Neural decoding of directions using generative models

Kristopher T. Jensen[@1], **Ta-Chu Kao**[1,2], **and Guillaume Hennequin**[1]

[1] Computational and Biological Learning Lab, Department of Engineering, University of Cambridge, Cambridge, UK
[2] Meta Reality Labs

[@] kristopher.torp@gmail.com

In neuroscience, we often want to understand what is being represented by some population of neurons. Often, we know that such a population represents physically relevant variables such as location in space or head direction, and we want to understand how the state being represented changes over time. This is commonly achieved using some form of neural decoding that maps from a set of neural activations to a predicted state. However, most such decoding models are inherently Euclidean; that is, they predict features such as location or velocity that consist of scalar variable. This is the case despite ample evidence that the brain commonly represents non-Euclidean quantities such as head direction, phase within a task, or location using a periodic grid. In this work, we develop a set of new neural decoding methods that facilitate the inference of *non-Euclidean* quantities from neural recordings in a supervised setting. We do this by learning a generative model *from* the manifold *to* neural activations on the training data and inverting this generative model to infer the instantenous hidden state at test time. By using an explicitly probabilistic model, we are able to capture features of neural data such as smooth tuning curves, continuity in time, and non-Gaussian firing statistics.

Introduction

It is common in neuroscience to want to 'decode' some quantity of interest from neural recordings. In many cases, we have a set of labeled data for training such a model, and we then want to predict the represented quantities on a set of held-out data where we don't have access to labels. However, the brain often represents non-Euclidean quantities, such as head direction of 'toroidal' grid cells (Jensen et al., 2020; Turner-Evans et al., 2020; Seelig and Jayaraman, 2015; Chaudhuri et al., 2019; Gardner et al., 2022). It is therefore natural to ask how we can generalize our decoding methods to these non-Euclidean problem settings. In general this is difficult as most of our out-of-the-box methods predict Euclidean quantities by default, which can lead to spurious discontinuities or reduced performance when projected onto a non-Euclidean manifold of interest post-hoc. In this work, we use Gaussian processes to define smooth functions on manifolds and invert these to generate predictions in a supervised learning setting.

Method

Problem setting

We imagine a problem setting where we are able to record a quantity of interest $\boldsymbol{Z} \in \mathcal{M}^T$ over some period of time T together with the associated responses of a population of N neurons

$\boldsymbol{X} \in \mathbb{R}^{N \times T}$. Given this training data, our goal is to *predict* the state $\boldsymbol{Z}^*$ on the basis of neural data $\boldsymbol{X}^*$ recorded at some set of test points. A common approach in neuroscience is to train a discriminative model $\boldsymbol{z}_t \approx f_\theta(\boldsymbol{x}_t)$ with parameters θ that are chosen to minimize the training loss

$$\theta = \text{argmin}_\theta \sum_t (\boldsymbol{z}_t - f_\theta(\boldsymbol{x}_t))^2. \tag{1}$$

Predictions are then made at any test point by passing the data through the function f. However, this approach suffers from a few shortcomings. One is that $\boldsymbol{z}$ is often some relatively noise-free quantity while neural activity $\boldsymbol{x}$ provides a very noisy readout of the underlying process. Another is that it is non-trivial to build in inductive biases common in neuroscience, such as the assumption that neural activity should be a smooth function of the underlying process $\boldsymbol{z}$ (Jensen et al., 2020; Stringer et al., 2019). Finally, it is non-trivial to define f_θ in cases where $\boldsymbol{z}$ does not live in Euclidean space.

Inverting a generative model

To overcome the challenges highlighted in the previous section, we instead define a generative model $\boldsymbol{x}_t \sim p_\theta(\boldsymbol{x}_t|\boldsymbol{z}_t)$ that maps *from* the latent quantity $\boldsymbol{z}$ *to* neural activity $\boldsymbol{x}_t$. Following previous work in neuroscience, we can then model $p(y_{nt}|\boldsymbol{z}_t)$ as a *Gaussian process* (Rasmussen and Williams, 2006), which allows us to build in notions of smoothness and non-Gaussian noise models important for modelling neural data (Wu et al., 2017, 2018; Jensen et al., 2020, 2021, 2022). A Gaussian process defines a jointly Gaussian distribution over the observations,

$$p(\boldsymbol{x}_n|\boldsymbol{Z}) = \mathcal{N}(\boldsymbol{x}_n; \boldsymbol{\mu} = 0, \boldsymbol{\Sigma} = k(\boldsymbol{Z}, \boldsymbol{Z})). \tag{2}$$

Here we have assumed that the *mean function* $\boldsymbol{\mu}$ is 0, and $k(\cdot, \cdot) : \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ is a *kernel* which defines a prior distribution over functions. Importantly, such kernels can also be defined for non-Euclidean manifolds (Jensen et al., 2020; Borovitskiy et al., 2020; Feragen et al., 2015), which allows us to build in the prior assumption that neural activity should be a smooth function on the manifold. In other words, we can build in the assumption that latent states $\boldsymbol{z}_1, \boldsymbol{z}_2$, which are separated by a small geodesic distance, should give rise to similar distributions over neural activity. Finally, we assume the generative model to factorize across neurons

$$p(\boldsymbol{X}|\boldsymbol{Z}) = \prod_n p(\boldsymbol{x}_n|\boldsymbol{Z}) \tag{3}$$

As shown in Figure 1, such a GP-based model leads to a much better data fit than other common models in neuroscience.

Importantly, having defined our generative model $p(\boldsymbol{X}|\boldsymbol{z})$, we can now *invert* it at test time. In other words, we can compute the posterior distribution over $\boldsymbol{x}^*$ conditioned on (i) the test observations $\boldsymbol{x}^*$, and (ii) the training data $(\boldsymbol{x}, \boldsymbol{z})$

$$p(\boldsymbol{Z}^*|\boldsymbol{X}^*, (\boldsymbol{X}, \boldsymbol{Z})) \propto p(\boldsymbol{X}^*|\boldsymbol{z}^*, (\boldsymbol{X}, \boldsymbol{z}))p(\boldsymbol{Z}^*) \tag{4}$$

$$= p(\boldsymbol{Z}^*) \prod_n p(\boldsymbol{x}_n^*|\boldsymbol{Z}^*, (\boldsymbol{x}_n, \boldsymbol{Z})). \tag{5}$$

Here, $p(\boldsymbol{z}^*)$ is a prior which we will discuss further below. $p(\boldsymbol{x}_n^*|\boldsymbol{z}^*, (\boldsymbol{x}_n, \boldsymbol{z}))$ is a standard Gaussian process posterior, which can be computed in closed form (Rasmussen and Williams, 2006).

Finally, we note that the kernel $k(\cdot, \cdot)$ often contains some set of hyperparameters θ characterizing e.g. the length scale and height of the tuning curve. It is common to optimize these on the training data by maximizing the marginal likelihood $\log p(\boldsymbol{X}|\boldsymbol{Z})$, which can be computed in closed form (Rasmussen and Williams, 2006).

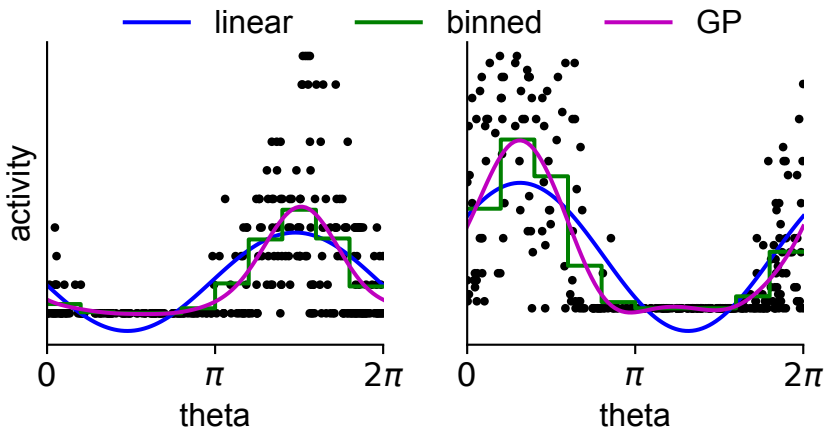

Figure 1: **Supervised learning with generative models. (A)** Here we show the utility of Gaussian processes for modeling neural data in the common setting of decoding on a ring using example neurons from Peyrache et al. (2015a). Black dots indicate empirically observed spike counts as a function of head direction (x axis). Blue curve indicates the best fit in the space of tuning curves that are linear in $\sin\theta$ and $\cos\theta$. Green curve indicates the best fit to the data of a binned decoder with a Poisson noise model. Purple curve indicates a Gaussian process fit to the data, which yields a better model of the data than the linear or binned approaches while also generalizing more readily to higher-dimensional manifolds.

Variational inference

The closed form expressions for the marginal likelihood and posterior predictive distribution for Gaussian processes are exact but suffer from two major shortcomings. Firstly, they can only be derived for a Gaussian noise model, while neural recordings are more accurately modelled using discrete noise models. Secondly, they suffers from a computational complexity of $\mathcal{O}(T^3)$, which can be prohibitively expensive. To overcome these two challenges, we resort to the stochastic variational GP (SVGP) framework of Hensman et al. (2015).

Choice of priors

In Equation 5, we introduced a prior $p(\boldsymbol{Z})$ over the latent states. A simple choice of prior would be a *uniform* prior on the manifold

$$p_{\mathcal{U}}(\boldsymbol{Z}) = \prod_t p_{\mathcal{U}}(\boldsymbol{z}_t) = V_{\mathcal{M}}^{-T}, \tag{6}$$

where $V_{\mathcal{M}}$ is the volume of the manifold (Jensen et al., 2020). However, we do in fact often have some prior knowledge about the processes being represented by the brain. In particular, most physically relevant processes tend to unfold continuously in time, which severely constrains the set of plausible trajectories $\boldsymbol{Z}$. We therefore expect that building in this prior knowledge would improve our predictive power at test time.

In Euclidean space, a simple way of building in this prior would be to use an *autoregressive* process, reminiscent of e.g. the Kalman filter commonly used for brain machine interfaces. In an autoregressive process of order M, we *predict* $\boldsymbol{z}_t$ from the previous m states as

$$\boldsymbol{z}_t \approx \boldsymbol{z}_{t-1} + \sum_{m=1}^{M} a_m \boldsymbol{z}_{t-m}, \tag{7}$$

or from their displacements

$$\boldsymbol{z}_t \approx \boldsymbol{z}_{t-1} + \sum_{m=1}^{M-1} a_m(\boldsymbol{z}_{t-m} - \boldsymbol{z}_{t-m-1}). \tag{8}$$

If we assume the error of this estimate to be Gaussian distributed, this induces a prior of $\boldsymbol{z}_t$ of the form

$$\boldsymbol{z}_t \sim \mathcal{N}(\boldsymbol{z}_t; \boldsymbol{\mu} = \boldsymbol{z}_{t-1} + \sum_{m=1}^{M-1} a_m \delta_{t-m}, \sigma^2), \tag{9}$$

where we have defined $\delta_{t-m} := \boldsymbol{z}_{t-m} - \boldsymbol{z}_{t-m-1}$ and σ^2 is the noise variance. This model has M parameters, namely σ^2 and $\{a_m\}_{m=1}^{M-1}$, which can either be fitted to the training data or treated as free parameters at inference time.

Unfortunately, $a_m \delta_m$ is not well-defined on Riemannian manifolds since they are not vector spaces, so subtraction and scalar multiplication is not defined. However, we can generalize the notion of an autoregressive prior to such manifolds by working in the *tangent space* of the manifold, which is in fact a vector space. We do this by defining

$$\delta_{t-m} := \mathrm{Log}\left[\boldsymbol{z}_{t-m-1}^{-1} \cdot \boldsymbol{z}_{t-m}\right]. \tag{10}$$

Here, $\boldsymbol{z}$ indicates a group element, $\cdot$ indicates group multiplication, $\boldsymbol{z}^{-1}$ indicates its group inverse (i.e. $\boldsymbol{z}^{-1} \cdot \boldsymbol{z} = \mathcal{I}$, the identity element), and Log indicates the logarithmic map from the group onto the tangent space. Notably, this recovers our previous definition of δ_{t-m} in the special case where the manifold $\mathcal{M}$ is Euclidean space. We can now define our approximation error

$$\epsilon := \mathrm{Exp}\left[\delta_t + \sum_{m=1}^{M-1} a_m \delta_{t-m}\right], \tag{11}$$

which has been projected onto the group by the *exponential map* Exp[]. Finally, we assume that this approximation error has a wrapped Gaussian distribution, the density of which can be estimated following Falorsi and Forré (2020):

$$p(\epsilon) = \sum_{x s.t. \mathrm{Exp}[x]=\epsilon} \mathcal{N}(x; 0, \sigma^2)|J(x)|^{-1}, \tag{12}$$

where $|J(x)|$ is the Jacobian of the exponential map at x.

Results

In this section, we apply our method to a range of biological and synthetic datasets to illustrate its utility for neuroscience.

Performance on biological data

We start by considering a dataset recorded by Peyrache et al. (2015b,a) from the mouse anterodorsal thalamic nucleus during free foraging. This dataset has previously been studied with the purpose of characterizing the head direction circuit of the mouse and as a testbed for various unsupervised learning methods in neuroscience (Chaudhuri et al., 2019; Jensen et al., 2020; Liu and Lengyel, 2021; Rubin et al., 2019). We split the data into 9 distinct training datasets, each with its own test dataset (which formed a separate training dataset). We then performed supervised decoding of the test head direction using three methods (Figure 1). First,

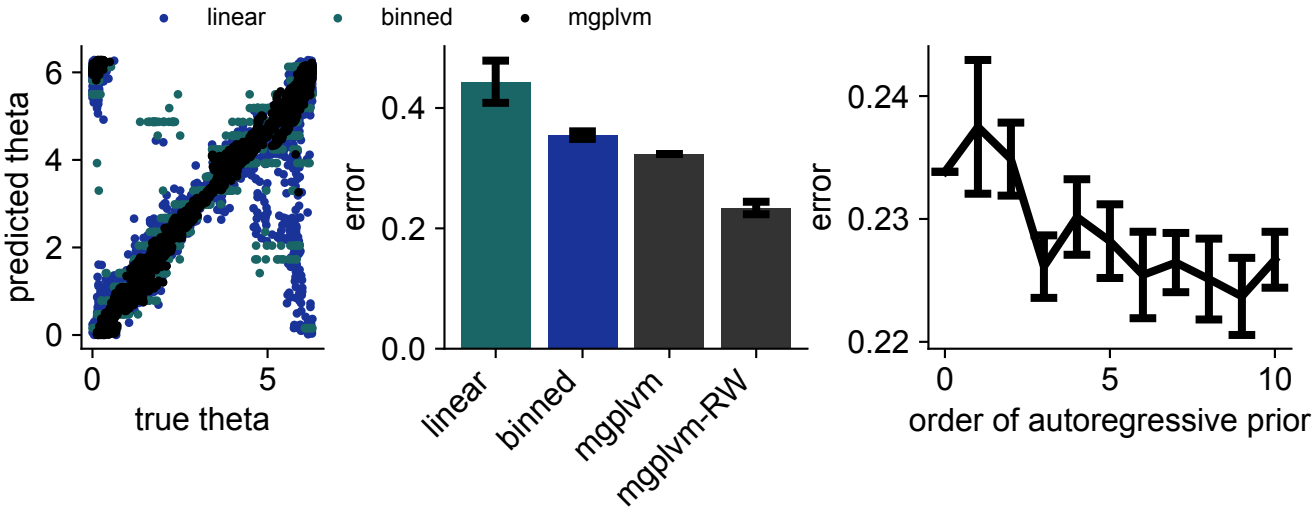

Figure 2: **Supervised learning for mouse head direction data. (A)** Predicted head direction as a function of the true head direction for an example dataset recorded by Peyrache et al. (2015a), with head direction predicted using either a linear model (blue), binned decoder (green) or mgplvm (black). **(B)** Mean discrepancy between true and predicted head direction for a linear decoder, binned decoder, mGPLVM with uniform prior, or mGPLVM with a random walk prior. Discrepancy was quantified as the geodesic distance between the true and inferred head direction at each time point, and error bars indicate standard error across 10 partitions of the data. **(C)** Mean discrepancy between true and predicted head direction as a function of the order of the mGPLVM autoregressive prior. An order of 0 corresponds to the random walk prior from (B) and error bars indicate standard error across 10 partitions of the data.

we considered a decoder that predicted $\sin\theta$ and $\cos\theta$ as a linear function of $\boldsymbol{x}$. Second, we considered a binned Bayesian decoder with a Poisson noise model. These two approaches are commonly used in the neuroscience literature for decoding head direction. Finally, we considered our generative modelling approach, using a 'random walk' prior (i.e. an autoregressive process of order 0). This mGPLVM-based approach appeared to capture the true head direction more faithfully than the alternatives, suggesting that it is a useful method for neural decoding (Figure 2A). We then quantified the error across all 10 data partitions, with the error ϵ defined as the average geodesic distance between the true and predicted head direction. Here we found that mGPLVM with a uniform prior ($\epsilon = 0.324$) outperformed both the linear ($\epsilon = 0.444 \pm 0.035$) and binned ($\epsilon = 0.355 \pm 0.006$) decoders. However, the benefit of such an explicitly generative model became particularly pronounced after introducing a random walk prior which reduced the error to $\epsilon = 0.234 \pm 0.010$ (Figure 2B).

Finally, to better understand the effect of the prior, we modeled the data with autoregressive priors of increasing order. For these analyses, the standard deviation of the prior (σ^2) was fitted to the training data while the autoregressive coefficients (a_m) were inferred at test time. Here we found a modest but significant effect of increasing the order of the prior, with a correlation between order and error of $\rho = -0.82$ (Figure 2C).

References

Borovitskiy, V., Terenin, A., Mostowsky, P., and Deisenroth, M. P. (2020). Matérn Gaussian processes on Riemannian manifolds. *arXiv preprint arXiv:2006.10160*.

Chaudhuri, R., Gerçek, B., Pandey, B., Peyrache, A., and Fiete, I. (2019). The intrinsic attractor manifold and population dynamics of a canonical cognitive circuit across waking and sleep. *Nature neuroscience*, 22(9):1512–1520.

Falorsi, L. and Forré, P. (2020). Neural ordinary differential equations on manifolds. *arXiv preprint arXiv:2006.06663*.

Feragen, A., Lauze, F., and Hauberg, S. (2015). Geodesic exponential kernels: When curvature and linearity conflict. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3032–3042.

Gardner, R. J., Hermansen, E., Pachitariu, M., Burak, Y., Baas, N. A., Dunn, B. A., Moser, M.-B., and Moser, E. I. (2022). Toroidal topology of population activity in grid cells. *Nature*, 602(7895):123–128.

Hensman, J., Matthews, A., and Ghahramani, Z. (2015). Scalable variational gaussian process classification. In *Artificial Intelligence and Statistics*, pages 351–360. PMLR.

Jensen, K., Kao, T.-C., Stone, J., and Hennequin, G. (2021). Scalable bayesian gpfa with automatic relevance determination and discrete noise models. *Advances in Neural Information Processing Systems*, 34:10613–10626.

Jensen, K., Kao, T.-C., Tripodi, M., and Hennequin, G. (2020). Manifold gplvms for discovering non-euclidean latent structure in neural data. *Advances in Neural Information Processing Systems*, 33:22580–22592.

Jensen, K. T., Liu, D., Kao, T.-C., Lengyel, M., and Hennequin, G. (2022). Beyond the euclidean brain: inferring non-euclidean latent trajectories from spike trains. *bioRxiv*.

Liu, D. and Lengyel, M. (2021). A universal probabilistic spike count model reveals ongoing modulation of neural variability. *Advances in Neural Information Processing Systems*, 34:13392–13405.

Peyrache, A., Lacroix, M. M., Petersen, P. C., and Buzsáki, G. (2015a). Internally organized mechanisms of the head direction sense. *Nature Neuroscience*, 18(4):569–575.

Peyrache, A., Petersen, P., and Buzsáki, G. (2015b). Extracellular recordings from multi-site silicon probes in the anterior thalamus and subicular formation of freely moving mice. *CRCNS.org*. Dataset. https://doi.org/10.6080/K0G15XS1.

Rasmussen, C. E. and Williams, C. K. (2006). *Gaussian processes for machine learning*. MIT press Cambridge, MA.

Rubin, A., Sheintuch, L., Brande-Eilat, N., Pinchasof, O., Rechavi, Y., Geva, N., and Ziv, Y. (2019). Revealing neural correlates of behavior without behavioral measurements. *Nature communications*, 10:1–14.

Seelig, J. D. and Jayaraman, V. (2015). Neural dynamics for landmark orientation and angular path integration. *Nature*, 521(7551):186–191.

Stringer, C., Pachitariu, M., Steinmetz, N., Carandini, M., and Harris, K. D. (2019). High-dimensional geometry of population responses in visual cortex. *Nature*, 571(7765):361–365.

Turner-Evans, D. B., Jensen, K. T., Ali, S., Paterson, T., Sheridan, A., Ray, R. P., Wolff, T., Lauritzen, J. S., Rubin, G. M., Bock, D. D., and Jayaraman, V. (2020). The neuroanatomical ultrastructure and function of a biological ring attractor. *Neuron*, 108:145–163.

Wu, A., Pashkovski, S., Datta, S. R., and Pillow, J. W. (2018). Learning a latent manifold of odor representations from neural responses in piriform cortex. *Advances in Neural Information Processing Systems*, 31.

Wu, A., Roy, N. A., Keeley, S., and Pillow, J. W. (2017). Gaussian process based nonlinear latent structure discovery in multivariate spike train data. *Advances in neural information processing systems*, 30.