bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

Modelling the neural code in large populations of correlated neurons

Sacha Sokoloski[1*], **Amir Aschner**[2], **Ruben Coen-Cagli**[1,2]

***For correspondence:** sacha.sokoloski@mailbox.org (SS)

[1]Department of Systems and Computational Biology; [2]Dominick P. Purpura Department of Neuroscience, Albert Einstein College of Medicine, The Bronx, New York, USA

Abstract The activity of a neural population encodes information about the stimulus that caused it, and decoding population activity reveals how neural circuits process that information. Correlations between neurons strongly impact both encoding and decoding, yet we still lack models that simultaneously capture stimulus encoding by large populations of correlated neurons and allow for accurate decoding of stimulus information, thus limiting our quantitative understanding of the neural code. To address this, we propose a class of models of large-scale population activity based on the theory of exponential family distributions. We apply our models to macaque primary visual cortex (V1) recordings, and show they capture a wide range of response statistics, facilitate accurate Bayesian decoding, and provide interpretable representations of fundamental properties of the neural code. Ultimately, our framework could allow researchers to quantitatively validate predictions of theories of neural coding against both large-scale response recordings and cognitive performance.

Introduction

A foundational idea in sensory neuroscience is that the activity of neural populations constitutes a "neural code" for representing stimuli (***Dayan and Abbott, 2005***; ***Doya, 2007***): the activity pattern of a population in response to a sensory stimulus encodes information about that stimulus, and downstream neurons decode, process, and re-encode this information in their own responses. Sequences of such neural populations implement the elementary functions that drive perception, cognition, and behaviour (***Pitkow and Angelaki, 2017***). Therefore, by studying the encoding and decoding of population responses, researchers may investigate how information is processed along neural circuits, and how this processing influences perception and behaviour (***Wei and Stocker, 2015***; ***Panzeri et al., 2017***; ***Kriegeskorte and Douglas, 2018***).

Given a true statistical model of how a neural population responds to (encodes information about) stimuli, Bayes' rule can transform the encoding model into an optimal decoder of stimulus information (***Zemel et al., 1998***; ***Pillow et al., 2010***). However, when validated as Bayesian decoders, existing statistical models of neural encoding are often outperformed by models trained to decode stimulus-information directly, indicating that the encoding models miss key statistics of the neural code (***Graf et al., 2011***; ***Walker et al., 2020***). In particular, the correlations between neurons' responses to repeated presentations of a given stimulus (noise correlations), and how these noise correlations are modulated by stimuli, can strongly impact coding in neural circuits (***Zohary et al., 1994***; ***Abbott and Dayan, 1999***; ***Sompolinsky et al., 2001***; ***Ecker et al., 2016***; ***Kohn et al., 2016***; ***Schneidman, 2016***), especially in large populations of neurons (***Moreno-Bote et al., 2014***; ***Montijn et al., 2019***; ***Bartolo et al., 2020***; ***Kafashan et al., 2020***; ***Rumyantsev et al., 2020***). Yet effectively modelling noise correlations has proven challenging.

Validating theories of population coding (***Ma et al., 2006***; ***Beck et al., 2011a***; ***Ganguli and Simon-***

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

celli, 2014; *Makin et al., 2015*; *Yerxa et al., 2020*) in large neural circuits thus depends on encoding models that support accurate Bayesian decoding, effectively capture noise-correlations, and efficiently fit large-scale neural recordings. Generalized linear models (GLMs) are one class of model that yield effective Bayesian decoders, and GLMs have been applied to analyzing spatio-temporal features of information processing in the retina and cortex (*Pillow et al., 2008*; *Park et al., 2014*; *Runyan et al., 2017*). Nevertheless, neural correlations are often the result of low-dimensional, shared variability (*Arieli et al., 1996*; *Ecker et al., 2014*; *Goris et al., 2014*; *Rabinowitz et al., 2015*; *Okun et al., 2015*; *Semedo et al., 2019*), and it is unknown whether extensions of the GLM approach to capture shared-variability (*Archer et al., 2014*; *Zhao and Park, 2017*) can support accurate Bayesian decoding. Similarly, methods based on factor analysis (*Yu et al., 2009*; *Ecker et al., 2014*; *Semedo et al., 2019*) have proven highly effective at modelling neural correlations in large-scale recordings, but it also unknown if they can support Bayesian decoding. Finally, a model class related to GLMs is pairwise-maximum entropy models (*Schneidman et al., 2006*; *Lyamzin et al., 2010*; *Granot-Atedgi et al., 2013*; *Meshulam et al., 2017*), which have been used to investigate semantic clustering of responses in the retinal code (*Ganmor et al., 2015*); yet these models have so-far been limited to population sizes of tens of neurons.

Towards modelling responses and accurate Bayesian decoding in large populations of correlated neurons, we have developed a class of spike-count encoding model based on conditional finite mixtures of multivariate Poisson distributions, which we refer to as CPMs (Conditional Poisson Mixtures). Within neuroscience, Poisson mixtures are widely applied to modelling the spike-count distributions of individual neurons (*Wiener and Richmond, 2003*; *Shidara et al., 2005*; *Goris et al., 2014*; *Taouali et al., 2015*). Outside of neuroscience, mixtures of multivariate Poisson distributions are an established model of multivariate count distributions that effectively capture correlations in count data (*Karlis and Meligkotsidou, 2007*; *Inouye et al., 2017*).

Building on the theory of exponential family distributions (*Wainwright and Jordan, 2008*; *Macke et al., 2011b*), our model extends previous mixture models of multivariate count data in two ways. Firstly, we develop a tractable extension of Poisson mixtures that captures both over- and under-dispersed response variability (i.e. where the response variance is larger or smaller than the mean, respectively) based on Conway-Maxwell Poisson distributions (*Shmueli et al., 2005*; *Stevenson, 2016*). Secondly, we introduce an explicit dependence of the model on a stimulus variable, which allows the model to accurately capture changes in response statistics (including noise correlations) across stimuli. Importantly, the resulting encoding model affords closed-form expressions for both its Fisher information and probability density function and thereby a rigorous quantification of the coding properties of a modelled neural population (*Dayan and Abbott, 2005*). Moreover, the model learns low-dimensional representations of stimulus-driven neural activity, and we show how it provides a parsimonious description of a fundamental property of population codes known as information-limiting correlations (*Moreno-Bote et al., 2014*; *Montijn et al., 2019*; *Bartolo et al., 2020*; *Kafashan et al., 2020*; *Rumyantsev et al., 2020*).

We apply the CPM framework to both synthetic data and recordings from macaque primary visual cortex (V1), and demonstrate that it effectively models responses of populations of hundreds of neurons, captures noise correlations, and supports accurate Bayesian decoding. Ultimately, our model of neural encoding and decoding can be used to quantify coding properties of a neural circuit, such as their efficiency, linearity, or information capacity.

Results

A critical part of our theoretical approach is based on expressing models of interest in exponential family form. An exponential family distribution $p(n)$ over some data n (in our case, neural responses) is defined by the proportionality relation $p(n) \propto e^{\theta \cdot \mathbf{s}(n)} b(n)$, where θ are the so-called natural parameters, $\mathbf{s}(n)$ is a vector-valued function of the data called the sufficient statistic, and $b(n)$ is a scalar-valued function called the base measure (*Wainwright and Jordan, 2008*). The exponential family form allows us to modify and extend existing models in a simple and flexible manner, and

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

to gain analytical insight into the coding properties of our models. We demonstrate our approach with applications to both synthetic data generated by example CPMs, and data recorded in V1 of anaesthetized and awake macaques viewing drifting grating stimuli at different orientations (for details see Materials and methods).

Extended Poisson mixture models capture spike-count variability and covariability

Our first goal is to define a class of models of neural population activity, that model neural activity directly as spike-counts, and that accurately capture single-neuron variability and pairwise covariability. We base our models on Poisson distributions, as they are widely-applied to modelling the trial-to-trial distribution of the number of spikes generated by a neuron (***Dayan and Abbott, 2005***; ***Macke et al., 2011a***). We will also generalize our Poisson models with Conway-Maxwell (CoM) Poisson distributions, because they can capture the broad range of Fano factors (FF; the variance divided by the mean) observed in cortex, in contrast with Poisson distributions for which the FF is always 1 (***Sur et al., 2015***; ***Stevenson, 2016***; ***Chanialidis et al., 2018***).

Mixtures of Poisson distributions are also used to capture complex spike-count distributions in cortex, and allow for over-dispersion (FF>1) (***Shidara et al., 2005***; ***Goris et al., 2014***; ***Taouali et al., 2015***) (Figure 1A). In our case we consider multivariate Poisson mixtures, as they capture covariability in count data as well (see ***Karlis and Meligkotsidou (2007)*** for the general definition). To construct a multivariate Poisson mixture we begin with a product of independent Poisson distributions, one per neuron. We then mix a finite number of such independent Poisson models, to arrive at a multivariate spike-count, finite mixture model (see Materials and methods). Importantly, although each mixture component is a product of independent Poisson distributions, randomly switching between components induces correlations between the neurons (Figure 1B,C). In fact, multivariate Poisson mixtures may model arbitrary pairwise covariability (see Materials and methods, Equation 6). Nevertheless, they are limited because the variance of individual neurons cannot be smaller than the mean, and are thus always over-dispersed (Equation 5, Materials and methods).

To address this limitation, we show how to express multivariate Poisson mixtures in an exponential family form, and then generalize the model with CoM-Poisson distributions. We first note that a multivariate Poisson mixture with d_K components may be expressed as a latent variable model over spike-count vectors $\mathbf{n}$ and latent component-indices k, where $1 \leq k \leq d_K$. In this formulation we denote the kth component distribution by $p(\mathbf{n} \mid k)$, and the probability of realizing (switching to) the kth component by $p(k)$. The mixture model over spike-counts $\mathbf{n}$ is then expressed as the marginal distribution $p(\mathbf{n}) = \sum_{k=1}^{d_K} p(\mathbf{n} \mid k)p(k) = \sum_{k=1}^{d_K} p(\mathbf{n}, k)$, of the joint distribution $p(\mathbf{n}, k)$. Under mild regularity assumptions (see Materials and methods), we may reparameterize this joint distribution in exponential family form as

$$p(\mathbf{n}, k) \propto \frac{e^{\theta_N \cdot \mathbf{n} + \theta_K \cdot \delta(k) + \mathbf{n} \cdot \Theta_{NK} \cdot \delta(k)}}{\prod_{i=1}^{d_N} n_i!}, \tag{1}$$

where θ_N, θ_K, and Θ_{NK} are the natural parameters of $p(\mathbf{n}, k)$, and $\delta(k) = (\delta_2(k), \ldots, \delta_{d_K}(k))$ is the Kronecker delta vector defined by $\delta_j(k) = 1$ if $j = k$, and 0 otherwise.

The exponential family form of a multivariate Poisson mixture represents the first component distribution (i.e. $p(\mathbf{n} \mid k)$ with index $k = 1$) as a baseline distribution, and the other components (where $k > 1$) as modulations of the baseline distribution, and this representation helps us extend multivariate Poisson mixtures. In particular, the first component distribution has natural (baseline) parameters θ_N, and for $k > 1$, the natural parameters of $p(n \mid k)$ are the sum of the baseline parameters θ_N and one row from the matrix of parameters Θ_{NK} (Equation 12, Materials and methods). Because the dimension of θ_N is much smaller than the total number of parameters in a given mixture, the baseline parameters provide a relatively low-dimensional means of affecting all the component distributions of the given mixture, as well as the index probabilities (Figure 1D; see Materials and methods, Equation 11 for how $p(k)$ depends on θ_N).

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

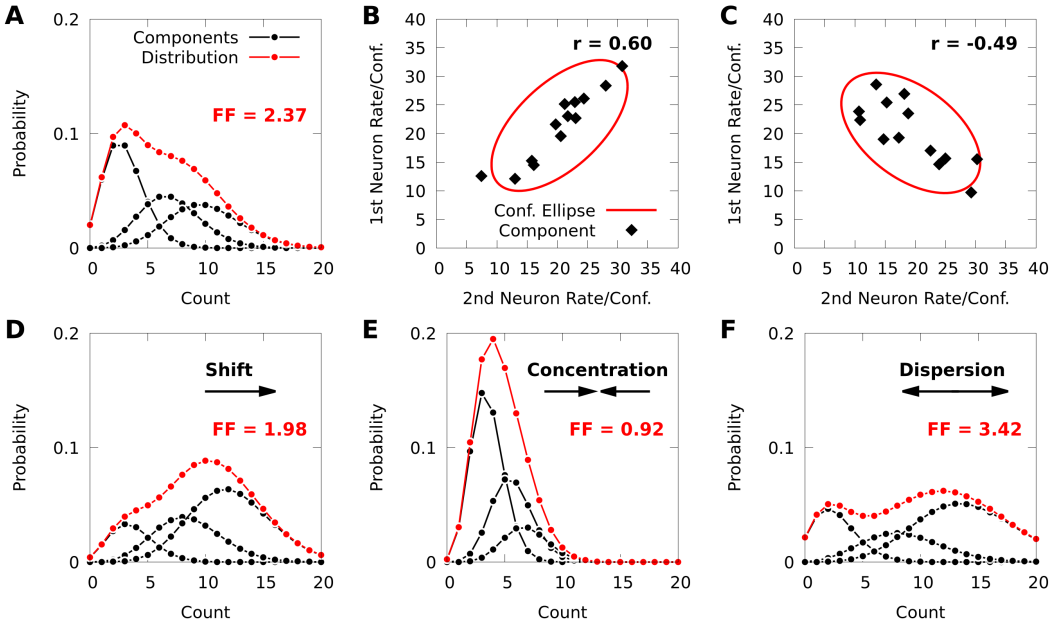

Figure 1. *Poisson mixtures and Conway-Maxwell extensions* **A:** A Poisson mixture distribution (red), defined as the weighted sum of three component Poisson distributions (black; scaled by their weights). FF denotes the Fano Factor (variance over mean) of the mixture. **B,C:** The average spike-count (rate) of the first and second neurons for each of 13 components (black dots) of a bivariate Poisson mixture model, and 68% confidence ellipses for the spike-count covariance of the mixture (red lines; see Equations 5 and 6). The spike-count correlation of each mixture is denoted by r. **D:** Same model as **A**, except we shift the distribution by increasing the baseline rate of the components. **E,F:** Same model as **A**, except we use an additional baseline parameter based on Conway-Maxwell Poisson distributions to concentrate (**E**) or disperse (**F**) the mixture distribution and its components.

We now extend Relation 1 with CoM-Poisson theory, and propose the latent variable exponential family

$$p(\mathbf{n}, k) \propto e^{\theta_N \cdot \mathbf{n} + \theta^*_N \cdot \mathbf{lf}(\mathbf{n}) + \theta_K \cdot \delta(k) + \mathbf{n} \cdot \Theta_{NK} \cdot \delta(k)}, \tag{2}$$

where $\mathbf{lf}(\mathbf{n}) = (\log(n_1!), \ldots, \log(n_{d_N}!))$ is the vector of log-factorials of the individual spike-counts, and θ^*_N are a set of natural parameters based on CoM-Poisson distributions (see Materials and methods). The exponential family form continues to represent the mixture in terms of a baseline distribution, in this case $p(\mathbf{n} \mid k)$ is a product of independent CoM-Poisson distributions, with baseline parameters θ_N and CoM-based parameters θ^*_N. However, whereas the rows of Θ_{NK} modulate θ_N depending on the component index k, the parameters θ^*_N are not modulated, and remain the same for each component distribution (Equation 15, Materials and methods, and see Equation 14 for index-probability formula). For the rest of this paper we refer to models described by Relation 1 as vanilla mixtures, and models described by Relation 2 as CoM-based mixtures.

Due to the addition of the CoM-based parameters, a CoM-based mixture can model under-dispersed (FF < 1) neural activity (Equation 16, Materials and methods). In Figures 1D-F we demonstrate how changing the parameters of the CoM-based mixture can concentrate or disperse both the mixture distribution and its components.

To validate our mixture models, we tested if they capture variability and covariability of V1 population responses to repeated presentations of a grating stimulus with fixed orientation ($d_N = 43$ neurons and $d_T = 355$ repetitions in one awake macaque; $d_N = 70$ and $d_T = 1,200$ in one anaesthetized macaque). We optimized model parameters as described in Materials and methods. The CoM-Poisson mixture accurately captured single-neuron variability (Figure 2A-B, red symbols), including both cases of over-dispersion and under-dispersion. In contrast, the simpler multivariate Poisson mixture (Figure 2A-B, blue symbols) could not accommodate under-dispersion, and also

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

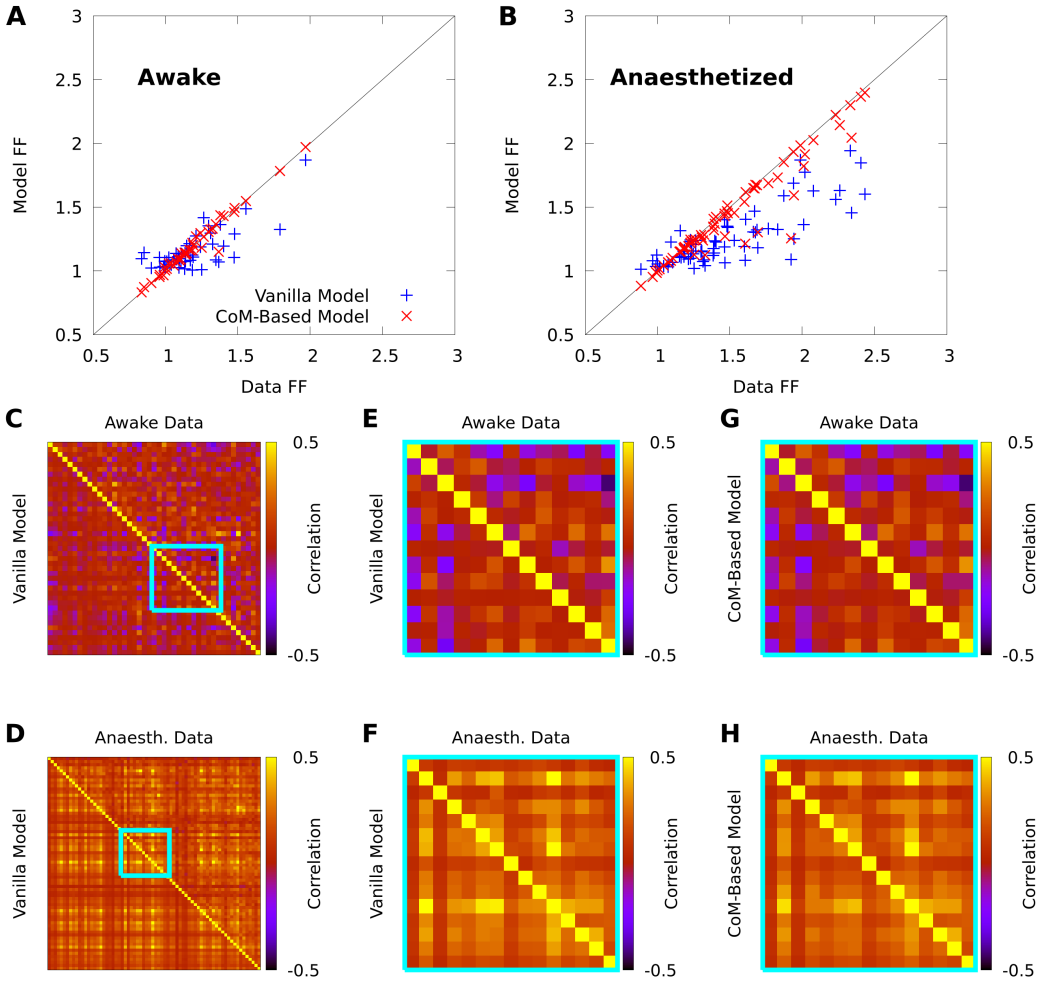

Figure 2. *Capturing neural variability in V1 responses to a single stimulus with CPMs*. We qualitatively compare vanilla Poisson mixtures (Relation 1) and CoM-based mixtures (Relation 2), on awake and anaesthetized V1 responses to stimulus orientation $x = 20°$; both mixtures are defined with $d_K = 4$ components for awake data, and $d_K = 8$ components for anaesthetized data (see Materials and methods for training algorithms). **A,B:** Empirical Fano factors of the awake (**A**) and anaesthetized data (**B**), compared to vanilla (blue) and CoM-based mixtures (red). **C,D:** Empirical correlation matrix (upper right) of awake (**C**) and anaesthetized data (**D**), compared to the correlation matrix of the corresponding vanilla mixtures (lower left). **E,F:** Correlations highlighted in **C** and **D**, respectively. **G,H:** Correlations highlighted in **C** and **D**, except model correlations are from CoM-based mixtures.

had a limited ability to model over-dispersion due to the coupling between the mean and variance (Equation 5). On the other hand, we found that both mixture models were flexible enough to qualitatively capture pairwise noise correlations, both in awake and anaesthetized animals (Fig. 2C-H) (in later sections we quantitatively compare the model performance).

Extended Poisson mixture models capture stimulus-dependent response statistics

So far we have introduced the exponential family theory of vanilla and CoM-based Poisson mixtures, and shown how they capture response variability and covariability for a fixed stimulus. To allow us to study stimulus encoding and decoding, we further extend our mixtures by inducing a dependency of the model parameters on a stimulus. When there are a finite number of stimulus conditions and sufficient data, we may define a stimulus-dependent model with a lookup table, and fit it by fitting a distinct model at each stimulus condition. However, this is inefficient

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

when the amount of data at each stimulus-condition is limited and the stimulus-dependent statistics have structure that is shared across conditions. A notable feature of the exponential family parameterizations in Relations 1 and 2 is that the baseline parameters influence both the index probabilities and all the component distributions of the model. This suggests that by restricting stimulus-dependence to the baseline parameters, we might model rich stimulus-dependent response structure, while bounding the complexity of the model.

In general we refer to any finite mixture of independent Poisson distributions with stimulus-dependent parameters as a conditional Poisson mixture (CPM), and depending on whether the CPM is based on Relations 1 or 2, we refer to it as a vanilla or CoM-based CPM, respectively. Although there are many ways we might induce stimulus-dependence, in this paper we consider two forms of CPM: (i) a maximal CPM, which we implement as a lookup table, such that all the parameters in Relation 1 or 2 depend on the stimulus, and (ii) a minimal CPM, for which we restrict stimulus-dependence to the baseline parameters θ_N, resulting in the CoM-based CPM

$$p(\mathbf{n}, k \mid x) \propto e^{\theta_N(x)\cdot\mathbf{n}+\theta_N^*\cdot\mathbf{lf}(\mathbf{n})+\theta_K\cdot\delta(k)+\mathbf{n}\cdot\mathbf{\Theta}_{NK}\cdot\delta(k)}, \tag{3}$$

where x is the stimulus, and $\theta_N(x)$ are the stimulus-dependent baseline parameters (we may recover a minimal, vanilla CPM by setting $\theta_N^* = -\mathbf{1}$). The tuning curves of the CPM neurons are the average spike-counts (firing rates) of each n_i as a function of the stimulus x, and we refer to $\theta_N(x)$ as the baseline tuning curve parameters, as they define how the firing rates of the baseline CPM distribution (i.e. $p(\mathbf{n} \mid x, k)$ when $k = 1$) depend on x. For $k > 1$, the modulated CPM $p(\mathbf{n} \mid x, k)$ is then a scaled, or "gain-modulated" version of the baseline CPM (see Equations 12 and 15 and the accompanying discussions).

Towards understanding the expressive power of CPMs, we study a minimal, CoM-based CPM with $d_N = 20$ neurons, $d_K = 5$ mixture components, and randomly chosen parameters (see Materials and methods). Moreover, we assume that the stimulus is periodic (e.g. the orientation of a grating), and that the baseline tuning curves have a von Mises shape which is a widely applied model of neural tuning to periodic stimuli (***Herz et al., 2017***). We may achieve such a baseline shape by defining the baseline tuning curve parameters as $\theta_N(x) = \theta_N^0 + \mathbf{\Theta}_{NX} \cdot \mathbf{vm}(x)$, where θ_N^0 and $\mathbf{\Theta}_{NX}$ are the tuning curve parameters, and $\mathbf{vm}(x) = (\cos 2x, \sin 2x)$. Figure 3A shows that the tuning curves of the CPM neurons are approximately bell-shaped, yet many also exhibit significant deviations.

We also study if CPMs can be effectively fit to datasets comparable to those obtained in typical neurophysiology experiments. We generated 200 responses from the CoM-based CPM described above — the ground truth CPM — to each of 10 orientations spread evenly over the half-circle, for a total of 2,000 stimulus-response sample points. We then used this data to fit a CPM with the same number of components. Towards this aim, we derived an approximate expectation-maximization algorithm (EM, a standard choice for training finite mixture models (***McLachlan et al., 2019***)) to optimize model parameters, that also accounts for the stimulus-dependence (see Materials and methods). Figure 3B shows that the tuning curves of the learned CPM are nearly indistinguishable from those of the ground truth CPM (Figure 3B).

To reveal the orientation-dependent latent structure of the model, in Figure 3C we plot the index probability $p(k \mid x)$ for every k as a function of the orientation x. In Figure 3D we show that the orientation-dependent index probabilities of the learned CPM qualitatively match the true index probabilities in Figure 3C. We also note that although the learned CPM does not correctly identify the indices themselves, this has no effect on the performance of the CPM.

The orientation-dependent index-probabilities provide a high-level picture of how the complexity and structure of model correlations varies with the orientation. The vertical dashed lines in Figures 3C-D denote two orientations that yield substantially different index probabilities $p(k \mid x)$. When a large number of index-probabilities are non-zero, the correlation-matrices of the CoM-based CPM can exhibit complex correlations with both negative and positive values (Figure 3E). However, when one index dominates, the correlation structure largely disappears (Figure 3F). In

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

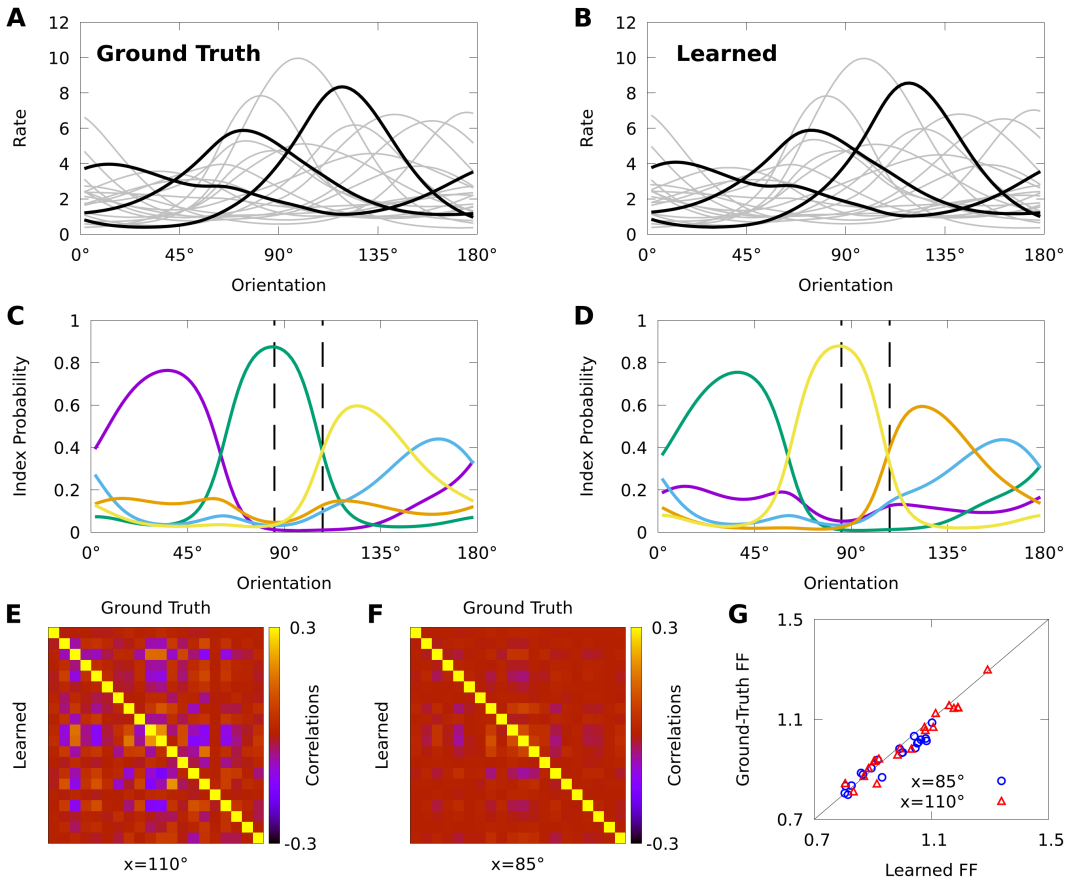

Figure 3. *Recovering a ground truth conditional Poisson mixture (CPM).* We compare a ground truth, CoM-based CPM with 20 neurons, 5 mixture components, von Mises baseline tuning, and randomized parameters to a learned CPM fit to 2,000 samples from the ground truth CPM. **A-B:** Tuning curves of the ground-truth CPM (**A**) and learned CPM (**B**). Three tuning curves are highlighted for effect. **C-D:** The orientation-dependent index probabilities of the ground truth CPM (**C**) and learned CPM (**D**), where colour indicates component index. Dashed lines indicate example stimulus-orientations used in Figures 3C-D.**E-F:** The correlation matrix of the ground truth CPM (upper right), compared to the correlation matrix of the learned CPM (lower left) at stimulus orientations $x = 85°$ (**E**) and $x = 110°$ (**F**). **G:** The FFs of the ground-truth CPM compared to the learned CPM at orientations $x = 85°$ (blue circles) and $x = 110°$ (red triangles).

Figure 3G we show that the FFs also depend on stimulus orientation. Lastly, we find that both the FF and the correlation-matrices of the learned CPM are nearly indistinguishable from the ground-truth CPM (Figure 3E-G).

In summary, our analyses show that CPMs can generate complex, stimulus-dependent response statistics, and that the learned CPM accurately recovers both the statistics and the latent structure of the neural responses from realistic amounts of data.

CPMs effectively model neural responses in macaque V1

A variety of models may be defined within the CPM framework illustrated by Relations 1, 2, and 3. Towards understanding how effectively CPMs can model real data, we compare different variants by their cross-validated log-likelihood. We consider both vanilla and CoM-based variants of each of the following conditional mixtures: (i) maximal CPMs where we learn a distinct mixture for each of d_X stimulus conditions, (ii) minimal CPMs with von Mises baseline tuning curves, and (iii) minimal CPMs with *discrete* baseline tuning curves given by $\theta_N(x) = \theta_N^0 + \mathbf{\Theta}_{NX} \cdot \delta(x)$, where δ is the Kronecker delta vector with $d_X - 1$ elements, and x is the index of the stimulus. In contrast with the von Mises CPM, the discrete CPM makes no assumptions about the form of baseline tuning.

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Encoding Performance

	Awake			Anaesthetized		
	Inf. Gain	d_K	Num. Params.	Inf. Gain	d_K	Num. Params.
Maximal Vanilla	2.30 ± 0.32	5	2,689	8.77 ± 0.71	8	5,103
Maximal CoM	2.44 ± 0.35	5	3,044	9.42 ± 0.70	7	4,464
VM Vanilla	2.01 ± 0.26	45	2,065	8.97 ± 0.70	40	2,979
VM CoM	2.10 ± 0.25	40	1,888	9.38 ± 0.69	35	2,694
Disrete Vanilla	2.25 ± 0.28	40	2,103	9.17 ± 0.70	35	3,044
Disrete CoM	2.35 ± 0.29	30	1,708	9.53 ± 0.68	30	2,689

Table 1. *The encoding performance of CPMs on neural responses in macaque V1.* We apply 10-fold cross-validation to estimate the mean and standard error of the information gain on held-out data, from either awake or anaesthetized macaque V1. We compare maximal CPMs (Maximal), minimal CPMs with von Mises baseline tuning (VM), and minimal CPMs with discrete baseline tuning (Discrete), and for each case we consider either Vanilla or CoM-based variants. For each variant, we indicate the number of CPM components d_K and the corresponding number of model parameters required to achieve peak information gain (cross-validated). For reference, the independent Poisson models use 129 and 210 parameters for the awake and anaesthetized data, respectively.

To provide an interpretable measure of the relative performance of each CPM variant, we measured the difference between the estimated log-likelihood of the given CPM and the log-likelihood of a von Mises-tuned, independent Poisson model, which is a standard model of uncorrelated neural responses to oriented stimuli (***Herz et al., 2017***). We refer to this quantity as the information gain.

Table 1 shows that the CPM variants considered achieve comparable performance, and perform substantially better than the independent Poisson lower bound on both the awake and anaesthetized data. Figure 4 shows that a performance peak emerges smoothly as the model complexity (number of parameters) is increased. In all cases, the CoM-based models outperform their vanilla counterparts, and typically with fewer parameters. The CoM-based discrete CPMs achieve high performance on both datasets. In contrast, von Mises CPMs perform well on the anaesthetized data but more poorly on the awake data, and maximal CPMs exhibit the opposite trend. Nevertheless, von Mises CPMs solve a more difficult statistical problem as they also interpolate between stimulus conditions, and so may still prove relevant even where performance is limited. On the other hand, even though maximal CPMs achieve high performance, they simply do so by replicating the high performance of stimulus-independent mixtures (Figure 2) at each stimulus condition, requiring significantly more parameters than minimal CPMs.

CPMs facilitate accurate and efficient Bayesian decoding of neural responses

To demonstrate that CPMs model the neural code, we must show that CPMs not only capture the features of neural responses, but that these features also encode stimulus-information. Given an encoding model $p(\mathbf{n} \mid x)$ and a response from the model $\mathbf{n}$, we may optimally decode the information in the response about the stimulus x by applying Bayes' rule $p(x \mid \mathbf{n}) \propto p(\mathbf{n} \mid x)p(x)$, where $p(x \mid \mathbf{n})$ is the posterior distribution (the decoded information), and $p(x)$ represents our prior assumptions about the stimulus (***Zemel et al., 1998***). When we do not know the true encoding model, and rather fit a statistical model to stimulus-response data, using the statistical model for Bayesian decoding and analyzing its performance can tell us how well it captures the features of the neural code.

We analyze the performance of Bayesian decoders based on CPMs by quantifying their decoding performance, and comparing the results to other common approaches to decoding. We quantify decoding performance by evaluating the average of the cross-validated log-posterior probability $\log p(x \mid \mathbf{n})$ of the true stimulus value x, for both our awake and anaesthetized V1 datasets. With regards to training the CPMs, we analyze the decoding performance of CPMs that achieved the

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.
Manuscript submitted to eLife

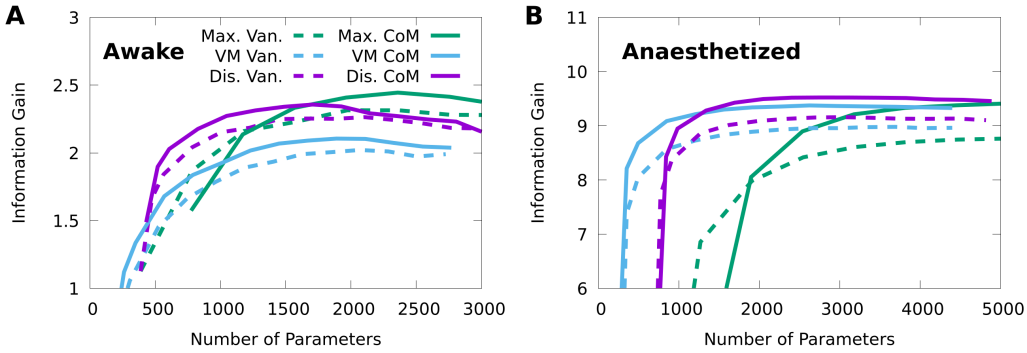

Figure 4. *Finding the optimal number of parameters for CPMs to model neural responses in macaque V1*. 10-fold cross-validation of the information gain given awake V1 data (**A**) and anaesthetized V1 data (**B**), as a function of the number of model parameters, for multiple forms of CPM: maximal CPMs (green); minimal CPMs with von Mises baseline tuning (blue); minimal CPMs with discrete baseline tuning (purple); and for each case we consider either vanilla (dashed lines) or CoM-based (solid lines) variants. Standard errors of the information gain are not depicted to avoid visual clutter, however they are approximately independent of the number of model parameters, and match the values indicated in Table 1.

best *encoding* performance based as indicated in Table 1 and depicted Figure 4, instead of applying distinct procedures for selecting CPMs based on decoding performance. This is because our goal is to understand how well the response features captured by CPMs reflect the neural code, rather than strictly maximizing decoding performance.

In our comparisons we focus on minimal, discrete CPMs as overall they achieved high performance on both datasets (Figure 4). To characterize the importance of neural correlations to Bayesian decoding, we compare our CPMs to the decoding performance of independent Poisson models with discrete tuning (IP). To characterize the optimality of our Bayesian decoders, we also evaluate the performance of linear multiclass decoders (Linear), as well nonlinear multiclass decoders defined as artificial neural networks (ANNs) with two hidden layers and a cross-validated number of hidden units (for details on the training and model selection procedure, see Materials and methods).

Table 2 shows that on the awake data, the performance of the CPMs is statistically indistinguishable from the ANN, and the CPMs and the ANN significantly exceed the performance of both the Linear and IP models. On the anaesthetized data, the minimal CPM approaches the performance of the ANN, and the minimal CPMs and ANN models again exceed the performance of the IP and Linear models. Yet in this case the Linear model is much more competitive, whereas the IP model performs very poorly, possibly because of the larger magnitude of noise correlations in this data. In both cases the ANN requires two orders of magnitude more parameters than the CPMs to achieve its performance gains. In addition, the CoM-based CPM achieves marginally better performance with fewer parameters than the vanilla CPM, indicating that although modelling individual variability is not essential for effective Bayesian decoding, doing so still results in a more parsimonious model of the neural code.

We also consider widely used alternative measures of decoding performance, namely the Fisher information (FI), which is an upper bound on the average precision (inverse variance) of the posterior (***Brunel and Nadal, 1998***), as well as the linear Fisher information (LFI), which is a linear approximation of the FI (***Seriès et al., 2004***) corresponding to the accuracy of the optimal, unbiased linear decoder of the stimulus (***Kanitscheider et al., 2015a***). The FI is especially helpful when the posterior cannot be evaluated directly (such as when it is continuous), and is widely adopted in theoretical (***Abbott and Dayan, 1999***; ***Ecker et al., 2014***; ***Moreno-Bote et al., 2014***; ***Kohn et al., 2016***) and experimental (***Ecker et al., 2011***; ***Rumyantsev et al., 2020***) studies of neural coding. As with other models based on exponential family theory (***Ma et al., 2006***; ***Beck et al., 2011b***; ***Ecker et al., 2016***), the FI of a minimal CPM may be expressed in closed-form, and is equal to its LFI (see Ma-

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

Decoding Performance

	Awake		Anaesthetized	
	Average Log-Post.	Num. Params.	Average Log-Post.	Num. Params.
CoM CPM	-0.206 ± 0.043	1,663	-0.441 ± 0.023	2,689
Vanilla CPM	-0.207 ± 0.039	2,103	-0.448 ± 0.026	3,044
Ind. Poisson	-0.272 ± 0.067	387	-0.967 ± 0.071	630
Linear	-0.256 ± 0.053	352	-0.457 ± 0.019	568
Artificial NN	-0.200 ± 0.032	527,108	-0.426 ± 0.015	408,008

Table 2. *The decoding performance of CPMs on neural responses in macaque V1.* We apply 10-fold cross-validation to estimate the mean and standard error of the average log-posteriors $\log p(x \mid \mathbf{n})$ on held-out data, from either awake or anaesthetized macaque V1. We compare discrete, minimal, CoM-based CPM (CoM. CPM) and vanilla CPM (Vanilla CPM); an independent Poisson model with discrete tuning (IP); a multiclass linear decoder (Linear); and a multiclass nonlinear decoder defined as an artificial neural network with two hidden layers (ANN). The number of CPM components d_K was chosen to achieve on peak information gain in Figure 4. The number of ANN hidden units was chosen based on peak cross-validation performance. In all cases we also indicate the number of model parameters required to achieve the indicated performance.

terials and methods), and therefore minimal CPMs can be used to study FI analytically and obtain model-based estimates of FI from data.

We generated 40 populations of $d_N = 20$ model neurons from the vanilla, minimal, von Mises CPM, with parameters corresponding to the best-fit parameters of 40 random subsets of neurons from our V1 datasets. For each population, we generated 50 responses at each of 10 evenly spaced orientations, for a total of $d_T = 500$ responses per population. We then fit a CPM to each set of 500 responses, and compared the FI of the fit CPM to the ground-truth FI at 50 evenly spaced orientations. Pooled over all populations and orientations, the relative error of the estimated FI was $-12.8\% \pm 18.6\%$ on the awake data and $-9.1\% \pm 22.4\%$ on the anaesthetized data.

The aforementioned measures allow us to assess decoding performance when we do not know the full posterior, however the full posterior is an essential part of probabilistic neural codes (***Pouget et al., 2016***; ***Drugowitsch et al., 2019***). To test whether CPMs can in principle recover full posteriors, we consider a ground truth CPM defined as discrete, CoM-based, minimal CPM with $d_N = 200$ neurons, $d_S = 20$ stimulus-conditions, $d_K = 30$ components, and randomized parameters, and we fit a discrete, CoM-based, minimal CPM with $d_K = 40$ components (chosen with cross-validation) to $d_T = 10,000$ responses from the ground-truth CPM (see Materials and methods). We then compute the average KL-divergence (a fundamental measure of the similarity of two distributions, see ***Cover and Thomas*** (***2006***); ***Amari and Nagaoka*** (***2007***)) of the learned posteriors from the ground-truth posterior over all the $d_T = 10,000$ responses, and find that the average posterior divergence is 0.047 ± 0.007 bits, indicating that on average the learned and ground-truth posteriors are extremely close.

To summarize, CPMs support accurate Bayesian decoding in awake and anaesthetized macaque V1 recordings, and are competitive with nonlinear decoders with two orders of magnitude more parameters. Moreover, CPMs afford closed-form expressions of FI and can interpolate good estimates of FI from modest amounts of data, and thereby support analyses of neural data based on this widely applied theoretical tool. Finally, a CPM fit to the responses of a ground-truth CPM can almost perfectly recover the ground-truth posterior distributions.

Minimal CPMs provide an interpretable latent representation of a fundamental feature of the neural code

Having shown that CPMs can be used to accurately decode stimuli, we next aim to demonstrate that the latent structure of CPMs offers an interpretable representation of a central phenomenon in neural coding known as information-limiting correlations, which are neural correlations that

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

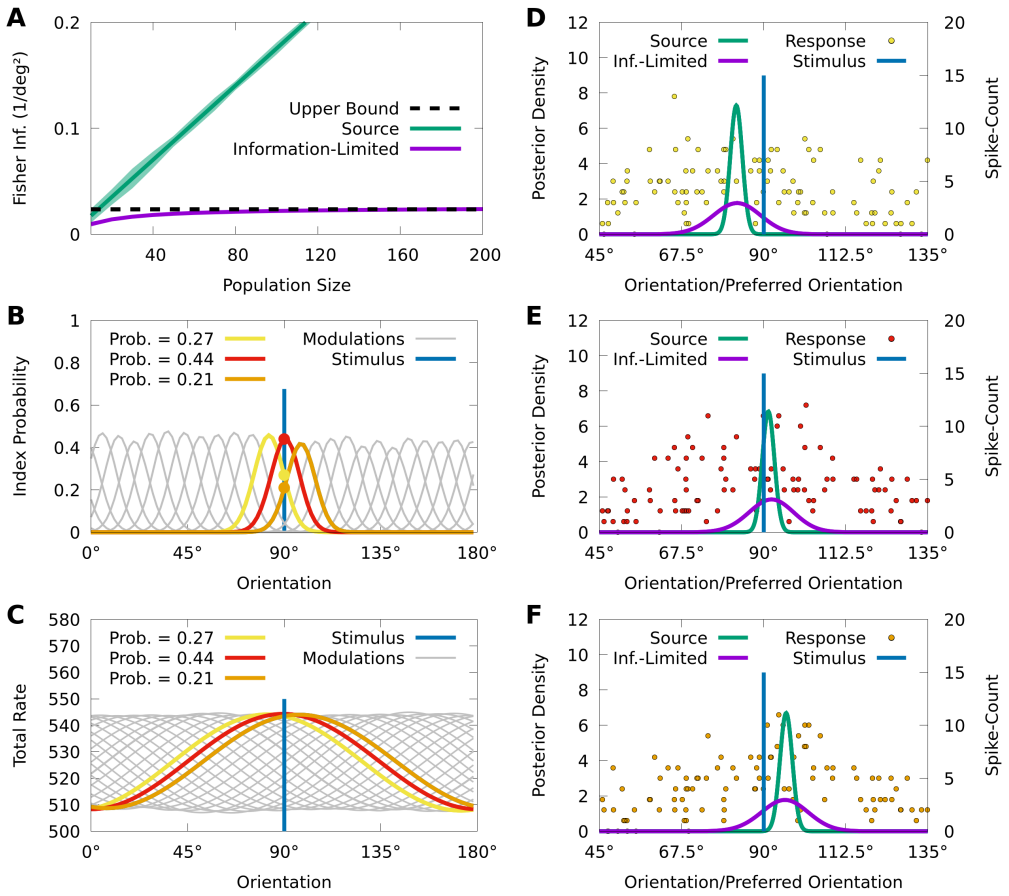

Figure 5. *Fisher information and information-limiting correlations in CPMs.* We consider a von Mises-tuned, independent Poisson source model (green) with $d_K = 200$ neurons, and an information-limited, CoM-based CPM (purple) with $d_K = 25$ components, fit to 10,000 responses of the source-model to stimuli obscured by von Mises noise. In **B**-**F** we consider a stimulus-orientation $x = 90°$ (blue line). **A:** The average (lines) and standard deviation (filled area) of the FI over orientations, for the source (green) and information-limited (purple) models, as a function of random subpopulations, starting with ten neurons, and gradually reintroducing missing neurons. Dashed black line indicates the theoretical upper bound. **B:** The index-probability curves (lines) of the CPM for indices $k > 1$ and the intersection (red, yellow, and orange circles) of the stimulus with three curves (orange, yellow, and orange lines). **C:** The sum of the firing rates of the modulated CPM for all indices $k > 1$ (lines) as a function of orientation, with three modulated CPMs highlighted (red, yellow, and orange lines) corresponding to the highlighted indices in **B**. **D-F:** Three responses from the yellow (**D**; yellow points), red (**E**; red points), and orange modulated CPMs (**F**; orange points) indicated in **C**. For each response we plot the posterior based on the source model (green line) and the information-limited model (purple line).

fundamentally limit stimulus-information in neural circuits (***Moreno-Bote et al., 2014***; ***Montijn et al., 2019***; ***Bartolo et al., 2020***; ***Kafashan et al., 2020***; ***Rumyantsev et al., 2020***). To illustrate this, we generate population responses with limited information, and then fit a CPM to these responses and study the learned latent representation. In particular, we consider a source population of 200 independent Poisson neurons $p(\mathbf{n} \mid s)$ with homogeneous, von Mises tuning curves responding to a noisy stimulus-orientation s, where the noise $p(s \mid x)$ follows a von Mises distribution centred at the true stimulus-orientation x (see Materials and methods). In Figure 5A we show that, as expected, the average FI in the source population about the noisy orientation s grows linearly with the size of randomized subpopulations, whereas the FI about the true orientation x is theoretically bounded by the precision (inverse variance) of the sensory noise.

Even though the neurons in the source model are uncorrelated, sensory noise ensures that

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

the information-limited encoding model $p(\mathbf{n} \mid x) = \int p(\mathbf{n} \mid s)p(s \mid x)ds$ contains information-limiting correlations that bound the FI about x (***Moreno-Bote et al., 2014***; ***Kanitscheider et al., 2015b***). To understand whether and how the latent structure of CPMs captures information-limiting noise correlations, we fit a minimal, von Mises, vanilla CPM with $d_K = 20$ mixture components to $d_T = 10,000$ responses from $p(\mathbf{n} \mid x)$. Figure 5A (purple) shows that the FI of the learned CPM saturates near the precision of the sensory noise, indicating that the learned CPM accurately captures the information-limiting correlations present in $p(\mathbf{n} \mid x)$.

To understand how the learned CPM represents the correlations in $p(\mathbf{n} \mid x)$ we study the relation between the latent modulations and the population activity. Figure 5B shows the index-probabilities of the learned CPM: given the true orientation $x = 90°$, there are 3 modulations with probabilities substantially greater than 0. To provide a high-level picture of how these modulations affect population responses, in Figure 5C we plot the sum of the modulated rates of the population as a function of orientation, and see that each modulation concentrates the tuning of the population around a particular orientation, and that two of the modulations in particular shift the tuning away from the true orientation.

Because there are essentially three modulations that are relevant to the responses of the CPM to the true orientation $x = 90°$, generating a response from the CPM approximately reduces to generating a response from one of the three possible modulated populations. In Figures 5D-F we depict a response to $x = 90°$ from each of the three modulated populations, as well as the optimal posterior based on the learned CPM (purple lines), and a suboptimal posterior based on the source model (i.e. ignoring noise correlations; green lines). We observe that the trial-to-trial variability of the learned CPM results in random shifts of the peak neural activity away from the true orientation, thus fundamentally limiting information. Furthermore, when the response of the population is concentrated at the true orientation (Figure 5E), the suboptimal posterior assigns a high probability to the true orientation, whereas when the responses are biased away from the true orientation (Figures 5D and 5F) the suboptimal posterior assigns nearly 0 probability to the true orientation. This is in contrast to the optimal posterior, which always assigns a significant probability to the true orientation.

In summary, CPMs accurately capture information-limiting correlations, and provide insight into how such correlations can be generated by a simple latent structure.

Discussion

In this paper we introduced a latent variable exponential family formulation of multivariate Poisson mixtures. We showed how this formulation allows us to effectively extend multivariate Poisson mixtures both to capture sub-Poisson variability, and to incorporate stimulus dependence, which we termed Conditional Poisson Mixtures (CPMs). Our analyses and simulations showed that CPMs can be fit efficiently and recover ground truth models in synthetic data, capture a wide range of V1 response statistics in real data, and can be easily inverted to obtain accurate Bayesian decoding that is competitive with nonlinear decoders, while using orders of magnitude less parameters. In addition, we illustrated how the latent structure of CPMs provides an interpretable representation of a fundamental feature of the neural code, e.g. information-limiting correlations.

Our framework is particularly relevant for probabilistic theories of neural coding based on the theory of exponential families (***Beck et al., 2007***), which include theories that address the linearity of Bayesian inference in neural circuits (***Ma et al., 2006***), the role of phenomena such as divisive normalization in neural computation (***Beck et al., 2011a***), Bayesian inference about dynamic stimuli (***Makin et al., 2015***; ***Sokoloski, 2017***), and the metabolic efficiency of neural coding (***Ganguli and Simoncelli, 2014***; ***Yerxa et al., 2020***). These theories have proven difficult to validate quantitatively with neural data due to a lack of statistical models which are both compatible with their exponential family formulation, and can model correlated activity in recordings of large neural populations. Our work suggests that CPMs can overcome these difficulties, and help connect the rich mathematical theory of neural coding with the state-of-the-art in parallel recording technologies.

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

CPMs are not limited to modelling neural responses to stimuli, and can model how arbitrary experimental variables modulate neural variability and covariability. Examples of experimental variables that have measurable effects on neural covariability include the spatial and temporal context around a stimulus (***Snyder et al., 2014***; ***Snow et al., 2016***, ***2017***; ***Festa et al., 2020***), as well as task-variables and the attentional state of the animal (***Maunsell, 2015***; ***Rabinowitz et al., 2015***; ***Kanashiro et al., 2017***; ***Bondy et al., 2018***; ***Ruff and Cohen, 2019***). Each of these variables could be incorporated into a CPM by either replacing the stimulus-variable in our equations, or combining it with the stimulus-variable to construct a CPM with multivariate dependence. This would allow researchers to explore how the stimulus and the experimental variables mutually interact to shape variability and covariability in large populations of neurons.

To understand how this variability and covariability effects neural coding, latent variable models such as CPMs are often applied to extract interpretable features of the neural code from data (***Whiteway and Butts, 2019***). The latent states of a CPM provide a soft classification of neural activity, and we may apply CPMs to model how an experimental variable modulates the class membership of neurons. In the aforementioned studies, models of neural activity yielded predictions of perceptual and behavioural performance. Because CPMs support Bayesian decoding, an appropriate CPM can also make predictions about how a class of neurons is likely to modulate perception and behaviour, and we may then test these predictions with experimental interventions on the neurons themselves (***Panzeri et al., 2017***). In this manner, we believe CPMs could form a critical part of a rigorous, Bayesian framework for "cracking the neural code" in large populations neurons.

In our applications we considered low-dimensional variables, and implemented the stimulus-dependence of the CPM parameters with linear functions. Nevertheless, the stimulus-dependence of a CPM can be implemented by arbitrary parametric functions of high-dimensional variables such as deep neural networks, and CPMs can also incorporate history-dependence via recurrent neural networks. As such, CPMs have the potential to integrate encoding models of higher cortical areas (***Yamins et al., 2014***) with models of the temporal features of the neural code (***Pillow et al., 2008***; ***Park et al., 2014***; ***Runyan et al., 2017***), towards analyzing the neural code in dynamic, correlated neural populations in higher cortex. Outside of neuroscience, high-dimensional count data exists in many fields such as corpus linguistics and genomics (***Inouye et al., 2017***), and researchers who aim to understand how this data depends on history or additional variables could benefit from our techniques.

Materials and methods

Notation

We use capital, bold letters (e.g. $\mathbf{\Theta}$) to indicate matrices; small, bold letters (e.g. $\boldsymbol{\theta}$) to indicate vectors; and regular letters (e.g. θ) to indicate scalars. We use subscript capital letters to indicate the role of a given variable, so that, in Relation 1 for example, $\boldsymbol{\theta}_K$ are the natural parameters that bias the index-probabilities, $\boldsymbol{\theta}_N$ are the baseline natural parameters of the neural firing rates, and $\mathbf{\Theta}_{NK}$ is the matrix of parameters through which the indicies and rates interact.

We denote the ith element of a vector $\boldsymbol{\theta}$ by θ_i, or e.g. of the vector $\boldsymbol{\theta}_K$ by $\theta_{K,i}$. We denote the ith row or jth column of $\mathbf{\Theta}$ by $\boldsymbol{\theta}_i$ or $\boldsymbol{\theta}_j$, respectively, and always state whether we are considering a row or column of the given matrix. When referring to the jth element of a vector $\boldsymbol{\theta}_i$ indexed by i, we write θ_{ij}. Finally, when indexing data points from a sample, or parameters that are tied to individual data points, we use parenthesized, superscript letters, e.g. $x^{(i)}$, or $\boldsymbol{\theta}_N^{(i)}$.

Poisson mixtures and their moments

The following derivations were presented in a more general form in ***Karlis and Meligkotsidou (2007)***, but we present the simpler case here for completeness. A Poisson distribution has the form $p(n;\lambda) = \frac{\lambda^n e^{-\lambda}}{n!}$, where n is the count and λ is the rate (in our case, spike count and firing rate, respectively). We may use a Poisson model to define a distribution over d_N spike counts $\mathbf{n} = (n_1, \ldots, n_{d_N})$

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

by supposing that the neurons generate spikes independently of one another, leading to the independent Poisson model $p(\mathbf{n};\boldsymbol{\lambda}) = \prod_{i=1}^{d_N} p(n_i;\lambda_i)$ with firing rates $\boldsymbol{\lambda} = (\lambda_1,\ldots,\lambda_{d_N})$. Finally, if we consider the d_K rate vectors $\boldsymbol{\lambda}_1,\ldots,\boldsymbol{\lambda}_{d_K}$, and d_K weights $w_1,\ldots,w_{d_K}$, where $0 \leq w_k$ for all k, and $w_1 = 1 - \sum_{k=2}^{d_K} w_k$, we then define a mixture of Poisson distributions as a latent variable model $p(n) = \sum_k p(n \mid k)p(k) = \sum_k p(n,k)$, where $p(n \mid k) = p(n;\boldsymbol{\lambda}_k)$, and $p(k) = w_k$.

The mean μ_i of the ith neuron of a mixture of independent Poisson distributions is

$$\mu_i = \sum_{n_i=0}^{\infty}\sum_{k=1}^{d_K} p(n_i \mid k)p(k)n_i = \sum_{k=1}^{d_K} p(k)\sum_{n_i=0}^{\infty} p(n_i \mid k)n_i = \sum_{k=1}^{d_K} w_k\lambda_{ik}. \tag{4}$$

The variance σ_i^2 of neuron i is

$$\sigma_i^2 = \sum_{n_i=0}^{\infty} p(n_i)n_i^2 - \mu_i^2 = \sum_{k=1}^{d_K} p(k)\sum_{n_i=0}^{\infty} p(n_i \mid k)n_i^2 - \mu_i^2 = \sum_{k=1}^{d_K} p(k)(\sigma_{ik}^2 + \lambda_{ik}^2) - \mu_i^2 = \mu_i + \sum_{k=1}^{d_K} w_k(\lambda_{ik} - \mu_i)^2, \tag{5}$$

where $\sigma_{ik}^2 = \lambda_{ik}$ is the variance of the ith neuron under the kth component distribution, i.e. the variance of $p(n_i \mid k)$, and where $\sum_{n_i=0}^{\infty} p(n_i \mid k)n_i^2 = \sigma_{ik}^2 + \lambda_{ik}^2$, and $\sum_{k=1}^{d_K} w_k\lambda_{ik}^2 - \mu_i^2 = \sum_{k=1}^{d_K} w_k(\lambda_{ik} - \mu_i)^2$ both follow from the fact that a distribution's variance equals the difference between its second moment and squared first moment.

The covariance σ_{ij}^2 between spike-counts n_i and n_j for $i \neq j$ is then

$$\begin{aligned}\sigma_{ij}^2 &= \sum_{n_i=0}^{\infty}\sum_{n_j=0}^{\infty} p(n_i,n_j)(n_i - \mu_i)(n_j - \mu_j) = \sum_{k=1}^{d_K} p(k)\sum_{n_i=0}^{\infty}\sum_{n_j=0}^{\infty} p(n_i,n_j \mid k)(n_i - \mu_i)(n_j - \mu_j) \\ &= \sum_{k=1}^{d_K} p(k)\sum_{n_i=0}^{\infty} p(n_i \mid k)(n_i - \mu_i)\sum_{n_j=0}^{\infty} p(n_j \mid k)(n_j - \mu_j) = \sum_{k=1}^{d_K} w_k(\lambda_{ik} - \mu_i)(\lambda_{jk} - \mu_j).\end{aligned} \tag{6}$$

Observe that if $w_k = \frac{1}{d_K - 1}$, then σ_{ij}^2 is simply the sample covariance between i and j, where the sample is composed of the rate components of the ith and jth neurons. Equation 6 thus implies that Poisson mixtures can model arbitrary covariances. Nevertheless, Equation 5 shows that the variance of individual neurons is restricted to being larger than their means.

Exponential family mixture models

In this section we show that the latent variable form for Poisson mixtures we introduced above is a member of the class of models known as exponential families. An exponential family distribution $p(x)$ over some data x has the form $p(x) = e^{\boldsymbol{\theta}\cdot\mathbf{s}(x) - \psi(\boldsymbol{\theta})}b(x)$, where $\boldsymbol{\theta}$ are the so-called natural parameters, $\mathbf{s}(x)$ is a vector-valued function of the data called the sufficient statistic, $b(x)$ is a scalar-valued function called the base measure, and $\psi(\boldsymbol{\theta}) = \log\int e^{\boldsymbol{\theta}\cdot\mathbf{s}(x)}b(x)dx$ is the log-partition function (***Wainwright and Jordan, 2008***). In the context of Poisson mixture models, we note that an independent Poisson model $p(\mathbf{n};\boldsymbol{\lambda})$ is an exponential family, with natural parameters $\boldsymbol{\theta}_N$ given by $\theta_{N,i} = \log\lambda_i$, base measure $b(\mathbf{n}) = \prod_i n_i!$ and sufficient statistic $\mathbf{s}_N(\mathbf{n}) = \mathbf{n}$, and log-partition function $\psi_N(\boldsymbol{\theta}_N) = \log\sum_i e^{\theta_{N,i}}$. Moreover, the distribution of component indices $p(k)$ (also known as a categorical distribution) also has an exponential family form, with natural parameters $\theta_{K,k} = \log\frac{w_{k+1}}{w_1}$ for $1 \leq k < d_K$, sufficient statistic $\boldsymbol{\delta}(k) = (\delta_2(k),\ldots,\delta_{d_K}(k))$, base measure $b(k) = 1$, and log-partition function $\psi_K(\boldsymbol{\theta}_K) = \log(1 + \sum_{k=1}^{d_K-1} e^{\theta_{K,k}})$. Note that in both cases, the exponential parameters are well-defined only if the rates and weights are strictly greater than 0 — in practice however this is not a significant limitation.

We claim that the joint distribution of a multivariate Poisson mixture model $p(\mathbf{n},k)$ can be reparameterized in the exponential family form

$$p(\mathbf{n},k) = \frac{e^{\boldsymbol{\theta}_N\cdot\mathbf{n} + \boldsymbol{\theta}_K\cdot\boldsymbol{\delta}(k) + \mathbf{n}\cdot\boldsymbol{\Theta}_{NK}\cdot\boldsymbol{\delta}(k) - \psi_{NK}(\boldsymbol{\theta}_N,\boldsymbol{\theta}_K,\boldsymbol{\Theta}_{NK})}}{\prod_i n_i!}, \tag{7}$$

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.
Manuscript submitted to eLife

where $\psi_{NK}(\theta_N, \theta_K, \mathbf{\Theta}_{NK}) = \log \sum_k e^{\theta_k \cdot \delta(k) + \psi_N(\theta_N + \mathbf{\Theta}_{NK} \cdot \delta(k))}$ is the log-partition function of $p(\mathbf{n} \mid k)$. To show this we show how to express the natural parameters θ_N, θ_K, and $\mathbf{\Theta}_{NK}$ as (invertible) functions of the component rate vectors $\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_{d_K}$, and the weights $w_1, \ldots, w_{d_K}$. In particular, we set

$$\theta_N = \log \boldsymbol{\lambda}_1, \tag{8}$$

where log is applied element-wise. Then, for $1 \leq k < d_K$, we set the kth row $\theta_{NK,k}$ of $\mathbf{\Theta}_{NK}$ to

$$\theta_{NK,k} = \log \boldsymbol{\lambda}_{k+1} - \log \boldsymbol{\lambda}_1, \tag{9}$$

and the kth element of θ_K to

$$\theta_{K,k} = \log \frac{w_{k+1}}{w_1} + \psi(\theta_N) - \psi_N(\theta_N + \mathbf{\Theta}_{NK} \cdot \delta(k)). \tag{10}$$

This reparameterization may then be checked by substituting Equations 8, 9, and 10 into Equation 7 to recover the joint distribution of the mixture model $p(\mathbf{n}, k) = p(\mathbf{n} \mid k)p(k) = w_k p(\mathbf{n}; \boldsymbol{\lambda}_k)$; for a more explicit derivation see ***Sokoloski*** (***2019***).

The equation for $p(\mathbf{n}, k)$ ensures that the index-probabilities are given by

$$p(k) = e^{\theta_K \cdot \delta(k) - \psi_{NK}(\theta_N, \theta_K, \mathbf{\Theta}_{NK})} \sum_{\mathbf{n}} \frac{e^{\mathbf{n} \cdot (\theta_N + \mathbf{\Theta}_{NK} \cdot \delta(k))}}{\prod_i n_i!} = e^{\theta_K \cdot \delta(k) - \psi_{NK}(\theta_N, \theta_K, \mathbf{\Theta}_{NK}) + \psi_N(\theta_N + \mathbf{\Theta}_{NK} \cdot \delta(k))}. \tag{11}$$

Consequently, the component distributions in exponential family form are given by

$$p(\mathbf{n} \mid k) = \frac{p(\mathbf{n}, k)}{p(k)} = e^{\mathbf{n} \cdot (\theta_N + \mathbf{\Theta}_{NK} \cdot \delta(k)) - \psi_N(\theta_N + \mathbf{\Theta}_{NK} \cdot \delta(k))}. \tag{12}$$

Observe that $p(\mathbf{n} \mid k)$ is a multivariate Poisson distribution with parameters $\theta_N + \mathbf{\Theta}_{NK} \cdot \delta(k)$, so that for $k > 1$, the parameters are the sum of θ_N and row $k - 1$ of $\mathbf{\Theta}_{NK}$. Because the exponential family parameters are the logarithms of the firing rates of $\mathbf{n}$, each row of $\mathbf{\Theta}_{NK}$ modulates the firing rates of $\mathbf{n}$ multiplicatively. When $\theta_N(x)$ depends on a stimulus and we consider the component distributions $p(\mathbf{n} \mid x, k)$, each row of $\mathbf{\Theta}_{NK}$ then scales the tuning curves of the baseline population (i.e. $(p(\mathbf{n} \mid x, k)$ for $k = 1$); in the neuroscience literature, such scaling factors are typically referred to as gain modulations.

The exponential family form has many advantages. However, it has a less intuitive relationship with the statistics of the model such as the mean and covariance. The most straightforward method to compute these statistics given a model in exponential family form is to first reparameterize it in terms of the weights and component rates, and then evaluate Equations 4, 5, and 6.

CoM-Poisson distributions and their mixtures

Conway-Maxwell (CoM) Poisson distributions decouple the location and shape of count distributions (***Shmueli et al., 2005***; ***Stevenson, 2016***; ***Chanialidis et al., 2018***). A CoM Poisson model has the form $p(n; \lambda, \nu) \propto \left(\frac{\lambda^n}{n!}\right)^\nu$. The floor function $\lfloor \lambda \rfloor$ of the location parameter λ is the mode of the given distribution. With regards to the shape parameter ν, $p(n; \lambda, \nu)$ is a Poisson distribution with rate λ when $\nu = 1$, and is under- or over-dispersed when $\nu > 1$ or $\nu < 1$, respectively. A CoM-Poisson model $p(n; \lambda, \nu)$ is also an exponential family, with natural parameters $\theta_C = (\nu \log \lambda, -\nu)$, sufficient statistic $\mathbf{s}_C(n) = (n, \log n!)$, and base measure $b(n) = 1$. The log-partition function does not have a closed-form expression, but it can be effectively approximated by truncating the series $\sum_{n=0}^{\infty} e^{\mathbf{s}_C(n) \cdot \theta_C}$ (***Shmueli et al., 2005***). More generally, when we consider a product of independent CoM-Poisson distributions, we denote its log-partition function by $\log \psi_C(\theta_N, \theta_N^*) = \sum_{i=1}^{d_N} \sum_{n_i=0}^{\infty} e^{n_i \theta_{N,i} + \log(n_i)! \theta_{N,i}^*}$, where $\theta_{C,i} = (\theta_{N,i}, \theta_{N,i}^*)$ are the parameters of the ith CoM-Poisson distribution. In this case we can also approximate the log-partition function ψ_C by truncating the d_N constituent series $\sum_{n_i=0}^{\infty} e^{n_i \theta_{N,i} + \log(n_i)! \theta_{N,i}^*}$ in parallel.

We define a multivariate CoM-based mixture as

$$p(\mathbf{n}, k) = e^{\theta_N \cdot \mathbf{n} + \theta_N^* \cdot \mathbf{lf}(\mathbf{n}) + \theta_K \cdot \delta(k) + \mathbf{n} \cdot \mathbf{\Theta}_{NK} \cdot \delta(k) - \psi_{CK}(\theta_N, \theta_N^*, \theta_K, \mathbf{\Theta}_{NK})}, \tag{13}$$

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

where $\mathbf{lf}(\mathbf{n}) = (\log(n_1!), \ldots, \log(n_{d_N}!))$ is the vector of log-factorials of the individual spike-counts, and $\psi_{CK}(\theta_N, \theta_N^*, \theta_K, \mathbf{\Theta}_{NK}) = \log \sum_k e^{\theta_k \cdot \delta(k) + \psi_C(\theta_N + \mathbf{\Theta}_{NK} \cdot \delta(k), \theta_N^*)}$ is the log-partition function. This form ensures that the index-probabilities satisfy

$$p(k) = e^{\theta_K \cdot \delta(k) - \psi_{CK}(\theta_N, \theta_N^*, \theta_K, \mathbf{\Theta}_{NK}) + \psi_C(\theta_N + \mathbf{\Theta}_{NK} \cdot \delta(k), \theta_N^*)}, \tag{14}$$

and consequently that each component distribution $p(\mathbf{n} \mid k)$ is a product of independent CoM Poisson distributions given by

$$p(\mathbf{n} \mid k) = e^{\mathbf{n} \cdot (\theta_N + \mathbf{\Theta}_{NK} \cdot \delta(k)) + \theta_N^* \cdot \mathbf{lf}(\mathbf{n}) - \psi_C(\theta_N + \mathbf{\Theta}_{NK} \cdot \delta(k), \theta_N^*)}. \tag{15}$$

Observe that, whereas the parameters $\theta_N + \mathbf{\Theta}_{NK} \cdot \delta(k)$ of $p(\mathbf{n} \mid k)$ depend on the index k, the parameters θ_N^* of $p(\mathbf{n} \mid k)$ are independent of the index and act exclusively as biases. Note as well that when considering a CoM-based, minimal CPM, the modulated populations ($p(\mathbf{n} \mid k, x)$ for $k > 1$) continue to scale the firing rates of the baseline population ($p(\mathbf{n} \mid k, x)$) monotonically, but not in a linear, multiplicative manner.

The moments of a CoM-Poisson distribution are not available in closed-form, yet they can also be effectively approximated through truncation. Given approximate means μ_{ik} and variances σ_{ik}^2 of $p(n_i \mid k)$, we may easily evaluate the means, variances, and covariances of $p(n_i)$. In particular, the mean of n_i is $\mu_i = \sum_{k=1}^{d_K} p(k)\mu_{ik}$, and its variance is

$$\sigma_i^2 = \bar{\sigma}_i^2 + \sum_{k=1}^{d_K} p(k)(\mu_{ik} - \mu_i)^2, \tag{16}$$

where $\bar{\sigma}_i^2 = \sum_{k=1}^{d_K} p(k)\sigma_{ik}^2$. Finally, similarly to Equation 6, the covariance σ_{ij} between n_i and n_j is $\sigma_{ij} = \sum_{k=1}^{d_K} p(k)(\mu_{ik} - \mu_i)(\mu_{jk} - \mu_j)$.

By comparing Equations 5 and 16, we see that the CoM-based mixture may address the limitations on the variances σ_i^2 of the vanilla mixture by setting the average variance $\bar{\sigma}_i^2$ of the components in Equation 16 to be small, while holding the value of the means μ_i fixed, and ensuring that the means of the components μ_{ik} cover a wide range of values to achieve the desired values of σ_i^2 and σ_{ij}. Solving the parameters of a CoM-based mixture for a desired covariance matrix is unfortunately not possible since we lack closed-form expressions for the means and variances. Nevertheless, we may justify the effectiveness of the CoM-based strategy by considering the approximations of the components means and variances $\mu_{ik} \approx \lambda_{ik} + \frac{1}{2\nu_{ik}} - \frac{1}{2}$ and $\sigma_{ik}^2 \approx \frac{\lambda_{ik}}{\nu_{ik}}$, which hold when neither λ_{ik} or ν_{ik} are too small (***Chanialidis et al., 2018***). Based on these approximations, observe that when ν_{ik} is large, σ_{ik}^2 is small, whereas μ_{ik} is more or less unaffected. Therefore, in the regime where these approximations hold, a small value for $\bar{\sigma}_i^2$ can be achieved by reducing the parameters ν_{ik}, without significantly restricting the values of μ_{ik} or μ_i.

Fisher information of a CPM

The Fisher information (FI) of an encoding model $p(\mathbf{n} \mid x)$ with respect to x is $I(x) = \sum_{\mathbf{n}} p(\mathbf{n} \mid x)(\partial_x \log p(\mathbf{n} \mid x))^2$ (***Cover and Thomas, 2006***). With regards to the FI of a CPM,

$$\begin{aligned}\partial_x \log p(\mathbf{n} \mid x) &= \frac{\sum_k \partial_x p(\mathbf{n}, k \mid x)}{p(\mathbf{n} \mid x)} = \frac{\sum_k \partial_x e^{\theta_N(x) \cdot \mathbf{n} + \theta_N^* \cdot \mathbf{lf}(\mathbf{n}) + \theta_K \cdot \delta(k) + \mathbf{n} \cdot \mathbf{\Theta}_{NK} \cdot \delta(k) - \psi_{CK}(\theta_N(x), \theta_N^*, \theta_K, \mathbf{\Theta}_{NK})}}{p(\mathbf{n} \mid x)} \\ &= \partial_x(\theta_N(x) \cdot \mathbf{n} - \psi_{CK}(\theta_N(x), \theta_N^*, \theta_K, \mathbf{\Theta}_{NK})) \frac{\sum_k p(\mathbf{n}, k \mid x)}{p(\mathbf{n} \mid x)} = \partial_x \theta_N(x) \cdot (\mathbf{n} - \mu_N(x)),\end{aligned}$$

where $\partial_x \psi_{CK}(\theta_N(x), \theta_N^*, \theta_K, \mathbf{\Theta}_{NK}) = \mu_N(x) \cdot \partial_x \theta_N(x)$ follows from the chain rule and properties of the log-partition function (***Wainwright and Jordan, 2008***). Therefore

$$I(x) = \sum_{\mathbf{n}} p(\mathbf{n} \mid x)(\partial_x \theta_N(x) \cdot (\mathbf{n} - \mu_N(x)))^2 = \partial_x \theta_N(x) \cdot \mathbf{\Sigma}_N(x) \cdot \partial_x \theta_N(x),$$

where $\mathbf{\Sigma}_N(x)$ is the covariance matrix of $p(\mathbf{n} \mid x)$. Moreover, because $\partial_x \theta_N(x) = \mathbf{\Sigma}_N^{-1}(x) \cdot \partial_x \mu(x)$ (***Wainwright and Jordan, 2008***), the FI of a CPM may also be expressed as $I(x) = \partial_x \mu_N(x) \cdot \mathbf{\Sigma}_N^{-1}(x) \cdot \partial_x \mu_N(x)$, which is the linear Fisher information (***Beck et al., 2011b***).

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

Note that when calculating the FI or other quantities based on the covariance matrix, vanilla CPMs have the advantage that their covariance matrices tend to have large diagonal elements and are thus inherently well-conditioned. Because decoding performance is not significantly different between vanilla and CoM-based CPMs (see Table 2), vanilla CPMs may be preferable when well-conditioned covariance matrices are critical. Nevertheless, the covariance matrices of CoM-based mixtures can be made well-conditioned by applying standard techniques.

Expectation-Maximization for CPMs

Expectation-maximization (EM) is an algorithm that maximizes the likelihood of a latent variable model given data by iterating two steps: generating model-based expectations of the latent variables, and maximizing the complete log-likelihood of the model given the data and latent expectations. Although the maximization step optimizes the *complete* log-likelihood, each iteration of EM is guaranteed to increase the *data* log-likelihood as well (***Neal and Hinton, 1998***).

EM is arguably the most widely-applied algorithm for fitting finite mixture models (***McLachlan et al., 2019***). As a form of latent variable exponential family, the expectation step for a finite mixture model reduces to computing average sufficient statistics, and the maximization step is a convex optimization problem (***Wainwright and Jordan, 2008***). In general, the average sufficient statistics, or mean parameters, correspond to (are dual to) the natural parameters of an exponential family, and where we denote natural parameters with θ, we denote their corresponding mean parameters with η.

Suppose we are given a dataset $(\mathbf{n}^{(1)}, \dots, \mathbf{n}^{(d_T)})$ of neural spike-counts, and a CoM-based mixture model with natural parameters θ_N, θ^*_N, θ_K, and $\mathbf{\Theta}_{NK}$ (see Equation 13). The expectation step for this model reduces to computing the data-dependent mean parameters $\boldsymbol{\eta}^{(i)}_K$ given by

$$\theta^{(i)}_K = \theta_K + \mathbf{n}^{(i)} \cdot \mathbf{\Theta}_{NK}, \qquad \eta^{(i)}_{K,k} = \frac{e^{\theta^{(i)}_{K,k}}}{1 + \sum_l e^{\theta^{(i)}_{K,l}}},$$

for all $0 < i \leq d_T$. The mean parameters $\boldsymbol{\eta}^{(i)}_K$ are the averages of the sufficient statistic $\boldsymbol{\delta}_k(k)$ under the distribution $p(k \mid \mathbf{n}^{(i)})$, and are what we use to complete the log-likelihood since we do not observe k.

Given $\boldsymbol{\eta}^{(i)}_K$, the maximization step of a CoM-based mixture thus reduces to maximizing the complete log-likelihood $\sum_{i=1}^{d_T} \mathcal{L}(\theta_K, \theta_N, \theta^*_N, \mathbf{\Theta}_{NK}, \boldsymbol{\eta}^{(i)}_K, \mathbf{n}^{(i)})$, where we substitute $\boldsymbol{\eta}^{(i)}_K$ into the place of $\boldsymbol{\delta}(k)$ in Equation 13, such that

$$\mathcal{L}(\theta_K, \theta_N, \theta^*_N, \mathbf{\Theta}_{NK}, \boldsymbol{\eta}^{(i)}_K, \mathbf{n}^{(i)}) = \theta_N \cdot \mathbf{n}^{(i)} + \theta^*_N \cdot \mathbf{lf}(\mathbf{n}^{(i)}) + \theta_K \cdot \boldsymbol{\eta}^{(i)}_K + \mathbf{n}^{(i)} \cdot \mathbf{\Theta}_{NK} \cdot \boldsymbol{\eta}^{(i)}_K - \psi_{CK}(\theta_N, \theta^*_N, \theta_K, \mathbf{\Theta}_{NK}).$$

This objective may be maximized in closed-form for a vanilla Poisson mixture (***Karlis and Meligkotsidou, 2007***), but this is not the case when the model has CoM-Poisson shape parameters or depends on the stimulus. Nevertheless, solving the resulting maximization step is still a convex optimization problem (***Wainwright and Jordan, 2008***), and may be approximately solved with gradient ascent. Doing so requires that we first compute the mean parameters $\boldsymbol{\eta}_N$, $\boldsymbol{\eta}^*_N$, $\boldsymbol{\eta}_K$, and $\mathbf{H}_{NK}$ that are dual to θ_N, θ^*_N, θ_K, and $\mathbf{\Theta}_{NK}$, respectively.

We compute the mean parameters by evaluating

$$\theta^\dagger_{K,k} = \theta_{K,k} + \psi_C(\theta_N + \mathbf{\Theta}_{NK} \cdot \boldsymbol{\delta}(k), \theta^*_N) - \psi(\theta_N), \qquad \eta_{K,k} = \frac{e^{\theta^\dagger_{K,k}}}{1 + \sum_{k=1}^{d_K - 1} e^{\theta^\dagger_{K,k}}}, \qquad \mu_{jk} = \sum_{n_j=0}^{\infty} n_j \; p(n_j \mid k),$$

$$\eta^*_{N,j} = \sum_{k=1}^{d_K} p(k) \sum_{n_j=0}^{\infty} \log n_j! \; p(n_j \mid k), \qquad \eta_{N,j} = \sum_{k=1}^{d_K} p(k) \mu_{jk}, \qquad \eta_{NK,jk} = \eta_{K,k} \mu_{j(k+1)},$$

where $\eta_{K,k}$ is the kth element of $\boldsymbol{\eta}_K$, $\eta_{N,j}$ is the jth element of $\boldsymbol{\eta}_N$, $\eta^*_{N,j}$ is the jth element of $\boldsymbol{\eta}^*_N$, and $\eta_{NK,jk}$ is the jth element of the kth column of $\mathbf{H}_{NK}$. Note as well that we truncate the series

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

$\sum_{n_j} n_j\ p(n_j \mid k)$ and $\sum_{n_j} \log n_j!\ p(n_j \mid k)$ to approximate μ_{jk} and $\eta^*_{N,j}$. Given these mean parameters, we may then express the gradients of $\mathcal{L}^{(i)} = \mathcal{L}(\theta_K, \theta_N, \theta^*_N, \mathbf{\Theta}_{NK}, \eta_{K,i}, \mathbf{n}^{(i)})$ as

$$\partial_{\theta_N} \mathcal{L}^{(i)} = \mathbf{n}^{(i)} - \eta_N, \qquad \partial_{\theta^*_N} \mathcal{L}^{(i)} = \mathbf{lf}(\mathbf{n}^{(i)}) - \eta^*_N,$$
$$\partial_{\theta_K} \mathcal{L}^{(i)} = \eta^{(i)}_K - \eta_K, \qquad \partial_{\mathbf{\Theta}_{NK}} \mathcal{L}^{(i)} = \mathbf{n}^{(i)} \otimes \eta^{(i)}_K - \mathbf{H}_{NK},$$

where $\otimes$ is the outer product operator, and where the second term in each equation follows from the fact that the derivative of ψ_{CK} with respect to θ_N, θ^*_N, θ_K, or $\mathbf{\Theta}_{NK}$ yields the dual parameters η_N, η^*_N, η_K, and $\mathbf{H}_{NK}$, respectively. By ascending the gradients of $\sum_{i=1}^{d_T} \mathcal{L}^{(i)}$ until convergence, we approximate a single iteration of the EM algorithm for a CoM-based mixture.

Finally, if our dataset $((\mathbf{n}^{(1)}, x^{(1)}), \ldots, (\mathbf{n}^{(d_T)}, x^{(d_T)}))$ includes stimuli x, and the parameters θ_N depend on the stimulus, then the gradients of the parameters of θ_N must also be computed. For a von Mises CPM where $\theta_N(x) = \theta^0_N + \mathbf{\Theta}_{NX} \cdot \mathbf{vm}(x)$, the gradients are given by

$$\partial_{\theta^0_N} \mathcal{L}^{(i)} = \partial_{\theta^{(i)}_N} \mathcal{L}^{(i)}, \qquad \partial_{\mathbf{\Theta}_{NX}} \mathcal{L}^{(i)} = \partial_{\theta^{(i)}_N} \mathcal{L}^{(i)} \otimes \mathbf{vm}(x^{(i)}),$$

where $\theta^{(i)}_N = \theta_N(x^{(i)})$ is the output of θ_N at $x^{(i)}$. Although in this paper we restrict our applications to Von Mises or discrete tuning curves for 1-dimensional stimuli, this formalism can be readily extended to the case where the baseline tuning curve parameters $\theta_N(x)$ are a generic nonlinear function of the stimulus, represented by a deep neural network. Then, the gradients of the parameters of θ_N can be computed through backpropagation, and $\partial_{\theta^{(i)}_N} \mathcal{L}^{(i)}$ is the error that must be backpropagated through the network to compute the gradients.

CPM initialization and training procedures

To fit a CPM to a dataset $((\mathbf{n}^{(1)}, x^{(1)}), \ldots, (\mathbf{n}^{(d_T)}, x^{(d_T)}))$, we first initialize the CPM and then optimize its parameters with our previously described EM algorithm. Naturally, initialization depends on exactly which form of CPM we consider, but in general we first initialize the baseline parameters θ_N, then add the categorical parameters θ_K and mixture component parameters $\mathbf{\Theta}_{NK}$. When training CoM-based CPMs we always first train a vanilla CPM, and so the initialization procedure remains the same for vanilla and CoM-based models.

To initialize a minimal, von Mises CPM with d_N neurons, we first fit d_N independent, von Mises-tuned neurons by maximizing the log-likelihood $\sum_{i=1}^{d_T} \log p(\mathbf{n}^{(i)} \mid x^{(i)})$ of $\theta_N(x) = \theta^0_N + \mathbf{\Theta}_{NX} \cdot \mathbf{vm}(x)$. This is a convex optimization problem and so can be easily solved by gradient ascent, in particular by following the gradients

$$\partial_{\theta^0_N} \sum_{i=1}^{d_T} \log p(\mathbf{n}^{(i)} \mid x^{(i)}) = \sum_{i=1}^{d_T} \mathbf{n}^{(i)} - \log(\theta_N(x^{(i)})),$$
$$\partial_{\mathbf{\Theta}_{NX}} \sum_{i=1}^{d_T} \log p(\mathbf{n}^{(i)} \mid x^{(i)}) = \sum_{i=1}^{d_T} \log(\mathbf{n}^{(i)} - \log \theta_N(x^{(i)})) \otimes \mathbf{vm}(x^{(i)}),$$

to convergence. For both discrete and maximal CPMs, where there are d_X distinct stimuli, we initialize $\theta_N(x) = \theta^0_N + \mathbf{\Theta}_{NX} \cdot \delta(x)$ by computing the average rate vector at each stimulus-condition and creating a lookup table for these rate vectors. Formally, where x_l is the lth stimulus value for $0 < l \leq d_X$, we may express the lth rate vector as $\boldsymbol{\lambda}_l = \frac{1}{\sum_{i=1}^{d_T} \delta(x_l, x^{(i)})} \sum_{i=1}^{d_T} \delta(x_l, x^{(i)}) \mathbf{n}^{(i)}$, where $\delta(x_l, x^{(i)})$ is 1 when $x_l = x^{(i)}$, and 0 otherwise. We then construct a lookup table for these rate vectors in exponential family form by setting $\theta^0_N = \log \boldsymbol{\lambda}_1$, and by setting the lth row $\theta_{NX,l}$ of $\mathbf{\Theta}_{NX}$ to $\theta_{NX,l} = \log \boldsymbol{\lambda}_{l+1} - \log \boldsymbol{\lambda}_1$.

In general we initialize the parameters θ_K by sampling the weights $w_1, \ldots, w_{d_K}$ of a categorical distribution from a Dirichlet distribution with a constant concentration of 2, and converting the weights into the natural parameters of a categorical distribution θ_K. For discrete and maximal CPMs we initialize the modulations $\mathbf{\Theta}_{NK}$ by generating each element of $\mathbf{\Theta}_{NK}$ from a uniform distribution over the range $[-0.0001, 0.0001]$. For von Mises CPMs we initialize each row $\theta_{NK,k}$ of

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

$\mathbf{\Theta}_{NK}$ as shifted sinusoidal functions of the preferred stimuli of the independent von Mises neurons. That is, given θ_N^0 and $\mathbf{\Theta}_{NX}$, we compute the preferred stimulus of the ith neuron given by $\rho_i = \text{atan2}(\theta_N^0 + \theta_{NX,i})$, where $\theta_{NX,i}$ is the ith row of $\mathbf{\Theta}_{NX}$. We then set the ith element $\theta_{NK,k,i}$ of $\theta_{NK,k}$ to $\theta_{NK,k,i} = 0.2\sin(\rho_i + \frac{k}{360}^\circ)$. Initializing von Mises CPMs in this way ensures that each modulation has a unique peak as a function of preferred stimuli, which helps differentiate the modulations from each other, and in our experience improves training speed.

With regards to training, the expectation step in our EM algorithm may be computed directly, and so the only challenge is solving the maximization step. Although the optimal solution strategy depends on the details of the model and data in question, in the context of this paper we settled on a strategy that is sufficient for all simulations we perform. For each model we perform a total of $d_I = 500$ EM iterations, and for each maximization step we take $d_S = 100$ gradient ascent steps with the Adam gradient ascent algorithm (***Kingma and Ba, 2014***) with the default momentum parameters (see ***Kingma and Ba*** (***2014***)). We restart the Adam algorithm at each iteration of EM and gradually reduce the learning rate. Where $\epsilon^+ = 0.002$ and $\epsilon^- = 0.0005$ are the initial and final learning rates, we set the learning rate ϵ_t at EM iteration t to

$$\epsilon_t = \exp\left(\frac{(d_I - 1 - t)\log(\epsilon^+) + t\log(\epsilon^-)}{d_I - 1}\right),$$

where we assume t starts at 0 and ends at $d_I - 1$.

Because we must evaluate large numbers of truncated series when working with CoM-based CPMs, training times are typically one to two orders of magnitude greater. To minimize training time of CoM-based CPMs over the d_I EM iterations, we therefore first train a vanilla CPM for $0.8d_I$ iterations. We then equate the parameters θ_N, θ_K, and $\mathbf{\Theta}_{NK}$ of the vanilla CPM (see Equation 7) with a CoM-based CPM (see Equation 13) and set $\theta_N^* = -\mathbf{1}$, which ensures that resulting CoM-based model has the same density function $p(\mathbf{n}, k \mid x)$ as the original vanilla model. We then train the CoM-based CPM for $0.2d_I$ iterations. We found this strategy results in practically no performance loss, while greatly reducing training time.

CPM parameter selection for simulations

In the section Extended Poisson mixture models capture stimulus-dependent response statistics and the section CPMs facilitate accurate and efficient Bayesian decoding of neural responses we considered CoM-based, minimal CPMs with randomized parameters $\theta_N(x)$, θ_N^*, θ_K, and $\mathbf{\Theta}_{NK}$, which for simplicity we refer to as models 1 and 2, respectively. We construct randomized CPMs piece by piece, in a similar fashion to our initialization procedure.

Firstly, where d_N is the number of neurons, we tile their preferred stimuli ρ_i over the circle such that $\rho_i = \frac{i}{d_N}360°$. We then generate the concentration κ_i and gain γ_i of the ith neuron by sampling from normal distributions in log-space, such that $\log\kappa_i \sim \text{N}(-0.1, 0.2)$, and $\log\gamma_i \sim \text{N}(0.2, 0.1)$. Finally, for von Mises baseline tuning curves $\theta_N(x) = \theta_N^0 + \mathbf{\Theta}_{NX} \cdot \mathbf{vm}(x)$, we set each row $\theta_{NX,i}$ of $\mathbf{\Theta}_{NX}$ to $\theta_{NX,i} = (\kappa_i\cos\rho_i, \kappa_i\sin\rho_i)$, and each element $\theta_{N,i}^0$ of θ_N^0 to $\theta_{N,i}^0 = \log\gamma_i - \psi_X(\theta_{NX,i})$, where ψ_X is the logarithm of the modified Bessel function of order 0, which is the log-partition function of the von Mises distribution.

We then set $\theta_K = \mathbf{0}$, and generated each element $\theta_{NK,i,k}$ of the modulation matrix θ_{NK} in the same matter as the gains, such that $\theta_{NK,i,k} \sim \text{N}(0.2, 0.1)$. Finally, to generate random CoM-based parameters we generate each element $\theta_{N,i}^*$ of θ_N^* from a uniform distribution, such that $\theta_{N,i}^* \sim \text{U}(-1.5, -0.8)$.

Model 2 entails two more steps. Firstly, when sampling from larger populations of neurons, single modulations often dominate the model activity around certain stimulus values. To suppress this we consider the natural parameters $\theta_K^0(x)$ of $p(k \mid x)$ (see Equation 14), and compute the maximum value of these natural parameters over the range of stimuli $\theta_{K,k}^+ = \max_x\{\theta_{K,k}^0(x)\}$. We then set each element $\theta_{K,k}$ of the parameters θ_K of the CPM to $\theta_{K,k} = \bar{\theta}_K^+ - \theta_{K,k}^+$, where $\bar{\theta}_K^+ = \sum_{i=1}^{d_K}\frac{\theta_{K,k}}{d_K}$, which helps ensure that multiple modulations are active at any given x. Finally, since model 2

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

is a discrete CPM, we replace the von Mises baseline tuning curves with discrete baseline tuning curves, by evaluating $\theta_N^0 + \mathbf{\Theta}_{NX} \cdot \mathbf{vm}(x)$ at each of the d_X valid stimulus-conditions, and assemble the resulting collection of natural parameters into a lookup table in the manner we described in our initialization procedures.

Decoding models

When constructing a Bayesian decoder for discrete stimuli, we first estimate the prior $p(x)$ by computing the relative frequency of stimulus presentations in the training data. For the given encoding model, we then evaluate $p(\mathbf{n} \mid x)$ at each stimulus condition, and then compute the posterior $p(x \mid \mathbf{n}) \propto p(\mathbf{n} \mid x)p(x)$ by brute-force normalization of $p(\mathbf{n} \mid x)p(x)$. When training the encoding model used for our Bayesian encoders, we only trained them to maximize encoding performance as previously described, and not to maximize decoding performance.

We considered two decoding models, namely the linear network and the artificial neural network (ANN) with sigmoid activation functions. In both cases the input of the network was a neural response vector, and the output the natural parameters θ_X of a categorical distribution. The form of the linear network was $\theta_X(\mathbf{n}) = \theta_X + \mathbf{\Theta}_{XN} \cdot \mathbf{n}$, and is otherwise fully determined by the structure of the data. For the ANN on the other hand, we had to choose both the number of hidden layers, and the number of neurons per hidden layer. We cross-validated the performance of both 1 and 2 hidden layer models, over a range of sizes from 100 to 2000 neurons. We found the performance of the networks with 2 hidden layers generally exceeded that of those with 1 hidden layer, and that 700 and 600 hidden neurons was optimal for the awake and anaesthetized networks, respectively.

Given a dataset $((\mathbf{n}^{(1)}, x^{(1)}), \ldots, (\mathbf{n}^{(d_T)}, x^{(d_T)}))$, we optimized the linear network and the ANN by maximizing $\sum_{i=1}^{d_T} \log p(x^{(i)} \mid \mathbf{n}^{(i)})$ via stochastic gradient ascent. We again used the Adam optimizer with default momentum parameters, and used a fixed learning rate of 0.0003, and randomly divided the dataset into minibatches of 500 data points. We also used early stopping, where for each fold of our 10-fold cross-validation simulation, we partitioned the dataset into 80% training data, 10% test data, and 10% validation data, and stopped the simulation when performance on the test data declined from epoch to epoch.

Experimental design

Throughout this paper we demonstrate our methods on two sets of parallel response recordings in macaque primary visual cortex (V1). The stimuli were drifting full contrast gratings at 9 distinct orientations spread evenly over the half-circle from 0° to 180° (2° diameter, 2 cycles per degree, 2.5 Hz drift rate). Stimuli were generated with custom software (EXPO by P. Lennie) and displayed on a cathode ray tube monitor (Hewlett Packard p1230; 1024×768 pixels, with ~ 40 cd/m^2 mean luminance and 100 Hz frame rate) viewed at a distance of 110 cm (for anaesthetized dataset) or 60 cm (for awake dataset). Grating orientations were randomly interleaved, each presented for 70 ms (for anaesthetized dataset) or 150 ms (for awake dataset), separated by a uniform gray screen (blank stimulus) for the same duration.

For each electrode, we extracted waveform signals (sampled at 30 kHz) whenever the extracellular voltage exceeded a user defined threshold (typically 5x the root mean square signal on each channel). We then sorted waveforms manually using the Plexon Offline Sorter, and isolated both single and multi-unit clusters, here both referred to as neurons. We computed spike counts in a fixed window with length equal to the stimulus duration, shifted by 50 ms after stimulus onset. We excluded from further analyses all neurons that were not driven by any stimulus above baseline + 3std.

In the first dataset the monkey was awake, and there were $d_T = 3168$ trials of the responses of 72 neurons; due to the presence of cross-talk between a small subset of electrodes, we removed all pairs of neurons in the dataset that exhibited correlations greater than 0.5, which left $d_N = 43$ neurons in the dataset. We refer to this dataset as the awake V1 dataset. After familiarization with the restraining chair, headpost surgery, and postoperative recovery time (methods and protocols

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

described in ***Festa et al.*** (***2020***)), the animal was trained to fixate in a $1.3^\circ \times 1.3^\circ$ window. Eye position was monitored with a high-speed infrared camera (Eyelink, 1000 Hz). A second surgery was performed over V1 to implant a 96 channel microelectrode array into V1 (electrode length 1 mm). After postoperative recovery, the spatial receptive fields of the sampled neurons were mapped by presenting small patches of drifting full contrast gratings (0.5° diameter; 4 orientations, 1 cycle per degree, 3 Hz drift rate, 250 ms presentation) at 25 distinct positions spanning a $3^\circ \times 4^\circ$ region of visual space. Subsequent stimuli were centered in the aggregate receptive field of the recorded units.

In the second dataset the monkey was anaesthetized and there were $d_T = 10,800$ trials of the responses of $d_N = 70$ neurons; we refer to this dataset as the anaesthetized V1 dataset. The protocol and general methods employed for the anaesthetized experiment have been described previously (***Smith and Kohn, 2008***). In short, anaesthesia was induced with ketamine (10 mg/kg) and maintained during surgery with isoflurane (1.5–2.5% in 95% O2), switching to sufentanil (6–18 μg/kg per h, adjusted as needed) during recordings. Eye movements were reduced using vecuronium bromide (0.15 mg/kg per h). Temperature was maintained in the 36 –37 C$^\circ$ range, and relevant vital signs (EEG, ECG, blood pressure, end-tidal PCO2, temperature, and lung pressure) were monitored continuously to ensure sufficient level of anaesthesia and well-being. A 10×10 multielectrode array (400 μm spacing, 1 mm length) was implanted into the upper layers of primary visual cortex, at a depth of 0.6-0.8 mm.

All procedures were approved by the Albert Einstein College of Medicine and followed the guidelines in the United States Public Health Service Guide for the Care and Use of Laboratory Animals.

Acknowledgments

We would like to thank all the members of the labs of Ruben Coen-Cagli and Adam Kohn for their regular feedback and support.

Competing interests

The authors declare they have no competing interests.

References

Abbott LF, Dayan P. The Effect of Correlated Variability on the Accuracy of a Population Code. Neural computation. 1999; 11(1):91–101.

Amari Si, Nagaoka H. Methods of Information Geometry, vol. 191. American Mathematical Soc.; 2007.

Archer EW, Koster U, Pillow JW, Macke JH. Low-Dimensional Models of Neural Population Activity in Sensory Cortical Circuits. In: Ghahramani Z, Welling M, Cortes C, Lawrence ND, Weinberger KQ, editors. *Advances in Neural Information Processing Systems 27* Curran Associates, Inc.; 2014.p. 343–351.

Arieli A, Sterkin A, Grinvald A, Aertsen A. Dynamics of Ongoing Activity: Explanation of the Large Variability in Evoked Cortical Responses. Science. 1996 Sep; 273(5283):1868–1871.

Bartolo R, Saunders RC, Mitz AR, Averbeck BB. Information-Limiting Correlations in Large Neural Populations. Journal of Neuroscience. 2020 Feb; 40(8):1668–1678.

Beck J, Ma WJ, Latham PE, Pouget A. Probabilistic Population Codes and the Exponential Family of Distributions. Progress in Brain Research. 2007; 165:509–519.

Beck J, Latham P, Pouget A. Marginalization in Neural Circuits with Divisive Normalization. The Journal of Neuroscience. 2011; 31(43):15310–15319.

Beck J, Bejjanki VR, Pouget A. Insights from a Simple Expression for Linear Fisher Information in a Recurrently Connected Population of Spiking Neurons. Neural Computation. 2011 Mar; 23(6):1484–1502.

Bondy AG, Haefner RM, Cumming BG. Feedback Determines the Structure of Correlated Variability in Primary Visual Cortex. Nature Neuroscience. 2018 Apr; 21(4):598–606.

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

731 **Brunel N**, Nadal JP. Mutual Information, Fisher Information, and Population Coding. Neural Computation.
732 1998; 10(7):1731–1757.

733 **Chanialidis C**, Evers L, Neocleous T, Nobile A. Efficient Bayesian Inference for COM-Poisson Regression Models.
734 Statistics and Computing. 2018 May; 28(3):595–608.

735 **Cover TM**, Thomas JA. Elements of Information Theory. 2nd ed ed. Hoboken, N.J: Wiley-Interscience; 2006.

736 **Dayan P**, Abbott LF. Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems.
737 Massachusetts Institute of Technology Press; 2005.

738 **Doya K**. Bayesian Brain: Probabilistic Approaches to Neural Coding. MIT press; 2007.

739 **Drugowitsch J**, Mendonça AG, Mainen ZF, Pouget A. Learning Optimal Decisions with Confidence. Proceedings
740 of the National Academy of Sciences. 2019 Dec; 116(49):24872–24880.

741 **Ecker AS**, Berens P, Cotton RJ, Subramaniyan M, Denfield GH, Cadwell CR, Smirnakis SM, Bethge M, Tolias AS.
742 State Dependence of Noise Correlations in Macaque Primary Visual Cortex. Neuron. 2014 Apr; 82(1):235–
743 248.

744 **Ecker AS**, Berens P, Tolias AS, Bethge M. The Effect of Noise Correlations in Populations of Diversely Tuned
745 Neurons. Journal of Neuroscience. 2011 Oct; 31(40):14272–14283.

746 **Ecker AS**, Denfield GH, Bethge M, Tolias AS. On the Structure of Neuronal Population Activity under Fluctuations
747 in Attentional State. The Journal of Neuroscience. 2016 Feb; 36(5):1775–1789.

748 **Festa D**, Aschner A, Davila A, Kohn A, Coen-Cagli R. Neuronal Variability Reflects Probabilistic Inference Tuned
749 to Natural Image Statistics. bioRxiv. 2020 Jun; p. 2020.06.17.142182.

750 **Ganguli D**, Simoncelli EP. Efficient Sensory Encoding and Bayesian Inference with Heterogeneous Neural Pop-
751 ulations. Neural Computation. 2014 Oct; 26(10):2103–2134.

752 **Ganmor E**, Segev R, Schneidman E. A Thesaurus for a Neural Population Code. eLife. 2015 Sep; 4:e06134.

753 **Goris RLT**, Movshon JA, Simoncelli EP. Partitioning Neuronal Variability. Nature Neuroscience. 2014 Jun;
754 17(6):858–865.

755 **Graf ABA**, Kohn A, Jazayeri M, Movshon JA. Decoding the Activity of Neuronal Populations in Macaque Primary
756 Visual Cortex. Nature Neuroscience. 2011 Feb; 14(2):239–245.

757 **Granot-Atedgi E**, Tkačik G, Segev R, Schneidman E. Stimulus-Dependent Maximum Entropy Models of Neural
758 Population Codes. PLOS Computational Biology. 2013 Mar; 9(3):e1002922.

759 **Herz AV**, Mathis A, Stemmler M. Periodic Population Codes: From a Single Circular Variable to Higher Dimen-
760 sions, Multiple Nested Scales, and Conceptual Spaces. Current Opinion in Neurobiology. 2017 Oct; 46:99–
761 108.

762 **Inouye DI**, Yang E, Allen GI, Ravikumar P. A Review of Multivariate Distributions for Count Data Derived from the
763 Poisson Distribution: A Review of Multivariate Distributions for Count Data. Wiley Interdisciplinary Reviews:
764 Computational Statistics. 2017 May; 9(3):e1398.

765 **Kafashan M**, Jaffe A, Chettih SN, Nogueira R, Arandia-Romero I, Harvey CD, Moreno-Bote R, Drugowitsch J.
766 Scaling of Information in Large Neural Populations Reveals Signatures of Information-Limiting Correlations.
767 bioRxiv. 2020 Jan; p. 2020.01.10.902171.

768 **Kanashiro T**, Ocker GK, Cohen MR, Doiron B. Attentional Modulation of Neuronal Variability in Circuit Models
769 of Cortex. eLife. 2017 Jun; 6:e23978.

770 **Kanitscheider I**, Coen-Cagli R, Kohn A, Pouget A. Measuring Fisher Information Accurately in Correlated Neural
771 Populations. PLoS computational biology. 2015; 11(6):e1004218.

772 **Kanitscheider I**, Coen-Cagli R, Pouget A. Origin of Information-Limiting Noise Correlations. Proceedings of the
773 National Academy of Sciences. 2015 Dec; 112(50):E6973–E6982.

774 **Karlis D**, Meligkotsidou L. Finite Mixtures of Multivariate Poisson Distributions with Application. Journal of
775 Statistical Planning and Inference. 2007 Jun; 137(6):1942–1960.

776 **Kingma D**, Ba J. Adam: A Method for Stochastic Optimization. arXiv preprint arXiv:14126980. 2014; .

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

Kohn A, Coen-Cagli R, Kanitscheider I, Pouget A. Correlations and Neuronal Population Information. Annual Review of Neuroscience. 2016 Jul; 39(1):237–256.

Kriegeskorte N, Douglas PK. Cognitive Computational Neuroscience. Nature Neuroscience. 2018 Aug; p. 1.

Lyamzin DR, Macke JH, Lesica NA. Modeling Population Spike Trains with Specified Time-Varying Spike Rates, Trial-to-Trial Variability, and Pairwise Signal and Noise Correlations. Frontiers in Computational Neuroscience. 2010; 4.

Ma WJ, Beck J, Latham P, Pouget A. Bayesian Inference with Probabilistic Population Codes. Nature Neuroscience. 2006 Oct; 9(11):1432–1438.

Macke JH, Buesing L, Cunningham JP, Yu BM, Shenoy KV, Sahani M. Empirical Models of Spiking in Neural Populations. In: Shawe-Taylor J, Zemel RS, Bartlett PL, Pereira F, Weinberger KQ, editors. *Advances in Neural Information Processing Systems 24* Curran Associates, Inc.; 2011.p. 1350–1358.

Macke JH, Murray I, Latham PE. How Biased Are Maximum Entropy Models? In: Shawe-Taylor J, Zemel RS, Bartlett PL, Pereira F, Weinberger KQ, editors. *Advances in Neural Information Processing Systems 24* Curran Associates, Inc.; 2011.p. 2034–2042.

Makin JG, Dichter BK, Sabes PN. Learning to Estimate Dynamical State with Probabilistic Population Codes. PLoS Comput Biol. 2015 Nov; 11(11):e1004554.

Maunsell JHR. Neuronal Mechanisms of Visual Attention. Annual Review of Vision Science. 2015 Nov; 1(1):373–391.

McLachlan GJ, Lee SX, Rathnayake SI. Finite Mixture Models. Annual Review of Statistics and Its Application. 2019; 6(1):355–378.

Meshulam L, Gauthier JL, Brody CD, Tank DW, Bialek W. Collective Behavior of Place and Non-Place Neurons in the Hippocampal Network. Neuron. 2017 Dec; 96(5):1178–1191.e4.

Montijn JS, Liu RG, Aschner A, Kohn A, Latham PE, Pouget A. Strong Information-Limiting Correlations in Early Visual Areas. bioRxiv. 2019 Nov; .

Moreno-Bote R, Beck J, Kanitscheider I, Pitkow X, Latham P, Pouget A. Information-Limiting Correlations. Nature Neuroscience. 2014 Oct; 17(10):1410–1417.

Neal RM, Hinton GE. A View of the EM Algorithm That Justifies Incremental, Sparse, and Other Variants. In: *Learning in Graphical Models* Springer; 1998.p. 355–368.

Okun M, Steinmetz NA, Cossell L, Iacaruso MF, Ko H, Barthó P, Moore T, Hofer SB, Mrsic-Flogel TD, Carandini M, Harris KD. Diverse Coupling of Neurons to Populations in Sensory Cortex. Nature. 2015 May; 521(7553):511–515.

Panzeri S, Harvey CD, Piasini E, Latham PE, Fellin T. Cracking the Neural Code for Sensory Perception by Combining Statistics, Intervention, and Behavior. Neuron. 2017 Feb; 93(3):491–507.

Park IM, Meister MLR, Huk AC, Pillow JW. Encoding and Decoding in Parietal Cortex during Sensorimotor Decision-Making. Nature Neuroscience. 2014 Oct; 17(10):1395–1403.

Pillow JW, Ahmadian Y, Paninski L. Model-Based Decoding, Information Estimation, and Change-Point Detection Techniques for Multineuron Spike Trains. Neural Computation. 2010 Oct; 23(1):1–45.

Pillow JW, Shlens J, Paninski L, Sher A, Litke AM, Chichilnisky EJ, Simoncelli EP. Spatio-Temporal Correlations and Visual Signalling in a Complete Neuronal Population. Nature. 2008 Aug; 454(7207):995–999.

Pitkow X, Angelaki DE. Inference in the Brain: Statistics Flowing in Redundant Population Codes. Neuron. 2017 Jun; 94(5):943–953.

Pouget A, Drugowitsch J, Kepecs A. Confidence and Certainty: Distinct Probabilistic Quantities for Different Goals. Nature neuroscience. 2016; 19(3):366–374.

Rabinowitz NC, Goris RL, Cohen M, Simoncelli EP. Attention Stabilizes the Shared Gain of V4 Populations. eLife. 2015 Nov; 4:e08998.

Ruff DA, Cohen MR. Simultaneous Multi-Area Recordings Suggest That Attention Improves Performance by Reshaping Stimulus Representations. Nature Neuroscience. 2019 Sep; p. 1–8.

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

824 **Rumyantsev OI**, Lecoq JA, Hernandez O, Zhang Y, Savall J, Chrapkiewicz R, Li J, Zeng H, Ganguli S, Schnitzer MJ.
825 Fundamental Bounds on the Fidelity of Sensory Cortical Coding. Nature. 2020 Mar; p. 1–6.

826 **Runyan CA**, Piasini E, Panzeri S, Harvey CD. Distinct Timescales of Population Coding across Cortex. Nature.
827 2017 Aug; 548(7665):92–96.

828 **Schneidman E**. Towards the Design Principles of Neural Population Codes. Current Opinion in Neurobiology.
829 2016 Apr; 37:133–140.

830 **Schneidman E**, Berry MJ, Segev R, Bialek W. Weak Pairwise Correlations Imply Strongly Correlated Network
831 States in a Neural Population. Nature. 2006 Apr; 440(7087):1007–1012.

832 **Semedo JD**, Zandvakili A, Machens CK, Yu BM, Kohn A. Cortical Areas Interact through a Communication Sub-
833 space. Neuron. 2019 Apr; 102(1):249–259.e4.

834 **Seriès P**, Latham PE, Pouget A. Tuning Curve Sharpening for Orientation Selectivity: Coding Efficiency and the
835 Impact of Correlations. Nature neuroscience. 2004; 7(10):1129.

836 **Shidara M**, Mizuhiki T, Richmond BJ. Neuronal Firing in Anterior Cingulate Neurons Changes Modes across
837 Trials in Single States of Multitrial Reward Schedules. Experimental Brain Research. 2005 May; 163(2):242–
838 245.

839 **Shmueli G**, Minka TP, Kadane JB, Borle S, Boatwright P. A Useful Distribution for Fitting Discrete Data: Re-
840 vival of the Conway–Maxwell–Poisson Distribution. Journal of the Royal Statistical Society: Series C (Applied
841 Statistics). 2005; 54(1):127–142.

842 **Smith MA**, Kohn A. Spatial and Temporal Scales of Neuronal Correlation in Primary Visual Cortex. Journal of
843 Neuroscience. 2008 Nov; 28(48):12591–12603.

844 **Snow M**, Coen-Cagli R, Schwartz O. Specificity and Timescales of Cortical Adaptation as Inferences about Nat-
845 ural Movie Statistics. Journal of Vision. 2016 Oct; 16(13).

846 **Snow M**, Coen-Cagli R, Schwartz O. Adaptation in the Visual Cortex: A Case for Probing Neuronal Populations
847 with Natural Stimuli. F1000Research. 2017 Jul; 6:1246.

848 **Snyder AC**, Morais MJ, Kohn A, Smith MA. Correlations in V1 Are Reduced by Stimulation Outside the Receptive
849 Field. Journal of Neuroscience. 2014 Aug; 34(34):11222–11227.

850 **Sokoloski S**. Implementing a Bayes Filter in a Neural Circuit: The Case of Unknown Stimulus Dynamics. Neural
851 Computation. 2017 Jun; 29(9):2450–2490.

852 **Sokoloski S**. Implementing Bayesian Inference with Neural Networks. Dissertation, University of Leipzig; 2019.

853 **Sompolinsky H**, Yoon H, Kang K, Shamir M. Population Coding in Neuronal Systems with Correlated Noise.
854 Physical Review E. 2001 Oct; 64(5).

855 **Stevenson IH**. Flexible Models for Spike Count Data with Both Over- and under- Dispersion. Journal of Com-
856 putational Neuroscience. 2016 Aug; 41(1):29–43.

857 **Sur P**, Shmueli G, Bose S, Dubey P. Modeling Bimodal Discrete Data Using Conway-Maxwell-Poisson Mixture
858 Models. Journal of Business & Economic Statistics. 2015 Jul; 33(3):352–365.

859 **Taouali W**, Benvenuti G, Wallisch P, Chavane F, Perrinet LU. Testing the Odds of Inherent vs. Observed Overdis-
860 persion in Neural Spike Counts. Journal of Neurophysiology. 2015 Oct; 115(1):434–444.

861 **Wainwright MJ**, Jordan MI. Graphical Models, Exponential Families, and Variational Inference. Foundations
862 and Trends® in Machine Learning. 2008; 1(1-2):1–305.

863 **Walker EY**, Cotton RJ, Ma WJ, Tolias AS. A Neural Basis of Probabilistic Computation in Visual Cortex. Nature
864 Neuroscience. 2020 Jan; 23(1):122–129.

865 **Wei XX**, Stocker AA. A Bayesian Observer Model Constrained by Efficient Coding Can Explain 'anti-Bayesian'
866 Percepts. Nature Neuroscience. 2015 Sep; 18(10):1509–1517.

867 **Whiteway MR**, Butts DA. The Quest for Interpretable Models of Neural Population Activity. Current Opinion
868 in Neurobiology. 2019 Oct; 58:86–93.

bioRxiv preprint doi: https://doi.org/10.1101/2020.11.05.369827; this version posted November 6, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

869 **Wiener MC**, Richmond BJ. Decoding Spike Trains Instant by Instant Using Order Statistics and the Mixture-of-
870 Poissons Model. Journal of Neuroscience. 2003 Mar; 23(6):2394–2406.

871 **Yamins DLK**, Hong H, Cadieu CF, Solomon EA, Seibert D, DiCarlo JJ. Performance-Optimized Hierarchical Models
872 Predict Neural Responses in Higher Visual Cortex. Proceedings of the National Academy of Sciences. 2014
873 Oct; 111(23):8619–8624.

874 **Yerxa TE**, Kee E, DeWeese MR, Cooper EA. Efficient Sensory Coding of Multidimensional Stimuli. PLOS Compu-
875 tational Biology. 2020 Sep; 16(9):e1008146.

876 **Yu BM**, Cunningham JP, Santhanam G, Ryu SI, Shenoy KV, Sahani M. Gaussian-Process Factor Analysis for
877 Low-Dimensional Single-Trial Analysis of Neural Population Activity. Journal of Neurophysiology. 2009 Jul;
878 102(1):614–635.

879 **Zemel RS**, Dayan P, Pouget A. Probabilistic Interpretation of Population Codes. Neural computation. 1998;
880 10(2):403–430.

881 **Zhao Y**, Park IM. Variational Latent Gaussian Process for Recovering Single-Trial Dynamics from Population
882 Spike Trains. Neural Computation. 2017 Mar; 29(5):1293–1316.

883 **Zohary E**, Shadlen MN, Newsome WT. Correlated Neuronal Discharge Rate and Its Implications for Psychophys-
884 ical Performance. Nature. 1994 Jul; 370(6485):140–143.