A universal probabilistic spike count model reveals ongoing modulation of neural variability

David Liu **Máté Lengyel**
Department of Engineering
University of Cambridge
dl543@cam.ac.uk, m.lengyel@eng.cam.ac.uk

Abstract

Neural responses are variable: even under identical experimental conditions, single neuron and population responses typically differ from trial to trial and across time. Recent work has demonstrated that this variability has predictable structure, can be modulated by sensory input and behaviour, and bears critical signatures of the underlying network dynamics and computations. However, current methods for characterising neural variability are primarily geared towards sensory coding in the laboratory: they require trials with repeatable experimental stimuli and behavioural covariates. In addition, they make strong assumptions about the parametric form of variability, rely on assumption-free but data-inefficient histogram-based approaches, or are altogether ill-suited for capturing variability modulation by covariates. Here we present a universal probabilistic spike count model that eliminates these shortcomings. Our method builds on sparse Gaussian processes and can model arbitrary spike count distributions (SCDs) with flexible dependence on observed as well as latent covariates, using scalable variational inference to jointly infer the covariate-to-SCD mappings and latent trajectories in a data efficient way. Without requiring repeatable trials, it can flexibly capture covariate-dependent joint SCDs, and provide interpretable latent causes underlying the statistical dependencies between neurons. We apply the model to recordings from a canonical non-sensory neural population: head direction cells in the mouse. We find that variability in these cells defies a simple parametric relationship with mean spike count as assumed in standard models, its modulation by external covariates can be comparably strong to that of the mean firing rate, and slow low-dimensional latent factors explain away neural correlations. Our approach paves the way to understanding the mechanisms and computations underlying neural variability under naturalistic conditions, beyond the realm of sensory coding with repeatable stimuli.

1 Introduction

Classical analyses of neural coding are based on mean spike counts or neural firing rates. Indeed, some of the most paradigmatic examples of the neural code were discovered by regressing neural firing rates to particular sensory stimuli [1, 2] or behavioural covariates [3, 4, 5, 6] to characterize their tuning properties. However, neural spiking is generally not regular. Recordings from many cortical areas show significantly different activity patterns within and across identical trials [7], despite fixing experimentally controlled variables. This irregularity is also seen in continual neural recordings without trial structure [8]. The resulting variability has classically been characterised as 'Poisson', with a Fano factor (variance to mean ratio) of one [9], but experimental data also often exhibits significantly more [10, 8, 11, 12] and sometimes less [13, 14] variability, respectively referred to as over- or underdispersion. Moreover, experimental studies have revealed that neural variability generally depends on stimulus input and behaviour [15, 16, 17, 18], and exhibits structured shared

Preprint. Under review.

variability ('noise correlations') across neurons even after conditioning on such covariates. Such correlations can have important consequences for decoding information from neural population activity [19, 20, 21] and reveal key properties of the underlying circuit dynamics [22]. Moreover, theories of neural representations of uncertainty have assigned computational significance to variability as a signature of Bayesian inference [23, 24, 25, 26]. Thus, just as classical tuning curves for firing rates have been crucial for understanding some of the fundamental properties of the neural code, a principled statistical characterisation of neural variability, and its dependence on stimulus and behavioral covariates, is a key step towards understanding the dynamics of neural circuits and the computations they subserve.

The traditional approach to characterising neural variability has been pioneered in sensory areas, and relies on repeatable trial structure with a sufficiently large number of trials using identical stimulus and behavioral correlates [27, 15, 28]. Variability in this case can be quantified by simple summary statistics of spike counts across trials of the same condition. However, this approach does not readily generalise to more naturalistic conditions where covariates cannot be precisely controlled and repeated in an experiment. This more general setting requires statistical methods that take into account temporal variation of covariates for predicting neural count activity. Generalised Linear Models are a popular choice [29], but they only model the dependence of firing rates on covariates – with changes in variability directly coupled to changes in the rate inherent to Poisson spiking. More complex methods for inferring neural tuning [30, 31] and latent structure [32, 33, 34] similarly use restrictive parametric families for spike count distributions, and thus also cannot model changes in variability that are not 'just' a consequence of changes in mean counts or firing rates. Conversely, statistical models capable of capturing arbitrary single neuron count statistics, such as histogram-based approaches or copulas [35], do not incorporate dependencies on covariates.

Here we unify these separate approaches, resulting in a single framework for jointly inferring neural tuning, single neuron count statistics, neural correlations, and latent structure. Our semi-parametric approach leads to a universal count model for counts ranging from 0 to K, in the sense that we can model arbitrary distributions over the joint count space of size $(K+1)^N$ of N neurons. The trade-off between computational overhead and model expressivity is controlled by hyperparameters, with expressivity upper bounded by the true universal model. Our approach extends the idea of a universal binary count model [36] to a finite range of integer counts, while allowing flexible dependence on observed and latent covariates to model non-stationary neural activity and correlations. The flexibility reduces biases from restrictive assumptions in any of the model components. Scalability is maintained by leveraging sparse Gaussian processes [37] with mini-batching [38, 39] to handle the size of modern neural recordings.

We first introduce the universal count model, and then describe how to interpret as well as evaluate model fits. As our model is able to capture arbitrary single neuron statistics, we build on the Kolmogorov-Smirnov test to construct more absolute goodness-of-fit measures. After validating our method on synthetic data that cannot be captured by currently used methods, we apply the model to electrophysiological recordings from two distinct brain regions in mice that show significant tuning to the head direction of the animal [40, 41]. We find that (1) neural activity tends to be more regular than common Poisson-like models at higher firing rates, and more irregular at low rates; (2) mean and variance of counts defy a simple parametric relationship imposed by parametric count distribution families; (3) variability modulation by behaviour can be comparable or even exceed that of the mean count or firing rate; (4) a two-dimensional latent trajectory varying on timescales of ~ 1 s is sufficient to explain away neural correlations but not the non-Poisson nature of single neuron variability. Finally, we discuss related work, limitations and proposed extensions of our model.

2 Universal count model

Notation Spike count activity of a population of N neurons recorded into T time bins is formally represented as an N-dimensional time series of non-negative integers. Due to biological constraints, the possible spike counts have some finite upper bound K, taken as the highest observed count. We denote probabilities of a spike count distribution (SCD) by a vector $\boldsymbol{\pi}$ of length $K+1$. Additionally, we denote the count activity over neurons n and time steps t by a matrix $Y \in [0, K]^{T \times N}$ with elements y_{tn}. The observed $X \in \mathbb{R}^{T \times D_x}$ and latent $Z \in \mathbb{R}^{T \times D_z}$ covariates (range depends on topology [42]) similarly consist of elements x_{td} and z_{tq}, respectively. Generally, we use capital versions of quantities to denote multidimensional concatenations.

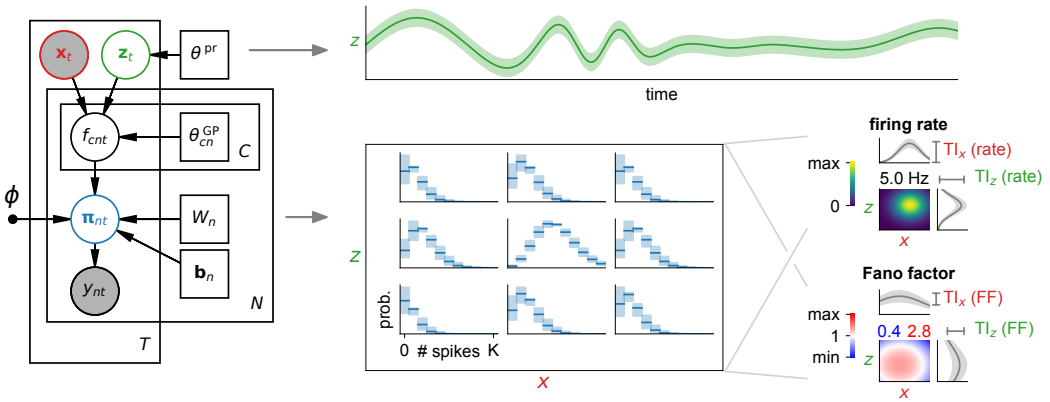

Figure 1: **Schematic of the universal spike count model and the workflow.** Left: graphical model corresponding to Equation 1, with shaded circles as observed, open circles as latent, and squares as deterministic variables. Filled dots as at ϕ indicate fixed quantities. Middle: example inference of model posterior Equation 3, with inferred latent trajectories (green, top) and covariate-dependent SCDs (blue, bottom) that depend on both observed x and latent z covariates. Note we only show the posterior over a single SCD evaluated on a (x,z) grid, whereas the full posterior defines SCDs over all neurons. Right: obtaining interpretable spike count statistics from the SCDs (see subsection 2.3). Examples show firing rate and Fano factor tuning curves over observed x and latent z covariates, either jointly (heatmaps) or marginalized (grey curves). The depth of modulation in marginalized tuning curves is used to extract a tuning index (TI) for the chosen subsets of covariates, see Equation 6.

2.1 Generative model

The big picture is to model counts Y with dependence on X. For each neuron, our model consists of C Gaussian process (GP) priors, a basis expansion $\boldsymbol{\phi} : \mathbb{R}^C \to \mathbb{R}^{\tilde{C}}$, and a linear-softmax mapping

$$\begin{aligned} z_{tq} &\sim p(Z; \theta^{\text{pr}}), \quad h_{cn}(\cdot) \sim \mathcal{GP}(0, k_{cn}(\cdot,\cdot;\theta^{\text{GP}}_{cn})) \\ f_{cnt} &= h_{cn}(\boldsymbol{x}_t, \boldsymbol{z}_t) \\ \boldsymbol{\pi}_{nt} &= \text{softmax}(W_n\, \boldsymbol{\phi}(\boldsymbol{f}_{nt}) + \boldsymbol{b}_n) \\ y_{nt} &\sim \text{Discrete}(\boldsymbol{\pi}_{nt}) \end{aligned} \tag{1}$$

where k_{cn} is the GP covariance function with kernel hyperparameters θ^{GP}_{cn}. The use of non-parametric GP mappings with point estimate parameters W and $\boldsymbol{b}$ leads to a semi-parametric model with parameters θ, see details in Appendix E. The overall generative model $P_\theta(Y|X)$ is depicted schematically in Figure 1. Note the model specifies a prior $p(\Pi|X)$ over joint SCDs, conceptually similar to Dirichlet priors [36] but allowing non-parametric dependence on X. With latent input Z, our model can flexibly describe multivariate dependencies in joint SCDs as conditional independence across neurons no longer holds when marginalizing over Z. In addition, $p(Z)$ models temporal correlations in the latent states. We use Markovian priors (details in Appendix E)

$$p_\theta(Z) = p_\theta(\boldsymbol{z}_1) \prod_{t=2}^{T} p_\theta(\boldsymbol{z}_t|\boldsymbol{z}_{t-1}) \tag{2}$$

denoting θ^{pr} with generative model parameters θ for compactness. This allows the model to flexibly capture both neural and temporal correlations in Y. To attain scalability, we use sparse GPs [37].

Depending on C and basis functions $\boldsymbol{\phi}(\cdot)$, we obtain an approximation to the true universal prior on joint SCDs, with 'universal' referring to the ability to capture any joint SCD over all neurons. Arbitrary single neuron statistics can be captured when $C = K$ with $\boldsymbol{\phi}(\boldsymbol{f}) = \boldsymbol{f}$, but is computationally expensive when $N \times C \gg 1$. For capturing all correlations, the model also requires a sufficiently large latent space. One controls the trade-off between model expressiveness and computational overhead through C and $\boldsymbol{\phi}$. Larger expansions $\boldsymbol{\phi}$ allow one to model count distributions more expressively with small C, e.g. the linear-exponential $\boldsymbol{\phi}(\boldsymbol{f}) = (f_1, e^{f_1}, f_2, e^{f_2} \ldots)$ covers a range of distributions including the truncated Poisson with only $C = 1$ (see subsection B.3).

2.2 Stochastic variational inference and learning

For the joint model distribution $p_\theta(Y, \Pi, Z|X) = P(Y|\Pi)\, p_\theta(\Pi|X, Z)\, p_\theta(Z)$, with count distributions $P(Y|\Pi)$, we approximate the posterior by $q_{\theta,\chi,\varphi}(\Pi, Z|X)$ of the form

$$q_{\theta,\chi}(\Pi|X, Z)\, q_\varphi(Z) = \left(\prod_n^N q_{\theta,\chi}(\Pi_n|X, Z)\right) \left(\prod_t^T q_\varphi(\boldsymbol{z}_t)\right) \tag{3}$$

with φ and χ the variational parameters for latent states and the sparse Gaussian process posterior (Appendix E), respectively. Note that we use a factorized normal $q(Z)$ for Euclidean Z, and a wrapped normal for circular Z based on the framework of reparameterized Lie groups [43, 42]. The posterior over count probabilities $q_{\theta,\chi}(\Pi|X, Z)$ is defined as mapping the sparse Gaussian process posterior $q_{\theta,\chi}(F|X, Z)$ through $\Pi(F)$ (Equation 1), a deterministic many-to-one mapping. This is analytically intractable, so in practice it is represented by Monte Carlo samples. An upper bound on the negative log marginal likelihood can be minimized using stochastic variational inference [44]

$$\mathcal{F}_{\theta,\chi,\varphi} = -\mathbb{E}_{Z\sim q_\varphi(Z)}\mathbb{E}_{\Pi\sim q_\theta(\Pi|X_\mathcal{D},Z)}\left[\log \frac{P(Y_\mathcal{D}|\Pi)\, p_\theta(\Pi|X_\mathcal{D}, Z)\, p_\theta(Z)}{q_\theta(\Pi|X_\mathcal{D}, Z) q_\varphi(Z)}\right] \tag{4}$$

with $\mathcal{D}$ denoting observed data. This objective leads to tractable terms (subsection E.1), allowing us to infer the approximate posterior as well as a lower bound of the log marginal likelihood [45, 39]. We use Adam [46] for optimization, see details of implementation and model fitting in Appendix E.

2.3 Obtaining interpretable spike count statistics from the model

Characterizing spike count distributions From the posterior $q(\Pi|X)$[1], we can compute samples of the posterior of any statistic of spike counts as a function of covariates. Single neuron statistics in particular can be characterized by tuning curves for both mean firing rates and Fano factors (FF)

$$\rho(X) = \frac{1}{\Delta}\mathbb{E}_{q(\Pi|X)}\mathbb{E}_{P(Y|\Pi)}[Y] \qquad \text{FF}(X) = \mathbb{E}_{q(\Pi|X)}\left[\frac{\text{Var}_{P(Y|\Pi)}[Y]}{\mathbb{E}_{P(Y|\Pi)}[Y]}\right] \tag{5}$$

with time bin length Δ. The model also quantifies private neuron variability that cannot be explained away by regressing to shared input (both observed and latent) through $P_n(y_{tn}|\boldsymbol{x}_t)$.

To quantify the sensitivity of a some aspect of neuron activity to a set of covariates $\boldsymbol{x}_*$, we define a tuning index (TI) with respect to a count statistic $T_y(\boldsymbol{x}_*)$

$$\text{TI} = \frac{\max_{\boldsymbol{x}_*} T_y(\boldsymbol{x}_*) - \min_{\boldsymbol{x}_*} T_y(\boldsymbol{x}_*)}{\max_{\boldsymbol{x}_*} T_y(\boldsymbol{x}_*) + \min_{\boldsymbol{x}_*} T_y(\boldsymbol{x}_*)} \tag{6}$$

that is evaluated under the mean posterior count distribution marginalized over all other covariate dimensions complementary to $\boldsymbol{x}_*$. These marginalized distributions are estimated using the input time series, see Appendix F. Resulting marginalized tuning curves used for TIs are depicted in Figure 1.

Generalized Z-scores and noise correlations The deviation of activity from the predicted statistics is commonly quantified through Z-scores [8, 47, 17], which are computed as $(y - \langle y\rangle)/\sqrt{\langle y\rangle}$ with $\langle y\rangle$ being the mean count in the time bin. If neural activity followed a Poisson distribution, the distribution of Z asymptotically tends to a unit normal when $N \gg 1$ (Appendix C). We generalize the Z-score using the probability integral transform and the inverse normal CDF $\Phi^{-1}(q)$

$$Z = \Phi^{-1}(q) \quad \text{with} \quad q(y) = \int_0^{y+\epsilon} p(\tilde{y})\, \mathrm{d}\tilde{y} = \sum_{k=0}^{y-1} P(k) + \epsilon P(y), \quad \epsilon \sim \mathcal{U}(0, 1) \tag{7}$$

which removes the bias away from Gaussianity at low counts and also generalizes to arbitrary count distributions. The dequantization noise ϵ leads to continuous q and Z.

With the Z-score, one can completely describe single neuron statistics with respect to the model. Correlations in the neural activity however will cause Z-scores to be correlated. We define generalized lagged correlations $r_{ij}(\Delta) \in [-1, 1]$ and Fisher $Z \in \mathbb{R}$ that is more convenient for statistical testing

$$r_{ij}(\Delta) = \langle Z_i(t)\, Z_j(t + \Delta)\rangle_t, \quad Z_{Fisher} = \frac{1}{2}\log\frac{1 + r}{1 - r} \tag{8}$$

which describes spatio-temporal correlations not captured by the model. Noise correlations [48] refer to the case of $\Delta = 0$, when r_{ij} becomes symmetric.

[1] For notational convenience, X denotes both observed and latent covariates here.

2.4 Assessing model fit

Our model depends on a hyperparameter $C \leq K$ that trades off flexibility with computational burden. In practice, one likely captures the neural activity accurately with C well below K and a simple basis expansion as the linear-exponential above or quadratic-exponential $\boldsymbol{\phi}(\boldsymbol{f}) = (f_1, f_1^2, e^{f_1}, \ldots, f_1 f_2, \ldots)$. This can be quantified by the statistical measures provided below, and allows us to select appropriate hyperparameters to capture the data sufficiently well.

To assess the model fit to neural spike count data, a conventional machine learning approach is to evaluate the expected log likelihood of the posterior predictive distribution on held-out data Y, leading to the cross-validated log-likelihood

$$\text{cvLL} = \mathbb{E}_{q(Z)} \mathbb{E}_{q(\Pi|X,Z)} [\log P(Y|\Pi)] \tag{9}$$

where we cross-validate over the neuron dimension by using the majority of neurons to infer the latent states $q(Z)$ in the held-out segment of the data, and then evaluate Equation 9 over the remaining neurons. Without latent variables, we simply take the expectation with respect to $q(\Pi|X)$. However, the cvLL does not reveal how well the data is described by the model in an absolute sense. Likelihood bootstrap methods are possible [28], but become cumbersome for large datasets. To assess whether the neural data is statistically distinguishable from the single neuron statistics predicted by the model, we use the Kolmogorov-Smirnov framework [49] along with and q and Z-scores from subsection F.4

$$T_{\text{KS}} = \max_i |F(q_i) - q_i| \tag{10}$$

with empirical distribution function $F(q)$, for details see Appendix C. This scalar number is positive and does not indicate whether the data is less or more regular than predicted by the model. A useful measure of dispersion is the variance of Z, in particular its logarithm

$$T_{\text{DS}} = \log \langle Z^2 \rangle_T + \left(\frac{1}{T} + \frac{1}{3T^2} \right) \tag{11}$$

which provides a real number indicating over- and underdispersion for positive and negative signs, respectively. T refers to the number of time steps or Z-score values. Its sampling distribution under $Z \sim \mathcal{N}(0, 1)$ is asymptotically normal, centered around 0 with a variance depending on T (Appendix D). This extends the notion of over- and underdispersion beyond Poisson reference models [50]. To quantify whether the model has captured noise correlations in the data, we can then compute Z-scores with respect to the mean posterior predictive distribution

$$Q_{\theta,\varphi}(Y|X) = \int \prod_t^T \left(\prod_n^N \mathbb{E}_{q(\boldsymbol{\pi}_{nt}|\boldsymbol{x}_t, \boldsymbol{z}_t)} \left[P(y_{tn}|\boldsymbol{\pi}_{nt}) \right] \right) q_\varphi(\boldsymbol{z}_t) \, \mathrm{d}\boldsymbol{z}_t \tag{12}$$

Correlations that are caused by co-modulation of neurons by low-dimensional factors can be captured with latent states Z inferred from the same data. Intuitively, this can be seen as treating latent states Z as if it was part of observed input or behaviour. Computing Equation 8 should then show a decrease in correlations r, as the Z-scores are whitened under the posterior predictive distribution.

3 Results

In the following results, we use $C = 3$ with an elementwise linear-exponential basis expansion as described in subsection 2.1. This empirically provided sufficient model flexibility to capture both the synthetic and real data as can be seen in goodness-of-fit metrics. We use an RBF kernel with Euclidean and cosine distances for Euclidean and angular input dimensions respectively (Appendix E).

3.1 Synthetic data

Animals maintain an internal estimate of their head direction in particular circuits of their brain [4, 41, 51]. Here, we extend simple statistical models of head direction circuits [52] for validating the ability of the universal model to capture complex count statistics, as well as neural correlations through latent structure. The task is to jointly recover the ground truth count likelihoods, their tuning to covariates, and latent trajectories if relevant from activity generated using two synthetic populations. The first population was generated with a parametric heteroscedastic Conway-Maxwell-Poisson

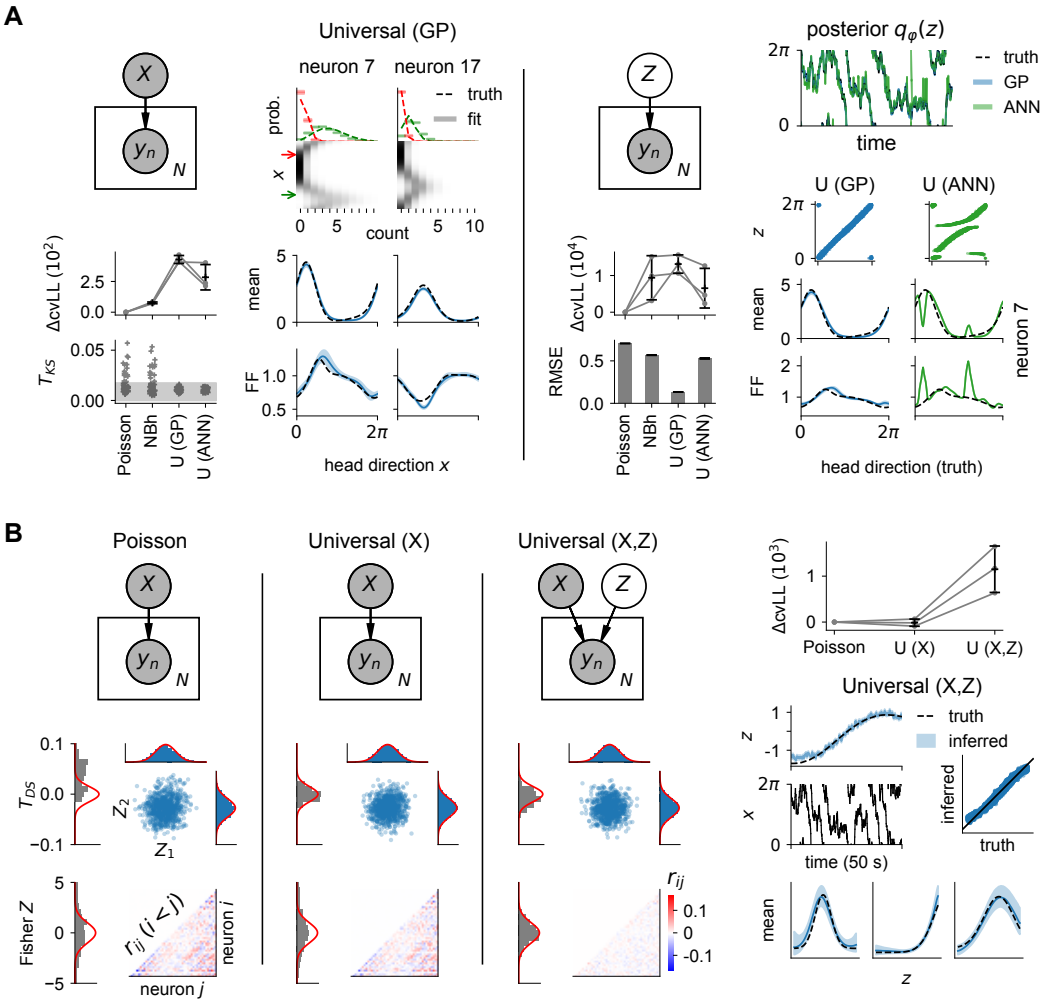

Figure 2: **Model validation with two synthetic head direction cell populations.** **(A)** Regression and latent variable validation experiments with synthetic data from the heteroscedastic Conway-Maxwell-Poisson population. Error bars indicate standard deviation over cross-validation runs. The shaded region for T_{KS} indicates a 95% confidence interval. The root mean squared error (RMSE) of the inferred latent is evaluated with the geodesic distance on the ring Appendix F, with errors bars indicating s.e.m. of RMSE. **(B)** Applying regression and latent-regression models to the modulated Poisson population. We visualize the single neurons fits with Z-scores T_{DS}, and correlations captured with r_{ij} and corresponding Fisher Z values (see subsection 2.4). Variational uncertainties shaded in tuning curves indicate 5th to 95th percentile bounds, while for $q(Z)$ they indicate standard deviations. Note that Z-scores are always computed under the posterior predictive distribution Equation 12, with Z inferred from the same data when a latent space is present.

(CMP) model [53], which has decoupled mean and variance modulation as well as simultaneously over- and underdispersed activity (Fano factors above and below 1). The second population consists of Poisson neurons tuned to head direction and an additional hidden signal, which gives rise to apparent overdispersion [28] as well as noise correlations when only regressing to observed covariates. For mathematical details, see Appendix F (synthetic populations) and Appendix B (count distributions).

We compare our universal model to the log Cox Gaussian process or Poisson GP model [33] and the heteroscedastic negative binomial GP (NBh) model which places GP priors on both the rate and shape parameter, a non-parametric extension of [53]. The more flexible CMPh model, analogous to NBh, has difficulty in scaling to large datasets due to the series approximation of the partition

function (Appendix B). To show the power of GP based approaches, we also compare to a universal model with an artificial neural network (ANN) mapping replacing the GP. For details of the baseline models, see Appendix F. For cross-validation we split the data into 10 roughly equal non-overlapping segments, and validated on 3 chosen segments that were evenly spread out across the data. When a latent space was present, we used 90% of the neurons to infer the latent signal while validating on the remaining neurons, and repeated this for non-overlapping subsets. We rescale the log likelihoods by the ratio of total neurons to neurons in subset and then take the average over all subsets to obtain comparable cross-validation runs to regression.

Figure 2A shows that the universal model successfully captures nontrivial count statistics of the heteroscedastic CMP population. Baseline models cannot capture cases where the Fano factor drops below 1, and indeed are outperformed. In addition, we observe that using a Bayesian GP over an ANN mapping in the model leads to a reduction in overfitting, especially in the latent setting where the ANN model fails to recover the ground truth latent signal. Figure 2B shows that the modulated Poisson population activity is seen by a Poisson regression model as overdispersed, indicated by T_{DS}. Our universal model flexibly captures the overdispersed single neuron statistics, independent from noise correlations r_{ij} that are captured when we introduce a Euclidean latent dimension. As expected, the Z-score scatter plots shows whitening under the posterior predictive distribution when the correlations are captured.

3.2 Mouse head direction cells

We apply our universal model to a recording of 33 head direction cells in the anterior nucleus of the thalamus (ANT) and the postsubiculum (PoS) of freely moving mice [40, 41]. Neural data was binned into 40 ms intervals, see details in Appendix F. Note that observed count statistics differ with bin sizes (Appendix A), which is expected as consecutive bins are not independent. Regression was performed against head direction (HD), angular head velocity (AHV), animal speed and position, and absolute time, which collectively form $X_{\mathcal{D}}$ in this model. We used 64 inducing points for regression, and added 8 for every latent dimension added (Appendix E). Cross-validation was performed similarly to the validation experiments, except that we used subsets with 85% of the neurons to infer Z.

Figure 3A shows that for regression NBh performs worst, likely due to overfitting, despite containing Poisson as a special case (only approximately reachable in practice though, see Appendix B). Only the universal model captures the training data satisfactorily with respect to confidence bounds for T_{KS} and T_{DS}, although the data remained slightly underdispersed to the model with T_{DS} values slightly skewed to negative. Compared to the Poisson model, the cvLL is only slightly higher for the universal model as the data deviates from Poisson statistics in subtle ways. We see both FF above and below 1 (over- and underdispersed) across the neural firing range in Figure 3B, with quite some neurons crossing 1. Correspondingly, FF-mean correlations coefficients are often negative. Their spread away from ± 1 indicates firing rate and FF do not generally satisfy a simple relationship, especially for examples such as cell 27. Furthermore, ANT neurons seem to deviate less from Poisson statistics. From Figure 3C, we note in particular that FFs tend to decrease at the preferred head direction, but rise transiently as the head direction approaches the preferred value. We also see that tuning to speed and time primarily modulates variability rather than firing rates. All of this is impossible to pick up with baseline models, which constrain FF ≥ 1 as well as FF increasing with firing rate (Appendix B). Finally, we see more tuning of the firing rate to position in PoS cells.

When adding latent dimensions, Figure 3D shows a peak in the cvLL at two dimensions, where correspondingly the Fisher Z samples are well described by a unit normal for the first time. Kernel length scales however did not indicate redundant latent subspaces for higher dimensions as expected for automatic relevance determination, possibly due to mixing of latent dimensions. Notice the noise correlation patterns in Figure 3E tend to show positive correlations for similarly tuned neurons roughly around the diagonal of blocks, as expected from ring attractor models [22]. Intrinsic neuron variability, roughly quantified by the average FF, further decreased and thus become even more underdispersed when considering additional tuning to latents, in particular for ANT. In addition, latent signals primarily modulate firing rate as seen from TIs in Figure 3F. When looking at time scales of covariates in Figure 3G (computed as the decay time constant of the autocorrelogram (Appendix F), the latent processes vary on time scales right in the gap of behavioural time scales.

Beside our main contribution of characterising the fine structure of neural variability, our results have another novel element. Using GP-based non-parametric methods, we successfully estimated

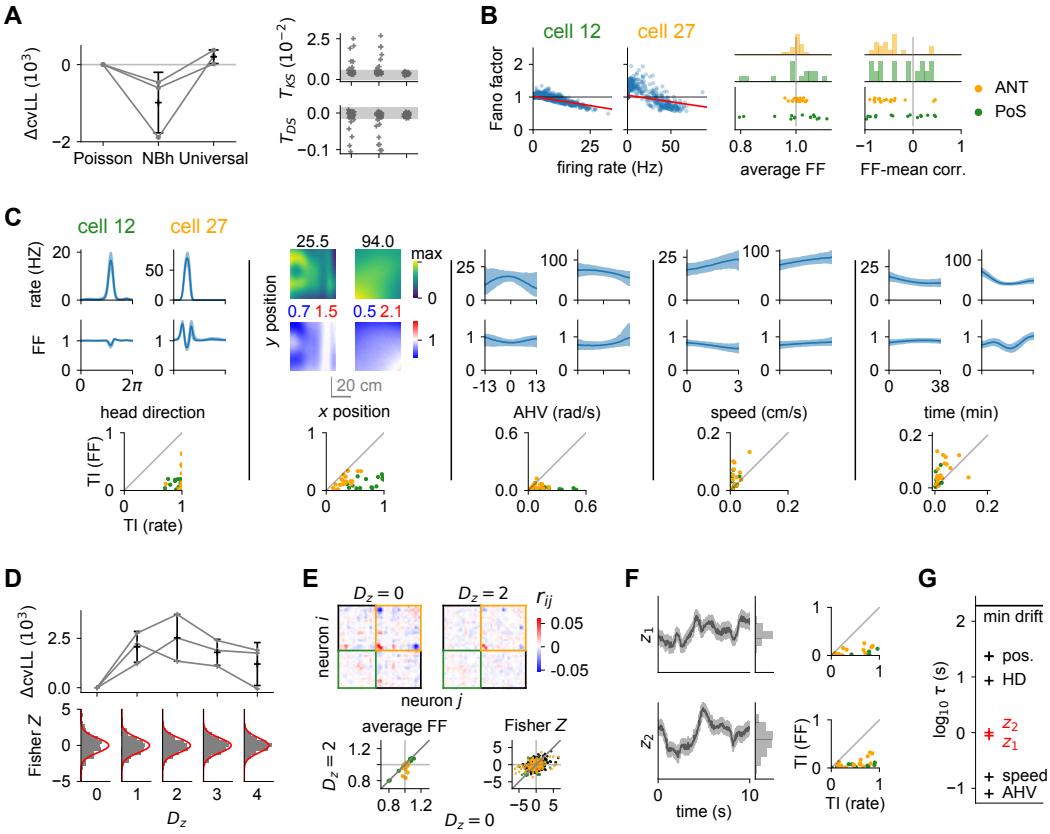

Figure 3: **Application to mouse head direction cells in the anterior nucleus of the thalamus (ANT) and the postsubiculum (PoS). (A)** Goodness-of-fit measures for Poisson, heteroscedastic negative binomial (NBh) and universal regression models. Error bars indicate standard deviations over cross-validation runs. Shaded regions for T_{KS} and T_{DS} indicate a 95% confidence interval. **(B)** Fano factor (FF) versus the mean count of the predictive distribution across time bins (i.e. marginalising over input covariates). We also plot the average of the FFs and the Pearson r correlation between FF and mean count per cell. **(C)** Visualizing conditional tuning curves, obtained by varying the relevant covariates while keeping all others fixed (at preferred HD and centre of the arena, with zero speed, AHV and at time $t = 0$). These are generally different from marginal tuning curves used for TIs. Variational uncertainties indicate 5th to 95th percentile bounds. **(D)** Adding latent dimensions to the universal regression model. **(E)** Comparison between the universal model without ($D_z = 0$) and with $Dz = 2$ latent covariates. Neurons in the noise correlation (diagonal elements of r_{ij} are always 1 and not included) plots are ordered by area first (PoS and ANT), and then by preferred head direction within area. Note that average FF is the same as in (B) but with sampled latents from $q(Z)$ as part of the input, as if they were observed. **(F)** Inferred latent timeseries for the 2D latent space with corresponding TIs. Variational uncertainties indicate standard deviations. **(G)** Time scales for covariates computed from their auto-correlations. For reference, horizontal line shows the estimated lower bound for the time scale of representational drift, computed as the minimum kernel length scale over absolute time across all neurons.

the tuning of cells to as many as 8 different covariates (6 observed + 2 latent) in a statistically sound fashion (even previous GP-based approaches considered a maximum of 4 covariates). Specifically, one of our covariates was absolute experimental time to capture non-stationarities in neural tuning. As a result, our model captured several experimental phenomena that are studied separately in the literature: drifting neural representations [54, 55, 56], anticipatory time intervals [52] and conjunctive tuning to behaviour [57]. We also applied the model in a purely latent setting similar to the example in Figure 2A, with the universal model uncovering a latent signal more closely correlated to the head direction compared to baseline models. These additional results are presented in Appendix A.

4 Discussion

Related work Neural encoding model provide a statistical description of neural count activity, and typically rely on a parametric count likelihood such as the Poisson [33], negative binomial [30, 31] or Conway-Maxwell-Poisson distribution [53]. This choice is independent of the empirical count statistics and is often mismatched to the data. Heteroscedastic count models, characterized by input-dependent noise [58], additionally regress the dispersion parameter of count distributions to covariates [59, 60]. This has shown improvements in stimulus decoding and more calibrated posterior uncertainties [53]. Copula-based models [35, 61] separate marginal distributions of single neurons from the multivariate dependency structure of the population parameterized by the copula family, and thus do not place parametric constraints on single neuron count statistics. The idea of a universal model that can capture arbitrary joint distributions has been explored for binary spike trains [36], using Bayesian non-parametric models to provide regularization and flexibility [36, 62]. However, neither approach naturally incorporates modulation of spike count distributions by input covariates.

Our model deals with discrete spike counts ranging from 0 to K in a manner similar to categorical output variables often considered in machine learning. Similar models have been proposed mainly in the context of Gaussian process classification [63, 64], which directly pass Gaussian process function points through a softmax nonlinearity. Our approach instead passes separate Gaussian processes through a linear-softmax mapping to compute count probabilities. Introducing unobserved input variables in a Gaussian process model leads to Gaussian process latent variable models [65, 66]. Such models have recently been applied to neural data to performs dimensionality reduction [33], with extensions to non-Euclidean latent spaces and non-reversible temporal priors [42, 67].

Limitations and further work The empirical choice of hyperparameters C and basis functions ϕ is based on achieving sufficient model flexibility, as confirmed with the Kolmogorov-Smirnov approach. Recently, a multivariate extension of the Kolmogorov-Smirnov test has been proposed to directly test multivariate samples against the model [68], instead of looking at single neuron statistics. Alternatively, one could perform ARD [69, 70] by placing a Gaussian prior on W, allowing automatic selection of relevant dimensions once a basis expansion is chosen. Another avenue for future work could consider going completely non-parametric and adding a count dimension to the input space, which is evaluated at counts 0 to K for every time bin. This however increases the number of evaluation points by a factor $K + 1$. In addition, extending our model with more powerful priors for latent covariates, such as Gaussian process priors [33, 67], can improve latent variable analysis, especially at smaller time bins where the temporal prior influence becomes more important. Regularization methods may help to decorrelate inferred trajectories [71, 72].

Conclusion and impact We introduced a universal probabilistic encoding model for neural spike count data. Our model flexibly capture both single neuron count statistics and their modulation by covariates. By adding latent variables, one can additionally capture neural correlations with potentially interpretable unobserved signals underlying the neural activity. We applied our model to mouse head direction cells and found count statistics that cannot be captured with current methods. Neural activity tends to be less variable at higher firing rates, with many cells showing both over- and underdispersion. Fano factors and mean counts generally do not show a simple relation and can even be decoupled, with Fano factor modulation comparable or in some cases even exceeding that of the rate. Finally, we found that a 2D latent trajectory with a timescale of around a second explained away noise correlations in these cells.

Neural variability is usually not considered on the same footing as mean firing rates, with models assigning most computational relevance to rates [73, 74]. However, recent work on V1 has started to explore variability as playing a computationally well-defined useful role in the representation of uncertainty [24, 25, 22, 26]. The framework introduced in this paper provides a principled tool for empirically characterising neural variability and its modulations – without the biases inherent in traditional approaches, which would likely miss potentially meaningful patterns in neural activities beyond mean rates. Our model has the potential to reveal new aspects of neural coding, and may find practical applications in designing and improving algorithms for brain-machine interfaces. As progress is made in scaling and applying such technology beyond research environments [75], it becomes increasingly more important to maintain transparency, e.g. through open source code, and to raise awareness of potential ethical issues [76].

Acknowledgments and Disclosure of Funding

This work was supported by the Cambridge European and Wolfson College Scholarship by the Cambridge Trust (D.L.) and by the Wellcome Trust (Investigator Award in Science 212262/Z/18/Z to M.L). We are grateful to K.T. Jensen and A. Melkonyan for helpful feedback on the manuscript.

References

[1] David H Hubel and Torsten N Wiesel. Receptive fields of single neurones in the cat's striate cortex. *The Journal of physiology*, 148(3):574, 1959.

[2] Frédéric E Theunissen, Kamal Sen, and Allison J Doupe. Spectral-temporal receptive fields of nonlinear auditory neurons obtained using natural sounds. *Journal of Neuroscience*, 20(6):2315–2331, 2000.

[3] John O'Keefe and Jonathan Dostrovsky. The hippocampus as a spatial map: preliminary evidence from unit activity in the freely-moving rat. *Brain research*, 1971.

[4] Jeffrey S Taube, Robert U Muller, and James B Ranck. Head-direction cells recorded from the postsubiculum in freely moving rats. i. description and quantitative analysis. *Journal of Neuroscience*, 10(2):420–435, 1990.

[5] Torkel Hafting, Marianne Fyhn, Sturla Molden, May-Britt Moser, and Edvard I Moser. Microstructure of a spatial map in the entorhinal cortex. *Nature*, 436(7052):801–806, 2005.

[6] Colin Lever, Stephen Burton, Ali Jeewajee, John O'Keefe, and Neil Burgess. Boundary vector cells in the subiculum of the hippocampal formation. *Journal of Neuroscience*, 29(31):9771–9777, 2009.

[7] Michael N Shadlen and William T Newsome. The variable discharge of cortical neurons: implications for connectivity, computation, and information coding. *Journal of neuroscience*, 18(10):3870–3896, 1998.

[8] André A Fenton and Robert U Muller. Place cell discharge is extremely variable during individual passes of the rat through the firing field. *Proceedings of the National Academy of Sciences*, 95(6):3182–3187, 1998.

[9] Alexandre Pouget, Peter Dayan, and Richard Zemel. Information processing with population codes. *Nature Reviews Neuroscience*, 1(2):125–132, 2000.

[10] George J Tomko and Donald R Crapper. Neuronal variability: non-stationary responses to identical visual stimuli. *Brain research*, 79(3):405–418, 1974.

[11] A Aldo Faisal, Luc PJ Selen, and Daniel M Wolpert. Noise in the nervous system. *Nature reviews neuroscience*, 9(4):292–303, 2008.

[12] Johannes Nagele, Andreas VM Herz, and Martin B Stemmler. Untethered firing fields and intermittent silences: Why grid-cell discharge is so variable. *Hippocampus*, 2020.

[13] Justin Keat, Pamela Reinagel, R Clay Reid, and Markus Meister. Predicting every spike: a model for the responses of visual neurons. *Neuron*, 30(3):803–817, 2001.

[14] Gaby Maimon and John A Assad. Beyond poisson: increased spike-time regularity across primate parietal cortex. *Neuron*, 62(3):426–440, 2009.

[15] Mark M Churchland, M Yu Byron, John P Cunningham, Leo P Sugrue, Marlene R Cohen, Greg S Corrado, William T Newsome, Andrew M Clark, Paymon Hosseini, Benjamin B Scott, et al. Stimulus onset quenches neural variability: a widespread cortical phenomenon. *Nature neuroscience*, 13(3):369, 2010.

[16] Adrián Ponce-Alvarez, Alexander Thiele, Thomas D Albright, Gene R Stoner, and Gustavo Deco. Stimulus-dependent variability and noise correlations in cortical mt neurons. *Proceedings of the National Academy of Sciences*, 110(32):13162–13167, 2013.

[17] Alexander S Ecker, Philipp Berens, R James Cotton, Manivannan Subramaniyan, George H Denfield, Cathryn R Cadwell, Stelios M Smirnakis, Matthias Bethge, and Andreas S Tolias. State dependence of noise correlations in macaque primary visual cortex. *Neuron*, 82(1):235–248, 2014.

[18] Neil C Rabinowitz, Robbe L Goris, Marlene Cohen, and Eero P Simoncelli. Attention stabilizes the shared gain of v4 populations. *Elife*, 4:e08998, 2015.

[19] Larry F Abbott and Peter Dayan. The effect of correlated variability on the accuracy of a population code. *Neural computation*, 11(1):91–101, 1999.

[20] Bruno B Averbeck, Peter E Latham, and Alexandre Pouget. Neural correlations, population coding and computation. *Nature reviews neuroscience*, 7(5):358–366, 2006.

[21] Rubén Moreno-Bote, Jeffrey Beck, Ingmar Kanitscheider, Xaq Pitkow, Peter Latham, and Alexandre Pouget. Information-limiting correlations. *Nature neuroscience*, 17(10):1410, 2014.

[22] Guillaume Hennequin, Yashar Ahmadian, Daniel B Rubin, Máté Lengyel, and Kenneth D Miller. The dynamical regime of sensory cortex: stable dynamics around a single stimulus-tuned attractor account for patterns of noise variability. *Neuron*, 98(4):846–860, 2018.

[23] Wei Ji Ma, Jeffrey M Beck, Peter E Latham, and Alexandre Pouget. Bayesian inference with probabilistic population codes. *Nature neuroscience*, 9(11):1432–1438, 2006.

[24] József Fiser, Pietro Berkes, Gergő Orbán, and Máté Lengyel. Statistically optimal perception and learning: from behavior to neural representations. *Trends in cognitive sciences*, 14(3):119–130, 2010.

[25] Gergő Orbán, Pietro Berkes, József Fiser, and Máté Lengyel. Neural variability and sampling-based probabilistic representations in the visual cortex. *Neuron*, 92(2):530–543, 2016.

[26] Rodrigo Echeveste, Laurence Aitchison, Guillaume Hennequin, and Máté Lengyel. Cortical-like dynamics in recurrent circuits optimized for sampling-based probabilistic inference. *Nature Neuroscience*, 23(9):1138–1149, 2020.

[27] David J Tolhurst, J Anthony Movshon, and Andrew F Dean. The statistical reliability of signals in single neurons in cat and monkey visual cortex. *Vision research*, 23(8):775–785, 1983.

[28] Robbe LT Goris, J Anthony Movshon, and Eero P Simoncelli. Partitioning neuronal variability. *Nature neuroscience*, 17(6):858, 2014.

[29] Jonathan W Pillow, Jonathon Shlens, Liam Paninski, Alexander Sher, Alan M Litke, EJ Chichilnisky, and Eero P Simoncelli. Spatio-temporal correlations and visual signalling in a complete neuronal population. *Nature*, 454(7207):995–999, 2008.

[30] Jonathan Pillow and James Scott. Fully bayesian inference for neural models with negative-binomial spiking. *Advances in neural information processing systems*, 25:1898–1906, 2012.

[31] Yuanjun Gao, Lars Busing, Krishna V Shenoy, and John P Cunningham. High-dimensional neural spike train analysis with generalized count linear dynamical systems. In *Advances in neural information processing systems*, pages 2044–2052, 2015.

[32] Byron M Yu, John P Cunningham, Gopal Santhanam, Stephen Ryu, Krishna V Shenoy, and Maneesh Sahani. Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. *Advances in neural information processing systems*, 21:1881–1888, 2008.

[33] Anqi Wu, Nicholas A Roy, Stephen Keeley, and Jonathan W Pillow. Gaussian process based nonlinear latent structure discovery in multivariate spike train data. In *Advances in neural information processing systems*, pages 3496–3505, 2017.

[34] Ryan J Low, Sam Lewallen, Dmitriy Aronov, Rhino Nevers, and David W Tank. Probing variability in a cognitive map using manifold inference from neural dynamics. *bioRxiv*, page 418939, 2018.

[35] Pietro Berkes, Frank Wood, and Jonathan Pillow. Characterizing neural dependencies with copula models. *Advances in neural information processing systems*, 21:129–136, 2008.

[36] Il Memming Park, Evan W Archer, Kenneth Latimer, and Jonathan W Pillow. Universal models for binary spike patterns using centered dirichlet processes. In *Advances in neural information processing systems*, pages 2463–2471, 2013.

[37] Michalis Titsias. Variational learning of inducing variables in sparse gaussian processes. In *Artificial Intelligence and Statistics*, pages 567–574, 2009.

[38] James Hensman, Nicolo Fusi, and Neil D Lawrence. Gaussian processes for big data. *arXiv preprint arXiv:1309.6835*, 2013.

[39] James Hensman, Alexander Matthews, and Zoubin Ghahramani. Scalable variational gaussian process classification. In *Artificial Intelligence and Statistics*, pages 351–360. PMLR, 2015.

[40] Adrien Peyrache and György Buzsáki. Extracellular recordings from multi-site silicon probes in the anterior thalamus and subicular formation of freely moving mice. *CRCNS*, 2015.

[41] Adrien Peyrache, Marie M Lacroix, Peter C Petersen, and György Buzsáki. Internally organized mechanisms of the head direction sense. *Nature neuroscience*, 18(4):569–575, 2015.

[42] Kristopher Jensen, Ta-Chu Kao, Marco Tripodi, and Guillaume Hennequin. Manifold gplvms for discovering non-euclidean latent structure in neural data. *Advances in Neural Information Processing Systems*, 33, 2020.

[43] Luca Falorsi, Pim de Haan, Tim R Davidson, and Patrick Forré. Reparameterizing distributions on lie groups. *arXiv preprint arXiv:1903.02958*, 2019.

[44] Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347, 2013.

[45] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations*, 2014.

[46] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[47] Adam Kohn and Matthew A Smith. Stimulus dependence of neuronal correlation in primary visual cortex of the macaque. *Journal of Neuroscience*, 25(14):3661–3673, 2005.

[48] I-Chun Lin, Michael Okun, Matteo Carandini, and Kenneth D Harris. The nature of shared cortical variability. *Neuron*, 87(3):644–656, 2015.

[49] Jonathan W Pillow. Time-rescaling methods for the estimation and assessment of non-poisson neural encoding models. In *Advances in neural information processing systems*, pages 1473–1481, 2009.

[50] Adam S Charles, Mijung Park, J Patrick Weller, Gregory D Horwitz, and Jonathan W Pillow. Dethroning the fano factor: a flexible, model-based approach to partitioning neural variability. *Neural computation*, 30(4):1012–1045, 2018.

[51] Rishidev Chaudhuri, Berk Gerçek, Biraj Pandey, Adrien Peyrache, and Ila Fiete. The intrinsic attractor manifold and population dynamics of a canonical cognitive circuit across waking and sleep. *Nature neuroscience*, 22(9):1512–1520, 2019.

[52] Johannes Zirkelbach, Martin Stemmler, and Andreas VM Herz. Anticipatory neural activity improves the decoding accuracy for dynamic head-direction signals. *Journal of Neuroscience*, 39(15):2847–2859, 2019.

[53] Abed Ghanbari, Christopher M Lee, Heather L Read, and Ian H Stevenson. Modeling stimulus-dependent variability improves decoding of population neural responses. *Journal of Neural Engineering*, 16(6):066018, 2019.

[54] Yaniv Ziv, Laurie D Burns, Eric D Cocker, Elizabeth O Hamel, Kunal K Ghosh, Lacey J Kitch, Abbas El Gamal, and Mark J Schnitzer. Long-term dynamics of ca1 hippocampal place codes. *Nature neuroscience*, 16(3):264, 2013.

[55] Michael E Rule, Adrianna R Loback, Dhruva V Raman, Laura N Driscoll, Christopher D Harvey, and Timothy O'Leary. Stable task information from an unstable neural population. *Elife*, 9:e51121, 2020.

[56] Daniel Deitch, Alon Rubin, and Yaniv Ziv. Representational drift in the mouse visual cortex. *bioRxiv*, 2020.

[57] Francesca Sargolini, Marianne Fyhn, Torkel Hafting, Bruce L McNaughton, Menno P Witter, May-Britt Moser, and Edvard I Moser. Conjunctive representation of position, direction, and velocity in entorhinal cortex. *Science*, 312(5774):758–762, 2006.

[58] Miguel Lázaro-Gredilla and Michalis K Titsias. Variational heteroscedastic gaussian process regression. In *ICML*, 2011.

[59] Seth D Guikema and Jeremy P Goffelt. A flexible count data regression model for risk analysis. *Risk Analysis: An International Journal*, 28(1):213–223, 2008.

[60] Kimberly F Sellers and Galit Shmueli. A flexible regression model for count data. *The Annals of Applied Statistics*, pages 943–961, 2010.

[61] Meng Hu, Kelsey L Clark, Xiajing Gong, Behrad Noudoost, Mingyao Li, Tirin Moore, and Hualou Liang. Copula regression analysis of simultaneously recorded frontal eye field and inferotemporal spiking activity during object-based working memory. *Journal of Neuroscience*, 35(23):8745–8757, 2015.

[62] Evan W Archer, Il Memming Park, and Jonathan W Pillow. Bayesian entropy estimation for binary spike train data using parametric prior knowledge. In *Advances in neural information processing systems*, pages 1700–1708, 2013.

[63] Kian Ming A Chai. Variational multinomial logit gaussian process. *The Journal of Machine Learning Research*, 13:1745–1808, 2012.

[64] Yarin Gal, Yutian Chen, and Zoubin Ghahramani. Latent gaussian processes for distribution estimation of multivariate categorical data. In *International Conference on Machine Learning*, pages 645–654. PMLR, 2015.

[65] Neil D Lawrence. Gaussian process latent variable models for visualisation of high dimensional data. In *Nips*, volume 2, page 5. Citeseer, 2003.

[66] Michalis Titsias and Neil D Lawrence. Bayesian gaussian process latent variable model. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 844–851, 2010.

[67] Virginia Rutten, Alberto Bernacchia, Maneesh Sahani, and Guillaume Hennequin. Non-reversible gaussian processes for identifying latent dynamical structure in neural data. *Advances in Neural Information Processing Systems*, 33, 2020.

[68] Michael Naaman. On the tight constant in the multivariate dvoretzky–kiefer–wolfowitz inequality. *Statistics & Probability Letters*, 173:109088, 2021.

[69] Christopher M Bishop. Bayesian pca. *Advances in neural information processing systems*, pages 382–388, 1999.

[70] Andreas C. Damianou, Carl Henrik Ek, Michalis K. Titsias, and Neil D. Lawrence. Manifold relevance determination. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012*. icml.cc / Omnipress, 2012.

[71] Irina Higgins, Loïc Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[72] Ricky TQ Chen, Xuechen Li, Roger Grosse, and David Duvenaud. Isolating sources of disentanglement in vaes. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 2615–2625, 2018.

[73] Guillaume Hennequin, Tim P Vogels, and Wulfram Gerstner. Optimal control of transient dynamics in balanced networks supports generation of complex movements. *Neuron*, 82(6):1394–1406, 2014.

[74] Klaus Wimmer, Duane Q Nykamp, Christos Constantinidis, and Albert Compte. Bump attractor dynamics in prefrontal cortex explains behavioral precision in spatial working memory. *Nature neuroscience*, 17(3):431–439, 2014.

[75] Elon Musk et al. An integrated brain-machine interface platform with thousands of channels. *Journal of medical Internet research*, 21(10):e16194, 2019.

[76] Jens Clausen. Man, machine and in between. *Nature*, 457(7233):1080–1081, 2009.

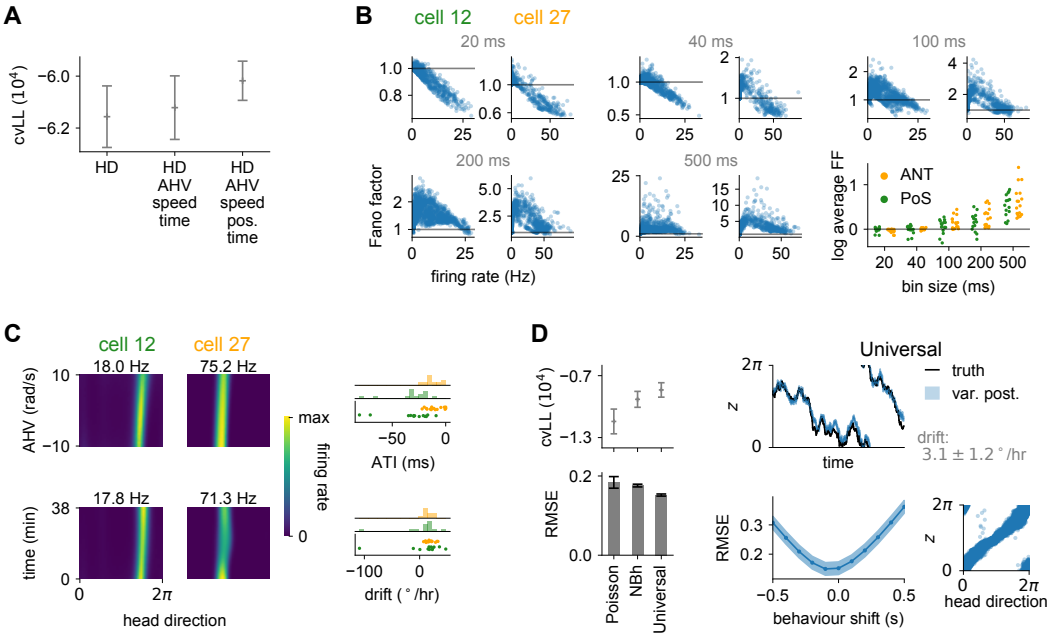

Figure 4: **Additional analysis on head direction data.** **(A)** Model comparison of universal regression models as in Figure 3 but with different regressors. Adding more behaviour improves the cross-validated log likelihood, indicating the model does not overfit when including many behavioural regressors. Error bars here indicate s.e.m. over cross-validation runs. **(B)** Fano factor and mean counts as predicted by the mean posterior count distribution, similar to panel (B) in Figure 3. Note that increasing the bin size leads to higher Fano factors. In addition, notice the consistent decrease of variability at higher firing rates. **(C)** Joint tuning curves of firing rate of AHV-HD and time-HD show two distinct experimental phenomena: anticipatory tuning and neural representational drift, respectively. The ATI and drifts are quantified by the circular-linear regression method as described in subsection A.2, with cells selected based on goodness-of-fit to the circular-linear relation. **(D)** A pure latent model with a ring topology of the latent variable uncovers a signal closely correlated to the animal head direction, up to a constant offset, linear drift term and reflection sign (Equation 57). The universal model outperforms baseline models, and interestingly its RMSE with respect to behavioural head direction is smallest. In fact, comparing to head direction shifted in time with respect to the spike train reveals the latent signal to be most correlated to behaviour ~ -100 ms in the past. Errors bars in RMSE and cvLL are s.e.m. over cross-validation runs.

A Additional analysis of head direction cells

A.1 Temporal bin sizes

In Figure 4B, we can see the sensitivity of the count analysis to bin length. Note when the bin size becomes very small leading to low spike counts, differences in count distributions matter less. In the limit of binary spikes, the count distribution will always be a Bernoulli distribution. In these cases, the latent trajectory dynamics becomes more important as it captures more of the correlations in time that were previously captured by non-Poisson count distributions in larger bins. The different results for different bin lengths is a consequence of consecutive bins being temporally correlated, leading to more extreme fluctuations in activity and thus higher or lower variability.

Spike count based approaches are inherently more coarse grained than methods dealing directly with individual spike times based on point processes [1, 2, 49]. In particular, when studying phenomena at time scales comparable to the interspike intervals, such as theta precession [4, 5], spike count binning may average away such effects if the bin size is too large. However, binning does reduce the number of total time points and may be more practical for studying large population activities recorded over long time periods. Point process models [6] are more suited for describing neural data at shorter time

scales. However, dependencies of consecutive spike intervals are complicated to model, and often models based on Markovian dependencies called renewal processes are used [2, 49]. Recent work with recurrent neural networks [7, 8, 9] and triangular mappings [10] have improved expressivity of such models.

A.2 Drifting and ATIs

Joint tuning curves can reveal neural representations that are not factorized over a set of covariates. The Bayesian nature of Gaussian processes takes care of undersampled regions that are rife in high dimensional input spaces, which is the setting in this work for studying joint tuning to behaviour.

In addition to behaviour, one can pick up representational drift [54, 55] by regressing against absolute time of the recording. The time regressor needs to be considered carefully as it may confound at time scales of latent trajectories. As long as the time scale in the kernel is much larger than the time scale over which the latent variables vary, we can interpret the temporal drift as a separate process from the latent trajectories. Indeed, by initializing at time scales equal to half the total recording time, these time scales of the Gaussian process kernel remain significantly higher than any behavioural time scale (Figure 3G). We find most cells cluster at a drift of ≈ 20 °/hr.

The joint AHV-HD plot reveals anticipatory tuning: when animals turn their head, the head direction tuning curves shift in response to head rotations such that cells expected to spike appear to fire earlier than expected. Theoretical studies have shown that this improves temporal decoding, in the sense that the bias-variance trade-off for decoding downstream can be improved with anticipatory tuning [52]. It appears that the head direction population anticipates the future head direction based on current movement statistics, which allows one to reduce the bias introduced with causal decoding. However, the ATI values in Figure 4C are negative, while in the literature they are postive and differ per region. The neural data description files [40] did mention that the zero time frame of behaviour was randomly misaligned to neural spiking data up to 60 ms. Behaviour may be shifted with respect to the neural spike train, indeed in preliminary analyses with shifted spike trains we found values consistent with literature for ATIs when shifiting ≈ 60 ms [52].

A.3 Latent variable analysis of head direction data

In Figure 4D, we see the inferred angular latent signal is closely correlated to the head direction. The linear drift 3.1 ± 1.2 °/hr is smaller but in the same direction as the drift found in the tuning curves in panel C, which cluster around 20 °/hr. The universal model again shows improvement over baseline models, and in particular the latent signal is more correlated to the behavioural head direction. and tentatively identify a delay in the signal represented compared to measured behaviour. Latent trajectory RMSE was computed with 3-fold cross-validation. We align the latent trajectory Equation 57 to the behaviour in the fitting segment, and compute the geodesic RMSE on the held-out validation segment (Appendix F).

At a bin size of 40 ms, the continuity of the trajectory breaks down more often. This shows up as more spread out off-diagonal points in the scatter plot in panel D of Figure 4. Weak prior, Gaussian process priors are more powerful. However, in this work they were not explored due to scalability issues of sampling functions from Gaussian process posteriors. However, recent work [15] has addressed this issue, and specialized techniques for regularly spaced input such as time points [16, 17] can be applied when using temporal Gaussian process priors for latent trajectories.

B Parametric count distributions

B.1 Poisson distribution

The Poisson count distribution is defined with a mean count λ

$$P_{\text{Poiss}}(n|\lambda) = \frac{\lambda^n}{n!}\, e^{-\lambda}. \tag{13}$$

where $n \in \mathbb{N}_0$. It describes a process where discrete events arriving in a time window are all independent of each other. Mathematically, this is consistent with Equation 13 being the limit of a binomial distribution $P_{\text{Bin}}(n, p)$ with $n \to \infty$ and $p \to 0$, such that $Np = \lambda$.

The Poisson distribution is characterized by the equality of its mean and variance, leading to a Fano factor $\mathrm{V}[n]/\mathbb{E}[n] = 1$. Another convenient property is that the sum of two independent Poisson processes is itself a Poisson process with $\lambda = \lambda_1 + \lambda_2$. This can be shown directly by considering $P(n) = \sum_0^n P(k)\, P(n-k)$ or casting it as a limit of the sum of two Bernoulli processes, and follows intuitively from the fact that spike times are independent of each other. As a consequence, for the inhomogeneous case where we have a time-dependent rate $\lambda(t)$ the count distribution over a longer interval is still Poisson with average $\int_0^T \lambda(t)\,\mathrm{d}t$. Note that this property does generally not hold for non-Poisson distributions, where the count distribution of a sum of counts in separate time windows is not related to the original count distribution in a simple way.

B.2 Non-Poisson count distributions

To account for over- and underdispersed neural activity in real data, i.e. Fano factors above and below 1, other distributions than the Poisson count distribution have been used, and we present common families below.

B.2.1 Zero-inflated Poisson

A common way to introduce overdispersion is to model excess zero counts, which in this context leads to the zero-inflated Poisson (ZIP) process [12]. The count distribution is given by

$$P_{\text{ZIP}}(n|\lambda,\alpha) = \begin{cases} \alpha + (1-\alpha)\, e^{-\lambda} & \text{if } n = 0 \\ (1-\alpha)\, \frac{\lambda^n}{n!}\, e^{-\lambda} & \text{if } n > 0. \end{cases} \tag{14}$$

The parameterization leads to $\mathbb{E}[n] = \lambda(1-\alpha)$ and $\mathrm{V}[n] = \lambda(1-\alpha) + \lambda^2\alpha(1-\alpha)$ using the law of total variance.

B.2.2 Modulated Poisson distributions

One perspective of non-Poisson distributions is that they arise from noise in the rate parameters λ. Such count processes are referred to as modulated Poisson processes. From a probabilistic point of view, the resulting count distribution is a marginalization

$$P(n|\theta) = \int P(n|\lambda,\theta)\, p(\lambda|\theta)\,\mathrm{d}\lambda, \tag{15}$$

with noise parameters θ. A recently proposed flexible spike count model that can give rise to different mean-variance relationships, including decreasing Fano factors at high firing rates similar to what is observed in Figure 4B, builds on this framework [50]. However, the modulated Poisson process can only account for overdispersion with respect to the base Poisson process. Adding noise cannot lead less variability here, and this implies that Fano factors are bounded from below by 1.

B.2.3 Negative binomial

The negative binomial distribution is based on independent Bernoulli trials like the binomial distribution. However, now we count the number of successes before r failures are observed. If we have Bernoulli trials with success probability p, one can obtain the negative binomial distribution with parameterization $p = \frac{\lambda}{r+\lambda}$

$$P_{\text{NB}}(n|\lambda,r) = \frac{\lambda^n}{n!}\frac{\Gamma(r+n)}{\Gamma(r)\,(r+\lambda)^n}\left(1+\frac{\lambda}{r}\right)^{-r}. \tag{16}$$

Note this distribution is a specific instance of a modulated Poisson process (Equation 15), with $\lambda \sim f_{\text{Gamma}}(\lambda; r, \frac{\lambda}{r})$. The parameterization is such that $\mathbb{E}[n] = \lambda$ holds, but $\mathrm{V}[n] = \lambda(1+\frac{\lambda}{r})$ making it overdispersed with respect to a Poisson distribution. In practice, numerical evaluation of the Poisson limit when $r = 0$ is only approximate due to the numerical precision of the relevant function implementations.

B.2.4 Conway-Maxwell-Poisson

A distribution that handles both over- and underdispersed count distributions is the Conway-Maxwell-Poisson distribution [20]

$$P_{\text{CMP}}(n|\lambda,\nu) = \frac{1}{Z(\lambda,\nu)}\frac{\lambda^n}{(n!)^\nu}. \tag{17}$$

The normalization constant has no closed form expression and must be evaluated numerically

$$Z(\lambda,\nu) = \sum_{k=0}^{\infty}\frac{\lambda^k}{(k!)^\nu}. \tag{18}$$

It contains the Bernoulli ($\nu \to \infty$), Poisson ($\nu = 1$) and geometric ($\nu \to 0$) distributions as limiting cases. The notably property is that the CMP distribution provides a smooth transition between these well-known distributions. At integer ν, the The moments of this distribution do not have a closed form expression in general, but can be computed using the partition function through the cumulant generating function $K(t) = \log \mathbb{E}[e^{tn}] = \log Z(\lambda e^t, \nu) - \log Z(\lambda,\nu)$. The expression for the mean and variance follow to be

$$\begin{aligned}\mathbb{E}[n] &= \lambda\frac{\mathrm{d}}{\mathrm{d}\lambda}\log Z(\lambda,\nu)\\ \text{Var}[n] &= \lambda\frac{\mathrm{d}}{\mathrm{d}\lambda}\mathbb{E}[n].\end{aligned} \tag{19}$$

with approximate expressions [20]

$$\begin{aligned}\mathbb{E}[n] &= \lambda^{1/\nu} + \frac{1}{2\nu} - \frac{1}{2}\\ \text{Var}[n] &= \frac{1}{\nu}\lambda^{1/\nu}.\end{aligned} \tag{20}$$

which hold well for $\nu \approx 1$ and $\lambda > 10^\nu$.

B.3 Linear-softmax count distributions

The count distributions used in this work rely on a linear mapping of the input $\boldsymbol{a}$ combined with a softmax

$$P(n|\boldsymbol{a}; W, \boldsymbol{b}) = \text{softmax}(W\boldsymbol{a} + \boldsymbol{b}), \quad \text{softmax}(\boldsymbol{x})_i = \frac{e^{x_i}}{e^{\sum_j x_j}} \tag{21}$$

To illustrate the connection of this softmax count distribution used in Equation 1 to Poisson models, consider the distribution specified by the softmax mapping for $C = 1$ and an element-wise linear-exponential $\boldsymbol{\phi}(\boldsymbol{f}) = (f_1, e^{f_1}, \ldots)$. This choice contains the truncated Poisson distribution with f as the logarithm of the mean count, corresponding to $W_{j0} = j$, $W_{j1} = -1$ and $b_j = 0$ with $\boldsymbol{a} = \boldsymbol{\phi}(f)$. Hence for $C > 1$, our model is a generalization of rate-based models that implicitly assume neurons can be described by a single scalar rate parameter. The variability in such models is determined by a simple parametric relationship to the rate set by the count distribution, as can be seen for the count distribution families above.

C Neural dispersion and goodness-of-fit quantification

C.1 Fano factors and traditional Z-scores

The traditional Z-score [8, 47, 23] and Fano factor [28, 50] have been used widely in the literature to quantify the variability in neural responses. The two measures are directly related

$$\text{FF} = \langle Z^2\rangle \quad \text{with} \quad Z = \frac{y - \langle y\rangle}{\sqrt{\langle y\rangle}}, \tag{22}$$

with y denoting spike counts and $\langle\cdot\rangle$ the average over the relevant set of trials or time segments of experimental data. These measures are mostly applied to trial-based data, but they can also be applied across separate time windows within a given trial or run in continual recordings. In fact, under stationarity the Fano factor across trials is related to the coefficient of variation in spike time

intervals within trials [25]. In continuous tasks as free animal navigation, the Z-score is often used to quantify dispersion [8, 23, 12].

Note the normality of Z under Poisson data is only asymptotically true, in the sense that we require the predicted average count $\langle y \rangle \ll 1$. The generalized Z-score in subsection F.4 are Gaussian under the true model by design, independent of the spike count magnitudes. However, segments with low expected spike counts around 1 are affected significantly by the dequantization noise, hence the normality in those cases is due to the dequantization rather than model fit.

From Equation 22, we can see that our definition of a dispersion measure T_{DS} in Equation 11 is very similar to the Fano factor. However, its 0 value (corresponding to the unity Fano factor) is defined with an arbitrary reference count models, while unity Fano factors refer specifically to Poisson models. The logarithm reduces skewness of the quantity, leading to a more Gaussian T_{DS} at finite data points. This is useful for statistical testing and computing accurate approximations of its sampling distribution.

C.2 Kolmogorov-Smirnov framework

For finite N, T_{KS} as in Equation 10 has an asymptotic sampling distribution from the Brownian bridge [26]. This statistic can be interpreted as an out-of-distribution score for the observed sample, with significant misfit when T_{KS} is above significance value.

Conventional statistics uses hypothesis testing to assess the model fit, with the null hypothesis being our model. We can obtain model acceptance regions based on some cutoff significance value of the test statistic under its sampling distribution, often taken to be 5%. An alternative is to assess how close the empirical distribution of the test statistic is to the sampling distribution, which is the expected distribution of the statistic under the predictive model. This can be done with another Kolmogorov-Smirnoff test. In this paper, we plot the acceptance regions of T_{KS} and show them compared to baseline models to highlight the model fit improvement on the data it was fit on. T_{DS} was treated similarly as a test statistic.

As we use a predictive model in the Kolmogorov-Smirnov framework, our method is applicable to data beyond repeating trial structure. Continual recordings such as freely moving animals in navigation can therefore be analyzed directly. Our model predicts neural activity at any input point, hence we do not need to rely on repetitive structure in the inputs to learn the activity distribution at a given point. The model plays the role of a reference distribution for evaluating Z-scores (subsection F.4), and thus quantifying dispersion T_{DS} (Equation 11) and goodness-of-fit T_{KS} (Equation 10) of the data to our predictive model.

D The sampling distribution of T_{DS}

Under the true model, generalized Z-scores subsection F.4 are i.i.d. Gaussian variables across neurons and time, hence the dispersion measure T_{DS} based on the sample variance of Z follows a χ^2-distribution. Here we present its asymptotic properties that justify our definition of T_{DS} in Equation 11, and provides the expressions of the moments for the asymptotic normal sampling distribution of T_{DS} used for statistical testing and confidence intervals.

For i.i.d. Gaussian $Z_i \sim \mathcal{N}(0, 1)$, the population variance

$$s^2 = \frac{1}{N} \sum_i Z_i^2 \tag{23}$$

has Ns^2 distributed as a χ^2-distribution with N degrees of freedom.

The moment generating function defined as $M(t) = \langle e^{-tX} \rangle_X$ is a useful quantity for computing the moments of a distribution $p(X)$. Note that $M^{(n)}(0)$, indicating the n-th derivative with respect to time, gives us $(-1)^n \langle X^n \rangle_X$. When we consider the asymptotic convergence to a normal distribution of the χ^2-distribution, the distribution of $\log s^2$ has more favourable convergence property as it is

less skewed due to the logarithmic transformation [27]. The moment generating function is

$$
\begin{aligned}
M(t) &= \int_0^\infty (s^2)^{-t} \left(\frac{Ns^2}{2\sigma^2}\right)^{\frac{N}{2}-1} e^{-\frac{Ns^2}{2\sigma^2}} \frac{Ns}{\sigma^2} \,\mathrm{d}s \,/\, \Gamma\left(\frac{N}{2}\right) \\
&= \left(\frac{2\sigma^2}{N}\right)^{-t} \Gamma\left(\frac{N}{2} - t\right) \,/\, \Gamma\left(\frac{N}{2}\right)
\end{aligned} \tag{24}
$$

which gives rise to the cumulant function

$$
K(t) = \log M(t) = t \log \frac{N}{2} + \log \Gamma\left(\frac{N}{2} - t\right) - \log \Gamma\left(\frac{N}{2}\right) \tag{25}
$$

From here we can compute the first two cumulants as $\kappa_n = K^{(n)}(0)$ similar to the moment generating function, which are equivalent to the mean and variance of the distribution

$$
\begin{aligned}
\mu &= \kappa_1 = \psi\left(\frac{N}{2}\right) - \log \frac{1}{2}N \\
\sigma^2 &= \kappa_2 = \psi'\left(\frac{N}{2}\right)
\end{aligned} \tag{26}
$$

with $\psi(x) = \Gamma'(x)$ i.e. the first derivative of the Gamma function, and the notation $f'(x) = \mathrm{d}f(x)/\mathrm{d}x$. For values $N \gtrapprox 20$, the following asymptotic expression hold well [27]

$$
\begin{aligned}
\mu &= -\left(\frac{1}{N} + \frac{1}{3N^2}\right) \\
\sigma^2 &= \frac{2}{N-1}
\end{aligned} \tag{27}
$$

E Implementation details

E.1 Mathematical details of the optimization objective

E.1.1 The sparse Gaussian process posterior

The Variational Sparse Gaussian Process (VSGP) combines Sparse Gaussian Processes [37] with variational inference [45] to deal with general likelihoods beyond Gaussian. To get scalability to large datasets with batched training, we apply stochastic variational inference (SVI) to obtain stochastic estimates of the ELBO for terms that are not tractable analytically [44]. To make the GP model amenable to subsampling, the sparse approximation fulfills a double role. Inducing points provide a set of global variables that allow subsampling or minibatching common in deep learning [38]. SVI relies on minibatching to estimate the ELBO with subsampled data, and allows scalability to very large datasets. This is similar to amortizing the inference into the inducing points, like neural networks and their weight parameters. Additionally, the inducing points reduce the overall computational complexity for evaluating the GP model to $O(TM^2)$ with M inducing points, assuming $M \ll T$ total number of time or data points.

For simplicity, we work with vectors of scalar GP function values $\boldsymbol{f}$ as locations X. We define the function values at inducing point locations $\boldsymbol{u}$. The GP kernel is evaluated as functions points K_{ff} or at inducing point locations X_u, denoted by K_{uu}. Cross-covariances are denoted by K_{fu} and K_{uf}. The joint variational distribution to the augmented posterior $p(\boldsymbol{f}, \boldsymbol{u}|\boldsymbol{y})$ is defined as

$$
q(\boldsymbol{f}, \boldsymbol{u}) = p(\boldsymbol{f}|\boldsymbol{u})\, q(\boldsymbol{u}) \tag{28}
$$

where the variational distribution $q(\boldsymbol{u}) = \mathcal{N}(\boldsymbol{m}, S)$. The variational distribution over GP function values $q(\boldsymbol{f})$ is simply obtained by marginalizing out $\boldsymbol{u}$, which leads to a Gaussian with

$$
\begin{aligned}
\mathbb{E}[\boldsymbol{f}] &= K_{fu} K_{uu}^{-1} \boldsymbol{m} \\
\mathrm{Cov}[\boldsymbol{f}] &= K_{ff} - K_{fu} K_{uu}^{-1} K_{uf} + K_{fu} K_{uu}^{-1} S K_{uu}^{-1} K_{uf}.
\end{aligned} \tag{29}
$$

this variational posterior can then be used in the variational inference objective Equation 4, more precisely as $q(F|X, Z)$ appearing in Equation 33. This allows one to learn the inducing point locations X_u, as well as the mean $\boldsymbol{m}$ and covariance S of $q(\boldsymbol{u})$.

To accelerate convergence, whitening was used. One performs a change of variables $\boldsymbol{v} = L_{uu}^{-1}\boldsymbol{u}$ with $L_{uu}L_{uu}^T = K_{uu}$. This transforms $p(\boldsymbol{u})$ into $p(\boldsymbol{v}) = \mathcal{N}(\mathbf{0}, I)$, and we now directly optimize $q(\boldsymbol{v}) = \mathcal{N}(\boldsymbol{m}_v, S_v)$ [39]. As in practice matrix products with K_{uu}^{-1} are evaluated using $(L_{uu}L_{uu}^T)^{-1}$, which splits up into two triangular matrices that are readily inverted, the whitened representation simplifies Equation 29 as we no longer explicitly compute $L_{uu}^{-1}\boldsymbol{m}$ and $L_{uu}^{-1}S(L_{uu}^{-1})^T$. The KL-divergence also simplifies as we now have unit normal $p(\boldsymbol{v})$.

To increase the expressivity of multi-output GPs, a separate set of inducing points locations is used for each output dimension (neuron in this work), along with separate kernel hyperparameters as lengthscales for each input and output dimension. This is equivalent to modelling each output dimension by a separate GP, and leads to an overall computational complexity of $O(NCTM^2)$ for our model (see section 2 for notation of quantities). A thorough description of a scalable multi-output VSGP framework is given in [33]. We denote the multi-output variational posterior by $q(F|X, Z)$ with inputs X and Z.

E.1.2 Generative model and variational inference

The overall generative model Equation 1 as depicted in Figure 1 is

$$P_\theta(Y|X) = \int\int P(Y|\Pi)\, p_\theta(\Pi|X, Z)\, p_\theta(Z)\, \mathrm{d}\Pi\, \mathrm{d}Z \tag{30}$$

with the product of individual count distributions $P(Y|\Pi)$. The model parameters θ include the GP θ^{GP} and the prior θ^{pr} (hyper)parameters, as well as the softmax mapping weights W_n and biases $\boldsymbol{b}_n$. Note that the distribution over count probabilities

$$p(\Pi|X, Z) = \int p(\Pi|F)\, p(F|X, Z)\, \mathrm{d}F \tag{31}$$

contains the Gaussian process prior over F. The mapping from F to Π denoted by $\Pi(F)$ (Equation 1) is deterministic, and therefore $p(\Pi|F)$ is a delta distribution $\delta(\Pi - \Pi(F))$.

The exact Bayesian posterior over Π and Z is intractable, hence we use an approximate posterior as defined in Equation 3. The variational parameters φ specify the latent variational posterior, while χ consists of inducing point locations X_u and the means and covariance matrices of $q(U)$ for the sparse Gaussian process posterior $q(F|X, Z)$ (Equation 28). The wrapped normal distribution used for circular dimensions in $q(Z)$, i.e. dimensions with $z \in [0, 2\pi)$, takes the form [43]

$$\mathcal{N}_{\text{wrap}}(z|\mu, \sigma^2) = \sum_{k=-\infty}^{\infty} \mathcal{N}(z|\mu + 2\pi k, \sigma^2) \tag{32}$$

and was evaluated with a finite cutoff at $k = \pm 5$ of the infinite sum. This is an accurate approximation as long as $\sigma \ll 2\pi$. When plotting the standard deviations of the approximate posterior $q(Z)$, we plot σ for both Euclidean as well as circular variables. This is similarly an accurate approximation in the circular case when $\sigma \ll 2\pi$, which was true in practice.

The marginal likelihood in Equation 30 is intractable. Instead, we minimize the negative ELBO or free energy loss objective using our approximate posterior

$$\begin{aligned}\mathcal{F}_{\theta,\varphi} &= -\mathbb{E}_{Z\sim q_\varphi(Z)}\mathbb{E}_{\Pi\sim q_{\theta,\chi}(\Pi|X_\mathcal{D},Z)}\left[\log \frac{P(Y_\mathcal{D}|\Pi)\, p_\theta(\Pi|X_\mathcal{D}, Z)\, p_\theta(Z)}{q_{\theta,\chi}(\Pi|X_\mathcal{D}, Z)\, q_\varphi(Z)}\right] \\ &= \mathcal{F}_{\text{lik}} + \mathcal{F}_{\text{reg}}\end{aligned} \tag{33}$$

which is an upper bound to the negative log marginal likelihood [45, 39]. The objective decomposes into a log likelihood expectation term $\mathcal{F}_{\text{lik}}$ and some regularization terms arising from the model priors $\mathcal{F}_{\text{reg}}$. These terms are amenable to Monte Carlo evaluation or quadrature approximation as we show next, and in some cases are even available in closed form.

The variational expectation of the log likelihood

$$\mathcal{F}_{\text{lik}} = -\mathbb{E}_{Z\sim q_\varphi(Z)}\mathbb{E}_{\Pi\sim q_{\theta,\chi}(\Pi|X_\mathcal{D},Z)}[P(Y_\mathcal{D}|\Pi)] \tag{34}$$

can be evaluated using Monte Carlo sampling to obtain unbiased estimates in the general case. As an alternative method, Gauss-Hermite quadratures can provide a deterministic approximation to the expectation with respect to $q(F|X, Z)$ [39]

$$\mathbb{E}_{\Pi\sim q_{\theta,\chi}(\Pi|X_\mathcal{D},Z)}[P(Y_\mathcal{D}|\Pi)] = \mathbb{E}_{F\sim q_{\theta,\chi}(F|X_\mathcal{D},Z)}[P(Y_\mathcal{D}|\Pi(F))] \tag{35}$$

where $\Pi(F)$ denotes the transformation from F to count probabilities as in Equation 1. Here we used

$$q(\Pi|X, Z) = \int \delta(\Pi - \Pi(F))\, q(F|X, Z)\, \mathrm{d}F \tag{36}$$

analogous to Equation 31. This corresponds to a zero variance estimator with a small bias for sufficiently many quadrature points. To allow fast MC sampling, the variational posterior $q(F)$ is approximated with its diagonal covariance matrix. This removes the correlations between posterior function points at different input values, but allows sampling from very high dimensional distributions (i.e. many time points, large batch sizes). The issue of efficiently sampling from the full posterior has been considered in [15].

As the ratio of $P_\theta(\Pi|X_\mathcal{D}, Z)$ and $q_{\theta,\chi}(\Pi|X_\mathcal{D}, Z)$ has the transformation from F to Π cancel out

$$D_{\mathrm{KL}}(q_{\theta,\chi}(\Pi|X_\mathcal{D}, Z)||p_\theta(\Pi|X_\mathcal{D}, Z)) = q(\Pi|X, Z) = \int \delta(\Pi - \Pi(F))\, q(F|X, Z)\, \mathrm{d}F \tag{37}$$

the regularization terms consist of Kullback-Leibler divergences

$$\mathcal{F}_{\text{reg}} = D_{\mathrm{KL}}(q_{\theta,\chi}(F|X_\mathcal{D}, Z)||p_\theta(F|X_\mathcal{D}, Z)) + D_{\mathrm{KL}}(q_\varphi(Z)||p_\theta(Z)) \tag{38}$$

that can be computed with analytical expressions in the case when all distributions are normal.

E.2 Latent space priors

We use the Markovian priors as specified in Equation 2. These priors can be specified on different manifolds, in particular we use for Euclidean spaces the linear dynamical system prior

$$p(\boldsymbol{z}_{t+1}|\boldsymbol{z}_t) = \mathcal{N}(A\boldsymbol{z}_t, \Sigma) \tag{39}$$

In particular, we use diagonal Σ and A to learn factorized latent states. We constrain $A_{ii} = a_i \in (-1, 1)$ for stability, and we fix $\Sigma_{ii} = \sigma_i^2 = 1/(1 - a_i^2)$ to obtain a prior process with stationary variance 1 while optimizing for a_i. On the toroidal manifold, we use

$$p(\boldsymbol{z}_{t+1}|\boldsymbol{z}_t) = \mathcal{N}(\boldsymbol{z}_t + \boldsymbol{c}, \Sigma) \tag{40}$$

as due to rotational symmetry $A = I$. Again, we use diagonal Σ. Both $\boldsymbol{c}$ and $\Sigma_{ii} = \sigma_i^2$ are learned as part of the generative model.

When temporally batching input, one has to be careful to retain the continuity in the prior $p(Z)$ with the previous batch (beyond the first batch at the start). This is done by ensuring that the first $\boldsymbol{z}_t$ in the batch is the last step in the previous batch, and this will correctly subsample the prior $p(Z)$ defined over the entire input time series. When performing cross-validation with validation segments within the overall input time series, we treat the gap as a discontinuity in the latent trajectory and do not include the latent state right before the validation segment.

E.3 Gaussian process kernel functions

In this work, we used the RBF kernel defined on Euclidean and toroidal manifolds [42]. In particular, this kernel function is given by

$$k(\boldsymbol{x}, \boldsymbol{y}) = \sigma^2\, e^{-\frac{1}{2}\, \sum_{i=1}^{D} d_i^2(x_i, y_i; l_i)} \tag{41}$$

with rescaled distances

$$\begin{aligned} d_{\mathbb{R}}^2(x, y; l) &= \left(\frac{x - y}{l}\right)^2 \\ d_{\mathbb{T}}^2(x, y; l) &= 2\left(\frac{1 - \cos(x - y)}{l}\right)^2 \end{aligned} \tag{42}$$

for Euclidean and toroidal spaces $\mathbb{R}$ and $\mathbb{T}$, respectively. To cover different input dimensions of different topologies, we use product kernels with suitable distances d per input dimension, resulting in sums over dimensions in Equation 41. These distance functions can be used to extend other kernels such as Matérn kernels to non-Euclidean spaces [42].

Algorithm 1 Joint latent-regression inference scheme

Input spike counts $Y_{\mathcal{D}}$, observed covariates (e.g. behaviour) $X_{\mathcal{D}}$
Define neurons N , maximum spike count K, channels C
Batch data over temporal dimension while taking into account continuity in the prior $p(Z)$ (see subsection E.2)
while not converged **and** iterations below bound **do**
    Adapt learning rates (if annealing)
    for each batch **do**
        Generate m MC samples $z \sim q_\varphi(z)$ (if relevant), m copies of $x = X$
        With C Gaussian processes for neuron compute posterior $q(F|X, Z)$
        Generate k MC samples per input sample of x and z
        Concatenate posterior C function values per neuron into vectors $\boldsymbol{f}$
        Evaluate the basis expansion $\boldsymbol{a} = \boldsymbol{\phi}(\boldsymbol{f})$
        Compute probabilities $P(y|\boldsymbol{a}) = \text{softmax}(W\boldsymbol{a} + \boldsymbol{b})$ for all neurons
        Compute the loss from the mean of $m \times k$ MC samples of the cross-entropy $-\log P(y_{\mathcal{D}}|\boldsymbol{a})$ rescaled by ratio of time points in data over batch size
        Perform the backward pass using automatic differentiation to compute gradients for parameters θ, χ and φ
        Take a gradient step with some optimizer
    end for
end while

E.4 Overall algorithm and code

Additionally, instead of drawing Monte Carlo samples for the Gaussian variational posterior $q(F)$, we provide the option to compute the Gaussian expectation using Gauss-Hermite quadratures []. This was used to estimate the cvLLs (Equation 9) for models after training, which reduced stochasticity in the cvLL estimate with a negligible bias using 100 quadrature points.

MC sample or quadrature point dimensions are parallelized over in addition to other dimensions like neurons or time, using extra tensor dimensions in modern automatic differentiation libraries. We use PyTorch [36] to implement the algorithm for inference of our model. We use Adam [46] as our optimizer, with no weight decay and default optimizer hyperparameters in PyTorch.

The code provided contains a library with implementations of Gaussian process and GLM based models with different likelihoods as used for baseline models in this paper. In addition to count likelihoods, it contains an implementation of spike-spike and spike-history couplings [29, 39] and modulated renewal processes [2, 49] to deal with data at the individual spike time level. All models can be run with both observed and latent inputs on Euclidean and toroidal manifolds [42].

E.5 Model fitting

E.5.1 Inducing point initialization

The first input dimension had its inducing points uniformly spaced between 0 and 2π for circular dimensions, and -1 to 1 for Euclidean latent dimensions. Observed dimensions had natural intervals defined by the behavioural statistics (e.g. 0 to the mean animal speed), and we placed inducing points uniformly throughout this interval. For the other dimensions, we initialized random inducing point locations based on the topology of the input variable. We place Euclidean variables as a random uniform distribution in its corresponding interval as described previously, while circular variables took on random uniform values in $[0, 2\pi]$.

The number of inducing points has been shown to scale favourably as $O((\log T)^D)$ for standard Gaussian process regression models [40]. In this work, we used $O(D \log T)$ which captured rich tuning and satisfactory model fits combined with the flexible count distributions. The suggested $O((\log T)^D)$ does become computationally expensive for high dimensional input, and was not tried with the high-dimensional regression models.

E.5.2 Fitting details

We select the model with the lowest loss from 3 separate model fits, initialized with randomized inducing points as described above. The maximum number of training epochs was 3000, but we stopped training before if the loss did not decrease more than $\approx 10^{-3}$ percent over 100 steps. The learning rate was set to 10^{-2}, and we also anneal the learning rate every 100 steps by a factor 0.9. In the case of latent spaces, we used a learning rate of 10^{-3} for standard deviations of the variational distribution $q(Z)$. All cases lead to satisfactory convergence of the model.

For latent variable models with a single angular latent, we initialize the lengthscale at large values. This avoided the model to overfit and fold the latent space as seen in panel A of Figure 3 for the ANN model. For these models, the best fits were achieved with an initial learning rate of $3 \cdot 10^{-2}$ and $5 \cdot 10^{-3}$ for the kernel lengthscale and the standard deviations of the variational distribution $q(Z)$.

E.5.3 Hardware and fitting time

Synthetic data was analyzed with GeForce RTX 2070 (8 GB of memory). Real data was analyzed with Nvidia GeForce RTX 2080Ti GPUs (with 11 GB of memory). Fitting 33 neurons with $\sim 6 \cdot 10^4$ time points with the regression model in Figure 3 takes around 20 minutes, while fitting with a four dimensional latent spaces added takes around 50 minutes. These numbers can fluctuate depending on the flexible stopping criterion above. Generally, there is a trade-off between memory usage and speed by setting the batch size, with larger batch sizes being generally faster but taking more memory.

F Analysis details

F.1 Synthetic data

We construct a synthetic head direction cell population inspired by bump attractor models [4, 42, 42]. Firing rate tuning curves to head direction are modelled as von Mises bumps with some constant offset

$$f(\theta; b, A, \beta, \theta_0) = A\, e^{\beta\, \cos\theta - \theta_0} + b \tag{43}$$

with $b > 0$ and $A > 0$. This results in $f \geq 0$ for all valid inputs and parameters. For modelling firing rates, we additionally restrict ourselves to $\beta > 0$ to avoid inverted bumps at the preferred head direction θ_0.

For the modulation by a hidden Euclidean signal in the modulated Poisson population, we additionally place Gaussian tuning curves on the latent dimensions with varying standard deviations and means. The Gaussian tuning curves tile the latent space that was traversed, which allows the model to infer the full trajectory. Note that tuning is factorized across the two dimensions (head direction x and latent signal z). Parameters were chosen from random distributions that led to firing rates and variability within the physiological regime.

In the Conway-Maxwell-Poisson (CMP) synthetic population, we place the tuning curves from Equation 43 on parameters ν and the approximate mean $\mu_y = \mathbb{E}[y]$ in Equation 20. Note both parameters have to non-negative to be in the valid range. Furthermore, the tuning curves of ν had potentially negative $\beta \in \mathbb{R}$ and different parameter statistics than for μ_y. Again, these were chosen such that firing rates and variability were within the physiological regime. From the approximate relation Equation 20 of the mean, we obtain

$$\lambda == \left(\mu_y - \frac{1}{2\nu} + \frac{1}{2}\right)^{\nu} \tag{44}$$

to match roughly the mean counts with von Mises bump patterns. To sample from the CMP distribution once we specified λ and ν, we use the fast rejection sampling method [43].

F.2 Neural data

Data was taken from Mouse 28, session 140313, during the wake phase [40]. The spiking data was recorded at a resolution of 20000 Hz, whereas behaviour was extracted from video recordings of animal body tracking at a resolution of 39.06 Hz. Note the time of the first video frame was randomly misaligned by 0–60 ms to the neural spike trains. We removed invalid behavioural segments in the

data and performed linear interpolation across those segments. For circular variables, interpolation was taken in the shortest geodesic distance. We binned spiking data at 1 ms, and interpolated behavioural data to reach the same sampling frequency that is higher than the behavioural recording frequency. At a binning of 40 ms used in our analysis, we had $K = 11$ as the maximum count value.

We selected head direction cells based on a sparsity criterion, after trying several criteria as mutual information typically used for place cells [44]. First, we binned the head direction variable into 60 equal bins over the range $[0, 2\pi]$. For each bin, we now compute the average spike counts y_i for head directions within bin i, and the relative occupancy P_i. Note $\sum_i P_i = 1$ is a probability distribution. Sparsity is defined as

$$1 - \frac{(\sum_i P_i \, y_i)^2}{\sum_i P_i \, y_i^2} \tag{45}$$

and with a selection criterion of sparsity ≥ 0.2 we obtained 33 head direction cells, of which 15 are in postsubiculum. Alternatively, although more computationally intensive, we could directly regress a Gaussian process model (e.g. Poisson baseline model Equation 46) and look at the kernel lengthscales on the angular input dimension. These will be appreciably larger than 2π for cells that are not tuned much to head direction.

Note that quite a few head direction units, which are supposed to represent single cells, show bimodal tuning curves or more to head direction. This is likely due to multiple neurons as signals can pollute in electrophysiological recordings and spike sorting can fail to distinguish between them [45, 46].

F.3 Baseline models

The log Cox Gaussian process model puts a GP prior on the rate function of an inhomogeneous Poisson process (Equation 13) with an inverse link function $f(x) = e^x$ that is exponential

$$\begin{aligned} h(x) &\sim \mathcal{GP}(\mu_x, k_{xx}) \\ \lambda(x) &= f(h(x)) \\ y &\sim P_{\text{Poiss}}(y|\lambda \cdot \Delta, \theta) \end{aligned} \tag{46}$$

where the time bin length is Δ, which turns λ into a proper rate quantity.

The heteroscedastic negative binomial model builds on this encoding model, More precisely, two GPs with an exponential inverse link function are used to model tuning to covariates of the rate λ and inverse shape $1/r$ of the negative binomial likelihood (Equation 16), leading to the model

$$\begin{aligned} h(x) &\sim \mathcal{GP}(\mu_x, k_{xx}), \quad g(x) \sim \mathcal{GP}(\mu_x, k_{xx}) \\ \lambda(x) &= f(h(x)), \quad \frac{1}{r} = f(g(x)) \\ y &\sim P_{\text{NB}}(y|\lambda \cdot \Delta, r) \end{aligned} \tag{47}$$

F.4 Generalized Z-scores

The generalized Z-scores in provide a normalized quantification of neural activity under the predictive model. The count distribution $P(y)$ is taken to be the mean posterior count distribution of the posterior $q(\Pi|X, Z)$. In the case of baseline models, the reference $P(y)$ is given by the parametric distribution (Poisson in Equation 13, negative binomial in Equation 16) evaluated at the mean posterior values of the count distribution parameters given by the Gaussian process mapping (see Equation 46 and Equation 47). This is strictly speaking different from the mean posterior count distribution, as the parametric distribution depends non-linearly on these parameters. However, the difference is insignificant when the variational uncertainties are small, which was often the case in practice.

F.5 Marginal and conditional tuning curves

Due to the high dimensional input space, we can either visualize slices of the tuning curve over the relevant input variables $\boldsymbol{x}_*$ or instead marginalize over other input variables. The conditional tuning curves are based on the count distributions $P(y|\boldsymbol{x}_*, \boldsymbol{x}^c)$, where $\boldsymbol{x}^c$ are fixed and cover the dimensions complementary to $\boldsymbol{x}_*$ (these are plotted in Figure 3C). On the other hand, marginalizing over $\boldsymbol{x}^c$ is equivalent to an experimenter only looking at neural tuning to $\boldsymbol{x}_*$, which automatically marginalizes

over all other behaviour not included. Mathematically, this can be interpreted as considering the $\boldsymbol{x}^c$-dimensions of a Markov Chain Monte Carlo path sampled from the joint density $p_{\mathcal{D}}(\boldsymbol{x})$

$$P(y|\boldsymbol{x}_*) = \int P(y|\boldsymbol{x}_*, \tilde{\boldsymbol{x}}^c)\, p_{\mathcal{D}}(\tilde{\boldsymbol{x}})\, \mathrm{d}\tilde{\boldsymbol{x}} \approx \sum_t P(y|\boldsymbol{x}_*, (\boldsymbol{x}^c_{\mathcal{D}})_t) \tag{48}$$

which defines the marginalization through the computation done in practice (summing over the time series of observed $\boldsymbol{x}^c$ while keeping $\boldsymbol{x}_*$ fixed). From this marginalized distribution, we can compute similar quantities as before.

For the tuning indices, we evaluate the the count statistic $T_y(\boldsymbol{x}_*)$ with respect to the posterior mean distribution $P(y|\boldsymbol{x}_*)$ after marginalizing (order does not matter as both are sums) to compute the tuning indices as described in Equation 6. Optimization over $\boldsymbol{x}_*$ of $T_y(\boldsymbol{x}_*)$ is done by grid search, as $\boldsymbol{x}_*$ is low-dimensional and we compute its values over a grid anyway for plotting tuning curves of mean, Fano factor or any other count statistic.

We used 300 Monte Carlo samples from $q(\Pi|X, Z)$ to compute the conditional tuning curves plotted in this paper. For marginalized tuning curves, we use 100 MC samples and temporally subsampled the observed input $X_{\mathcal{D}}$ to retain the first time step for every 10 time steps, and used this to evaluate Equation 48. As behaviour shows strong temporal correlations at short time scales (Figure 3G), this allows us to estimate the marginal tuning curves more efficiently. The mean of these samples was used to compute the mean posterior tuning curves for evaluating the TIs. When evaluating the average mean count and Fano factor at every time step (Figure 3B and Figure 4B), we used 10 MC samples from $q(\Pi|X, Z)$. When latent variables were present (Figure 3E), the 10 MC samples were drawn from $q(Z)$, corresponding to $m = 10$ and $k = 1$ in Algorithm 1.

F.6 Temporal cross-correlations of covariates

We use the cross-correlation between time series x_t and y_t

$$r_{xy}(\Delta) = \frac{\langle (x_{t+\Delta} - \langle x_{t+\Delta}\rangle)(y_t - \langle y_t\rangle)\rangle}{\sigma_x \sigma_y} \tag{49}$$

which includes the auto-correlation as a special case, e.g. $r_{xx}(\Delta)$. When one of the variables is a circular variable θ_t, we use the linear-circular correlation coefficient in [47]

$$\begin{aligned} s_t &= \sin\theta_t, \quad c_t = \cos\theta_t \\ R_{xs} &= r_{xs}(\Delta), \quad R_{xc} = r_{xs}(\Delta), \quad R_{cs} = r_{cs}(\Delta) \\ r_{x\theta} &= \frac{R_{xs}^2 + R_{xc}^2 - 2\, R_{xs}\, R_{xc}\, R_{cs}}{1 - R_{cs}^2} \end{aligned} \tag{50}$$

and for the case when both are circular, we use the circular correlation coefficient proposed by [48]

$$\begin{aligned} s_\theta &= \sin\left(\theta_t - \mathrm{Arg}\,\mathbb{E}[e^{i\theta_t}]\right), \quad \text{same for } \phi \\ r_{\theta\phi}(\Delta) &= \frac{\mathbb{E}[s_\theta \cdot s_\phi]}{\mathbb{E}[s_\theta^2]^{\frac{1}{2}}\, \mathbb{E}[s_\phi^2]^{\frac{1}{2}}} \end{aligned} \tag{51}$$

Time scales are estimated from the auto-correlations of covariates. The time scale τ is then chosen as the time step at which the value of the auto-correlation dropped by a factor e from 1 at $\Delta = 0$.

F.7 Preferred head direction

To compute the preferred head direction θ_{pref}, we use the centre-of-mass of the firing rate profile $r(\theta)$ of head direction θ

$$\theta_{pref} = \mathrm{Arg}[r(\theta)e^{i\theta}] \tag{52}$$

which is more robust to noise than taking the angle at which $r(\theta)$ is at a maximum. We can evaluate θ_{pref} as a function of angular head velocity (AHV) and absolute time to compute the ATIs and the neural drift as described in Appendix A.

F.8 Circular-linear regression

We computed the circular-linear regression [49] using a measure of the correlation between circular variables θ_1 and θ_2

$$R = |\mathbb{E}[e^{i(\theta_1 - \theta_2)}]| \tag{53}$$

By computing R between a circular-linear function $\phi(t)$

$$\phi(t) = 2\pi a t + b \tag{54}$$

and the circular data time series θ_t, we can perform the regression by maximizing R through optimizing the parameter a with gradient descent. The offset b is obtained analytically

$$b = \text{Arg}\, \mathbb{E}_t[e^{i(\theta_t - \phi(t))}] \tag{55}$$

From the values a after fitting, one can compute the linear drift values and ATIs as described in Appendix A. In addition, not all cells are well-described by the linear drift or ATIs, so we discarded cells which had an optimized value of $R < 0.999$. This cutoff was chosen as it retains cells that are visually in agreement with linear relations as seen in Figure 4, while discarding a few outlier cells.

F.9 Latent alignments

To align 1D circular latent trajectories z_c to a target trajectory, we minimize their mean geodesic distance under a constant shift μ and potential sign flip $s = \pm 1$

$$\tilde{z}_c = s \cdot z_c + \mu \tag{56}$$

We add a linear drift Δ

$$\tilde{z}_c = s \cdot z_c + \mu + t \cdot \Delta \tag{57}$$

to find potential drifting of the inferred trajectory as done in panel D of Figure 4. This is similar to the circular-linear regression above [49], but with the geodesic distance on the ring instead. This is consistent with root-mean-square errors in the latent signal from behaviour that are computed with the geodesic distances. For 1D Euclidean latent trajectories, we align by fitting a translation and scaling parameter.

In all cases, the root mean squared error (RMSE) of the alignment is evaluated in a cross-validated manner. For circular variables, we use the geodesic distance for computing the squared error just as in aligning. In more detail, we fit the trajectory transformation parameters such that we minimize the errors on the validation segment, and then use these fitted parameters to compute the transformed latent trajectory in the held-out segment. This is then used to compute the RMSE for the alignment of the cross-validation fold.

References

[1] Riccardo Barbieri, Michael C Quirk, Loren M Frank, Matthew A Wilson, and Emery N Brown. Construction and analysis of non-poisson stimulus-response models of neural spiking activity. *Journal of neuroscience methods*, 105(1):25–37, 2001.

[2] Emery N Brown, Riccardo Barbieri, Valérie Ventura, Robert E Kass, and Loren M Frank. The time-rescaling theorem and its application to neural spike train data analysis. *Neural computation*, 14(2):325–346, 2002.

[3] Jonathan W Pillow. Time-rescaling methods for the estimation and assessment of non-poisson neural encoding models. In *Advances in neural information processing systems*, pages 1473–1481, 2009.

[4] William E Skaggs, Bruce L McNaughton, Matthew A Wilson, and Carol A Barnes. Theta phase precession in hippocampal neuronal populations and the compression of temporal sequences. *Hippocampus*, 6(2):149–172, 1996.

[5] Angus Chadwick, Mark CW van Rossum, and Matthew F Nolan. Independent theta phase coding accounts for ca1 population sequences and enables flexible remapping. *Elife*, 4:e03542, 2015.

[6] Jakob Gulddahl Rasmussen. Lecture notes: Temporal point processes and the conditional intensity function. *arXiv preprint arXiv:1806.00221*, 2018.

[7] Hongyuan Mei and Jason Eisner. The neural hawkes process: A neurally self-modulating multivariate point process. *arXiv preprint arXiv:1612.09328*, 2016.

[8] Shuai Xiao, Junchi Yan, Xiaokang Yang, Hongyuan Zha, and Stephen Chu. Modeling the intensity function of point process via recurrent neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

[9] Takahiro Omi, Naonori Ueda, and Kazuyuki Aihara. Fully neural network based model for general temporal point processes. *arXiv preprint arXiv:1905.09690*, 2019.

[10] Oleksandr Shchur, Nicholas Gao, Marin Biloš, and Stephan Günnemann. Fast and flexible temporal point processes with triangular maps. *Advances in Neural Information Processing Systems*, 33, 2020.

[11] Yaniv Ziv, Laurie D Burns, Eric D Cocker, Elizabeth O Hamel, Kunal K Ghosh, Lacey J Kitch, Abbas El Gamal, and Mark J Schnitzer. Long-term dynamics of ca1 hippocampal place codes. *Nature neuroscience*, 16(3):264, 2013.

[12] Michael E Rule, Adrianna R Loback, Dhruva V Raman, Laura N Driscoll, Christopher D Harvey, and Timothy O'Leary. Stable task information from an unstable neural population. *Elife*, 9:e51121, 2020.

[13] Johannes Zirkelbach, Martin Stemmler, and Andreas VM Herz. Anticipatory neural activity improves the decoding accuracy for dynamic head-direction signals. *Journal of Neuroscience*, 39(15):2847–2859, 2019.

[14] Adrien Peyrache and György Buzsáki. Extracellular recordings from multi-site silicon probes in the anterior thalamus and subicular formation of freely moving mice. *CRCNS*, 2015.

[15] James T. Wilson, Viacheslav Borovitskiy, Alexander Terenin, Peter Mostowsky, and Marc Peter Deisenroth. Efficiently sampling functions from gaussian process posteriors. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 10292–10302. PMLR, 2020.

[16] Andrew Wilson and Hannes Nickisch. Kernel interpolation for scalable structured gaussian processes (kiss-gp). In *International Conference on Machine Learning*, pages 1775–1784. PMLR, 2015.

[17] Andrew Gordon Wilson, Christoph Dann, and Hannes Nickisch. Thoughts on massively scalable gaussian processes. *arXiv preprint arXiv:1511.01870*, 2015.

[18] Johannes Nagele, Andreas VM Herz, and Martin B Stemmler. Untethered firing fields and intermittent silences: Why grid-cell discharge is so variable. *Hippocampus*, 2020.

[19] Adam S Charles, Mijung Park, J Patrick Weller, Gregory D Horwitz, and Jonathan W Pillow. Dethroning the fano factor: a flexible, model-based approach to partitioning neural variability. *Neural computation*, 30(4):1012–1045, 2018.

[20] Galit Shmueli, Thomas P Minka, Joseph B Kadane, Sharad Borle, and Peter Boatwright. A useful distribution for fitting discrete data: revival of the conway–maxwell–poisson distribution. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 54(1):127–142, 2005.

[21] André A Fenton and Robert U Muller. Place cell discharge is extremely variable during individual passes of the rat through the firing field. *Proceedings of the National Academy of Sciences*, 95(6):3182–3187, 1998.

[22] Adam Kohn and Matthew A Smith. Stimulus dependence of neuronal correlation in primary visual cortex of the macaque. *Journal of Neuroscience*, 25(14):3661–3673, 2005.

[23] André A Fenton, William W Lytton, Jeremy M Barry, Pierre-Pascal Lenck-Santini, Larissa E Zinyuk, Štepan Kubík, Jan Bureš, Bruno Poucet, Robert U Muller, and Andrey V Olypher. Attention-like modulation of hippocampus place cell discharge. *Journal of Neuroscience*, 30(13):4613–4625, 2010.

[24] Robbe LT Goris, J Anthony Movshon, and Eero P Simoncelli. Partitioning neuronal variability. *Nature neuroscience*, 17(6):858, 2014.

[25] Martin P Nawrot, Clemens Boucsein, Victor Rodriguez Molina, Alexa Riehle, Ad Aertsen, and Stefan Rotter. Measurement of variability dynamics in cortical spike trains. *Journal of neuroscience methods*, 169(2):374–390, 2008.

[26] Nickolay Smirnov. Table for estimating the goodness of fit of empirical distributions. *The annals of mathematical statistics*, 19(2):279–281, 1948.

[27] Maurice S Bartlett and DG Kendall. The statistical analysis of variance-heterogeneity and the logarithmic transformation. *Supplement to the Journal of the Royal Statistical Society*, 8(1):128–138, 1946.

[28] Michalis Titsias. Variational learning of inducing variables in sparse gaussian processes. In *Artificial Intelligence and Statistics*, pages 567–574, 2009.

[29] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations*, 2014.

[30] Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347, 2013.

[31] James Hensman, Nicolo Fusi, and Neil D Lawrence. Gaussian processes for big data. *arXiv preprint arXiv:1309.6835*, 2013.

[32] James Hensman, Alexander Matthews, and Zoubin Ghahramani. Scalable variational gaussian process classification. In *Artificial Intelligence and Statistics*, pages 351–360. PMLR, 2015.

[33] Mark van der Wilk, Vincent Dutordoir, ST John, Artem Artemev, Vincent Adam, and James Hensman. A framework for interdomain and multioutput gaussian processes. *arXiv preprint arXiv:2003.01115*, 2020.

[34] Luca Falorsi, Pim de Haan, Tim R Davidson, and Patrick Forré. Reparameterizing distributions on lie groups. *arXiv preprint arXiv:1903.02958*, 2019.

[35] Kristopher Jensen, Ta-Chu Kao, Marco Tripodi, and Guillaume Hennequin. Manifold gplvms for discovering non-euclidean latent structure in neural data. *Advances in Neural Information Processing Systems*, 33, 2020.

[36] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703*, 2019.

[37] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[38] Jonathan W Pillow, Jonathon Shlens, Liam Paninski, Alexander Sher, Alan M Litke, EJ Chichilnisky, and Eero P Simoncelli. Spatio-temporal correlations and visual signalling in a complete neuronal population. *Nature*, 454(7207):995–999, 2008.

[39] Alison I Weber and Jonathan W Pillow. Capturing the dynamical repertoire of single neurons with generalized linear models. *Neural computation*, 29(12):3260–3289, 2017.

[40] David R Burt, Carl Edward Rasmussen, and Mark van der Wilk. Convergence of sparse variational inference in gaussian processes regression. *Journal of Machine Learning Research*, 21:1–63, 2020.

[41] Jeffrey S Taube, Robert U Muller, and James B Ranck. Head-direction cells recorded from the postsubiculum in freely moving rats. i. description and quantitative analysis. *Journal of Neuroscience*, 10(2):420–435, 1990.

[42] Sung Soo Kim, Hervé Rouault, Shaul Druckmann, and Vivek Jayaraman. Ring attractor dynamics in the drosophila central brain. *Science*, 356(6340):849–853, 2017.

[43] Alan Benson, Nial Friel, et al. Bayesian inference, model selection and likelihood estimation using fast rejection sampling: the conway-maxwell-poisson distribution. *Bayesian Analysis*, 2021.

[44] Sijie Zhang, Fabian Schönfeld, Laurenz Wiskott, and Denise Manahan-Vaughan. Spatial representations of place cells in darkness are supported by path integration and border information. *Frontiers in behavioral neuroscience*, 8:222, 2014.

[45] David Carlson and Lawrence Carin. Continuing progress of spike sorting in the era of big data. *Current opinion in neurobiology*, 55:90–96, 2019.

[46] Jeyathevy Sukiban, Nicole Voges, Till A Dembek, Robin Pauli, Veerle Visser-Vandewalle, Michael Denker, Immo Weber, Lars Timmermann, and Sonja Grün. Evaluation of spike sorting algorithms: Application to human subthalamic nucleus recordings and simulations. *Neuroscience*, 414:168–185, 2019.

[47] Richard A Johnson and Thomas Wehrly. Measures and models for angular correlation and angular–linear correlation. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(2):222–229, 1977.

[48] Nick I Fisher and AJ Lee. A correlation coefficient for circular data. *Biometrika*, 70(2):327–332, 1983.

[49] Richard Kempter, Christian Leibold, György Buzsáki, Kamran Diba, and Robert Schmidt. Quantifying circular–linear associations: Hippocampal phase precession. *Journal of neuroscience methods*, 207(1):113–124, 2012.