

# Interrogating theoretical models of neural computation with deep inference

Sean R. Bittner, Agostina Palmigiano, Alex T. Piet, Chunyu A. Duan, Carlos D. Brody,  
Kenneth D. Miller, and John P. Cunningham.

## <sup>1</sup> 1 Abstract

<sup>2</sup> The cornerstone of theoretical neuroscience is the circuit model: a system of equations that captures  
<sup>3</sup> a hypothesized neural mechanism. Such models are valuable when they give rise to an experimen-  
<sup>4</sup> tally observed phenomenon – whether behavioral or in terms of neural activity – and thus can offer  
<sup>5</sup> insights into neural computation. The operation of these circuits, like all models, critically depends  
<sup>6</sup> on the choices of model parameters. Historically, the gold standard has been to analytically derive  
<sup>7</sup> the relationship between model parameters and computational properties. However, this enterprise  
<sup>8</sup> quickly becomes infeasible as biologically realistic constraints are included into the model increas-  
<sup>9</sup> ing its complexity, often resulting in *ad hoc* approaches to understanding the relationship between  
<sup>10</sup> model and computation. We bring recent machine learning techniques – the use of deep generative  
<sup>11</sup> models for probabilistic inference – to bear on this problem, learning distributions of parameters  
<sup>12</sup> that produce the specified properties of computation. Importantly, the techniques we introduce  
<sup>13</sup> offer a principled means to understand the implications of model parameter choices on compu-  
<sup>14</sup> tational properties of interest. We motivate this methodology with a worked example analyzing  
<sup>15</sup> sensitivity in the stomatogastric ganglion. We then use it to generate insights into neuron-type  
<sup>16</sup> input-responsivity in a model of primary visual cortex, a new understanding of rapid task switch-  
<sup>17</sup> ing in superior colliculus models, and attribution of bias in recurrent neural networks solving a toy  
<sup>18</sup> mathematical problem. More generally, this work offers a quantitative grounding for theoretical  
<sup>19</sup> models going forward, pointing a way to how rigorous statistical inference can enhance theoretical  
<sup>20</sup> neuroscience at large.

## <sup>21</sup> 2 Introduction

<sup>22</sup> The fundamental practice of theoretical neuroscience is to use a mathematical model to understand  
<sup>23</sup> neural computation, whether that computation enables perception, action, or some intermediate  
<sup>24</sup> processing [1]. In this field, a neural computation is systematized with a set of equations – the  
<sup>25</sup> model – and these equations are motivated by biophysics, neurophysiology, and other conceptual  
<sup>26</sup> considerations. The function of this system is governed by the choice of model parameters, which

when configured appropriately, 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 suitable parameter configurations? The inverse problem is crucial for reasoning about the model’s likely parameter values, uniquenesses and degeneracies, attractor states and phase transitions, and most importantly, the predictions made.

Consider the idealized practice: one carefully designs a model and analytically derives how model parameters govern the computation. Seminal examples of this gold standard include our field’s understanding of memory capacity in associative neural networks [?] and chaos and autocorrelation timescales in random neural networks [2] (which use models and analyses originating in physics), as well as the paradoxical effect in excitatory/inhibitory networks [3], we need [?], more examples [?]. Unfortunately, as circuit models include more biological realism, theory via analytic 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), producing accurate inferences about parameters of 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 (simply examining simulated activity), potentially resulting in bad inferences and thus erroneous scientific predictions and conclusions.

Of course, this same tradeoff has been confronted in many scientific fields and engineering problems 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 [4, 5], 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 [6, 7, 8, 9] and animal behavior [10, 11, 12]. These works have used deep neural networks to expand the expressivity and accuracy of statistical models of neural data [13].

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 carefully constructed biological models do 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: first, 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 second, 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 [?]. To find this distribution, 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 [14, 15, 16]. We then 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 (unlike neural data analysis). Conditioning on an emergent property requires a variant of deep probabilistic inference methods, which we have previously introduced [17]. Finally, 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 [18]. 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 [?], 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 bias in a low-rank recurrent neural network executing a toy mathematical computation. The novel scientific insights offered by EPI contextualize and clarify the previous studies exploring these models [19, 20, 21, 22] and more generally, suggests a departure from realism vs tractability considerations towards the use of modern machine learning for sophisticated interrogation of biologically relevant models.

We note that, during our preparation and early presentation of this work [23, 24], another work has arisen with broadly similar goals: bringing statistical inference to mechanistic models of neural circuits [25]. We are excited by this broad problem being recognized by the community, and we emphasize that these works offer complementary neuroscientific contributions and use different technical methodologies. Scientifically, our work has focused primarily on systems-level theoretical models, while their focus has been on lower-level cellular models. Secondly, there are several key technical differences in the approaches (see Section A.1.4) perhaps most notably is our focus on the emergent property – the measurable signal of the computation in question, vs their focus on observed datasets; both certainly are worthy pursuits. The existence of these complementary methodologies emphasizes 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 [26]. 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 [27]. A model of the STG [19] is shown schematically in Figure 1A, and note that the behavior of this model will be critically

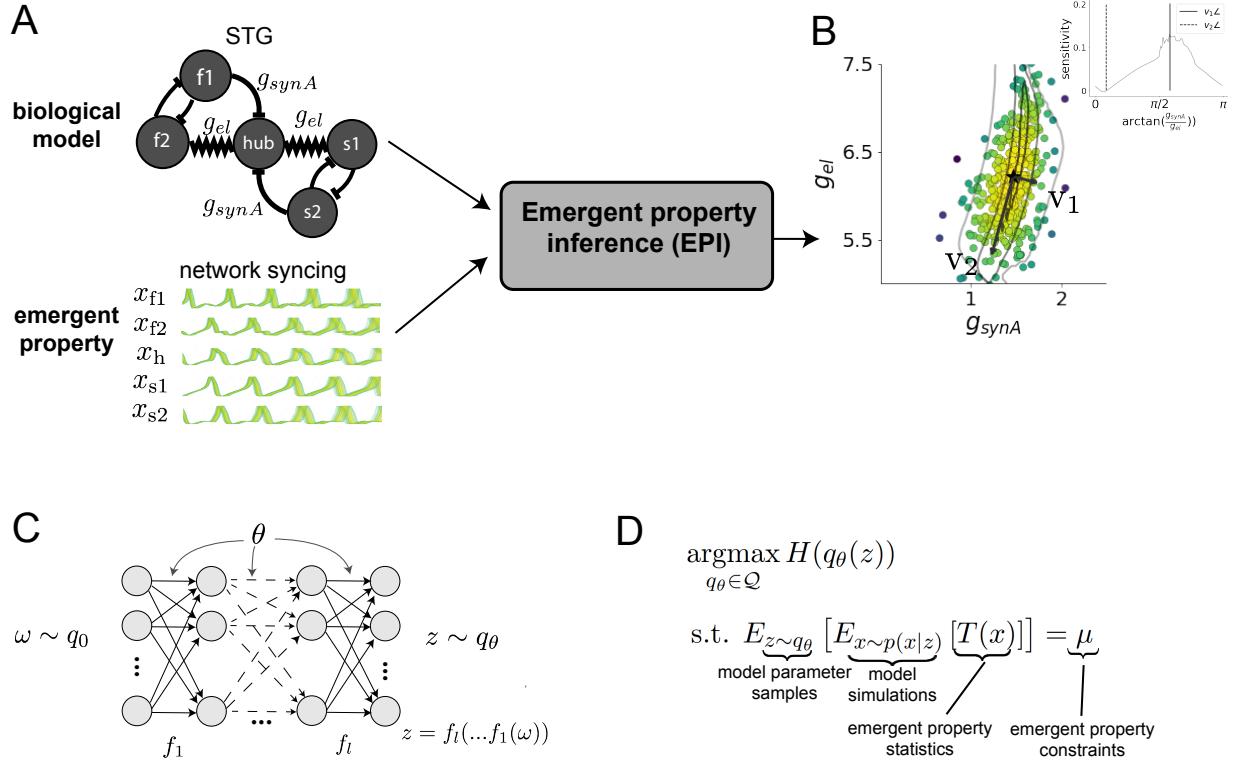


Figure 1: Emergent property inference (EPI) in the stomatogastric ganglion. A. For a choice of model (STG) and emergent property (network syncing), emergent property inference (EPI) learns a posterior distribution of the model parameters  $z = [g_{el}, g_{synA}]^\top$  conditioned on network syncing. B. An EPI distribution of STG model parameters producing network syncing. The eigenvectors of the Hessian at the mode of the inferred distribution are indicated as  $v_1$  and  $v_2$ . (Inset) Sensitivity of the system with respect to network syncing along all dimensions of parameter space away from the mode. (see Section A.2.1). C. Deep probability distributions map a latent random variable  $\omega \sim q_0$ , where  $q_0$  is chosen to be simple distribution such as an isotropic Gaussian, through a highly expressive function family  $f_\theta(\omega) = f_l(\dots f_1(\omega))$  parameterized by the neural network weights and biases  $\theta \in \Theta$ . This mapping induces an implicit probability model  $q(g_\theta(\omega)) \in \mathcal{Q}$ . D. EPI learns a distribution  $q_\theta(z)$  of model parameters that produce an emergent property: the emergent property statistics  $T(x)$  are fixed in expectation over parameter distribution samples  $z \sim q_\theta(z)$  to particular values  $\mu$ . EPI distributions maximize randomness via entropy, although other measures are sensible.

dependent on its parameterization – the choices of conductance parameters  $z = [g_{el}, g_{synA}]$ . 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  $s2$ ), and the hub neuron (hub) couples with 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 0.55Hz.

Third, qualitative parameter analysis ensues: since precise mathematical analysis is intractable in this model, a brute force sweep of parameters is done [19]. Subsequently, a qualitative description is formulated to describe of 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: the black box in Figure 1A takes, as input, the model and the specified emergent property, and produces as output 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. It conveys parameter regions critical to the emergent property, directions in parameter space that will be invariant (or not) to that property, and more. In the STG model, this distribution can be specifically queried to determine the prototypical parameter configuration for network syncing (the mode; Figure 1B star), and then how quickly network syncing will decay based on changes away from that mode. The inset of Figure 1B validates that indeed network syncing behaves as the distribution predicts, when moving away from the mode (Figure 1B star). 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 A.1.1).

## 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

[19]. In the running STG example, the dynamical state  $x = [x_{f1}, x_{f2}, x_{hub}, x_{s1}, x_{s2}]$  is the membrane potential for each neuron, which evolves according to the biophysical conductance-based equation:

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

where  $C_m=1\text{nF}$ , and  $h_{leak}$ ,  $h_{Ca}$ ,  $h_K$ ,  $h_{hyp}$ ,  $h_{elec}$ ,  $h_{syn}$  are the leak, calcium, potassium, hyperpolarization, electrical, and synaptic currents, all of which have their own complicated dependence on  $x$  and  $z = [g_{el}, g_{synA}]$  (see Section A.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  $\omega_{f1}(x)$ ,  $\omega_{f2}(x)$ , etc. – is close to an intermediate frequency of 0.55Hz. Mathematically, we achieve this via constraints on the mean and variance of  $\omega_i(x)$  for each neuron  $i \in \{f1, f2, hub, s1, s2\}$ , and thus:

$$E[T(x)] \triangleq E \begin{bmatrix} \omega_{f1}(x) \\ \vdots \\ (\omega_{f1}(x) - 0.55)^2 \\ \vdots \end{bmatrix} = \begin{bmatrix} 0.55 \\ \vdots \\ 0.025^2 \\ \vdots \end{bmatrix} \triangleq \mu, \quad (2)$$

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 14. 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 [14, 15, 16] – neural networks which transform a simple distribution into a suitably complicated distribution (as is needed here). This deep distribution is represented in Figure 1C (and see Methods for more detail). Then, mathematically, we must solve the following optimization program:

$$\begin{aligned} & \underset{q_\theta \in \mathcal{Q}}{\operatorname{argmax}} H(q_\theta(z)) \\ & \text{s.t. } E_{z \sim q_\theta} [E_{x \sim p(x|z)} [T(x)]] = \mu, \end{aligned} \quad (3)$$

where  $T(x), \mu$  are defined as in Equation 14, 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 [28, 29, 17, 30], but we emphasize that the EPI method is unaffected by this choice (but the results of course will depend on the primal objective chosen).

EPI optimizes the weights and biases  $\theta$  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. 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 A.1.2). Equipped with this method, we now prove out the value of EPI by using it to investigate three prominent models in neuroscience, using EPI to produce new insights about these models.

### 3.3 Comprehensive input-responsivity in a nonlinear sensory system

In studies of primary visual cortex (V1), theoretical models with excitatory (E) and inhibitory (I) populations have reproduced a host of experimentally documented phenomena. In particular regimes of excitation and inhibition, these E/I models exhibit the paradoxical effect [3], selective amplification [31], surround suppression [32], and sensory integrative properties [33]. Extending this model using experimental evidence of three genetically-defined classes of inhibitory neurons [34, 35], recent work [20] has investigated a four-population model – excitatory (E), parvalbumin (P), somatostatin (S), and vasointestinal peptide (V) neurons – as shown in Fig. 2A. The dynamical state of this model is the firing rate of each neuron-type population  $x = [x_E, x_P, x_S, x_V]^\top$ , which evolves according to rectified ( $\llbracket \cdot \rrbracket_+$ ) and exponentiated dynamics:

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

with effective connectivity weights  $W$  and input  $h$ . In our analysis, we set the time constant  $\tau = 20\text{ms}$  and dynamics coefficient  $n = 2$ . Also, as is fairly standard, we obtain an informative estimate of the effective connectivities between these neuron-types  $W$  in mice by multiplying their probability of connection with their average synaptic strength [?, ?] (see Section A.2.2). Given

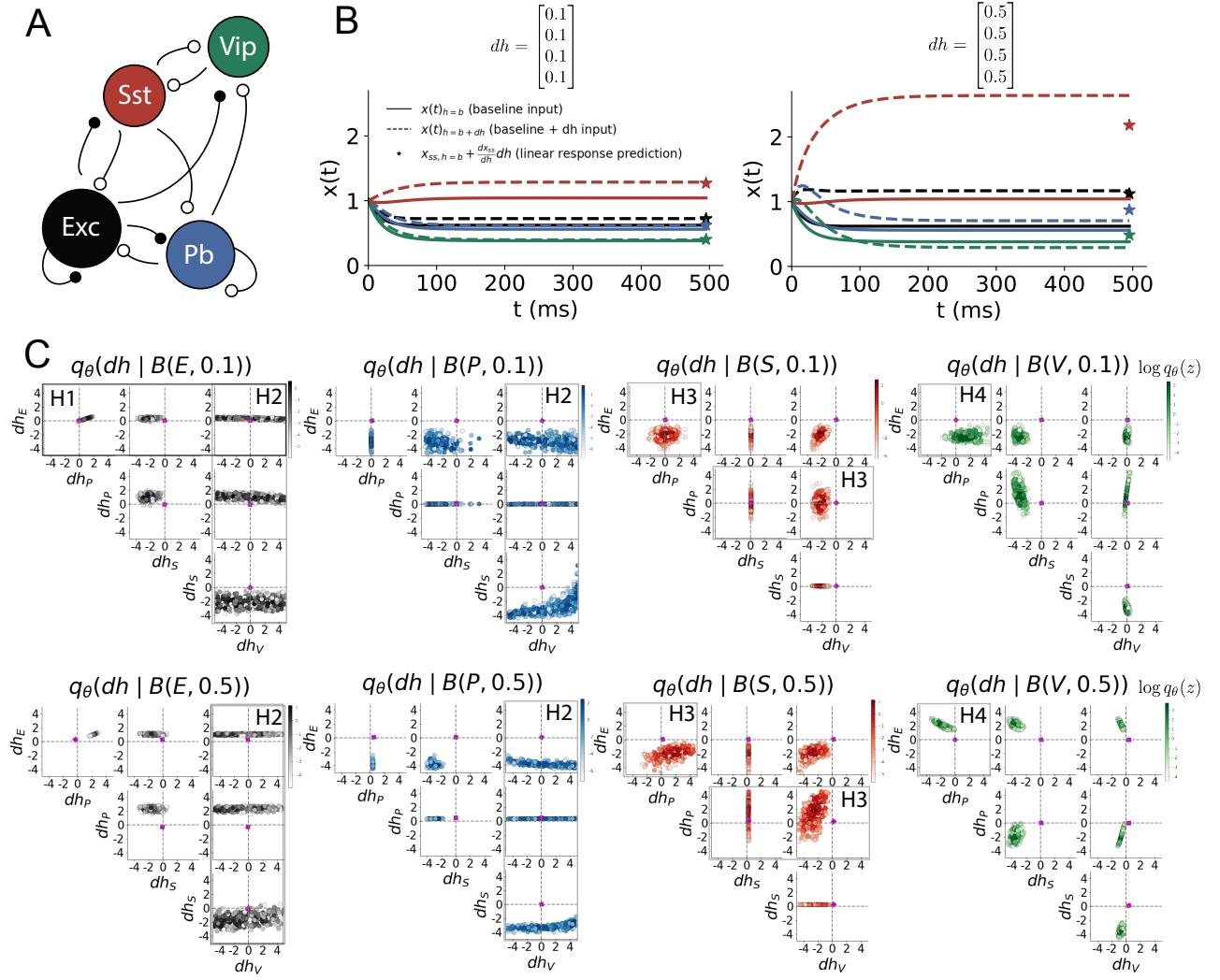


Figure 2: Hypothesis generation through EPI in a V1 model. A. Four-population model of primary visual cortex with excitatory (black), parvalbumin (blue), somatostatin (red), and vip (green) neurons. Some neuron-types largely do not form synaptic projections to others (excitatory and inhibitory projections filled and unfilled, respectively). B. Linear response predictions become inaccurate with greater input strength. V1 model simulations for input ( $b$ ) solid and ( $b + dh$ ) dashed.  $b = [1, 1, 1, 1]^T$  and (left)  $dh = [0.1, 0.1, 0.1, 0.1]^T$  (right)  $dh = [0.5, 0.5, 0.5, 0.5]^T$ . 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).

201 these fixed choices of  $W$ ,  $n$ , and  $\tau$ , we studied the system's response to input

$$h = b + dh, \quad (5)$$

202 where the input  $h$  is comprised of a baseline input  $b = [b_E, b_P, b_S, b_V]^\top$  and a differential input  
 203  $dh = [dh_E, dh_P, dh_S, dh_V]^\top$  to each neuron-type population. Throughout subsequent analyses, the  
 204 baseline input is  $b = [1, 1, 1, 1]^\top$ .

205 Having established our model, we now define the emergent property. We begin with the linearized  
 206 response of the system to input  $\frac{dx_{ss}}{dh}$  at the steady state  $x_{ss}$ , i.e. a fixed point. While this lin-  
 207 earization accurately predicts differential responses  $dx_{ss} = [dx_{E,ss}, dx_{P,ss}, dx_{S,ss}, dx_{V,ss}]$  for small  
 208 differential inputs to each population  $dh = [0.1, 0.1, 0.1, 0.1]$  (Fig. 2B, left), linearization is a poor  
 209 predictor in this nonlinear model more generally (Fig. 3B, right). Currently available approaches  
 210 to deriving the steady state response of this system are limited.

211 To get a more comprehensive picture of the input-responsivity of each neuron-type, we used EPI  
 212 to learn a distribution of the differential inputs to each population  $dh$  that produce an increase  
 213 of  $y \in \{0.1, 0.5\}$  in the rate of each neuron-type population  $\alpha \in \{E, P, S, V\}$ . We want to know  
 214 the differential inputs  $dh$  that result in a differential steady state  $dx_{\alpha,ss}$  (the change in  $x_{\alpha,ss}$  when  
 215 receiving input  $h = b + dh$  with respect to the baseline  $h = b$ ) of value  $y$  with some small, arbitrarily  
 216 chosen amount of variance  $0.01^2$ . These statements amount to the emergent property

$$\mathcal{B}(\alpha, y) \triangleq E \begin{bmatrix} dx_{\alpha,ss} \\ (dx_{\alpha,ss} - y)^2 \end{bmatrix} = \begin{bmatrix} y \\ 0.01^2 \end{bmatrix} \quad (6)$$

217 We continue to use  $\mathcal{B}(\cdot)$  throughout the rest of the study as short hand for emergent property, which  
 218 represents a different signature of computation in each application. In Each column of Figure 2C  
 219 visualizes the inferred distribution of  $dh$  corresponding to a excitatory (red), parvalbumin (blue),  
 220 somatostatin (red) and vip (green) neuron-type increase, while each row corresponds to amounts of  
 221 increase 0.1 and 0.5. These distributions conditioned on such emergent properties are now available  
 222 through EPI. For each pair of parameters we show the two-dimensional marginal distribution of  
 223 samples colored by  $\log q_\theta(dh | \mathcal{B}(\alpha, y))$ . The inferred distributions immediately suggest four hy-  
 224 potheses:

225

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

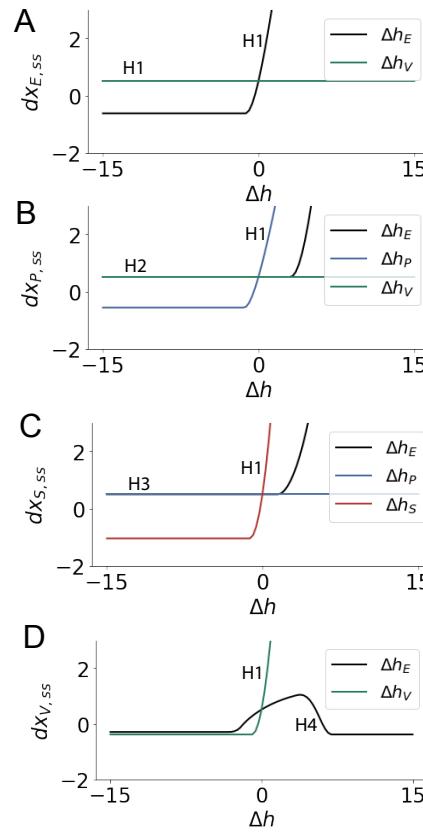


Figure 3: Confirming EPI generated hypotheses in V1. A. Differential responses by the E-population to changes in individual input  $\Delta h_\alpha u_\alpha$  away from the mode of the EPI distribution  $dh^*$ . B-D Same plots for the P-, S-, and V-populations. Labels H1, H2, H3, and H4 indicate which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

229 H2: the E- and P-populations should be largely unaffected by  $dh_V$  (Fig. 2C H2 indicates high  
 230 variance in  $dh_V$  when  $\alpha \in \{E, P\}$ );  
 231 H3: the S-population should be largely unaffected by  $dh_P$  (Fig. 2C H3 indicate high variance in  
 232  $dh_P$  when  $\alpha = S$ );  
 233 H4: there should be a nonmonotonic response of  $dx_{V,ss}$  with  $dh_E$  (Fig. 2C H4 indicates that  
 234 negative  $dh_E$  should result in small  $dx_{V,ss}$ , but positive  $dh_E$  should elicit a larger  $dx_{V,ss}$ );  
 235 We evaluate these hypotheses by taking steps in individual neuron-type input  $\Delta h_\alpha$  away from the  
 236 modes of the inferred distributions at  $y = 0.1$ .

$$dh^* = z^* = \underset{z}{\operatorname{argmax}} \log q_\theta(z | \mathcal{B}(\alpha, 0.1)) \quad (7)$$

237 Now,  $dx_{\alpha,ss}$  is the steady state response to the system with input  $h = b + dh^* + \Delta h_\alpha u_\alpha$  where  $u_\alpha$   
 238 is a unit vector in the dimension of  $\alpha$ . The EPI-generated hypotheses are confirmed.

- 239 • the neuron-type responses are sensitive to their direct inputs (Fig. 3A black, 3B blue, 3C  
 240 red, 3D green);  
 241 • the E- and P-populations are not affected by  $dh_V$  (Fig. 3A green, 3B green);

- 242     • the S-population is not affected by  $dh_P$  (Fig. 3C blue);  
 243     • the V-population exhibits a nonmonotonic response to  $dh_E$  (Fig. 3D black), and is in fact  
 244       the on population to do so (Fig. 3A-C black).

245 These hypotheses were in stark contrast to what was available to us via traditional analytical linear  
 246 prediction (Fig. 2C, magenta). To this point, we have shown the utility of EPI on relatively low-  
 247 level emergent properties like network syncing and differential neuron-type population responses.  
 248 In the remainder of the study, we focus on using EPI to understand models of more abstract  
 249 cognitive function.

#### 250 3.4 Identifying neural mechanisms of behavioral learning.

251 Identifying measurable biological changes that result in improved behavior is important for neuro-  
 252 science, since they may indicate how the learning brain adapts. In a rapid task switching experiment  
 253 [36], rats were explicitly cued on each trial to either orient towards a visual stimulus in the Pro  
 254 (P) task or orient away from a visual stimulus in the Anti (A) task (Fig. 3a). Neural recordings  
 255 in the midbrain superior colliculus (SC) exhibited two populations of neurons that simultaneously  
 256 represented both task context (Pro or Anti) and motor response (contralateral or ipsilateral to the  
 257 recorded side): the Pro/Contra and Anti/Ipsi neurons [21]. Duan et al. proposed a model of SC  
 258 that, like the V1 model analyzed in the previous section, is a four-population dynamical system.  
 259 Here, the neuron-type populations are functionally-defined as the Pro- and Anti-populations in each  
 260 hemisphere (left (L) and right (R)). The Pro- or Anti-populations receive an input determined by  
 261 the cue, and then the left and right populations receive an input based on the side of the light  
 262 stimulus. Activities were bounded between 0 and 1, so that a high output of the Pro population  
 263 in a given hemisphere corresponds to the contralateral response. An additional stipulation is that  
 264 when one Pro population responds with a high-output, the opposite Pro population must respond  
 265 with a low output. Finally, this circuit operates in the presence of Gaussian noise resulting in trial-  
 266 to-trial variability (see Section A.2.3). The connectivity matrix is parameterized by the geometry  
 267 of the population arrangement (Fig. 3B).

268 Here, we used EPI to learn distributions of the SC weight matrix parameters  $z = W$  conditioned  
 269 on various levels of rapid task switching accuracy  $\mathcal{B}(p)$  for  $p \in \{50\%, 60\%, 70\%, 80\%, 90\%\}$  (see  
 270 Section A.2.3). Following the approach in Duan et al., we decomposed the connectivity matrix  
 271  $W = QAQ^{-1}$  in such a way (the Schur decomposition) that the basis vectors  $q_i$  are the same for all

<sup>272</sup>  $W$  (Fig. 3C). These basis vectors have intuitive roles in processing for this task, and are accordingly  
<sup>273</sup> named the *all* mode - all neurons co-fluctuate, *side* mode - one side dominates the other, *task* mode  
<sup>274</sup> - the Pro or Anti populations dominate the other, and *diag* mode - Pro- and Anti-populations of  
<sup>275</sup> opposite hemispheres dominate the opposite pair. The corresponding eigenvalues (e.g.  $a_{\text{task}}$ , which  
<sup>276</sup> change according to  $W$ ) indicate the degree to which activity along that mode is increased or  
<sup>277</sup> decreased by  $W$ .

<sup>278</sup> EPI demonstrates that, for greater task accuracies, the task mode eigenvalue increases, indicating  
<sup>279</sup> the importance of  $W$  to the task representation (Fig. 4D, purple). Stepping from random chance  
<sup>280</sup> (50%) networks to marginally task-performing (60%) networks, there is a marked decrease of the  
<sup>281</sup> side mode eigenvalues (Fig. 3D, orange). Such side mode suppression remains in the models  
<sup>282</sup> achieving greater accuracy, revealing its importance towards task performance. There were no  
<sup>283</sup> interesting trends with learning in the all or diag mode (hence not shown in Fig. 3). Importantly,  
<sup>284</sup> we can conclude from our methodology that side mode suppression in  $W$  allows rapid task switching,  
<sup>285</sup> and that greater task-mode representations in  $W$  increase accuracy. These hypotheses are confirmed  
<sup>286</sup> by forward simulation of the SC model (Fig. 3E). Thus, EPI produces novel, experimentally testable  
<sup>287</sup> predictions: effective connectivity between these populations changes throughout learning, in a way  
<sup>288</sup> that increases its task mode and decreases its side mode eigenvalues.

### <sup>289</sup> 3.5 Linking RNN connectivity to computational error

<sup>290</sup> So far, each model we have studied was designed from fundamental biophysical principles, genetically-  
<sup>291</sup> or functionally-defined neuron types. At a more abstract level of modeling, recurrent neural net-  
<sup>292</sup> works (RNNs) are high-dimensional dynamical models of computation that are becoming increas-  
<sup>293</sup> ingly popular in neuroscience research [37]. In theoretical neuroscience, RNN dynamics usually  
<sup>294</sup> follow the equation

$$\frac{dx}{dt} = -x(t) + W\phi(x(t)) + I(t), \quad (8)$$

<sup>295</sup> where  $x(t)$  is the network activity,  $W$  is the network connectivity,  $\phi(\cdot) = \tanh(\cdot)$ , and  $I(t)$  is the  
<sup>296</sup> input to the system. Such RNNs are trained to do a task from a systems neuroscience experiment,  
<sup>297</sup> and then the unit activations of the trained RNN are compared to recorded neural activity. Fully-  
<sup>298</sup> connected RNNs with tens of thousands of parameters are challenging to characterize [38], especially  
<sup>299</sup> making statistical inferences about their parameterization. Alternatively, we consider a rank-1,  $N$ -  
<sup>300</sup> neuron RNN with connectivity

$$W = g\chi + \frac{1}{N}mn^\top, \quad (9)$$

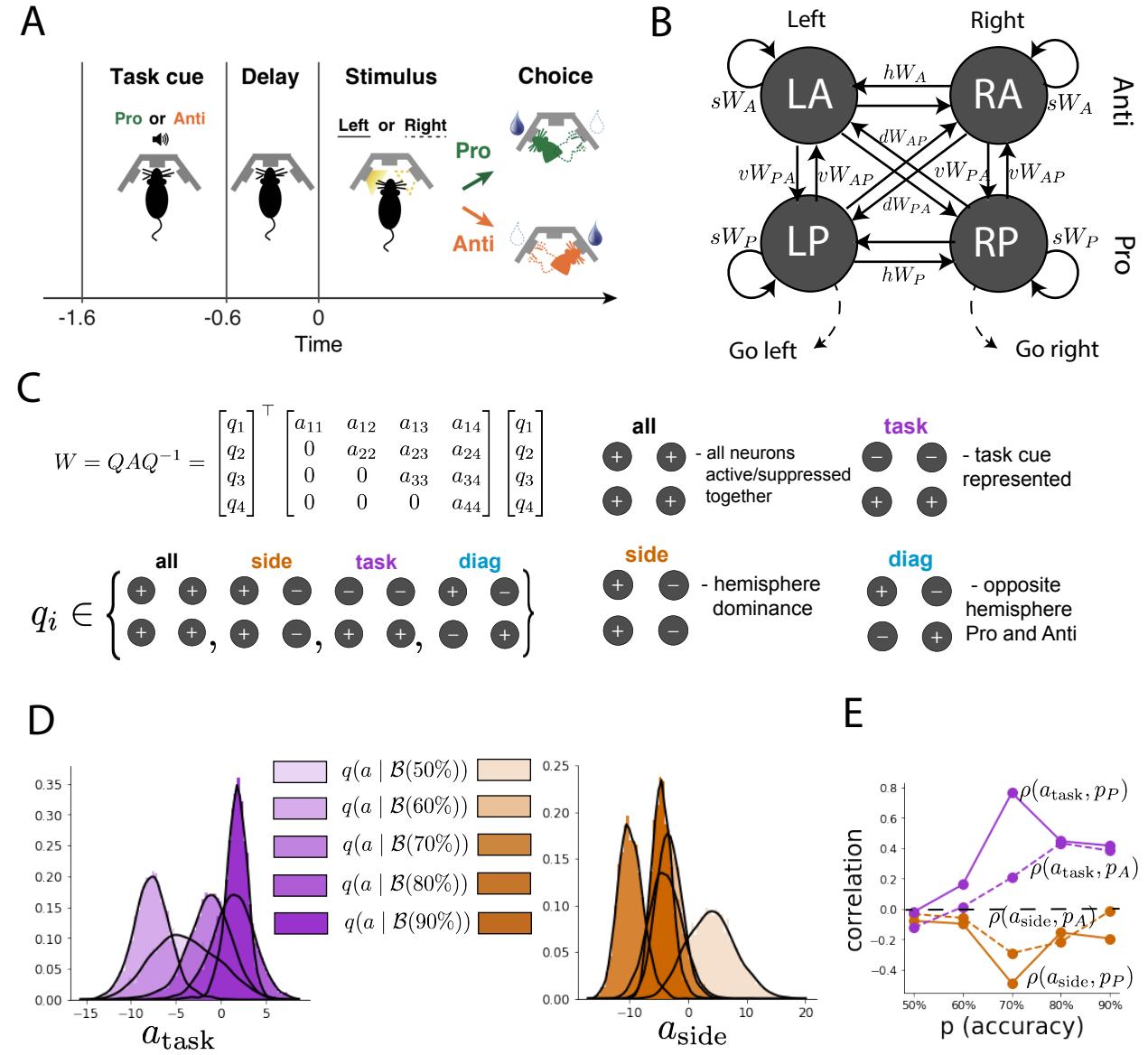


Figure 4: EPI reveals changes in SC [21] connectivity that control task accuracy. A. Rapid task switching behavioral paradigm (see text). B. Model of superior colliculus (SC). Neurons: LP - left pro, RP - right pro, LA - left anti, RA - right anti. Parameters:  $sW$  - self,  $hW$  - horizontal,  $vW$  - vertical,  $dW$  - diagonal weights. C. The Schur decomposition of the weight matrix  $W = QAQ^{-1}$  is a unique decomposition with orthogonal  $Q$  and upper triangular  $A$ . Schur modes:  $q_{\text{all}}$ ,  $q_{\text{task}}$ ,  $q_{\text{side}}$ , and  $q_{\text{diag}}$ . 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.

301 where  $\chi_{ij} \sim \mathcal{N}(0, \frac{1}{N})$ ,  $g$  is the random strength, and the entries of  $m$  and  $n$  are drawn from Gaussian  
 302 distributions  $m_i \sim \mathcal{N}(M_m, 1)$  and  $n_i \sim \mathcal{N}(M_n, 1)$ . We use EPI to infer the parameterizations of  
 303 rank-1 RNNs solving an example task, enabling discovery of properties of connectivity that result  
 304 in different types of computational errors.

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

$$\mu_{post} = \frac{\frac{\mu_0}{\sigma_0^2} + \frac{y}{\sigma_y^2}}{\frac{1}{\sigma_0^2} + \frac{1}{\sigma_y^2}} \quad \sigma_{post}^2 = \frac{1}{\frac{1}{\sigma_0^2} + \frac{1}{\sigma_y^2}}. \quad (10)$$

309 The RNN is trained to solve this task by producing readout activity that is on average the posterior  
 310 mean  $\mu_{post}$ , and activity whose variability is the posterior variance  $\sigma_{post}^2$  (a setup inspired by  
 311 [?]). To solve this Gaussian posterior conditioning task, the RNN response to a constant input  
 312  $I(t) = yw + (n - M_n)$  must equal the posterior mean along readout vector  $w$ , where

$$\kappa_w = \frac{1}{N} \sum_{j=1}^N w_j \phi(x_j) \quad (11)$$

313 Additionally, the amount of chaotic variance  $\Delta_T$  must equal the posterior variance.  $\kappa_w$  and  $\Delta_T$  can  
 314 be expressed in terms of each other through a solvable system of nonlinear equations (see Section  
 315 A.2.4) [22]. This theory allows us to mathematically formalize the execution of this task into an  
 316 emergent property, where the emergent property statistics of the RNN activity are  $k_w$  and  $\Delta_T$  and  
 317 the emergent property values are the ground truth posterior mean  $\mu_{post}$  and variance  $\sigma_{post}^2$ :

$$E \begin{bmatrix} \kappa_w \\ \Delta_T \\ (\kappa_w - \mu_{post})^2 \\ (\Delta_T^2 - \sigma_{post}^2)^2 \end{bmatrix} = \begin{bmatrix} \mu_{post} \\ \sigma_{post}^2 \\ 0.1 \\ 0.1 \end{bmatrix} \quad (12)$$

318 We specify a substantial amount of variability in the variance constraints so that the inferred  
 319 distribution results in RNNs with a variety biases in their solutions to the gaussian posterior  
 320 conditioning problem.

321 We used EPI to learn distributions of RNN connectivity properties  $z = [g \ M_m \ M_n]$  executing  
 322 Gaussian posterior conditioning given an input of  $y = 2$ . (see Section A.2.4) (Fig. 5B). The true  
 323 Gaussian conditioning posterior for an input of  $y = 2$  is  $\mu_{post} = 3$  and  $\sigma_{post} = 0.5$ . We examined  
 324 the nature of the over- and under-estimation of the posterior means (Fig. 5B, left) and variances

(Fig. 5B, right) in the inferred distributions. There is rough symmetry in the  $M_m$ - $M_n$  plane, suggesting a degeneracy in the product of  $M_m$  and  $M_n$  (Fig. 5B). The product of  $M_m$  and  $M_n$  almost completely determines the posterior mean (Fig. 5B, left), and the random strength  $g$  is the most influential variable on the temporal variance (Fig. 5B, right). Neither of these observations were obvious from what mathematical analysis is available in networks of this type (see Section A.2.4). They lead to the following hypotheses:

H1: The posterior mean of the RNN increases with the product of  $M_m$  and  $M_n$ ;

H2: The posterior variance increases with  $g$ ;

Testing these now in finite-size networks. Will write end of this later.

This novel procedure of doing inference in interpretable parameterizations of RNNs conditioned on the emergent property of task execution is straightforwardly generalizable to other tasks like noisy integration and context-dependent decision making (Fig. S1).

## 4 Discussion

### 4.1 EPI is a general tool for theoretical neuroscience

Models of biological systems are often comprised of complex nonlinear differential equations, making traditional theoretical analysis and statistical inference intractable. In contrast, EPI is capable of learning distributions of parameters in such models producing measurable signatures of computation. We have demonstrated its utility 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 firing rates of membrane potentials, as well as high-level cognitive function like Gaussian 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 utility of EPI.

In this study, we have focused on applying EPI to low dimensional parameter spaces of models with low dimensional dynamical state. 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

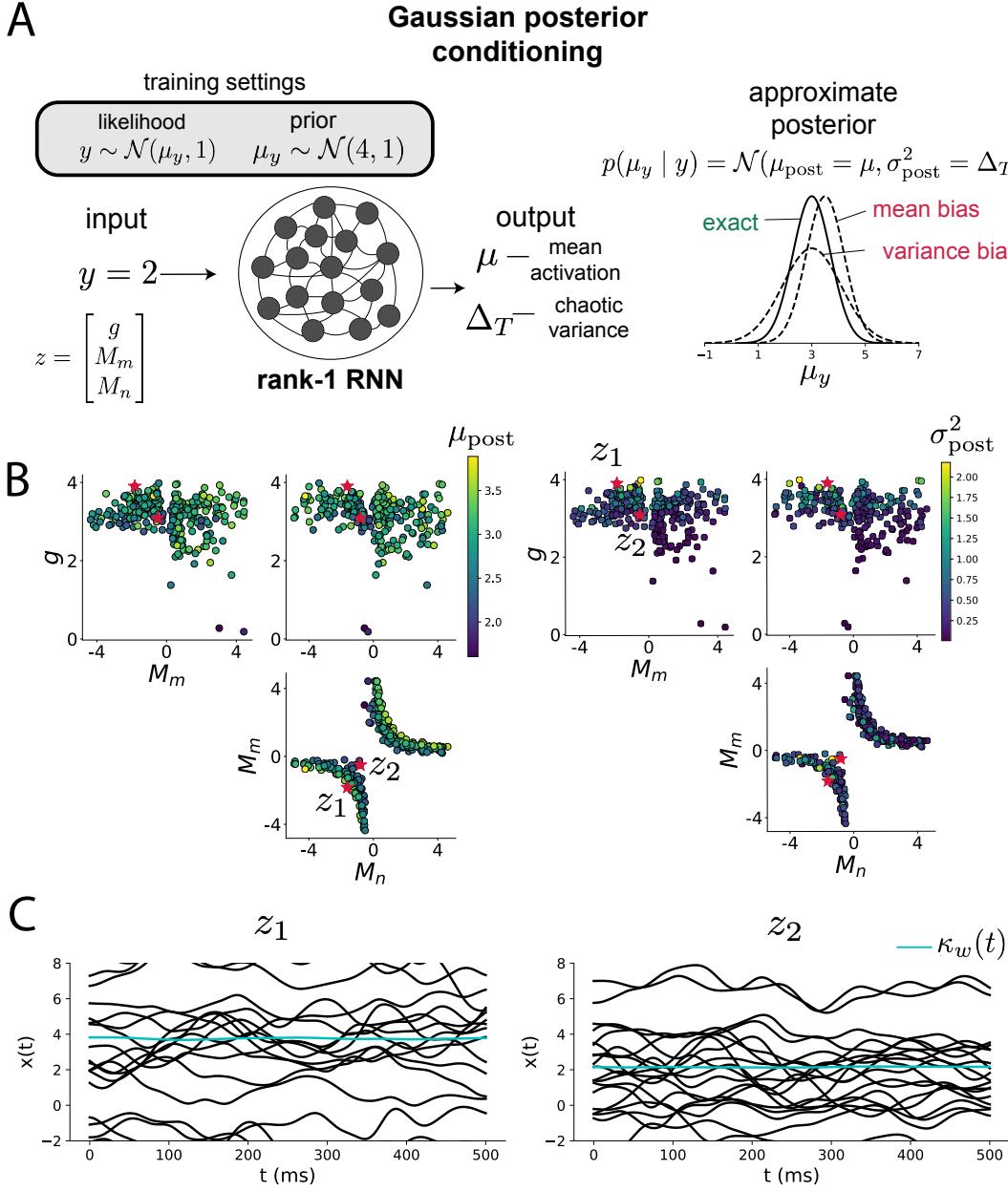


Figure 5: Sources of solution bias in an RNN computation. A. (left) A rank-1 RNN executing a Gaussian posterior conditioning computation on  $\mu_y$ . (right) Bias in this computation can come from over- or under-estimating the posterior mean or variance. B. EPI distribution of rank-1 RNNs executing Gaussian posterior conditioning. Samples are colored by (left) posterior mean  $\mu_{\text{post}} = \kappa_w$  and (right) posterior variance  $\sigma_{\text{post}}^2 = \Delta_T$ . C. Finite-size networks sampled from the distribution perform the calculation and have the computational biases expected from their parameter values. Activity along readout  $\kappa_w$  (cyan).

354 produced state-of-the-art performance on high-dimensional tasks such as texture generation [17].  
 355 Of course, increasing the dimensionality of the dynamical state of the model makes optimization  
 356 more expensive, and there is a practical limit there as with any machine learning approach. For  
 357 systems with high dimensional state, we recommend using theoretical approaches (e.g. [22]) to  
 358 reason about reduced parameterizations of such high-dimensional systems.

359 There are additional technical considerations when assessing the suitability of EPI for a particu-  
 360 lar modeling question. First and foremost, as in any optimization problem, the defined emergent  
 361 property should always be appropriately conditioned (constraints should not have wildly different  
 362 units). Furthermore, if the program is underconstrained (not enough constraints), the distribution  
 363 grows (in entropy) unstably unless mapped to a finite support. If overconstrained, there is no pa-  
 364 rameter set producing the emergent property, and EPI optimization will fail (appropriately). Next,  
 365 one should consider the computational cost of the gradient calculations. In the best circumstance,  
 366 there is a simple, closed form expression (e.g. Section A.1.1) for the emergent property statistic  
 367 given the model parameters. On the other end of the spectrum, many forward simulation iterations  
 368 may be required before a high quality measurement of the emergent property statistic is available  
 369 (e.g. Section A.2.1). In such cases, optimization will be expensive.

370 **4.2 Novel hypotheses from EPI**

371 Machine learning has played an effective, multifaceted role in neuroscientific progress. Primarily,  
 372 it has revealed structure in large-scale neural datasets [39, 40, 41, 42, 43, 44] (see review, [13]).  
 373 Secondarily, trained algorithms of varying degrees of biological relevance are beginning to be viewed  
 374 as fully-observable computational systems comparable to the brain [38, 45].

375 For example, consider the fact that we do not fully understand the four-dimensional models of V1  
 376 [20]. Because analytical approaches to studying nonlinear dynamical systems become increasingly  
 377 complicated when stepping from two-dimensional to three- or four-dimensional systems in the  
 378 absence of restrictive simplifying assumptions [46], it is unsurprising that this model has been a  
 379 challenge. In Section 3.3, we showed that EPI was far more informative about neuron-type input  
 380 responsibility than the predictions afforded through analysis. By flexibly conditioning this V1 model  
 381 on different emergent properties, we performed an exploratory analysis of a *model* rather than a  
 382 dataset, which generated and proved out a set of testable predictions.

383 Of course, exploratory analyses can also be directed. For example, when interested in model

384 changes during learning, one can use EPI to condition as we did in Section 3.4. This analysis  
385 identified experimentally testable predictions (proved out *in-silico*) of changes in connectivity in  
386 SC throughout learning. Precisely, we predict that an initial reduction in side mode eigenvalue,  
387 and a steady increase in task mode eigenvalue will take place, during learning, in the effective  
388 connectivity matrices of learning rats.

389 In our final analysis, we present a novel procedure for doing statistical inference on interpretable  
390 parameterizations of RNNs executing simple tasks . This methodology relies on recently extended  
391 theory of responses in random neural networks with minimal structure [22]. With this methodology,  
392 we can finally open the probabilistic model selection toolkit reasoning about the connectivity of  
393 RNNs solving tasks.

## 394 References

- 395 [1] Larry F Abbott. Theoretical neuroscience rising. *Neuron*, 60(3):489–495, 2008.
- 396 [2] Haim Sompolinsky, Andrea Crisanti, and Hans-Jurgen Sommers. Chaos in random neural  
397 networks. *Physical review letters*, 61(3):259, 1988.
- 398 [3] Misha V Tsodyks, William E Skaggs, Terrence J Sejnowski, and Bruce L McNaughton. Para-  
399 doxical effects of external modulation of inhibitory interneurons. *Journal of neuroscience*,  
400 17(11):4382–4388, 1997.
- 401 [4] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *International Confer-  
402 ence on Learning Representations*, 2014.
- 403 [5] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation  
404 and variational inference in deep latent gaussian models. *International Conference on Machine  
405 Learning*, 2014.
- 406 [6] Yuanjun Gao, Evan W Archer, Liam Paninski, and John P Cunningham. Linear dynamical  
407 neural population models through nonlinear embeddings. In *Advances in neural information  
408 processing systems*, pages 163–171, 2016.
- 409 [7] Yuan Zhao and Il Memming Park. Recursive variational bayesian dual estimation for nonlinear  
410 dynamics and non-gaussian observations. *stat*, 1050:27, 2017.

- 411 [8] Gabriel Barello, Adam Charles, and Jonathan Pillow. Sparse-coding variational auto-encoders.  
412 *bioRxiv*, page 399246, 2018.
- 413 [9] Chethan Pandarinath, Daniel J O’Shea, Jasmine Collins, Rafal Jozefowicz, Sergey D Stavisky,  
414 Jonathan C Kao, Eric M Trautmann, Matthew T Kaufman, Stephen I Ryu, Leigh R Hochberg,  
415 et al. Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature  
416 methods*, page 1, 2018.
- 417 [10] Alexander B Wiltschko, Matthew J Johnson, Giuliano Iurilli, Ralph E Peterson, Jesse M  
418 Katon, Stan L Pashkovski, Victoria E Abraira, Ryan P Adams, and Sandeep Robert Datta.  
419 Mapping sub-second structure in mouse behavior. *Neuron*, 88(6):1121–1135, 2015.
- 420 [11] Matthew J Johnson, David K Duvenaud, Alex Wiltschko, Ryan P Adams, and Sandeep R  
421 Datta. Composing graphical models with neural networks for structured representations and  
422 fast inference. In *Advances in neural information processing systems*, pages 2946–2954, 2016.
- 423 [12] Eleanor Batty, Matthew Whiteway, Shreya Saxena, Dan Biderman, Taiga Abe, Simon Musall,  
424 Winthrop Gillis, Jeffrey Markowitz, Anne Churchland, John Cunningham, et al. Behavenet:  
425 nonlinear embedding and bayesian neural decoding of behavioral videos. *Advances in Neural  
426 Information Processing Systems*, 2019.
- 427 [13] Liam Paninski and John P Cunningham. Neural data science: accelerating the experiment-  
428 analysis-theory cycle in large-scale neuroscience. *Current opinion in neurobiology*, 50:232–241,  
429 2018.
- 430 [14] Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows.  
431 *International Conference on Machine Learning*, 2015.
- 432 [15] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp.  
433 *arXiv preprint arXiv:1605.08803*, 2016.
- 434 [16] George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density  
435 estimation. In *Advances in Neural Information Processing Systems*, pages 2338–2347, 2017.
- 436 [17] Gabriel Loaiza-Ganem, Yuanjun Gao, and John P Cunningham. Maximum entropy flow  
437 networks. *International Conference on Learning Representations*, 2017.

- [438] [18] 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.
- [441] [19] 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.
- [444] [20] 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.
- [447] [21] 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.
- [450] [22] Francesca Mastrogiovanni and Srdjan Ostojic. Linking connectivity, dynamics, and computations in low-rank recurrent neural networks. *Neuron*, 99(3):609–623, 2018.
- [452] [23] 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.
- [455] [24] 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.
- [458] [25] Jan-Matthis Lueckmann, Pedro Goncalves, Chaitanya Chintaluri, William F Podlaski, Giacomo Bassetto, Tim P Vogels, and Jakob H Macke. Amortised inference for mechanistic models of neural dynamics. In *Computational and Systems Neuroscience Meeting (COSYNE), Lisbon, Portugal*, 2019.
- [462] [26] Eve Marder and Vatsala Thirumalai. Cellular, synaptic and network effects of neuromodulation. *Neural Networks*, 15(4-6):479–493, 2002.
- [464] [27] Astrid A Prinz, Dirk Bucher, and Eve Marder. Similar network activity from disparate circuit parameters. *Nature neuroscience*, 7(12):1345, 2004.

- 466 [28] Edwin T Jaynes. Information theory and statistical mechanics. *Physical review*, 106(4):620,  
467 1957.
- 468 [29] Gamaleldin F Elsayed and John P Cunningham. Structure in neural population recordings:  
469 an expected byproduct of simpler phenomena? *Nature neuroscience*, 20(9):1310, 2017.
- 470 [30] Cristina Savin and Gašper Tkačik. Maximum entropy models as a tool for building precise  
471 neural controls. *Current opinion in neurobiology*, 46:120–126, 2017.
- 472 [31] Brendan K Murphy and Kenneth D Miller. Balanced amplification: a new mechanism of  
473 selective amplification of neural activity patterns. *Neuron*, 61(4):635–648, 2009.
- 474 [32] Hirofumi Ozeki, Ian M Finn, Evan S Schaffer, Kenneth D Miller, and David Ferster. Inhibitory  
475 stabilization of the cortical network underlies visual surround suppression. *Neuron*, 62(4):578–  
476 592, 2009.
- 477 [33] Daniel B Rubin, Stephen D Van Hooser, and Kenneth D Miller. The stabilized supralinear  
478 network: a unifying circuit motif underlying multi-input integration in sensory cortex. *Neuron*,  
479 85(2):402–417, 2015.
- 480 [34] Henry Markram, Maria Toledo-Rodriguez, Yun Wang, Anirudh Gupta, Gilad Silberberg, and  
481 Caizhi Wu. Interneurons of the neocortical inhibitory system. *Nature reviews neuroscience*,  
482 5(10):793, 2004.
- 483 [35] Bernardo Rudy, Gordon Fishell, SooHyun Lee, and Jens Hjerling-Leffler. Three groups of  
484 interneurons account for nearly 100% of neocortical gabaergic neurons. *Developmental neuro-*  
485 *biology*, 71(1):45–61, 2011.
- 486 [36] Chunyu A Duan, Jeffrey C Erlich, and Carlos D Brody. Requirement of prefrontal and midbrain  
487 regions for rapid executive control of behavior in the rat. *Neuron*, 86(6):1491–1503, 2015.
- 488 [37] Omri Barak. Recurrent neural networks as versatile tools of neuroscience research. *Current*  
489 *opinion in neurobiology*, 46:1–6, 2017.
- 490 [38] David Sussillo and Omri Barak. Opening the black box: low-dimensional dynamics in high-  
491 dimensional recurrent neural networks. *Neural computation*, 25(3):626–649, 2013.
- 492 [39] Robert E Kass and Valérie Ventura. A spike-train probability model. *Neural computation*,  
493 13(8):1713–1720, 2001.

- [40] 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.
- [41] Liam Paninski. Maximum likelihood estimation of cascade point-process neural encoding models. *Network: Computation in Neural Systems*, 15(4):243–262, 2004.
- [42] 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.
- [43] 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.
- [44] 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.
- [45] Blake A Richards and et al. A deep learning framework for neuroscience. *Nature Neuroscience*, 2019.
- [46] Steven H Strogatz. Nonlinear dynamics and chaos: with applications to physics, chemistry, and engineering (Studies in Nonlinearity), Perseus, Cambridge, UK, 1994.
- [47] Rajesh Ranganath, Sean Gerrish, and David Blei. Black box variational inference. In *Artificial Intelligence and Statistics*, pages 814–822, 2014.
- [48] 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.
- [49] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *Proceedings of the 5th International Conference on Learning Representations*, 2017.
- [50] 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.

523 **A Methods**

524 **A.1 Emergent property inference (EPI)**

525 Emergent property inference (EPI) learns distributions of theoretical model parameters that pro-  
 526 duce emergent properties of interest. EPI combines ideas from likelihood-free variational inference  
 527 [18] and maximum entropy flow networks [17]. A maximum entropy flow network is used as a deep  
 528 probability distribution for the parameters, while these samples often parameterize a differentiable  
 529 model simulator, which may lack a tractable likelihood function.

530 Consider model parameterization  $z$  and data  $x$  generated from some theoretical model simulator  
 531 represented as  $p(x | z)$ , which may be deterministic or stochastic. Theoretical models usually have  
 532 known sampling procedures for simulating activity given a circuit parameterization, yet often lack  
 533 an explicit likelihood function due to the nonlinearities and dynamics. With EPI, a distribution  
 534 on parameters  $z$  is learned, that yields an emergent property of interest  $\mathcal{B}$ ,

$$\mathcal{B} \leftrightarrow E_{z \sim q_\theta} [E_{x \sim p(x|z)} [T(x)]] = \mu \quad (13)$$

535 by making an approximation  $q_\theta(z)$  to  $p(z | \mathcal{B})$  (see Section A.1.5). So, over the DSN distribution  
 536  $q_\theta(z)$  of model  $p(x | z)$  for behavior  $\mathcal{B}$ , the emergent properties  $T(x)$  are constrained in expectation  
 537 to  $\mu$ .

538 In deep probability distributions, a simple random variable  $w \sim p_0$  is mapped deterministically  
 539 via a function  $f_\theta$  parameterized by a neural network to the support of the distribution of interest  
 540 where  $z = f_\theta(w) = f_l(\dots f_1(w))$ . Given a theoretical model  $p(x | z)$  and some behavior of interest  
 541  $\mathcal{B}$ , the deep probability distributions are trained by optimizing the neural network parameters  $\theta$  to  
 542 find a good approximation  $q_\theta^*$  within the deep variational family  $Q$  to  $p(z | \mathcal{B})$ .

543 In most settings (especially those relevant to theoretical neuroscience) the likelihood of the behavior  
 544 with respect to the model parameters  $p(T(x) | z)$  is unknown or intractable, requiring an alternative  
 545 to stochastic gradient variational Bayes [4] or black box variational inference[47]. These types  
 546 of methods called likelihood-free variational inference (LFVI, [18]) skate around the intractable  
 547 likelihood function in situations where there is a differentiable simulator. Akin to LFVI, DSNs are  
 548 optimized with the following objective for a given theoretical model, emergent property statistics  
 549  $T(x)$ , and emergent property constraints  $\mu$ :

$$\begin{aligned} q_\theta^*(z) &= \operatorname{argmax}_{q_\theta \in Q} H(q_\theta(z)) \\ \text{s.t. } E_{z \sim q_\theta} [E_{x \sim p(x|z)} [T(x)]] &= \mu \end{aligned} \tag{14}$$

550 Optimizing this objective is a technological accomplishment in its own right, the details of which  
 551 we elaborate in Section A.1.2. Before going through those details, we ground this optimization in  
 552 a toy example.

553 **A.1.1 Example: 2D LDS**

554 To gain intuition for EPI, consider two-dimensional linear dynamical systems,  $\tau \dot{x} = Ax$  with

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

555 that produce a band of oscillations. To do EPI with the dynamics matrix elements as the free  
 556 parameters  $z = [a_1, a_2, a_3, a_4]$ , and fixing  $\tau = 1$ , such that the posterior yields a band of oscillations,  
 557 the emergent property statistics  $T(x)$  are chosen to contain the first- and second-moments of the  
 558 oscillatory frequency  $\Omega$  and the growth/decay factor  $d$  of the oscillating system. To learn the  
 559 distribution of real entries of  $A$  that yield a distribution of  $d$  with mean zero with variance  $0.25^2$ ,  
 560 and oscillation frequency  $\Omega$  with mean 1 Hz with variance  $(0.1\text{Hz})^2$ , then we would select the real  
 561 part of the complex conjugate eigenvalues  $\operatorname{real}(\lambda_1) = d$  (via an arbitrary choice of eigenvalue of the  
 562 dynamics matrix  $\lambda_1$ ) and the positive imaginary component of one of the eigenvalues  $\operatorname{imag}(\lambda_1) =$   
 563  $2\pi\Omega$  as the emergent property statistics. Those emergent property statistics are then constrained  
 564 to

$$\mu = E \begin{bmatrix} \operatorname{real}(\lambda_1) \\ \operatorname{imag}(\lambda_1) \\ (\operatorname{real}(\lambda_1) - 0)^2 \\ (\operatorname{imag}(\lambda_1) - 2\pi\Omega)^2 \end{bmatrix} = \begin{bmatrix} 0.0 \\ 2\pi\Omega \\ 0.25^2 \\ (2\pi 0.1)^2 \end{bmatrix} \tag{15}$$

565 where  $\Omega = 1\text{Hz}$ . Unlike the models we study in the paper which calculate  $E_{x \sim p(x|z)} [T(x)]$  via  
 566 forward simulation, we have a closed form for the eigenvalues of the dynamics matrix.  $\lambda$  can be  
 567 calculated using the quadratic formula:

$$\lambda = \frac{\left(\frac{a_1+a_4}{\tau}\right) \pm \sqrt{\left(\frac{a_1+a_4}{\tau}\right)^2 + 4\left(\frac{a_2a_3-a_1a_4}{\tau}\right)}}{2} \tag{16}$$

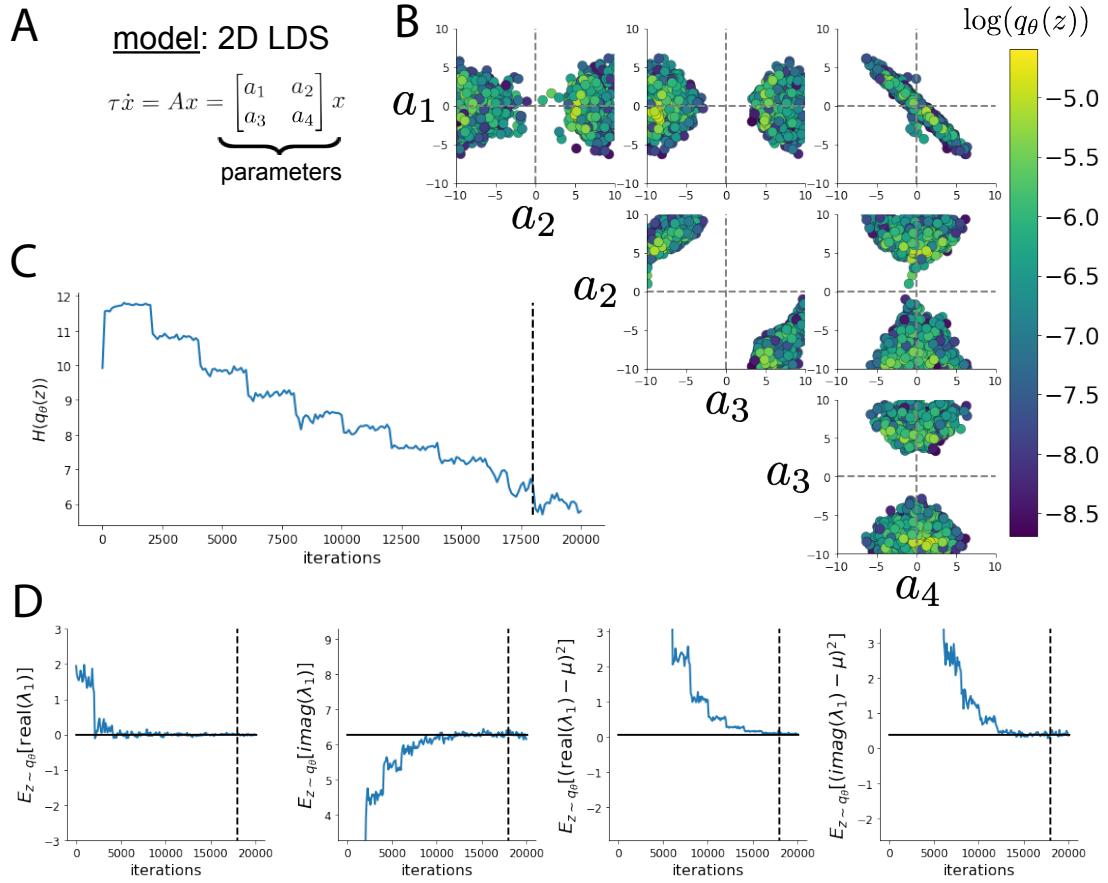


Fig. S2: A. Two-dimensional linear dynamical system model, where real entries of the dynamics matrix  $A$  are the parameters. B. The DSN distribution for a 2D LDS with  $\tau = 1$  that produces an average of 1Hz oscillations with some small amount of variance. C. Entropy throughout the optimization. At the beginning of each augmented Lagrangian epoch (5,000 iterations), the entropy dips 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 move closer to their constraints.

568 where  $\lambda_1$  is the eigenvalue of  $\frac{1}{\tau}A$  with greatest real part. Even though  $E_{x \sim p(x|z)}[T(x)]$  is calculable  
 569 directly via a closed form function and does not require simulation, we cannot derive the distribution  
 570  $q_\theta^*$  directly. This is due to the formally hard problem of the backward mapping: finding the natural  
 571 parameters  $\eta$  from the mean parameters  $\mu$  of an exponential family distribution [48]. Instead, we  
 572 can use EPI to learn the linear system parameters producing such a band of oscillations (Fig. S2B).  
 573 Even this relatively simple system has nontrivial (though intuitively sensible) structure in the  
 574 parameter distribution. To validate our method (further than that of the underlying technology  
 575 on a ground truth solution [17]) we can analytically derive the contours of the probability density  
 576 from the emergent property statistics and values (Fig. S3). In the  $a_1 - a_4$  plane, is a black line  
 577 at  $\text{real}(\lambda_1) = \frac{a_1+a_4}{2} = 0$ , a dotted black line at the standard deviation  $\text{real}(\lambda_1) = \frac{a_1+a_4}{2} \pm 1$ , and a  
 578 grey line at twice the standard deviation  $\text{real}(\lambda_1) = \frac{a_1+a_4}{2} \pm 2$  (Fig. S3A). Here the lines denote the  
 579 set of solutions at fixed behaviors, which overlay the posterior obtained through EPI. The learned  
 580 DSN distribution precisely reflects the desired statistical constraints and model degeneracy in the  
 581 sum of  $a_1$  and  $a_4$ . Intuitively, the parameters equivalent with respect to emergent property statistic  
 582  $\text{real}(\lambda_1)$  have similar log densities.

583 To explain the structure in the bimodality of the DSN posterior, we can look at the imaginary  
 584 component of  $\lambda_1$ . When  $\text{real}(\lambda_1) = \frac{a_1+a_4}{2} = 0$ , we have

$$\text{imag}(\lambda_1) = \begin{cases} \sqrt{\frac{a_1a_4-a_2a_3}{\tau}}, & \text{if } a_1a_4 < a_2a_3 \\ 0 & \text{otherwise} \end{cases} \quad (17)$$

585 When  $\tau = 1$  and  $a_1a_4 > a_2a_3$  (center of distribution above), we have the following equation for the  
 586 other two dimensions:

$$\text{imag}(\lambda_1)^2 = a_1a_4 - a_2a_3 \quad (18)$$

587 Since we constrained  $E_{q_\theta}[\text{imag}(\lambda)] = 2\pi$  (with  $\omega = 1$ ), we can plot contours of the equation  
 588  $\text{imag}(\lambda_1)^2 = a_1a_4 - a_2a_3 = (2\pi)^2$  for various  $a_1a_4$  (Fig. S3A). If  $\sigma_{1,4} = E_{q_\theta}(|a_1a_4 - E_{q_\theta}[a_1a_4]|)$ ,  
 589 then we plot the contours as  $a_1a_4 = 0$  (black),  $a_1a_4 = -\sigma_{1,4}$  (black dotted), and  $a_1a_4 = -2\sigma_{1,4}$   
 590 (grey dotted) (Fig. S3B). This validates the curved structure of the inferred distribution learned  
 591 through EPI. We take steps in negative standard deviation of  $a_1a_4$  (dotted and gray lines), since  
 592 there are few positive values  $a_1a_4$  in the posterior. Subtler model-behavior combinations will have  
 593 even more complexity, further motivating the use of EPI for understanding these systems. Indeed,  
 594 we sample a distribution of systems oscillating near 1Hz (Fig. S4).

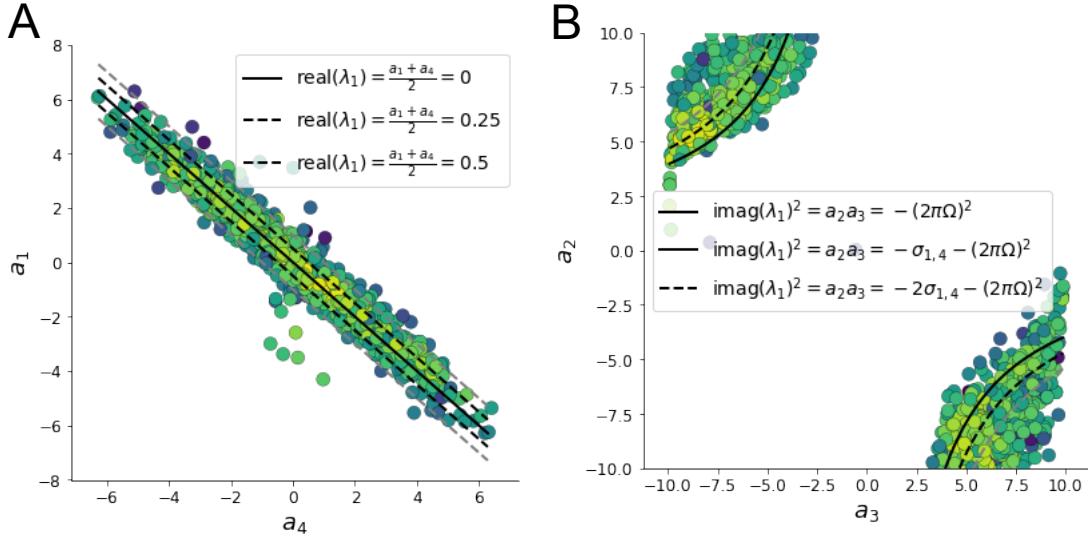


Fig. S3: A. Probability contours in the  $a_1 - a_4$  plane can be derived from the relationship to emergent property statistic of growth/decay factor. B. Probability contours in the  $a_2 - a_3$  plane can be derived from relationship to the emergent property statistic of oscillation frequency.

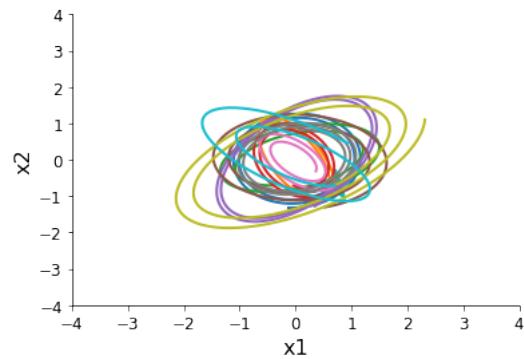


Fig. S4: Sampled dynamical system trajectories from the EPI distribution. Each trajectory is initialized at  $x(0) = \left[ \frac{\sqrt{2}}{2} \quad -\frac{\sqrt{2}}{2} \right]$ .

595 **A.1.2 Augmented Lagrangian optimization**

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

$$L(\theta; \alpha, c) = -H(q_\theta) + \alpha^\top \delta(\theta) + \frac{c}{2} \|\delta(\theta)\|^2 \quad (19)$$

598 where  $\delta(\theta) = E_{z \sim q_\theta} [E_{x \sim p(x|z)} [T(x) - \mu]]$ ,  $\alpha \in \mathcal{R}^m$  are the Lagrange multipliers and  $c$  is the penalty  
 599 coefficient. For a fixed  $(\alpha, c)$ ,  $\theta$  is optimized with stochastic gradient descent. A low value of  $c$  is  
 600 used initially, and increased during each augmented Lagrangian epoch – a period of optimization  
 601 with fixed  $\alpha$  and  $c$  for a given number of stochastic optimization iterations. Similarly,  $\alpha$  is tuned  
 602 each epoch based on the constraint violations. For the linear 2-dimensional system (Fig. S2C)  
 603 optimization hyperparameters are initialized to  $c_1 = 10^{-4}$  and  $\alpha_1 = 0$ . The penalty coefficient  
 604 is updated based on a hypothesis test regarding the reduction in constraint violation. The p-  
 605 value of  $E[\|\delta(\theta_{k+1})\|] > \gamma E[\|\delta(\theta_k)\|]$  is computed, and  $c_{k+1}$  is updated to  $\beta c_k$  with probability  
 606  $1 - p$ . Throughout the project,  $\beta = 4.0$  and  $\gamma = 0.25$  is used. The other update rule is  $\alpha_{k+1} =$   
 607  $\alpha_k + c_k \frac{1}{n} \sum_{i=1}^n (T(x^{(i)}) - \mu)$ . In this example, each augmented Lagrangian epoch ran for 2,000  
 608 iterations. We consider the optimization to have converged when a null hypothesis test of constraint  
 609 violations being zero is accepted for all constraints at a significance threshold 0.05. This is the dotted  
 610 line on the plots below depicting the optimization cutoff of EPI optimization for the 2-dimensional  
 611 linear system. If the optimization is left to continue running, entropy usually decreases, and  
 612 structural pathologies in the distribution may be introduced.

613 The intention is that  $c$  and  $\alpha$  start at values encouraging entropic growth early in optimization.  
 614 Then, as they increase in magnitude with each training epoch, the constraint satisfaction terms are  
 615 increasingly weighted, resulting in a decrease in entropy. Rather than using a naive initialization,  
 616 before EPI, we optimize the deep probability distribution parameters to generate samples of an  
 617 isotropic Gaussian of a selected variance, such as 1.0 for the 2D LDS example. This provides a  
 618 convenient starting point, whose level of entropy is controlled by the user.

619 **A.1.3 Normalizing flows**

620 Since we are optimizing parameters  $\theta$  of our deep probability distribution with respect to the  
 621 entropy, we will need to take gradients with respect to the log-density of samples from the deep  
 622 probability distribution.

$$H(q_\theta(z)) = \int -q_\theta(z) \log(q_\theta(z)) dz = E_{z \sim q_\theta} [-\log(q_\theta(z))] = E_{\omega \sim q_0} [-\log(q_\theta(f_\theta(\omega)))] \quad (20)$$

623

$$\nabla_\theta H(q_\theta(z)) = E_{\omega \sim q_0} [-\nabla_\theta \log(q_\theta(f_\theta(\omega)))] \quad (21)$$

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

$$q(z') = q(f^{-1}(z')) \left| \det \frac{\partial f^{-1}(z')}{\partial z'} \right| = q(z) \left| \det \frac{\partial f(z)}{\partial z} \right|^{-1} \quad (22)$$

627 However, this computation has cubic complexity in dimensionality for fully connected layers. By  
 628 restricting our layers to normalizing flows [14] – bijective functions with fast log determinant ja-  
 629 cobian computations, we can tractably optimize deep generative models with objectives that are a  
 630 function of sample density, like entropy. Most of our analyses use real NVP [49], which have proven  
 631 effective in our architecture searches, and have the advantageous features of fast sampling and fast  
 632 density evaluation.

633 **A.1.4 Related work**

634 (To come)

635

636 **A.1.5 Emergent property inference as variational inference in an exponential family**

637 (To come)

638

639 **A.2 Theoretical models**

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

642 **A.2.1 Stomatogastric ganglion**

643 Each neuron's membrane potential  $x_m(t)$  is the solution of the following differential equation.

$$C_m \frac{dx_m}{dt} = -[h_{leak}(x; z) + h_{Ca}(x; z) + h_K(x; z) + h_{hyp}(x; z) + h_{elec}(x; z) + h_{syn}(x; z)] \quad (23)$$

644 The membrane potential of each neuron is affected by the leak, calcium, potassium, hyperpolariza-  
 645 tion, electrical and synaptic currents, respectively. The capacitance of the cell membrane was set to  
 646  $C_m = 1nF$ . Each current is a function of the neuron's membrane potential  $x_m$  and the parameters  
 647 of the circuit such as  $g_{el}$  and  $g_{syn}$ , whose effect on the circuit is considered in the motivational  
 648 example of EPI in Fig. 1. Specifically, the currents are the difference in the neuron's membrane  
 649 potential and that current type's reversal potential multiplied by a conductance:

$$h_{leak}(x; z) = g_{leak}(x_m - V_{leak}) \quad (24)$$

$$h_{elec}(x; z) = g_{el}(x_m^{post} - x_m^{pre}) \quad (25)$$

$$h_{syn}(x; z) = g_{syn}S_\infty^{pre}(x_m^{post} - V_{syn}) \quad (26)$$

$$h_{Ca}(x; z) = g_{Ca}M_\infty(x_m - V_{Ca}) \quad (27)$$

$$h_K(x; z) = g_KN(x_m - V_K) \quad (28)$$

$$h_{hyp}(x; z) = g_hH(x_m - V_{hyp}) \quad (29)$$

655 The reversal potentials were set to  $V_{leak} = -40mV$ ,  $V_{Ca} = 100mV$ ,  $V_K = -80mV$ ,  $V_{hyp} = -20mV$ ,  
 656 and  $V_{syn} = -75mV$ . The other conductance parameters were fixed to  $g_{leak} = 1 \times 10^{-4}\mu S$ .  $g_{Ca}$ ,  
 657  $g_K$ , and  $g_{hyp}$  had different values based on fast, intermediate (hub) or slow neuron. Fast:  $g_{Ca} =$   
 658  $1.9 \times 10^{-2}$ ,  $g_K = 3.9 \times 10^{-2}$ , and  $g_{hyp} = 2.5 \times 10^{-2}$ . Intermediate:  $g_{Ca} = 1.7 \times 10^{-2}$ ,  $g_K = 1.9 \times 10^{-2}$ ,  
 659 and  $g_{hyp} = 8.0 \times 10^{-3}$ . Intermediate:  $g_{Ca} = 8.5 \times 10^{-3}$ ,  $g_K = 1.5 \times 10^{-2}$ , and  $g_{hyp} = 1.0 \times 10^{-2}$ .

660 Furthermore, the Calcium, Potassium, and hyperpolarization channels have time-dependent gating  
 661 dynamics dependent on steady-state gating variables  $M_\infty$ ,  $N_\infty$  and  $H_\infty$ , respectively.

$$M_\infty = 0.5 \left( 1 + \tanh \left( \frac{x_m - v_1}{v_2} \right) \right) \quad (30)$$

$$\frac{dN}{dt} = \lambda_N(N_\infty - N) \quad (31)$$

$$N_\infty = 0.5 \left( 1 + \tanh \left( \frac{x_m - v_3}{v_4} \right) \right) \quad (32)$$

$$\lambda_N = \phi_N \cosh \left( \frac{x_m - v_3}{2v_4} \right) \quad (33)$$

$$\frac{dH}{dt} = \frac{(H_\infty - H)}{\tau_h} \quad (34)$$

$$H_\infty = \frac{1}{1 + \exp\left(\frac{x_m + v_5}{v_6}\right)} \quad (35)$$

$$\tau_h = 272 - \left( \frac{-1499}{1 + \exp\left(\frac{-x_m + v_7}{v_8}\right)} \right) \quad (36)$$

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  $v_{th} = -25mV$ . These are the same parameter values used in [19].

671 Finally, there is a synaptic gating variable as well:

$$S_\infty = \frac{1}{1 + \exp\left(\frac{v_{th} - x_m}{v_9}\right)} \quad (37)$$

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  $T = 500$  time steps of  $dt = 25ms$ . In EPI, since gradients are taken through the simulation process, the number of time steps are kept as modest if possible. The chosen  $dt$  and  $T$  were the most computationally convenient choices yielding accurate frequency measurement.

Our original approach to measuring frequency was to take the max of the fast Fourier transform (FFT) of the simulated time series. There are a few key considerations here. One is resolution in frequency space. Each FFT entry will correspond to a signal frequency of  $\frac{F_s k}{N}$ , where  $N$  is the number of samples used for the FFT,  $F_s = \frac{1}{dt}$ , and  $k \in [0, 1, \dots, N - 1]$ . Our resolution is improved by increasing  $N$  and decreasing  $dt$ . Increasing  $N = T - b$ , where  $b$  is some fixed number of buffer burn-in initialization samples, necessitates an increase in simulation time steps  $T$ , which directly increases computational cost. Increasing  $F_s$  (decreasing  $dt$ ) increases system approximation accuracy, but requires more time steps before a full cycle is observed. At the level of  $dt = 0.025$ , thousands of temporal samples were required for resolution of .01Hz. These challenges in frequency resolution with the discrete Fourier transform motivated the use of an alternative basis of complex exponentials. Instead, we used a basis of complex exponentials with frequencies from 0.0-1.0 Hz at 0.01Hz resolution,  $\Phi = [0.0, 0.01, \dots, 1.0]^\top$

690 Another consideration was that the frequency spectra of the hub neuron has several peaks. This  
691 was due to high-frequency sub-threshold activity. The maximum frequency was often not the firing

frequency. 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 pre-processing, the maximum frequency in the filter bank accurately reflected the firing frequency.

Finally, to differentiate through the maximum frequency identification step, we used a sum-of-powers normalization strategy: Let  $\mathcal{X}_i \in \mathcal{C}^{|\Phi|}$  be the complex exponential filter bank dot products with the signal  $x_i \in \mathcal{R}^N$ , where  $i \in \{\text{f1}, \text{f2}, \text{hub}, \text{s1}, \text{s2}\}$ . The “frequency identification” vector is

$$u_i = \frac{|\mathcal{X}_i|^\alpha}{\sum_{k=1}^N |\mathcal{X}_i(k)|^\alpha} \quad (38)$$

The frequency is then calculated as  $\Omega_i = u_i^\top \Phi$  with  $\alpha = 100$ .

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 of the firing frequencies. The first moments are set to 0.55Hz, while the second moments are set to  $0.025\text{Hz}^2$ .

$$E \begin{bmatrix} \Omega_{\text{f1}} \\ \Omega_{\text{f2}} \\ \Omega_{\text{hub}} \\ \Omega_{\text{s1}} \\ \Omega_{\text{s2}} \\ (\Omega_{\text{f1}} - 0.55)^2 \\ (\Omega_{\text{f2}} - 0.55)^2 \\ (\Omega_{\text{hub}} - 0.55)^2 \\ (\Omega_{\text{s1}} - 0.55)^2 \\ (\Omega_{\text{s2}} - 0.55)^2 \end{bmatrix} = \begin{bmatrix} 0.55 \\ 0.55 \\ 0.55 \\ 0.55 \\ 0.55 \\ 0.025^2 \\ 0.025^2 \\ 0.025^2 \\ 0.025^2 \\ 0.025^2 \end{bmatrix} \quad (39)$$

For EPI in Fig 2C, we used a real NVP architecture with two coupling layers. Each coupling layer had two hidden layers of 10 units each, and we mapped onto a support of  $z \in \left[ \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 10 \\ 8 \end{bmatrix} \right]$ . We have shown the EPI optimization that converged with maximum entropy across 2 random seeds and augmented Lagrangian coefficient initializations of  $c_0=0, 2$ , and 5.

708 **A.2.2 Primary visual cortex**709 The dynamics of each neural populations average rate  $x = \begin{bmatrix} x_E \\ x_P \\ x_S \\ x_V \end{bmatrix}$  are given by:

$$\tau \frac{dx}{dt} = -x + [Wx + h]_+^n \quad (40)$$

710 Some neuron-types largely lack synaptic projections to other neuron-types [50], and it is popular  
711 to only consider a subset of the effective connectivities [20].

$$W = \begin{bmatrix} W_{EE} & W_{EP} & W_{ES} & 0 \\ W_{PE} & W_{PP} & W_{PS} & 0 \\ W_{SE} & 0 & 0 & W_{SV} \\ W_{VE} & W_{VP} & W_{VS} & 0 \end{bmatrix} \quad (41)$$

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

$$M = \begin{bmatrix} 0.36 & 0.48 & 0.31 & 0.28 \\ 1.49 & 0.68 & 0.50 & 0.18 \\ 0.86 & 0.42 & 0.15 & 0.32 \\ 1.31 & 0.41 & 0.52 & 0.37 \end{bmatrix} \quad (42)$$

714 and connection probability

$$C = \begin{bmatrix} 0.16 & 0.411 & 0.424 & 0.087 \\ 0.395 & .451 & 0.857 & 0.02 \\ 0.182 & 0.03 & 0.082 & 0.625 \\ 0.105 & 0.22 & 0.77 & 0.028 \end{bmatrix} \quad (43)$$

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

$$W_{\text{full}} = C \odot M = \begin{bmatrix} 0.16 & 0.411 & 0.424 & 0.087 \\ 0.395 & .451 & 0.857 & 0.02 \\ 0.182 & 0.03 & 0.082 & 0.625 \\ 0.105 & 0.22 & 0.77 & 0.028 \end{bmatrix} \quad (44)$$

717 From use the entries of this full effective connectivity matrix that are not considered to be ineffectual.

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

$$h = b + dh \quad (45)$$

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

725 We can describe the dynamics of this system more generally by

$$\dot{x}_i = -x_i + f(u_i) \quad (46)$$

726 where the input to each neuron is

$$u_i = \sum_j W_{ij}x_j + h_i \quad (47)$$

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

$$\frac{dx_{ss}}{dh} = F(W \frac{dx_{ss}}{dh} + I) \quad (48)$$

728 which is calculable by

$$\frac{dx_{ss}}{dh} = (F^{-1} - W)^{-1} \quad (49)$$

729 The emergent property we considered was the first and second moments of the change in rate  $dx$   
 730 between the baseline input  $h = b$  and  $h = b + dh$ . We use the following notation to indicate that  
 731 the emergent property statistics were set to the following values:

$$\mathcal{B}(\alpha, y) \leftrightarrow E \begin{bmatrix} dx_{\alpha,ss} \\ (dx_{\alpha,ss} - y)^2 \end{bmatrix} = \begin{bmatrix} y \\ 0.01^2 \end{bmatrix} \quad (50)$$

732 In the final analysis for this model, we sweep the input one neuron at a time away from the mode  
 733 of each inferred distributions  $dh^* = z^* = \text{argmax}_z \log q_\theta(z \mid \mathcal{B}(\alpha, 0.1))$ . The differential responses  
 734  $dx_{\alpha,ss}$  are examined at perturbed inputs  $h = b + dh^* + \Delta h_\alpha u_\alpha$  where  $u_\alpha$  is a unit vector in the  
 735 dimension of  $\alpha$  and  $\Delta h_\alpha \in [-15, 15]$ .

736 For each  $\mathcal{B}(\alpha, y)$  with  $\alpha \in \{E, P, S, V\}$  and  $y \in \{0.1, 0.5\}$ , we ran EPI with five different random  
 737 initial seeds using an architecture of four coupling layers, each with two hidden layers of 10 units.

738 We set  $c_0 = 10^5$ . The support of the learned distribution was restricted to  $z_i \in [-5, 5]$ .

<sup>739</sup> **A.2.3 Superior colliculus**

<sup>740</sup> There are four total units: two in each hemisphere corresponding to the Pro/Contra and Anti/Ipsi  
<sup>741</sup> populations. Each unit has an activity ( $x_i$ ) and internal variable ( $u_i$ ) related by

$$x_i(t) = \left( \frac{1}{2} \tanh \left( \frac{v_i(t) - \epsilon}{\zeta} \right) + \frac{1}{2} \right) \quad (51)$$

<sup>742</sup>  $\epsilon = 0.05$  and  $\zeta = 0.5$  control the position and shape of the nonlinearity, respectively.

<sup>743</sup> We can order the elements of  $x_i$  and  $v_i$  into vectors  $x$  and  $v$  with elements

$$x = \begin{bmatrix} x_{LP} \\ x_{LA} \\ x_{RP} \\ x_{RA} \end{bmatrix} \quad v = \begin{bmatrix} v_{LP} \\ v_{LA} \\ v_{RP} \\ v_{RA} \end{bmatrix} \quad (52)$$

<sup>744</sup> The internal variables follow dynamics:

$$\tau \frac{dv}{dt} = -v + Wx + h + \sigma dB \quad (53)$$

<sup>745</sup> with time constant  $\tau = 0.09s$  and Gaussian noise  $\sigma dB$  controlled by the magnitude of  $\sigma = 1.0$ . The  
<sup>746</sup> weight matrix has 8 parameters  $sW_P$ ,  $sW_A$ ,  $vW_{PA}$ ,  $vW_{AP}$ ,  $hW_P$ ,  $hW_A$ ,  $dW_{PA}$ , and  $dW_{AP}$  (Fig.  
<sup>747</sup> 4B).

$$W = \begin{bmatrix} sW_P & vW_{PA} & hW_P & dW_{PA} \\ vW_{AP} & sW_A & dW_{AP} & hW_A \\ hW_P & dW_{PA} & sW_P & vW_{PA} \\ dW_{AP} & hW_A & vW_{AP} & sW_A \end{bmatrix} \quad (54)$$

<sup>748</sup> The system receives five inputs throughout each trial, which has a total length of 1.8s.

$$h = h_{\text{rule}} + h_{\text{choice-period}} + h_{\text{light}} \quad (55)$$

<sup>749</sup> There are rule-based inputs depending on the condition,

$$h_{P,\text{rule}}(t) = \begin{cases} I_{P,\text{rule}} \begin{bmatrix} 1 & 0 & 0 & 1 \end{bmatrix}^\top, & \text{if } t \leq 1.2s \\ 0, & \text{otherwise} \end{cases} \quad (56)$$

<sup>750</sup>

$$h_{A,\text{rule}}(t) = \begin{cases} I_{A,\text{rule}} \begin{bmatrix} 0 & 1 & 1 & 0 \end{bmatrix}^\top, & \text{if } t \leq 1.2s \\ 0, & \text{otherwise} \end{cases} \quad (57)$$

751 a choice-period input,

$$h_{\text{choice}}(t) = \begin{cases} I_{\text{choice}} \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix}^\top, & \text{if } t > 1.2s \\ 0, & \text{otherwise} \end{cases} \quad (58)$$

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

$$h_{\text{light}}(t) = \begin{cases} I_{\text{light}} \begin{bmatrix} 1 & 1 & 0 & 0 \end{bmatrix}^\top, & \text{if } t > 1.2s \text{ and Left} \\ I_{\text{light}} \begin{bmatrix} 0 & 0 & 1 & 1 \end{bmatrix}^\top, & \text{if } t > 1.2s \text{ and Right} \\ 0, & t \leq 1.2s \end{cases} \quad (59)$$

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

754 To produce a Bernoulli rate of  $p_{LP}$  in the Left, Pro condition (we can generalize this to either cue,  
755 or stimulus condition), let  $\hat{p}_i$  be the empirical average steady state (ss) response (final  $x_{LP}$  at end  
756 of task) over M=500 Gaussian noise draws for a given SC model parameterization  $z_i$ :

$$\hat{p}_i = E_{\sigma dB} [x_{LP,ss} | s = L, c = P, z_i] = \frac{1}{M} \sum_{j=1}^M x_{LP,ss}(s = L, c = P, z_i, \sigma dB_j) \quad (60)$$

757 For the first constraint, the average over posterior samples (from  $q_\theta(z)$ ) to be  $p_{LP}$ :

$$E_{z_i \sim q_\phi} [E_{\sigma dB} [x_{LP,ss} | s = L, c = P, z_i]] = E_{z_i \sim q_\phi} [\hat{p}_i] = p_{LP} \quad (61)$$

758 We can then ask that the variance of the steady state responses across Gaussian draws, is the  
759 Bernoulli variance for the empirical rate  $\hat{p}_i$ .

$$E_{z \sim q_\phi} [\sigma_{err}^2] = 0 \quad (62)$$

760

$$\sigma_{err}^2 = Var_{\sigma dB} [x_{LP,ss} | s = L, c = P, z_i] - \hat{p}_i(1 - \hat{p}_i) \quad (63)$$

761 We have an additional constraint that the Pro neuron on the opposite hemisphere should have the  
762 opposite value. We can enforce this with a final constraint:

$$E_{z \sim q_\phi} [d_P] = 1 \quad (64)$$

763

$$E_{\sigma dB} [(x_{LP,ss} - x_{RP,ss})^2 | s = L, c = P, z_i] \quad (65)$$

764 We refer to networks obeying these constraints as Bernoulli, winner-take-all networks. Since the  
765 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 two variables bounded from 0 to 1 is 1, we do not need to control the second moment of these test statistics. In reality, 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 can only be undershot. 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 for these technically infeasible constraints to 0.05.

Training DSNs to learn distributions of dynamical system parameterizations that produce Bernoulli responses at a given rate (with small variance around that rate) was harder to do 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.

In total, the emergent property of rapid task switching accuracy at level  $p$  was defined as

$$\mathcal{B}(p) \leftrightarrow \begin{bmatrix} \hat{p}_P \\ \hat{p}_A \\ (\hat{p}_P - p)^2 \\ (\hat{p}_A - p)^2 \\ \sigma_{P,err}^2 \\ \sigma_{A,err}^2 \\ d_P \\ d_A \end{bmatrix} = \begin{bmatrix} p \\ p \\ 0.15^2 \\ 0.15^2 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} \quad (66)$$

For each accuracy level  $p$ , we ran EPI for 10 different random seeds and selected the maximum entropy solution using an architecture of 10 planar flows with  $c_0 = 2$ . The support of  $z$  was  $\mathcal{R}^8$ .

#### 783 A.2.4 Rank-1 RNN

Recent work establishes a link between RNN connectivity weights and the resulting dynamical responses of the network, using dynamic mean field theory (DMFT) [22]. Specifically, DMFT describes the properties of activity in infinite-size neural networks given a distribution on the connectivity weights. In such a model, the connectivity of a rank-1 RNN (which was sufficient for

788 our task), has weight matrix  $W$ , whis is the sum of a random component with strength determined  
 789 by  $g$  and a structured component determined by the outer product of vectors  $m$  and  $n$ :

$$W = g\chi + \frac{1}{N}mn^\top, \quad (67)$$

790 where the activity  $x$  evolves as and  $I(t)$  is some input,  $\phi$  is the tanh nonlinearity, and  $\chi_{ij} \sim \mathcal{N}(0, \frac{1}{N})$ .  
 791 The entries of  $m$  and  $n$  are drawn from Gaussian distributions  $m_i \sim \mathcal{N}(M_m, 1)$  and  $n_i \sim \mathcal{N}(M_n, 1)$ .  
 792 From such a parameterization, this theory produces consistency equations for the dynamic mean  
 793 field variables in terms of parameters like  $g$ ,  $M_m$ , and  $M_n$ , which we study in Section 3.5. That  
 794 is the dynamic mean field variables (e.g. the activity along along a vector  $\kappa_v$ , the total variance  
 795  $\Delta_0$ , structured variance  $\Delta_\infty$ , and the chaotic variance  $\Delta_T$ ) are written as functions of one another  
 796 in terms of connectivity parameters. The values of these variables can be used obtained using a  
 797 nonlinear system of equations solver. These dynamic mean field variables are then cast as task-  
 798 relevant variables with respect to the context of the provided inputs. Mastrogiuseppe et al. designed  
 799 low-rank RNN connectivities via minimalist connectivity parameters to solve canonical tasks from  
 800 behavioral neuroscience.

801 We consider the DMFT equation solver as a black box that takes in a low-rank parameterization  $z$   
 802 (e.g.  $z = [g \ M_m \ M_n]$ ) and outputs the values of the dynamic mean field variables, of which we  
 803 cast  $\kappa_w$  and  $\Delta_T$  as task-relevant variables  $\mu_{\text{post}}$  and  $\sigma_{\text{post}}^2$  in the Gaussian posterior conditioning  
 804 toy example. Importantly, the solution produced by the solver is differentiable with respect to the  
 805 input parameters, allowing us to use DMFT to calculate the emergent property statistics in EPI  
 806 to learn distributions on such connectivity parameters of RNNs that execute tasks.

807 Specifically, we solve for the mean field variables  $\kappa_w$ ,  $\kappa_n$ ,  $\Delta_0$  and  $\Delta_\infty$ , where the readout is nominally  
 808 chosen to point in the unit orthant  $w = [1 \ \dots \ 1]^\top$ . The consistency equations for these variables  
 809 in the presence of an constant input  $I(t) = y - (n - M_n)$  can be derived following [22] are

$$\begin{aligned} \kappa_w &= F(\kappa_w, \kappa_n, \Delta_0, \Delta_\infty) = M_m \kappa_n + y \\ \kappa_n &= G(\kappa_w, \kappa_n, \Delta_0, \Delta_\infty) = M_n \langle [\phi_i] \rangle + \langle [\phi'_i] \rangle \\ \frac{\Delta_0^2 - \Delta_\infty^2}{2} &= H(\kappa_w, \kappa_n, \Delta_0, \Delta_\infty) = g^2 \left( \int \mathcal{D}z \Phi^2(\kappa_w + \sqrt{\Delta_0} z) - \int \mathcal{D}z \int \mathcal{D}x \Phi(\kappa_w + \sqrt{\Delta_0 - \Delta_\infty} x + \sqrt{\Delta_\infty} z) \right) \\ &\quad + (\kappa_n^2 + 1)(\Delta_0 - \Delta_\infty) \\ \Delta_\infty &= L(\kappa_w, \kappa_n, \Delta_0, \Delta_\infty) = g^2 \int \mathcal{D}z \left[ \int \mathcal{D}x \phi(\kappa_w + \sqrt{\Delta_0 - \Delta_\infty} x + \sqrt{\Delta_\infty} z) \right]^2 + \kappa_n^2 + 1 \end{aligned} \quad (68)$$

810 where  $z$  here is a gaussian integration variable. We can solve these equations by simulating the

811 following Langevin dynamical system.

$$\begin{aligned}
 x(t) &= \frac{\Delta_0(t)^2 - \Delta_\infty(t)^2}{2} \\
 \Delta_0(t) &= \sqrt{2x(t) + \Delta_\infty(t)^2} \\
 \dot{\kappa}_w(t) &= -\kappa_w(t) + F(\kappa_w(t), \kappa_n(t), \Delta_0(t), \Delta_\infty(t)) \\
 \dot{\kappa}_n(t) &= -\kappa_n + G(\kappa_w(t), \kappa_n(t), \Delta_0(t), \Delta_\infty(t)) \\
 \dot{x}(t) &= -x(t) + H(\kappa_w(t), \kappa_n(t), \Delta_0(t), \Delta_\infty(t)) \\
 \dot{\Delta}_\infty(t) &= -\Delta_\infty(t) + L(\kappa_w(t), \kappa_n(t), \Delta_0(t), \Delta_\infty(t))
 \end{aligned} \tag{69}$$

812 Then, the temporal variance, which is necessary for the Gaussian posterior conditioning example,  
813 is simply calculated via

$$\Delta_T = \Delta_0 - \Delta_\infty \tag{70}$$

### 814 A.3 Supplementary Figures

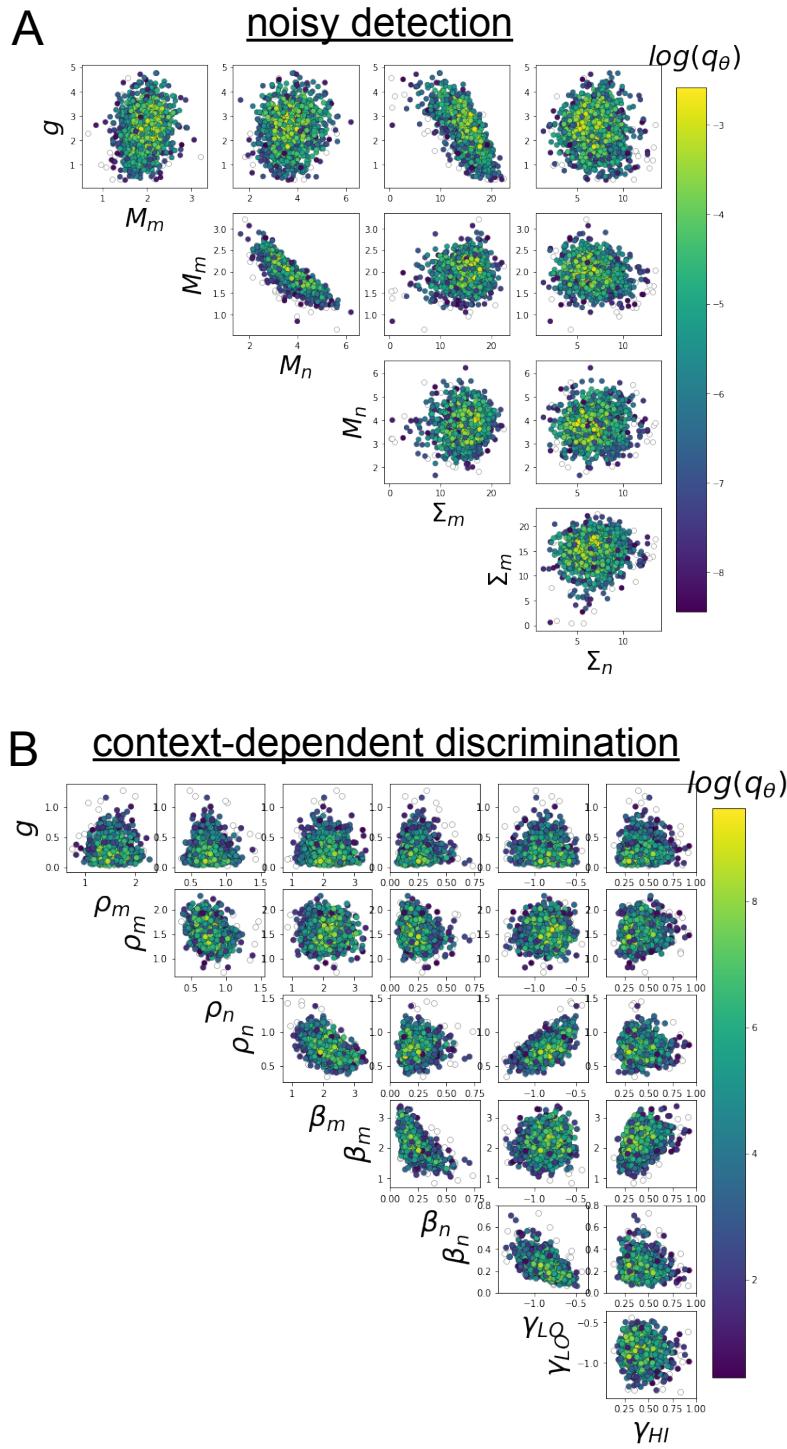


Fig. S1: A. EPI for rank-1 networks doing discrimination. B. EPI for rank-2 networks doing context-dependent discrimination.