

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.

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 increasing its complexity, often resulting in *ad hoc* approaches to understanding
¹⁰ the relationship between model and computation. We bring recent machine learning techniques –
¹¹ the use of deep generative models for probabilistic inference – to bear on this problem, learning
¹² distributions of parameters that produce the specified properties of computation. Importantly, the
¹³ techniques we introduce offer a principled means to understand the implications of model parameter
¹⁴ choices on computational properties of interest. We motivate this methodology with a worked
¹⁵ example analyzing sensitivity in the stomatogastric ganglion. We then use it to generate insights
¹⁶ into neuron-type input-responsivity in a model of primary visual cortex, a new understanding
¹⁷ of rapid task switching in superior colliculus models, and attribution of bias in recurrent neural
¹⁸ networks solving a toy mathematical problem. More generally, this work offers a quantitative
¹⁹ grounding for theoretical models going forward, pointing a way to how rigorous statistical inference
²⁰ can enhance theoretical neuroscience at large.

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
58 set of insights into these models. Indeed, only in the last few years has deep learning research

59 advanced 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 feature
75 of the model, which is ultimately the object of interest to the theorist (compared to the statisti-
76 cal 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, suggests a depar-
 95 ture from realism vs tractability considerations towards the use of modern machine learning for
 96 sophisticated interrogation of biologically relevant models.

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]. Despite full knowledge of STG connectivity
 114 and a precise characterization of its rhythmic pattern generation, biophysical models of the STG
 115 have complicated relationships between circuit parameters and neural activity [28]. A model of the
 116 STG [20] is shown schematically in Figure 1A, and note that the behavior of this model will be crit-
 117 ically dependent on its parameterization – the choices of conductance parameters $z = [g_{el}, g_{synA}]$.
 118 Specifically, the two fast neurons (f_1 and f_2) mutually inhibit one another, and oscillate at a
 119 faster frequency than the mutually inhibiting slow neurons (s_1 and s_2), and the hub neuron (hub)
 120 couples with the fast or slow population or both.

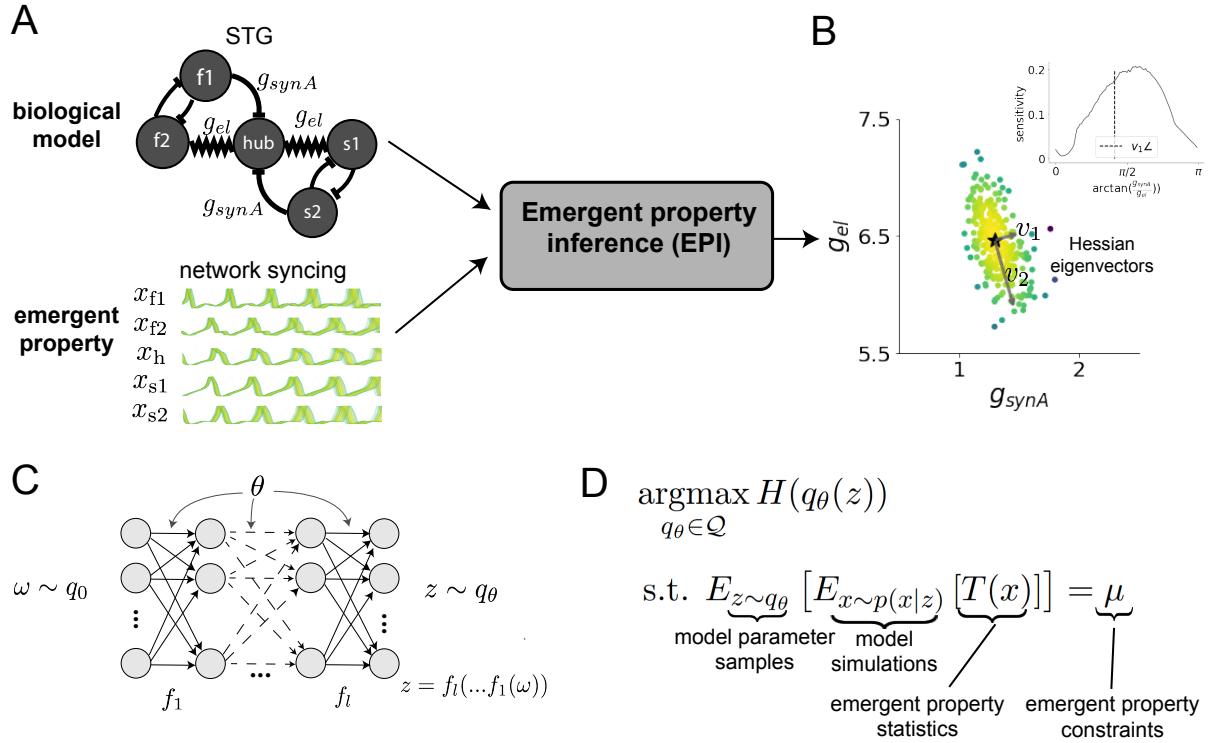


Figure 1: Emergent property inference (EPI) in the stomatogastric ganglion. A. For a choice of model (STG) and emergent property (network syncing), emergent property inference (EPI) learns a posterior distribution of the model parameters $z = [g_{el}, g_{synA}]^\top$ conditioned on network syncing. B.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

121 Second, once the model is selected, the theorist defines the emergent property, the measurable
 122 signal of scientific interest. To continue our running STG example, one such emergent property
 123 is the phenomenon of *network syncing* – in certain parameter regimes, the frequency of the hub
 124 neuron matches that of the fast and slow populations at an intermediate frequency. This emergent
 125 property is shown in Figure 1A at a frequency of 0.55Hz.

126 Third, qualitative parameter analysis ensues: since precise mathematical analysis is intractable in
 127 this model, a brute force sweep of parameters is done [20]. Subsequently, a qualitative description is
 128 formulated to describe of the different parameter configurations that lead to the emergent property.
 129 In this last step lies the opportunity for a precise quantification of the emergent property as a
 130 statistical feature of the model. Once we have such a methodology, we can infer a probability
 131 distribution over parameter configurations that produce this emergent property.

132 Before presenting technical details (in the following section), let us understand emergent property
 133 inference schematically: the black box in Figure 1A takes, as input, the model and the specified
 134 emergent property, and produces as output the parameter distribution shown in Figure 1B. This
 135 distribution – represented for clarity as samples from the distribution – is then a scientifically
 136 meaningful and mathematically tractable object. It conveys parameter regions critical to the emer-
 137 gent property, directions in parameter space that will be invariant (or not) to that property, and
 138 more. In the STG model, this distribution can be specifically queried to determine the prototypical
 139 parameter configuration for network syncing (the mode; Figure 1B star), and then how quickly
 140 network syncing will decay based on changes away from that mode. The inset of Figure 1B vali-
 141 dates that indeed network syncing behaves as the distribution predicts, when moving away from
 142 the mode (Figure 1B star). Further validation of EPI is available in the supplementary materials,
 143 where we analyze a simpler model for which ground-truth statements can be made (Section A.1.1).

144 3.2 A deep generative modeling approach to emergent property inference

145 Emergent property inference (EPI) systematizes the three-step procedure of the previous section.
 146 First, we consider the model as a coupled set of differential (and potentially stochastic) equations
 147 [20]. In the running STG example, the dynamical state $x = [x_{f1}, x_{f2}, x_{hub}, x_{s1}, x_{s2}]$ is the membrane
 148 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 $z = [g_{\text{el}}, g_{\text{synA}}]$ (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 $\omega_{\text{f1}}(x)$, $\omega_{\text{f2}}(x)$, 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 $i \in \{\text{f1}, \text{f2}, \text{hub}, \text{s1}, \text{s2}\}$, and thus:

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

which completes the quantification of the emergent property.

Third, we perform emergent property inference: we find a distribution over parameter configurations z , and insist that samples from this distribution produce the emergent property; in other words, they obey the constraints introduced in Equation 2. This distribution will be chosen from a family of probability distributions $\mathcal{Q} = \{q_\theta(z) : \theta \in \Theta\}$, defined by a deep generative distribution of the normalizing flow class [15, 16, 17] – 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 2, and $p(x|z)$ is the intractable distribution of data from the model (x), given that model’s parameters z (we access samples from this distribution by running the model forward). The purpose of each element in this program is detailed in Figure 1D. Finally, we recognize that many distributions in \mathcal{Q} will respect the emergent property constraints, so we require a normative principle to select amongst them. This principle is captured in Equation 3 by the primal objective H . Here we chose Shannon entropy as a means to find parameter distributions with minimal assumptions beyond some chosen structure [29, 30, 18, 31], but we emphasize that

¹⁷⁴ the EPI method is unaffected by this choice (but the results of course will depend on the primal
¹⁷⁵ objective chosen).

¹⁷⁶ EPI optimizes the weights and biases θ of the deep neural network (which induces the probability
¹⁷⁷ distribution) by iteratively solving Equation 3. The optimization is complete when the sampled
¹⁷⁸ models with parameters $z \sim q_\theta$ produce activity consistent with the specified emergent property.
¹⁷⁹ Such convergence is evaluated with a hypothesis test that the mean of each emergent property
¹⁸⁰ statistic is not different than its emergent property value (see Section A.1.2). Equipped with this
¹⁸¹ method, we now prove out the value of EPI by using it to investigate three prominent models in
¹⁸² neuroscience, using EPI to produce new insights about these models.

¹⁸³ **3.3 Comprehensive input-responsivity in a nonlinear sensory system**

¹⁸⁴ In studies of primary visual cortex (V1), theoretical models with excitatory (E) and inhibitory
¹⁸⁵ (I) populations have reproduced a host of experimentally documented phenomena. In particular
¹⁸⁶ regimes of excitation and inhibition, these E/I models exhibit the paradoxical effect [4], selective
¹⁸⁷ amplification [32], surround suppression [33], and sensory integrative properties [34]. Extending
¹⁸⁸ this model using experimental evidence of three genetically-defined classes of inhibitory neurons
¹⁸⁹ [35, 36], recent work [21] has investigated a four-population model – excitatory (E), parvalbumin
¹⁹⁰ (P), somatostatin (S), and vasointestinal peptide (V) neurons – as shown in Fig. 2A. The dynamical
¹⁹¹ state of this model is the firing rate of each neuron-type population $x = [x_E, x_P, x_S, x_V]^\top$, which
¹⁹² evolves according to rectified ($\llbracket \cdot \rrbracket_+$) and exponentiated dynamics:

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

¹⁹³ with effective connectivity weights W and input h . In our analysis, we set the time constant
¹⁹⁴ $\tau = 20\text{ms}$ and dynamics coefficient $n = 2$. Also, as is fairly standard, we obtain an informative
¹⁹⁵ estimate of the effective connectivities between these neuron-types W in mice by multiplying their
¹⁹⁶ probability of connection with their average synaptic strength [37, 38] (see Section A.2.2). Given
¹⁹⁷ these fixed choices of W , n , and τ , we studied the system’s response to input

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

¹⁹⁸ where the input h is comprised of a baseline input $b = [b_E, b_P, b_S, b_V]^\top$ and a differential input
¹⁹⁹ $dh = [dh_E, dh_P, dh_S, dh_V]^\top$ to each neuron-type population. Throughout subsequent analyses, the
²⁰⁰ baseline input is $b = [1, 1, 1, 1]^\top$.

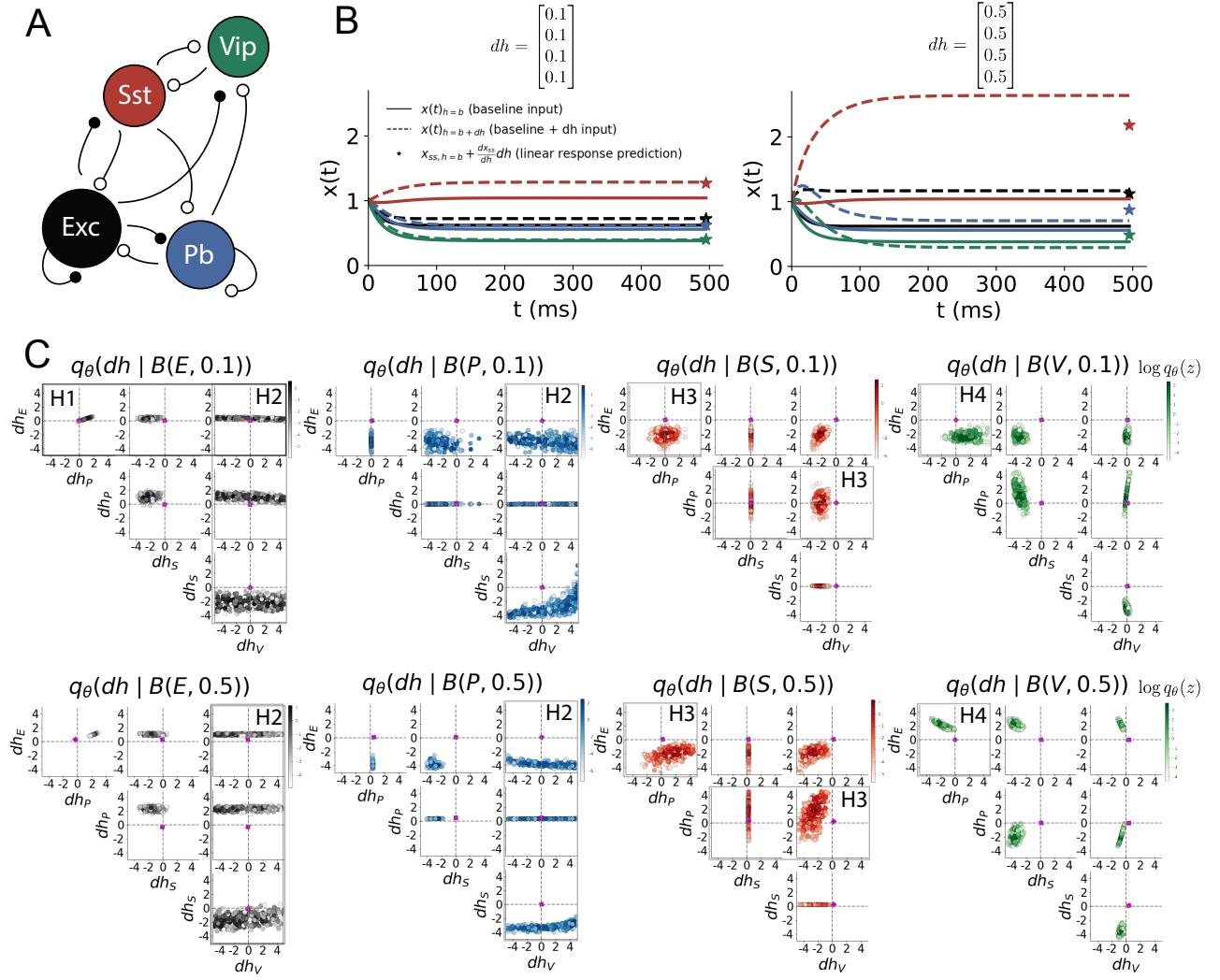


Figure 2: Hypothesis generation through EPI in a V1 model. A. Four-population model of primary visual cortex with excitatory (black), parvalbumin (blue), somatostatin (red), and vip (green) neurons. Some neuron-types largely do not form synaptic projections to others (excitatory and inhibitory projections 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).

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

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

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

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

221

- 222 H1: as is intuitive, each neuron-type's firing rate should be sensitive to that neuron-type's direct input (e.g. Fig. 2C H1 indicates low variance in dh_E when $\alpha = E$. Same observation in all inferred distributions);
- 225 H2: the E- and P-populations should be largely unaffected by dh_V (Fig. 2C H2 indicates high variance in dh_V when $\alpha \in \{E, P\}$);
- 227 H3: the S-population should be largely unaffected by dh_P (Fig. 2C H3 indicate high variance in dh_P when $\alpha = S$);
- 229 H4: there should be a nonmonotonic response of $dx_{V,ss}$ with dh_E (Fig. 2C H4 indicates that negative dh_E should result in small $dx_{V,ss}$, but positive dh_E should elicit a larger $dx_{V,ss}$);

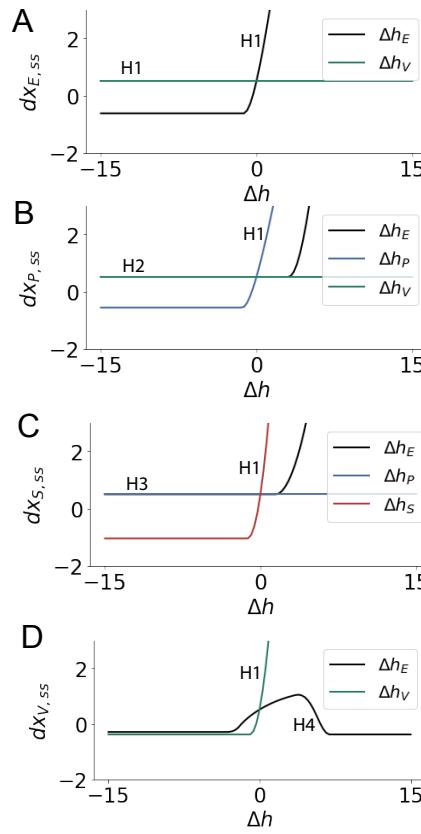


Figure 3: Confirming EPI generated hypotheses in V1. A. Differential responses by the E-population to changes in individual input $\Delta h_\alpha u_\alpha$ away from the mode of the EPI distribution dh^* . B-D Same plots for the P-, S-, and V-populations. Labels H1, H2, H3, and H4 indicate which curves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

231 We evaluate these hypotheses by taking steps in individual neuron-type input Δh_α away from the
232 modes of the inferred distributions at $y = 0.1$.

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

233 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
234 is a unit vector in the dimension of α . The EPI-generated hypotheses are confirmed.

- 235 • the neuron-type responses are sensitive to their direct inputs (Fig. 3A black, 3B blue, 3C
236 red, 3D green);
- 237 • the E- and P-populations are not affected by dh_V (Fig. 3A green, 3B green);
- 238 • the S-population is not affected by dh_P (Fig. 3C blue);
- 239 • the V-population exhibits a nonmonotonic response to dh_E (Fig. 3D black), and is in fact
240 the only population to do so (Fig. 3A-C black).

241 These hypotheses were in stark contrast to what was available to us via traditional analytical linear
242 prediction (Fig. 2C, magenta). To this point, we have shown the utility of EPI on relatively low-
243 level emergent properties like network syncing and differential neuron-type population responses.

²⁴⁴ In the remainder of the study, we focus on using EPI to understand models of more abstract
²⁴⁵ cognitive function.

²⁴⁶ **3.4 Identifying neural mechanisms of behavioral learning.**

²⁴⁷ Identifying measurable biological changes that result in improved behavior is important for neuro-
²⁴⁸ science, since they may indicate how the learning brain adapts. In a rapid task switching experiment
²⁴⁹ [39], rats were explicitly cued on each trial to either orient towards a visual stimulus in the Pro
²⁵⁰ (P) task or orient away from a visual stimulus in the Anti (A) task (Fig. 3a). Neural recordings
²⁵¹ in the midbrain supeior colliculus (SC) exhibited two population of neurons that simultaneously
²⁵² represented both task context (Pro or Anti) and motor response (contralateral or ipsilateral to the
²⁵³ recoreded side): the Pro/Contra and Anti/Ipsi neurons [22]. Duan et al. proposed a model of SC
²⁵⁴ that, like the V1 model analyzed in the previous section, is a four-population dynamical system.
²⁵⁵ Here, the neuron-type populations are functionally-defined as the Pro- and Anti-populations in each
²⁵⁶ hemisphere (left (L) and right (R)). The Pro- or Anti-populations receive an input determined by
²⁵⁷ the cue, and then the left and right populations receive an input based on the side of the light
²⁵⁸ stimulus. Activities were bounded between 0 and 1, so that a high output of the Pro population
²⁵⁹ in a given hemisphere corresponds to the contralateral response. An additional stipulation is that
²⁶⁰ when one Pro population responds with a high-output, the opposite Pro population must respond
²⁶¹ with a low output. Finally, this circuit operates in the presence of Gaussian noise resulting in trial-
²⁶² to-trial variability (see Section A.2.3). The connectivity matrix is parameterized by the geometry
²⁶³ of the population arrangement (Fig. 3B).

²⁶⁴ Here, we used EPI to learn distributions of the SC weight matrix parameters $z = W$ conditioned
²⁶⁵ on of various levels of rapid task switching accuracy $\mathcal{B}(p)$ for $p \in \{50\%, 60\%, 70\%, 80\%, 90\%\}$ (see
²⁶⁶ Section A.2.3). Following the approach in Duan et al., we decomposed the connectivity matrix
²⁶⁷ $W = QAQ^{-1}$ in such a way (the Schur decomposition) that the basis vectors q_i are the same for all
²⁶⁸ W (Fig. 3C). These basis vectors have intuitive roles in processing for this task, and are accordingly
²⁶⁹ named the *all* mode - all neurons co-fluctuate, *side* mode - one side dominates the other, *task* mode
²⁷⁰ - the Pro or Anti populations dominate the other, and *diag* mode - Pro- and Anti-populations of
²⁷¹ opposite hemispheres dominate the opposite pair. The corresponding eigenvalues (e.g. a_{task} , which
²⁷² change according to W) indicate the degree to which activity along that mode is increased or
²⁷³ decreased by W .

²⁷⁴ EPI demonstrates that, for greater task accuracies, the task mode eigenvalue increases, indicating

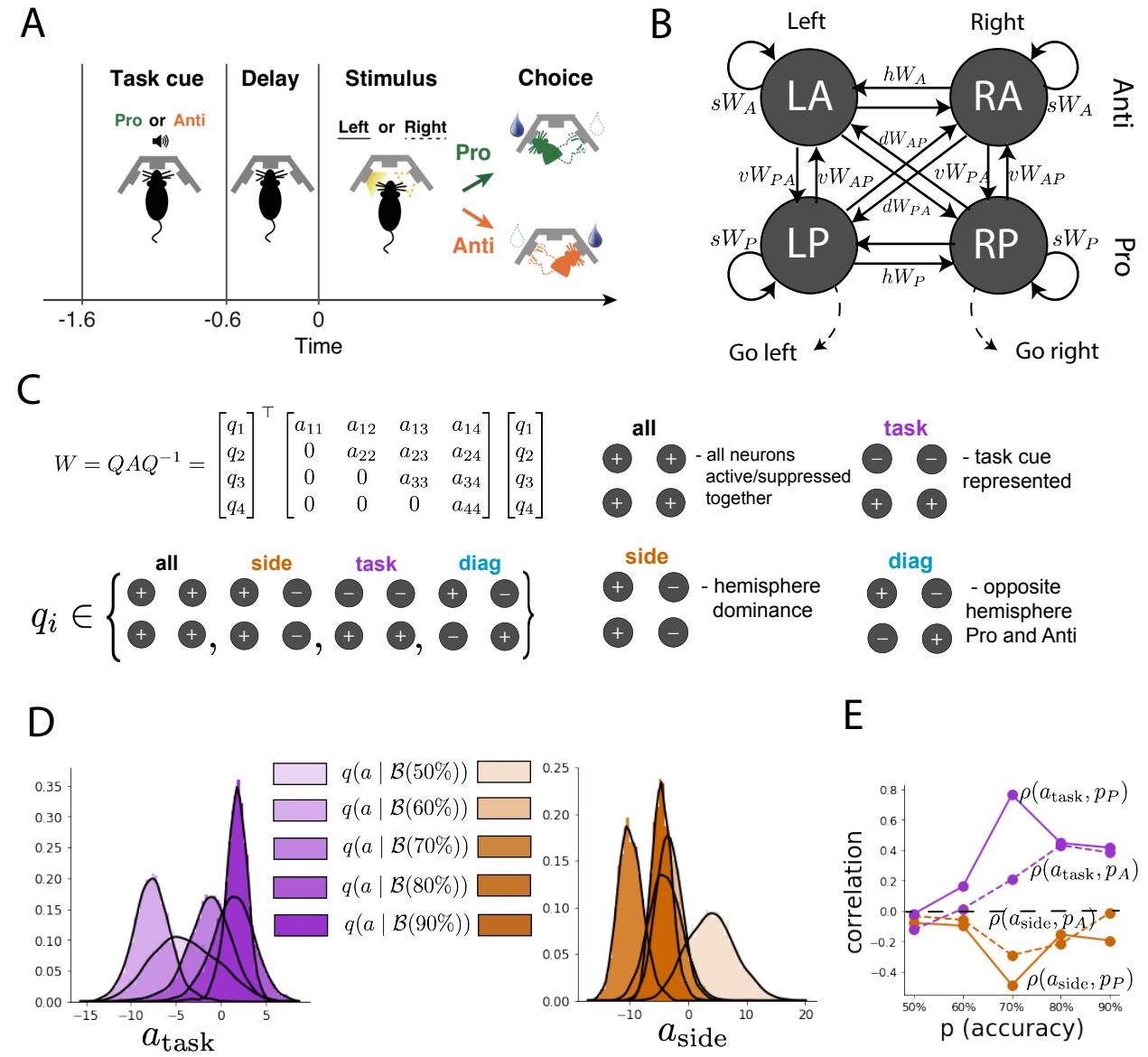


Figure 4: EPI reveals changes in SC [22] connectivity that control task accuracy. A. Rapid task switching behavioral paradigm (see text). B. Model of superior colliculus (SC). Neurons: LP - left pro, RP - right pro, LA - left anti, RA - right anti. Parameters: sW - self, hW - horizontal, vW - vertical, 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_{all} , q_{task} , q_{side} , and q_{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.

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

3.5 Characterizing biases in RNNs solving a posterior conditioning task

So far, each model we have studied was designed from fundamental biophysical principles, genetically- or functionally-defined neuron types. At a more abstract level of modeling, recurrent neural networks (RNNs) are high-dimensional dynamical models of computation becoming increasingly popular in neuroscience research [40]. In theoretical neuroscience, RNNs dynamics usually follow the equation

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

where $x(t)$ is the network activity, W is the network connectivity, $\phi(\cdot) = \tanh(\cdot)$, and $I(t)$ is the input to the system. Such RNNs are trained to do a task from a systems neuroscience experiment, and then the unit activations of the trained RNN are compared to recorded neural activity. Such highly parameterized models are challenging to characterize, let alone probabilistically infer. Predominantly, our understanding of RNN function comes from the identification of fixed points and their local linearized dynamics [41], yet these analyses do not afford a direct link between macroscopic connectivity parameters and dynamics. Alternatively, we use EPI to characterize the parameteric sources of solution bias in an RNN solving a toy mathematical problem.

Here, the task we consider is Gaussian posterior conditioning: calculate the parameters of a Gaussian posterior distribution on the mean of a Gaussian likelihood μ_y , given a single observation of $y \sim \mathcal{N}(\mu_y, \sigma_y^2 = 1)$ and a prior $p(\mu_y) = \mathcal{N}(\mu_0 = 4, \sigma_0^2 = 1)$ (Fig. 5A). Conjugacy in the Gaussian likelihood and prior result in ground truth calculation for the Gaussian posterior mean

$$\mu_{\text{post}} = \frac{\frac{\mu_0}{\sigma_0^2} + \frac{y}{\sigma_y^2}}{\frac{1}{\sigma_0^2} + \frac{1}{\sigma_y^2}} \quad (9)$$

303 and posterior variance

$$\sigma_{\text{post}}^2 = \frac{1}{\frac{1}{\sigma_0^2} + \frac{1}{\sigma_y^2}} \quad (10)$$

304 First, we predicate that the RNN should produce activity along a readout vector w corresponding
 305 to its estimation of the posterior mean μ_{post} . Second, we ask that the RNN produce a degree
 306 of chaotic variability matching the posterior variance σ_{post}^2 . This problem setup is inspired by
 307 dynamical systems modeling of approximate inference in the brain [42]. Although, we avoid using
 308 the term “approximate inference” to describe this task, since we are demonstrating the utility of
 309 EPI, which is a different form of approximate inference (see Section A.1.5).

310 Drawing conclusions about the role tens of thousands of weight matrix parameters in producing a
 311 readout projection and chaotic variance is a daunting challenge. However, we can leverage recent
 312 theoretical work establishing a link between macroscopic parameterizations of RNN connectivity
 313 and the emerging dynamics [23]. Specifically, we consider an N -neuron, rank-1 RNN with connec-
 314 tivity

$$W = g\chi + \frac{1}{N}mn^\top, \quad (11)$$

315 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
 316 Gaussian distributions $m_i \sim \mathcal{N}(M_m, 1)$ and $n_i \sim \mathcal{N}(M_n, 1)$. This theory allows us to calculate the
 317 RNN response along a readout vector

$$\kappa_w = \frac{1}{N} \sum_{j=1}^N w_j \phi(x_j) \quad (12)$$

318 to a constant input $I(t) = yw + (n - M_n)$. Additionally, the amount of chaotic variance Δ_T can
 319 be expressed through consistency equations of dynamic mean field parameters, the solver of which
 320 we take gradients through (see Section A.2.4). This theory allows us to mathematically formalize
 321 the execution of this task into an emergent property, where the emergent property statistics of the
 322 RNN activity are k_w and Δ_T and the emergent property values are the ground truth μ_{post} and
 323 σ_{post}^2 :

$$E \begin{bmatrix} \kappa_w \\ \Delta_T \\ (\kappa_w - \mu_{\text{post}})^2 \\ (\Delta_T - \sigma_{\text{post}}^2) \end{bmatrix} = \begin{bmatrix} \mu_{\text{post}} \\ \sigma_{\text{post}}^2 \\ 0.1 \\ 0.1 \end{bmatrix} \quad (13)$$

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

327 We used EPI to learn distributions of RNNs executing Gaussian posterior conditioning given an
 328 input of $y = 2$. (see Section A.2.4) (Fig. 5B). The true Gaussian conditioning posterior for an input
 329 of $y = 2$ is $\mu_{\text{post}} = 3$ and $\sigma_{\text{post}} = 0.5$. We can examined the nature of the over- and under-estimation
 330 of the posterior means (Fig. 5B, left) and variances (Fig. 5B, right) in the inferred distributions.
 331 There is rough symmetry in the M_m - M_n plane, suggesting a degeneracy in the product of M_m and
 332 M_n (Fig. 5B). The product of M_m and M_n almost completely determines the posterior mean (Fig.
 333 5B, left), and the random strength g is the most influential variable on the temporal variance (Fig.
 334 5B, right). Neither of these observations were obvious from the consistency equations afforded by
 335 DMFT (see Section A.2.4).

336 While the theory used for emergent property statistic calculation is exact in the limit of infinite
 337 neurons [23]. is exact2,000-neuron realizations of drawn parameters z_1 and z_2 from the inferred
 338 distribution support these conclusions. z_1 has relatively high $M_m M_n$, and thusly produces an
 339 RNN overestimating the posterior mean, since mean activity $\mu(t) > 3$ (Fig. 5C, left cyan). In
 340 turn, z_2 , having relatively low $M_m M_n$, produces an RNN underestimates the posterior mean, since
 341 $\mu(t) < 3$ (Fig. 5C, right cyan). Finally, the evidently greater level of chaotic variance in RNNs
 342 with z_1 compared to z_2 make sense given that g is greater in z_1 than in z_2 . This novel procedure of
 343 doing inference in interpretable parameterizations of RNNs conditioned on the emergent property
 344 of task execution is straightforwardly generalizable to other tasks like noisy integration and context-
 345 dependent decision making (Fig. S1).

346 4 Discussion

347 4.1 EPI is a general tool for theoretical neuroscience

348 Models of biological systems are often comprised of complex nonlinear differential equations, mak-
 349 ing traditional theoretical analysis and statistical inference intractable. In contrast, EPI is capable
 350 of learning distributions of parameters in such models producing measurable signatures of compu-
 351 tation. We have demonstrated its utility on biological models (STG), intermediate-level models of
 352 interacting genetically- and functionally-defined neuron-types (V1, SC), and the most abstract of
 353 models (RNNs). We are able to condition both deterministic and stochastic models on low-level
 354 emergent properties like firing rates of membrane potentials, as well as high-level cognitive func-
 355 tion like Gaussian posterior conditioning. Technically, EPI is tractable when the emergent property
 356 statistics are continuously differentiable with respect to the model parameters, which is very often

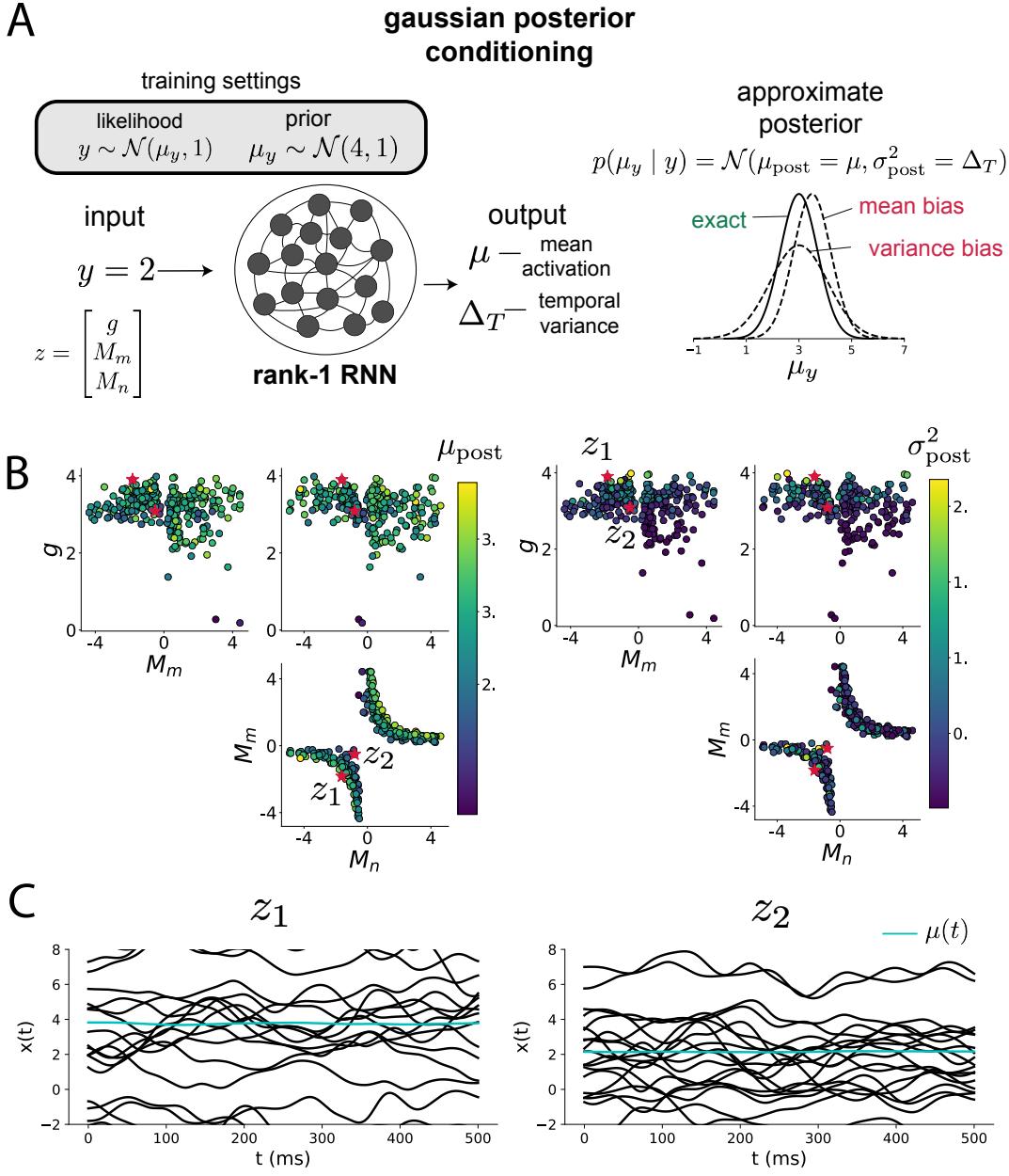


Figure 5: Sources of bias in RNN computation. A. (left) A rank-1 RNN running approximate Bayesian inference on μ_y assuming a gaussian likelihood variance of 1 and a prior of $\mathcal{N}(4, 1)$. (center) The rank-1 RNN represents the computed Gaussian posterior mean μ_{post} and variance σ_{post}^2 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.

357 the case; this emphasizes the general utility of EPI.

358 In this study, we have focused on applying EPI to low dimensional parameter spaces of models
359 with low dimensional dynamical state. These choices were made to present the reader with a series
360 of interpretable conclusions, which is more challenging in high dimensional spaces. In fact, EPI
361 should scale reasonably to high dimensional parameter spaces, as the underlying technology has
362 produced state-of-the-art performance on high-dimensional tasks such as texture generation [18].
363 Of course, increasing the dimensionality of the dynamical state of the model makes optimization
364 more expensive, and there is a practical limit there as with any machine learning approach. For
365 systems with high dimensional state, we recommend using theoretical approaches (e.g. [23]) to
366 reason about reduced parameterizations of such high-dimensional systems.

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

378 4.2 Novel hypotheses from EPI

379 Machine learning has played an effective, multifaceted role in neuroscientific progress. Primarily,
380 it has revealed structure in large-scale neural datasets [43, 44, 45, 46, 47, 48] (see review, [14]).
381 Secondarily, trained algorithms of varying degrees of biological relevance are beginning to be viewed
382 as fully-observable computational systems comparable to the brain [41, 49].

383 For example, consider the fact that we do not fully understand the four-dimensional models of V1
384 [21]. Because analytical approaches to studying nonlinear dynamical systems become increasingly
385 complicated when stepping from two-dimensional to three- or four-dimensional systems in the
386 absence of restrictive simplifying assumptions [50], it is unsurprising that this model has been a

challenge. In Section 3.3, we showed that EPI was far more informative about neuron-type input responsivity than the predictions afforded through analysis. By flexibly conditioning this V1 model on different emergent properties, we performed an exploratory analysis of a *model* rather than a dataset, which generated and proved out a set of testable predictions.

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

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

402 References

- 403 [1] Larry F Abbott. Theoretical neuroscience rising. *Neuron*, 60(3):489–495, 2008.
- 404 [2] John J Hopfield. Neural networks and physical systems with emergent collective computational
405 abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558, 1982.
- 406 [3] Haim Sompolinsky, Andrea Crisanti, and Hans-Jurgen Sommers. Chaos in random neural
407 networks. *Physical review letters*, 61(3):259, 1988.
- 408 [4] Misha V Tsodyks, William E Skaggs, Terrence J Sejnowski, and Bruce L McNaughton. Para-
409 doxical effects of external modulation of inhibitory interneurons. *Journal of neuroscience*,
410 17(11):4382–4388, 1997.
- 411 [5] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *International Confer-
412 ence on Learning Representations*, 2014.

- 413 [6] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation
414 and variational inference in deep latent gaussian models. *International Conference on Machine*
415 *Learning*, 2014.
- 416 [7] Yuanjun Gao, Evan W Archer, Liam Paninski, and John P Cunningham. Linear dynamical
417 neural population models through nonlinear embeddings. In *Advances in neural information*
418 *processing systems*, pages 163–171, 2016.
- 419 [8] Yuan Zhao and Il Memming Park. Recursive variational bayesian dual estimation for nonlinear
420 dynamics and non-gaussian observations. *stat*, 1050:27, 2017.
- 421 [9] Gabriel Barello, Adam Charles, and Jonathan Pillow. Sparse-coding variational auto-encoders.
422 *bioRxiv*, page 399246, 2018.
- 423 [10] Chethan Pandarinath, Daniel J O’Shea, Jasmine Collins, Rafal Jozefowicz, Sergey D Stavisky,
424 Jonathan C Kao, Eric M Trautmann, Matthew T Kaufman, Stephen I Ryu, Leigh R Hochberg,
425 et al. Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature*
426 *methods*, page 1, 2018.
- 427 [11] Alexander B Wiltschko, Matthew J Johnson, Giuliano Iurilli, Ralph E Peterson, Jesse M
428 Katon, Stan L Pashkovski, Victoria E Abraira, Ryan P Adams, and Sandeep Robert Datta.
429 Mapping sub-second structure in mouse behavior. *Neuron*, 88(6):1121–1135, 2015.
- 430 [12] Matthew J Johnson, David K Duvenaud, Alex Wiltschko, Ryan P Adams, and Sandeep R
431 Datta. Composing graphical models with neural networks for structured representations and
432 fast inference. In *Advances in neural information processing systems*, pages 2946–2954, 2016.
- 433 [13] Eleanor Batty, Matthew Whiteway, Shreya Saxena, Dan Biderman, Taiga Abe, Simon Musall,
434 Winthrop Gillis, Jeffrey Markowitz, Anne Churchland, John Cunningham, et al. Behavenet:
435 nonlinear embedding and bayesian neural decoding of behavioral videos. *Advances in Neural*
436 *Information Processing Systems*, 2019.
- 437 [14] Liam Paninski and John P Cunningham. Neural data science: accelerating the experiment-
438 analysis-theory cycle in large-scale neuroscience. *Current opinion in neurobiology*, 50:232–241,
439 2018.
- 440 [15] Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows.
441 *International Conference on Machine Learning*, 2015.

- 442 [16] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp.
443 *arXiv preprint arXiv:1605.08803*, 2016.
- 444 [17] George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density
445 estimation. In *Advances in Neural Information Processing Systems*, pages 2338–2347, 2017.
- 446 [18] Gabriel Loaiza-Ganem, Yuanjun Gao, and John P Cunningham. Maximum entropy flow
447 networks. *International Conference on Learning Representations*, 2017.
- 448 [19] Dustin Tran, Rajesh Ranganath, and David Blei. Hierarchical implicit models and likelihood-
449 free variational inference. In *Advances in Neural Information Processing Systems*, pages 5523–
450 5533, 2017.
- 451 [20] Gabrielle J Gutierrez, Timothy O’Leary, and Eve Marder. Multiple mechanisms switch an
452 electrically coupled, synaptically inhibited neuron between competing rhythmic oscillators.
453 *Neuron*, 77(5):845–858, 2013.
- 454 [21] Ashok Litwin-Kumar, Robert Rosenbaum, and Brent Doiron. Inhibitory stabilization and vi-
455 sual coding in cortical circuits with multiple interneuron subtypes. *Journal of neurophysiology*,
456 115(3):1399–1409, 2016.
- 457 [22] Chunyu A Duan, Marino Pagan, Alex T Piet, Charles D Kopec, Athena Akrami, Alexander J
458 Riordan, Jeffrey C Erlich, and Carlos D Brody. Collicular circuits for flexible sensorimotor
459 routing. *bioRxiv*, page 245613, 2018.
- 460 [23] Francesca Mastrogiovanni and Srdjan Ostojic. Linking connectivity, dynamics, and computa-
461 tions in low-rank recurrent neural networks. *Neuron*, 99(3):609–623, 2018.
- 462 [24] Sean R Bittner, Agostina Palmigiano, Kenneth D Miller, and John P Cunningham. Degener-
463 ate solution networks for theoretical neuroscience. *Computational and Systems Neuroscience
464 Meeting (COSYNE), Lisbon, Portugal*, 2019.
- 465 [25] Sean R Bittner, Alex T Piet, Chunyu A Duan, Agostina Palmigiano, Kenneth D Miller,
466 Carlos D Brody, and John P Cunningham. Examining models in theoretical neuroscience with
467 degenerate solution networks. *Bernstein Conference*, 2019.
- 468 [26] Jan-Matthis Lueckmann, Pedro Goncalves, Chaitanya Chintaluri, William F Podlaski, Gia-
469 como Bassetto, Tim P Vogels, and Jakob H Macke. Amortised inference for mechanistic models

- 470 of neural dynamics. In *Computational and Systems Neuroscience Meeting (COSYNE), Lisbon, Portugal*, 2019.
- 471
- 472 [27] Eve Marder and Vatsala Thirumalai. Cellular, synaptic and network effects of neuromodulation. *Neural Networks*, 15(4-6):479–493, 2002.
- 473
- 474 [28] Astrid A Prinz, Dirk Bucher, and Eve Marder. Similar network activity from disparate circuit parameters. *Nature neuroscience*, 7(12):1345, 2004.
- 475
- 476 [29] Edwin T Jaynes. Information theory and statistical mechanics. *Physical review*, 106(4):620, 1957.
- 477
- 478 [30] Gamaleldin F Elsayed and John P Cunningham. Structure in neural population recordings: an expected byproduct of simpler phenomena? *Nature neuroscience*, 20(9):1310, 2017.
- 479
- 480 [31] Cristina Savin and Gašper Tkačik. Maximum entropy models as a tool for building precise neural controls. *Current opinion in neurobiology*, 46:120–126, 2017.
- 481
- 482 [32] Brendan K Murphy and Kenneth D Miller. Balanced amplification: a new mechanism of selective amplification of neural activity patterns. *Neuron*, 61(4):635–648, 2009.
- 483
- 484 [33] Hirofumi Ozeki, Ian M Finn, Evan S Schaffer, Kenneth D Miller, and David Ferster. Inhibitory stabilization of the cortical network underlies visual surround suppression. *Neuron*, 62(4):578–592, 2009.
- 485
- 486
- 487 [34] Daniel B Rubin, Stephen D Van Hooser, and Kenneth D Miller. The stabilized supralinear network: a unifying circuit motif underlying multi-input integration in sensory cortex. *Neuron*, 85(2):402–417, 2015.
- 488
- 489
- 490 [35] Henry Markram, Maria Toledo-Rodriguez, Yun Wang, Anirudh Gupta, Gilad Silberberg, and Caizhi Wu. Interneurons of the neocortical inhibitory system. *Nature reviews neuroscience*, 5(10):793, 2004.
- 491
- 492
- 493 [36] Bernardo Rudy, Gordon Fishell, SooHyun Lee, and Jens Hjerling-Leffler. Three groups of interneurons account for nearly 100% of neocortical gabaergic neurons. *Developmental neurobiology*, 71(1):45–61, 2011.
- 494
- 495
- 496 [37] (2018) Allen Institute for Brain Science. Layer 4 model of v1. available from:
- 497 <https://portal.brain-map.org/explore/models/l4-mv1>.

- 498 [38] Yazan N Billeh, Binghuang Cai, Sergey L Gratiy, Kael Dai, Ramakrishnan Iyer, Nathan W
499 Gouwens, Reza Abbasi-Asl, Xiaoxuan Jia, Joshua H Siegle, Shawn R Olsen, et al. Systematic
500 integration of structural and functional data into multi-scale models of mouse primary visual
501 cortex. *bioRxiv*, page 662189, 2019.
- 502 [39] Chunyu A Duan, Jeffrey C Erlich, and Carlos D Brody. Requirement of prefrontal and midbrain
503 regions for rapid executive control of behavior in the rat. *Neuron*, 86(6):1491–1503, 2015.
- 504 [40] Omri Barak. Recurrent neural networks as versatile tools of neuroscience research. *Current*
505 *opinion in neurobiology*, 46:1–6, 2017.
- 506 [41] David Sussillo and Omri Barak. Opening the black box: low-dimensional dynamics in high-
507 dimensional recurrent neural networks. *Neural computation*, 25(3):626–649, 2013.
- 508 [42] Rodrigo Echeveste, Laurence Aitchison, Guillaume Hennequin, and Máté Lengyel. Cortical-like
509 dynamics in recurrent circuits optimized for sampling-based probabilistic inference. *bioRxiv*,
510 page 696088, 2019.
- 511 [43] Robert E Kass and Valérie Ventura. A spike-train probability model. *Neural computation*,
512 13(8):1713–1720, 2001.
- 513 [44] Emery N Brown, Loren M Frank, Dengda Tang, Michael C Quirk, and Matthew A Wilson.
514 A statistical paradigm for neural spike train decoding applied to position prediction from
515 ensemble firing patterns of rat hippocampal place cells. *Journal of Neuroscience*, 18(18):7411–
516 7425, 1998.
- 517 [45] Liam Paninski. Maximum likelihood estimation of cascade point-process neural encoding
518 models. *Network: Computation in Neural Systems*, 15(4):243–262, 2004.
- 519 [46] M Yu Byron, John P Cunningham, Gopal Santhanam, Stephen I Ryu, Krishna V Shenoy, and
520 Maneesh Sahani. Gaussian-process factor analysis for low-dimensional single-trial analysis
521 of neural population activity. In *Advances in neural information processing systems*, pages
522 1881–1888, 2009.
- 523 [47] Kenneth W Latimer, Jacob L Yates, Miriam LR Meister, Alexander C Huk, and Jonathan W
524 Pillow. Single-trial spike trains in parietal cortex reveal discrete steps during decision-making.
525 *Science*, 349(6244):184–187, 2015.

- 526 [48] Lea Duncker, Gergo Bohner, Julien Boussard, and Maneesh Sahani. Learning interpretable
 527 continuous-time models of latent stochastic dynamical systems. *Proceedings of the 36th Interna-*
 528 *tional Conference on Machine Learning*, 2019.
- 529 [49] Blake A Richards and et al. A deep learning framework for neuroscience. *Nature Neuroscience*,
 530 2019.
- 531 [50] Steven H Strogatz. Nonlinear dynamics and chaos: with applications to physics. *Biology,*
 532 *Chemistry, and Engineering (Studies in Nonlinearity)*, Perseus, Cambridge, UK, 1994.
- 533 [51] Rajesh Ranganath, Sean Gerrish, and David Blei. Black box variational inference. In *Artificial*
 534 *Intelligence and Statistics*, pages 814–822, 2014.
- 535 [52] Martin J Wainwright, Michael I Jordan, et al. Graphical models, exponential families, and
 536 variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.
- 537 [53] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp.
 538 *Proceedings of the 5th International Conference on Learning Representations*, 2017.
- 539 [54] Carsten K Pfeffer, Mingshan Xue, Miao He, Z Josh Huang, and Massimo Scanziani. Inhi-
 540 bition of inhibition in visual cortex: the logic of connections between molecularly distinct
 541 interneurons. *Nature Neuroscience*, 16(8):1068, 2013.

542 **A Methods**

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

544 Emergent property inference (EPI) learns distributions of theoretical model parameters that pro-
 545 duce emergent properties of interest. EPI combines ideas from likelihood-free variational inference
 546 [19] and maximum entropy flow networks [18]. A maximum entropy flow network is used as a deep
 547 probability distribution for the parameters, while these samples often parameterize a differentiable
 548 model simulator, which may lack a tractable likelihood function.

549 Consider model parameterization z and data x generated from some theoretical model simulator
 550 represented as $p(x | z)$, which may be deterministic or stochastic. Theoretical models usually have
 551 known sampling procedures for simulating activity given a circuit parameterization, yet often lack
 552 an explicit likelihood function due to the nonlinearities and dynamics. With EPI, a distribution

553 on parameters z is learned, that yields an emergent property of interest \mathcal{B} ,

$$\mathcal{B} \leftrightarrow E_{z \sim q_\theta} [E_{x \sim p(x|z)} [T(x)]] = \mu \quad (14)$$

554 by making an approximation $q_\theta(z)$ to $p(z | \mathcal{B})$ (see Section A.1.5). So, over the DSN distribution
 555 $q_\theta(z)$ of model $p(x | z)$ for behavior \mathcal{B} , the emergent properties $T(x)$ are constrained in expectation
 556 to μ .

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

562 In most settings (especially those relevant to theoretical neuroscience) the likelihood of the behavior
 563 with respect to the model parameters $p(T(x) | z)$ is unknown or intractable, requiring an alternative
 564 to stochastic gradient variational Bayes [5] or black box variational inference[51]. These types
 565 of methods called likelihood-free variational inference (LFVI, [19]) skate around the intractable
 566 likelihood function in situations where there is a differentiable simulator. Akin to LFVI, DSNs are
 567 optimized with the following objective for a given theoretical model, emergent property statistics
 568 $T(x)$, and emergent property constraints μ :

$$\begin{aligned} q_\theta^*(z) &= \underset{q_\theta \in Q}{\operatorname{argmax}} H(q_\theta(z)) \\ \text{s.t. } E_{z \sim q_\theta} [E_{x \sim p(x|z)} [T(x)]] &= \mu \end{aligned} \quad (15)$$

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

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

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

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

574 that produce a band of oscillations. To do EPI with the dynamics matrix elements as the free
 575 parameters $z = [a_1, a_2, a_3, a_4]$, and fixing $\tau = 1$, such that the posterior yields a band of oscillations,

576 the emergent property statistics $T(x)$ are chosen to contain the first- and second-moments of the
 577 oscillatory frequency Ω and the growth/decay factor d of the oscillating system. To learn the
 578 distribution of real entries of A that yield a distribution of d with mean zero with variance 0.25^2 ,
 579 and oscillation frequency Ω with mean 1 Hz with variance $(0.1\text{Hz})^2$, then we would select the real
 580 part of the complex conjugate eigenvalues $\text{real}(\lambda_1) = d$ (via an arbitrary choice of eigenvalue of the
 581 dynamics matrix λ_1) and the positive imaginary component of one of the eigenvalues $\text{imag}(\lambda_1) =$
 582 $2\pi\Omega$ as the emergent property statistics. Those emergent property statistics are then constrained
 583 to

$$\mu = E \begin{bmatrix} \text{real}(\lambda_1) \\ \text{imag}(\lambda_1) \\ (\text{real}(\lambda_1) - 0)^2 \\ (\text{imag}(\lambda_1) - 2\pi\Omega)^2 \end{bmatrix} = \begin{bmatrix} 0.0 \\ 2\pi\Omega \\ 0.25^2 \\ (2\pi 0.1)^2 \end{bmatrix} \quad (16)$$

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

$$\lambda = \frac{\left(\frac{a_1+a_4}{\tau}\right) \pm \sqrt{\left(\frac{a_1+a_4}{\tau}\right)^2 + 4\left(\frac{a_2a_3-a_1a_4}{\tau}\right)}}{2} \quad (17)$$

587 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
 588 directly via a closed form function and does not require simulation, we cannot derive the distribution
 589 q_θ^* directly. This is due to the formally hard problem of the backward mapping: finding the natural
 590 parameters η from the mean parameters μ of an exponential family distribution [52]. Instead, we
 591 can use EPI to learn the linear system parameters producing such a band of oscillations (Fig. S2B).

592 Even this relatively simple system has nontrivial (though intuitively sensible) structure in the
 593 parameter distribution. To validate our method (further than that of the underlying technology
 594 on a ground truth solution [18]) we can analytically derive the contours of the probability density
 595 from the emergent property statistics and values (Fig. S3). In the $a_1 - a_4$ plane, is a black line
 596 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
 597 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
 598 set of solutions at fixed behaviors, which overlay the posterior obtained through EPI. The learned
 599 DSN distribution precisely reflects the desired statistical constraints and model degeneracy in the
 600 sum of a_1 and a_4 . Intuitively, the parameters equivalent with respect to emergent property statistic
 601 $\text{real}(\lambda_1)$ have similar log densities.



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.

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

$$\text{imag}(\lambda_1) = \begin{cases} \sqrt{\frac{a_1a_4-a_2a_3}{\tau}}, & \text{if } a_1a_4 < a_2a_3 \\ 0 & \text{otherwise} \end{cases} \quad (18)$$

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

$$\text{imag}(\lambda_1)^2 = a_1a_4 - a_2a_3 \quad (19)$$

606 Since we constrained $E_{q_\theta}[\text{imag}(\lambda)] = 2\pi$ (with $\omega = 1$), we can plot contours of the equation
 607 $\text{imag}(\lambda_1)^2 = a_1a_4 - a_2a_3 = (2\pi)^2$ for various a_1a_4 (Fig. S3A). If $\sigma_{1,4} = E_{q_\theta}(|a_1a_4 - E_{q_\theta}[a_1a_4]|)$,
 608 then we plot the contours as $a_1a_4 = 0$ (black), $a_1a_4 = -\sigma_{1,4}$ (black dotted), and $a_1a_4 = -2\sigma_{1,4}$
 609 (grey dotted) (Fig. S3B). This validates the curved structure of the inferred distribution learned
 610 through EPI. We take steps in negative standard deviation of a_1a_4 (dotted and gray lines), since
 611 there are few positive values a_1a_4 in the posterior. Subtler model-behavior combinations will have
 612 even more complexity, further motivating the use of EPI for understanding these systems. Indeed,
 613 we sample a distribution of systems oscillating near 1Hz (Fig. S4).

614 A.1.2 Augmented Lagrangian optimization

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

$$L(\theta; \alpha, c) = -H(q_\theta) + \alpha^\top \delta(\theta) + \frac{c}{2} \|\delta(\theta)\|^2 \quad (20)$$

617 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
 618 coefficient. For a fixed (α, c) , θ is optimized with stochastic gradient descent. A low value of c is
 619 used initially, and increased during each augmented Lagrangian epoch – a period of optimization
 620 with fixed α and c for a given number of stochastic optimization iterations. Similarly, α is tuned
 621 each epoch based on the constraint violations. For the linear 2-dimensional system (Fig. S2C)
 622 optimization hyperparameters are initialized to $c_1 = 10^{-4}$ and $\alpha_1 = 0$. The penalty coefficient
 623 is updated based on a hypothesis test regarding the reduction in constraint violation. The p-
 624 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

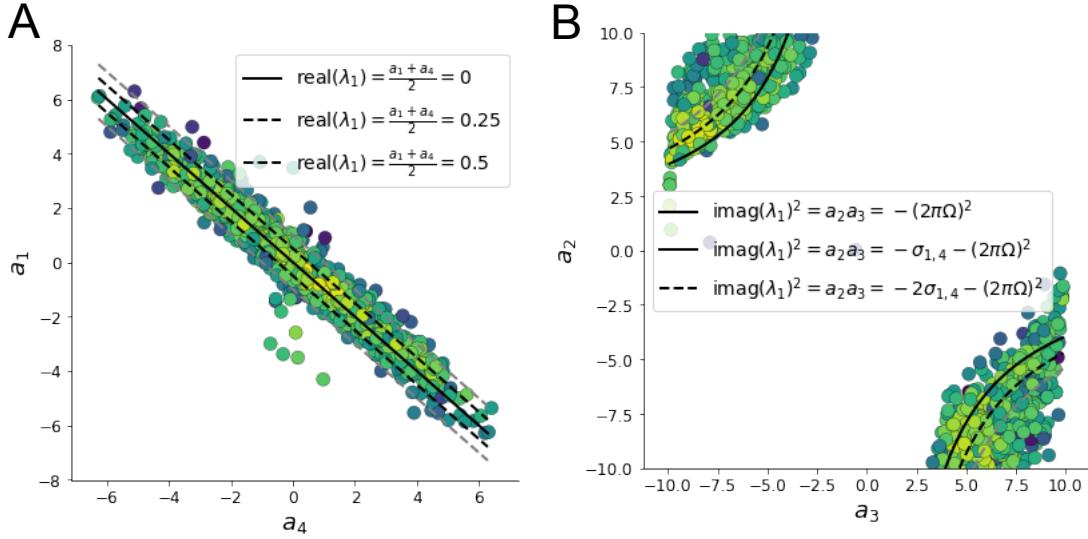


Fig. S3: A. Probability contours in the $a_1 - a_4$ plane can be derived from the relationship to emergent property statistic of growth/decay factor. B. Probability contours in the $a_2 - a_3$ plane can be derived from relationship to the emergent property statistic of oscillation frequency.

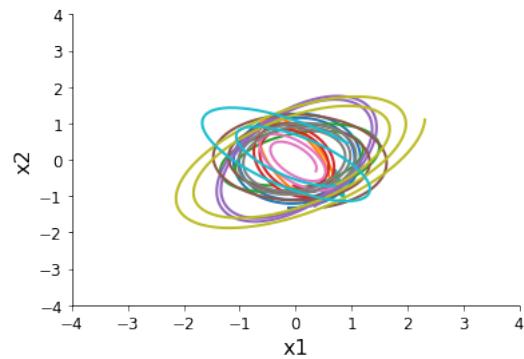


Fig. S4: Sampled dynamical system trajectories from the EPI distribution. Each trajectory is initialized at $x(0) = \left[\frac{\sqrt{2}}{2} \quad -\frac{\sqrt{2}}{2} \right]$.

625 $1 - p$. Throughout the project, $\beta = 4.0$ and $\gamma = 0.25$ is used. The other update rule is $\alpha_{k+1} =$
 626 $\alpha_k + c_k \frac{1}{n} \sum_{i=1}^n (T(x^{(i)}) - \mu)$. In this example, each augmented Lagrangian epoch ran for 2,000
 627 iterations. We consider the optimization to have converged when a null hypothesis test of constraint
 628 violations being zero is accepted for all constraints at a significance threshold 0.05. This is the dotted
 629 line on the plots below depicting the optimization cutoff of EPI optimization for the 2-dimensional
 630 linear system. If the optimization is left to continue running, entropy usually decreases, and
 631 structural pathologies in the distribution may be introduced.

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

638 A.1.3 Normalizing flows

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

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

$$\nabla_\theta H(q_\theta(z)) = E_{\omega \sim q_0} [-\nabla_\theta \log(q_\theta(f_\theta(\omega)))] \quad (22)$$

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

$$q(z') = q(f^{-1}(z')) \left| \det \frac{\partial f^{-1}(z')}{\partial z'} \right| = q(z) \left| \det \frac{\partial f(z)}{\partial z} \right|^{-1} \quad (23)$$

646 However, this computation has cubic complexity in dimensionality for fully connected layers. By
 647 restricting our layers to normalizing flows [15] – bijective functions with fast log determinant ja-
 648 cobian computations, we can tractably optimize deep generative models with objectives that are a
 649 function of sample density, like entropy. Most of our analyses use real NVP [53], which have proven

650 effective in our architecture searches, and have the advantageous features of fast sampling and fast
 651 density evaluation.

652 **A.1.4 Related work**

653 (To come)

654

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

656 (To come)

657

658 **A.2 Theoretical models**

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

661 **A.2.1 Stomatogastric ganglion**

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

$$C_m \frac{dx_m}{dt} = -[h_{leak}(x; z) + h_{Ca}(x; z) + h_K(x; z) + h_{hyp}(x; z) + h_{elec}(x; z) + h_{syn}(x; z)] \quad (24)$$

663 The membrane potential of each neuron is affected by the leak, calcium, potassium, hyperpolariza-
 664 tion, electrical and synaptic currents, respectively. The capacitance of the cell membrane was set to
 665 $C_m = 1nF$. Each current is a function of the neuron's membrane potential x_m and the parameters
 666 of the circuit such as g_{el} and g_{syn} , whose effect on the circuit is considered in the motivational
 667 example of EPI in Fig. 1. Specifically, the currents are the difference in the neuron's membrane
 668 potential and that current type's reversal potential multiplied by a conductance:

$$h_{leak}(x; z) = g_{leak}(x_m - V_{leak}) \quad (25)$$

669

$$h_{elec}(x; z) = g_{el}(x_m^{post} - x_m^{pre}) \quad (26)$$

670

$$h_{syn}(x; z) = g_{syn}S_\infty^{pre}(x_m^{post} - V_{syn}) \quad (27)$$

671

$$h_{Ca}(x; z) = g_{Ca}M_\infty(x_m - V_{Ca}) \quad (28)$$

672

$$h_K(x; z) = g_K N(x_m - V_K) \quad (29)$$

673

$$h_{hyp}(x; z) = g_h H(x_m - V_{hyp}) \quad (30)$$

- 674 The reversal potentials were set to $V_{leak} = -40mV$, $V_{Ca} = 100mV$, $V_K = -80mV$, $V_{hyp} = -20mV$,
 675 and $V_{syn} = -75mV$. The other conductance parameters were fixed to $g_{leak} = 1 \times 10^{-4}\mu S$. g_{Ca} ,
 676 g_K , and g_{hyp} had different values based on fast, intermediate (hub) or slow neuron. Fast: $g_{Ca} =$
 677 1.9×10^{-2} , $g_K = 3.9 \times 10^{-2}$, and $g_{hyp} = 2.5 \times 10^{-2}$. Intermediate: $g_{Ca} = 1.7 \times 10^{-2}$, $g_K = 1.9 \times 10^{-2}$,
 678 and $g_{hyp} = 8.0 \times 10^{-3}$. Intermediate: $g_{Ca} = 8.5 \times 10^{-3}$, $g_K = 1.5 \times 10^{-2}$, and $g_{hyp} = 1.0 \times 10^{-2}$.
- 679 Furthermore, the Calcium, Potassium, and hyperpolarization channels have time-dependent gating
 680 dynamics dependent on steady-state gating variables M_∞ , N_∞ and H_∞ , respectively.

$$M_\infty = 0.5 \left(1 + \tanh \left(\frac{x_m - v_1}{v_2} \right) \right) \quad (31)$$

681

$$\frac{dN}{dt} = \lambda_N(N_\infty - N) \quad (32)$$

682

$$N_\infty = 0.5 \left(1 + \tanh \left(\frac{x_m - v_3}{v_4} \right) \right) \quad (33)$$

683

$$\lambda_N = \phi_N \cosh \left(\frac{x_m - v_3}{2v_4} \right) \quad (34)$$

684

$$\frac{dH}{dt} = \frac{(H_\infty - H)}{\tau_h} \quad (35)$$

685

$$H_\infty = \frac{1}{1 + \exp \left(\frac{x_m + v_5}{v_6} \right)} \quad (36)$$

686

$$\tau_h = 272 - \left(\frac{-1499}{1 + \exp \left(\frac{-x_m + v_7}{v_8} \right)} \right) \quad (37)$$

- 687 where we set $v_1 = 0mV$, $v_2 = 20mV$, $v_3 = 0mV$, $v_4 = 15mV$, $v_5 = 78.3mV$, $v_6 = 10.5mV$,
 688 $v_7 = -42.2mV$, $v_8 = 87.3mV$, $v_9 = 5mV$, and $v_{th} = -25mV$. These are the same parameter
 689 values used in [20].

- 690 Finally, there is a synaptic gating variable as well:

$$S_\infty = \frac{1}{1 + \exp \left(\frac{v_{th} - x_m}{v_9} \right)} \quad (38)$$

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

693 In order to measure the frequency of the hub neuron during EPI, the STG model was simulated
 694 for $T = 500$ time steps of $dt = 25ms$. In EPI, since gradients are taken through the simulation
 695 process, the number of time steps are kept as modest if possible. The chosen dt and T were the
 696 most computationally convenient choices yielding accurate frequency measurement.

697 Our original approach to measuring frequency was to take the max of the fast Fourier transform
 698 (FFT) of the simulated time series. There are a few key considerations here. One is resolution
 699 in frequency space. Each FFT entry will correspond to a signal frequency of $\frac{F_s k}{N}$, where N is
 700 the number of samples used for the FFT, $F_s = \frac{1}{dt}$, and $k \in [0, 1, \dots, N - 1]$. Our resolution is
 701 improved by increasing N and decreasing dt . Increasing $N = T - b$, where b is some fixed number
 702 of buffer burn-in initialization samples, necessitates an increase in simulation time steps T , which
 703 directly increases computational cost. Increasing F_s (decreasing dt) increases system approximation
 704 accuracy, but requires more time steps before a full cycle is observed. At the level of $dt = 0.025$,
 705 thousands of temporal samples were required for resolution of .01Hz. These challenges in frequency
 706 resolution with the discrete Fourier transform motivated the use of an alternative basis of complex
 707 exponentials. Instead, we used a basis of complex exponentials with frequencies from 0.0-1.0 Hz at
 708 0.01Hz resolution, $\Phi = [0.0, 0.01, \dots, 1.0]^\top$

709 Another consideration was that the frequency spectra of the hub neuron has several peaks. This
 710 was due to high-frequency sub-threshold activity. The maximum frequency was often not the firing
 711 frequency. Accordingly, subthreshold activity was set to zero, and the whole signal was low-pass
 712 filtered with a moving average window of length 20. The signal was subsequently mean centered.
 713 After this pre-processing, the maximum frequency in the filter bank accurately reflected the firing
 714 frequency.

715 Finally, to differentiate through the maximum frequency identification step, we used a sum-of-
 716 powers normalization strategy: Let $\mathcal{X}_i \in \mathcal{C}^{|\Phi|}$ be the complex exponential filter bank dot products
 717 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

$$u_i = \frac{|\mathcal{X}_i|^\alpha}{\sum_{k=1}^N |\mathcal{X}_i(k)|^\alpha} \quad (39)$$

718 The frequency is then calculated as $\Omega_i = u_i^\top \Phi$ with $\alpha = 100$.
 719 Network syncing, like all other emergent properties in this work, are defined by the emergent
 720 property statistics and values. The emergent property statistics are the first- and second-moments
 721 of the firing frequencies. The first moments are set to 0.55Hz, while the second moments are set to

722 0.025Hz².

$$E \begin{bmatrix} \Omega_{f1} \\ \Omega_{f2} \\ \Omega_{hub} \\ \Omega_{s1} \\ \Omega_{s2} \\ (\Omega_{f1} - 0.55)^2 \\ (\Omega_{f2} - 0.55)^2 \\ (\Omega_{hub} - 0.55)^2 \\ (\Omega_{s1} - 0.55)^2 \\ (\Omega_{s2} - 0.55)^2 \end{bmatrix} = \begin{bmatrix} 0.55 \\ 0.55 \\ 0.55 \\ 0.55 \\ 0.55 \\ 0.025^2 \\ 0.025^2 \\ 0.025^2 \\ 0.025^2 \\ 0.025^2 \end{bmatrix} \quad (40)$$

723 For EPI in Fig 2C, we used a real NVP architecture with two coupling layers. Each coupling layer
 724 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
 725 have shown the EPI optimization that converged with maximum entropy across 2 random seeds
 726 and augmented Lagrangian coefficient initializations of $c_0=0, 2$, and 5.

727 **A.2.2 Primary visual cortex**

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

$$\tau \frac{dx}{dt} = -x + [Wx + h]_+^n \quad (41)$$

729 Some neuron-types largely lack synaptic projections to other neuron-types [54], and it is popular
 730 to only consider a subset of the effective connectivities [21].

$$W = \begin{bmatrix} W_{EE} & W_{EP} & W_{ES} & 0 \\ W_{PE} & W_{PP} & W_{PS} & 0 \\ W_{SE} & 0 & 0 & W_{SV} \\ W_{VE} & W_{VP} & W_{VS} & 0 \end{bmatrix} \quad (42)$$

731 By consolidating information from many experimental datasets, Billeh et al. [38] produce estimates

⁷³² of the synaptic strength (in mV)

$$M = \begin{bmatrix} 0.36 & 0.48 & 0.31 & 0.28 \\ 1.49 & 0.68 & 0.50 & 0.18 \\ 0.86 & 0.42 & 0.15 & 0.32 \\ 1.31 & 0.41 & 0.52 & 0.37 \end{bmatrix} \quad (43)$$

⁷³³ and connection probability

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

⁷³⁴ Multiplying these connection probabilities and synaptic efficacies gives us an effective connectivity
⁷³⁵ matrix:

$$W_{\text{full}} = C \odot M = \begin{bmatrix} 0.16 & 0.411 & 0.424 & 0.087 \\ 0.395 & .451 & 0.857 & 0.02 \\ 0.182 & 0.03 & 0.082 & 0.625 \\ 0.105 & 0.22 & 0.77 & 0.028 \end{bmatrix} \quad (45)$$

⁷³⁶ From use the entries of this full effective connectivity matrix that are not considered to be ineffec-
⁷³⁷ tual.

⁷³⁸ We look at how this four-dimensional nonlinear dynamical model of V1 responds to different inputs,
⁷³⁹ and compare the predictions of the linear response to the approximate posteriors obtained through
⁷⁴⁰ EPI. The input to the system is the sum of a baseline input $b = [1 \ 1 \ 1 \ 1]^T$ and a differential
⁷⁴¹ input dh :

$$h = b + dh \quad (46)$$

⁷⁴² All simulations of this system had $T = 100$ time points, a time step $dt = 5\text{ms}$, and time constant
⁷⁴³ $\tau = 20\text{ms}$. And the system was initialized to a random draw $x(0)_i \sim \mathcal{N}(1, 0.01)$.

⁷⁴⁴ We can describe the dynamics of this system more generally by

$$\dot{x}_i = -x_i + f(u_i) \quad (47)$$

⁷⁴⁵ where the input to each neuron is

$$u_i = \sum_j W_{ij} x_j + h_i \quad (48)$$

⁷⁴⁶ Let $F_{ij} = \gamma_i \delta(i, j)$, where $\gamma_i = f'(u_i)$. Then, the linear response is

$$\frac{dx_{ss}}{dh} = F(W \frac{dx_{ss}}{dh} + I) \quad (49)$$

⁷⁴⁷ which is calculable by

$$\frac{dx_{ss}}{dh} = (F^{-1} - W)^{-1} \quad (50)$$

⁷⁴⁸ The emergent property we considered was the first and second moments of the change in rate dx
⁷⁴⁹ between the baseline input $h = b$ and $h = b + dh$. We use the following notation to indicate that
⁷⁵⁰ the emergent property statistics were set to the following values:

$$\mathcal{B}(\alpha, y) \leftrightarrow E \begin{bmatrix} dx_{\alpha,ss} \\ (dx_{\alpha,ss} - y)^2 \end{bmatrix} = \begin{bmatrix} y \\ 0.01^2 \end{bmatrix} \quad (51)$$

⁷⁵¹ In the final analysis for this model, we sweep the input one neuron at a time away from the mode
⁷⁵² of each inferred distributions $dh^* = z^* = \text{argmax}_z \log q_\theta(z | \mathcal{B}(\alpha, 0.1))$. The differential responses
⁷⁵³ $dx_{\alpha,ss}$ are examined at perturbed inputs $h = b + dh^* + \Delta h_\alpha u_\alpha$ where u_α is a unit vector in the
⁷⁵⁴ dimension of α and $\Delta h_\alpha \in [-15, 15]$.

⁷⁵⁵ 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
⁷⁵⁶ initial seeds using an architecture of four coupling layers, each with two hidden layers of 10 units.
⁷⁵⁷ We set $c_0 = 10^5$. The support of the learned distribution was restricted to $z_i \in [-5, 5]$.

⁷⁵⁸ A.2.3 Superior colliculus

⁷⁵⁹ There are four total units: two in each hemisphere corresponding to the Pro/Contra and Anti/Ipsi
⁷⁶⁰ populations. Each unit has an activity (x_i) and internal variable (u_i) related by

$$x_i(t) = \left(\frac{1}{2} \tanh \left(\frac{v_i(t) - \epsilon}{\zeta} \right) + \frac{1}{2} \right) \quad (52)$$

⁷⁶¹ $\epsilon = 0.05$ and $\zeta = 0.5$ control the position and shape of the nonlinearity, repsectively.

⁷⁶² We can order the elements of x_i and v_i into vectors x and v with elements

$$x = \begin{bmatrix} x_{LP} \\ x_{LA} \\ x_{RP} \\ x_{RA} \end{bmatrix} \quad v = \begin{bmatrix} v_{LP} \\ v_{LA} \\ v_{RP} \\ v_{RA} \end{bmatrix} \quad (53)$$

763 The internal variables follow dynamics:

$$\tau \frac{dv}{dt} = -v + Wx + h + \sigma dB \quad (54)$$

764 with time constant $\tau = 0.09s$ and Gaussian noise σdB controlled by the magnitude of $\sigma = 1.0$. The
765 weight matrix has 8 parameters sW_P , sW_A , vW_{PA} , vW_{AP} , hW_P , hW_A , dW_{PA} , and dW_{AP} (Fig.
766 4B).

$$W = \begin{bmatrix} sW_P & vW_{PA} & hW_P & dW_{PA} \\ vW_{AP} & sW_A & dW_{AP} & hW_A \\ hW_P & dW_{PA} & sW_P & vW_{PA} \\ dW_{AP} & hW_A & vW_{AP} & sW_A \end{bmatrix} \quad (55)$$

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

$$h = h_{\text{rule}} + h_{\text{choice-period}} + h_{\text{light}} \quad (56)$$

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

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

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

770 a choice-period input,

$$h_{\text{choice}}(t) = \begin{cases} I_{\text{choice}} \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix}^\top, & \text{if } t > 1.2s \\ 0, & \text{otherwise} \end{cases} \quad (59)$$

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

$$h_{\text{light}}(t) = \begin{cases} I_{\text{light}} \begin{bmatrix} 1 & 1 & 0 & 0 \end{bmatrix}^\top, & \text{if } t > 1.2s \text{ and Left} \\ I_{\text{light}} \begin{bmatrix} 0 & 0 & 1 & 1 \end{bmatrix}^\top, & \text{if } t > 1.2s \text{ and Right} \\ 0, & t \leq 1.2s \end{cases} \quad (60)$$

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

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

$$\hat{p}_i = E_{\sigma dB} [x_{LP,ss} \mid s = L, c = P, z_i] = \frac{1}{M} \sum_{j=1}^M x_{LP,ss}(s = L, c = P, z_i, \sigma dB_j) \quad (61)$$

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

$$E_{z_i \sim q_\phi} [E_{\sigma dB} [x_{LP,ss} \mid s = L, c = P, z_i]] = E_{z_i \sim q_\phi} [\hat{p}_i] = p_{LP} \quad (62)$$

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

$$E_{z \sim q_\phi} [\sigma_{err}^2] = 0 \quad (63)$$

779

$$\sigma_{err}^2 = Var_{\sigma dB} [x_{LP,ss} \mid s = L, c = P, z_i] - \hat{p}_i(1 - \hat{p}_i) \quad (64)$$

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

$$E_{z \sim q_\phi} [d_P] = 1 \quad (65)$$

782

$$E_{\sigma dB} [(x_{LP,ss} - x_{RP,ss})^2 \mid s = L, c = P, z_i] \quad (66)$$

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

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

794 There is a pathology in this optimization setup, where the learned distribution of weights is bimodal
 795 attributing a fraction p of the samples to an expansive mode (which always sends x_{LP} to 1), and a
 796 fraction $1 - p$ to a decaying mode (which always sends x_{LP} to 0). This pathology was avoided using
 797 an inequality constraint prohibiting parameter samples that resulted in low variance of responses
 798 across noise.

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

$$\mathcal{B}(p) \leftrightarrow \begin{bmatrix} \hat{p}_P \\ \hat{p}_A \\ (\hat{p}_P - p)^2 \\ (\hat{p}_A - p)^2 \\ \sigma_{P,err}^2 \\ \sigma_{A,err}^2 \\ d_P \\ d_A \end{bmatrix} = \begin{bmatrix} p \\ p \\ 0.15^2 \\ 0.15^2 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} \quad (67)$$

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

802 **A.2.4 Rank-1 RNN**

803 Recent work establishes a link between RNN connectivity weights and the resulting dynamical
804 responses of the network, using dynamic mean field theory (DMFT) [23]. Specifically, DMFT
805 describes the properties of activity in infinite-size neural networks given a distribution on the
806 connectivity weights. In such a model, the connectivity of a rank-1 RNN (which was sufficient for
807 our task), has weight matrix W , which is the sum of a random component with strength determined
808 by g and a structured component determined by the outer product of vectors m and n :

$$W = g\chi + \frac{1}{N}mn^\top, \quad (68)$$

809 where the activity x evolves as and $I(t)$ is some input, ϕ is the tanh nonlinearity, and $\chi_{ij} \sim \mathcal{N}(0, \frac{1}{N})$.
810 The entries of m and n are drawn from Gaussian distributions $m_i \sim \mathcal{N}(M_m, 1)$ and $n_i \sim \mathcal{N}(M_n, 1)$.
811 From such a parameterization, this theory produces consistency equations for the dynamic mean
812 field variables in terms of parameters like g , M_m , and M_n , which we study in Section 3.5. That
813 is the dynamic mean field variables (e.g. the activity along a vector κ_v , the total variance
814 Δ_0 , structured variance Δ_∞ , and the chaotic variance Δ_T) are written as functions of one another
815 in terms of connectivity parameters. The values of these variables can be used obtained using a
816 nonlinear system of equations solver. These dynamic mean field variables are then cast as task-
817 relevant variables with respect to the context of the provided inputs. Mastrogiuseppe et al. designed
818 low-rank RNN connectivities via minimalist connectivity parameters to solve canonical tasks from
819 behavioral neuroscience.

820 We consider the DMFT equation solver as a black box that takes in a low-rank parameterization
 821 z (e.g. $z = [g, M_m, M_n]$) and outputs the values of the dynamic mean field variables, of which we
 822 cast κ_w and Δ_T as task-relevant variables μ_{post} and σ_{post}^2 in the Gaussian posterior conditioning
 823 toy example. Importantly, the solution produced by the solver is differentiable with respect to the
 824 input parameters, allowing us to use DMFT to calculate the emergent property statistics in EPI
 825 to learn distributions on such connectivity parameters of RNNs that execute tasks.

826 Specifically, we solve for the mean field variables κ_w , κ_n , Δ_0 and Δ_∞ , where the readout is nominally
 827 chosen to point in the unit orthant $w = \begin{bmatrix} 1 & \dots & 1 \end{bmatrix}^\top$. The consistency equations for these variables
 828 in the presence of a constant input $I(t) = y - (n - M_n)$ can be derived following [23] are

$$\begin{aligned} \kappa_w &= F(\kappa_w, \kappa_n, \Delta_0, \Delta_\infty) = M_m \kappa_n + y \\ \kappa_n &= G(\kappa_w, \kappa_n, \Delta_0, \Delta_\infty) = M_n \langle [\phi_i] \rangle + \langle [\phi'_i] \rangle \\ \frac{\Delta_0^2 - \Delta_\infty^2}{2} &= H(\kappa_w, \kappa_n, \Delta_0, \Delta_\infty) = g^2 \left(\int \mathcal{D}z \Phi^2(\kappa_w + \sqrt{\Delta_0} z) - \int \mathcal{D}z \int \mathcal{D}x \Phi(\kappa_w + \sqrt{\Delta_0 - \Delta_\infty} x + \sqrt{\Delta_\infty} z) \right) \\ &\quad + (\kappa_n^2 + 1)(\Delta_0 - \Delta_\infty) \\ \Delta_\infty &= L(\kappa_w, \kappa_n, \Delta_0, \Delta_\infty) = g^2 \int \mathcal{D}z \left[\int \mathcal{D}x \phi(\kappa_w + \sqrt{\Delta_0 - \Delta_\infty} x + \sqrt{\Delta_\infty} z) \right]^2 + \kappa_n^2 + 1 \end{aligned} \tag{69}$$

829 where z here is a gaussian integration variable. We can solve these equations by simulating the
 830 following Langevin dynamical system.

$$\begin{aligned} x(t) &= \frac{\Delta_0(t)^2 - \Delta_\infty(t)^2}{2} \\ \Delta_0(t) &= \sqrt{2x(t) + \Delta_\infty(t)^2} \\ \dot{\kappa}_w(t) &= -\kappa_w(t) + F(\kappa_w(t), \kappa_n(t), \Delta_0(t), \Delta_\infty(t)) \\ \dot{\kappa}_n(t) &= -\kappa_n + G(\kappa_w(t), \kappa_n(t), \Delta_0(t), \Delta_\infty(t)) \\ \dot{x}(t) &= -x(t) + H(\kappa_w(t), \kappa_n(t), \Delta_0(t), \Delta_\infty(t)) \\ \dot{\Delta}_\infty(t) &= -\Delta_\infty(t) + L(\kappa_w(t), \kappa_n(t), \Delta_0(t), \Delta_\infty(t)) \end{aligned} \tag{70}$$

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

$$\Delta_T = \Delta_0 - \Delta_\infty \tag{71}$$

833 A.3 Supplementary Figures

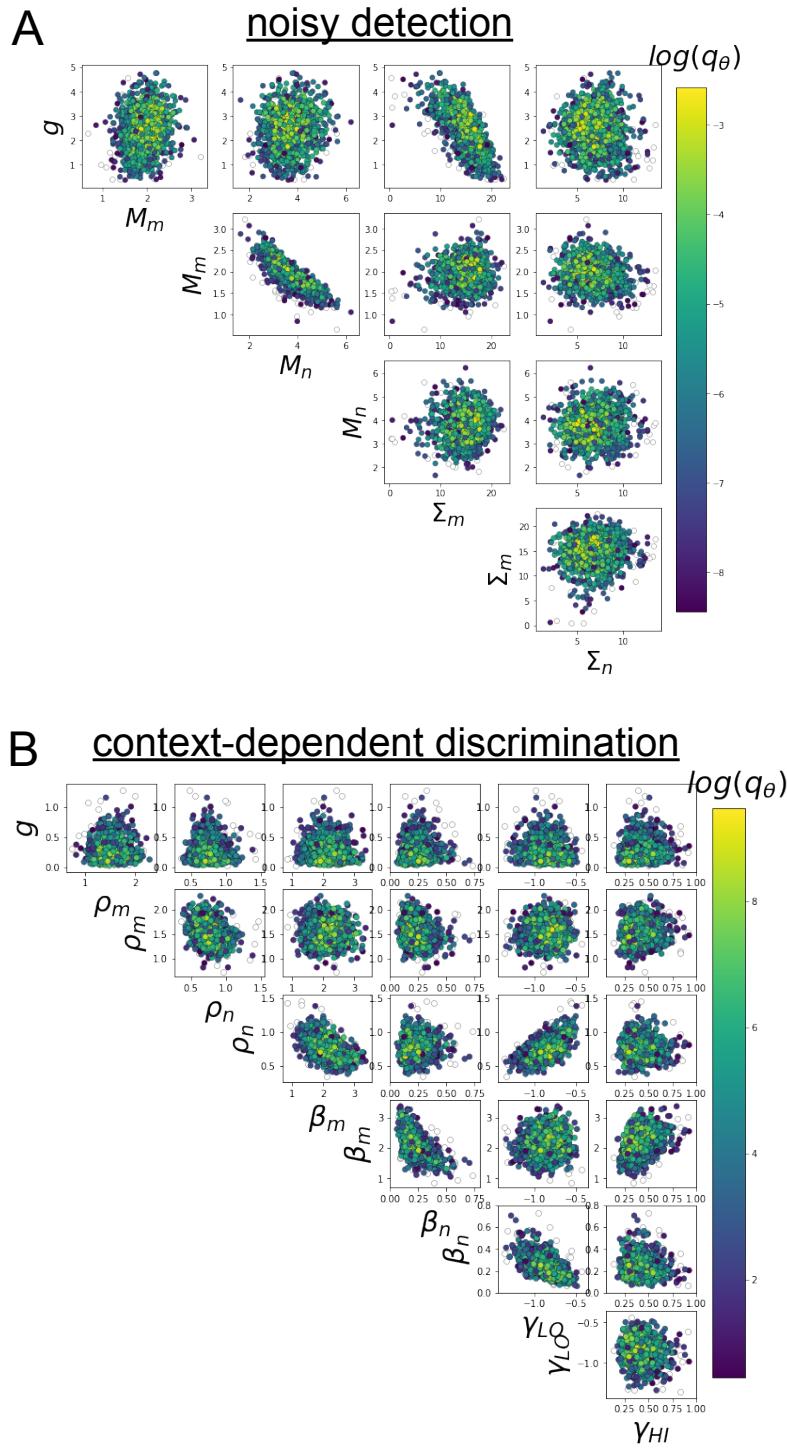


Fig. S1: A. EPI for rank-1 networks doing discrimination. B. EPI for rank-2 networks doing context-dependent discrimination.