Spike2Vec: Converting Spike Trains to Vectors to Analyse Network States and State Transitions:

Dr Russell Jarvis Pablo de Abreu Urbizagastegui Yeshwanth Bethi

***Abstract*— A scalable algorithm that can detect fine grained repetitions quickly across large spiking datasets is desirable, as it provides a means to test for the tendency of activity to revisit states. By quantifying repetitions large spiking datasets, using geometric representations of complex spike patterns, we can quantify the frequency of repitition, and achieve a better understanding of a networks ability to revisit states. To this end we represented time bound neural activity as simple geometric coordinates in a highdimensional space. Working with geometric representations of chaotic spike train recordings may enable researchers to interrogate the state-fullness of both biologically recorded spike trains and their digitally simulated counterparts. Furthermore, there is reason to believe that when mammal brains enact visual object recognition encoded memories guide cortical neurons to "replay" previously observed neural states, as replayed spiking states may cohere with the visual brains perceptual recognition of a familiar scene.**

***Index terms*—A, B, C, D**

I. Introduction

There is a great demand for a scalable algorithm that can detect repeating temporal spatial features in biological and synthetic data sets of cortical neuronal networks.

Multivariate approaches to spike train network analysis often involves the computation of some kind of statistic between each possible pair of neurons in the network. To analyse causality in networks, spike train recordings are divided into time windows, and analysis compares previous (lagged time), with current time. Exhaustive pairwise iteration of multivariate statistics is not computationally tractible at the scale of billions of neurons, and adding time lagged analysis of network cross-correlation, or transfer entropy makes the prospoect of scaled temporal analysis even worse. Autocovariance acts on anolog signals (dense vectors), however autocovariance analysis of continuous membrane potentials would be another way to arrive at a network state description.

Two common models of cortical spiking networks are the, Potjan's and Diesmon model and the Brunel model, both of these models are said exist within a fluctuation driven regime, when these are simulated, observed spike times are typically chaotic and random, but some fine grained recognizable repeating patterns also occur. Under the dynamic systems view of the brain neuronal memories are analogous to attractor basins [Hopfield,Lin, Hairong, et al]. If the view of memories as basins is correct then it should be possible to demonstrate synaptic learning as the mechanism that encodes memories as basins. Network attractor basins may be derived from the interleaved application of Spike Timing Dependent Plasticity (STPD) and sleep when synapses are able to change in a way that strongly biases some future spiking activities towards stereotyped patterns.

The application of STDP learning within the fluctuation driven regime necessitates a simple method to optimise network parameters a way that maximises the networks capacity to encode and revisit attractor states. A spike2vec algorithm will enable researchers to investigate the state-fullness of spike trains, the corruption of information caused by STDP in the absence of sleep and resistance to the degradation of memories that may be concomitant with neuronal death and synaptic pruning, as many of these network level phenonemana can be re-construed as network parameters: for example neuronal death relates to synaptic count and neuron count.

II. Theoretical Framework

III. Methodological Framework

IV. Result Analysis

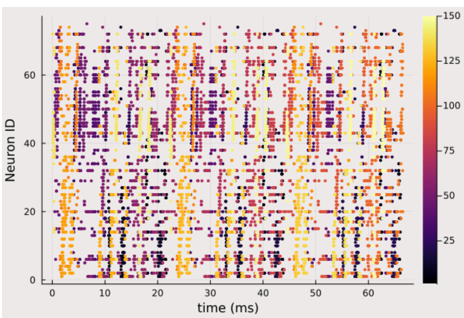
V. Statement of Need

Scalable methods for representing the transient behavior of large populations of neurons are needed. The spike2vec algorithm will enable researchers to track the trajectory of the network between familiar and unfamiliar states using a highdimensional coordinate scheme. A network's ability to revisit an encoded coordinate is testable, and so a spike2vector test of object recognition could be construed as a formal hypothesis test.

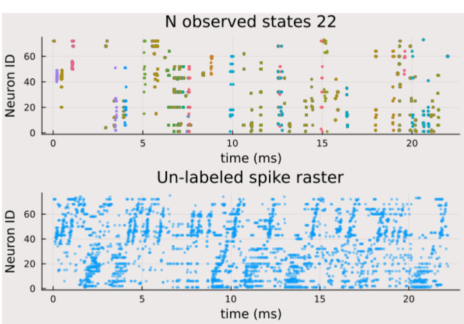
VI. Reproducibility

Some preliminary code that performs the Spike2Vec analysis is avaialble at the following link. the code is implemented in Julia, a modern language alternative to Python that makes large-scale model visualization and analysis more computationally tractable. A docker file is included.

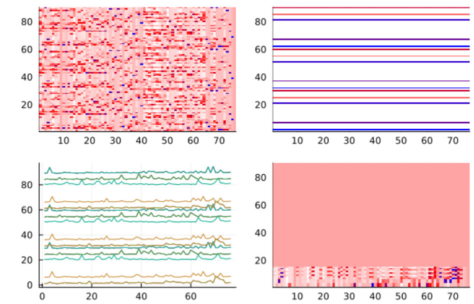
Herein lies a heatmap of dis-simarity matrices constructed using the NMNIST dataset, ie the heatmap above, comes from analysing spike train distance across the NMNIST data set numbers: 0-9 represented as spiking events. There are 300 total presentation number presentations. All nine numbers are incrementally cycled through. Number presentations within the one number are contiguous, (the data set isn't shuffled), and this contiguity is reflected in the heatmap too.



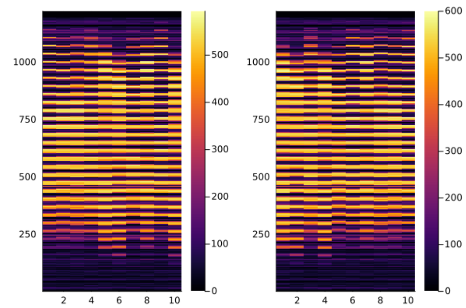
In order to first if the spike2vec analysis code worked as expected, we downloaded a alcium imaging recording from Zebra finch (a song bird's) High Vocal Centre (brain region) source [1]. Although the actual data source was from



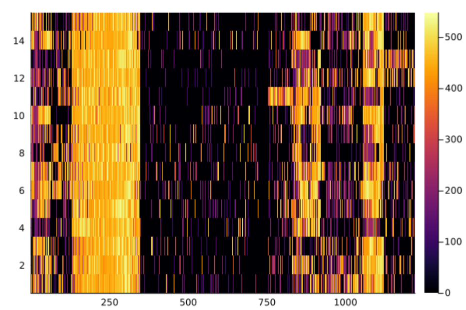
Glaciers form an important part of the earth's climate system.



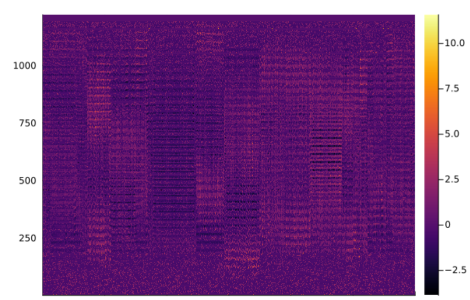
Glaciers form an important part of the earth's climate system.



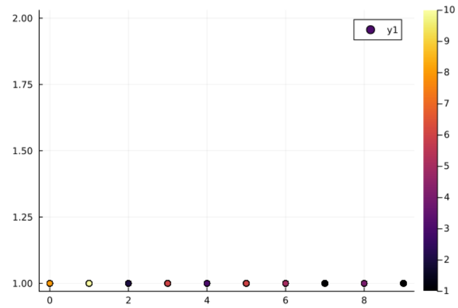
Unormalize NMNINST vectors for 1000 neurons over 10 channels



Unormalize NMNINST vectors for 1000 neurons over 10 channels



Unormalize NMNINST vectors for 1000 neurons over 10 channels



Unormalize NMNINST vectors for 1000 neurons over 10 channels

clustering_NMNIST.png cluster_sort_MNMIST.png cluster_sort_pablo.png cluster_sort_song_birds.png didit_work_NM.png didit_work.png everything_includinding_repeated_pattern_pablo.png everythin_includinding_repeated_pattern_pablo.png heatmap_after.png heatmap_before.png heatmap.png just_two_pablo_raw_vectors.png just_two_song_bird_raw_vectors.png labelled_mat_of_distancesNMINST.png labelled_mat_of_distancesNMINST_test_train.png labelled_mat_of_distances_pablo.png labelled_mat_of_distances.png LabelledSpikes18.png LabelledSpikesPartition18.png Levine13-CD4.png Normalised_heatmap_pablo.png Normalised_heatmap_song_bird.png normal.png not_cluster_sort_MNMIST.png pablo_umap.png reference_labelled_mat_of_distances_pablo.png relative_to_uniform_referenceNMMIST.png repeated_pattern_pablo.png repeated_pattern.png repeated_pattern_song_bird.png scatternmnist_angles.png scatternmnist_distances.png scatternmnist.png slice_one_window.png slice_three_window.png slice_two_window.png sorted_train_map.png test_map.png train_map.png umap_of_NMNIST_Data.png UMAP_song_bird.png UniformSpikes.png Unormalised_heatmap_pablo.png Unormalised_heatmap.png Unormalised_heatmap_song_bird.png vector_differences_another_NMNIST.png vector_differences_another.png

A. *References*

References

[1] E. L. Mackevicius, A. H. Bahle, et al., "Unsupervised discovery of temporal sequences in high-dimensional datasets, with applications to neuroscience," *Elife*, vol. 8, 2019.