Interrogating theoretical models of neural computation with deep inference

Sean R. Bittner[1], Agostina Palmigiano[1], Alex T. Piet[2,3], Chunyu A. Duan[4], Carlos D. Brody[2,3,5], Kenneth D. Miller[1], and John P. Cunningham[6].

[1]Department of Neuroscience, Columbia University,

[2]Princeton Neuroscience Institute,

[3]Princeton University,

[4]Institute of Neuroscience, Chinese Academy of Sciences,

[5]Howard Hughes Medical Institute,

[6]Department of Statistics, Columbia University

1 Abstract

A cornerstone of theoretical neuroscience is the circuit model: a system of equations that captures a hypothesized neural mechanism. Such models are valuable when they give rise to an experimentally observed phenomenon – whether behavioral or in terms of neural activity – and thus can offer insights into neural computation. The operation of these circuits, like all models, critically depends on the choices of model parameters. Historically, the gold standard has been to analytically derive the relationship between model parameters and computational properties. However, this enterprise quickly becomes infeasible as biologically realistic constraints are included into the model increasing its complexity, often resulting in *ad hoc* approaches to understanding the relationship between model and computation. We bring recent machine learning techniques – the use of deep generative models for probabilistic inference – to bear on this problem, learning distributions of parameters that produce the specified properties of computation. Importantly, the techniques we introduce offer a principled means to understand the implications of model parameter choices on computational properties of interest. We motivate this methodology with a worked example analyzing sensitivity in the stomatogastric ganglion. We then use it to go beyond linear theory of neuron-type input-responsivity in a model of primary visual cortex, gain a mechanistic understanding of rapid task switching in superior colliculus models, and attribute error to connectivity properties in recurrent neural networks solving a simple mathematical task. More generally, this work suggests a departure from realism vs tractability considerations, towards the use of modern machine learning for sophisticated interrogation of biologically relevant models.

2 Introduction

The fundamental practice of theoretical neuroscience is to use a mathematical model to understand neural computation, whether that computation enables perception, action, or some intermediate processing [1]. A neural computation is systematized with a set of equations – the model – and these equations are motivated by biophysics, neurophysiology, and other conceptual considerations. The function of this system is governed by the choice of model parameters, which when configured in a particular way, give rise to a measurable signature of a computation. The work of analyzing a model then requires solving the inverse problem: given a computation of interest, how can we reason about these particular parameter configurations? The inverse problem is crucial for reasoning about likely parameter values, uniquenesses and degeneracies, attractor states and phase transitions, and predictions made by the model.

Consider the idealized practice: one carefully designs a model and analytically derives how model parameters govern the computation. Seminal examples of this gold standard (which often adopt approaches from statistical physics) include our field's understanding of memory capacity in associative neural networks [2], chaos and autocorrelation timescales in random neural networks [3], the paradoxical effect [4], and decision making [5]. Unfortunately, as circuit models include more biological realism, theory via analytical derivation becomes intractable. This creates an unfavorable tradeoff. On the one hand, one may tractably analyze systems of equations with unrealistic assumptions (for example symmetry or gaussianity), mathematically formalizing how parameters affect computation in a too-simple model. On the other hand, one may choose a more biologically accurate, scientifically relevant model at the cost of *ad hoc* approaches to analysis (such as simply examining simulated activity), potentially resulting in bad inference of parameters and thus erroneous scientific predictions or conclusions.

Of course, this same tradeoff has been confronted in many scientific fields characterized by the need to do inference in complex models. In response, the machine learning community has made remarkable progress in recent years, via the use of deep neural networks as a powerful inference engine: a flexible function family that can map observed phenomena (in this case the measurable signal of some computation) back to probability distributions quantifying the likely parameter configurations. One celebrated example of this approach from machine learning, of which we draw key inspiration for this work, is the variational autoencoder [6, 7], which uses a deep neural network to induce an (approximate) posterior distribution on hidden variables in a latent variable model, given data. Indeed, these tools have been used to great success in neuroscience as well, in particular for interrogating parameters (sometimes treated as hidden states) in models of both cortical population activity [8, 9, 10, 11] and animal behavior [12, 13, 14]. These works have used deep neural networks to expand the expressivity and accuracy of statistical models of neural data [15].

However, these inference tools have not significantly influenced the study of theoretical neuroscience models, for at least three reasons. First, at a practical level, the nonlinearities and dynamics of many theoretical models are such that conventional inference tools typically produce a narrow set of insights into these models. Indeed, only in the last few years has deep learning research advanced to a point of relevance to this class of problem. Second, the object of interest from a theoretical model is not typically data itself, but rather a qualitative phenomenon – inspection of model behavior, or better, a measurable signature of some computation – an *emergent property* of the model. Third, because theoreticians work carefully to construct a model that has biological relevance, such a model as a result often does not fit cleanly into the framing of a statistical model. Technically, because many such models stipulate a noisy system of differential equations that can only be sampled or realized through forward simulation, they lack the explicit likelihood and priors central to the probabilistic modeling toolkit.

To address these three challenges, we developed an inference methodology – 'emergent property inference' – which learns a distribution over parameter configurations in a theoretical model. This distribution has two critical properties: *(i)* it is chosen such that draws from the distribution (parameter configurations) correspond to systems of equations that give rise to a specified emergent property (a set of constraints); and *(ii)* it is chosen to have maximum entropy given those constraints, such that we identify all likely parameters and can use the distribution to reason about parametric sensitivity and degeneracies [16]. First, we stipulate a bijective deep neural network that induces a flexible family of probability distributions over model parameterizations with a probability density we can calculate [17, 18, 19]. Second, we quantify the notion of emergent properties as a set of moment constraints on datasets generated by the model. Thus, an emergent property is not a single data realization, but a phenomenon or a feature of the model, which is ultimately the object of interest in theoretical neuroscience. Conditioning on an emergent property requires a variant of deep probabilistic inference methods, which we have previously introduced [20]. Third, because we cannot assume the theoretical model has explicit likelihood on data or the emergent property of interest, we use stochastic gradient techniques in the spirit of likelihood free variational inference

[21]. Taken together, emergent property inference (EPI) provides a methodology for inferring parameter configurations consistent with a particular emergent phenomena in theoretical models. We use a classic example of parametric degeneracy in a biological system, the stomatogastric ganglion [22], to motivate and clarify the technical details of EPI.

Equipped with this methodology, we then investigated three models of current importance in theoretical neuroscience. These models were chosen to demonstrate generality through ranges of biological realism (from conductance-based biophysics to recurrent neural networks), neural system function (from pattern generation to abstract cognitive function), and network scale (from four to infinite neurons). First, we use EPI to produce a set of verifiable hypotheses of input-responsivity in a four neuron-type dynamical model of primary visual cortex; we then validate these hypotheses in the model. Second, we demonstrated how the systematic application of EPI to levels of task performance can generate experimentally testable hypotheses regarding connectivity in superior colliculus. Third, we use EPI to uncover the sources of error in a low-rank recurrent neural network executing a simple mathematical task. The novel scientific insights offered by EPI contextualize and clarify the previous studies exploring these models [23, 24, 25, 26], and more generally, these results point to the value of deep inference for the interrogation of biologically relevant models.

We note that, during our preparation and early presentation of this work [27, 28], another work has arisen with broadly similar goals: bringing statistical inference to mechanistic models of neural circuits [29, 30]. We are encouraged by this general problem being recognized by others in the community, and we emphasize that these works offer complementary neuroscientific contributions (different theoretical models of focus) and use different technical methodologies (ours is built on our prior work [20], theirs similarly [31]). These distinct methodologies and scientific investigations emphasize the increased importance and timeliness of both works.

3 Results

3.1 Motivating emergent property inference of theoretical models

Consideration of the typical workflow of theoretical modeling clarifies the need for emergent property inference. First, one designs or chooses an existing model that, it is hypothesized, captures the computation of interest. To ground this process in a well-known example, consider the stomatogastric ganglion (STG) of crustaceans, a small neural circuit which generates multiple rhythmic muscle activation patterns for digestion [32]. Despite full knowledge of STG connectivity and a precise characterization of its rhythmic pattern generation, biophysical models of the STG have complicated relationships between circuit parameters and neural activity [22, 33]. A model of the STG [23] is shown schematically in Figure 1A, and note that the behavior of this model will be critically dependent on its parameterization – the choices of conductance parameters $z = [g_{\text{el}}, g_{\text{synA}}]$. Specifically, the two fast neurons ($f1$ and $f2$) mutually inhibit one another, and oscillate at a faster frequency than the mutually inhibiting slow neurons ($s1$ and $s2$). The hub neuron (hub) couples with either the fast or slow population or both.

Second, once the model is selected, one defines the emergent property, the measurable signal of scientific interest. To continue our running STG example, one such emergent property is the phenomenon of *network syncing* – in certain parameter regimes, the frequency of the hub neuron matches that of the fast and slow populations at an intermediate frequency. This emergent property is shown in Figure 1A at a frequency of 0.53Hz.

Third, qualitative parameter analysis ensues: since precise mathematical analysis is intractable in this model, a brute force sweep of parameters is done [23]. Subsequently, a qualitative description is formulated to describe the different parameter configurations that lead to the emergent property. In this last step lies the opportunity for a precise quantification of the emergent property as a statistical feature of the model. Once we have such a methodology, we can infer a probability distribution over parameter configurations that produce this emergent property.

Before presenting technical details (in the following section), let us understand emergent property inference schematically: EPI (Fig. 1A gray box) takes, as input, the model and the specified emergent property, and as its output, produces the parameter distribution shown in Figure 1B. This distribution – represented for clarity as samples from the distribution – is then a scientifically meaningful and mathematically tractable object. In the STG model, this distribution can be specifically queried to reveal the prototypical parameter configuration for network syncing (the mode; Figure 1B yellow star), and how network syncing decays based on changes away from the mode. The eigenvectors (of the Hessian of the distribution at the mode) quantitatively formalize the robustness of network syncing (Fig. 1B solid (v_1) and dashed (v_2) black arrows). Indeed, samples equidistant from the mode along these EPI-identified dimensions of sensitivity (v_1) and degeneracy (v_2) agree with error contours (Fig. 1B, contours) and have diminished or preserved network syncing, respectively (Figure 1B inset and activity traces) (see Section 5.2.1).

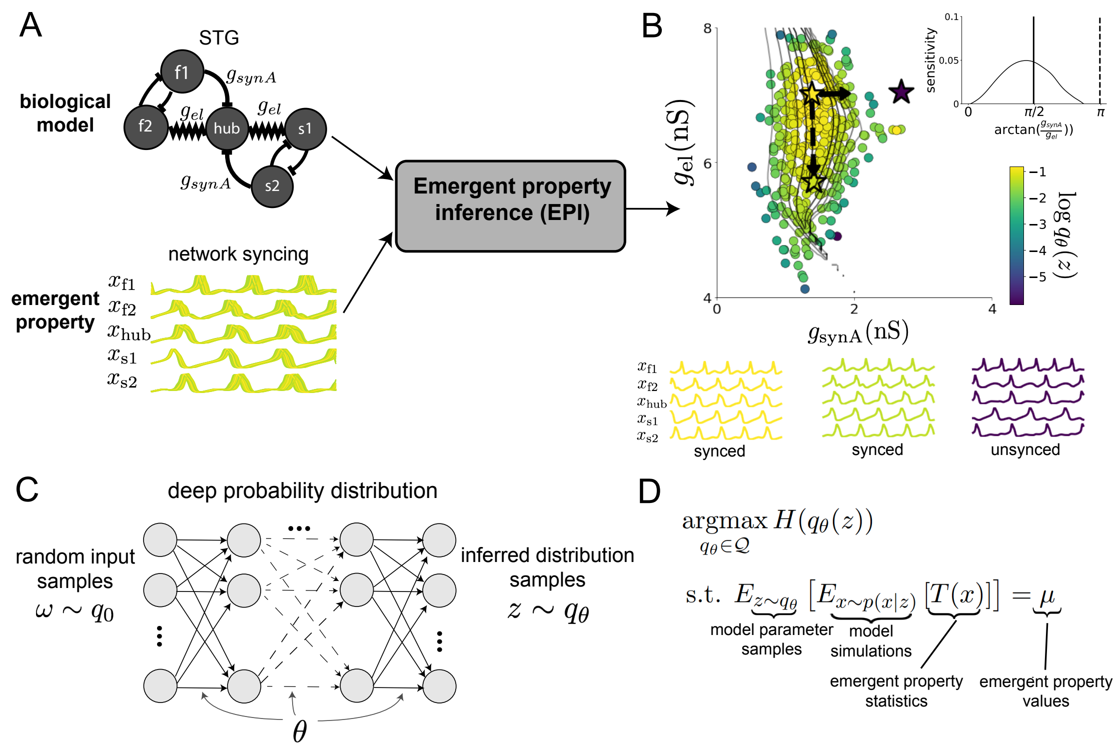

Figure 1: Emergent property inference (EPI) in the stomatogastric ganglion. A. For a choice of model (STG) and emergent property (network syncing), emergent property inference (EPI, gray box) learns a distribution of the model parameters $z = [g_{\text{el}}, g_{\text{synA}}]$ producing network syncing. In the STG model, jagged connections indicate electrical coupling having electrical conductance g_{el}. Other connections in the diagram are inhibitory synaptic projections having strength g_{synA} onto the hub neuron, and $g_{\text{synB}} = 5\text{nS}$ for mutual inhibitory connections. Network syncing traces are colored by log probability density of their generating parameters (stars) in the EPI-inferred distribution. B. The EPI distribution of STG model parameters producing network syncing. Samples are colored by log probability density. Distribution contours of emergent property value error are shown at levels of 2.5×10^{-5}, 5×10^{-5}, 1×10^{-4}, 2×10^{-4}, and 4×10^{-4} (dark to light gray). Eigenvectors of the Hessian at the mode of the inferred distribution are indicated as v_1 (solid) and v_2 (dashed) with lengths scaled by the square root of the absolute value of their eigenvalues. Simulated activity is shown for three samples (stars). (Inset) Sensitivity of the system with respect to network syncing along all dimensions of parameter space away from the mode. v_1 is sensitive to network syncing ($p < 10^{-4}$), while v_2 is not ($p = 0.67$) (see Section 5.2.1). C. Deep probability distributions map a latent random variable w through a deep neural network with weights and biases θ to parameters $z = f_\theta(w)$ distributed as $q_\theta(z)$. D. EPI optimization: To learn the EPI distribution $q_\theta(z)$ of model parameters that produce an emergent property, the emergent property statistics $T(x)$ are set in expectation over model parameter samples $z \sim q_\theta(z)$ and model simulations $x \sim p(x \mid z)$ to emergent property values μ.

3.2 A deep generative modeling approach to emergent property inference

Emergent property inference (EPI) systematizes the three-step procedure of the previous section. First, we consider the model as a coupled set of differential (and potentially stochastic) equations [23]. In the running STG example, the model activity $x = [x_{\text{f1}}, x_{\text{f2}}, x_{\text{hub}}, x_{\text{s1}}, x_{\text{s2}}]$ is the membrane potential for each neuron, which evolves according to the biophysical conductance-based equation:

$$C_m \frac{dx}{dt} = -h(x; z) = -\left[h_{leak}(x; z) + h_{Ca}(x; z) + h_K(x; z) + h_{hyp}(x; z) + h_{elec}(x; z) + h_{syn}(x; z)\right] \tag{1}$$

where C_m=1nF, and h_{leak}, h_{Ca}, h_K, h_{hyp}, h_{elec}, and h_{syn} are the leak, calcium, potassium, hyperpolarization, electrical, and synaptic currents, all of which have their own complicated dependence on x and $z = [g_{\text{el}}, g_{\text{synA}}]$ (see Section 5.2.1).

Second, we define the emergent property, which as above is network syncing: oscillation of the entire population at an intermediate frequency of our choosing (Figure 1A bottom). Quantifying this phenomenon is straightforward: we define network syncing to be that each neuron's spiking frequency – denoted $\omega_{\text{f1}}(x)$, $\omega_{\text{f2}}(x)$, etc. – is close to an intermediate frequency of 0.53Hz. Mathematically, we achieve this via constraints on the mean and variance of $\omega_\alpha(x)$ for each neuron $\alpha \in \{\text{f1}, \text{f2}, \text{hub}, \text{s1}, \text{s2}\}$:

$$\mathbb{E}\left[T(x)\right] \triangleq \mathbb{E}\begin{bmatrix} \omega_{\text{f1}}(x) \\ \vdots \\ (\omega_{\text{f1}}(x) - 0.53)^2 \\ \vdots \end{bmatrix} = \begin{bmatrix} 0.53 \\ \vdots \\ 0.025^2 \\ \vdots \end{bmatrix} \triangleq \mu, \tag{2}$$

which completes the quantification of the emergent property.

Third, we perform emergent property inference: we find a distribution over parameter configurations z, and insist that samples from this distribution produce the emergent property; in other words, they obey the constraints introduced in Equation 2. This distribution will be chosen from a family of probability distributions $\mathcal{Q} = \{q_\theta(z) : \theta \in \Theta\}$, defined by a deep generative distribution of the normalizing flow class [17, 18, 19] – neural networks which transform a simple distribution into a suitably complicated distribution (as is needed here). This deep distribution is represented in Figure 1C (see Section 5.1). Then, mathematically, we must solve the following optimization program:

$$\begin{aligned} &\underset{q_\theta \in \mathcal{Q}}{\text{argmax}}\, H(q_\theta(z)) \\ &\text{s.t. } \mathbb{E}_{z \sim q_\theta}\left[\mathbb{E}_{x \sim p(x|z)}\left[T(x)\right]\right] = \mu, \end{aligned} \tag{3}$$

where $T(x), \mu$ are defined as in Equation 2, and $p(x|z)$ is the intractable distribution of data from the model, x, given that model's parameters z (we access samples from this distribution by running the model forward). The purpose of each element in this program is detailed in Figure 1D. Finally, we recognize that many distributions in $\mathcal{Q}$ will respect the emergent property constraints, so we require a normative principle to select amongst them. This principle is captured in Equation 3 by the primal objective H. Here we chose Shannon entropy as a means to find parameter distributions with minimal assumptions beyond some chosen structure [34, 35, 20, 36], but we emphasize that the EPI methodology is unaffected by this choice (although the results of course depend on the primal objective chosen).

EPI optimizes the weights and biases θ of the deep neural network (which induces the probability distribution) by iteratively solving Equation 3. The optimization is complete when the sampled models with parameters $z \sim q_\theta$ produce activity consistent with the specified emergent property (Fig. S4). Such convergence is evaluated with a hypothesis test that the mean of each emergent property statistic is not different than its emergent property value (see Section 5.1.2). Further validation of EPI is available in the supplementary materials, where we analyze a simpler model for which ground-truth statements can be made (Section 5.1.1). In relation to broader methodology, inspection of the EPI objective reveals a natural relationship to posterior inference. Specifically, EPI executes variational inference in an exponential family model, the sufficient statistics and mean parameter of which are defined by the emergent property statistics and values, respectively (see Section 5.1.4). Equipped with this method, we now prove out the value of EPI by using it to investigate and produce novel insights about three prominent models in neuroscience.

3.3 Comprehensive input-responsivity in a nonlinear sensory system

Dynamical models of excitatory (E) and inhibitory (I) populations with supralinear input-output function have succeeded in explaining a host of experimentally documented phenomena. In a regime characterized by inhibitory stabilization of strong recurrent excitation, these models give rise to paradoxical responses [4], selective amplification [37], surround suppression [38] and normalization [39]. Despite their strong predictive power, E-I circuit models rely on the assumption that inhibition can be studied as an indivisible unit. However, experimental evidence shows that inhibition is composed of distinct elements – parvalbumin (P), somatostatin (S), VIP (V) – composing 80% of GABAergic interneurons in V1 [40, 41, 42], and that these inhibitory cell types follow specific connectivity patterns (Fig. 2A) [43]. Recent theoretical advances [24, 44, 45], have only started

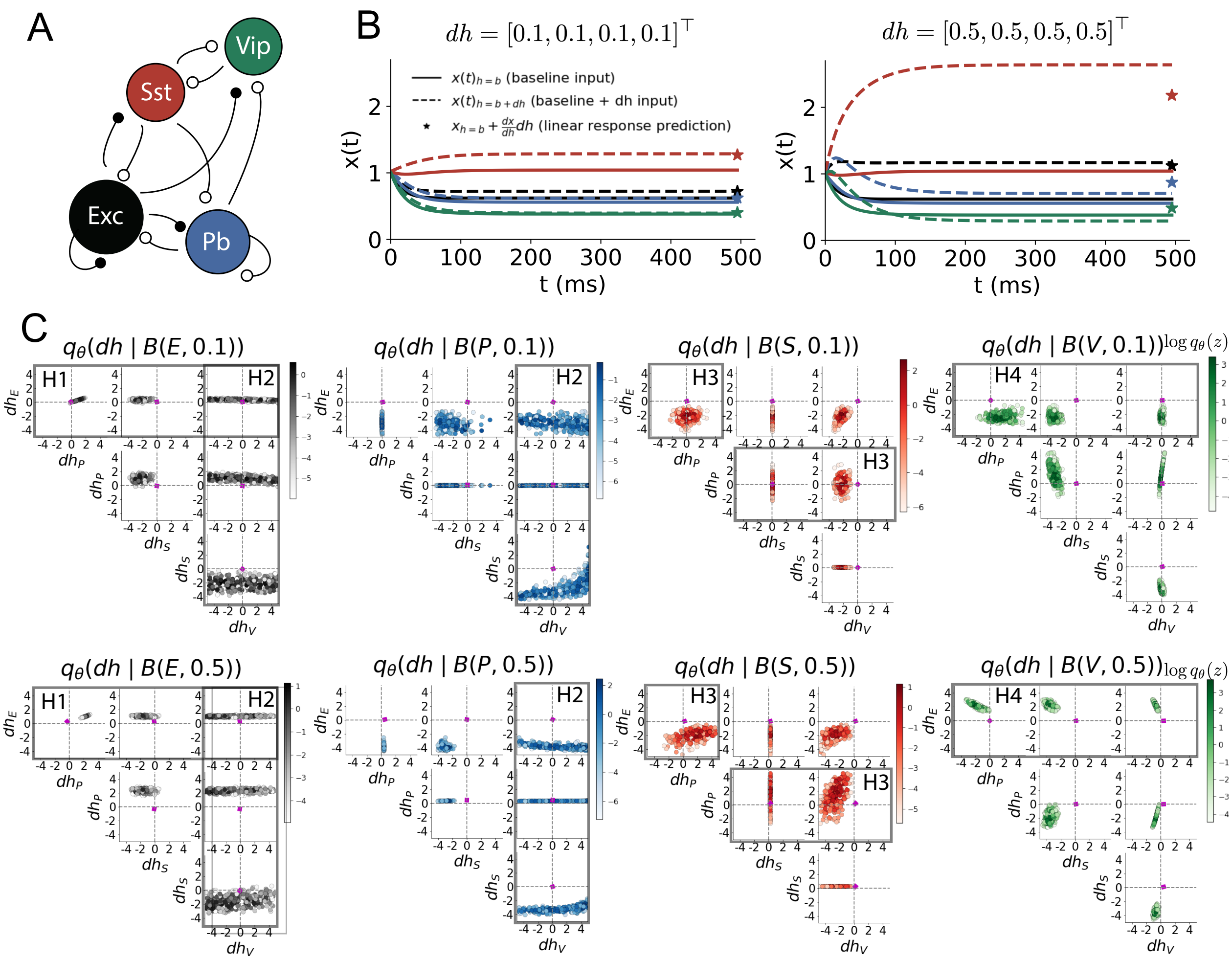

Figure 2: Hypothesis generation through EPI in a V1 model. A. Four-population model of primary visual cortex with excitatory (black), parvalbumin (blue), somatostatin (red), and VIP (green) neurons. Some neuron-types largely do not form synaptic projections to others (excitatory and inhibitory projections filled and unfilled, respectively). B. Linear response predictions become inaccurate with greater input strength. V1 model simulations for input (solid) $h = b$ and (dashed) $h = b + dh$. Stars indicate the linear response prediction. C. EPI distributions on differential input dh conditioned on differential response $\mathcal{B}(\alpha, y)$. Supporting evidence for the four generated hypotheses are indicated by gray boxes with labels H1, H2, H3, and H4. The linear prediction from two standard deviations away from y (from negative to positive) is overlaid in magenta (very small, near origin).

to address the consequences of this multiplicity in the dynamics of V1, strongly relying on linear theoretical tools. Here, we go beyond linear theory by systematically generating and evaluating hypotheses of circuit model function using EPI distributions of neuron-type inputs producing various neuron-type population responses.

Specifically, we consider a four-dimensional circuit model with dynamical state given by the firing rate x of each neuron-type population $x = [x_E, x_P, x_S, x_V]^\top$. Given a time constant of $\tau = 20$ ms and a power $n = 2$, the dynamics are driven by the rectified and exponentiated sum of recurrent (Wx) and external h inputs:

$$\tau \frac{dx}{dt} = -x + [Wx + h]_+^n. \tag{4}$$

The effective connectivity weights W were obtained from experimental recordings of publicly available datasets of mouse V1 [46, 47] (see Section 5.2.2). The input $h = b + dh$ is comprised of a baseline input $b = [b_E, b_P, b_S, b_V]^\top$ and a differential input $dh = [dh_E, dh_P, dh_S, dh_V]^\top$ to each neuron-type population. Throughout subsequent analyses, the baseline input is $b = [1, 1, 1, 1]^\top$.

With this model, we are interested in the differential responses of each neuron-type population to changes in input dh. Initially, we studied the linearized response of the system to input $\frac{dx_{ss}}{dh}$ at the steady state response x_{ss}, i.e. a fixed point. All analyses of this model consider the steady state response, so we drop the notation ss from here on. While this linearization accurately predicts differential responses $dx = [dx_E, dx_P, dx_S, dx_V]^\top$ for small differential inputs to each population $dh = [0.1, 0.1, 0.1, 0.1]^\top$ (Fig 2B left), the linearization is a poor predictor in this nonlinear model more generally (Fig. 2B right). Currently available approaches to deriving the steady state response of the system are limited.

To get a more comprehensive picture of the input-responsivity of each neuron-type beyond linear theory, we used EPI to learn a distribution of the differential inputs to each population dh that produce an increase of y in the rate of each neuron-type population $\alpha \in \{E, P, S, V\}$. We want to know the differential inputs dh that result in a differential steady state dx_α (the change in x_α when receiving input $h = b + dh$ with respect to the baseline $h = b$) of value y with some small, arbitrarily chosen amount of variance 0.01^2. These statements amount to the emergent property

$$\mathcal{B}(\alpha, y) \triangleq \mathbb{E}\begin{bmatrix} dx_\alpha \\ (dx_\alpha - y)^2 \end{bmatrix} = \begin{bmatrix} y \\ 0.01^2 \end{bmatrix}. \tag{5}$$

We maintain the notation $\mathcal{B}(\cdot)$ throughout the rest of the study as short hand for emergent property, which represents a different signature of computation in each application.

Using EPI, we inferred the distribution of dh shown in Figure 2C producing $\mathcal{B}(\alpha, y)$. Columns correspond to inferred distributions of excitatory ($\alpha = E$, red), parvalbumin ($\alpha = P$, blue), somatostatin ($\alpha = S$, red) and VIP ($\alpha = V$, green) neuron-type response increases, while each row corresponds to increase amounts of $y \in \{0.1, 0.5\}$. For each pair of parameters, we show the two-dimensional marginal distribution of samples colored by $\log q_\theta(dh \mid \mathcal{B}(\alpha, y))$. The inferred distributions immediately suggest four hypotheses:

H1: as is intuitive, each neuron-type's firing rate should be sensitive to that neuron-type's direct input (e.g. Fig. 2C H1 gray boxes indicate low variance in dh_E when $\alpha = E$. Same observation in all inferred distributions);

H2: the E- and P-populations should be largely unaffected by input to the V-population (Fig. 2C H2 gray boxes indicate high variance in dh_V when $\alpha \in \{E, P\}$);

H3: the S-population should be largely unaffected by input to the P-population (Fig. 2C H3 gray boxes indicate high variance in dh_P when $\alpha = S$);

H4: there should be a nonmonotonic response of the V-population with input to the E-population (Fig. 2C H4 gray boxes indicate that negative dh_E should result in small dx_V, but positive dh_E should elicit a larger dx_V);

We evaluate these hypotheses by taking perturbations in individual neuron-type input δh_α away from the modes of the inferred distributions at $y = 0.1$

$$dh^* = z^* = \underset{z}{\operatorname{argmax}} \log q_\theta(z \mid \mathcal{B}(\alpha, 0.1)). \tag{6}$$

Here δx_α is the change in steady state response of the system with input $h = b + dh^* + \delta h_\alpha \hat{u}_\alpha$ compared to $h = b + dh^*$, where $\hat{u}_\alpha$ is a unit vector in the dimension of α. The EPI-generated hypotheses are confirmed (for details, see Section 5.2.2):

H1: the neuron-type responses are sensitive to their direct inputs (Fig. 3A black, 3B blue, 3C red, 3D green);

H2: the E- and P-populations are not affected by δh_V (Fig. 3A green, 3B green);

H3: the S-population is not affected by δh_P (Fig. 3C blue);

H4: the V-population exhibits a nonmonotonic response to δh_E (Fig. 3D black), and is in fact the only population to do so (Fig. 3A-C black).

These hypotheses were in stark contrast to what was available to us via traditional analytical

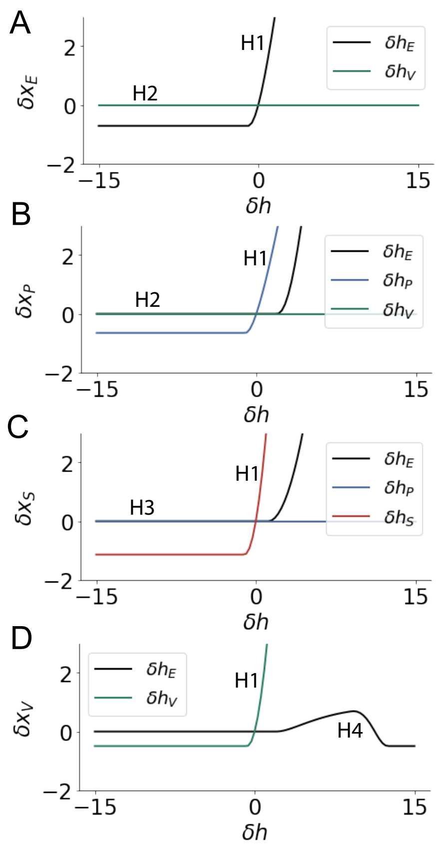

Figure 3: Confirming EPI generated hypotheses in V1. A. Differential responses δx_E by the E-population to changes in individual input $\delta h_\alpha \hat{u}_\alpha$ away from the mode of the EPI distribution dh^*. B-D Same plots for the P-, S-, and V-populations. Labels H1, H2, H3, and H4 indicate which curves confirm which hypotheses.

linear prediction (Fig. 2C, magenta, see Section 5.2.2). To this point, we have shown the utility of EPI on relatively low-level emergent properties like network syncing and differential neuron-type population responses. In the remainder of the study, we focus on using EPI to understand models of more abstract cognitive function.

3.4 Identifying neural mechanisms of flexible task switching

In a rapid task switching experiment [48], rats were explicitly cued on each trial to either orient towards a visual stimulus in the Pro (P) task or orient away from a visual stimulus in the Anti (A) task (Fig. 4a). Neural recordings in the midbrain superior colliculus (SC) exhibited two populations of neurons that simultaneously represented both task context (Pro or Anti) and motor response (contralateral or ipsilateral to the recorded side): the Pro/Contra and Anti/Ipsi neurons [25]. Duan et al. proposed a model of SC that, like the V1 model analyzed in the previous section, is a four-population dynamical system. We analyzed this model, where the neuron-type populations are functionally-defined as the Pro- and Anti-populations in each hemisphere (left (L) and right (R)), their connectivity is parameterized geometrically (Fig. 4B). The input-output function of this model is chosen such that the population responses $x = [x_{LP}, x_{LA}, x_{RP}, x_{RA}]^\top$ are bounded from 0 to 1 giving rise to high (1) or low (0) responses at the end of the trial:

$$x_\alpha = \left(\frac{1}{2} \tanh \left(\frac{u_\alpha - \epsilon}{\zeta} \right) + \frac{1}{2} \right) \tag{7}$$

where $\epsilon = 0.05$ and $\zeta = 0.5$. The dynamics evolve with timescale $\tau = 0.09$ via an internal variable u governed by connectivity weights W

$$\tau \frac{du}{dt} = -u + Wx + h + \sigma dB \tag{8}$$

with gaussian noise of variance $\sigma^2 = 1$. The input h is comprised of a cue-dependent input to the Pro or Anti populations, a stimulus orientation input to either the Left or Right populations, and a choice-period input to the entire network (see Section 5.2.3). Here, we use EPI to determine the changes in network connectivity $z = [sW_P, sW_A, vW_{PA}, vW_{AP}, dW_{PA}, dW_{AP}, hW_P, hW_A]$ resulting in greater levels of rapid task switching accuracy.

To quantify the emergent property of rapid task switching at various levels of accuracy, we considered the requirements of this model in this behavioral paradigm. At the end of successful trials, the response of the Pro population in the hemisphere of the correct choice must have a value near 1, while the Pro population in the opposite hemisphere must have a value near 0. Constraining a population response $x_\alpha \in [0, 1]$ to be either 0 or 1 can be achieved by requiring that it has Bernoulli variance (see Section 5.2.3). Thus, we can formulate rapid task switching at a level of accuracy $p \in [0, 1]$ in both tasks in terms of the average steady response of the Pro population $\hat{p}$ of the correct choice, the error in Bernoulli variance of that Pro neuron σ^2_{err}, and the average difference in Pro neuron responses d in both Pro and Anti trials:

$$\mathcal{B}(p) \quad \triangleq \quad \mathbb{E} \begin{bmatrix} \hat{p}_P \\ \hat{p}_A \\ (\hat{p}_P - p)^2 \\ (\hat{p}_A - p)^2 \\ \sigma^2_{P,err} \\ \sigma^2_{A,err} \\ d_P \\ d_A \end{bmatrix} = \begin{bmatrix} p \\ p \\ 0.15^2 \\ 0.15^2 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}. \tag{9}$$

Thus, $\mathcal{B}(p)$ denotes Bernoulli, winner-take-all responses between Pro neurons in a model executing rapid task switching near accuracy level p.

We used EPI to learn distributions of the SC weight matrix parameters z conditioned on of various levels of rapid task switching accuracy $\mathcal{B}(p)$ for $p \in \{50\%, 60\%, 70\%, 80\%, 90\%\}$. To make sense

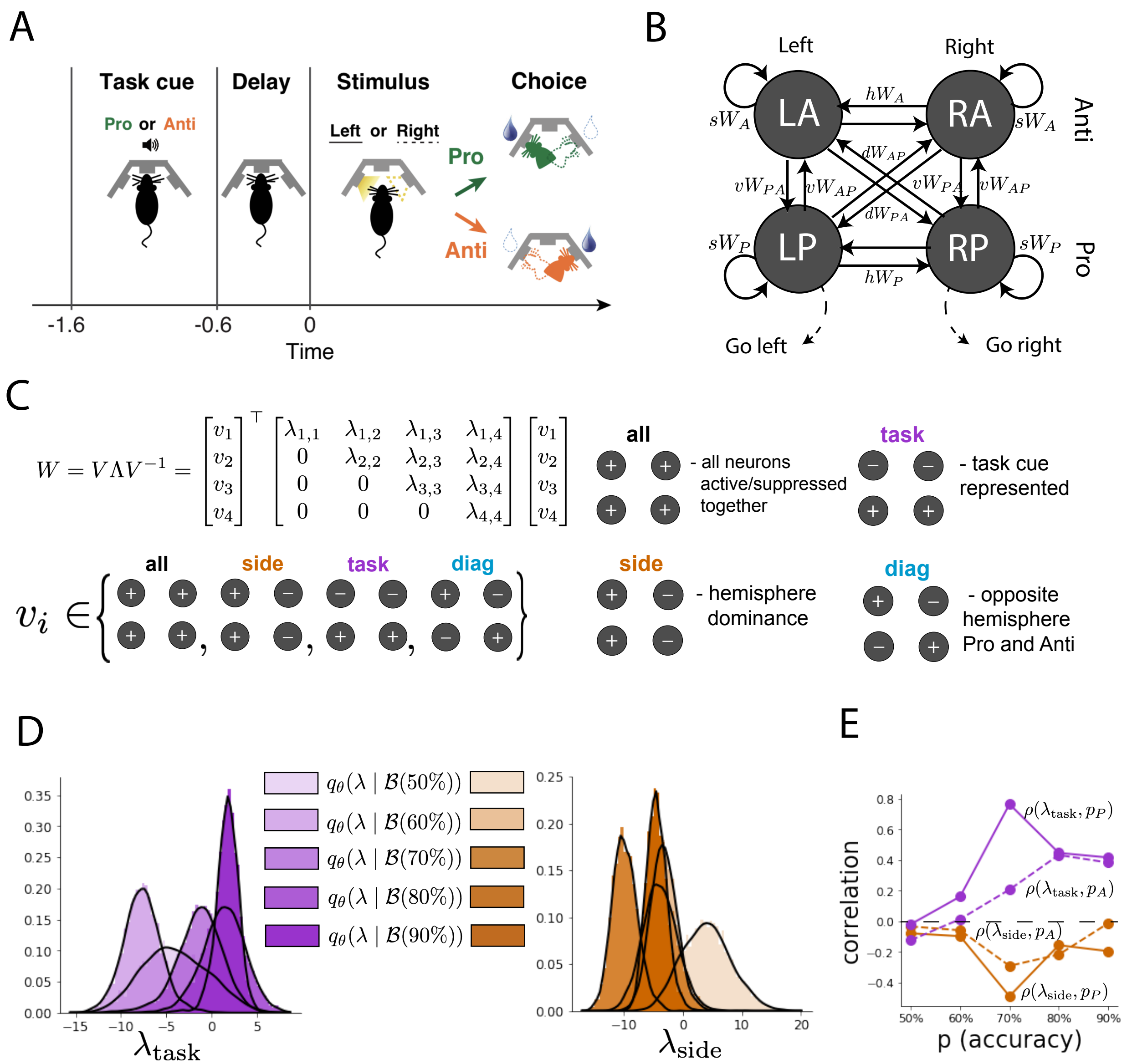

Figure 4: EPI reveals changes in SC [25] connectivity that control task accuracy. A. Rapid task switching behavioral paradigm (see text). B. Model of superior colliculus (SC). Neurons: LP - left pro, RP - right pro, LA - left anti, RA - right anti. Parameters: sW - self, hW - horizontal, vW -vertical, dW - diagonal weights. Subscripts P and A of connectivity weights indicate Pro or Anti populations, and e.g. vW_{PA} is a vertical weight from an Anti to a Pro population. C. The Schur decomposition of the weight matrix $W = V\Lambda V^{-1}$ is a unique decomposition with orthogonal V and upper triangular Λ. Schur modes: v_{all}, v_{task}, v_{side}, and v_{diag}. D. The marginal EPI distributions of the Schur eigenvalues at each level of task accuracy. E. The correlation of Schur eigenvalue with task performance in each learned EPI distribution.

of these inferred distributions, we followed the approach of Duan et al. by decomposing the connectivity matrix $W = V \Lambda V^{-1}$ in such a way (the Schur decomposition) that the basis vectors v_i are the same for all W (Fig. 4C). These basis vectors have intuitive roles in processing for this task, and are accordingly named the *all* mode - all neurons co-fluctuate, *side* mode - one side dominates the other, *task* mode - the Pro or Anti populations dominate the other, and *diag* mode - Pro- and Anti-populations of opposite hemispheres dominate the opposite pair. The corresponding eigenvalues (e.g. λ_{task}, which change according to W) indicate the degree to which activity along that mode is increased or decreased by W.

We found that for greater task accuracies, the task mode eigenvalue increases, indicating the importance of W to the task representation (Fig. 4D, purple; adjacent distributions from 60% to 90% have $p < 10^{-4}$, Mann-Whitney test with 50 estimates and 100 samples). Stepping from random chance (50%) networks to marginally task-performing (60%) networks, there is a marked decrease of the side mode eigenvalues (Fig. 4D, orange; $p < 10^{-4}$). Such side mode suppression relative to 50% remains in the models achieving greater accuracy, revealing its importance towards task performance. There were no interesting trends with task accuracy in the all or diag mode (hence not shown in Fig. 4). Importantly, we can conclude from our methodology that side mode suppression in W allows rapid task switching, and that greater task-mode representations in W increase accuracy. These hypotheses are confirmed by forward simulation of the SC model (Fig. 4E, see Section 5.2.3) suggesting experimentally testable predictions: increase in rapid task switching performance should be correlated with changes in effective connectivity corresponding to an increase in task mode and decrease in side mode eigenvalues.

3.5 Linking RNN connectivity to error

So far, each model we have studied was designed from fundamental biophysical principles, genetically- or functionally-defined neuron types. At a more abstract level of modeling, recurrent neural networks (RNNs) are high-dimensional dynamical models of computation that are becoming increasingly popular in neuroscience research [49]. In theoretical neuroscience, RNN dynamics usually follow the equation

$$\frac{dx}{dt} = -x + W\phi(x) + h, \tag{10}$$

where x is the network activity, W is the network connectivity, $\phi(\cdot) = \tanh(\cdot)$, and h is the input to the system. Such RNNs are trained to do a task from a systems neuroscience experiment, and then the unit activations of the trained RNN are compared to recorded neural activity. Fully-connected

RNNs with tens of thousands of parameters are challenging to characterize [50], especially making statistical inferences about their parameterization. Alternatively, we considered a rank-1, N-neuron RNN with connectivity consisting of the sum of a random and a structured component:

$$W = g\chi + \frac{1}{N} mn^\top. \tag{11}$$

The random component $g\chi$ has strength g, and random component weights are Gaussian distributed $\chi_{i,j} \sim \mathcal{N}(0, \frac{1}{N})$. The structured component $\frac{1}{N}mn\top$ has entries of m and n drawn from Gaussian distributions $m_i \sim \mathcal{N}(M_m, 1)$ and $n_i \sim \mathcal{N}(M_n, 1)$. Recent theoretical work derives the low-dimensional response properties of low-rank networks from statistical parameterizations of their connectivity, such as $z = [g, M_m, M_n]$ [26]. We used EPI to infer the parameterizations of rank-1 RNNs solving an example task, enabling discovery of properties of connectivity that result in different types of error in the computation.

The task we consider is Gaussian posterior conditioning: calculate the parameters of a posterior distribution induced by a prior $p(\mu_y) = \mathcal{N}(\mu_0 = 4, \sigma_0^2 = 1)$ and a likelihood $p(y|\mu_y) = \mathcal{N}(\mu_y, \sigma_y^2 = 1)$, given a single observation y. Conjugacy offers the result analytically; $p(\mu_y|y) = \mathcal{N}\left(\mu_{post}, \sigma^2_{post}\right)$, where:

$$\mu_{\text{post}} = \frac{\frac{\mu_0}{\sigma_0^2} + \frac{y}{\sigma_y^2}}{\frac{1}{\sigma_0^2} + \frac{1}{\sigma_y^2}} \qquad \sigma^2_{\text{post}} = \frac{1}{\frac{1}{\sigma_0^2} + \frac{1}{\sigma_y^2}}. \tag{12}$$

To solve this Gaussian posterior conditioning task, the RNN response to a constant input $h = yr + (n - M_n)$ must equal the posterior mean along readout vector r , where

$$\kappa_r = \frac{1}{N} \sum_{j=1}^{N} r_j \phi(x_j). \tag{13}$$

Additionally, the amount of chaotic variance Δ_T must equal the posterior variance. Theory for low-rank RNNs allows us to express κ_r and Δ_T in terms of each other through a solvable system of nonlinear equations (see Section 5.2.4) [26]. This theory facilitates the mathematical formalization of task execution into an emergent property, where the emergent property statistics of the RNN activity are κ_r and Δ_T, and the emergent property values are the ground truth posterior mean μ_{post} and variance σ^2_{post}:

$$\mathbb{E}\begin{bmatrix} \kappa_r \\ \Delta_T \\ (\kappa_r - \mu_{\text{post}})^2 \\ (\Delta_T^2 - \sigma^2_{\text{post}})^2 \end{bmatrix} = \begin{bmatrix} \mu_{\text{post}} \\ \sigma^2_{\text{post}} \\ 0.1 \\ 0.1 \end{bmatrix}. \tag{14}$$

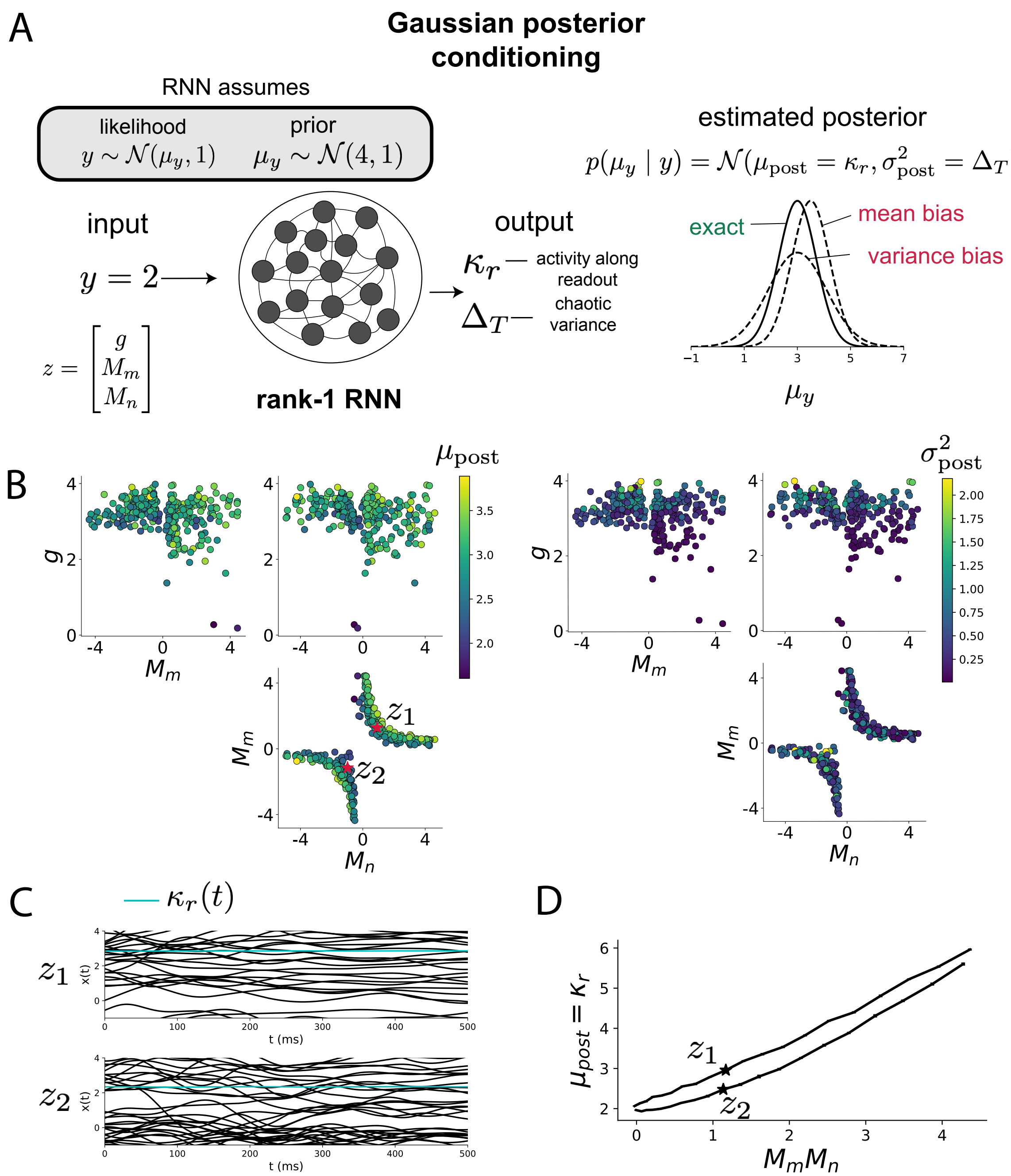

Figure 5: Sources of error in an RNN solving a simple task. A. (left) A rank-1 RNN executing a Gaussian posterior conditioning computation on μ_y. (right) Error in this computation can come from over- or under-estimating the posterior mean or variance. B. EPI distribution of rank-1 RNNs executing Gaussian posterior conditioning. Samples are colored by (left) posterior mean $\mu_{\text{post}} = \kappa_r$ and (right) posterior variance $\sigma^2_{\text{post}} = \Delta_T$ C. Finite-size network simulations of 2,000 neurons with parameters z_1 and z_2 sampled from the inferred distribution. Activity along readout κ_r (cyan) is stable despite chaotic fluctuations. D. The posterior mean computed by RNNs parameterized by z_1 and z_2 perturbed in the dimension of the product of M_m and M_n. Means and standard errors are shown across 10 realizations of 2,000-neuron networks.

We chose a substantial amount of variance in these emergent property statistics, so that the inferred distribution resulted in RNNs with a variety of errors in their solutions to the gaussian posterior conditioning problem.

EPI was used to learn distributions of RNN connectivity properties $z = [g, M_m, M_n]$ executing Gaussian posterior conditioning given an input of $y = 2$, where the true posterior is $\mu_{\text{post}} = 3$ and $\sigma_{\text{post}} = 0.5$ (Fig. 5A). We examined the nature of the over- and under-estimation of the posterior means (Fig. 5B left) and variances (Fig. 5B right) in the inferred distributions (300 samples). The symmetry in the M_m-M_n plane, suggests a degeneracy in the product of M_m and M_n (Fig. 5B). Indeed, $M_m M_n$ strongly determines the posterior mean ($r = 0.62$, $p < 10^{-4}$). Furthermore, the random strength g strongly determines the chaotic variance ($r = 0.56$, $p < 10^{-4}$). Neither of these observations were obvious from what mathematical analysis is available in networks of this type (see Section 5.2.4). While the link between random strength g and chaotic variance Δ_T (and resultingly posterior variance in this problem) is well-known [3], the distribution admits a novel hypothesis: the estimation of the posterior mean by the RNN increases with $M_m M_n$.

We tested this prediction by taking parameters z_1 and z_2 as representative samples from the positive and negative M_m-M_n quadrants, respectively. Instead of using the theoretical predictions shown in Figure 5B, we simulated finite-size realizations of these networks with 2,000 neurons (e.g. Fig. 5C). We perturbed these parameter choices by $M_m M_n$ clarifying that the posterior mean can be directly controlled in this way (Fig. 5D; $p < 10^{-4}$), see Section 5.2.4). Thus, EPI confers a clear picture of error in this computation: the product of the low rank vector means M_m and M_n modulates the estimated posterior mean while the random strength g modulates the estimated posterior variance. This novel procedure of inference on reduced parameterizations of RNNs conditioned on the emergent property of task execution is generalizable to other settings modeled in [26] like noisy integration and context-dependent decision making (Fig. S5).

4 Discussion

4.1 EPI is a general tool for theoretical neuroscience

Biologically realistic models of neural circuits are comprised of complex nonlinear differential equations, making traditional theoretical analysis and statistical inference intractable. We advance the capabilities of statistical inference in theoretical neuroscience by presenting EPI, a deep inference methodology for learning parameter distributions of theoretical models performing neural computation. We have demonstrated the utility of EPI on biological models (STG), intermediate-level models of interacting genetically- and functionally-defined neuron-types (V1, SC), and the most abstract of models (RNNs). We are able to condition both deterministic and stochastic models on low-level emergent properties like spiking frequency of membrane potentials, as well as high-level cognitive function like posterior conditioning. Technically, EPI is tractable when the emergent property statistics are continuously differentiable with respect to the model parameters, which is very often the case; this emphasizes the general applicability of EPI.

In this study, we have focused on applying EPI to low dimensional parameter spaces of models with low dimensional dynamical states. These choices were made to present the reader with a series of interpretable conclusions, which is more challenging in high dimensional spaces. In fact, EPI should scale reasonably to high dimensional parameter spaces, as the underlying technology has produced state-of-the-art performance on high-dimensional tasks such as texture generation [20]. Of course, increasing the dimensionality of the dynamical state of the model makes optimization more expensive, and there is a practical limit there as with any machine learning approach. Although, theoretical approaches (e.g. [26]) can be used to reason about the wholistic activity of such high dimensional systems by introducing some degree of additional structure into the model.

4.2 Novel hypotheses from EPI

In neuroscience, machine learning has primarily been used to reveal structure in large-scale neural datasets [51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61] (see review, [15]). Such careful inference procedures are developed for these statistical models allowing precise, quantitative reasoning, which clarifies the way data informs beliefs about the model parameters. However, these statistical models lack resemblance to the underlying biology, making it unclear how to go from the structure revealed by these methods, to the neural mechanisms giving rise to it. In contrast, theoretical neuroscience has focused on careful mechanistic modeling and the production of emergent properties of computation. The careful steps of *i.)* model design and *ii.)* emergent property definition, are followed by *iii.)* practical inference methods resulting in an opaque characterization of the way model parameters govern computation. In this work, we replaced this opaque procedure of parameter identification in theoretical neuroscience with emergent property inference, opening the door to careful inference in careful models of neural computation.

Biologically realistic models of neural circuits often prove formidable to analyze. Two main factors contribute to the difficulty of this endeavor. First, in most neural circuit models, the number of parameters scales quadratically with the number of neurons, limiting analysis of its parameter space. Second, even in low dimensional circuits, the structure of the parametric regimes governing emergent properties is intricate. For example, these circuit models can support more than one steady state [62] and non-trivial dynamics on strange attractors [63].

In Section 3.3, we advanced the tractability of low-dimensional neural circuit models by showing that EPI offers insights about cell-type specific input-responsivity that cannot be afforded through the available linear analytical methods [24, 44, 45]. By flexibly conditioning this V1 model on different emergent properties, we performed an exploratory analysis of a *model* rather than a dataset, generating a set of testable hypotheses, which were proved out. Furthermore, exploratory analyses can be directed towards formulating hypotheses of a specific form. For example, model parameter dependencies on behavioral performance can be assessed by using EPI to condition on various levels of task accuracy (See Section 3.4). This analysis identified experimentally testable predictions (proved out *in-silico*) of patterns of effective connectivity in SC that should be correlated with increased performance.

In our final analysis, we presented a novel procedure for doing statistical inference on interpretable parameterizations of RNNs executing simple tasks. Specifically, we analyzed RNNs solving a posterior conditioning problem in the spirit of [64, 65]. This methodology relies on recently extended theory of responses in random neural networks with low-rank structure [26]. While we focused on rank-1 RNNs, which were sufficient for solving this task, this inference procedure generalizes to RNNs of greater rank necessary for more complex tasks. The ability to apply the probabilistic model selection toolkit to RNNs should prove invaluable as their use in neuroscience increases.

EPI leverages deep learning technology for neuroscientific inquiry in a categorically different way than approaches focused on training neural networks to execute behavioral tasks [66]. These works focus on examining optimized deep neural networks while considering the objective function, learning rule, and architecture used. This endeavor efficiently obtains sets of parameters that can be reasoned about with respect to such considerations, but lacks the careful probabilistic treatment of parameter inference in EPI. These approaches can be used complementarily to enhance the practice of theoretical neuroscience.

Acknowledgements:

This work was funded by NSF Graduate Research Fellowship, DGE-1644869, McKnight Endowment Fund, NIH NINDS 5R01NS100066, Simons Foundation 542963, NSF NeuroNex Award, DBI-1707398, The Gatsby Charitable Foundation, Simons Collaboration on the Global Brain Postdoctoral Fellowship, Chinese Postdoctoral Science Foundation, and International Exchange Program Fellowship. Helpful conversations were had with Francesca Mastrogiuseppe, Srdjan Ostojic, James Fitzgerald, Stephen Baccus, Dhruva Raman, Liam Paninski, and Larry Abbott.

Data availability statement:

The datasets generated during and/or analysed during the current study are available from the corresponding author upon reasonable request.

Code availability statement:

The software written for the current study is available from the corresponding author upon reasonable request.

References

[1] Larry F Abbott. Theoretical neuroscience rising. *Neuron*, 60(3):489–495, 2008.

[2] John J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558, 1982.

[3] Haim Sompolinsky, Andrea Crisanti, and Hans-Jurgen Sommers. Chaos in random neural networks. *Physical review letters*, 61(3):259, 1988.

[4] Misha V Tsodyks, William E Skaggs, Terrence J Sejnowski, and Bruce L McNaughton. Paradoxical effects of external modulation of inhibitory interneurons. *Journal of neuroscience*, 17(11):4382–4388, 1997.

[5] Kong-Fatt Wong and Xiao-Jing Wang. A recurrent network mechanism of time integration in perceptual decisions. *Journal of Neuroscience*, 26(4):1314–1328, 2006.

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

[7] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and variational inference in deep latent gaussian models. *International Conference on Machine Learning*, 2014.

[8] Yuanjun Gao, Evan W Archer, Liam Paninski, and John P Cunningham. Linear dynamical neural population models through nonlinear embeddings. In *Advances in neural information processing systems*, pages 163–171, 2016.

[9] Yuan Zhao and Il Memming Park. Recursive variational bayesian dual estimation for nonlinear dynamics and non-gaussian observations. *stat*, 1050:27, 2017.

[10] Gabriel Barello, Adam Charles, and Jonathan Pillow. Sparse-coding variational auto-encoders. *bioRxiv*, page 399246, 2018.

[11] Chethan Pandarinath, Daniel J O'Shea, Jasmine Collins, Rafal Jozefowicz, Sergey D Stavisky, Jonathan C Kao, Eric M Trautmann, Matthew T Kaufman, Stephen I Ryu, Leigh R Hochberg, et al. Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature methods*, page 1, 2018.

[12] Alexander B Wiltschko, Matthew J Johnson, Giuliano Iurilli, Ralph E Peterson, Jesse M Katon, Stan L Pashkovski, Victoria E Abraira, Ryan P Adams, and Sandeep Robert Datta. Mapping sub-second structure in mouse behavior. *Neuron*, 88(6):1121–1135, 2015.

[13] Matthew J Johnson, David K Duvenaud, Alex Wiltschko, Ryan P Adams, and Sandeep R Datta. Composing graphical models with neural networks for structured representations and fast inference. In *Advances in neural information processing systems*, pages 2946–2954, 2016.

[14] Eleanor Batty, Matthew Whiteway, Shreya Saxena, Dan Biderman, Taiga Abe, Simon Musall, Winthrop Gillis, Jeffrey Markowitz, Anne Churchland, John Cunningham, et al. Behavenet: nonlinear embedding and bayesian neural decoding of behavioral videos. *Advances in Neural Information Processing Systems*, 2019.

[15] Liam Paninski and John P Cunningham. Neural data science: accelerating the experiment-analysis-theory cycle in large-scale neuroscience. *Current opinion in neurobiology*, 50:232–241, 2018.

[16] Mark K Transtrum, Benjamin B Machta, Kevin S Brown, Bryan C Daniels, Christopher R Myers, and James P Sethna. Perspective: Sloppiness and emergent theories in physics, biology, and beyond. *The Journal of chemical physics*, 143(1):07B201_1, 2015.

[17] Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. *International Conference on Machine Learning*, 2015.

[18] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

[19] George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density estimation. In *Advances in Neural Information Processing Systems*, pages 2338–2347, 2017.

[20] Gabriel Loaiza-Ganem, Yuanjun Gao, and John P Cunningham. Maximum entropy flow networks. *International Conference on Learning Representations*, 2017.

[21] Dustin Tran, Rajesh Ranganath, and David Blei. Hierarchical implicit models and likelihood-free variational inference. In *Advances in Neural Information Processing Systems*, pages 5523–5533, 2017.

[22] Mark S Goldman, Jorge Golowasch, Eve Marder, and LF Abbott. Global structure, robustness, and modulation of neuronal models. *Journal of Neuroscience*, 21(14):5229–5238, 2001.

[23] Gabrielle J Gutierrez, Timothy O'Leary, and Eve Marder. Multiple mechanisms switch an electrically coupled, synaptically inhibited neuron between competing rhythmic oscillators. *Neuron*, 77(5):845–858, 2013.

[24] Ashok Litwin-Kumar, Robert Rosenbaum, and Brent Doiron. Inhibitory stabilization and visual coding in cortical circuits with multiple interneuron subtypes. *Journal of neurophysiology*, 115(3):1399–1409, 2016.

[25] Chunyu A Duan, Marino Pagan, Alex T Piet, Charles D Kopec, Athena Akrami, Alexander J Riordan, Jeffrey C Erlich, and Carlos D Brody. Collicular circuits for flexible sensorimotor routing. *bioRxiv*, page 245613, 2018.

[26] Francesca Mastrogiuseppe and Srdjan Ostojic. Linking connectivity, dynamics, and computations in low-rank recurrent neural networks. *Neuron*, 99(3):609–623, 2018.

[27] Sean R Bittner, Agostina Palmigiano, Kenneth D Miller, and John P Cunningham. Degenerate solution networks for theoretical neuroscience. *Computational and Systems Neuroscience Meeting (COSYNE), Lisbon, Portugal*, 2019.

[28] Sean R Bittner, Alex T Piet, Chunyu A Duan, Agostina Palmigiano, Kenneth D Miller, Carlos D Brody, and John P Cunningham. Examining models in theoretical neuroscience with degenerate solution networks. *Bernstein Conference 2019, Berlin, Germany*, 2019.

[29] Marcel Nonnenmacher, Pedro J Goncalves, Giacomo Bassetto, Jan-Matthis Lueckmann, and Jakob H Macke. Robust statistical inference for simulation-based models in neuroscience. In *Bernstein Conference 2018, Berlin, Germany*, 2018.

[30] Deistler Michael, , Pedro J Goncalves, Kaan Oecal, and Jakob H Macke. Statistical inference for analyzing sloppiness in neuroscience models. In *Bernstein Conference 2019, Berlin, Germany*, 2019.

[31] Jan-Matthis Lueckmann, Pedro J Goncalves, Giacomo Bassetto, Kaan Öcal, Marcel Nonnenmacher, and Jakob H Macke. Flexible statistical inference for mechanistic models of neural dynamics. In *Advances in Neural Information Processing Systems*, pages 1289–1299, 2017.

[32] Eve Marder and Vatsala Thirumalai. Cellular, synaptic and network effects of neuromodulation. *Neural Networks*, 15(4-6):479–493, 2002.

[33] Astrid A Prinz, Dirk Bucher, and Eve Marder. Similar network activity from disparate circuit parameters. *Nature neuroscience*, 7(12):1345, 2004.

[34] Edwin T Jaynes. Information theory and statistical mechanics. *Physical review*, 106(4):620, 1957.

[35] Gamaleldin F Elsayed and John P Cunningham. Structure in neural population recordings: an expected byproduct of simpler phenomena? *Nature neuroscience*, 20(9):1310, 2017.

[36] Cristina Savin and Gašper Tkačik. Maximum entropy models as a tool for building precise neural controls. *Current opinion in neurobiology*, 46:120–126, 2017.

[37] Brendan K Murphy and Kenneth D Miller. Balanced amplification: a new mechanism of selective amplification of neural activity patterns. *Neuron*, 61(4):635–648, 2009.

[38] Hirofumi Ozeki, Ian M Finn, Evan S Schaffer, Kenneth D Miller, and David Ferster. Inhibitory stabilization of the cortical network underlies visual surround suppression. *Neuron*, 62(4):578–592, 2009.

[39] Daniel B Rubin, Stephen D Van Hooser, and Kenneth D Miller. The stabilized supralinear network: a unifying circuit motif underlying multi-input integration in sensory cortex. *Neuron*, 85(2):402–417, 2015.

[40] Henry Markram, Maria Toledo-Rodriguez, Yun Wang, Anirudh Gupta, Gilad Silberberg, and Caizhi Wu. Interneurons of the neocortical inhibitory system. *Nature reviews neuroscience*, 5(10):793, 2004.

[41] Bernardo Rudy, Gordon Fishell, SooHyun Lee, and Jens Hjerling-Leffler. Three groups of interneurons account for nearly 100% of neocortical gabaergic neurons. *Developmental neurobiology*, 71(1):45–61, 2011.

[42] Robin Tremblay, Soohyun Lee, and Bernardo Rudy. GABAergic Interneurons in the Neocortex: From Cellular Properties to Circuits. *Neuron*, 91(2):260–292, 2016.

[43] Carsten K Pfeffer, Mingshan Xue, Miao He, Z Josh Huang, and Massimo Scanziani. Inhibition of inhibition in visual cortex: the logic of connections between molecularly distinct interneurons. *Nature Neuroscience*, 16(8):1068, 2013.

[44] Luis Carlos Garcia Del Molino, Guangyu Robert Yang, Jorge F. Mejias, and Xiao Jing Wang. Paradoxical response reversal of top- down modulation in cortical circuits with three interneuron types. *Elife*, 6:1–15, 2017.

[45] Guang Chen, Carl Van Vreeswijk, David Hansel, and David Hansel. Mechanisms underlying the response of mouse cortical networks to optogenetic manipulation. 2019.

[46] (2018) Allen Institute for Brain Science. Layer 4 model of v1. available from: https://portal.brain-map.org/explore/models/l4-mv1.

[47] Yazan N Billeh, Binghuang Cai, Sergey L Gratiy, Kael Dai, Ramakrishnan Iyer, Nathan W Gouwens, Reza Abbasi-Asl, Xiaoxuan Jia, Joshua H Siegle, Shawn R Olsen, et al. Systematic integration of structural and functional data into multi-scale models of mouse primary visual cortex. *bioRxiv*, page 662189, 2019.

[48] Chunyu A Duan, Jeffrey C Erlich, and Carlos D Brody. Requirement of prefrontal and midbrain regions for rapid executive control of behavior in the rat. *Neuron*, 86(6):1491–1503, 2015.

[49] Omri Barak. Recurrent neural networks as versatile tools of neuroscience research. *Current opinion in neurobiology*, 46:1–6, 2017.

[50] David Sussillo and Omri Barak. Opening the black box: low-dimensional dynamics in high-dimensional recurrent neural networks. *Neural computation*, 25(3):626–649, 2013.

[51] Robert E Kass and Valérie Ventura. A spike-train probability model. *Neural computation*, 13(8):1713–1720, 2001.

[52] Emery N Brown, Loren M Frank, Dengda Tang, Michael C Quirk, and Matthew A Wilson. A statistical paradigm for neural spike train decoding applied to position prediction from

ensemble firing patterns of rat hippocampal place cells. *Journal of Neuroscience*, 18(18):7411–7425, 1998.

[53] Liam Paninski. Maximum likelihood estimation of cascade point-process neural encoding models. *Network: Computation in Neural Systems*, 15(4):243–262, 2004.

[54] Wilson Truccolo, Uri T Eden, Matthew R Fellows, John P Donoghue, and Emery N Brown. A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects. *Journal of neurophysiology*, 93(2):1074–1089, 2005.

[55] Shaul Druckmann, Yoav Banitt, Albert A Gidon, Felix Schürmann, Henry Markram, and Idan Segev. A novel multiple objective optimization framework for constraining conductance-based neuron models by experimental data. *Frontiers in neuroscience*, 1:1, 2007.

[56] M Yu Byron, John P Cunningham, Gopal Santhanam, Stephen I Ryu, Krishna V Shenoy, and Maneesh Sahani. Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. In *Advances in neural information processing systems*, pages 1881–1888, 2009.

[57] Il Memming Park and Jonathan W Pillow. Bayesian spike-triggered covariance analysis. In *Advances in neural information processing systems*, pages 1692–1700, 2011.

[58] Kenneth W Latimer, Jacob L Yates, Miriam LR Meister, Alexander C Huk, and Jonathan W Pillow. Single-trial spike trains in parietal cortex reveal discrete steps during decision-making. *Science*, 349(6244):184–187, 2015.

[59] Kaushik J Lakshminarasimhan, Marina Petsalis, Hyeshin Park, Gregory C DeAngelis, Xaq Pitkow, and Dora E Angelaki. A dynamic bayesian observer model reveals origins of bias in visual path integration. *Neuron*, 99(1):194–206, 2018.

[60] Lea Duncker, Gergo Bohner, Julien Boussard, and Maneesh Sahani. Learning interpretable continuous-time models of latent stochastic dynamical systems. *Proceedings of the 36th International Conference on Machine Learning*, 2019.

[61] Josef Ladenbauer, Sam McKenzie, Daniel Fine English, Olivier Hagens, and Srdjan Ostojic. Inferring and validating mechanistic models of neural microcircuits based on spike-train data. *Nature Communications*, 10(4933), 2019.

[62] Nataliya Kraynyukova and Tatjana Tchumatchenko. Stabilized supralinear network can give rise to bistable, oscillatory, and persistent activity. *Proceedings of the National Academy of Sciences*, 115(13):3464–3469, 2018.

[63] Katherine Morrison, Anda Degeratu, Vladimir Itskov, and Carina Curto. Diversity of emergent dynamics in competitive threshold-linear networks: a preliminary report. *arXiv preprint arXiv:1605.04463*, 2016.

[64] Xaq Pitkow and Dora E Angelaki. Inference in the brain: statistics flowing in redundant population codes. *Neuron*, 94(5):943–953, 2017.

[65] Rodrigo Echeveste, Laurence Aitchison, Guillaume Hennequin, and Máté Lengyel. Cortical-like dynamics in recurrent circuits optimized for sampling-based probabilistic inference. *bioRxiv*, page 696088, 2019.

[66] Blake A Richards and et al. A deep learning framework for neuroscience. *Nature Neuroscience*, 2019.

[67] David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017.

[68] Rajesh Ranganath, Sean Gerrish, and David Blei. Black box variational inference. In *Artificial Intelligence and Statistics*, pages 814–822, 2014.

[69] Martin J Wainwright, Michael I Jordan, et al. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.

[70] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 2015.

[71] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *Proceedings of the 5th International Conference on Learning Representations*, 2017.

[72] Nicolas Brunel. Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. *Journal of computational neuroscience*, 8(3):183–208, 2000.

[73] Herbert Jaeger and Harald Haas. Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *science*, 304(5667):78–80, 2004.

[74] David Sussillo and Larry F Abbott. Generating coherent patterns of activity from chaotic neural networks. *Neuron*, 63(4):544–557, 2009.

5 Methods

5.1 Emergent property inference (EPI)

Emergent property inference (EPI) learns distributions of theoretical model parameters that produce emergent properties of interest by combining ideas from maximum entropy flow networks (MEFNs) [20] and likelihood-free variational inference (LFVI) [21]. Consider model parameterization z and data x which has an intractable likelihood $p(x \mid z)$ defined by a model simulator of which samples are available $x \sim p(x \mid z)$. EPI optimizes a distribution $q_\theta(z)$ (itself parameterized by θ) of model parameters z to produce an emergent property of interest $\mathcal{B}$,

$$\mathcal{B} \quad \triangleq \quad \mathbb{E}_{z \sim q_\theta}\left[\mathbb{E}_{x \sim p(x|z)}\left[T(x)\right]\right] = \mu. \tag{15}$$

Precisely, the emergent property statistics $T(x)$ must equal the emergent property values μ, in expectation over the EPI distribution of parameters $q_\theta(z)$ and the distribution of simulated activity $p(x \mid z)$. This is a viable way to represent emergent properties in theoretical models, as we have demonstrated in the main text, and enables the EPI optimization.

With EPI, we use deep probability distributions to learn flexible approximations to model parameter distributions $q_\theta(z)$. In deep probability distributions, a simple random variable $w \sim q_0(w)$ is mapped deterministically via a sequence of deep neural network layers (f_1, .. f_l) parameterized by weights and biases θ to the support of the distribution of interest:

$$z = f_\theta(\omega) = f_l(..f_1(w)). \tag{16}$$

Given a simulator defined by a theoretical model $x \sim p(x \mid z)$ and some emergent property of interest $\mathcal{B}$, $q_\theta(z)$ is optimized via the neural network parameters θ to find a maximally entropic distribution q_θ^* within the deep variational family $\mathcal{Q}$ producing the emergent property:

$$\begin{aligned} q_\theta^*(z) &= \operatorname*{argmax}_{q_\theta \in Q} H(q_\theta(z)) \\ &\text{s.t. } \mathbb{E}_{z \sim q_\theta}\left[\mathbb{E}_{x \sim p(x|z)}\left[T(x)\right]\right] = \mu. \end{aligned} \tag{17}$$

Since we are optimizing parameters θ of our deep probability distribution with respect to the entropy $H(q_\theta(z))$, we must take gradients with respect to the log probability density of samples from the deep probability distribution. Entropy of $q_\theta(z)$ can be expressed as an expectation of the negative log density of parameter samples z over the randomness in the parameterless initial distribution q_0:

$$H(q_\theta(z)) = \int -q_\theta(z)\log(q_\theta(z))dz = \mathbb{E}_{z \sim q_\theta}\left[-\log(q_\theta(z))\right] = \mathbb{E}_{w \sim q_0}\left[-\log(q_\theta(f_\theta(w)))\right]. \tag{18}$$

Thus, the gradient of the entropy of the deep probability distribution can be estimated as an average of gradients of the log density of samples z:

$$\nabla_\theta H(q_\theta(z)) = \mathbb{E}_{w\sim q_0}\left[-\nabla_\theta \log(q_\theta(f_\theta(w)))\right]. \tag{19}$$

In EPI, MEFNs are purposed towards variational learning of model parameter distributions. A closely related methodology, variational inference, uses optimization to approximate posterior distributions [67]. Standard methods like stochastic gradient variational Bayes [6] or black box variational inference [68] simply do not work for inference in theoretical models of neural circuits, since they require tractable likelihoods $p(x \mid z)$. Work on likelihood-free variational inference (LFVI) [21], which like EPI seeks to do inference in models with intractable likelihoods, employs an addtional deep neural network as a ratio estimator, enabling an estimation of the optimization objective for variational inference. Like LFVI, EPI can be framed as variational inference (see Section 5.1.4). But, unlike LFVI, EPI uses a single deep network to learn a distribution and is optimized to produce an emergent property, rather than condition on data points. Optimizing the EPI objective is a technological challenge, the details of which we elaborate in Section 5.1.2. Before going through those details, we ground this optimization in a toy example.

5.1.1 Example: 2D LDS

To gain intuition for EPI, consider a two-dimensional linear dynamical system (2D LDS) model (Fig. S1A):

$$\tau \frac{dx}{dt} = Ax \tag{20}$$

with

$$A = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix}. \tag{21}$$

To run EPI with the dynamics matrix elements as the free parameters $z = [a_1, a_2, a_3, a_4]$ (fixing $\tau = 1$), the emergent property statistics $T(x)$ were chosen to contain the first and second moments of the oscillatory frequency, $2\pi\text{imag}(\lambda_1)$, and the growth/decay factor, $\text{real}(\lambda_1)$, of the oscillating system. λ_1 is the eigenvalue of greatest real part when the imaginary component is zero, and alternatively of positive imaginary component when the eigenvalues are complex conjugate pairs. To learn the distribution of real entries of A that produce a band of oscillating systems around 1Hz, we formalized this emergent property as $\text{real}(\lambda_1)$ having mean zero with variance 0.25^2, and the oscillation frequency $2\pi\text{imag}(\lambda_1)$ having mean $\omega = 1$ Hz with variance $(0.1\text{Hz})^2$:

$$\mathbb{E}\left[T(x)\right] \triangleq \mathbb{E}\begin{bmatrix} \text{real}(\lambda_1) \\ \text{imag}(\lambda_1) \\ (\text{real}(\lambda_1) - 0)^2 \\ (\text{imag}(\lambda_1) - 2\pi\omega)^2 \end{bmatrix} = \begin{bmatrix} 0.0 \\ 2\pi\omega \\ 0.25^2 \\ (2\pi 0.1)^2 \end{bmatrix} \triangleq \mu. \tag{22}$$

Unlike the models we presented in the main text, this model admits an analytical form for the mean emergent property statistics given parameter z, since the eigenvalues can be calculated using the quadratic formula:

$$\lambda = \frac{(\frac{a_1+a_4}{\tau}) \pm \sqrt{(\frac{a_1+a_4}{\tau})^2 + 4(\frac{a_2a_3-a_1a_4}{\tau})}}{2}. \tag{23}$$

Importantly, even though $\mathbb{E}_{x\sim p(x|z)}\left[T(x)\right]$ is calculable directly via a closed form function and does not require simulation, we cannot derive the distribution q_θ^* directly. This fact is due to the formally hard problem of the backward mapping: finding the natural parameters η from the mean parameters μ of an exponential family distribution [69]. Instead, we used EPI to approximate this distribution (Fig. S1B). We used a real-NVP normalizing flow architecture with four masks, two neural network layers of 15 units per mask, with batch normalization momentum 0.99, mapped onto a support of $z_i \in [-10, 10]$. (see Section 5.1.3).

Even this relatively simple system has nontrivial (though intuitively sensible) structure in the parameter distribution. To validate our method, we analytically derived the contours of the probability density from the emergent property statistics and values. In the a_1-a_4 plane, the black line at $\text{real}(\lambda_1) = \frac{a_1+a_4}{2} = 0$, dotted black line at the standard deviation $\text{real}(\lambda_1) = \frac{a_1+a_4}{2} \pm 0.25$, and the dotted gray line at twice the standard deviation $\text{real}(\lambda_1) = \frac{a_1+a_4}{2} \pm 0.5$ follow the contour of probability density of the samples (Fig. S2A). The distribution precisely reflects the desired statistical constraints and model degeneracy in the sum of a_1 and a_4. Intuitively, the parameters equivalent with respect to emergent property statistic $\text{real}(\lambda_1)$ have similar log densities.

To explain the bimodality of the EPI distribution, we examined the imaginary component of λ_1. When $\text{real}(\lambda_1) = \frac{a_1+a_4}{2} = 0$, we have

$$\text{imag}(\lambda_1) = \begin{cases} \sqrt{\frac{a_1a_4-a_2a_3}{\tau}}, & \text{if } a_1a_4 < a_2a_3 \\ 0 & \text{otherwise} \end{cases}. \tag{24}$$

When $\tau = 1$ and $a_1a_4 > a_2a_3$ (center of distribution above), we have the following equation for the

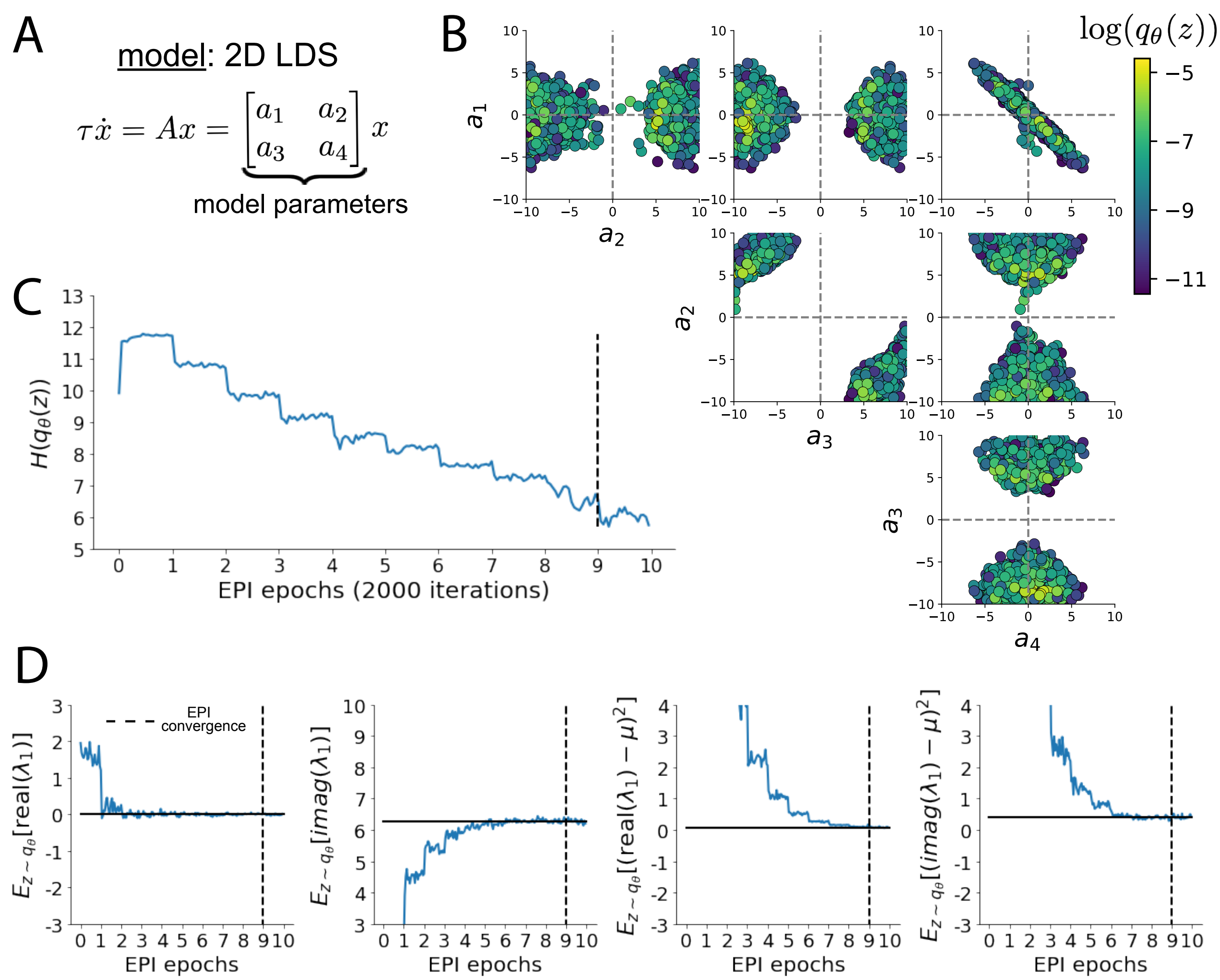

Fig. S1: A. Two-dimensional linear dynamical system model, where real entries of the dynamics matrix A are the parameters. B. The EPI distribution for a two-dimensional linear dynamical system with $\tau = 1$ that produces an average of 1Hz oscillations with some small amount of variance. Dashed lines indicate the parameter axes. C. Entropy throughout the optimization. At the beginning of each augmented Lagrangian epoch (2,000 iterations), the entropy dipped due to the shifted optimization manifold where emergent property constraint satisfaction is increasingly weighted. D. Emergent property moments throughout optimization. At the beginning of each augmented Lagrangian epoch, the emergent property moments adjust closer to their constraints.

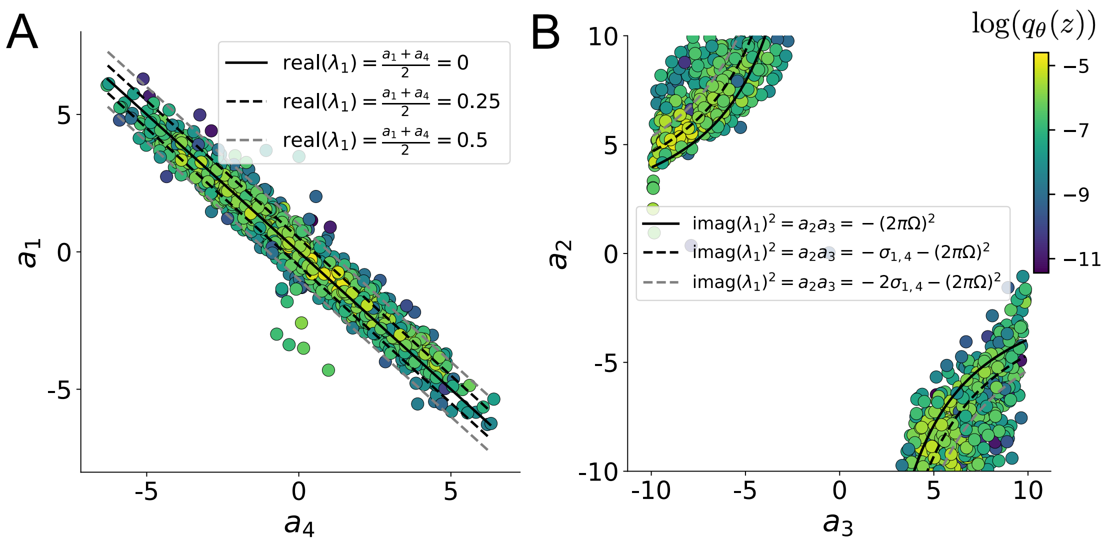

Fig. S2: A. Probability contours in the a_1-a_4 plane were derived from the relationship to emergent property statistic of growth/decay factor real(λ_1). B. Probability contours in the a_2-a_3 plane were derived from the emergent property statistic of oscillation frequency $2\pi\mathrm{imag}(\lambda_1)$.

other two dimensions:

$$\mathrm{imag}(\lambda_1)^2 = a_1a_4 - a_2a_3 \tag{25}$$

Since we constrained $\mathbb{E}_{z\sim q_\theta}[\mathrm{imag}(\lambda)] = 2\pi$ (with $\omega = 1$), we can plot contours of the equation $\mathrm{imag}(\lambda_1)^2 = a_1a_4 - a_2a_3 = (2\pi)^2$ for various a_1a_4 (Fig. S2B). With $\sigma_{1,4} = \mathbb{E}_{z\sim q_\theta}(|a_1a_4 - E_{q_\theta}[a_1a_4]|)$, we show the contours as $a_1a_4 = 0$ (black), $a_1a_4 = -\sigma_{1,4}$ (black dotted), and $a_1a_4 = -2\sigma_{1,4}$ (grey dotted). This validates the curved structure of the inferred distribution learned through EPI. We took steps in negative standard deviation of a_1a_4 (dotted and gray lines), since there are few positive values a_1a_4 in the learned distribution. Subtler combinations of model and emergent property will have more complexity, further motivating the use of EPI for understanding these systems. As we expect, the distribution results in samples of two-dimensional linear systems oscillating near 1Hz (Fig. S3).

5.1.2 Augmented Lagrangian optimization

To optimize $q_\theta(z)$ in Equation 17, the constrained optimization is executed using the augmented Lagrangian method. The following objective is minimized:

$$L(\theta; \eta, c) = -H(q_\theta) + \eta^\top R(\theta) + \frac{c}{2}||R(\theta)||^2 \tag{26}$$

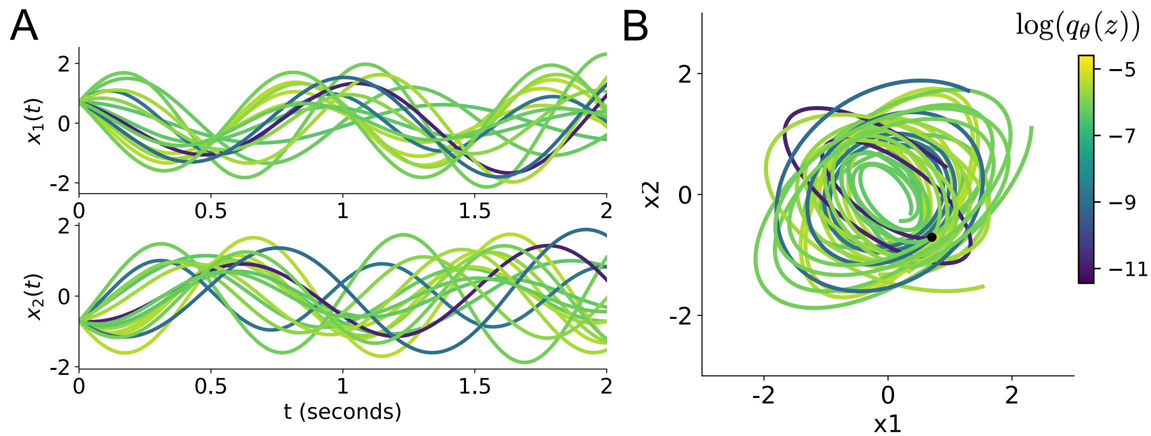

Fig. S3: Sampled dynamical systems $z \sim q_\theta(z)$ and their simulated activity from $x(0) = [\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2}]$ colored by log probability. A. Each dimension of the simulated trajectories throughout time. B The simulated trajectories in phase space.

where $R(\theta) = \mathbb{E}_{z \sim q_\theta}\left[\mathbb{E}_{x \sim p(x|z)}\left[T(x) - \mu\right]\right]$, $\eta \in \mathbb{R}^m$ are the Lagrange multipliers where $m = |\mu| = |T(x)|$, and c is the penalty coefficient. These Lagrange multipliers are closely related to the natural parameters of exponential families (see Section 5.1.4). Deep neural network weights and biases θ of the deep probability distribution are optimized according to Equation 26 using the Adam optimizer with its standard parameterization [70]. η is initialized to the zero vector and adapted following each augmented Lagrangian epoch, which is a period of optimization with fixed (η, c) for a given number of stochastic optimization iterations. A low value of c is used initially, and conditionally increased after each epoch based on constraint error reduction. For example, the initial value of c was $c_0 = 10^{-3}$ during EPI with the oscillating 2D LDS (Fig. S1C). The penalty coefficient is updated based on the result of a hypothesis test regarding the reduction in constraint violation. The p-value of $\mathbb{E}[||R(\theta_{k+1})||] > \gamma \mathbb{E}\left[||R(\theta_k)||\right]$ is computed, and c_{k+1} is updated to βc_k with probability $1-p$. The other update rule is $\eta_{k+1} = \eta_k + c_k \frac{1}{n}\sum_{i=1}^{n}(T(x^{(i)}) - \mu)$ given a batch size n. Throughout the study, $\beta = 4.0$, $\gamma = 0.25$, and the batch size was a hyperparameter, which varied according to the application of EPI.

The intention is that c and η start at values encouraging entropic growth early in optimization. With each training epoch in which the update rule for c is invoked by unsatisfactory constraint error reduction, the constraint satisfaction terms are increasingly weighted, resulting in a decreased entropy. This encourages the discovery of suitable regions of parameter space, and the subsequent refinement of the distribution to produce the emergent property. In the oscillating 2D LDS example, each augmented Lagrangian epoch ran for 2,000 iterations (Fig. S1C-D). Notice the initial entropic growth, and subsequent reduction upon each update of η and c. The momentum parameters of the Adam optimizer were reset at the end of each augmented Lagrangian epoch.

Rather than starting optimization from some θ drawn from a randomized distribution, we found that initializing $q_\theta(z)$ to approximate an isotropic Gaussian distribution conferred more stable, consistent optimization. The parameters of the Gaussain initialization were chosen on an application-specific basis. Throughout the study, we chose isotropic Gaussian initializations with mean μ_{init} at the center of the distribution support and some standard deviation σ_{init}, except for one case, where an initialization informed by random search was used (see Section 5.2.2).

To assess whether EPI distribution $q_\theta(z)$ produces the emergent property, we defined a hypothesis testing convergence criteria. The algorithm has converged when a null hypothesis test of constraint violations $R(\theta)_i$ being zero is accepted for all constraints $i \in \{1, ..., m\}$ at a significance threshold $\alpha = 0.05$. This significance threshold is adjusted through Bonferroni correction according to the number of constraints m. The p-values for each constraint are calculated according to a two-tailed nonparametric test, where 200 estimations of the sample mean $R(\theta)^i$ are made from k resamplings of z from a finite sample of size n taken at the end of the augmented Lagrangian epoch. k is determined by a fraction of the batch size ν, which varies according to the application. In the linear two-dimensional system example, we used a batch size of $n = 1000$ and set $\nu = 0.1$ resulting in convergence after the ninth epoch of optimization. (Fig. S1C-D black dotted line).

When assessing the suitability of EPI for a particular modeling question, there are some important technical considerations. First and foremost, as in any optimization problem, the defined emergent property should always be appropriately conditioned (constraints should not have wildly different units). Furthermore, if the program is underconstrained (not enough constraints), the distribution grows (in entropy) unstably unless mapped to a finite support. If overconstrained, there is no parameter set producing the emergent property, and EPI optimization will fail (appropriately). Next, one should consider the computational cost of the gradient calculations. In the best circumstance, there is a simple, closed form expression (e.g. Section 5.1.1) for the emergent property statistic given the model parameters. On the other end of the spectrum, many forward simulation iterations may be required before a high quality measurement of the emergent property statistic is available (e.g. Section 5.2.1). In such cases, optimization will be expensive.

5.1.3 Normalizing flows

Deep probability models typically consist of several layers of fully connected neural networks. When each neural network layer is restricted to be a bijective function, the sample density can be calculated using the change of variables formula at each layer of the network. For $z' = f(z)$,

$$q(z') = q(f^{-1}(z')) \left| \det \frac{\partial f^{-1}(z')}{\partial z'} \right| = q(z) \left| \det \frac{\partial f(z)}{\partial z} \right|^{-1}. \tag{27}$$

However, this computation has cubic complexity in dimensionality for fully connected layers. By restricting our layers to normalizing flows [17] – bijective functions with fast log determinant Jacobian computations, we can tractably optimize deep generative models with objectives that are a function of sample density, like entropy. Most of our analyses use either a planar flow [17] or real NVP [71], which have proven effective in our architecture searches. Planar flow architectures are specified by the number of planar bijection layers used, while real NVP architectures are specified by the number of masks, neural network layers per mask, units per layer, and batch normalization momentum parameter.

5.1.4 Emergent property inference as variational inference in an exponential family

Now that we have fully described the EPI method, we consider its broader contextualization as a statistical method and its relation to Bayesian inference. In Bayesian inference a prior belief about model parameters z is formalized into a prior distribution $p(z)$, and the statistical model capturing the effect of z on observed data points x is formalized in the likelihood distribution $p(x \mid z)$. In Bayesian inference, we obtain a posterior distribution $p(z \mid x)$, which captures how the data inform our knowledge of model parameters using Bayes' rule:

$$p(z \mid x) = \frac{p(x \mid z)p(z)}{p(x)}. \tag{28}$$

The posterior disribution is analytically available when the prior is conjugate with the likehood. However, conjugacy is rare in practice, and alternative methods, such as variational inference [67], are utilized.

As we compare EPI to variational inference, it is important to consider that EPI is a maximum entropy method, and that maximum entropy methods have a fundamental relationship with exponential family distributions. A maximum entropy distribution of form:

$$\begin{aligned} p^*(z) = \operatorname*{argmax}_{p \in \mathcal{P}} H(p(z)) \\ \text{s.t. } \mathbb{E}_{z \sim p}\left[T(z)\right] = \mu. \end{aligned} \tag{29}$$

will have probability density in the exponential family:

$$p^*(z) \propto \exp(\eta^\top T(z)). \tag{30}$$

The mappings between the mean parameterization μ and the natural parameterization η are formally hard to identify [69].

Now, consider the goal of doing variational inference with an exponential family posterior distribution $p(z \mid x)$. We use the following abbreviated notation to collect the base measure $b(z)$ and sufficient statistics $T(z)$ into $\tilde{T}(z)$ and likewise concatenate a 1 onto the end of the natural parameter $\tilde{\eta}(x)$. The log normalizing constant $A(\eta(x))$ remains unchanged:

$$\begin{aligned} p(z \mid x) = b(z) \exp\left(\eta(x)^\top T(z) - A(\eta(x))\right) &= \exp\left(\begin{bmatrix} \eta(x) \\ 1 \end{bmatrix}^\top \begin{bmatrix} T(z) \\ b(z) \end{bmatrix} - A(\eta(x)) \right). \\ &= \exp\left(\tilde{\eta(x)}^\top \tilde{T}(z) - A(\eta(x))\right) \end{aligned} \tag{31}$$

Variational inference with an exponential family posterior distribution uses optimization to minimize the following divergence [67]:

$$q_\theta^* = \operatorname*{argmin}_{q_\theta \in Q} KL(q_\theta \mid\mid p(z \mid x)). \tag{32}$$

$q_\theta(z)$ is the variational approximation to the posterior with variational parameters θ. We can write this KL divergence in terms of entropy of the variational approximation:

$$KL(q_\theta \mid\mid p(z \mid x)) = \mathbb{E}_{z \sim q_\theta}\left[\log(q_\theta(z))\right] - \mathbb{E}_{z \sim q_\theta}\left[\log(p(z \mid x))\right] \tag{33}$$

$$= -H(q_\theta) - \mathbb{E}_{z \sim q_\theta}\left[\tilde{\eta}(x)^\top \tilde{T}(z) - A(\eta(x))\right]. \tag{34}$$

As far as the variational optimization is concerned, the log normalizing constant is independent of $q_\theta(z)$, so it can be dropped

$$\operatorname*{argmin}_{q_\theta \in Q} KL(q_\theta \mid\mid p(z \mid x)) = \operatorname*{argmin}_{q_\theta \in Q} -H(q_\theta) - \mathbb{E}_{z \sim q_\theta}\left[\tilde{\eta}(x)^\top \tilde{T}(z)\right]. \tag{35}$$

Further, we can write the objective in terms of the first moment of the sufficient statistics $\mu = \mathbb{E}_{z \sim p(z|x)}\left[T(z)\right]$:

$$= \operatorname*{argmin}_{q_\theta \in Q} -H(q_\theta) - \mathbb{E}_{z \sim q_\theta}\left[\tilde{\eta}(x)^\top \left(\tilde{T}(z) - \mu\right)\right] + \tilde{\eta}(x)^\top \mu, \tag{36}$$

which simplifies to

$$= \underset{q_\theta \in Q}{\operatorname{argmin}} -H(q_\theta) - \mathbb{E}_{z \sim q_\theta}\left[\tilde{\eta}(x)^\top \left(\tilde{T}(z) - \mu\right)\right]. \tag{37}$$

.

In comparison, in emergent property inference (EPI), we solve the following problem:

$$q_\theta^*(z) = \underset{q_\theta \in Q}{\operatorname{argmax}} H(q_\theta(z)), \text{ s.t. } \mathbb{E}_{z \sim q_\theta}\left[\mathbb{E}_{x \sim p(x|z)}\left[T(x)\right]\right] = \mu. \tag{38}$$

The Lagrangian objective (without augmentation) is

$$q_\theta^* = \underset{q_\theta \in Q}{\operatorname{argmin}} -H(q_\theta) + \eta_{\text{opt}}^\top \left(\mathbb{E}_{z \sim q_\theta}\left[\tilde{T}(z)\right] - \mu\right). \tag{39}$$

Thus, as the optimization proceeds, $\eta_{\text{opt}}^\top$ should converge to the natural parameter $\tilde{\eta}(x)$ through its adaptations in each epoch (see Section 5.1.2).

We have shown that there is indeed a clear relationship between Bayesian inference and EPI. Specifically, EPI is executing variational inference in an exponential family posterior, whose sufficient statistics are the emergent property statistics and mean parameterization are the emergent property values. However, in EPI we have not specified a prior distribution, or collected data, which can inform us about model parameters. Instead we have a mathematical specification of an emergent property, which the model must produce, and a maximum entropy selection principle. Accordingly, we replace the notation of $p(z \mid x)$ with $p(z \mid \mathcal{B})$ conceptualizing an inferred distribution that obeys emergent property $\mathcal{B}$ (see Section 5.1).

5.2 Theoretical models

In this study, we used emergent property inference to examine several models relevant to theoretical neuroscience. Here, we provide the details of each model and the related analyses.

5.2.1 Stomatogastric ganglion

We analyze how the parameters $z = [g_{\text{el}}, g_{\text{synA}}]$ govern the emergent phenomena of network syncing in a model of the stomatogastric ganglion (STG) [23] shown in Figure 1A with activity $x = [x_{\text{f1}}, x_{\text{f2}}, x_{\text{hub}}, x_{\text{s1}}, x_{\text{s2}}]$, using the same hyperparameter choices as Gutierrez et al. Each neuron's membrane potential $x_\alpha(t)$ for $\alpha \in \{\text{f1}, \text{f2}, \text{hub}, \text{s1}, \text{s2}\}$ is the solution of the following differential equation:

$$C_m \frac{dx_\alpha}{dt} = -\left[h_{leak}(x; z) + h_{Ca}(x; z) + h_K(x; z) + h_{hyp}(x; z) + h_{elec}(x; z) + h_{syn}(x; z)\right]. \tag{40}$$

The membrane potential of each neuron is affected by the leak, calcium, potassium, hyperpolarization, electrical and synaptic currents, respectively, which are functions of all membrane potentials and the conductance parameters z. The capacitance of the cell membrane was set to $C_m = 1nF$. Specifically, the currents are the difference in the neuron's membrane potential and that current type's reversal potential multiplied by a conductance:

$$h_{leak}(x;z) = g_{leak}(x_\alpha - V_{leak}) \tag{41}$$

$$h_{elec}(x;z) = g_{\text{el}}(x_\alpha^{post} - x_\alpha^{pre}) \tag{42}$$

$$h_{syn}(x;z) = g_{syn}S_\infty^{pre}(x_\alpha^{post} - V_{syn}) \tag{43}$$

$$h_{Ca}(x;z) = g_{Ca}M_\infty(x_\alpha - V_{Ca}) \tag{44}$$

$$h_K(x;z) = g_K N(x_\alpha - V_K) \tag{45}$$

$$h_{hyp}(x;z) = g_h H(x_\alpha - V_{hyp}). \tag{46}$$

The reversal potentials were set to $V_{leak} = -40mV$, $V_{Ca} = 100mV$, $V_K = -80mV$, $V_{hyp} = -20mV$, and $V_{syn} = -75mV$. The other conductance parameters were fixed to $g_{leak} = 1 \times 10^{-4}\mu S$. g_{Ca}, g_K, and g_{hyp} had different values based on fast, intermediate (hub) or slow neuron. The fast conductances had values $g_{Ca} = 1.9\times10^{-2}$, $g_K = 3.9\times10^{-2}$, and $g_{hyp} = 2.5\times10^{-2}$. The intermediate conductances had values $g_{Ca} = 1.7 \times 10^{-2}$, $g_K = 1.9 \times 10^{-2}$, and $g_{hyp} = 8.0 \times 10^{-3}$. Finally, the slow conductances had values $g_{Ca} = 8.5 \times 10^{-3}$, $g_K = 1.5 \times 10^{-2}$, and $g_{hyp} = 1.0 \times 10^{-2}$.

Furthermore, the Calcium, Potassium, and hyperpolarization channels have time-dependent gating dynamics dependent on steady-state gating variables M_∞, N_∞ and H_∞, respectively:

$$M_\infty = 0.5\left(1 + \tanh\left(\frac{x_\alpha - v_1}{v_2}\right)\right) \tag{47}$$

$$\frac{dN}{dt} = \lambda_N(N_\infty - N) \tag{48}$$

$$N_\infty = 0.5\left(1 + \tanh\left(\frac{x_\alpha - v_3}{v_4}\right)\right) \tag{49}$$

$$\lambda_N = \phi_N \cosh\left(\frac{x_\alpha - v_3}{2v_4}\right) \tag{50}$$

$$\frac{dH}{dt} = \frac{(H_\infty - H)}{\tau_h} \tag{51}$$

$$H_\infty = \frac{1}{1 + \exp\left(\frac{x_\alpha + v_5}{v_6}\right)} \tag{52}$$

$$\tau_h = 272 - \left(\frac{-1499}{1 + \exp\left(\frac{-x_\alpha + v_7}{v_8}\right)} \right). \tag{53}$$

where we set $v_1 = 0mV$, $v_2 = 20mV$, $v_3 = 0mV$, $v_4 = 15mV$, $v_5 = 78.3mV$, $v_6 = 10.5mV$, $v_7 = -42.2mV$, $v_8 = 87.3mV$, $v_9 = 5mV$, and $v_{th} = -25mV$.

Finally, there is a synaptic gating variable as well:

$$S_\infty = \frac{1}{1 + \exp\left(\frac{v_{th} - x_\alpha}{v_9}\right)}. \tag{54}$$

When the dynamic gating variables are considered, this is actually a 15-dimensional nonlinear dynamical system.

In order to measure the frequency of the hub neuron during EPI, the STG model was simulated for $T = 200$ time steps of $dt = 25ms$. In EPI, since gradients are taken through the simulation process, the number of time steps are kept modest if possible. The chosen dt and T were the most computationally convenient choices yielding accurate frequency measurement. Poor resolution afforded by the discrete Fourier transform motivated the use of an alternative basis of complex exponentials to measure spiking frequency. Instead, we used a basis of complex exponentials with frequencies from 0.0-1.0 Hz at 0.01Hz resolution, $\Phi = [0.0, 0.01, ..., 1.0]^\top$

Another consideration was that the frequency spectra of the neuron membrane potentials had several peaks. High-frequency sub-threshold activity obscured the maximum frequency measurement in the complex exponential basis. Accordingly, subthreshold activity was set to zero, and the whole signal was low-pass filtered with a moving average window of length 20. The signal was subsequently mean centered. After this preprocessing, the maximum frequency in the filter bank accurately reflected the firing frequency.

Finally, to differentiate through the maximum frequency identification, we used a sum-of-powers normalization. Let $\mathcal{X}_\alpha \in \mathcal{C}^{|\Phi|}$ be the complex exponential filter bank dot products with the signal $x_\alpha \in \mathbb{R}^N$, where $\alpha \in \{\text{f1}, \text{f2}, \text{hub}, \text{s1}, \text{s2}\}$. The "frequency identification" vector is

$$v_\alpha = \frac{|\mathcal{X}_\alpha|^\beta}{\sum_{k=1}^N |\mathcal{X}_\alpha(k)|^\beta}. \tag{55}$$

The frequency is then calculated as $\omega_\alpha = v_\alpha^\top \Phi$ with $\beta = 100$.

Network syncing, like all other emergent properties in this work, are defined by the emergent property statistics and values. The emergent property statistics are the first and second moments

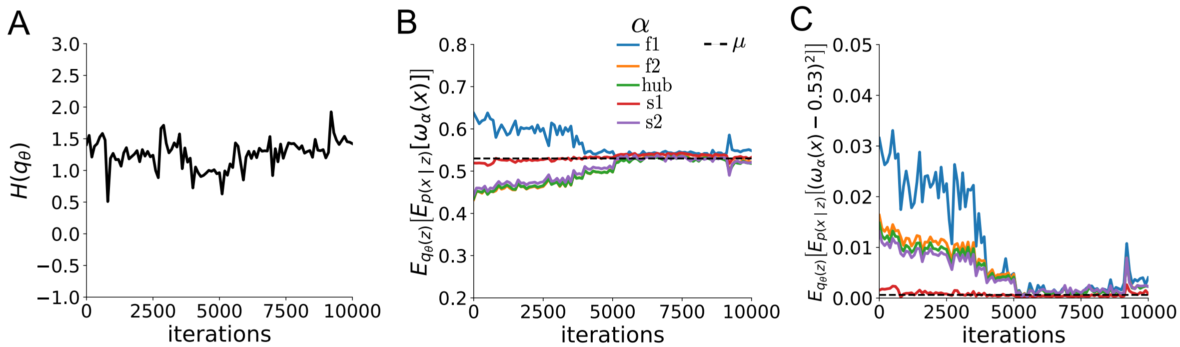

Fig. S4: EPI optimization of the STG model producing network syncing. A. Entropy throughout optimization. B. The first moment emergent property statistics converge to the emergent property values at 10,000 iterations, following the fourth augmented Lagrangian epoch of 2,500 iterations. Since $q_\theta(z)$ failed to produce enough samples yielding $\omega_{\text{f1}}(x)$ less than 0.53Hz, the convergence criteria were not satisfied after the third epoch at 7,500 iterations. C. The second moment emergent property statistics converge to the emergent property values.

of the firing frequencies. The first moments were set to 0.53Hz, and the second moments were set to 0.025Hz^2:

$$E\begin{bmatrix} \omega_{\text{f1}} \\ \omega_{\text{f2}} \\ \omega_{\text{hub}} \\ \omega_{\text{s1}} \\ \omega_{\text{s2}} \\ (\omega_{\text{f1}} - 0.53)^2 \\ (\omega_{\text{f2}} - 0.53)^2 \\ (\omega_{\text{hub}} - 0.53)^2 \\ (\omega_{\text{s1}} - 0.53)^2 \\ (\omega_{\text{s2}} - 0.53)^2 \end{bmatrix} = \begin{bmatrix} 0.53 \\ 0.53 \\ 0.53 \\ 0.53 \\ 0.53 \\ 0.025^2 \\ 0.025^2 \\ 0.025^2 \\ 0.025^2 \\ 0.025^2 \end{bmatrix} \tag{56}$$

for the EPI distribution shown in Fig. 1B. Throughout optimization, the augmented Lagrangian parameters η and c, were updated after each epoch of 2,500 iterations (see Section 5.1.2). The optimization converged after four epochs (Fig. S4).

For EPI in Fig 2C, we used a real NVP architecture with four masks and two layers of 10 units per mask, and batch normalization momentum of 0.99 mapped onto a support of $z = [g_{\text{el}}, g_{\text{synA}}] \in [4, 8] \times [0, 4]$. We used an augmented Lagrangian coefficient of $c_0 = 10^2$, a batch size $n = 300$, set $\nu = 0.1$, and initialized $q_\theta(z)$ to produce an isotropic Gaussian with mean $\mu_{\text{init}} = [6, 2]$ with standard deviation $\sigma_{\text{init}} = 0.5$.

We calculated the Hessian at the mode of the inferred EPI distribution. The Hessian of a probability model is the second order gradient of the log probability density $\log q_\theta(z)$ with respect to the parameters z: $\frac{\partial^2 \log q_\theta(z)}{\partial z \partial z^\top}$. With EPI, we can examine the Hessian, which is analytically available throughout distribution, to indicate the dimensions of parameter space that are sensitive (high magnitude eigenvalue), and which are degenerate (low magnitude eigenvalue) with respect to the emergent property produced. In Figure 1B, the eigenvectors of the Hessian v_1 and v_2 are shown evaluated at the mode of the distribution. The length of the arrows is inversely proportional to the square root of absolute value of their eigenvalues $\lambda_1 = -10.8$ and $\lambda_2 = -2.27$. We quantitatively measured the sensitivity of the model with respect to network syncing along the eigenvectors of the Hessian (Fig. 1B, inset). Sensitivity was measured as the slope coefficient of linear regression fit to network syncing error (the sum of squared differences of each neuron's frequency from 0.53Hz) as a function of parametric perturbation magnitude (maximum 0.25) away from the mode along both orientations indicated by the eigenvector with 100 equally spaced samples. The sensitivity slope coefficient of eigenvector v_1 with respect to network syncing was significant ($\beta = 4.82 \times 10^{-2}$, $p < 10^{-4}$). In contrast, eigenvector v_2 did not identify a dimension of parameter space significantly sensitive to network syncing ($\beta = 8.65 \times 10^{-4}$ with $p = .67$). These sensitivities were compared to all other dimensions of parameter space (100 equally spaced angles from 0 to π), revealing that the Hessian eigenvectors indeed identified the directions of greatest sensitivity and degeneracy (Fig. 1B, inset). The contours of Figure 1 were calculated as error in $T(x)$ from μ in both the first and second moment emergent property statistics.

5.2.2 Primary visual cortex

The dynamics of each neural populations average rate $x = [x_E, x_P, x_S, x_V]^\top$ are given by:

$$\tau \frac{dx}{dt} = -x + [Wx + h]_+^n. \tag{57}$$

By consolidating information from many experimental datasets, Billeh et al. [47] produce estimates of the synaptic strength (in mV)

$$M = \begin{bmatrix} 0.36 & 0.48 & 0.31 & 0.28 \\ 1.49 & 0.68 & 0.50 & 0.18 \\ 0.86 & 0.42 & 0.15 & 0.32 \\ 1.31 & 0.41 & 0.52 & 0.37 \end{bmatrix} \tag{58}$$

and connection probability

$$C = \begin{bmatrix} 0.16 & 0.411 & 0.424 & 0.087 \\ 0.395 & .451 & 0.857 & 0.02 \\ 0.182 & 0.03 & 0.082 & 0.625 \\ 0.105 & 0.22 & 0.77 & 0.028 \end{bmatrix}. \tag{59}$$

Multiplying these connection probabilities and synaptic efficacies gives us an effective connectivity matrix:

$$W_{\text{full}} = C \odot M = \begin{bmatrix} 0.16 & 0.411 & 0.424 & 0.087 \\ 0.395 & .451 & 0.857 & 0.02 \\ 0.182 & 0.03 & 0.082 & 0.625 \\ 0.105 & 0.22 & 0.77 & 0.028 \end{bmatrix}. \tag{60}$$

Theoretical work on these systems considers a subset of the effective connectivities [24, 44, 45]

$$W = \begin{bmatrix} W_{EE} & W_{EP} & W_{ES} & 0 \\ W_{PE} & W_{PP} & W_{PS} & 0 \\ W_{SE} & 0 & 0 & W_{SV} \\ W_{VE} & W_{VP} & W_{VS} & 0 \end{bmatrix}. \tag{61}$$

In coherence with this work, we only keep the entries of W_{full} corresponding to parameters in Equation 61.

We look at how this four-dimensional nonlinear dynamical model of V1 responds to different inputs, and compare the predictions of the linear response to the approximate posteriors obtained through EPI. The input to the system is the sum of a baseline input $b = [1, 1, 1, 1]^\top$ and a differential input dh:

$$h = b + dh. \tag{62}$$

All simulations of this system had $T = 100$ time points, a time step $dt = 5$ms, and time constant $\tau = 20$ms. The system was initialized to a random draw $x(0)_i \sim \mathcal{N}(1, 0.01)$.

We can describe the dynamics of this system more generally by

$$\dot{x}_i = -x_i + f(u_i) \tag{63}$$

where the input to each neuron is

$$u_i = \sum_j W_{ij} x_j + h_i. \tag{64}$$

Let $F_{ij} = \gamma_i \delta(i,j)$, where $\gamma_i = f'(u_i)$. Then, the linear response is

$$\frac{dx_{ss}}{dh} = F(W\frac{dx_{ss}}{dh} + I) \tag{65}$$

which is calculable by

$$\frac{dx_{ss}}{dh} = (F^{-1} - W)^{-1}. \tag{66}$$

This calculation is used to produce the magenta lines in Figure 2C, which show the linearly predicted inputs that generate a response from two standard deviations (of $\mathcal{B}$) below and above y.

The emergent property we considered was the first and second moments of the change in steady state rate dx_{ss} between the baseline input $h = b$ and $h = b + dh$. We use the following notation to indicate that the emergent property statistics were set to the following values:

$$\mathcal{B}(\alpha, y) \triangleq \mathbb{E}\begin{bmatrix} dx_{\alpha,ss} \\ (dx_{\alpha,ss} - y)^2 \end{bmatrix} = \begin{bmatrix} y \\ 0.01^2 \end{bmatrix}. \tag{67}$$

In the final analysis for this model, we sweep the input one neuron at a time away from the mode of each inferred distributions $dh^* = z^* = \text{argmax}_z \log q_\theta(z \mid \mathcal{B}(\alpha, 0.1)$. The differential responses $\delta x_{\alpha,ss}$ are examined at perturbed inputs $h = b + dh^* + \delta h_\alpha \hat{u}_\alpha$ where $\hat{u}_\alpha$ is a unit vector in the dimension of α and δx is evaluated at 101 equally spaced samples of δh_α from -15 to 15.

We measured the linear regression slope between neuron-types of δx and δh to confirm the hypotheses H1-H3 (H4 is simply observing the nonmonotonicity) and report the p values for tests of non-zero slope.

H1: the neuron-type ressponses are sensitive to their direct inputs. E-population: $\beta = 1.62$, $p < 10^{-4}$ (Fig. 3A black), P-population: $\beta = 1.06$, $p < 10^{-4}$ (Fig. 3B blue), S-population: $\beta = 6.80$, $p < 10^{-4}$ (Fig. 3C red), V-population: $\beta = 6.41$, $p < 10^{-4}$ (Fig. 3D green).

H2: the E-population ($\beta = 0$, $p = 1$) and P-populations ($\beta = 0$, $p = 1$) are not affected by δh_V (Fig. 3A green, 3B green);

H3: the S-population is not affected by δh_P ($\beta = 0$, $p = 1$) (Fig. 3C blue);

For each $\mathcal{B}(\alpha, y)$ with $\alpha \in \{E, P, S, V\}$ and $y \in \{0.1, 0.5\}$, we ran EPI using a real NVP architecture of four masks layers with two hidden layers of 10 units, mapped to a support of $z_i \in [-5, 5]$ with no batch normalization. We used an augmented Lagrangian coefficient of $c_0 = 10^5$, a batch size $n = 1000$, set $\nu = 0.5$. The EPI distributions shown in Fig. 2 are the converged distributions with maximum entropy across random seeds.

We set the parameters of the Gaussian initialization μ_{init} and Σ_{init} to the mean and covariance of random samples $z^{(i)} \sim \mathcal{U}(-5, 5)$ that produced emergent property statistic $dx_{\alpha,ss}$ within a bound ϵ of the emergent property value y. $\epsilon = 0.01$ was set to be one standard deviation of the emergent property value according to the emergent property value 0.01^2 of the variance emergent property statistic.

5.2.3 Superior colliculus

In the model of Duan et al [25], there are four total units: two in each hemisphere corresponding to the Pro/Contra and Anti/Ipsi populations. They are denoted as left Pro (LP), left Anti (LA), right Pro (RP) and right Anti (RA). Each unit has an activity (x_α) and internal variable (u_α) related by

$$x_\alpha = \left(\frac{1}{2}\tanh\left(\frac{u_\alpha - \epsilon}{\zeta}\right) + \frac{1}{2}\right) \tag{68}$$

where $\alpha \in \{LP, LA, RA, RP\}$ $\epsilon = 0.05$ and $\zeta = 0.5$ control the position and shape of the nonlinearity, respectively.

We order the elements of x and u in the following manner

$$x = \begin{bmatrix} x_{LP} \\ x_{LA} \\ x_{RP} \\ x_{RA} \end{bmatrix} \qquad u = \begin{bmatrix} u_{LP} \\ u_{LA} \\ u_{RP} \\ u_{RA} \end{bmatrix}. \tag{69}$$

The internal variables follow dynamics:

$$\tau \frac{du}{dt} = -u + Wx + h + \sigma dB \tag{70}$$

with time constant $\tau = 0.09s$ and Gaussian noise σdB controlled by the magnitude of $\sigma = 1.0$. The weight matrix has 8 parameters sW_P, sW_A, vW_{PA}, vW_{AP}, hW_P, hW_A, dW_{PA}, and dW_{AP} (Fig.

4B):

$$W = \begin{bmatrix} sW_P & vW_{PA} & hW_P & dW_{PA} \\ vW_{AP} & sW_A & dW_{AP} & hW_A \\ hW_P & dW_{PA} & sW_P & vW_{PA} \\ dW_{AP} & hW_A & vW_{AP} & sW_A \end{bmatrix}. \tag{71}$$

The system receives five inputs throughout each trial, which has a total length of 1.8s.

$$h = h_{\text{rule}} + h_{\text{choice-period}} + h_{\text{light}}. \tag{72}$$

There are rule-based inputs depending on the condition,

$$h_{\text{P,rule}}(t) = \begin{cases} I_{\text{P,rule}}[1,0,1,0]^\top, & \text{if } t \leq 1.2s \\ 0, & \text{otherwise} \end{cases} \tag{73}$$

$$h_{\text{A,rule}}(t) = \begin{cases} I_{\text{A,rule}}[0,1,0,1]^\top, & \text{if } t \leq 1.2s \\ 0, & \text{otherwise} \end{cases} \tag{74}$$

a choice-period input,

$$h_{\text{choice}}(t) = \begin{cases} I_{\text{choice}}[1,1,1,1]^\top, & \text{if } t > 1.2s \\ 0, & \text{otherwise} \end{cases} \tag{75}$$

and an input to the right or left-side depending on where the light stimulus is delivered.

$$h_{\text{light}}(t) = \begin{cases} I_{\text{light}}[1,1,0,0]^\top, & \text{if } t > 1.2s \text{ and Left} \\ I_{\text{light}}[0,0,1,1]^\top, & \text{if } t > 1.2s \text{ and Right} \cdot \\ 0, & t \leq 1.2s \end{cases} \tag{76}$$

The input parameterization was fixed to $I_{\text{P,rule}} = 10$, $I_{\text{A,rule}} = 10$, $I_{\text{choice}} = 2$, and $I_{\text{light}} = 1$.

To produce an accuracy rate of p_{LP} in the Left, Pro condition, let $\hat{p}_i$ be the empirical average steady state response (final x_{LP} at end of task) over M=500 Gaussian noise draws for a given SC model parameterization z_i:

$$\hat{p}_i = \mathbb{E}_{\sigma dB}\left[x_{LP} \mid s = L, c = P, z = z_i\right] = \frac{1}{M}\sum_{j=1}^{M} x_{LP}(s = L, c = P, z = z_i, \sigma dB_j) \tag{77}$$

where stimulus $s \in \{L, R\}$, cue $c \in \{P, A\}$, and σdB_j is the Gaussian noise on trial j. As with the V1 model, we only consider steady state responses of x, so x_α is used from here on to denote the steady state activity at the end of the trial. For the first emergent property statistic, the average over EPI samples (from $q_\theta(z)$) is set to the desired value p_{LP}:

$$\mathbb{E}_{z_i \sim q_\phi} \left[\mathbb{E}_{\sigma dB} \left[x_{LP,\text{ss}} \mid s = L, c = P, z = z_i \right] \right] = \mathbb{E}_{z_i \sim q_\phi} \left[\hat{p}_i \right] = p_{LP}. \tag{78}$$

For the next emergent property statistic, we ask that the variance of the steady state responses across Gaussian draws, is the Bernoulli variance for the empirical rate $\hat{p}_i$:

$$\mathbb{E}_{z \sim q\phi} \left[\sigma^2_{err} \right] = 0 \tag{79}$$

where the Bernoulli variance error σ^2_{err} for the Pro task, left condition is

$$\sigma^2_{err} = Var_{\sigma dB} \left[x_{LP} \mid s = L, c = P, z = z_i \right] - \hat{p}_i(1 - \hat{p}_i). \tag{80}$$

We have an additional constraint that the Pro neuron on the opposite hemisphere should have the opposite value (0 and 1). We can enforce this with another constraint:

$$\mathbb{E}_{z \sim q\phi} \left[d_P \right] = 1, \tag{81}$$

where the distance between Pro neuron steady states d_P in the Pro condition is

$$d_P = \mathbb{E}_{\sigma dB} \left[(x_{LP} - x_{RP})^2 \mid s = L, c = P, z = z_i \right] \tag{82}$$

The emergent property statistics only need to be measured during the Left stimulus condition of the Pro and Anti tasks, since the network is symmetrically parameterized. In total, the emergent property of rapid task switching at accuracy level p was defined as

$$\mathcal{B}(p) \triangleq \mathbb{E} \begin{bmatrix} \hat{p}_P \\ \hat{p}_A \\ (\hat{p}_P - p)^2 \\ (\hat{p}_A - p)^2 \\ \sigma^2_{P,err} \\ \sigma^2_{A,err} \\ d_P \\ d_A \end{bmatrix} = \begin{bmatrix} p \\ p \\ 0.15^2 \\ 0.15^2 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}. \tag{83}$$

Since the maximum variance of a random variable bounded from 0 to 1 is the Bernoulli variance $\hat{p}(1 - \hat{p})$, and the maximum squared difference between to variables bounded from 0 to 1 is 1, we do not need to control the second moment of these test statistics. These variables are dynamical system states and can only exponentially decay (or saturate) to 0 (or 1), so the Bernoulli variance error and squared difference constraints cannot be satisfied exactly in simulation. This is important to be mindful of when evaluating the convergence criteria. Instead of using our usual hypothesis testing criteria for convergence to the emergent property, we set a slack variable threshold only for these technically infeasible emergent property values to 0.05.

Using EPI to learn distributions of dynamical systems producing Bernoulli responses at a given rate (with small variance around that rate) was more challenging than expected. There is a pathology in this optimization setup, where the learned distribution of weights is bimodal attributing a fraction p of the samples to an expansive mode (which always sends x_{LP} to 1), and a fraction $1 - p$ to a decaying mode (which always sends x_{LP} to 0). This pathology was avoided using an inequality constraint prohibiting parameter samples that resulted in low variance of responses across noise.

λ	$\hat{p}$	$q_\theta(z)$	r	p-value
λ_{task}	$\hat{p}_P$	$q(z \mid \mathcal{B}(60\%))$	1.24×10^{-01}	$p < 10^{-4}$
λ_{task}	$\hat{p}_P$	$q(z \mid \mathcal{B}(70\%))$	7.56×10^{-01}	$p < 10^{-4}$
λ_{task}	$\hat{p}_P$	$q(z \mid \mathcal{B}(80\%))$	4.59×10^{-01}	$p < 10^{-4}$
λ_{task}	$\hat{p}_P$	$q(z \mid \mathcal{B}(90\%))$	3.76×10^{-01}	$p < 10^{-4}$
λ_{task}	$\hat{p}_A$	$q(z \mid \mathcal{B}(60\%))$	4.80×10^{-02}	$p < .01$
λ_{task}	$\hat{p}_A$	$q(z \mid \mathcal{B}(70\%))$	2.08×10^{-01}	$p < 10^{-4}$
λ_{task}	$\hat{p}_A$	$q(z \mid \mathcal{B}(80\%))$	4.84×10^{-01}	$p < 10^{-4}$
λ_{task}	$\hat{p}_A$	$q(z \mid \mathcal{B}(90\%))$	4.25×10^{-01}	$p < 10^{-4}$
λ_{side}	$\hat{p}_P$	$q(z \mid \mathcal{B}(50\%))$	-7.57×10^{-02}	$p < 10^{-4}$
λ_{side}	$\hat{p}_P$	$q(z \mid \mathcal{B}(60\%))$	-6.73×10^{-02}	$p < 10^{-4}$
λ_{side}	$\hat{p}_P$	$q(z \mid \mathcal{B}(70\%))$	-4.86×10^{-01}	$p < 10^{-4}$
λ_{side}	$\hat{p}_P$	$q(z \mid \mathcal{B}(80\%))$	-1.43×10^{-01}	$p < 10^{-4}$
λ_{side}	$\hat{p}_P$	$q(z \mid \mathcal{B}(90\%))$	-1.93×10^{-01}	$p < 10^{-4}$
λ_{side}	$\hat{p}_A$	$q(z \mid \mathcal{B}(60\%))$	-7.60×10^{-02}	$p < 10^{-4}$
λ_{side}	$\hat{p}_A$	$q(z \mid \mathcal{B}(70\%))$	-2.73×10^{-01}	$p < 10^{-4}$
λ_{side}	$\hat{p}_A$	$q(z \mid \mathcal{B}(80\%))$	-2.74×10^{-01}	$p < 10^{-4}$

Table 1: Table of significant correlation values from Fig. 4E.

For each accuracy level p, we ran EPI for 10 different random seeds using an architecture of 10 planar flows with a support of $z \in \mathbb{R}^8$. We used an augmented Lagrangian coefficient of $c_0 = 10^2$, a batch size $n = 300$, and set $\nu = 0.5$, and initialized $q_\theta(z)$ to produce an isotropic Gaussian of zero mean with standard deviation $\sigma_{\text{init}} = 1$. The EPI distributions shown in Fig. 4 are the converged distributions with maximum entropy across random seeds.

We report significant correlations r and their p-values from Figure 4E in Table 1. Correlations were measured from 5,000 samples of $q_\theta(z \mid \mathcal{B}(p))$ and p-values are reported for one-tailed tests, since we hypothesized a positive correlation between task accuracies p_P or p_A and λ_{task}, and a negative correlation between task accuracies p_P and p_A and λ_{side}.

5.2.4 Rank-1 RNN

Extensive research on random fully-connected recurrent neural networks has resulted in foundational theories of their activity [3, 72]. Furthermore, independent research on training these models to perform computations suggests that learning occurs through low-rank perturbations to the connectivity (e.g. [73, 74]). Recent theoretical work extends theory for random neural networks [3] to those with added low-rank structure [26]. In Section 3.5, we used this theory to enable EPI on RNN parameters conditioned on the emergent property of task execution.

Such RNNs have the following dynamics:

$$\frac{dx}{dt} = -x + W\phi(x) + h, \tag{84}$$

where x is network activity, W is the connectivity weight matrix, $\phi(\cdot) = \tanh(\cdot)$ is the input-output function, and h is the input to the system. In a rank-1 RNN (which was sufficiently complex for the Gaussian posterior conditioning task), W is the sum of a random component with strength g and a structured component determined by the outer product of vectors m and n:

$$W = g\chi + \frac{1}{N} m n^\top, \tag{85}$$

where $\chi_{ij} \sim \mathcal{N}(0, \frac{1}{N})$, and the entries of m and n are distributed as $m_i \sim \mathcal{N}(M_m, 1)$ and $n_i \sim \mathcal{N}(M_n, 1)$. For EPI, we consider $z = [g, M_m, M_n]$, which are the parameters governing the connectivity properties of the RNN.

From such a parameterization z, the theory of Mastrogiuseppe et al. produces solutions for variables describing the low dimensional response properties of the RNN. These "dynamic mean field" (DMF) variables (e.g. the activity along a vector κ_v, the total variance Δ_0, structured variance Δ_∞, and the chaotic variance Δ_T) are derived to be functions of one another and connectivity parameters z. The collection of these derived functions results in a system of equations, whose solution must be obtained through a nonlinear system of equations solver. The iterative steps of this system of equations solver are differentiable, so we take gradients through this solve process. The DMF variables provide task-relevant information about the RNN's response to task inputs.

In the Gaussian posterior conditioning example, κ_r and Δ_T are DMF variables used as task-relevant emergent property statistics μ_{post} and σ^2_{post}. Specifically, we solve for the DMF variables κ_r, κ_n, Δ_0 and Δ_∞, where the readout is nominally chosen to point in the unit orthant $r = [1, ..., 1]^\top$. The consistency equations for these variables in the presence of a constant input $h = yr - (n - M_n)$ can be derived following [26]:

$$
\begin{aligned}
\kappa_r &= G_1(\kappa_r, \kappa_n, \Delta_0, \Delta_\infty) = M_m \kappa_n + y \\
\kappa_n &= G_2(\kappa_r, \kappa_n, \Delta_0, \Delta_\infty) = M_n \langle [\phi_i] \rangle + \langle [\phi_i'] \rangle \\
\frac{\Delta_0^2 - \Delta_\infty^2}{2} &= G_3(\kappa_r, \kappa_n, \Delta_0, \Delta_\infty) = g^2 \left(\int \mathcal{D}z \Phi^2(\kappa_r + \sqrt{\Delta_0} z) - \int \mathcal{D}z \int \mathcal{D}x \Phi(\kappa_r + \sqrt{\Delta_0 - \Delta_\infty} x + \sqrt{\Delta_\infty} z) \right) \\
&\quad + (\kappa_n^2 + 1)(\Delta_0 - \Delta_\infty) \\
\Delta_\infty &= G_4(\kappa_r, \kappa_n, \Delta_0, \Delta_\infty) = g^2 \int \mathcal{D}z \left[\int \mathcal{D}x \phi(\kappa_r + \sqrt{\Delta_0 - \Delta_\infty} x + \sqrt{\Delta_\infty} z \right]^2 + \kappa_n^2 + 1
\end{aligned}
\tag{86}
$$

where here z is a gaussian integration variable. We can solve these equations by simulating the following Langevin dynamical system to a steady state:

$$
\begin{aligned}
l(t) &= \frac{\Delta_0(t)^2 - \Delta_\infty(t)^2}{2} \\
\Delta_0(t) &= \sqrt{2l(t) + \Delta_\infty(t)^2} \\
\frac{d\kappa_r(t)}{dt} &= -\kappa_r(t) + G_1(\kappa_r(t), \kappa_n(t), \Delta_0(t), \Delta_\infty(t)) \\
\frac{d\kappa_n(t)}{dt} &= -\kappa_n + G_2(\kappa_r(t), \kappa_n(t), \Delta_0(t), \Delta_\infty(t)) \\
\frac{dl(t)}{dt} &= -l(t) + G_3(\kappa_r(t), \kappa_n(t), \Delta_0(t), \Delta_\infty(t)) \\
\frac{d\Delta_\infty(t)}{dt} &= -\Delta_\infty(t) + G_4(\kappa_r(t), \kappa_n(t), \Delta_0(t), \Delta_\infty(t))
\end{aligned}
.
\tag{87}
$$

Then, the chaotic variance, which is necessary for the Gaussian posterior conditioning example, is simply calculated via $\Delta_T = \Delta_0 - \Delta_\infty$.

We ran EPI using a real NVP architecture of two masks and two layers per mask with 10 units mapped to a support of $z = [g, M_m, M_n] \in [0, 5] \times [-5, 5] \times [-5, 5]$ with no batch normalization. We used an augmented Lagrangian coefficient of $c_0 = 1$, a batch size $n = 300$, set $\nu = 0.15$, and initialized $q_\theta(z)$ to produce an isotropic Gaussian with mean $\mu_{\text{init}} = [2.5, 0, 0]$ with standard deviation $\sigma_{\text{init}} = 2.0$. The EPI distribution shown in Fig. 5 is the converged distributions with maximum entropy across five random seeds.

To examine the effect of product $M_m M_n$ on the posterior mean, μ_{post} we took perturbations in $M_m M_n$ away from two representative parameters z_1 and z_2 in 21 equally space increments from -1 to 1. For each perturbation, we sampled 10 2,000-neuron RNNs and measure the calculated posterior means. In Fig. 5D, we plot the product of $M_m M_n$ in the perturbation versus the average posterior mean across 10 network realizations with standard error bars. The correlation between perturbation product $M_m M_n$ and μ_{post} was measured over all simulations. For perturbations away from z_1 the correlation was 0.995 with $p < 10^{-4}$, and for perturbations away from z_2 the correlation was 0.983 with $p < 10^{-4}$.

In addition to the Gaussian posterior conditioning example in Section 3.5, we modeled two tasks from Mastrogiuseppe et al.: noisy detection and context-dependent discrimination. We used the same theoretical equations and task setups described in their study.

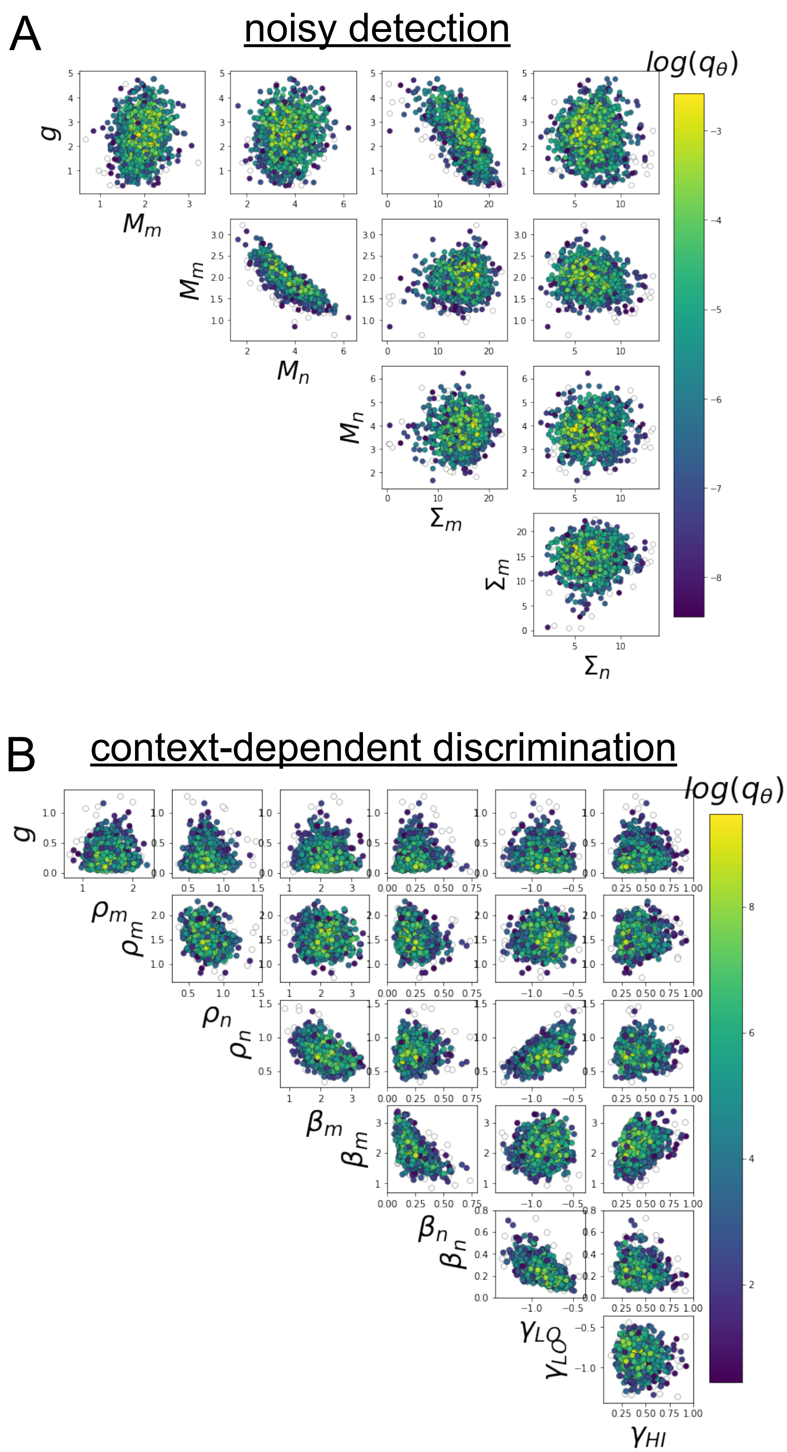

Fig. S5: A. EPI for rank-1 networks doing noisy discrimination. B. EPI for rank-2 networks doing context-dependent discrimination. See [26] for theoretical equations and task description.