A network-based framework for detecting communities of similar neurons in neural data

Indra Ghosh

MOCAM Seminar Series

September 02, 2026

University College Dublin

Email: indra.ghosh@ucd.ie

About me

i. Call me “Indra”.
ii. Postdoctoral fellow in applied mathematics and computational neuroscience @ University College Dublin

iii. B.Sc. Physics
iv. M.Sc. Physics
v. Ph.D. Applied Mathematics

vi. Visiting fellow @ Trinity College Dublin
vii. Write softwares in and
viii. Research in dynamical systems, network science, neurodynamics, mathematical epidemiology, etc.
ix. Find me at https://indrag49.github.io/

Importance

1. Detect functional neural assemblies directly from spike trains.

2. Bridge neural spike-train analysis with network science and community detection.

3. Because neural recordings are high dimensional we need frameworks to reduce them into interpretable population structure.

4. Functional assemblies are not directly observable: so activity based grouping is necessary.

5. It is important to realise biological mesoscale structures.

source: https://hero.fandom.com/wiki/Cuthbert_Calculus

Humphries (2011)

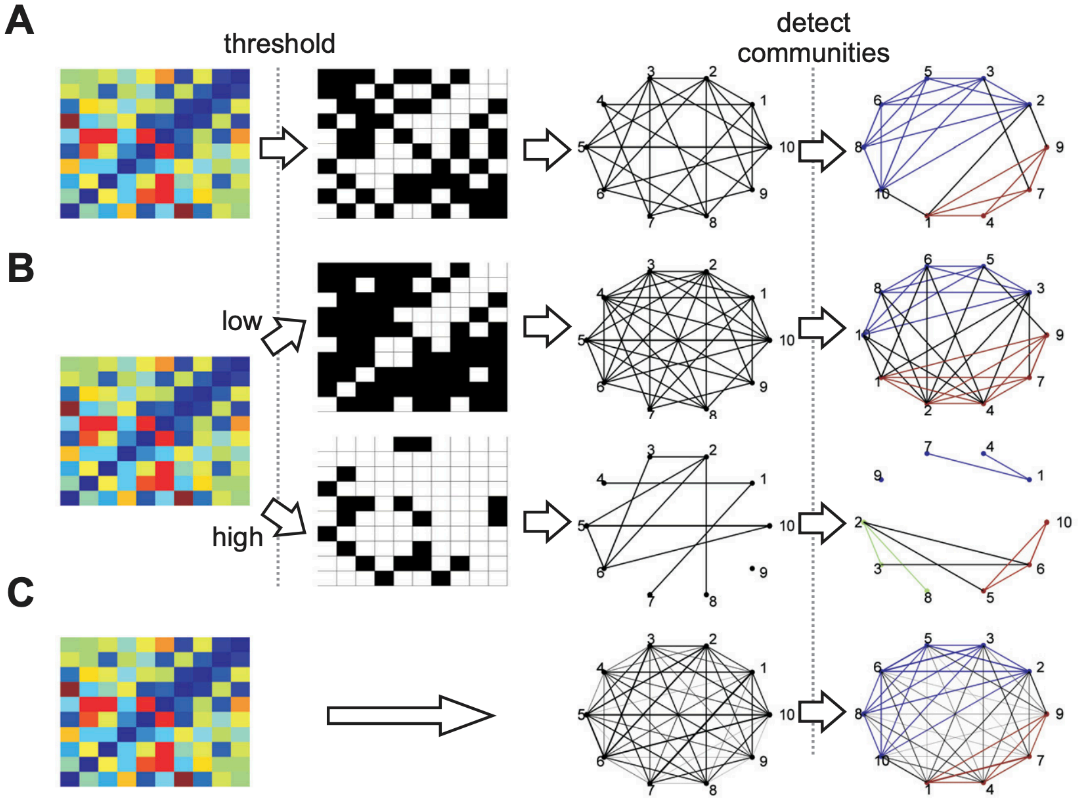
The Journal of Neuroscience, February 9, 2011 • 31(6):2321–2336 • **2321**

Behavioral/Systems/Cognitive

Spike-Train Communities: Finding Groups of Similar Spike Trains

Mark D. Humphries
Group for Neural Theory, Department d'Etudes Cognitives, Ecole Normale Superieure, 75005 Paris, France, and Adaptive Behaviour Research Group, Department of Psychology, University of Sheffield, Western Bank, Sheffield S10 2TN, United Kingdom

$$s_{ij} = \frac{\vec{g}_i \cdot \vec{g}_j}{\|\vec{g}_i\| \cdot \|\vec{g}_j\|},$$



Neuropixels data (Allen Brain Institute)

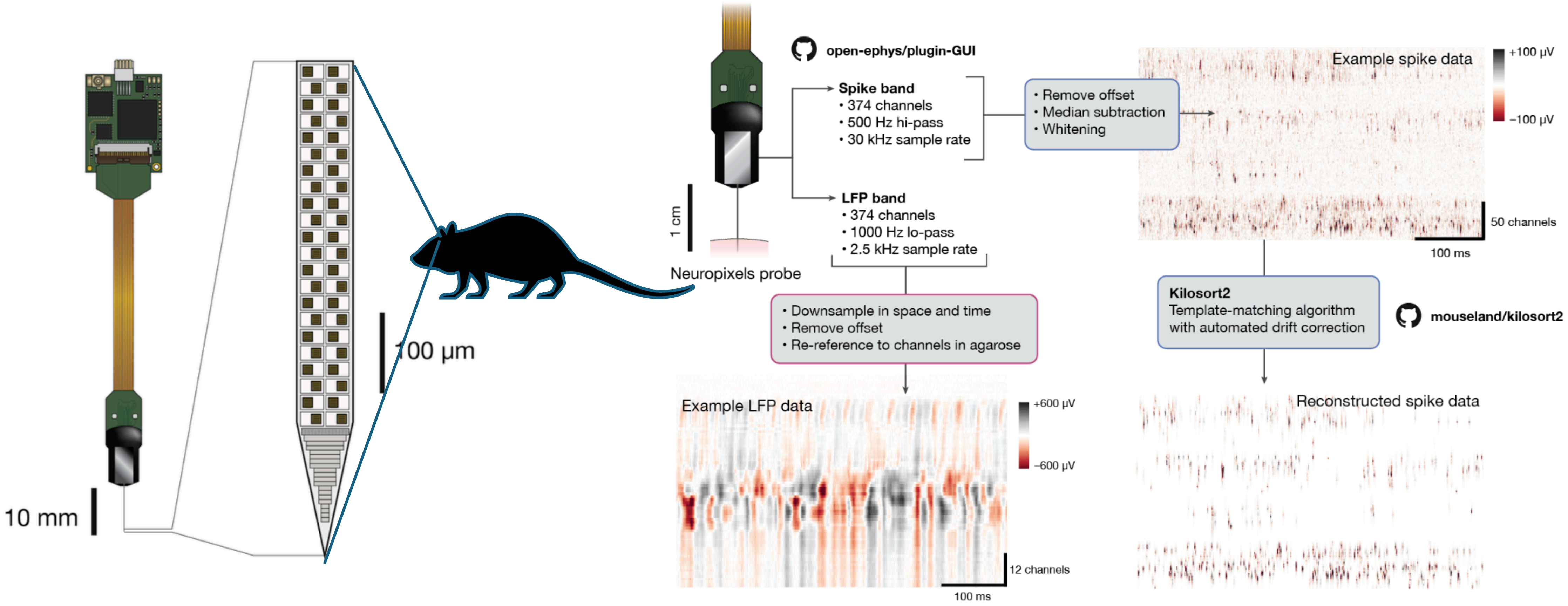


Image credit: Allen Institute for Brain Science. [https://brain-map.org/our-research/circuits-behavior/visual-coding], [https://allensdk.readthedocs.io/en/latest/visual_coding_neuropixels.html]

van Rossum distance

1. We convolve a discrete spike train data

$$\mathcal{K}(t) = \exp\left(-\frac{t}{t_R}\right) H(t),$$

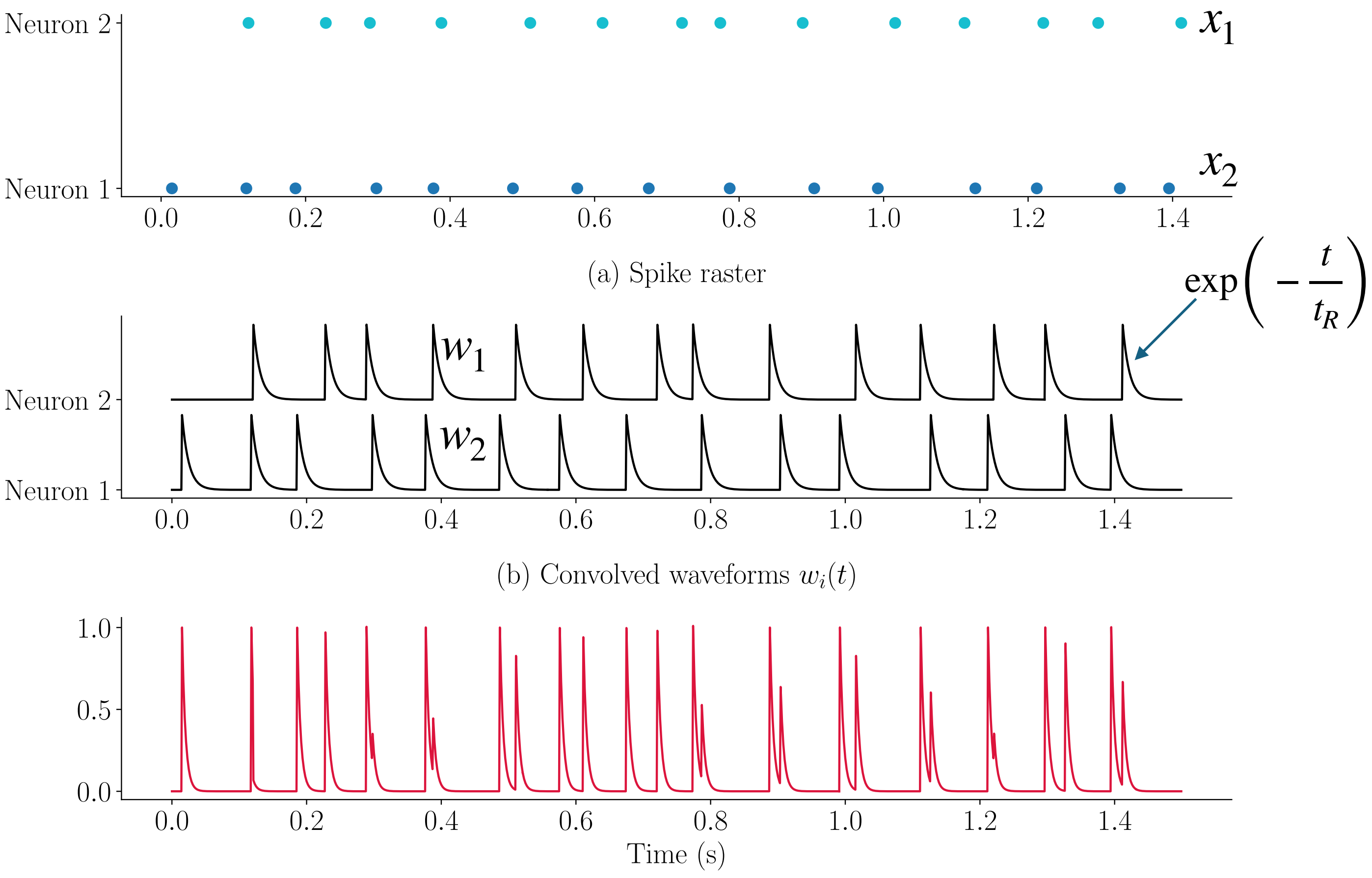
2. Convolution:

$$w_i(t) = \sum_k \exp\left(-\frac{t - t_i^k}{t_R}\right) H(t - t_i^k)\,.$$

3. Pairwise distance:

$$D_{ij}(t_R) = \sqrt{\frac{1}{t_R} \lim_{T\to\infty} \int_0^T (w_i(t) - w_j(t))^2 dt}\,.$$

4. Normalised: $\tilde{D}_{ij}(t_R) = D_{ij}(t_R)/\sqrt{\tilde{N}_{ij}}\,.$



(a) Spike raster

(b) Convolved waveforms $w_i(t)$

(c) $(w_1(t) - w_2(t))^2$

Principal workflow (vR-FCD)

1. **vR-FCD**: van Rossum-Functional Community Detection.

2. Compute van Rossum distance matrix $\tilde{D}$.

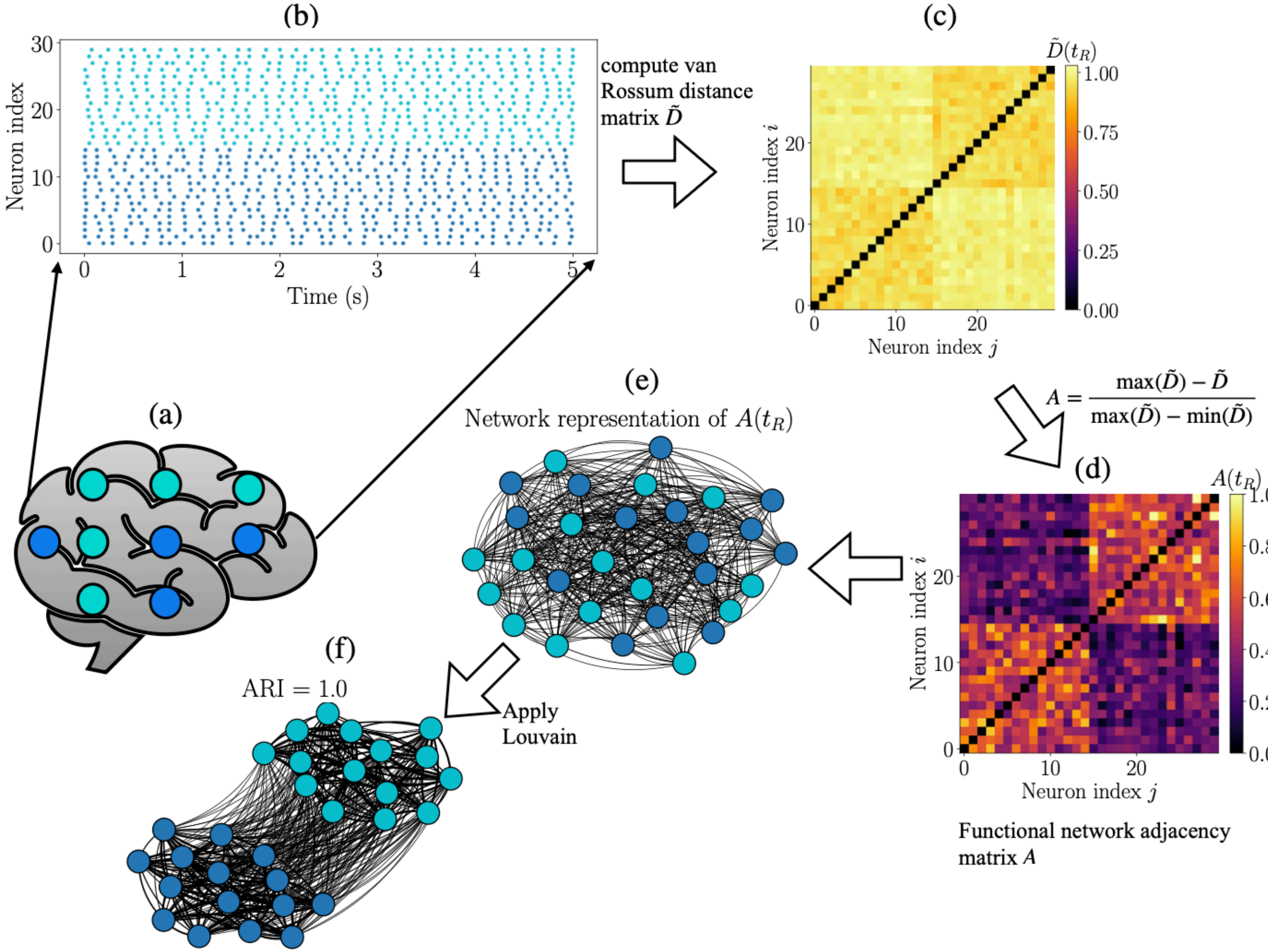
3. Apply the min-max scheme to get the similarity matrix A:

$$A = \frac{\max(\tilde{D}) - \tilde{D}}{\max(\tilde{D}) - \min(\tilde{D})}.$$

4. This is the network representation.

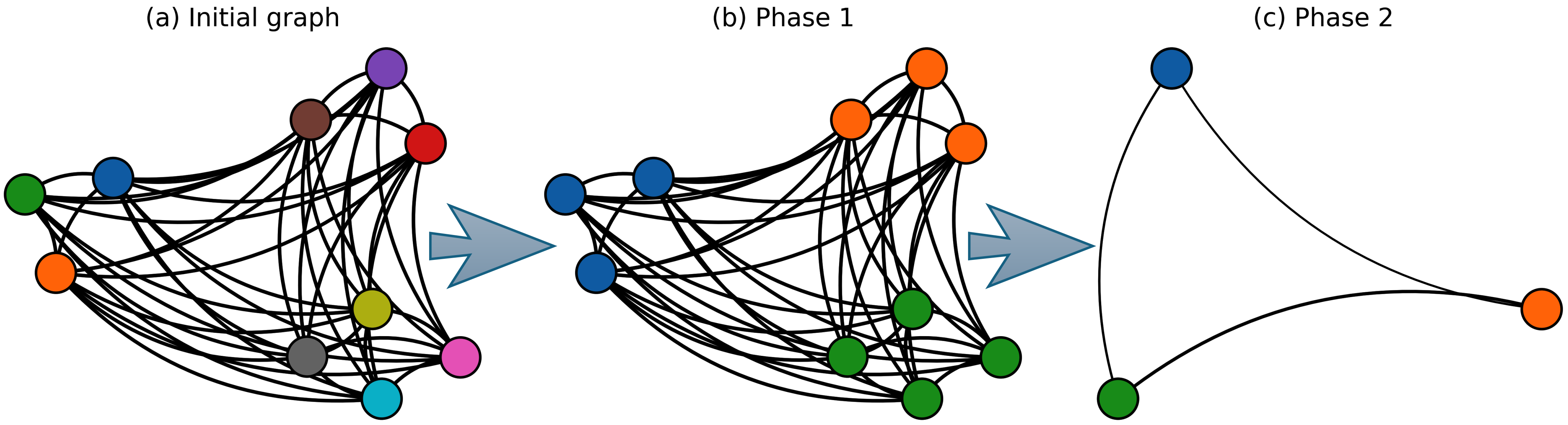
5. Apply Louvain.

6. Measure error if ground-truth is known!



Louvain algorithm: community detection

1. This algorithm depends on optimising the modularity of the network: $Q = \dfrac{1}{2m}\displaystyle\sum_{i=1}^{N}\sum_{j=1}^{N}\left[A_{ij} - \dfrac{k_i k_j}{2m}\right]\delta(c_i, c_j)\,.$
2. Here A_{ij}: element of the adjacency matrix, $k_i = \sum_j A_{ij}$, and c_i, c_j: the communities to which nodes i and j belong respectively, and $m = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} A_{ij}$.
3. Works in two phases:

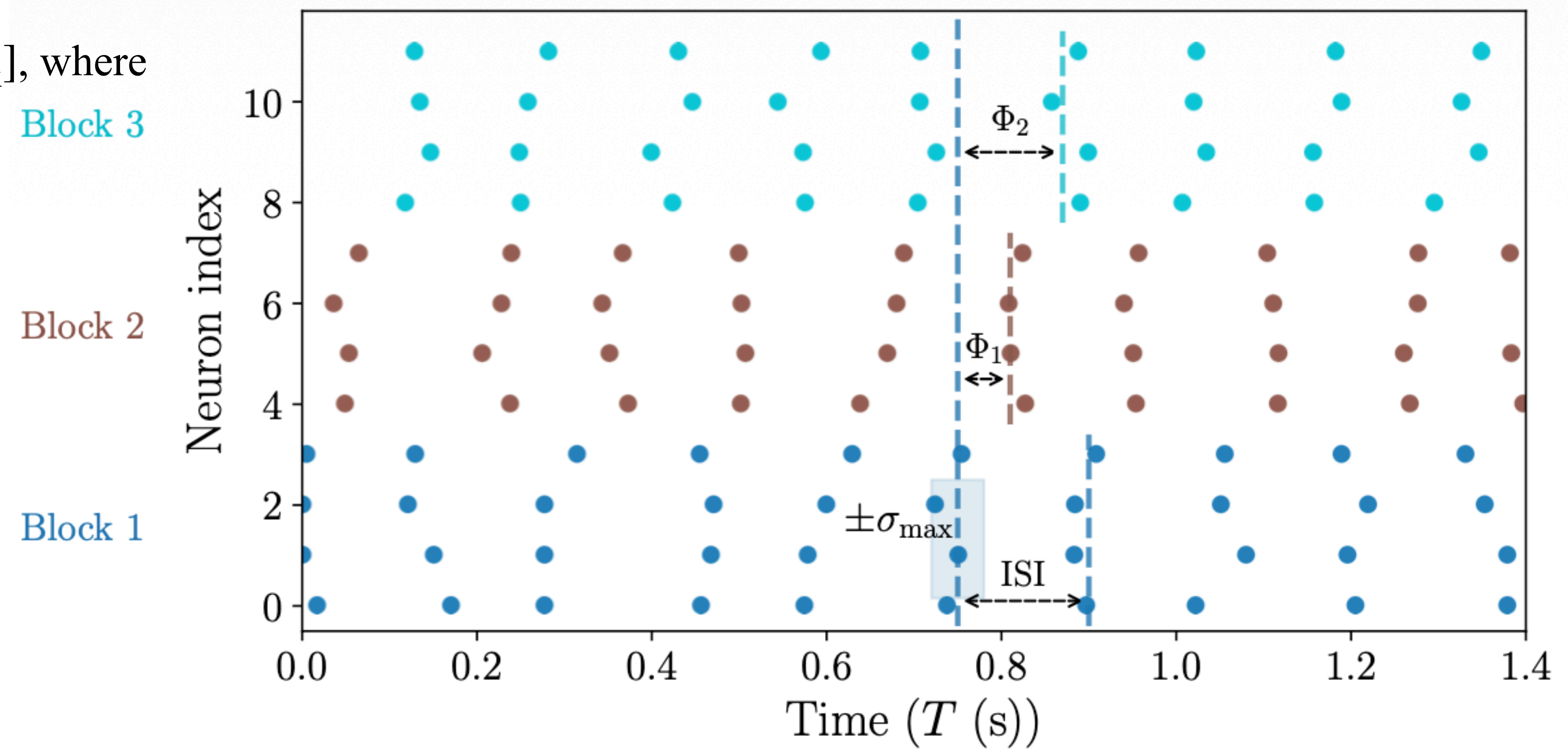


Synthetic data

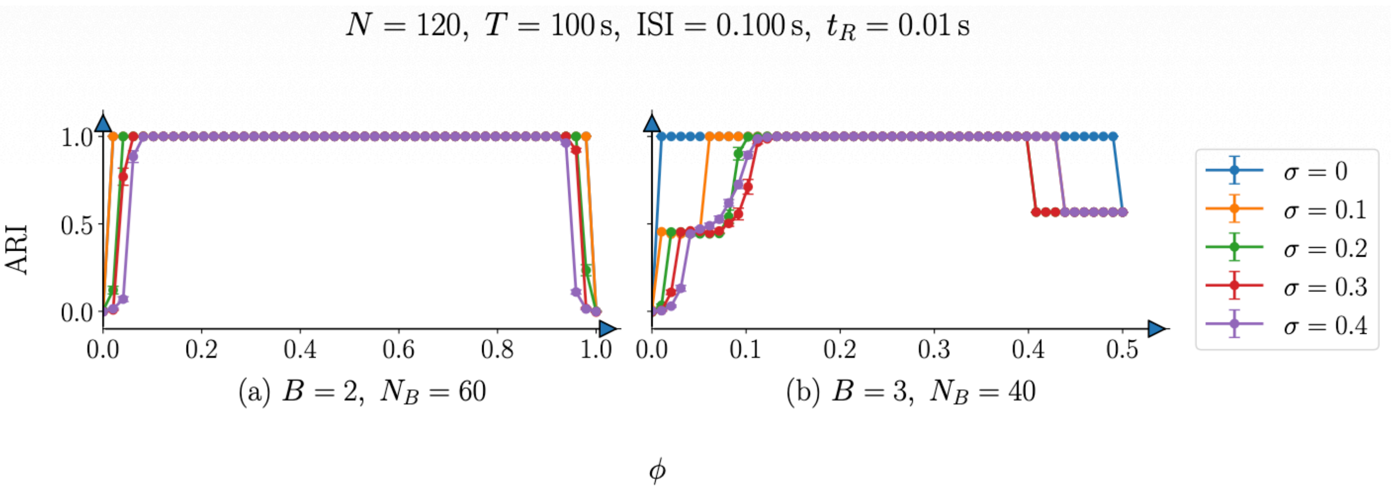
1. Synthetic raster: $t_b^k = (\phi_b + k) \times \mathrm{ISI} + \zeta^k$

2. Noise $\zeta^k \sim U[-\sigma_{\max}, \sigma_{\max}]$, where $\sigma_{\max} = \mathrm{round}(\mathrm{ISI} \times \sigma)$.

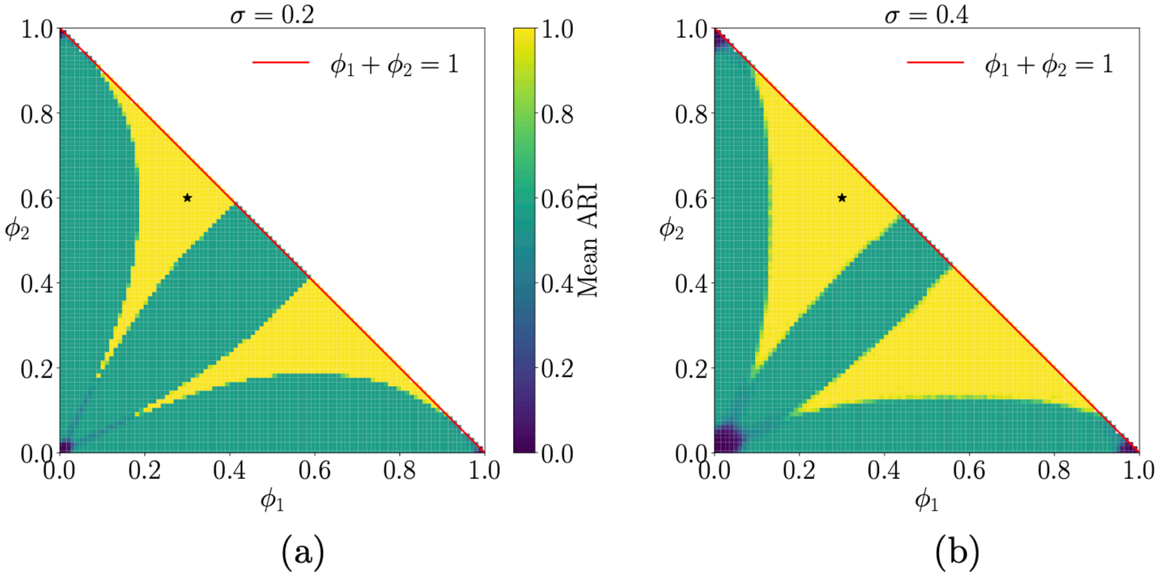
3. Here $\Phi_b = \phi_b \times \mathrm{ISI}$.



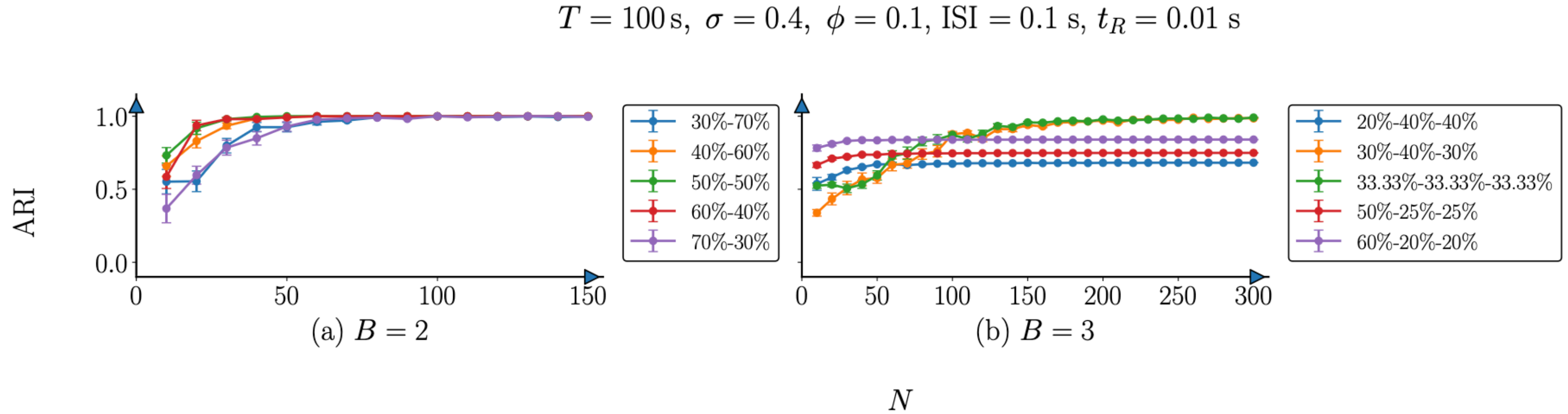
Results



$N = 120,\ T = 100\,\mathrm{s},\ \mathrm{ISI} = 0.100\,\mathrm{s},\ t_R = 0.01\,\mathrm{s}$
ARI
ϕ
$\sigma = 0$
$\sigma = 0.1$
$\sigma = 0.2$
$\sigma = 0.3$
$\sigma = 0.4$
(a) $B = 2,\ N_B = 60$
(b) $B = 3,\ N_B = 40$



$\sigma = 0.2$
$\sigma = 0.4$
$\phi_1 + \phi_2 = 1$
ϕ_1
ϕ_2
Mean ARI
(a)
(b)



$T = 100\,\mathrm{s},\ \sigma = 0.4,\ \phi = 0.1,\ \mathrm{ISI} = 0.1\ \mathrm{s},\ t_R = 0.01\ \mathrm{s}$
ARI
N
30%-70%
40%-60%
50%-50%
60%-40%
70%-30%
20%-40%-40%
30%-40%-30%
33.33%-33.33%-33.33%
50%-25%-25%
60%-20%-20%
(a) $B = 2$
(b) $B = 3$

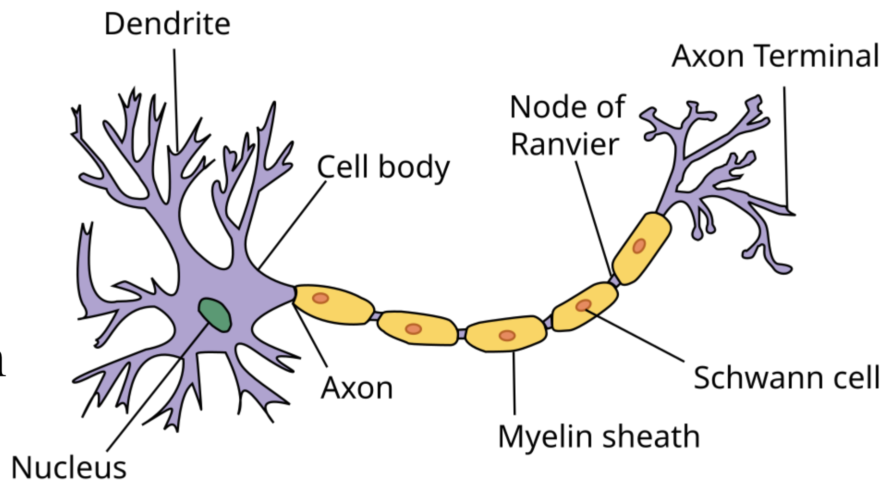
Modelling neuron dynamics

1. Neurons evolve over time.

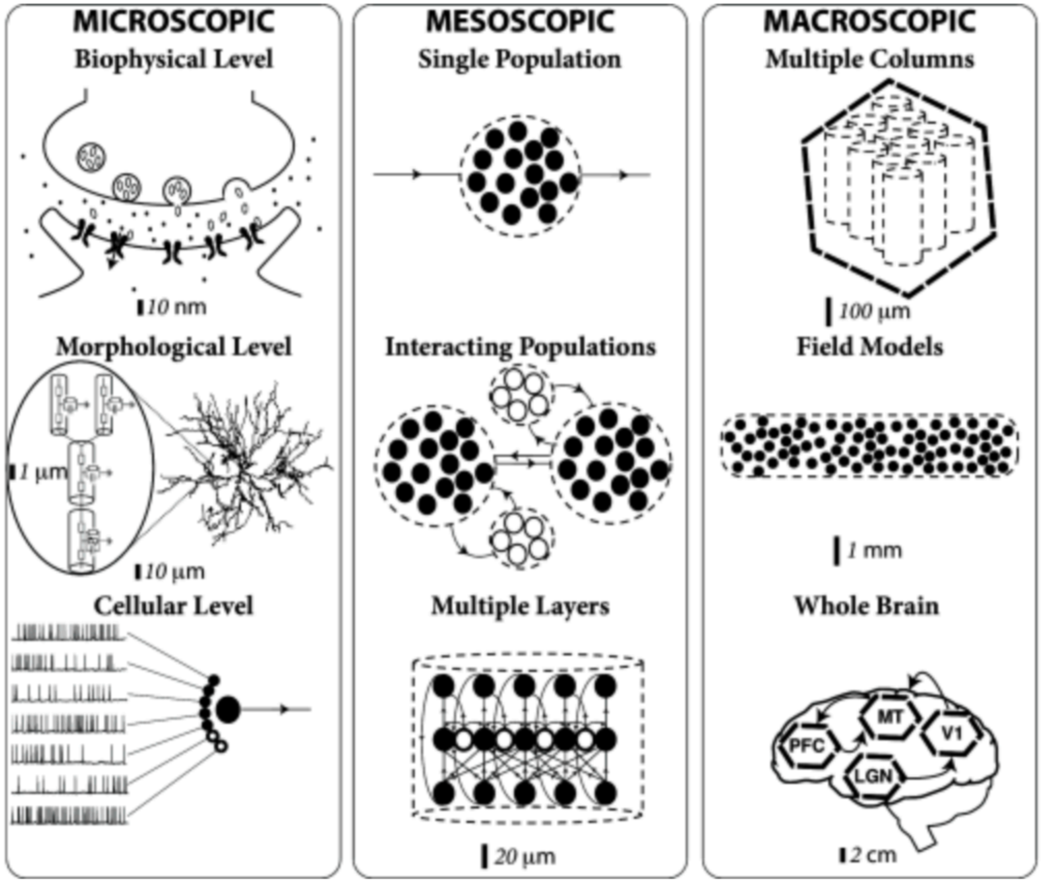
2. Examples of dynamical variables are
   (a) membrane voltage
   (b) firing rate
   (c) synaptic current
   (d) population activity

3. Mathematically written as an ODE:

$$\frac{dx}{dt} = f(x, t)$$



Source: https://commons.wikimedia.org/wiki/File%3ANeuron.svg



Source: https://neuronaldynamics.epfl.ch/online/Ch12.html

Application to Spiking Neural Networks

1. We apply this methodology to a stochastic block model (SBM) network of Leaky Integrate and Fire (LIF) neurons,

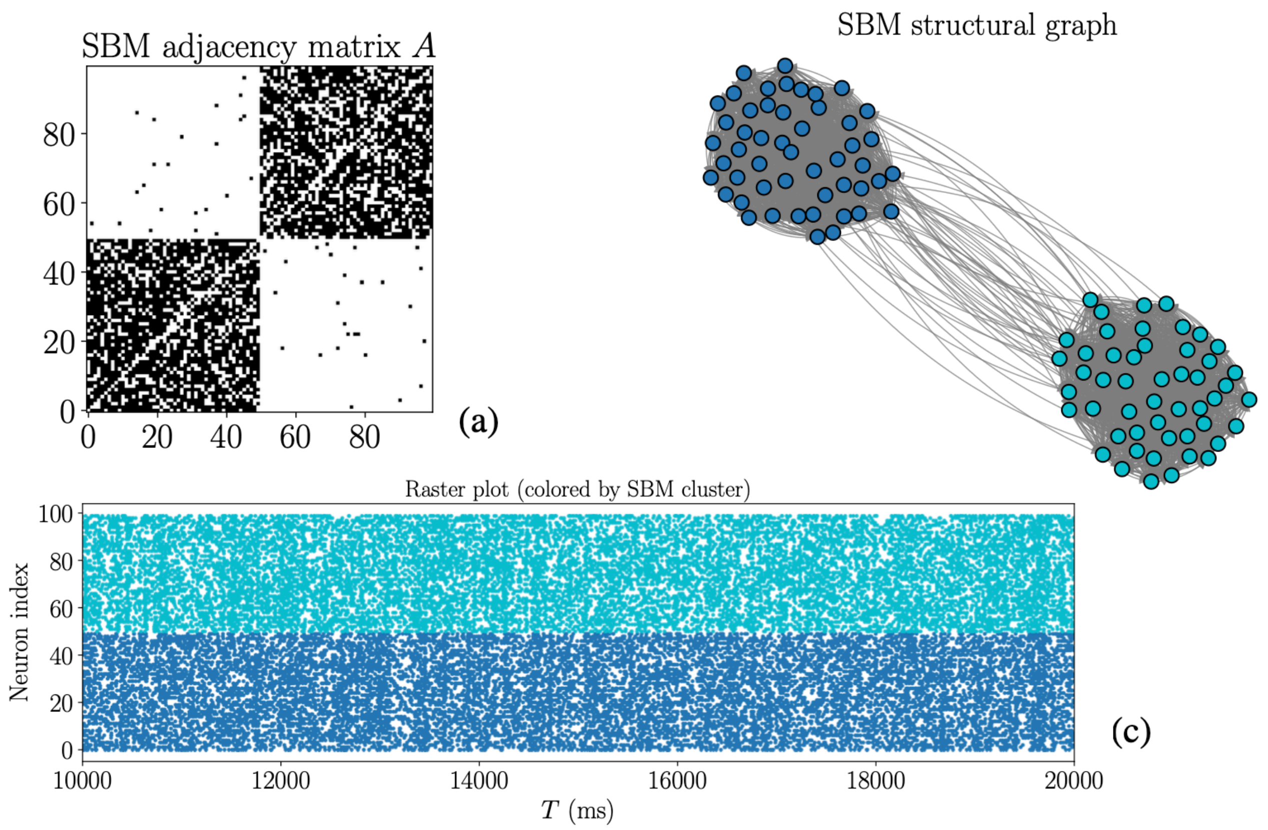
2. Subjected to external Poisson drive:

$$C_m \frac{dV_i}{dt} = g_{L,i}(V_L - V_i) + s_{E,i}(t)(V_E - V_i) + s_{I,i}(t)(V_I - V_i)\,.$$

$$\frac{ds_{E,j}}{dt} = -\frac{s_{E,j}}{\tau_E} + \sum_{i\in E} \frac{g_E}{\tau_E} A_{ij}\delta(t - t_i^{\text{spike}}) + \frac{f_{\mathscr{C}(j)}}{\tau_E}\frac{dQ_j(t)}{dt},$$

$$\frac{ds_{I,j}}{dt} = -\frac{s_{I,j}}{\tau_I} + \sum_{i\in I} \frac{g_I}{\tau_I} A_{ij}\delta(t - t_i^{\text{spike}})\,.$$

3. 80% E (excitatory), and 20% I (inhibitory) neurons.



Application to Spiking Neural Networks

1. We apply this methodology to a stochastic block model (SBM) network of Leaky Integrate and Fire (LIF) neurons,

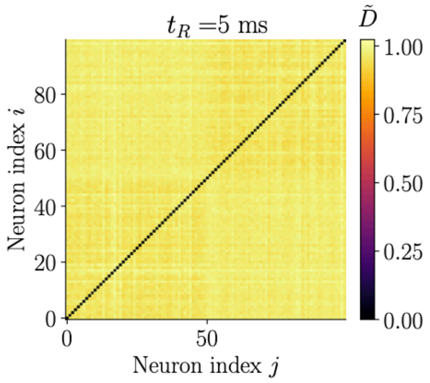
2. Subjected to external Poisson drive:

$$C_m \frac{dV_i}{dt} = g_{L,i}(V_L - V_i) + s_{E,i}(t)(V_E - V_i) + s_{I,i}(t)(V_I - V_i)$$

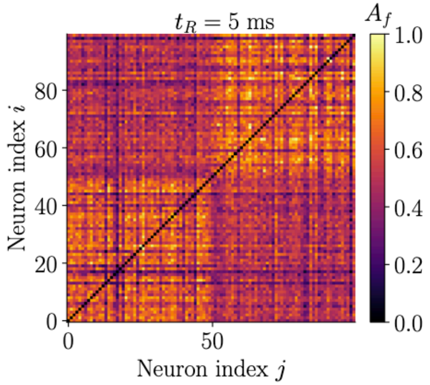
$$\frac{ds_{E,j}}{dt} = -\frac{s_{E,j}}{\tau_E} + \sum_{i \in E} \frac{g_E}{\tau_E} A_{ij} \delta(t - t_i^{\text{spike}}) + \frac{f_{\mathscr{C}(j)}}{\tau_E} \frac{dQ_j(t)}{dt},$$

$$\frac{ds_{I,j}}{dt} = -\frac{s_{I,j}}{\tau_I} + \sum_{i \in I} \frac{g_I}{\tau_I} A_{ij} \delta(t - t_i^{\text{spike}}) \,.$$

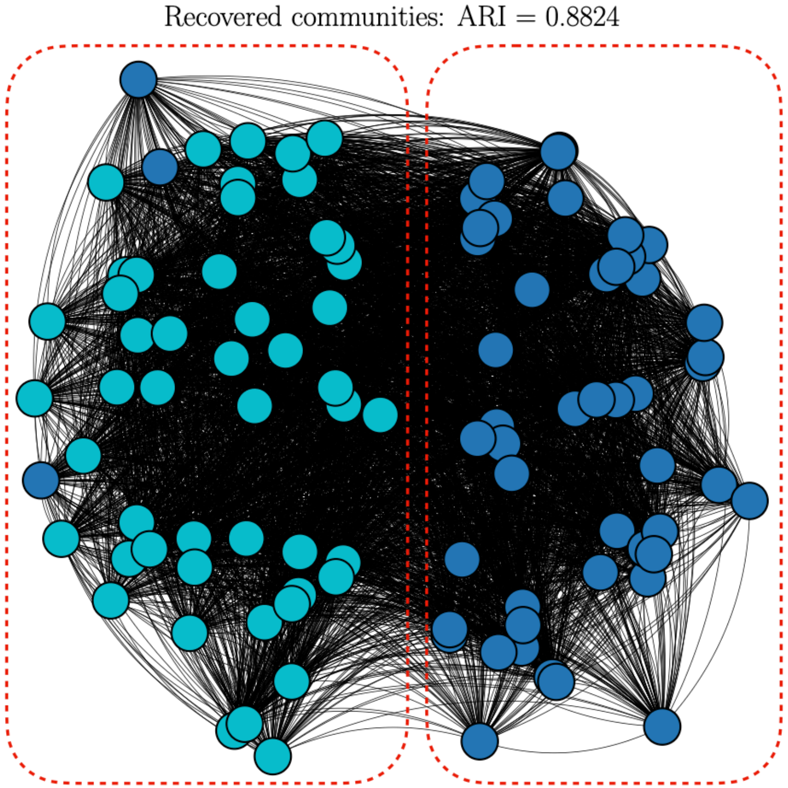
3. 80% E (excitatory), and 20% I (inhibitory) neurons.



(a)

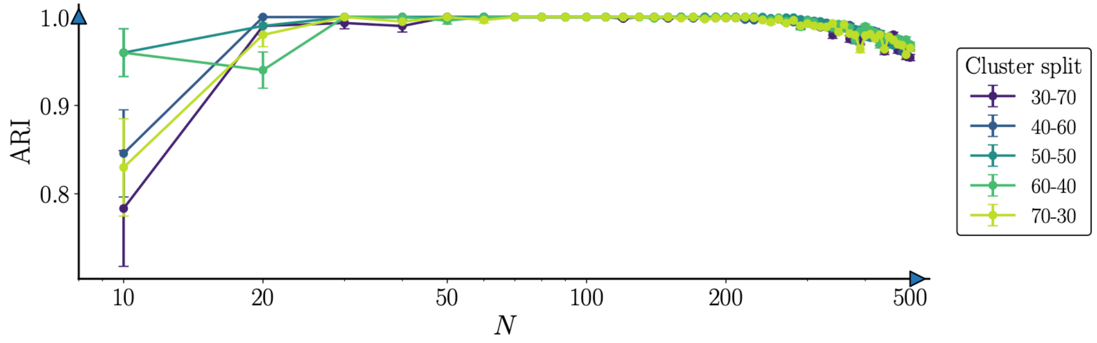


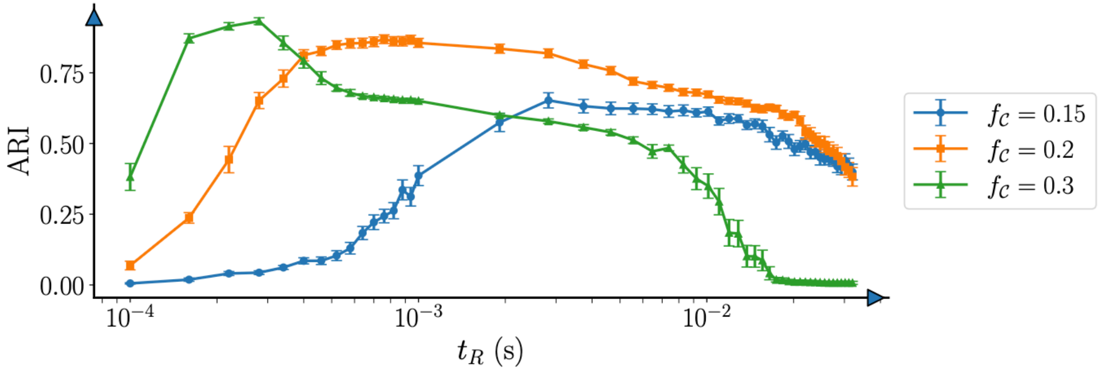
(b)

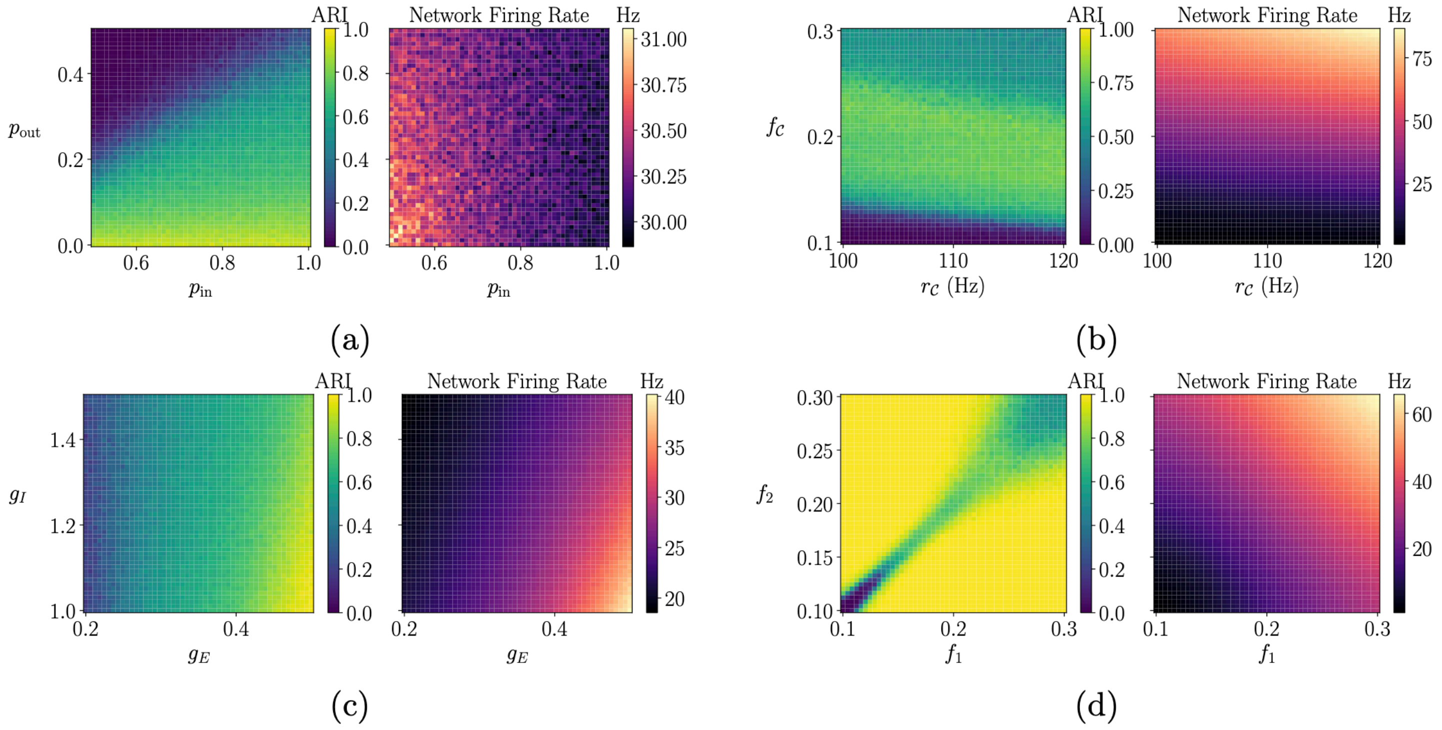


(c)

Application to Spiking Neural Networks







Application to Neuropixels data

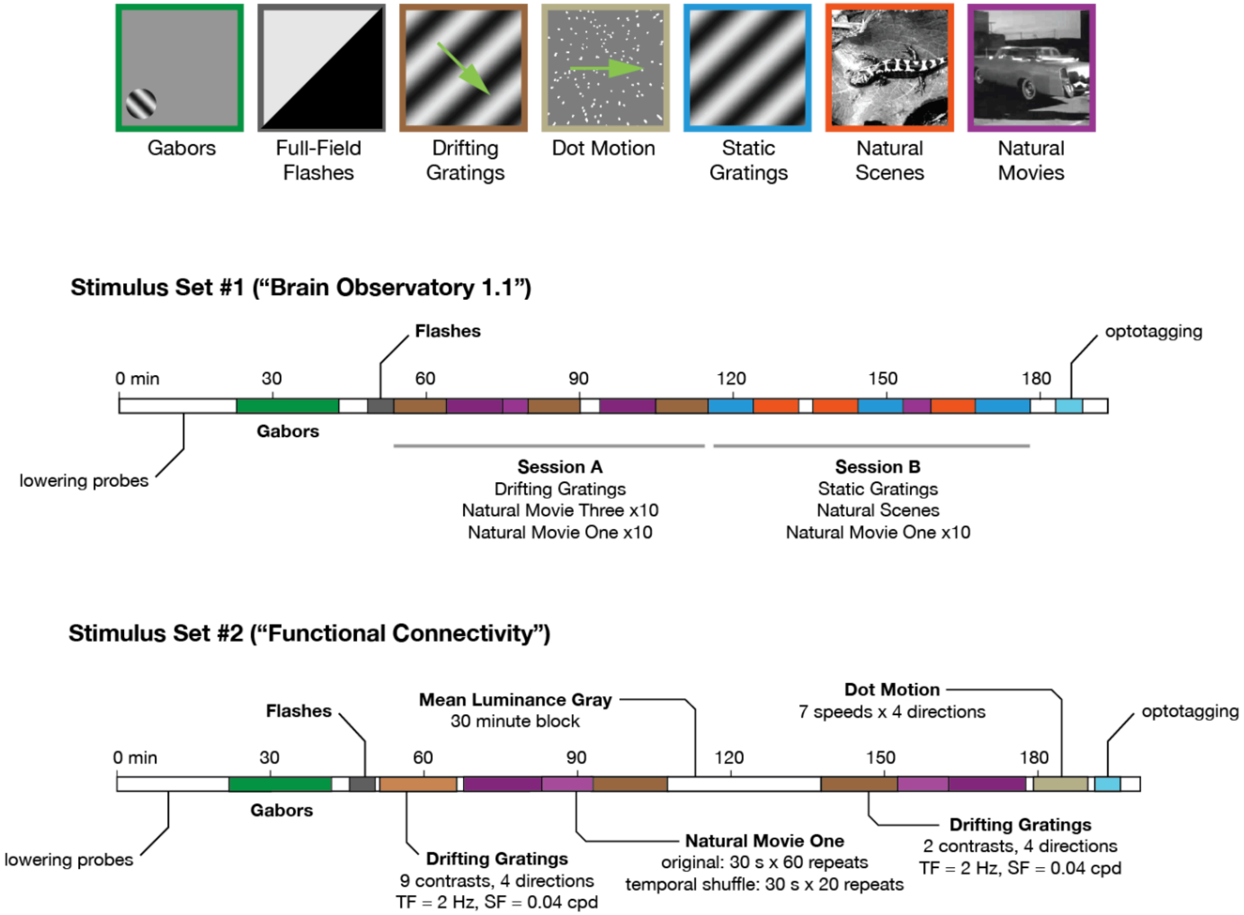
1. We also apply to visual coding Neuropixels data from the Allen Institute dataset

2. We only look at the “spontaneous” block between two stimuli for just the wild type mice.

3. Only the “functional connectivity” dataset.

Session ID	Specimen ID	Age (days)	Sex	Unit count	Channel count	Probe count
766640955	744912849	133.0	M	842	2233	6
767871931	753795610	135.0	M	713	2231	6
768515987	754477358	136.0	M	802	2217	6
771160300	754488979	142.0	M	930	2230	6
771990200	756578435	108.0	M	546	2229	6
774875821	759711152	114.0	M	649	2233	6
778240327	760938797	120.0	M	784	2234	6
778998620	759674770	121.0	M	793	2229	6
779839471	760960653	122.0	M	863	2220	6
781842082	760946813	126.0	M	833	2232	6
793224716	769319624	120.0	M	781	2229	6
819186360	800249587	128.0	F	531	1696	5
821695405	800250057	134.0	F	474	1856	5
847657808	827809884	126.0	F	874	2298	6

Table 1: Allen SDK extracellular electrophysiology (ecephys) data.

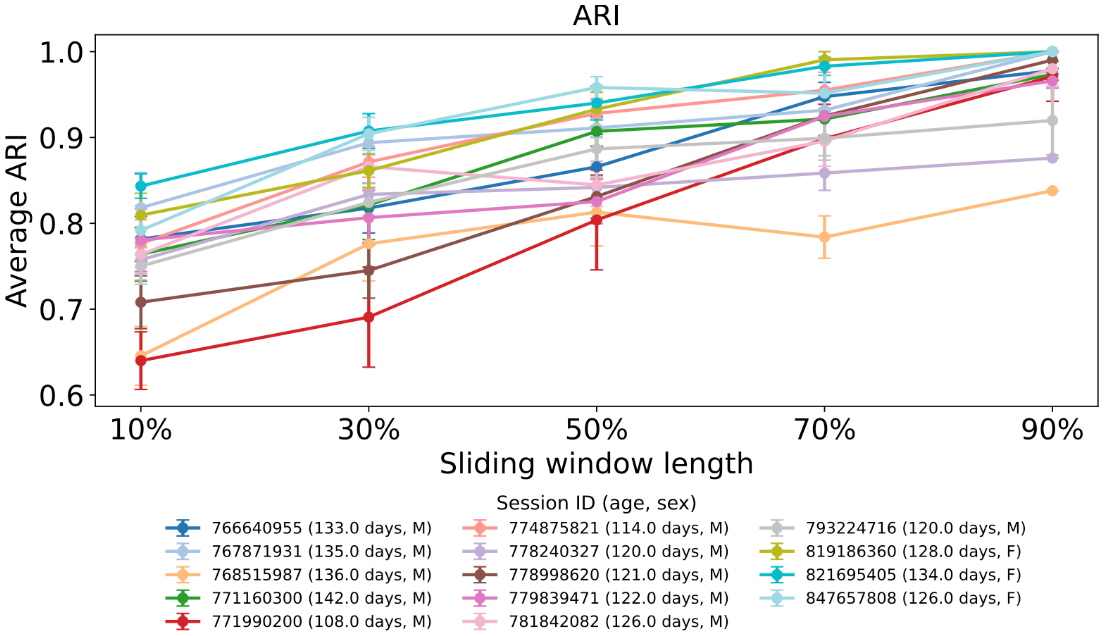


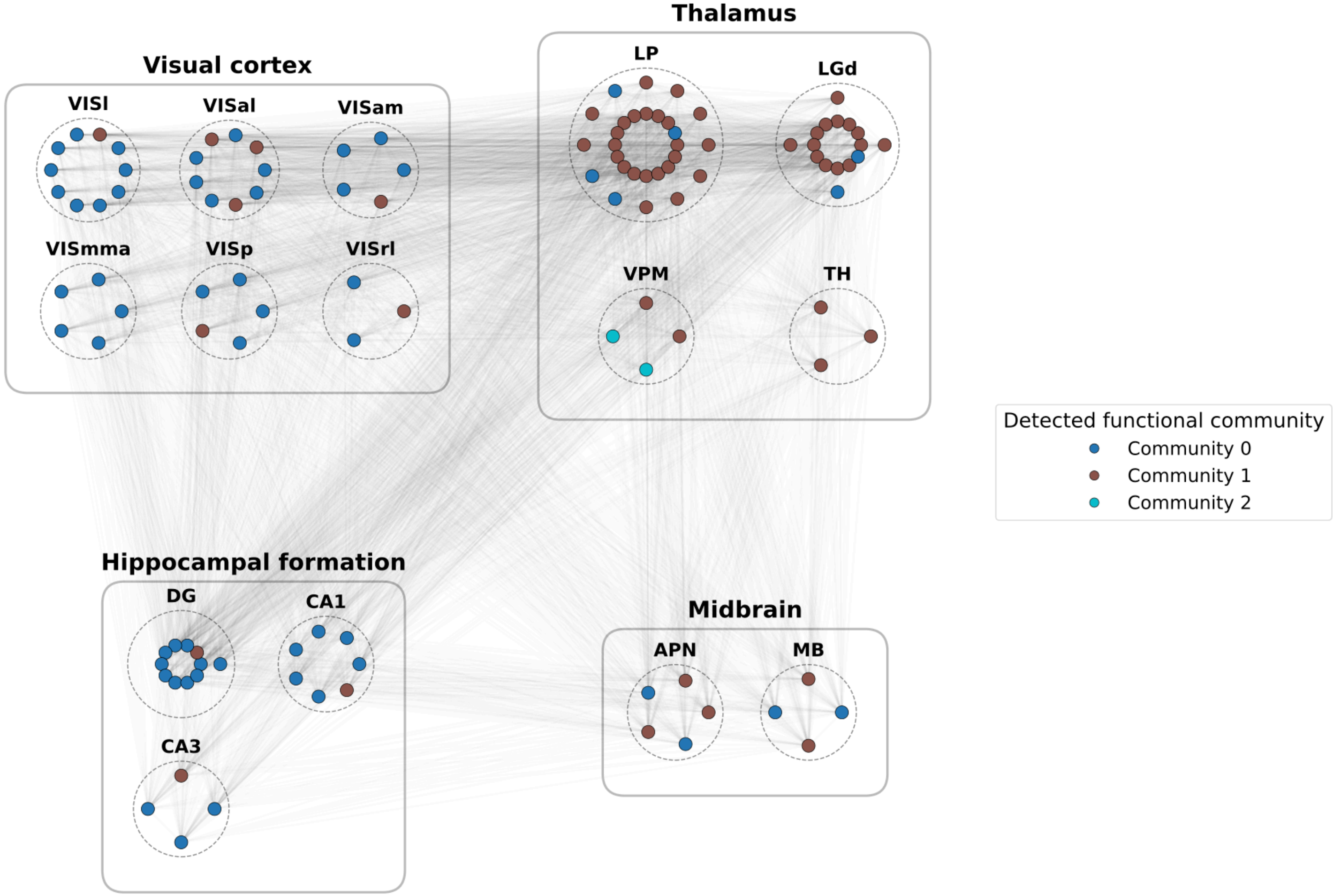
Visual stimuli sets showing “Brain Observatory 1.1” and “Functional Connectivity” types. Image credit: Allen Institute for Brain Science. [https://allensdk.readthedocs.io/en/latest/visual_coding_neuropixels.html]

Application to Neuropixels data

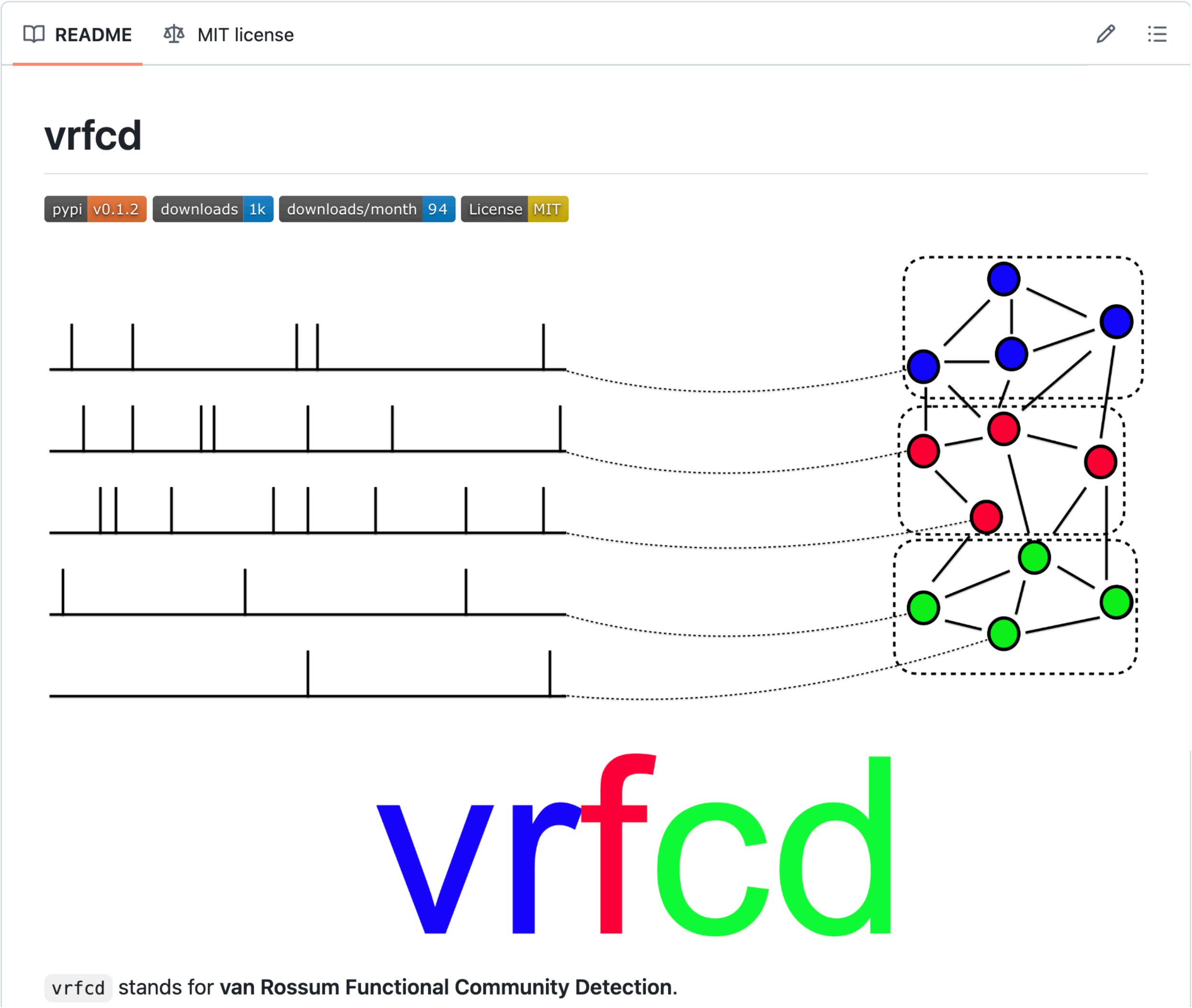
Stimulus ID	Mean duration (s)	Median duration (s)
0	60.067	60.067
3646	288.993	288.992
3797	61.802	61.802
4338	120.601	120.601
40639	1802.514	1802.506
40940	1.001	1.001

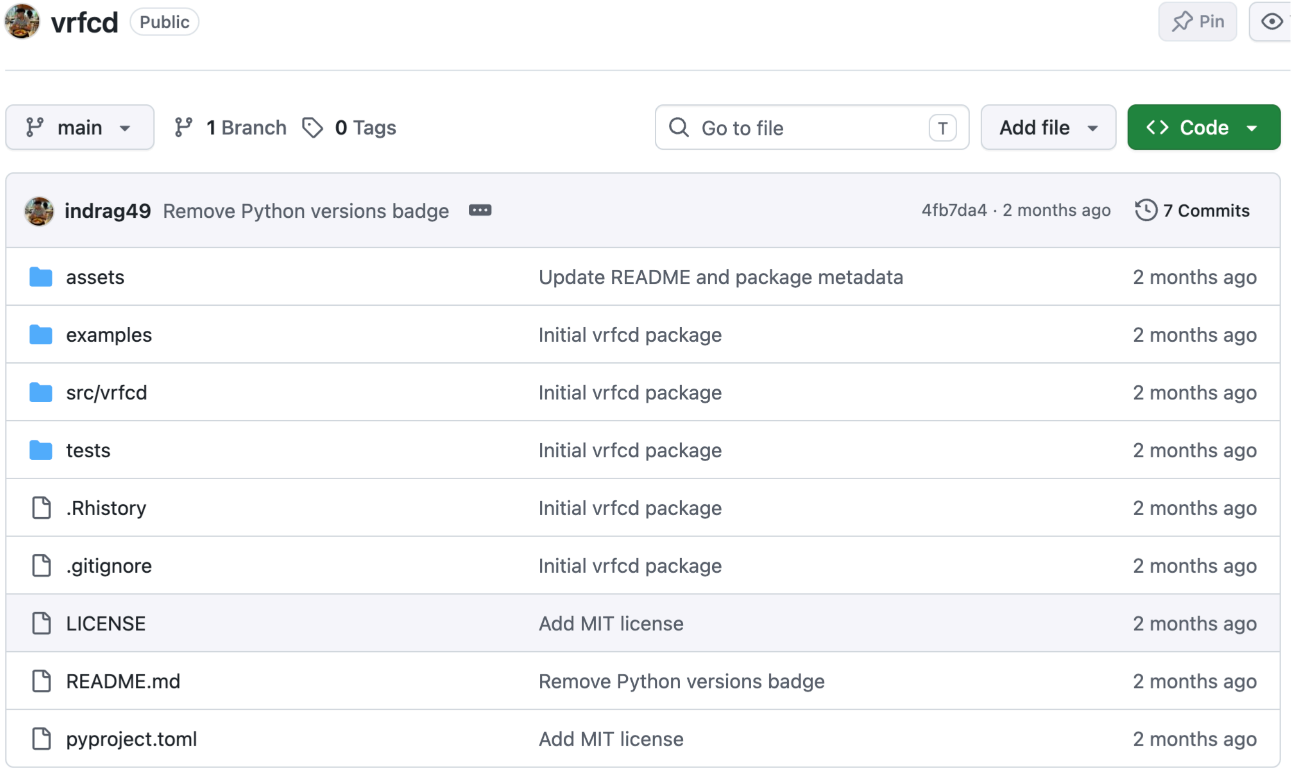
Table 2: Stimulus IDs with mean and median durations in seconds.





package: ‘vrfcd’ (pip install vrfcd)





Find it at github.com/indrag49/vrfcd

Future directions

1. Temporal and multiplex networks to track evolving communities.
2. Directed and causal connectivity beyond symmetric similarities.
3. Higher-order interactions beyond pairwise relations.
4. Alternative spike-train metrics: Victor-Purpura distance.
5. Better GPU and scalable implementations for larger Neuropixel datasets.

source: https://looneytunes.fandom.com/wiki/Wile_E._Coyote

Questions? I’m all ears