linear dynamic to be written as,

$$\dot{x} = (-D + W)x + b$$

Upshot of Threshold-Linear Condition

After applying a threshold-linear condition to the dynamic equation, the stability of the network is determined by the eigenvalues of $-D + W$.

1. Since memories are considered to be stable sets of neurons, and we want to find sets that can both remember and forget, we look for architectures that are marginally stable.
2. Marginal stability is categorized as having no positive real eigenvalues, and at least one eigenvalue that is purely imaginary.

Silent Synapses

In the brain, we observe that many synaptic connections are not functional.
We want to model small perturbation without altering these silent connections.
To account for this, we will utilize graph theory.

Applying Graph Theory

To work around silent synapses, we define:

Definition

Let $G = (V, E)$ be a graph with vertices V and edges E. We say that an architecture matrix J is *constrained* by the graph G if $J_{ij} = 0$ for all edges $(ij) \notin E$.

Using this definition, if $(ij) \in E$, and $J_{ij} = 0$, we say there is a silent connection from neuron j to i.

ϵ-Perturbation and Notation

An ϵ-perturbation of J is given whenever $|A_{ij}| \leq \epsilon$.
An ϵ-perturbation is consistent with G whenever it only alters nonzero entries. We only consider consistent matrices A.

Definition

The following notation is used to show the set of all G-constrained network architectures,

$$N(G) = \{(J, D)_G\} = \{(J, D) \; s.t. \; J_{ij} = 0 \; for \; all \; (ij) \notin G\}$$

The set of network architectures on the complete graph (no constraint), we write $N(n)$.

Cliques

Definition

A *clique* in a graph G is a subset of vertices that are all one-to-one connected, and the clique complex of G, $X(G)$, is the set of all cliques.

1. If a set, σ, of a network $(W, D)_G$ is stable and $\sigma \in X(G)$, then σ is a stable clique. Marginal and unstable follow.
2. A *maximally stable clique* is a stable clique that is not properly contained in any larger clique. Minimally unstable clique follows from the smallest unstable clique.

Definition

A subset of neurons σ is called a *flexible clique* of $(J, D)_G$ if for every $\epsilon > 0$ there exist consistent ϵ-perturbations A_s and A_u such that σ is a maximally stable clique of $(J + A_s, D)_G$ and a minimally unstable clique of $(J + A_u, D)_G$.

Cliques

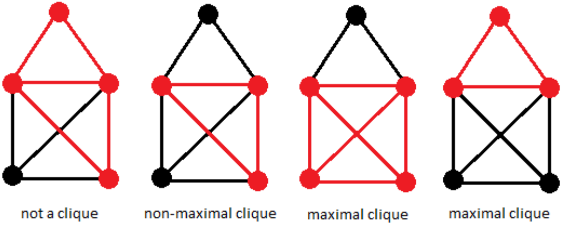

Figure: Cliques

Definition

Maximal: We want to find the network that allows for the highest number of neuron sets to become stable or unstable. A "maximally flexible network" will achieve the upper bound of flexibility.

Flexible Cliques

1. We define the flexibility of a network as the number of flexible cliques (or neuron sets), and denote it: $flex(J, D)_G$.
2. We define the rank of a network $(J, D)_G$ to be the rank of the matrix $-D + J$.
3. We say that a G-constrained network on n neurons, $(J, D)_G$, has a rank k completion if there exists a network $(\bar{J}, \bar{D})_G \in N(n)$ of rank k such that $\bar{D} = D$ and $\bar{J}_{ij} = J_{ij}$ for all $i = j$ and all distinct pairs $(ij) \in G$.

If a complete graph network has rank 1, the same architecture (aside from extra edges) as some graph G, then the network of G has rank 1 completion.

Rank and Maximal Flexibility

Since single neurons cannot be flexible cliques, we can bound the flexibility as,

$$flex(J,D)_G \leq |X(G)| - n - 1$$

Theorem

All rank 1 threshold-linear networks of n neurons are maximally flexible in $N(n)$, and have flexibility $2^n - n - 1$. All G-constrained threshold-linear networks with a rank 1 completion are maximally flexible in $N(G)$, and have flexibility $|X(G)| - n - 1$.

Why rank 1 matters

A condensed proof of the last theorem:
Assume $-D+J$ is a rank 1 matrix, by set up all entries are negative on the diagonal. There exist vectors $x, y \in \mathbf{R}^n$ s.t. $-D+J=-xy^T$. Constructing the diagonal matrix $d=diag(\sqrt{\frac{y_i}{x_i}})$, and let $P=d(-D+J)d^{-1}$. Then,

$$P_{ij}=-\sqrt{x_i y_i}\sqrt{x_j y_j}$$

P is a symmetric rank 1 matrix, so there exists a vector v s.t. $P=-vv^T$. For any perturbation, A, the matrix given by $-vv^T+diag(x)Adiag(x)$ has principal submatrices whose stability is determined by the signs of the principal minors. Thus there exist stable and unstable perturbations, A_s and A_u, and thus P is maximally flexible in $N(n)$.

Why rank 1 matters

Since P and $(-D+J)$ are similar, their principal submatrices are similar. Thus there exist stable and unstable perturbations, A_s and A_u, and therefore $(-D+J)$ is maximally flexible in $N(n)$.

Through a "pruning" of the complete graph, it can be shown that any rank 1 completion is maximally flexible in $N(G)$.

Simplicial Complexes

Luckily, using topology makes the characterization of these networks easier. Clique complexes are abstract simplicial complexes, so we can apply simplicial homology.

Theorem

Let $(J, D)_G$ be a maximally flexible threshold-linear network in $N(G)$, and suppose that the clique complex $X(G)$ satisfies $H_1(X(G); Z) = 0$. Then $(J, D)_G$ has a rank 1 completion. In particular, $(J, D)_G$ has no silent connections.

Surprisingly, this condition is likely naturally occurring. According to Kahle (2009), an Erdos-Renyi graph with edge probability p, n independent vertices, $p \geq n^{\alpha}$ with $\alpha < \frac{1}{3}$ implies that the probability that $H_1(X(G); Z) = 0$ approaches 1 as the number of vertices tends to infinity. For a neural network with 10^4 neurons, p can be as low as 0.05.

Homology Condition

1. From other resources, we see that the condition is likely natural.
2. The proof follows a series of matrix properties and calculations to show rank 1 completion of the network.
3. The important take away: Since all vertices of cliques are one-to-one connected, if H_1 has a nonzero rank, then there is an edge with a weight of zero (silent synapse). This begs the question, are there maximally flexible networks that contain silent connections?

References

Curto, Carina Degeratu, Anda Itskov, Vladimir. (2011). Flexible Memory Networks. Bulletin of mathematical biology. 74. 590-614. 10.1007/s11538-011-9678-9.

Curto, Carina Morrison, Katherine. Pattern Completion in Symmetric Threshold-Linear Networks. Neural Comput 2016; 28 (12): 2825–2852. doi: https://doi.org/10.1162/NECO_a_00869

Kahle, M. (2009). Topology of random clique complexes. Discrete Math., 309(6), 1658–1671.

"Recurrent Neural Networks Cheatsheet Star." CS 230 - Recurrent Neural Networks Cheatsheet, https://stanford.edu/ shervine/teaching/cs-230/cheatsheet-recurrent-neural-networks.

von Bernhardi R, Bernhardi LE, Eugenín J. What Is Neural Plasticity? Adv Exp Med Biol. 2017;1015:1-15. doi: 10.1007/978-3-319-62817-2$_1 . PMID : 29080018.$