

Interrogating theoretical models of neural computation with deep inference

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

¹ 1 Abstract

² The cornerstone of theoretical neuroscience is the circuit model: a system of equations that captures
³ a hypothesized neural mechanism of scientific importance. Such models are valuable when they give
⁴ rise to an experimentally observed phenomenon – whether behavioral or in terms of neural activity –
⁵ and thus can offer insight into neural computation. The operation of these circuits, like all models,
⁶ critically depends on the choices of model parameters. Historically, the gold standard has been
⁷ to analytically derive the relationship between model parameters and computational properties.
⁸ However, this enterprise quickly becomes infeasible as biologically realistic constraints are included
⁹ into the model, often resulting in *ad hoc* approaches to understanding the relationship between
¹⁰ model and computation. We bring recent machine learning techniques – the use of deep generative
¹¹ models for probabilistic inference – to bear on this problem, learning distributions of parameters
¹² that produce the specified properties of computation. Importantly, the techniques we introduce offer
¹³ a principled means to understand the implications of model parameter choices on computational
¹⁴ properties of interest. We motivate this methodology with a worked example analyzing sensitivity in
¹⁵ the stomatogastric ganglion. We then use it to generate insights into neuron-type input-responsivity
¹⁶ in a model of primary visual cortex, a new understanding of rapid task switching in superior
¹⁷ colliculus models, and attribution of bias in recurrent neural networks solving a toy mathematical
¹⁸ problem. More generally, this work suggests a departure from realism vs tractability considerations,
¹⁹ towards the use of modern machine learning for sophisticated interrogation of biologically relevant
²⁰ models.

²¹ 2 Introduction

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

27 when configured appropriately, give rise to a measurable signature of a computation. The work of
28 analyzing a model then becomes the inverse problem: given a computation of interest, how can we
29 reason about these suitable parameter configurations – their likely values, their uniquenesses and
30 degeneracies, their attractor states and phase transitions, and more?

31 Consider the idealized practice: a theorist considers a model carefully and analytically derives how
32 model parameters govern the computation. Seminal examples of this gold standard include our
33 field’s understanding of memory capacity in associative neural networks [2], chaos and autocorrela-
34 tion timescales in random neural networks [3], and the paradoxical effect in excitatory/inhibitory
35 networks [4]. Unfortunately, as circuit models include more biological realism, theory via analytic
36 derivation becomes intractable. This fact creates an unfavorable tradeoff for the theorist. On the
37 one hand, one may tractably analyze systems of equations with unrealistic assumptions (for ex-
38 ample symmetry or gaussianity), producing accurate inferences about parameters of a too-simple
39 model. On the other hand, one may choose a more biologically relevant model at the cost of *ad hoc*
40 approaches to analysis (simply examining simulated activity), producing questionable or partial
41 inferences about parameters of an appropriately complex, scientifically relevant model.

42 Of course, this same tradeoff has been confronted in many scientific fields and engineering problems
43 characterized by the need to do inference in complex models. In response, the machine learning
44 community has made remarkable progress in recent years, via the use of deep neural networks as a
45 powerful inference engine: a flexible function family that can map observed phenomena (in this case
46 the measurable signal of some computation) back to probability distributions quantifying the likely
47 parameter configurations. One celebrated example of this approach from the machine learning
48 community, from which we draw key inspiration for this work, is the variational autoencoder [5, 6],
49 which uses a deep neural network to induce an (approximate) posterior distribution on hidden
50 variables in a latent variable model, given data. Indeed, these tools have been used to great success
51 in neuroscience as well, in particular for interrogating parameters (sometimes treated as hidden
52 states) in models of both cortical population activity [7, 8, 9, 10] and animal behavior [11, 12, 13].
53 These works have used deep neural networks to expand the expressivity and accuracy of statistical
54 models of neural data [14].

55 However, these inference tools have not significantly influenced the study of theoretical neuroscience
56 models, for at least three reasons. First, at a practical level, the nonlinearities and dynamics of
57 many theoretical models are such that conventional inference tools typically produce a narrow set
58 of insights into these models. Indeed, only in the last few years has the deep learning toolkit

59 expanded to a point of relevance to this class of problem. Second, the object of interest from a
60 theoretical model is not typically data itself, but rather a qualitative phenomenon – inspection of
61 model behavior, or better, a measurable signature of some computation – an *emergent property* of
62 the model. Third, because theoreticians work carefully to construct a model that has biological
63 relevance, such a model as a result often does not fit cleanly into the framing of a statistical model.
64 Technically, because many such models stipulate a noisy system of differential equations that can
65 only be sampled or realized through forward simulation, they lack the explicit likelihood and priors
66 central to the probabilistic modeling toolkit.

67 To address these three challenges, we developed an inference methodology – ‘emergent property
68 inference’ – which learns a distribution over parameter configurations in a theoretical model. Crit-
69 ically, this distribution is such that draws from the distribution (parameter configurations) corre-
70 spond to systems of equations that give rise to a specified emergent property. First, we stipulate a
71 bijective deep neural network that induces a flexible family of probability distributions over model
72 parameterizations with a probability density we can calculate [15, 16, 17]. Second, we quantify
73 the notion of emergent properties as a set of moment constraints on datasets generated by the
74 model. Thus, an emergent property is not a single data realization, but a phenomenon or a fea-
75 ture of the model, which is the central object of interest to the theorist (unlike say the statistical
76 neuroscientist). Conditioning on an emergent property requires a variant of deep probabilistic
77 inference methods, which we have previously introduced [18]. Third, because we cannot assume
78 the theoretical model has explicit likelihood on data or the emergent property of interest, we use
79 stochastic gradient techniques in the spirit of likelihood free variational inference [19]. Taken to-
80 gether, emergent property inference (EPI) provides a methodology for inferring and then reasoning
81 about parameter configurations that give rise to particular emergent phenomena in theoretical
82 models. To clarify the technical details of EPI, we use it to analyze network syncing in a classic
83 model of the stomatogastric ganglion [20].

84 Equipped with this methodology, we then investigated three models of current importance in theo-
85 retical neuroscience. These models were chosen to demonstrate generality through ranges of biolog-
86 ical realism (conductance-based biophysics to recurrent neural networks), neural system function
87 (pattern generation to abstract cognitive function), and network scale (four to infinite neurons).
88 First, we use EPI to produce a set of verifiable hypotheses of input-responsivity in a four neuron-
89 type dynamical model of primary visual cortex; we then validate these hypotheses in the model.
90 Second, we demonstrated how the systematic application of EPI to levels of task performance can

91 generate experimentally testable hypotheses regarding connectivity in superior colliculus. Third,
 92 we use EPI to uncover the sources of bias in a low-rank recurrent neural network executing a toy
 93 mathematical computation. The novel scientific insights offered by EPI contextualize and clarify
 94 the previous studies exploring these models [20, 21, 22, 23] and more generally offer a quantitative
 95 grounding for theoretical models going forward, pointing a way to how rigorous statistical inference
 96 can enhance theoretical neuroscience at large.

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

107 3 Results

108 3.1 Motivating emergent property inference of theoretical models

109 Consideration of the typical workflow of theoretical modeling clarifies the need for emergent prop-
 110 erty inference. First, the theorist designs or chooses an existing model that, it is hypothesized,
 111 captures the computation of interest. To ground this process in a well-known example, consider
 112 the stomatogastric ganglion (STG) of crustaceans, a small neural circuit which generates multiple
 113 rhythmic muscle activation patterns for digestion [27]. A model of the STG [20] is shown schemat-
 114 ically in Figure 1A, and note that the behavior of this model will be critically dependent on its
 115 parameterization – the choices of conductance parameters $z = [g_{el}, g_{synA}]$. Specifically, the two
 116 fast neurons ($f1$ and $f2$) mutually inhibit one another, and oscillate at a faster frequency than the
 117 mutually inhibiting slow neurons ($s1$ and $s2$), and the hub neuron (hub) couples with the fast or
 118 slow population or both.

119 Second, once the model is selected, the theorist defines the emergent property, the measurable
 120 signal of scientific interest. To continue our running STG example, one such emergent property

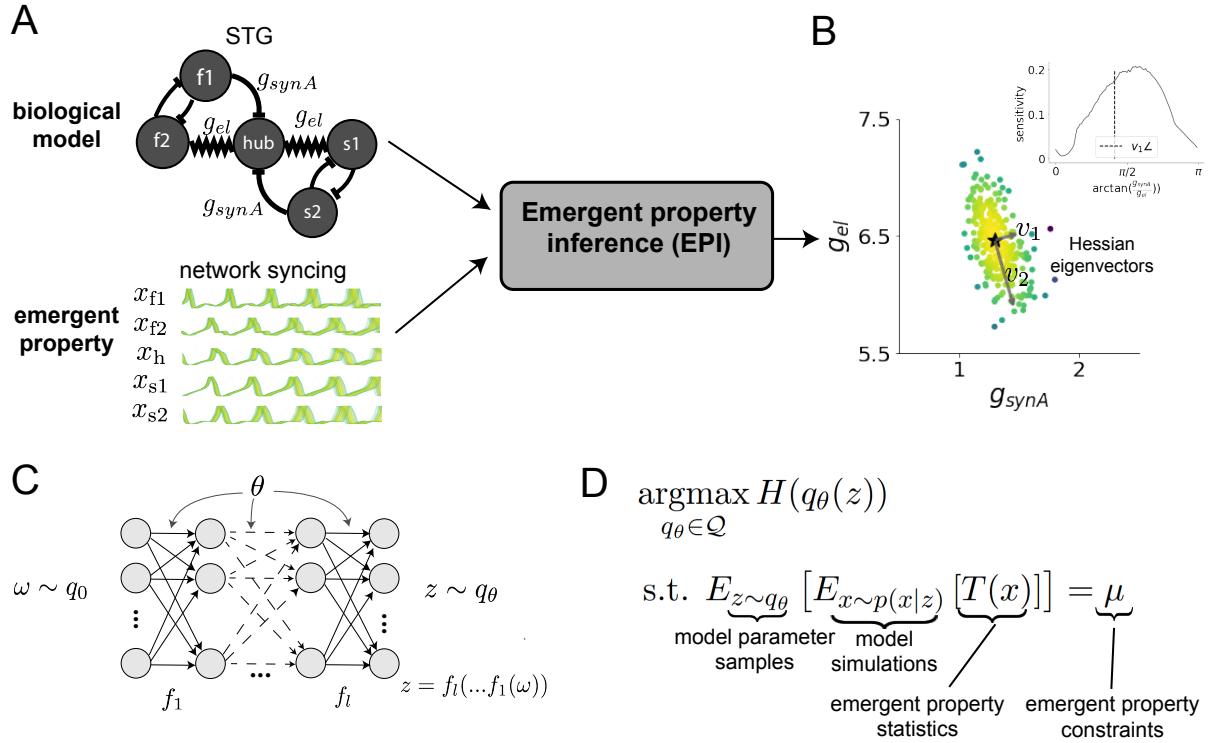


Figure 1: Emergent property inference (EPI) in the stomatogastric ganglion. A. For a choice of model (STG) and emergent property (network syncing), emergent property inference (EPI) learns a posterior distribution of the model parameters $z = [g_{\text{el}}, g_{\text{synA}}]^T$ conditioned on network 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 μ . EPI distributions maximize randomness via entropy, although other measures are sensible.

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. 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 [20]. 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:

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

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

¹⁴⁹ and $z = [g_{\text{el}}, g_{\text{synA}}]$ (see Section A.2.1).

¹⁵⁰ Second, we define the emergent property, which as above is network syncing: the phase locking of
¹⁵¹ the population and its oscillation at an intermediate frequency of our choosing (Figure 1A bottom).

¹⁵² Quantifying this phenomenon is straightforward: we define network syncing to be that each neuron’s
¹⁵³ spiking frequency – denoted $\omega_{\text{f1}}(x), \omega_{\text{f2}}(x)$, etc. – is close to an intermediate frequency of 0.55Hz.

¹⁵⁴ Mathematically, we achieve this via constraints on the mean and variance of $\omega_i(x)$ for each neuron
¹⁵⁵ $i \in \{\text{f1}, \text{f2}, \text{hub}, \text{s1}, \text{s2}\}$, and thus:

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

¹⁵⁶ which completes the quantification of the emergent property.

¹⁵⁷ Third, we perform emergent property inference: we find a distribution over parameter configura-
¹⁵⁸ tions z , and insist that samples from this distribution produce the emergent property; in other
¹⁵⁹ words, they obey the constraints introduced in Equation 2. This distribution will be chosen from
¹⁶⁰ a family of probability distributions $\mathcal{Q} = \{q_\theta(z) : \theta \in \Theta\}$, defined by a deep generative distribution
¹⁶¹ of the normalizing flow class [15, 16, 17] – neural networks which transform a simple distribution
¹⁶² into a suitably complicated distribution (as is needed here). This deep distribution is represented
¹⁶³ in Figure 1C (and see Methods for more detail). Then, mathematically, we must solve the following
¹⁶⁴ optimization program:

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

¹⁶⁵ where $T(x), \mu$ are defined as in Equation 3, and $p(x|z)$ is the intractable distribution of data from
¹⁶⁶ the model (x) , given that model’s parameters z (we access samples from this distribution by running
¹⁶⁷ the model forward). The purpose of each element in this program is detailed in Figure 1D. Finally,
¹⁶⁸ we recognize that many distributions in \mathcal{Q} will respect the emergent property constraints, so we
¹⁶⁹ require a normative principle to select amongst them. This principle is captured in Equation 3 by
¹⁷⁰ the primal objective H . Here we chose Shannon entropy as a means to find parameter distributions
¹⁷¹ with minimal assumptions beyond some chosen structure [28, 29, 18, 30], but we emphasize that
¹⁷² the EPI method is unaffected by this choice (but the results of course will depend on the primal
¹⁷³ objective chosen).

¹⁷⁴ EPI optimizes the weights and biases θ of the deep neural network (which induces the probability
¹⁷⁵ distribution) by iteratively solving Equation 3. The optimization is complete when the sampled
¹⁷⁶ models with parameters $z \sim q_\theta$ produce activity consistent with the specified emergent property.
¹⁷⁷ Such convergence is evaluated with a hypothesis test that the mean of each emergent property
¹⁷⁸ statistic is not different than its emergent property value (see Section A.1.2). Equipped with this
¹⁷⁹ method, we now prove out the value of EPI by using it to investigate three prominent models in
¹⁸⁰ neuroscience, using EPI to produce new insights about these models.

¹⁸¹ 3.3 Comprehensive input-responsivity in a nonlinear sensory system

¹⁸² In studies of primary visual cortex (V1), theoretical models with excitatory (E) and inhibitory
¹⁸³ (I) populations have reproduced a host of experimentally documented phenomena. In particular
¹⁸⁴ regimes of excitation and inhibition, these E/I models exhibit the paradoxical effect [4], selective
¹⁸⁵ amplification [31], surround suppression [32], and sensory integrative properties [33]. Extending
¹⁸⁶ this model using experimental evidence of three genetically-defined classes of inhibitory neurons
¹⁸⁷ [34, 35], recent work [21] has investigated a four-population model – excitatory (E), parvalbumin
¹⁸⁸ (P), somatostatin (S), and vasointestinal peptide (V) neurons – as shown in Fig. 2A. The dynamical
¹⁸⁹ state of this model is the firing rate of each neuron-type population $x = [x_E, x_P, x_S, x_V]^\top$, which
¹⁹⁰ evolves according to rectified and exponentiated dynamics:

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

¹⁹¹ with effective connectivity weights W and input h . In our analysis, we set the time constant
¹⁹² $\tau = 20\text{ms}$ and dynamics coefficient $n = 2$. Also, as is fairly standard, we obtain an informative
¹⁹³ estimate of the effective connectivities between these neuron-types W in mice by multiplying their
¹⁹⁴ probability of connection with their average synaptic strength [?] (see Section A.2.2). Given these
¹⁹⁵ fixed choices of W , n , and τ , we studied the system’s response to input

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

¹⁹⁶ where the input h is comprised of a baseline input $b = [b_E, b_P, b_S, b_V]^\top$ and a differential input
¹⁹⁷ $dh = [dh_E, dh_P, dh_S, dh_V]^\top$ to each neuron-type population. Throughout subsequent analyses, the
¹⁹⁸ baseline input is $b = [1, 1, 1, 1]^\top$.

¹⁹⁹ Having established our model, we now define the emergent property. We begin with the linearized
²⁰⁰ response of the system $\frac{dx_{ss}}{dh}$ at a fixed point x_{ss} . While this linearization accurately predicts differ-
²⁰¹ ential responses $dx_{ss} = [dx_{E,ss}, dx_{P,ss}, dx_{S,ss}, dx_{V,ss}]$ for small differential inputs to each population

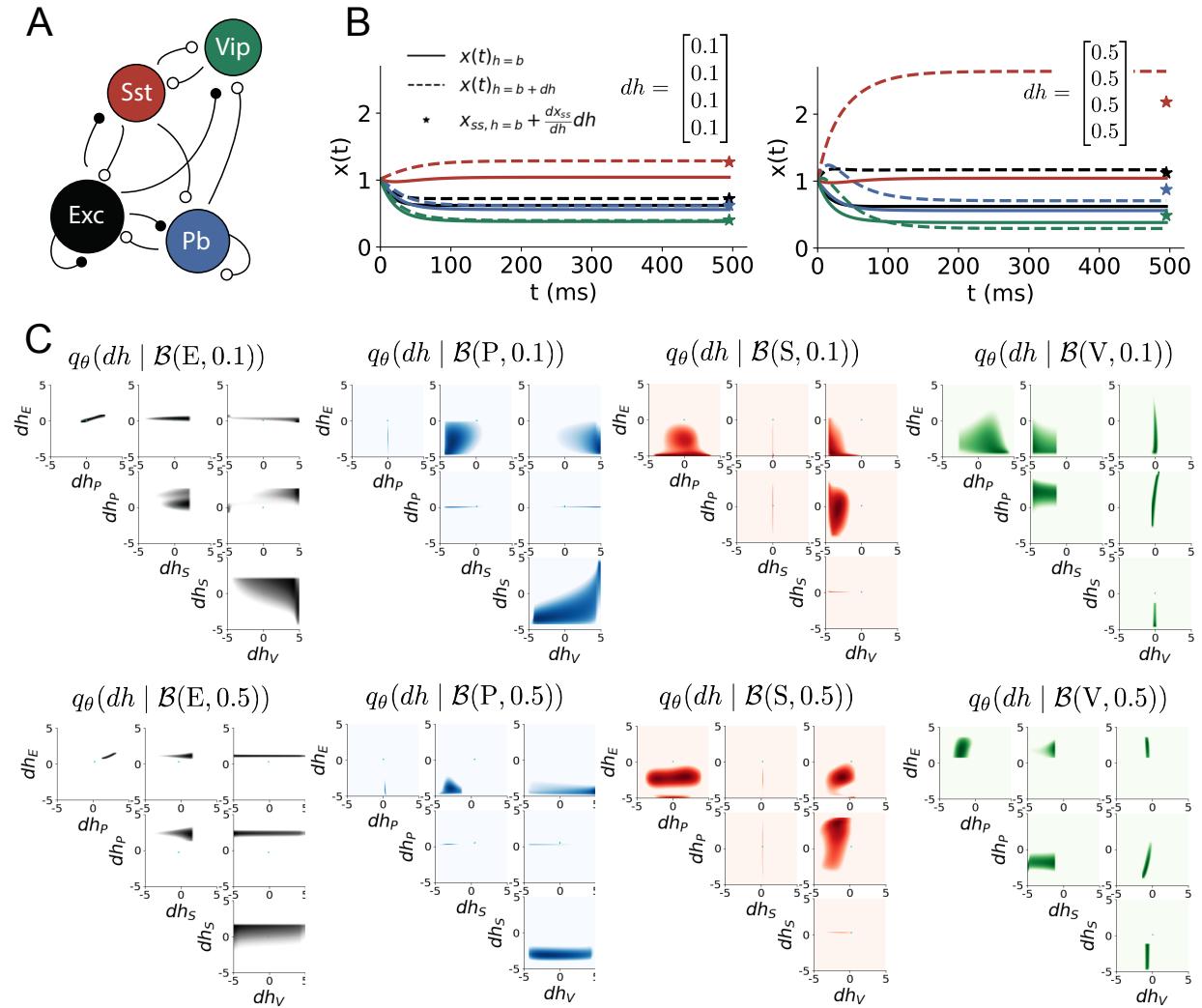


Figure 2: Exploring neuron-type responsivity in V1. A. Four-population model of primary visual cortex with excitatory (black), parvalbumin (blue), somatostatin (red), and vip (green) neurons. Some neuron-types largely do not form synaptic projections to others (excitatory and inhibitory projections filled and unfilled, respectively). B. Linear response predictions become inaccurate with greater input strength. V1 model simulations for input (solid) $h = b$ and (dashed) $h = b + dh$ with $b = [1, 1, 1, 1]^\top$ and (left) $dh = [0.1, 0.1, 0.1, 0.1]^\top$ (right) $dh = [0.5, 0.5, 0.5, 0.5]^\top$. Stars indicate the linear response prediction. C. EPI distributions on differential input dh conditioned on differential response $\mathcal{B}(\alpha, y)$ (see text). The linear prediction from two standard deviations away from y (from negative to positive) is overlaid in cyan (very small, near origin).

202 $dh = [0.1, 0.1, 0.1, 0.1]$ (Fig. 2B, left), linearization is a poor predictor in this nonlinear model more
203 generally (Fig. 3B, right). Currently available approaches to deriving the steady state response of
204 this system are limited.

205 To get a more comprehensive picture of the input-responsivity of each neuron-type, we used EPI
206 to learn a distribution of differential inputs dh that cause the rate of each neuron-type population
207 $\alpha \in \{E, P, S, V\}$ to increase by a value $y \in 0.1, 0.5$. These statements amount to the emergent
208 property

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

209 Note that we restrict the variance of the emergent property statistic $dx_{\alpha,ss}$ by constraining its
210 variance to a small value. In Fig. 2C, each column visualizes the inferred distribution of dh
211 corresponding to a specific neuron-type increase, while each row corresponds to amounts of increase
212 0.1 and 0.5. For visualization of this four-dimensional distribution, we show the two-dimensional
213 marginal densities. The inferred distributions immediately suggest four hypotheses:

- 214 1. as is intuitive, each neuron-type's firing rate should be sensitive to that neuron-type's direct
215 input;
- 216 2. the E- and P-populations should be largely unaffected by dh_V ;
- 217 3. the S-population should be largely unaffected by dh_P ;
- 218 4. EPI indicated that negative dh_E should result in small $dx_{V,ss}$, but positive dh_E should elicit
219 a larger $dx_{V,ss}$; that is, there should be a nonmonotonic response of $dx_{V,ss}$ with dh_E .

220 We evaluate these hypotheses by taking steps in individual neuron-type input Δh_α away from the
221 modes of the inferred distributions

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

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

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

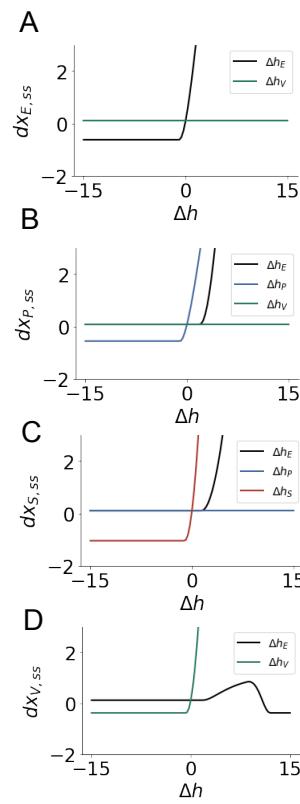


Figure 3: Confirming EPI generated hypotheses in V1. A. Differential responses by the E-population to changes in individual input $\Delta h_\alpha u_\alpha$ away from the mode of the EPI distribution dh^* . B-D Same plots for the P-, S-, and V-populations for the inputs for which hypotheses were formulated.

- the S-population is not affected by $d h_P$ (Fig. 3C blue);
- the V-population has a nonmonotonic response to $d h_E$ (Fig. 3D black).

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

3.4 Identifying neural mechanisms of behavioral learning.

Identifying measurable biological changes that result in improved behavior is important for neuroscience, since they may indicate how the learning brain adapts. In a rapid task switching experiment [36], where rats were to respond right (R) or left (L) to the side of a light stimulus in the pro (P) task, and oppositely in the anti (A) task predicated by an auditory cue (Fig. 3A), neural recordings exhibited two population of neurons in each hemisphere of superior colliculus (SC) that simultaneously represented both task condition and motor response: the Pro/contralateral and

241 Anti/ipsilateral neurons [22]. Duan et al. proposed a model of SC that, like the V1 model analyzed
 242 in the previous section, is a four-population dynamical system. Here, the neuron-type populations
 243 are functionally-defined as the Pro- and Anti-populations in each hemisphere (left (L) and right
 244 (R)). The Pro- or Anti-populations receive an input determined by the cue, and then the left and
 245 right populations receive an input based on the side of the light stimulus. Activities were bounded
 246 between 0 and 1, so that a high output of the Pro population in a given hemisphere corresponds
 247 to the contralateral response. An additional stipulation is that when one Pro population responds
 248 with a high-output, the opposite Pro population must respond with a low output. Finally, this
 249 circuit operates in the presence of gaussian noise resulting in trial-to-trial variability (see Section
 250 A.2.3). The connectivity matrix is parameterized by the geometry of the population arrangement
 251 (Fig. 3B).

252 Here, we used EPI to learn distributions of the SC weight matrix parameters $z = W$ conditioned
 253 on of various levels of rapid task switching accuracy $\mathcal{B}(p)$ for $p \in \{50\%, 60\%, 70\%, 80\%, 90\%\}$ (see
 254 Section A.2.3). As is standard, we decomposed the connectivity matrix $W = QAQ^{-1}$ in such a
 255 way (the Schur decomposition) that the basis vectors q_i are the same for all W (Fig. 3C). These
 256 basis vectors have intuitive roles in processing for this task, and are accordingly named the *all*
 257 mode - all neurons co-fluctuate, *side* mode - one side dominates the other, *task* mode - the Pro
 258 or Anti populations dominate the other, and *diag* mode - Pro- and Anti-populations of opposite
 259 hemispheres dominate the opposite pair. The corresponding eigenvalues (e.g. a_{task} , which change
 260 according to W) indicate the degree to which activity along that mode is increased or decreased
 261 by W .

262 EPI demonstrates that, for greater task accuracies, the task mode eigenvalue increases, indicating
 263 the importance of W to the task representation (Fig. 4D, purple). Stepping from random chance
 264 (50%) networks to marginally task-performing (60%) networks, there is a marked decrease of the
 265 side mode eigenvalues (Fig. 3D, orange). Such side mode suppression remains in the models
 266 achieving greater accuracy, revealing its importance towards task performance. There were no
 267 interesting trends with learning in the all or diag mode (hence not shown in Fig. 3). Importantly,
 268 we can conclude from our methodology that side mode suppression in W allows rapid task switching,
 269 and that greater task-mode representations in W increase accuracy. These hypotheses are confirmed
 270 by forward simulation of the SC model (Fig. 3E). Thus, EPI produces novel, experimentally testable
 271 predictions: effective connectivity between these populations changes throughout learning, in a way
 272 that increases its task mode and decreases its side mode eigenvalues.

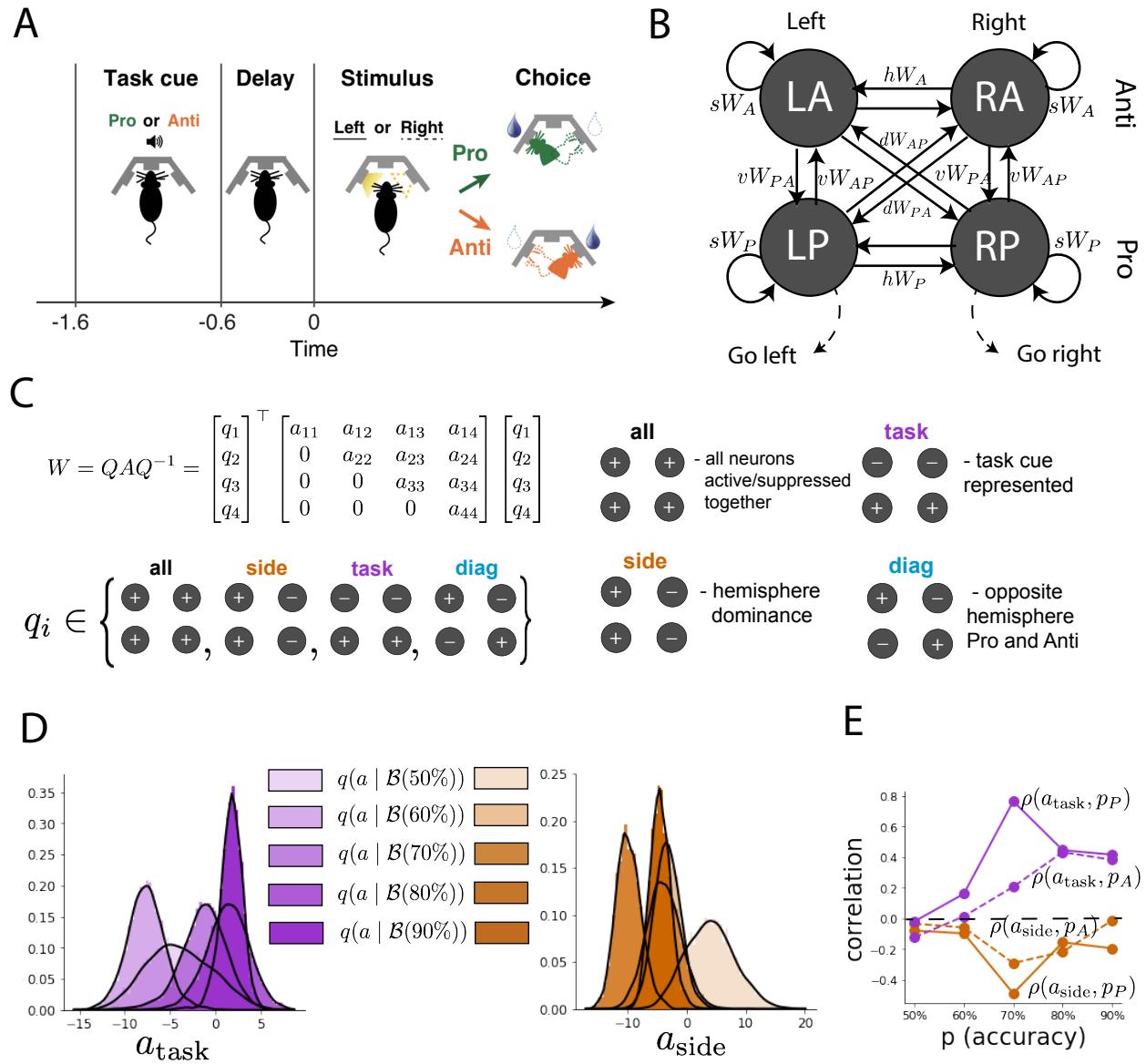


Figure 4: EPI reveals changes in SC [22] connectivity that result in greater task accuracy. A. Rapid task switching behavioral paradigm. In the Pro (Anti) condition indicated by an auditory cue, rats respond by poking into a side port to the same (opposite) side as the light stimulus that is provided after a delay to receive a reward. 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 . The invariant Schur modes are labeled by their hypothesized role in computation: q_{all} , q_{task} , q_{side} , and q_{diag} . The values of A are what change for different realizations of W . 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.

²⁷³ **3.5 Characterizing the sources of bias in RNN computation**

²⁷⁴ Each model we thus far have studied was designed from fundamental biophysical principles, genetically-
²⁷⁵ or functionally-defined neuron types. At a more abstract level of modeling, recurrent neural net-
²⁷⁶ works (RNNs) are high-dimensional models of computation and have become increasingly popular
²⁷⁷ in neuroscience research [37]. Typically, RNNs are trained to do a task from a systems neuro-
²⁷⁸ science experiment, and then the unit activations of the trained RNN are compared to recorded
²⁷⁹ neural activity. Here we run EPI on the connectivity matrices of RNNs trained to solve a sample
²⁸⁰ task.

²⁸¹ Recent work establishes a link between RNN connectivity weights and the resulting dynamical re-
²⁸² sponds of the network, using dynamic mean field theory (DMFT) [3]. Specifically, DMFT describes
²⁸³ the properties of activity in infinite-size neural networks given a distribution on the connectivity
²⁸⁴ weights. This theory has been extended from random neural networks to low-rank RNNs, which
²⁸⁵ have low-dimensional parameterizations of RNN connectivity via the pairwise correlations of the
²⁸⁶ low-rank vectors (i.e. the low-rank “geometry”) [23]. In such a model, the connectivity of a rank-1
²⁸⁷ RNN’s weight matrix J is the sum of a random component with strength determined by g and a
²⁸⁸ structured component determined by the outer product of vectors m and n :

$$J = g\chi + \frac{1}{N}mn^\top, \quad (8)$$

²⁸⁹ where the activity x evolves as

$$\frac{\partial x}{\partial t} = -x(t) + J\phi(x(t)) + I(t), \quad (9)$$

²⁹⁰ and $I(t)$ is some input, ϕ is the tanh nonlinearity, and $\chi_{ij} \sim \mathcal{N}(0, \frac{1}{N})$. The entries of m and n are
²⁹¹ drawn from gaussian distributions $m_i \sim \mathcal{N}(M_m, 1)$ and $n_i \sim \mathcal{N}(M_n, 1)$.

²⁹² Mastrogiovisepppe et al. designed low-rank connectivities via the pairwise correlations of the vectors
²⁹³ m, n in models that solve tasks from behavioral neuroscience. We can consider the DMFT equation
²⁹⁴ solver as a black box that takes in a low-rank parameterization z (e.g. $z = [g, M_m, M_n]$) and outputs
²⁹⁵ task-relevant response variables (e.g. average network activity μ , the temporal variability in the
²⁹⁶ network Δ_T , or network activity along a given dimension κ). Importantly, the solution produced
²⁹⁷ by the solver is differentiable with respect to the input parameters, allowing us to combine DMFT
²⁹⁸ with EPI to learn distributions on such connectivity parameters of RNNs that execute tasks via an
²⁹⁹ emergent property defined on the task-relevant responses produced by DMFT.

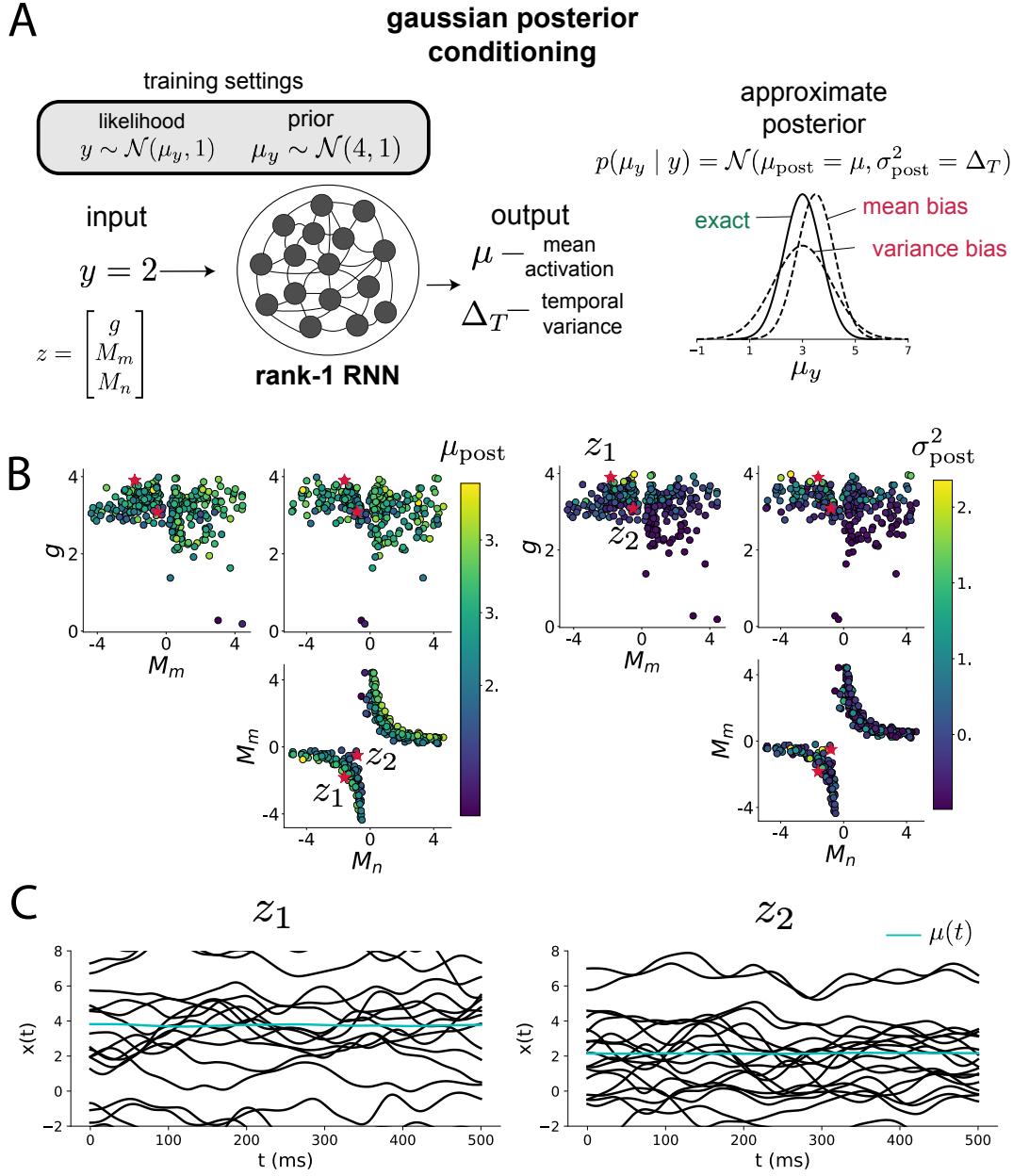


Figure 5: Sources of bias in RNN computation. A. (left) A rank-1 RNN running approximate Bayesian inference on μ_y assuming a gaussian likelihood variance of 1 and a prior of $\mathcal{N}(4, 1)$. (center) The rank-1 RNN represents the computed gaussian posterior mean μ_{post} and variance σ_{post}^2 in its mean activity μ and its temporal variance Δ_T . (right) Bias in this computation can come from over- or under-estimating the posterior mean or variance. B. Distribution of rank-1 RNNs executing approximate Bayesian inference. Samples are colored by (left) posterior mean $\mu_{\text{post}} = \mu$ and (right) posterior variance $\sigma_{\text{post}}^2 = \Delta_T$. C. Finite size realizations agree with the DMFT theory.

300 We train the network to solve gaussian posterior conditioning: calculate the parameters of a gaus-
 301 sian posterior distribution on the mean of a gaussian likelihood μ_y , given a single observation of
 302 $y \sim \mathcal{N}(\mu_y, 1)$ and a prior $p(\mu_y) = \mathcal{N}(4, 1)$ (Fig. 5A). The true posterior for an input of $y = 2$
 303 is $p(\mu_y | y) = \mathcal{N}(3, 0.5)$. We used EPI to learn distributions of RNNs producing the correct
 304 posterior mean and variance in their mean activity $\mu = \mu_{\text{post}}$ and temporal variance $\Delta_T = \sigma_{\text{post}}^2$
 305 (respectively), given an input of $y = 2$. (see Section A.2.4) (Fig. 5B).

306 When specifying the emergent property of gaussian posterior conditioning, we allowed a substantial
 307 amount of variability in the second moment constraints of the network mean μ and temporal
 308 variance Δ_T . This resulted in a distribution of rank-1 RNN parameterizations having a wide
 309 variety of biases in the resulting μ_{post} and σ_{post}^2 (under- or over-estimates of the posterior means
 310 and variances). We can examine the nature of the biases in this computation by visualizing the
 311 produced posterior means (Fig. 5B, left) and variances (Fig. 5B, right) in the EPI distribution.
 312 The inferred distribution has rough symmetry in the M_m - M_n plane, suggesting a degeneracy in the
 313 product of M_m and M_n (Fig. 5B). The product of M_m and M_n almost completely determines the
 314 posterior mean (Fig. 5B, left), and the random strength g is the most influential variable on the
 315 temporal variance (Fig. 5B, right). Neither of these observations were obvious from the consistency
 316 equations afforded by DMFT (see Section A.2.4).

317 When working with DMFT, it's important to check that finite-size realizations of these infinite-
 318 size networks match the theoretical predictions. We check 2,000-neuron realizations of drawn
 319 parameters z_1 and z_2 from the inferred distribution. z_1 has relatively high g and high $M_m M_n$,
 320 whereas z_2 has relatively low g and low $M_m M_n$. Confirming our intuition, z_1 overestimates the
 321 posterior mean, since mean activity $\mu(t) > 3$ (Fig. 5C, left cyan). In turn, z_2 underestimates the
 322 posterior mean, since $\mu(t) < 3$ (Fig. 5C, right cyan). Finally, z_1 results in evidently greater temporal
 323 variance than z_2 . This novel procedure of doing inference in interpretable parameterizations of
 324 RNNs conditioned on task execution is straightforwardly generalizable to other tasks like noisy
 325 integration and context-dependent decision making (Fig. S1).

326 4 Discussion

327 4.1 EPI is a general tool for theoretical neuroscience

328 Models of biological systems are often comprised of complex nonlinear differential equations, mak-
 329 ing 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 produced state-of-the-art performance on high-dimensional tasks such as texture generation [18]. Of course, increasing the dimensionality of the dynamical state of the model makes optimization more expensive, and there is a practical limit there as with any machine learning approach. For systems with high dimensional state, we recommend using theoretical approaches (e.g. [23]) to reason about reduced parameterizations of such high-dimensional systems.

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

4.2 Novel hypotheses from EPI

Machine learning has played an effective, multifaceted role in neuroscientific progress. Primarily, it has revealed structure in large-scale neural datasets [38, 39, 40, 41, 42, 43] (see review, [14]).

361 Secondly, trained algorithms of varying degrees of biological relevance are beginning to be viewed
362 as fully-observable computational systems comparable to the brain [44, 45].

363 For example, consider the fact that we do not fully understand the four-dimensional models of V1
364 [21]. Because analytical approaches to studying nonlinear dynamical systems become increasingly
365 complicated when stepping from two-dimensional to three- or four-dimensional systems in the
366 absence of restrictive simplifying assumptions [46], it is unsurprising that this model has been a
367 challenge. In Section 3.3, we showed that EPI was far more informative about neuron-type input
368 responsibility than the predictions afforded through analysis. By flexibly conditioning this V1 model
369 on different emergent properties, we performed an exploratory analysis of a *model* rather than a
370 dataset, which generated and proved out a set of testable predictions.

371 Of course, exploratory analyses can also be directed. For example, when interested in model
372 changes during learning, one can use EPI to condition as we did in Section 3.4. This analysis
373 identified experimentally testable predictions (proved out *in-silico*) of changes in connectivity in
374 SC throughout learning. Precisely, we predict that an initial reduction in side mode eigenvalue,
375 and a steady increase in task mode eigenvalue will take place, during learning, in the effective
376 connectivity matrices of learning rats.

377 In our final analysis, we present a novel procedure for doing statistical inference on interpretable
378 parameterizations of RNNs executing simple tasks . This methodology relies on recently extended
379 theory of responses in random neural networks with minimal structure [23]. With this methodology,
380 we can finally open the probabilistic model selection toolkit reasoning about the connectivity of
381 RNNs solving tasks.

382 References

- 383 [1] Larry F Abbott. Theoretical neuroscience rising. *Neuron*, 60(3):489–495, 2008.
- 384 [2] John J Hopfield. Neurons with graded response have collective computational properties like
385 those of two-state neurons. *Proceedings of the national academy of sciences*, 81(10):3088–3092,
386 1984.
- 387 [3] Haim Sompolinsky, Andrea Crisanti, and Hans-Jurgen Sommers. Chaos in random neural
388 networks. *Physical review letters*, 61(3):259, 1988.

- [4] Misha V Tsodyks, William E Skaggs, Terrence J Sejnowski, and Bruce L McNaughton. Paradoxical effects of external modulation of inhibitory interneurons. *Journal of neuroscience*, 17(11):4382–4388, 1997.
- [5] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *International Conference on Learning Representations*, 2014.
- [6] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and variational inference in deep latent gaussian models. *International Conference on Machine Learning*, 2014.
- [7] Yuanjun Gao, Evan W Archer, Liam Paninski, and John P Cunningham. Linear dynamical neural population models through nonlinear embeddings. In *Advances in neural information processing systems*, pages 163–171, 2016.
- [8] Yuan Zhao and Il Memming Park. Recursive variational bayesian dual estimation for nonlinear dynamics and non-gaussian observations. *stat*, 1050:27, 2017.
- [9] Gabriel Barello, Adam Charles, and Jonathan Pillow. Sparse-coding variational auto-encoders. *bioRxiv*, page 399246, 2018.
- [10] Chethan Pandarinath, Daniel J O’Shea, Jasmine Collins, Rafal Jozefowicz, Sergey D Stavisky, Jonathan C Kao, Eric M Trautmann, Matthew T Kaufman, Stephen I Ryu, Leigh R Hochberg, et al. Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature methods*, page 1, 2018.
- [11] Alexander B Wiltschko, Matthew J Johnson, Giuliano Iurilli, Ralph E Peterson, Jesse M Katon, Stan L Pashkovski, Victoria E Abraira, Ryan P Adams, and Sandeep Robert Datta. Mapping sub-second structure in mouse behavior. *Neuron*, 88(6):1121–1135, 2015.
- [12] Matthew J Johnson, David K Duvenaud, Alex Wiltschko, Ryan P Adams, and Sandeep R Datta. Composing graphical models with neural networks for structured representations and fast inference. In *Advances in neural information processing systems*, pages 2946–2954, 2016.
- [13] Eleanor Batty, Matthew Whiteway, Shreya Saxena, Dan Biderman, Taiga Abe, Simon Musall, Winthrop Gillis, Jeffrey Markowitz, Anne Churchland, John Cunningham, et al. Behavenet: nonlinear embedding and bayesian neural decoding of behavioral videos. *Advances in Neural Information Processing Systems*, 2019.

- 418 [14] Liam Paninski and John P Cunningham. Neural data science: accelerating the experiment-
419 analysis-theory cycle in large-scale neuroscience. *Current opinion in neurobiology*, 50:232–241,
420 2018.
- 421 [15] Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows.
422 *International Conference on Machine Learning*, 2015.
- 423 [16] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp.
424 *arXiv preprint arXiv:1605.08803*, 2016.
- 425 [17] George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density
426 estimation. In *Advances in Neural Information Processing Systems*, pages 2338–2347, 2017.
- 427 [18] Gabriel Loaiza-Ganem, Yuanjun Gao, and John P Cunningham. Maximum entropy flow
428 networks. *International Conference on Learning Representations*, 2017.
- 429 [19] Dustin Tran, Rajesh Ranganath, and David Blei. Hierarchical implicit models and likelihood-
430 free variational inference. In *Advances in Neural Information Processing Systems*, pages 5523–
431 5533, 2017.
- 432 [20] Gabrielle J Gutierrez, Timothy O’Leary, and Eve Marder. Multiple mechanisms switch an
433 electrically coupled, synaptically inhibited neuron between competing rhythmic oscillators.
434 *Neuron*, 77(5):845–858, 2013.
- 435 [21] Ashok Litwin-Kumar, Robert Rosenbaum, and Brent Doiron. Inhibitory stabilization and vi-
436 sual coding in cortical circuits with multiple interneuron subtypes. *Journal of neurophysiology*,
437 115(3):1399–1409, 2016.
- 438 [22] Chunyu A Duan, Marino Pagan, Alex T Piet, Charles D Kopec, Athena Akrami, Alexander J
439 Riordan, Jeffrey C Erlich, and Carlos D Brody. Collicular circuits for flexible sensorimotor
440 routing. *bioRxiv*, page 245613, 2018.
- 441 [23] Francesca Mastrogiovanni and Srdjan Ostojic. Linking connectivity, dynamics, and computa-
442 tions in low-rank recurrent neural networks. *Neuron*, 99(3):609–623, 2018.
- 443 [24] Sean R Bittner, Agostina Palmigiano, Kenneth D Miller, and John P Cunningham. Degener-
444 ate solution networks for theoretical neuroscience. *Computational and Systems Neuroscience
445 Meeting (COSYNE), Lisbon, Portugal*, 2019.

- 446 [25] Sean R Bittner, Alex T Piet, Chunyu A Duan, Agostina Palmigiano, Kenneth D Miller,
447 Carlos D Brody, and John P Cunningham. Examining models in theoretical neuroscience with
448 degenerate solution networks. *Bernstein Conference*, 2019.
- 449 [26] Jan-Matthis Lueckmann, Pedro Goncalves, Chaitanya Chintaluri, William F Podlaski, Gia-
450 como Bassetto, Tim P Vogels, and Jakob H Macke. Amortised inference for mechanistic models
451 of neural dynamics. In *Computational and Systems Neuroscience Meeting (COSYNE), Lisbon,*
452 *Portugal*, 2019.
- 453 [27] Eve Marder and Vatsala Thirumalai. Cellular, synaptic and network effects of neuromodula-
454 tion. *Neural Networks*, 15(4-6):479–493, 2002.
- 455 [28] Edwin T Jaynes. Information theory and statistical mechanics. *Physical review*, 106(4):620,
456 1957.
- 457 [29] Gamaleldin F Elsayed and John P Cunningham. Structure in neural population recordings:
458 an expected byproduct of simpler phenomena? *Nature neuroscience*, 20(9):1310, 2017.
- 459 [30] Cristina Savin and Gašper Tkačik. Maximum entropy models as a tool for building precise
460 neural controls. *Current opinion in neurobiology*, 46:120–126, 2017.
- 461 [31] Brendan K Murphy and Kenneth D Miller. Balanced amplification: a new mechanism of
462 selective amplification of neural activity patterns. *Neuron*, 61(4):635–648, 2009.
- 463 [32] Hirofumi Ozeki, Ian M Finn, Evan S Schaffer, Kenneth D Miller, and David Ferster. Inhibitory
464 stabilization of the cortical network underlies visual surround suppression. *Neuron*, 62(4):578–
465 592, 2009.
- 466 [33] Daniel B Rubin, Stephen D Van Hooser, and Kenneth D Miller. The stabilized supralinear
467 network: a unifying circuit motif underlying multi-input integration in sensory cortex. *Neuron*,
468 85(2):402–417, 2015.
- 469 [34] Henry Markram, Maria Toledo-Rodriguez, Yun Wang, Anirudh Gupta, Gilad Silberberg, and
470 Caizhi Wu. Interneurons of the neocortical inhibitory system. *Nature reviews neuroscience*,
471 5(10):793, 2004.
- 472 [35] Bernardo Rudy, Gordon Fishell, SooHyun Lee, and Jens Hjerling-Leffler. Three groups of
473 interneurons account for nearly 100% of neocortical gabaergic neurons. *Developmental neuro-
474 biology*, 71(1):45–61, 2011.

- 475 [36] Chunyu A Duan, Jeffrey C Erlich, and Carlos D Brody. Requirement of prefrontal and midbrain
476 regions for rapid executive control of behavior in the rat. *Neuron*, 86(6):1491–1503, 2015.
- 477 [37] Omri Barak. Recurrent neural networks as versatile tools of neuroscience research. *Current*
478 *opinion in neurobiology*, 46:1–6, 2017.
- 479 [38] Robert E Kass and Valérie Ventura. A spike-train probability model. *Neural computation*,
480 13(8):1713–1720, 2001.
- 481 [39] Emery N Brown, Loren M Frank, Dengda Tang, Michael C Quirk, and Matthew A Wilson.
482 A statistical paradigm for neural spike train decoding applied to position prediction from
483 ensemble firing patterns of rat hippocampal place cells. *Journal of Neuroscience*, 18(18):7411–
484 7425, 1998.
- 485 [40] Liam Paninski. Maximum likelihood estimation of cascade point-process neural encoding
486 models. *Network: Computation in Neural Systems*, 15(4):243–262, 2004.
- 487 [41] M Yu Byron, John P Cunningham, Gopal Santhanam, Stephen I Ryu, Krishna V Shenoy, and
488 Maneesh Sahani. Gaussian-process factor analysis for low-dimensional single-trial analysis
489 of neural population activity. In *Advances in neural information processing systems*, pages
490 1881–1888, 2009.
- 491 [42] Kenneth W Latimer, Jacob L Yates, Miriam LR Meister, Alexander C Huk, and Jonathan W
492 Pillow. Single-trial spike trains in parietal cortex reveal discrete steps during decision-making.
493 *Science*, 349(6244):184–187, 2015.
- 494 [43] Lea Duncker, Gergo Bohner, Julien Boussard, and Maneesh Sahani. Learning interpretable
495 continuous-time models of latent stochastic dynamical systems. *Proceedings of the 36th Inter-*
496 *national Conference on Machine Learning*, 2019.
- 497 [44] David Sussillo and Omri Barak. Opening the black box: low-dimensional dynamics in high-
498 dimensional recurrent neural networks. *Neural computation*, 25(3):626–649, 2013.
- 499 [45] Blake A Richards and et al. A deep learning framework for neuroscience. *Nature Neuroscience*,
500 2019.
- 501 [46] Steven H Strogatz. Nonlinear dynamics and chaos: with applications to physics, *Biology,*
502 *Chemistry, and Engineering (Studies in Nonlinearity)*, Perseus, Cambridge, UK, 1994.

- 503 [47] Rajesh Ranganath, Sean Gerrish, and David Blei. Black box variational inference. In *Artificial
504 Intelligence and Statistics*, pages 814–822, 2014.
- 505 [48] Martin J Wainwright, Michael I Jordan, et al. Graphical models, exponential families, and
506 variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.
- 507 [49] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp.
508 *Proceedings of the 5th International Conference on Learning Representations*, 2017.
- 509 [50] Carsten K Pfeffer, Mingshan Xue, Miao He, Z Josh Huang, and Massimo Scanziani. Inhi-
510 bition of inhibition in visual cortex: the logic of connections between molecularly distinct
511 interneurons. *Nature neuroscience*, 16(8):1068, 2013.

512 **A Methods**

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

514 Emergent property inference (EPI) learns distributions of theoretical model parameters that pro-
515 duce emergent properties of interest. EPI combines ideas from likelihood-free variational inference
516 [19] and maximum entropy flow networks [18]. A maximum entropy flow network is used as a deep
517 probability distribution for the parameters, while these samples often parameterize a differentiable
518 model simulator, which may lack a tractable likelihood function.

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

$$\mathcal{B} \leftrightarrow E_{z \sim q_\theta} [E_{x \sim p(x|z)} [T(x)]] = \mu \quad (10)$$

524 by making an approximation $q_\theta(z)$ to $p(z | \mathcal{B})$ (see Section A.1.5). So, over the DSN distribution
525 $q_\theta(z)$ of model $p(x | z)$ for behavior \mathcal{B} , the emergent properties $T(x)$ are constrained in expectation
526 to μ .

527 In deep probability distributions, a simple random variable $w \sim p_0$ is mapped deterministically
528 via a function f_θ parameterized by a neural network to the support of the distribution of interest

529 where $z = f_\theta(\omega) = f_l(\dots f_1(\omega))$. Given a theoretical model $p(x | z)$ and some behavior of interest
 530 \mathcal{B} , the deep probability distributions are trained by optimizing the neural network parameters θ to
 531 find a good approximation q_θ^* within the deep variational family Q to $p(z | \mathcal{B})$.

532 In most settings (especially those relevant to theoretical neuroscience) the likelihood of the behavior
 533 with respect to the model parameters $p(T(x) | z)$ is unknown or intractable, requiring an alternative
 534 to stochastic gradient variational Bayes [5] or black box variational inference[47]. These types
 535 of methods called likelihood-free variational inference (LFVI, [19]) skate around the intractable
 536 likelihood function in situations where there is a differentiable simulator. Akin to LFVI, DSNs are
 537 optimized with the following objective for a given theoretical model, emergent property statistics
 538 $T(x)$, and emergent property constraints μ :

$$\begin{aligned} q_\theta^*(z) &= \underset{q_\theta \in Q}{\operatorname{argmax}} H(q_\theta(z)) \\ \text{s.t. } E_{z \sim q_\theta} [E_{x \sim p(x|z)} [T(x)]] &= \mu \end{aligned} \tag{11}$$

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

542 A.1.1 Example: 2D LDS

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

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

544 that produce a band of oscillations. To do EPI with the dynamics matrix elements as the free
 545 parameters $z = [a_1, a_2, a_3, a_4]$, and fixing $\tau = 1$, such that the posterior yields a band of oscillations,
 546 the emergent property statistics $T(x)$ are chosen to contain the first- and second-moments of the
 547 oscillatory frequency Ω and the growth/decay factor d of the oscillating system. To learn the
 548 distribution of real entries of A that yield a distribution of d with mean zero with variance 0.25^2 ,
 549 and oscillation frequency Ω with mean 1 Hz with variance $(0.1\text{Hz})^2$, then we would select the real
 550 part of the complex conjugate eigenvalues $\text{real}(\lambda_1) = d$ (via an arbitrary choice of eigenvalue of the
 551 dynamics matrix λ_1) and the positive imaginary component of one of the eigenvalues $\text{imag}(\lambda_1) =$
 552 $2\pi\Omega$ as the emergent property statistics. Those emergent property statistics are then constrained

553 to

$$\mu = E \begin{bmatrix} \text{real}(\lambda_1) \\ \text{imag}(\lambda_1) \\ (\text{real}(\lambda_1) - 0)^2 \\ (\text{imag}(\lambda_1) - 2\pi\Omega)^2 \end{bmatrix} = \begin{bmatrix} 0.0 \\ 2\pi\Omega \\ 0.25^2 \\ (2\pi 0.1)^2 \end{bmatrix} \quad (12)$$

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

$$\lambda = \frac{\left(\frac{a_1+a_4}{\tau}\right) \pm \sqrt{\left(\frac{a_1+a_4}{\tau}\right)^2 + 4\left(\frac{a_2a_3-a_1a_4}{\tau}\right)}}{2} \quad (13)$$

557 where λ_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
 558 directly via a closed form function and does not require simulation, we cannot derive the distribution
 559 q_θ^* directly. This is due to the formally hard problem of the backward mapping: finding the natural
 560 parameters η from the mean parameters μ of an exponential family distribution [48]. Instead, we
 561 can use EPI to learn the linear system parameters producing such a band of oscillations (Fig. S2B).

562 Even this relatively simple system has nontrivial (though intuitively sensible) structure in the
 563 parameter distribution. To validate our method (further than that of the underlying technology
 564 on a ground truth solution [18]) we can analytically derive the contours of the probability density
 565 from the emergent property statistics and values (Fig. S3). In the $a_1 - a_4$ plane, is a black line
 566 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
 567 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
 568 set of solutions at fixed behaviors, which overlay the posterior obtained through EPI. The learned
 569 DSN distribution precisely reflects the desired statistical constraints and model degeneracy in the
 570 sum of a_1 and a_4 . Intuitively, the parameters equivalent with respect to emergent property statistic
 571 $\text{real}(\lambda_1)$ have similar log densities.

572 To explain the structure in the bimodality of the DSN posterior, we can look at the imaginary
 573 component of λ_1 . When $\text{real}(\lambda_1) = \frac{a_1+a_4}{2} = 0$, we have

$$\text{imag}(\lambda_1) = \begin{cases} \sqrt{\frac{a_1a_4-a_2a_3}{\tau}}, & \text{if } a_1a_4 < a_2a_3 \\ 0 & \text{otherwise} \end{cases} \quad (14)$$

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

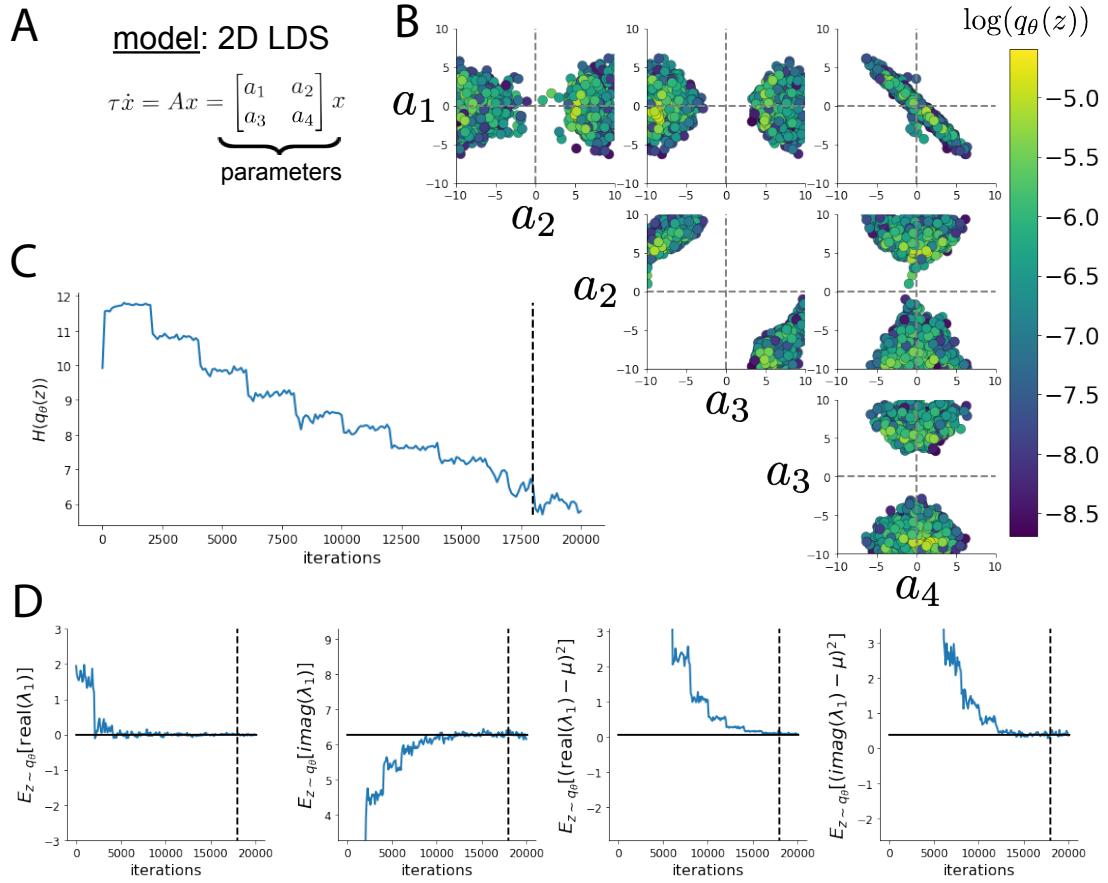


Fig. S2: A. Two-dimensional linear dynamical system model, where real entries of the dynamics matrix A are the parameters. B. The DSN distribution for a 2D LDS with $\tau = 1$ that produces an average of 1Hz oscillations with some small amount of variance. C. Entropy throughout the optimization. At the 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.

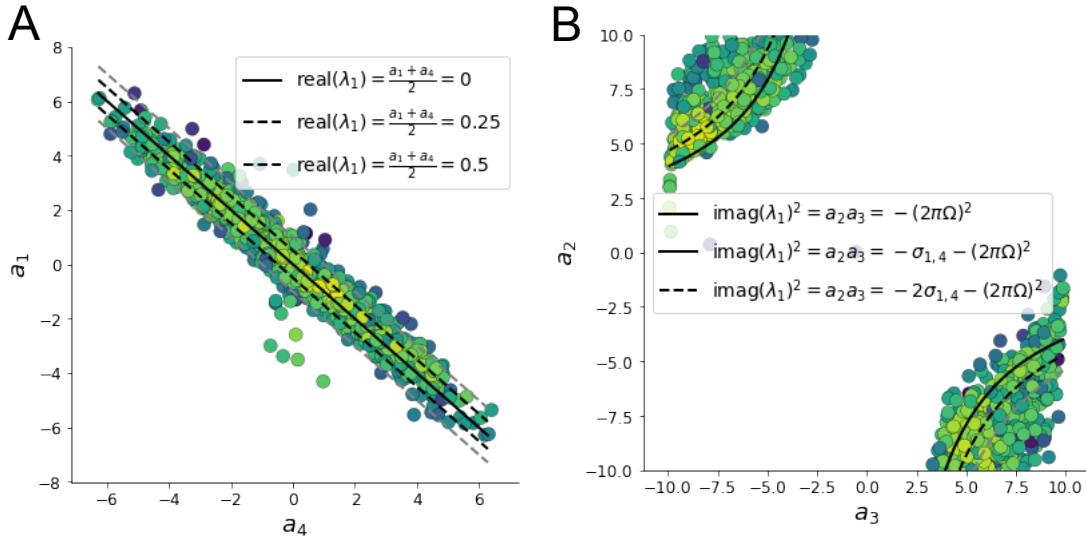


Fig. S3: A. Probability contours in the $a_1 - a_4$ plane can be derived from the relationship to emergent property statistic of growth/decay factor. B. Probability contours in the $a_2 - a_3$ plane can be derived from relationship to the emergent property statistic of oscillation frequency.

$$\text{imag}(\lambda_1)^2 = a_1 a_4 - a_2 a_3 \quad (15)$$

576 Since we constrained $E_{q_\theta} [\text{imag}(\lambda)] = 2\pi$ (with $\omega = 1$), we can plot contours of the equation
 577 $\text{imag}(\lambda_1)^2 = a_1 a_4 - a_2 a_3 = (2\pi)^2$ for various $a_1 a_4$ (Fig. S3A). If $\sigma_{1,4} = E_{q_\theta} (|a_1 a_4 - E_{q_\theta}[a_1 a_4]|)$,
 578 then we plot the contours as $a_1 a_4 = 0$ (black), $a_1 a_4 = -\sigma_{1,4}$ (black dotted), and $a_1 a_4 = -2\sigma_{1,4}$
 579 (grey dotted) (Fig. S3B). This validates the curved structure of the inferred distribution learned
 580 through EPI. We take steps in negative standard deviation of $a_1 a_4$ (dotted and gray lines), since
 581 there are few positive values $a_1 a_4$ in the posterior. Subtler model-behavior combinations will have
 582 even more complexity, further motivating the use of EPI for understanding these systems. Indeed,
 583 we sample a distribution of systems oscillating near 1Hz (Fig. S4).

584 **A.1.2 Augmented Lagrangian optimization**

585 To optimize $q_\theta(z)$ in equation 1, the constrained optimization is performed using the augmented
 586 Lagrangian method. The following objective is minimized:

$$L(\theta; \alpha, c) = -H(q_\theta) + \alpha^\top \delta(\theta) + \frac{c}{2} \|\delta(\theta)\|^2 \quad (16)$$

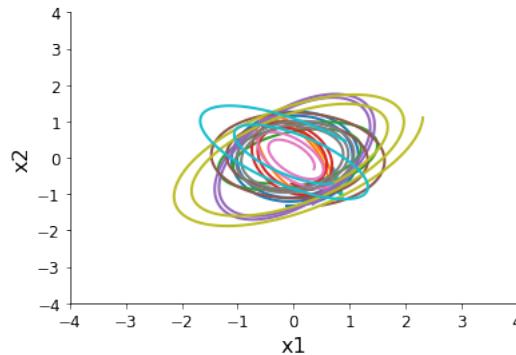


Fig. S4: Sampled dynamical system trajectories from the EPI distribution. Each trajectory is initialized at $x(0) = \begin{bmatrix} \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \end{bmatrix}$.

587 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
 588 coefficient. For a fixed (α, c) , θ is optimized with stochastic gradient descent. A low value of c is
 589 used initially, and increased during each augmented Lagrangian epoch – a period of optimization
 590 with fixed α and c for a given number of stochastic optimization iterations. Similarly, α is tuned
 591 each epoch based on the constraint violations. For the linear 2-dimensional system (Fig. S2C)
 592 optimization hyperparameters are initialized to $c_1 = 10^{-4}$ and $\alpha_1 = 0$. The penalty coefficient
 593 is updated based on a hypothesis test regarding the reduction in constraint violation. The p-
 594 value of $E[|\delta(\theta_{k+1})|] > \gamma E[|\delta(\theta_k)|]$ is computed, and c_{k+1} is updated to βc_k with probability
 595 $1 - p$. Throughout the project, $\beta = 4.0$ and $\gamma = 0.25$ is used. The other update rule is $\alpha_{k+1} =$
 596 $\alpha_k + c_k \frac{1}{n} \sum_{i=1}^n (T(x^{(i)}) - \mu)$. In this example, each augmented Lagrangian epoch ran for 2,000
 597 iterations. We consider the optimization to have converged when a null hypothesis test of constraint
 598 violations being zero is accepted for all constraints at a significance threshold 0.05. This is the dotted
 599 line on the plots below depicting the optimization cutoff of EPI optimization for the 2-dimensional
 600 linear system. If the optimization is left to continue running, entropy usually decreases, and
 601 structural pathologies in the distribution may be introduced.

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

608 **A.1.3 Normalizing flows**

609 Since we are optimizing parameters θ of our deep probability distribution with respect to the
 610 entropy, we will need to take gradients with respect to the log-density of samples from the deep
 611 probability distribution.

$$H(q_\theta(z)) = \int -q_\theta(z) \log(q_\theta(z)) dz = E_{z \sim q_\theta} [-\log(q_\theta(z))] = E_{\omega \sim q_0} [-\log(q_\theta(f_\theta(\omega)))] \quad (17)$$

$$612 \quad \nabla_\theta H(q_\theta(z)) = E_{\omega \sim q_0} [-\nabla_\theta \log(q_\theta(f_\theta(\omega)))] \quad (18)$$

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

$$q(z') = q(f^{-1}(z')) \left| \det \frac{\partial f^{-1}(z')}{\partial z'} \right| = q(z) \left| \det \frac{\partial f(z)}{\partial z} \right|^{-1} \quad (19)$$

616 However, this computation has cubic complexity in dimensionality for fully connected layers. By
 617 restricting our layers to normalizing flows [15] – bijective functions with fast log determinant ja-
 618 cobian computations, we can tractably optimize deep generative models with objectives that are a
 619 function of sample density, like entropy. Most of our analyses use real NVP [49], which have proven
 620 effective in our architecture searches, and have the advantageous features of fast sampling and fast
 621 density evaluation.

622 **A.1.4 Related work**

623 (To come)

624

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

626 (To come)

627

628 **A.2 Theoretical models**

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

631 **A.2.1 Stomatogastric ganglion**

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

$$C_m \frac{\partial x_m}{\partial t} = -[h_{leak}(x; z) + h_{Ca}(x; z) + h_K(x; z) + h_{hyp}(x; z) + h_{elec}(x; z) + h_{syn}(x; z)] \quad (20)$$

633 The membrane potential of each neuron is affected by the leak, calcium, potassium, hyperpolariza-
 634 tion, electrical and synaptic currents, respectively. The capacitance of the cell membrane was set to
 635 $C_m = 1nF$. Each current is a function of the neuron's membrane potential x_m and the parameters
 636 of the circuit such as g_{el} and g_{syn} , whose effect on the circuit is considered in the motivational
 637 example of EPI in Fig. 1. Specifically, the currents are the difference in the neuron's membrane
 638 potential and that current type's reversal potential multiplied by a conductance:

$$h_{leak}(x; z) = g_{leak}(x_m - V_{leak}) \quad (21)$$

$$h_{elec}(x; z) = g_{el}(x_m^{post} - x_m^{pre}) \quad (22)$$

$$h_{syn}(x; z) = g_{syn}S_\infty^{pre}(x_m^{post} - V_{syn}) \quad (23)$$

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

$$h_K(x; z) = g_KN(x_m - V_K) \quad (25)$$

$$h_{hyp}(x; z) = g_hH(x_m - V_{hyp}) \quad (26)$$

644 The reversal potentials were set to $V_{leak} = -40mV$, $V_{Ca} = 100mV$, $V_K = -80mV$, $V_{hyp} = -20mV$,
 645 and $V_{syn} = -75mV$. The other conductance parameters were fixed to $g_{leak} = 1 \times 10^{-4}\mu S$. g_{Ca} ,
 646 g_K , and g_{hyp} had different values based on fast, intermediate (hub) or slow neuron. Fast: $g_{Ca} =$
 647 1.9×10^{-2} , $g_K = 3.9 \times 10^{-2}$, and $g_{hyp} = 2.5 \times 10^{-2}$. Intermediate: $g_{Ca} = 1.7 \times 10^{-2}$, $g_K = 1.9 \times 10^{-2}$,
 648 and $g_{hyp} = 8.0 \times 10^{-3}$. Intermediate: $g_{Ca} = 8.5 \times 10^{-3}$, $g_K = 1.5 \times 10^{-2}$, and $g_{hyp} = 1.0 \times 10^{-2}$.

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

$$M_\infty = 0.5 \left(1 + \tanh \left(\frac{x_m - v_1}{v_2} \right) \right) \quad (27)$$

$$\frac{\partial N}{\partial t} = \lambda_N(N_\infty - N) \quad (28)$$

$$N_\infty = 0.5 \left(1 + \tanh \left(\frac{x_m - v_3}{v_4} \right) \right) \quad (29)$$

$$\lambda_N = \phi_N \cosh \left(\frac{x_m - v_3}{2v_4} \right) \quad (30)$$

$$\frac{\partial H}{\partial t} = \frac{(H_\infty - H)}{\tau_h} \quad (31)$$

$$H_\infty = \frac{1}{1 + \exp \left(\frac{x_m + v_5}{v_6} \right)} \quad (32)$$

$$\tau_h = 272 - \left(\frac{-1499}{1 + \exp \left(\frac{-x_m + v_7}{v_8} \right)} \right) \quad (33)$$

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 [20].

Finally, there is a synaptic gating variable as well:

$$S_\infty = \frac{1}{1 + \exp \left(\frac{v_{th} - x_m}{v_9} \right)} \quad (34)$$

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, $\Phi = [0.0, 0.01, \dots, 1.0]^\top$

Another consideration was that the frequency spectra of the hub neuron has several peaks. This 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

$$u_i = \frac{|\mathcal{X}_i|^\alpha}{\sum_{k=1}^N |\mathcal{X}_i(k)|^\alpha} \quad (35)$$

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 0.025Hz².

$$E \begin{bmatrix} \Omega_{\text{f1}} \\ \Omega_{\text{f2}} \\ \Omega_{\text{hub}} \\ \Omega_{\text{s1}} \\ \Omega_{\text{s2}} \\ (\Omega_{\text{f1}} - 0.55)^2 \\ (\Omega_{\text{f2}} - 0.55)^2 \\ (\Omega_{\text{hub}} - 0.55)^2 \\ (\Omega_{\text{s1}} - 0.55)^2 \\ (\Omega_{\text{s2}} - 0.55)^2 \end{bmatrix} = \begin{bmatrix} 0.55 \\ 0.55 \\ 0.55 \\ 0.55 \\ 0.55 \\ 0.025^2 \\ 0.025^2 \\ 0.025^2 \\ 0.025^2 \\ 0.025^2 \end{bmatrix} \quad (36)$$

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

695 have shown the EPI optimization that converged with maximum entropy across 2 random seeds
 696 and augmented Lagrangian coefficient initializations of $c_0=0$, 2, and 5.

697 **A.2.2 Primary visual cortex**

698 The dynamics of each neural populations average rate $x = \begin{bmatrix} x_E \\ x_P \\ x_S \\ x_V \end{bmatrix}$ are given by:

$$\tau \frac{dx}{dt} = -x + [Wx + h]_+^n \quad (37)$$

699 Some neuron-types largely lack synaptic projections to other neuron-types [50], and it is popular
 700 to only consider a subset of the effective connectivities [21].

$$W = \begin{bmatrix} W_{EE} & W_{EP} & W_{ES} & 0 \\ W_{PE} & W_{PP} & W_{PS} & 0 \\ W_{SE} & 0 & 0 & W_{SV} \\ W_{VE} & W_{VP} & W_{VS} & 0 \end{bmatrix} \quad (38)$$

701 Estimates of the probability of connection and strength of connection from the Allen institute
 702 result in an estimate of the effective connectivity [?]:

$$W = \begin{bmatrix} 0.0576 & 0.19728 & 0.13144 & 0 \\ 0.58855 & 0.30668 & 0.4285 & 0 \\ 0.15652 & 0 & 0 & 0.2 \\ 0.13755 & 0.0902 & 0.4004 & 0 \end{bmatrix} \quad (39)$$

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

$$h = b + dh \quad (40)$$

707 All simulations of this system had $T = 100$ time points, a time step $dt = 5\text{ms}$, and time constant
 708 $\tau = 20\text{ms}$. And the system was initialized to a random draw $x(0)_i \sim \mathcal{N}(1, 0.01)$.

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

$$\dot{x}_i = -x_i + f(u_i) \quad (41)$$

710 where the input to each neuron is

$$u_i = \sum_j W_{ij}x_j + h_i \quad (42)$$

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

$$\frac{\partial x_{ss}}{\partial h} = F(W \frac{\partial x_{ss}}{\partial h} + I) \quad (43)$$

712 which is calculable by

$$\frac{\partial x_{ss}}{\partial h} = (F^{-1} - W)^{-1} \quad (44)$$

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

$$\mathcal{B}(\alpha, y) \leftrightarrow E \begin{bmatrix} dx_{\alpha,ss} \\ (dx_{\alpha,ss} - y)^2 \end{bmatrix} = \begin{bmatrix} y \\ 0.01^2 \end{bmatrix} \quad (45)$$

716 In the final analysis for this model, we sweep the input one neuron at a time away from the mode
 717 of each inferred distributions $dh^* = z^* = \text{argmax}_z \log q_\theta(z | \mathcal{B}(\alpha, 0.1))$. The differential responses
 718 $dx_{\alpha,ss}$ are examined at perturbed inputs $h = b + dh^* + \Delta h_\alpha u_\alpha$ where u_α is a unit vector in the
 719 dimension of α and $\Delta h_\alpha \in [-15, 15]$.

720 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
 721 initial seeds using an architecture of four coupling layers, each with two hidden layers of 10 units.
 722 We set $c_0 = 10^5$. The support of the learned distribution was restricted to $z_i \in [-5, 5]$.

723 A.2.3 Superior colliculus

724 There are four total units: two in each hemisphere corresponding to the Pro/contralateral and
 725 Anti/ipsilateral populations. Each unit has an activity (x_i) and internal variable (u_i) related by

$$x_i(t) = \left(\frac{1}{2} \tanh \left(\frac{v_i(t) - \epsilon}{\zeta} \right) + \frac{1}{2} \right) \quad (46)$$

726 $\epsilon = 0.05$ and $\zeta = 0.5$ control the position and shape of the nonlinearity, respectively.

727 We can order the elements of x_i and v_i into vectors x and v with elements

$$x = \begin{bmatrix} x_{LP} \\ x_{LA} \\ x_{RP} \\ x_{RA} \end{bmatrix} \quad v = \begin{bmatrix} v_{LP} \\ v_{LA} \\ v_{RP} \\ v_{RA} \end{bmatrix} \quad (47)$$

728 The internal variables follow dynamics:

$$\tau \frac{\partial v}{\partial t} = -v + Wx + h + \sigma \partial B \quad (48)$$

729 with time constant $\tau = 0.09s$ and gaussian noise $\sigma \partial B$ controlled by the magnitude of $\sigma = 1.0$. The
730 weight matrix has 8 parameters sW_P , sW_A , vW_{PA} , vW_{AP} , hW_P , hW_A , dW_{PA} , and dW_{AP} (Fig.
731 4B).

$$W = \begin{bmatrix} sW_P & vW_{PA} & hW_P & dW_{PA} \\ vW_{AP} & sW_A & dW_{AP} & hW_A \\ hW_P & dW_{PA} & sW_P & vW_{PA} \\ dW_{AP} & hW_A & vW_{AP} & sW_A \end{bmatrix} \quad (49)$$

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

$$h = h_{\text{rule}} + h_{\text{choice-period}} + h_{\text{light}} \quad (50)$$

733 There are rule-based inputs depending on the condition,

$$h_{P,\text{rule}}(t) = \begin{cases} I_{P,\text{rule}} \begin{bmatrix} 1 & 0 & 0 & 1 \end{bmatrix}^\top, & \text{if } t \leq 1.2s \\ 0, & \text{otherwise} \end{cases} \quad (51)$$

$$h_{A,\text{rule}}(t) = \begin{cases} I_{A,\text{rule}} \begin{bmatrix} 0 & 1 & 1 & 0 \end{bmatrix}^\top, & \text{if } t \leq 1.2s \\ 0, & \text{otherwise} \end{cases} \quad (52)$$

734 a choice-period input,

$$h_{\text{choice}}(t) = \begin{cases} I_{\text{choice}} \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix}^\top, & \text{if } t > 1.2s \\ 0, & \text{otherwise} \end{cases} \quad (53)$$

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

$$h_{\text{light}}(t) = \begin{cases} I_{\text{light}} \begin{bmatrix} 1 & 1 & 0 & 0 \end{bmatrix}^\top, & \text{if } t > 1.2s \text{ and Left} \\ I_{\text{light}} \begin{bmatrix} 0 & 0 & 1 & 1 \end{bmatrix}^\top, & \text{if } t > 1.2s \text{ and Right} \\ 0, & t \leq 1.2s \end{cases} \quad (54)$$

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

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

$$\hat{p}_i = E_{\sigma \partial B} [x_{LP,ss} | s = L, c = P, z_i] = \frac{1}{M} \sum_{j=1}^M x_{LP,ss}(s = L, c = P, z_i, \sigma \partial B_j) \quad (55)$$

⁷⁴¹ For the first constraint, the average over posterior samples (from $q_\theta(z)$) to be p_{LP} :

$$E_{z_i \sim q_\phi} [E_{\sigma \partial B} [x_{LP,ss} | s = L, c = P, z_i]] = E_{z_i \sim q_\phi} [\hat{p}_i] = p_{LP} \quad (56)$$

⁷⁴² We can then ask that the variance of the steady state responses across gaussian draws, is the
⁷⁴³ Bernoulli variance for the empirical rate \hat{p}_i .

$$E_{z \sim q_\phi} [\sigma_{err}^2] = 0 \quad (57)$$

⁷⁴⁴

$$\sigma_{err}^2 = Var_{\sigma \partial B} [x_{LP,ss} | s = L, c = P, z_i] - \hat{p}_i(1 - \hat{p}_i) \quad (58)$$

⁷⁴⁵ We have an additional constraint that the Pro neuron on the opposite hemisphere should have the
⁷⁴⁶ opposite value. We can enforce this with a final constraint:

$$E_{z \sim q_\phi} [d_P] = 1 \quad (59)$$

⁷⁴⁷

$$E_{\sigma \partial W} [(x_{LP,ss} - x_{RP,ss})^2 | s = L, c = P, z_i] \quad (60)$$

⁷⁴⁸ We refer to networks obeying these constraints as Bernoulli, winner-take-all networks. Since the
⁷⁴⁹ maximum variance of a random variable bounded from 0 to 1 is the Bernoulli variance ($\hat{p}(1 - \hat{p})$),
⁷⁵⁰ and the maximum squared difference between two variables bounded from 0 to 1 is 1, we do not
⁷⁵¹ need to control the second moment of these test statistics. In reality, these variables are dynamical
⁷⁵² system states and can only exponentially decay (or saturate) to 0 (or 1), so the Bernoulli variance
⁷⁵³ error and squared difference constraints can only be undershot. This is important to be mindful
⁷⁵⁴ of when evaluating the convergence criteria. Instead of using our usual hypothesis testing criteria
⁷⁵⁵ for convergence to the emergent property, we set a slack variable threshold for these technically
⁷⁵⁶ infeasible constraints to 0.05.

⁷⁵⁷ Training DSNs to learn distributions of dynamical system parameterizations that produce Bernoulli
⁷⁵⁸ responses at a given rate (with small variance around that rate) was harder to do than expected.
⁷⁵⁹ There is a pathology in this optimization setup, where the learned distribution of weights is bimodal
⁷⁶⁰ attributing a fraction p of the samples to an expansive mode (which always sends x_{LP} to 1), and a
⁷⁶¹ fraction $1 - p$ to a decaying mode (which always sends x_{LP} to 0). This pathology was avoided using
⁷⁶² an inequality constraint prohibiting parameter samples that resulted in low variance of responses
⁷⁶³ across noise.

⁷⁶⁴ In total, the emergent property of rapid task switching accuracy at level p was defined as

$$\mathcal{B}(p) \leftrightarrow \begin{bmatrix} \hat{p}_P \\ \hat{p}_A \\ (\hat{p}_P - p)^2 \\ (\hat{p}_A - p)^2 \\ \sigma_{P,err}^2 \\ \sigma_{A,err}^2 \\ d_P \\ d_A \end{bmatrix} = \begin{bmatrix} p \\ p \\ 0.15^2 \\ 0.15^2 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} \quad (61)$$

⁷⁶⁵ For each accuracy level p , we ran EPI for 10 different random seeds and selected the maximum

⁷⁶⁶ entropy solution using an architecture of 10 planar flows with $c_0 = 2$. The support of z was \mathcal{R}^8 .

⁷⁶⁷ **A.2.4 Rank-1 RNN**

⁷⁶⁸ The network dynamics of neuron i 's rate x evolve according to:

$$\dot{x}_i(t) = -x_i(t) + \sum_{j=1}^N J_{ij}\phi(x_j(t)) + I_i \quad (62)$$

⁷⁶⁹ where the connectivity is comprised of a random and structured component:

$$J_{ij} = g\chi_{ij} + P_{ij} \quad (63)$$

⁷⁷⁰ The random bulk component has elements drawn from $\chi_{ij} \sim \mathcal{N}(0, \frac{1}{N})$, and the structured compo-
⁷⁷¹ nent is a sum of r unit rank terms:

$$P_{ij} = \sum_{k=1}^r \frac{m_i^{(k)} n_j^{(k)}}{N} \quad (64)$$

⁷⁷² Rank-1 vectors m and n have elements drawn

$$m_i \sim \mathcal{N}(M_m, \Sigma_m)$$

⁷⁷³

$$n_i \sim \mathcal{N}(M_n, \Sigma_n)$$

⁷⁷⁴ The current has the following statistics:

$$I = M_I + \frac{\Sigma_{mI}}{\Sigma_m} x_1 + \frac{\Sigma_{nI}}{\Sigma_n} x_2 + \Sigma_\perp h$$

⁷⁷⁵ where x_1 , x_2 , and h are standard normal random variables following the rank-1 input-driven ex-

⁷⁷⁶ ample from [23].

⁷⁷⁷ We followed their prescription for deriving the consistency equations in the presence of chaos. The
⁷⁷⁸ $\ddot{\Delta}$ equation is broken into the equation for Δ_0 and Δ_∞ by the autocorrelation dynamics assertions.

$$\ddot{\Delta}(\tau) = -\frac{\partial V}{\partial \Delta}$$

⁷⁷⁹

$$\ddot{\Delta} = \Delta - \{g^2 \langle [\phi_i(t)\phi_i(t+\tau)] \rangle + \Sigma_m^2 \kappa^2 + 2\Sigma_{mI}\kappa + \Sigma_I^2\}$$

⁷⁸⁰ We can write out the potential function by integrating the negated RHS.

$$V(\Delta, \Delta_0) = \int \mathcal{D}\Delta \frac{\partial V(\Delta, \Delta_0)}{\partial \Delta}$$

⁷⁸¹

$$V(\Delta, \Delta_0) = -\frac{\Delta^2}{2} + g^2 \langle [\Phi_i(t)\Phi_i(t+\tau)] \rangle + (\Sigma_m^2 \kappa^2 + 2\Sigma_{mI}\kappa + \Sigma_I^2)\Delta + C$$

⁷⁸² We assume that as time goes to infinity, the potential relaxes to a steady state.

$$\frac{\partial V(\Delta_\infty, \Delta_0)}{\partial \Delta} = -\Delta + \{g^2 \langle [\phi_i(t)\phi_i(t+\infty)] \rangle + \Sigma_m^2 \kappa^2 + 2\Sigma_{mI}\kappa + \Sigma_I^2\} = 0$$

⁷⁸³

$$\Delta_\infty = g^2 \langle [\phi_i(t)\phi_i(t+\infty)] \rangle + \Sigma_m^2 \kappa^2 + 2\Sigma_{mI}\kappa + \Sigma_I^2$$

⁷⁸⁴ This can be written more explicitly in terms of the gaussian integrals which are relatively (with
⁷⁸⁵ respect to nongaussian distributions) cheap to evaluate.

$$\Delta_\infty = g^2 \int \mathcal{D}z \left[\int \mathcal{D}x \phi(\mu + \sqrt{\Delta_0 - \Delta_\infty}x + \sqrt{\Delta_\infty}z) \right]^2 + \Sigma_m^2 \kappa^2 + 2\Sigma_{mI}\kappa + \Sigma_I^2$$

⁷⁸⁶ Also, we assume that the energy of the system is preserved throughout the entirety of its evolution.

$$V(\Delta_0, \Delta_0) = V(\Delta_\infty, \Delta_0)$$

⁷⁸⁷

$$-\frac{\Delta_0^2}{2} + g^2 \langle [\Phi_i(t)\Phi_i(t)] \rangle + (\Sigma_m^2 \kappa^2 + 2\Sigma_{mI}\kappa + \Sigma_I^2)\Delta_0 + C = -\frac{\Delta_\infty^2}{2} + g^2 \langle [\Phi_i(t)\Phi_i(t)] \rangle + (\Sigma_m^2 \kappa^2 + 2\Sigma_{mI}\kappa + \Sigma_I^2)\Delta_\infty + C$$

⁷⁸⁸ We can arrange the terms into a difference of squares in Δ_0 and Δ_∞ .

$$\frac{\Delta_0^2 - \Delta_\infty^2}{2} = g^2 (\langle [\Phi_i(t)\Phi_i(t)] \rangle - \langle [\Phi_i(t)\Phi_i(t)] \rangle) + (\Sigma_m^2 \kappa^2 + 2\Sigma_{mI}\kappa + \Sigma_I^2)(\Delta_0 - \Delta_\infty)$$

⁷⁸⁹ Similarly, we write out the resulting equation explicitly in terms of the gaussian integrals present.

$$\frac{\Delta_0^2 - \Delta_\infty^2}{2} = g^2 \left(\int \mathcal{D}z \Phi^2(\mu + \sqrt{\Delta_0}z) - \int \mathcal{D}z \int \mathcal{D}x \Phi(\mu + \sqrt{\Delta_0 - \Delta_\infty}x + \sqrt{\Delta_\infty}z) \right) + (\Sigma_m^2 \kappa^2 + 2\Sigma_{mI}\kappa + \Sigma_I^2)(\Delta_0 - \Delta_\infty)$$

This results in a set of consistency equations for the dynamic mean field variables μ , κ , Δ_0 , and Δ_∞ . In order to obtain the values of these variables for a given parameterization, we must solve the following system of equations.

$$\begin{aligned} \mu &= F(\mu, \kappa, \Delta_0, \Delta_\infty) = M_m \kappa + M_I \\ \kappa &= G(\mu, \kappa, \Delta_0, \Delta_\infty) = M_n \langle [\phi_i] \rangle + \Sigma_{nI} \langle [\phi'_i] \rangle \\ \frac{\Delta_0^2 - \Delta_\infty^2}{2} &= H(\mu, \kappa, \Delta_0, \Delta_\infty) = g^2 \left(\int \mathcal{D}z \Phi^2(\mu + \sqrt{\Delta_0} z) - \int \mathcal{D}z \int \mathcal{D}x \Phi(\mu + \sqrt{\Delta_0 - \Delta_\infty} x + \sqrt{\Delta_\infty} z) \right) \\ &\quad + (\Sigma_m^2 \kappa^2 + 2\Sigma_{mI} \kappa + \Sigma_I^2)(\Delta_0 - \Delta_\infty) \\ \Delta_\infty &= L(\mu, \kappa, \Delta_0, \Delta_\infty) = g^2 \int \mathcal{D}z \left[\int \mathcal{D}x \phi(\mu + \sqrt{\Delta_0 - \Delta_\infty} x + \sqrt{\Delta_\infty} z) \right]^2 + \Sigma_m^2 \kappa^2 + 2\Sigma_{mI} \kappa + \Sigma_I^2 \end{aligned} \tag{65}$$

We can solve these equations by simulating the following Langevin dynamical system.

$$\begin{aligned} x(t) &= \frac{\Delta_0(t)^2 - \Delta_\infty(t)^2}{2} \\ \Delta_0(t) &= \sqrt{2x(t) + \Delta_\infty(t)^2} \\ \dot{\mu}(t) &= -\mu(t) + F(\mu(t), \kappa(t), \Delta_0(t), \Delta_\infty(t)) \\ \dot{\kappa}(t) &= -\kappa + G(\mu(t), \kappa(t), \Delta_0(t), \Delta_\infty(t)) \\ \dot{x}(t) &= -x(t) + H(\mu(t), \kappa(t), \Delta_0(t), \Delta_\infty(t)) \\ \dot{\Delta_\infty}(t) &= -\Delta_\infty(t) + L(\mu(t), \kappa(t), \Delta_0(t), \Delta_\infty(t)) \end{aligned} \tag{66}$$

Then, the temporal variance, which is necessary for the gaussian posterior conditioning example, is simply calculated via

$$\Delta_T = \Delta_0 - \Delta_\infty \tag{67}$$

797 A.3 Supplementary Figures

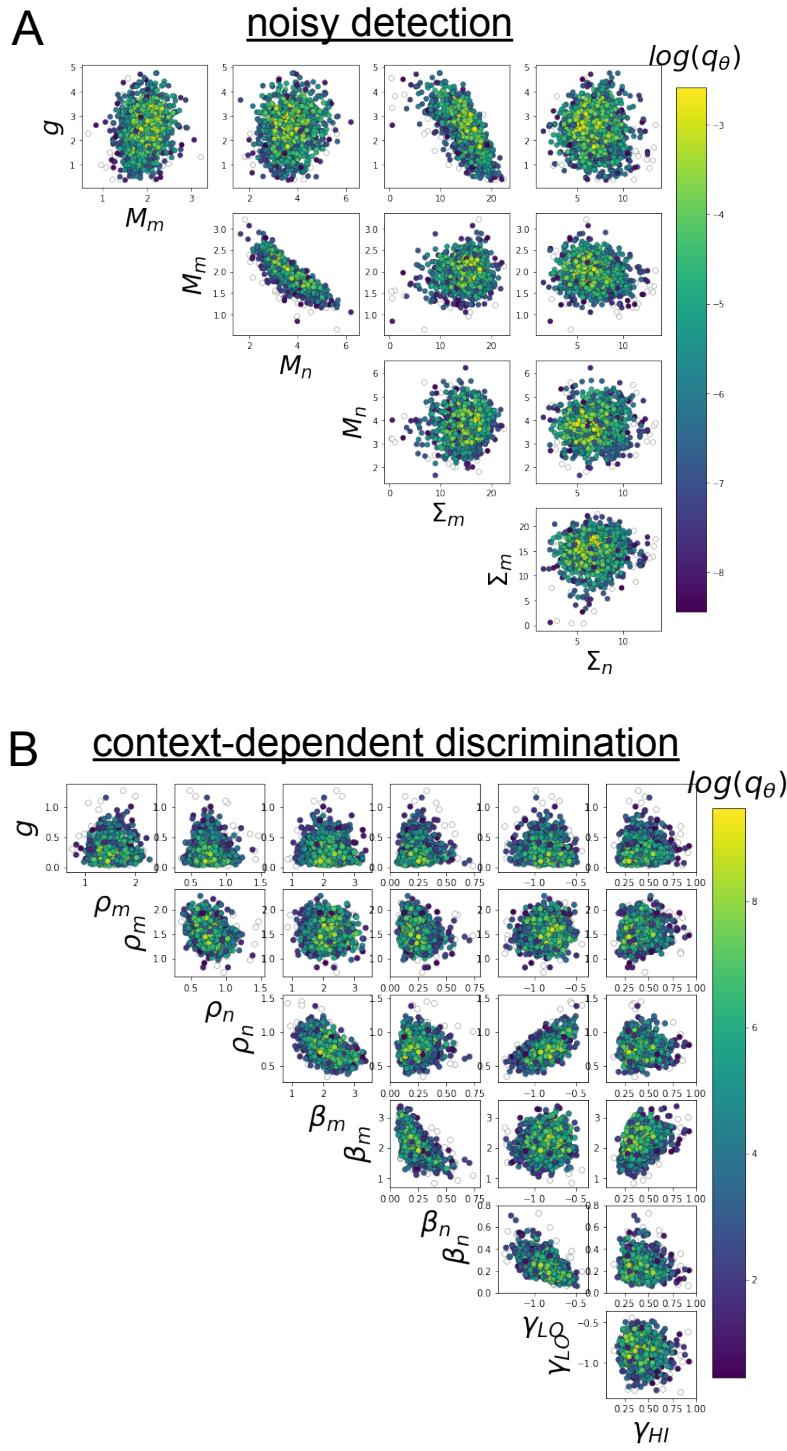


Fig. S1: A. EPI for rank-1 networks doing discrimination. B. EPI for rank-2 networks doing context-dependent discrimination.