COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK

Toward Bayesian permutation inference for identifying neurons in *C. elegans*.

Gonzalo E. Mena[1,*], Scott Linderman [1,*], David Belanger [2], Jasper Snoek [2], John Cunningham[1,3], Liam Paninski[1,3],
1. Department of Statistics, Columbia University, New York, NY, USA 2. Google Brain, Cambridge, MA. 3. Center for Theoretical Neuroscience and Grossman Center for the Statistics of Mind, Columbia University, New York, NY, USA.,

Summary

Overarching goal

- State and infer Bayesian hierarchical models for the activity in C.elegans combining information (calcium traces) from several worms.
- This is possible as C.elegans nervous system is stereotypical, neurons and connectome don't change across individuals.

Challenge

- If neural identity is known for each trace, one can apply standard bayesian methodology
- In practice, laborious human supervision is needed to match recorded traces to canonical neural identities (i.e. names)

Our contribution

- We Developed three methods for learning latent matchings. These can be used in variational inference (VI) to jointly estimate a dynamical system and the matching between traces and true neural identities.
- Potentially it may serve to automatize the matching procedure.
- From a statistical machine learning perspective, the relevance is that outperforms a simple MCMC sampler for permutations.

Future work

- We used real connectome a position information. In the future we plan to use real traces.
- Two new levels of complexity: partially observed brain recordings, more sophisticated dynamical systems.

Model

We focus on a simple linear autoregressive model for neural dynamics,

$$\widetilde{Y}_t^{(j)} = (W \odot A)\widetilde{Y}_{t-1}^{(j)} + \epsilon_t^{(j)}, \tag{1}$$

where $W \in \mathbb{R}^{N\times N}$ is the weight matrix we wish to infer; $A \in \{0,1\}^{N\times N}$ is the known adjacency matrix or connectome; $\odot$ denotes element-wise multiplication; $\epsilon_t^{(j)} \sim \mathcal{N}(0, \sigma^2 I)$; and $\widetilde{Y}_t^{(j)} \in \mathbb{R}^N$ is the measured neural activity at time t in worm j. The catch is that $\widetilde{Y}_t^{(j)}$ is assumed to be in canonical order; i.e. in the same order as the rows and columns of W and A. We actually observe,

$$Y_t^{(j)} = P^{(j)}\widetilde{Y}_t^{(j)}, \tag{2}$$

vectors that are permuted by matrix $P^{(j)}$. In order to learn about W, we must also infer the permutation matrices. We place a Gaussian prior on W.

The permutation matrices are constrained by side information: we use neural position along the worm's body to constrain the possible neural identities for a given recorded neuron. We only allow an observed neuron to be mapped to a known identity if the observed location is within η of the expected location.

This is illustrated in Fig. 1B. We represent these constraints with the matrix $\mathcal{C}^{(j)}$ so that $\mathcal{C}^{(j)}_{mn} = 1$ if and only if observed neuron m is within η of canonical neuron n's expected location. An example is shown in Figure 1C. We let $P^{(j)}$ have a uniform prior over the set of matrices allowable under the given constraints. We aim to perform posterior inference of $p(\{W, P^{(j)}\} \mid A, \{Y^{(j)}\})$.

Experimental setup

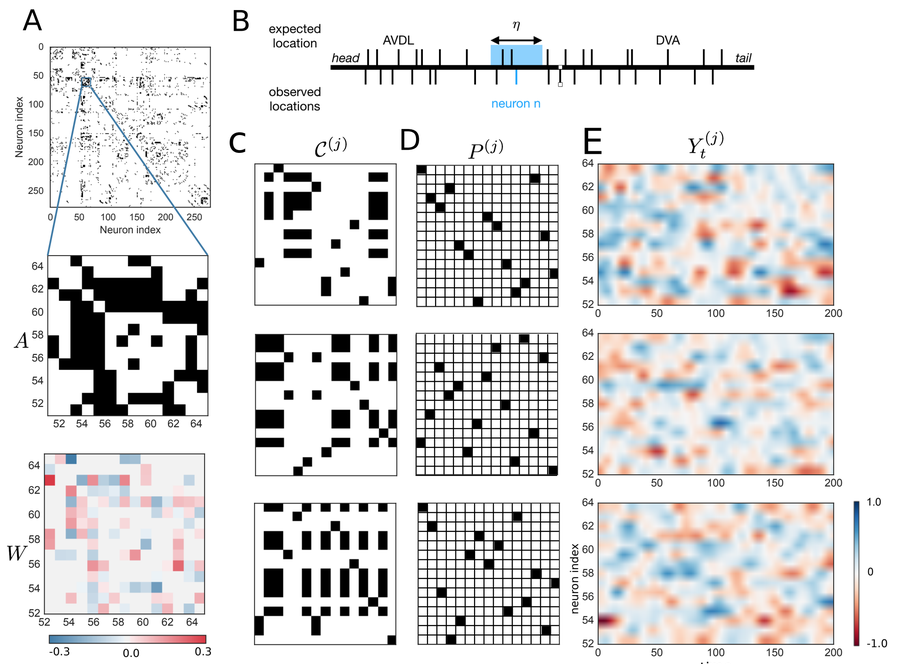

Figure: *Hierarchical Bayesian framework*. **A** We are given the actual adjacency matrix A from [5]. The full matrix is shown (top) along with a zoom-in to 14 neurons (center). We wish to infer the corresponding weight matrix W, an example of which is shown below. **B** We also know the typical locations of the neurons [6, 3]. Given observed locations, we constrain possible assignments to neuron identities within η of the observed location. **C** These constraints are represented as a matrix $\mathcal{C}^{(j)}$ for worm j which specifies possible assignments of observed neurons to known identities. This illustration shows three worms. **D** To infer the weights, we must first infer the permutation $P^{(j)}$ that matches the observed neurons in worm j to the set of known identities. **E** The observed data is a matrix $Y^{(j)}$ whose rows are ordered according to the order in which neurons were observed in that worm. The permutation matrix maps this to the canonical ordering of the adjacency and weight matrices. Given $\{Y^{(j)}\}_{j=1}^J$ and A, we infer $\{P^{(j)}\}_{j=1}^J$ and W.

Three reparameterizations for permutations

We developed three reparameterizations for permutations; all of them Inspired by the recently introduced methods by [1, 4]. Briefly, one can relax the *hard* sample of a category using the *soft*max, leading to a probability-simplex valued distribution, the so-called *Concrete* or *Gumbel-Softmax* distributions. These are proposed as alternative to REINFORCE [7] for learning latent discrete variables in stochastic computation graphs, e.g. approximate posterior inference. Specifically, one applies the the *reparameterization trick* machinery developed in [**?**].

Critically, REINFORCE cannot be used here, as it requires the evaluation of an intractable partition function for any non-trivial distribution on permutations. Extending *Gumbel-Softmax* is then the only choice. Two of our extensions, *Stick-breaking* and *rounding*, have tractable densities [2]. The third one, *Gumbel-Sinkhorn* does not does not. However, in practice the latter leads to best results as it allows a more efficient use of a regularizing prior [**?**].
In all relaxations we are particularly concerned with $\mathcal{B}_N$, the Birkhoff polytope or set of doubly-stochastic matrices. $\mathcal{B}_N$ is the convex hull of the set of permutation matrices; it is therefore the most suited to

Stick-Breaking and Rounding

On the stick-breaking method, we generalize the standard construction on the simplex [**?**] to stick-breaking of $\mathcal{B}_N$. We show how to consistently "break the stick" while satisfying both the row and column constraints that characterize a doubly stochastic matrix. For the rounding construction, we start with a noise distribution and force it to be close to permutation matrices by rounding them towards the extreme-points of the Birkhoff polytope (i.e. permutation matrices).

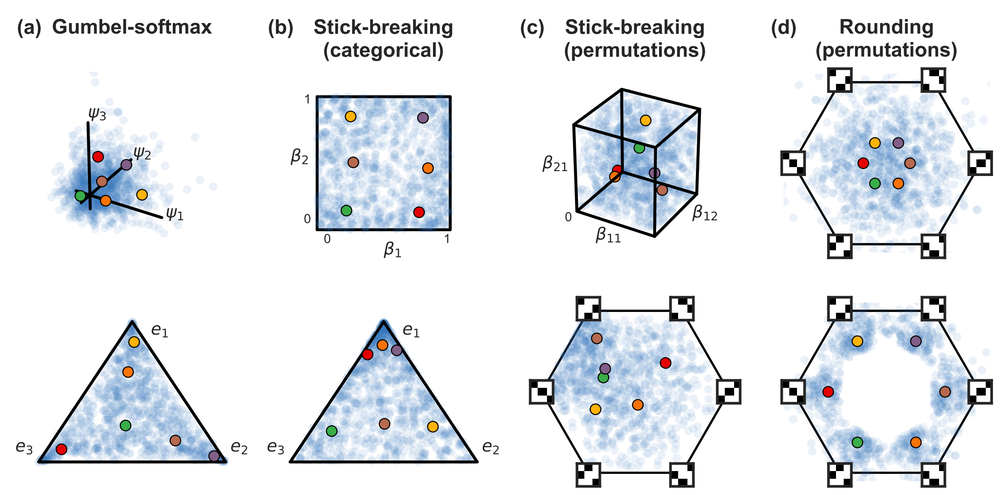

Figure: (a) The Gumbel-softmax, or "Concrete" transformation maps Gumbel r.v.'s (blue dots) to points in the simplex by applying the softmax. Colored dots are random variates that aid in visualizing the transformation. (b) Stick-breaking offers and alternative transformation for categorical inference, but the ordering of the stick-breaking induces an asymmetry in the transformation. (c) We extend this stick-breaking transformation to reparameterize the Birkhoff polytope, i.e. the set of doubly-stochastic matrices. We show how $\mathcal{B}_3$ is reparameterized in terms of matrices $B \in [0,1]^{2\times 2}$ These points are mapped to doubly-stochastic matrices, which we have projected onto $\mathbb{R}^2$ below. (d) Finally, we derive a "rounding" transformation that moves points in $\mathbb{R}^{N\times N}$ nearer to the closest permutation matrix. This is more symmetric, but does not map strictly onto $\mathcal{B}_N$.

Gumbel-Sinkhorn ($\mathcal{G}.\mathcal{S}$) distribution

Finally, for the Gumbel-Sinkhorn method we notice that the so-called *Sinkhorn operator* $S(\cdot)$, or infinite and successive row and column normalization of a matrix approximates the choice of a permutation through the matching operator $M(X)$; i.e. $M(X) = \lim_{\tau\to 0} S(X/\tau)$. By adding Gumbel noise we conceive the Gumbel Matching distribution and its approximation, the *G.S.* distribution. distribution, which approximates the sampling of a discrete distribution over permutations.

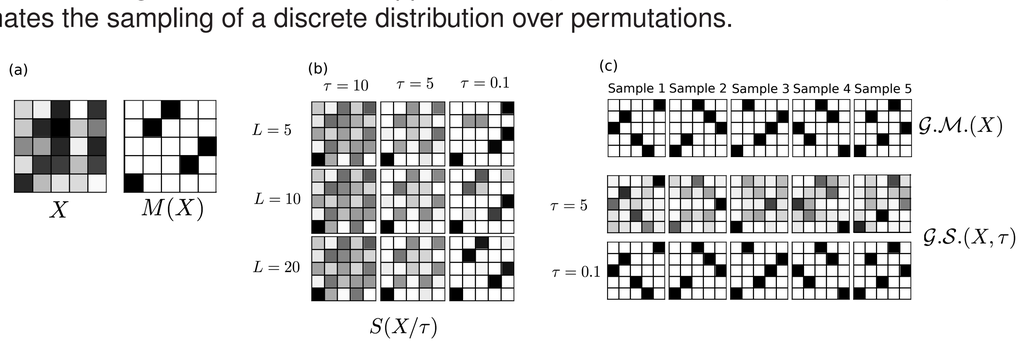

Figure: Illustrating the Matching and Sinkhorn operators, and the Gumbel-Matching and Gumbel-Sinkhorn distributions. Each 5x5 grid represents a matrix, with the shading indicating cell values (a) Matching operator $M(X)$ applied to a parameter matrix X. (b) Sinkhorn Operator $S(X/\tau)$ approximating $M(X)$ for different temperature τ and number of Sinkhorn iterations, L. (c). First row: samples from the Matching Sinkhorn distribution. Second and third rows: samples from the Gumbel-Sinkhorn distribution at two temperatures. At low temperature, both distributions are indistinguishable.

Results

We compared against three alternatives: (i) naïve variational inference, where we do not enforce the constraint that $P^{(j)}$ be a permutation and instead treat each row of $P^{(j)}$ as a Dirichlet distributed vector; (ii) MCMC, where we alternate between sampling from the conditionals of W (Gaussian) and $P^{(j)}$, from which one can sample by proposing local swaps, as described in [**?**], and (iii) maximum a posteriori estimation (MAP).
We found that our method outperforms these alternative approaches. When there are many possible candidates (Table 1) and when only a small proportion of neurons are known with certitude (Table 2), variational inference via continuous relaxation with the Gumbel-Sinkhorn method performs best.

Table: Accuracy in the C.elegans neural identification problem, for varying mean number of candidate neurons (10, 30, 45, 60) and number of worms.

	10		30		45		60	
	1 worm	4 worms	1 Worm	4 worms	1 worm	4 worms	1 worms	4 worms
NAIVE VI	.34	.32	.16	.16	.13	.12	.11	.12
MAP	.34	.32	.17	.17	.14	.13	.13	.12
MCMC	.34	.65	.18	.28	.14	.17	.13	.15
VI	**.79**	**.94**	**.4**	**.69**	**.25**	**.51**	**.21**	**.44**

Table: Accuracy in inferring true neural identity for different of proportion of known neurons and η.

	40.%		30.%		20.%		10.%	
	$\eta = 0.1$	$\eta = 0.2$	$\eta = 0.1$	$\eta = 0.2$	$\eta = 0.1$	$\eta = 0.2$	$\eta = 0.1$	$\eta = 0.2$
Naive VI	.43	.41	.33	.31	.23	.22	.12	.1
MAP	.42	.41	.33	.32	.23	.22	.12	.11
MCMC	.85	.80	.52	.46	.3	.26	.15	.12
VI	**.97**	**.96**	**.92**	**.84**	**.74**	**.58**	**.44**	**.23**

Conclusion

Our results provide promising evidence that a Bayesian hierarchical approach to the study of neural dynamics on C. elegans is feasible. We note we made many simplifying assumptions that are not justified in practice: first, we assumed a linear dynamical system, while actual dynamics are highly nonlinear [**?**]. Fortunately, there exist many methods for inference in nonlinear systems [**? ?**]. Also, we assumed all neurons were observed, while in reality we only see about 100 neurons at a time. The methods of **?**] may help infer the weights, but reasoning about partial permutations requires more care.

References

[1] E. Jang, S. Gu, and B. Poole. Categorical reparameterization with Gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.
[2] S. W. Linderman, G. E. Mena, H. Cooper, L. Paninski, and J. P. Cunningham. Reparameterizing the Birkhoff polytope for variational permutation inference. *arXiv preprint arXiv:1710.09508*, 2017.
[3] R. Lints, Z. F. Altun, H. Weng, T. Stephney, G. Stephney, M. Volaski, and D. H. Hall. WormAtlas Update. 2005.
[4] C. J. Maddison, A. Mnih, and Y. W. Teh. The Concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

Email: gem2131@columbia.edu