

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. Such models are valuable when they give rise to an experimen-
⁴ tally observed phenomenon – whether behavioral or in terms of neural activity – and thus can offer
⁵ insights into neural computation. The operation of these circuits, like all models, critically depends
⁶ on the choices of model parameters. Historically, the gold standard has been to analytically derive
⁷ the relationship between model parameters and computational properties. However, this enterprise
⁸ quickly becomes infeasible as biologically realistic constraints are included into the model increas-
⁹ ing its complexity, 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
¹³ offer a principled means to understand the implications of model parameter choices on compu-
¹⁴ tational properties of interest. We motivate this methodology with a worked example analyzing
¹⁵ sensitivity in the stomatogastric ganglion. We then use it to generate insights into neuron-type
¹⁶ input-responsivity in a model of primary visual cortex, a new understanding of rapid task switch-
¹⁷ ing in superior colliculus models, and attribution of bias in recurrent neural networks solving a toy
¹⁸ mathematical problem. More generally, this work offers a quantitative grounding for theoretical
¹⁹ models going forward, pointing a way to how rigorous statistical inference can enhance theoretical
²⁰ 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

when configured appropriately, give rise to a measurable signature of a computation. The work of analyzing a model then requires solving the inverse problem: given a computation of interest, how can we reason about these suitable parameter configurations? The inverse problem is crucial for reasoning about the model’s likely parameter values, uniquenesses and degeneracies, attractor states and phase transitions, and most importantly, the predictions made.

Consider the idealized practice: one carefully designs a model and analytically derives how model parameters govern the computation. Seminal examples of this gold standard include our field’s understanding of memory capacity in associative neural networks [2] and chaos and autocorrelation timescales in random neural networks [3] (which use models and analyses originating in physics), as well as the paradoxical effect in excitatory/inhibitory networks [4], we need [?], more examples [?]. Unfortunately, as circuit models include more biological realism, theory via analytic derivation becomes intractable. This creates an unfavorable tradeoff. On the one hand, one may tractably analyze systems of equations with unrealistic assumptions (for example symmetry or gaussianity), producing accurate inferences about parameters of a too-simple model. On the other hand, one may choose a more biologically accurate, scientifically relevant model at the cost of *ad hoc* approaches to analysis (simply examining simulated activity), potentially resulting in bad inferences and thus erroneous scientific predictions and conclusions.

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

However, these inference tools have not significantly influenced the study of theoretical neuroscience models, for at least three reasons. First, at a practical level, the nonlinearities and dynamics of

many theoretical models are such that conventional inference tools typically produce a narrow set of insights into these models. Indeed, only in the last few years has deep learning research advanced to a point of relevance to this class of problem. Second, the object of interest from a theoretical model is not typically data itself, but rather a qualitative phenomenon – inspection of model behavior, or better, a measurable signature of some computation – an *emergent property* of the model. Third, because carefully constructed biological models do not fit cleanly into the framing of a statistical model. Technically, because many such models stipulate a noisy system of differential equations that can only be sampled or realized through forward simulation, they lack the explicit likelihood and priors central to the probabilistic modeling toolkit.

To address these three challenges, we developed an inference methodology – ‘emergent property inference’ – which learns a distribution over parameter configurations in a theoretical model. Critically, this distribution is such that draws from the distribution (parameter configurations) correspond to systems of equations that give rise to a specified emergent property. First, we stipulate a bijective deep neural network that induces a flexible family of probability distributions over model parameterizations with a probability density we can calculate [15, 16, 17]. Second, we quantify the notion of emergent properties as a set of moment constraints on datasets generated by the model. Thus, an emergent property is not a single data realization, but a phenomenon or a feature of the model, which is ultimately the object of interest in theoretical neuroscience (unlike neural data analysis). Conditioning on an emergent property requires a variant of deep probabilistic inference methods, which we have previously introduced [18]. Third, because we cannot assume the theoretical model has explicit likelihood on data or the emergent property of interest, we use stochastic gradient techniques in the spirit of likelihood free variational inference [19]. Taken together, emergent property inference (EPI) provides a methodology for inferring parameter configurations consistent with a particular emergent phenomena in theoretical models. Not only are these parameter distributions optimized to identify all likely parameters, they give us a probabilistic tool for reasoning about parametric sensitivity and robustness, which should prove invaluable in biological models with pervasive degeneracies [20]. We use a classic example of parametric degeneracy in a biological system, the stomatogastric ganglion [21], to motivate and clarify the technical details of EPI.

Equipped with this methodology, we then investigated three models of current importance in theoretical neuroscience. These models were chosen to demonstrate generality through ranges of biological realism (from conductance-based biophysics to recurrent neural networks), neural system

function (from pattern generation to abstract cognitive function), and network scale (from four to infinite neurons). First, we use EPI to produce a set of verifiable hypotheses of input-responsivity in a four neuron-type dynamical model of primary visual cortex; we then validate these hypotheses in the model. Second, we demonstrated how the systematic application of EPI to levels of task performance can generate experimentally testable hypotheses regarding connectivity in superior colliculus. Third, we use EPI to uncover the sources of bias in a low-rank recurrent neural network executing a toy mathematical computation. The novel scientific insights offered by EPI contextualize and clarify the previous studies exploring these models [22, 23, 24, 25] and more generally, suggests a departure from realism vs tractability considerations towards the use of modern machine learning for sophisticated interrogation of biologically relevant models.

We note that, during our preparation and early presentation of this work [26, 27], another work has arisen with broadly similar goals: bringing statistical inference to mechanistic models of neural circuits [28]. We are excited by this broad problem being recognized by the community, and we emphasize that these works offer complementary neuroscientific contributions and use different technical methodologies. Scientifically, our work has focused primarily on systems-level theoretical models, while their focus has been on lower-level cellular models. Secondly, there are several key technical differences in the approaches (see Section A.1.4) perhaps most notably is our focus on the emergent property – the measurable signal of the computation in question, vs their focus on observed datasets; both certainly are worthy pursuits. The existence of these complementary methodologies emphasizes the increased importance and timeliness of both works.

3 Results

3.1 Motivating emergent property inference of theoretical models

Consideration of the typical workflow of theoretical modeling clarifies the need for emergent property inference. First, one designs or chooses an existing model that, it is hypothesized, captures the computation of interest. To ground this process in a well-known example, consider the stomatogastric ganglion (STG) of crustaceans, a small neural circuit which generates multiple rhythmic muscle activation patterns for digestion [29]. Despite full knowledge of STG connectivity and a precise characterization of its rhythmic pattern generation, biophysical models of the STG have complicated relationships between circuit parameters and neural activity [30]. A model of the STG [22] is shown schematically in Figure 1A, and note that the behavior of this model will be critically

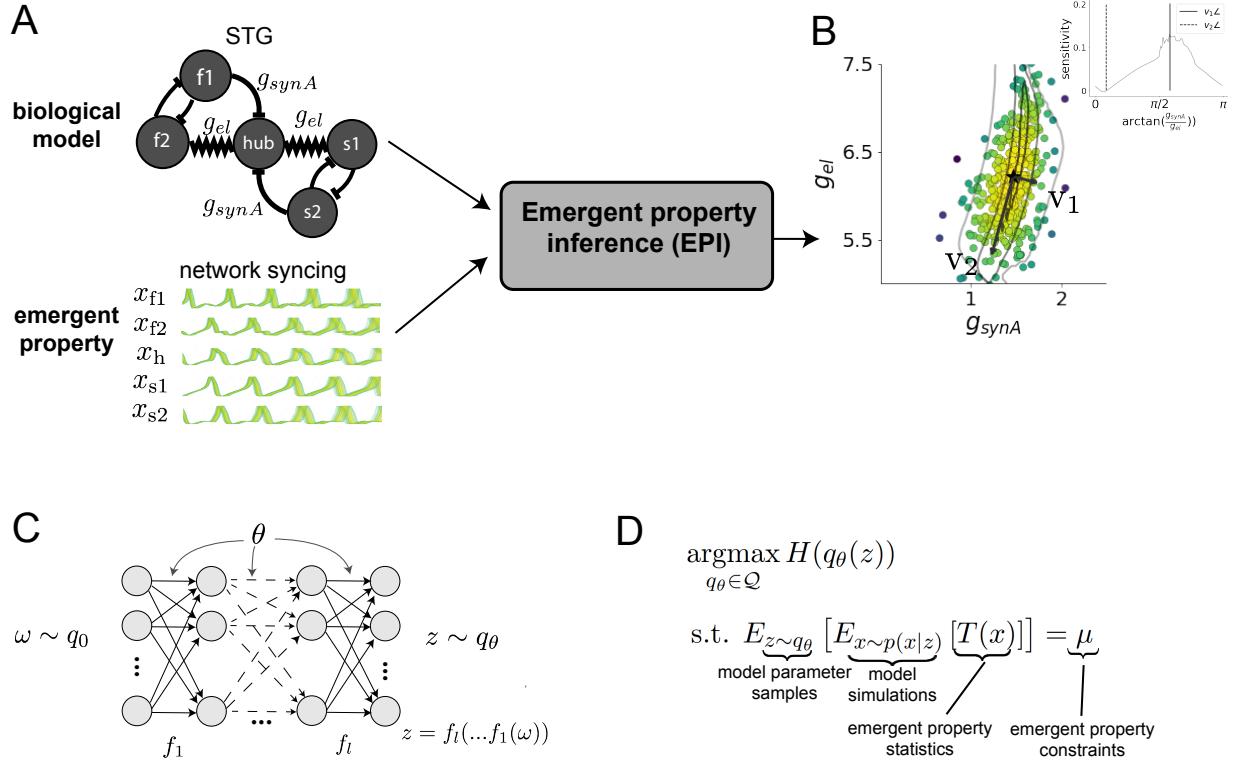


Figure 1: Emergent property inference (EPI) in the stomatogastric ganglion. A. For a choice of model (STG) and emergent property (network syncing), emergent property inference (EPI) learns a posterior distribution of the model parameters $z = [g_{el}, g_{synA}]^\top$ conditioned on network syncing. B.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

dependent on its parameterization – the choices of conductance parameters $z = [g_{el}, g_{synA}]$. Specifically, the two fast neurons ($f1$ and $f2$) mutually inhibit one another, and oscillate at a faster frequency than the mutually inhibiting slow neurons ($s1$ and $s2$), and the hub neuron (hub) couples with the fast or slow population or both.

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

Third, qualitative parameter analysis ensues: since precise mathematical analysis is intractable in this model, a brute force sweep of parameters is done [22]. 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

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

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

where $C_m=1\text{nF}$, and h_{leak} , h_{Ca} , h_K , h_{hyp} , h_{elec} , h_{syn} are the leak, calcium, potassium, hyperpolarization, electrical, and synaptic currents, all of which have their own complicated dependence on x and $z = [g_{el}, g_{synA}]$ (see Section A.2.1).

Second, we define the emergent property, which as above is network syncing: oscillation of the entire population at an intermediate frequency of our choosing (Figure 1A bottom). Quantifying this phenomenon is straightforward: we define network syncing to be that each neuron’s spiking frequency – denoted $\omega_{f1}(x)$, $\omega_{f2}(x)$, etc. – is close to an intermediate frequency of 0.55Hz. Mathematically, we achieve this via constraints on the mean and variance of $\omega_i(x)$ for each neuron $i \in \{f1, f2, hub, s1, s2\}$, and thus:

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

which completes the quantification of the emergent property.

Third, we perform emergent property inference: we find a distribution over parameter configurations z , and insist that samples from this distribution produce the emergent property; in other words, they obey the constraints introduced in Equation 14. This distribution will be chosen from a family of probability distributions $\mathcal{Q} = \{q_\theta(z) : \theta \in \Theta\}$, defined by a deep generative distribution of the normalizing flow class [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 14, and $p(x|z)$ is the intractable distribution of data from the model (x), given that model’s parameters z (we access samples from this distribution by running

the model forward). The purpose of each element in this program is detailed in Figure 1D. Finally, we recognize that many distributions in \mathcal{Q} will respect the emergent property constraints, so we require a normative principle to select amongst them. This principle is captured in Equation 3 by the primal objective H . Here we chose Shannon entropy as a means to find parameter distributions with minimal assumptions beyond some chosen structure [31, 32, 18, 33], but we emphasize that the EPI method is unaffected by this choice (but the results of course will depend on the primal objective chosen).

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

3.3 Comprehensive input-responsivity in a nonlinear sensory system

In studies of primary visual cortex (V1), theoretical models with excitatory (E) and inhibitory (I) populations have reproduced a host of experimentally documented phenomena. In particular regimes of excitation and inhibition, these E/I models exhibit the paradoxical effect [4], selective amplification [34], surround suppression [35], and sensory integrative properties [36]. Extending this model using experimental evidence of three genetically-defined classes of inhibitory neurons [37, 38], recent work [23] has investigated a four-population model – excitatory (E), parvalbumin (P), somatostatin (S), and vasointestinal peptide (V) neurons – as shown in Fig. 2A. The dynamical state of this model is the firing rate of each neuron-type population $x = [x_E, x_P, x_S, x_V]^\top$, which evolves according to rectified ($\llbracket \cdot \rrbracket_+$) and exponentiated dynamics:

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

with effective connectivity weights W and input h . In our analysis, we set the time constant $\tau = 20\text{ms}$ and dynamics coefficient $n = 2$. Also, as is fairly standard, we obtain an informative estimate of the effective connectivities between these neuron-types W in mice by multiplying their probability of connection with their average synaptic strength [39, 40] (see Section A.2.2). Given

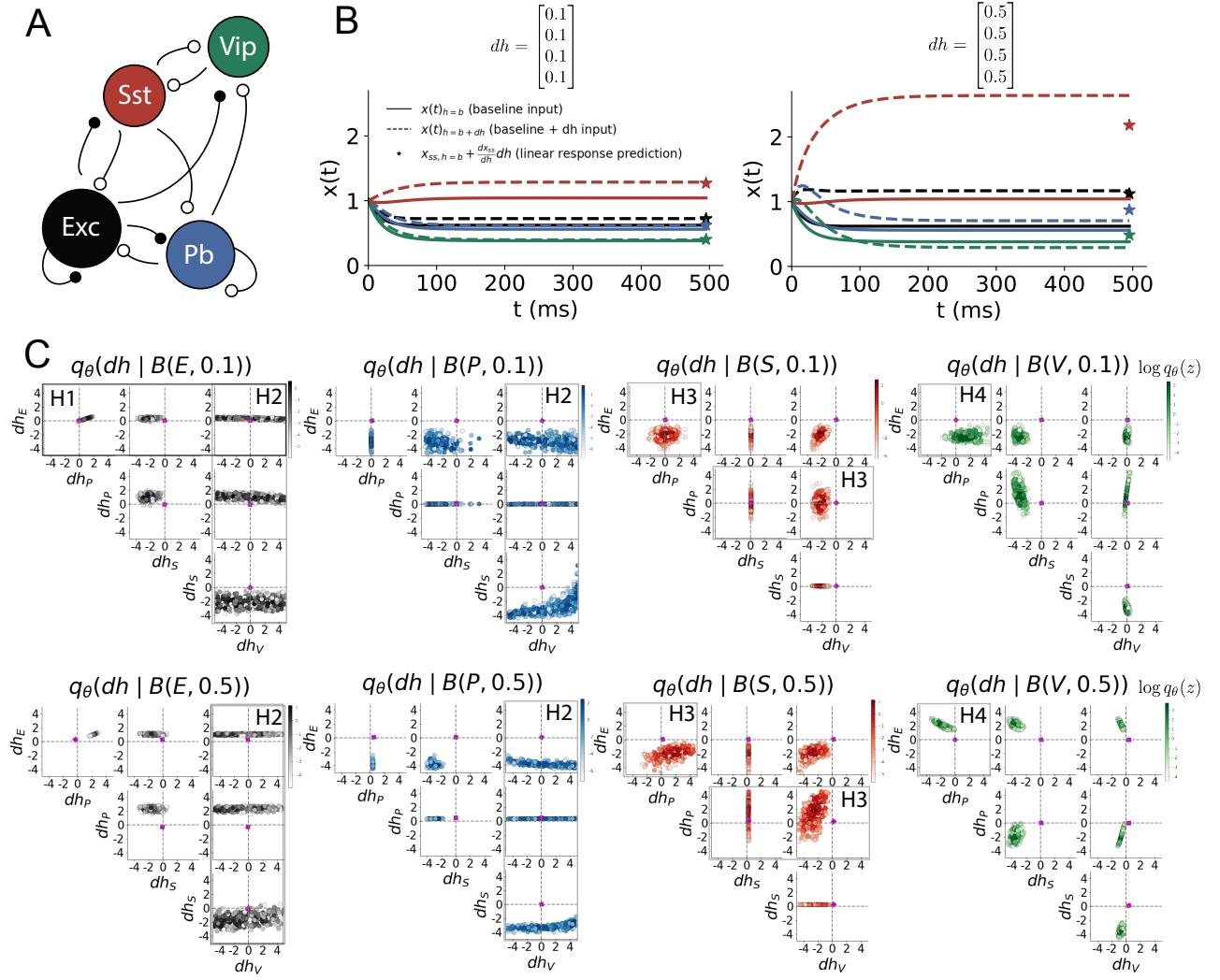


Figure 2: Hypothesis generation through EPI in a V1 model. A. Four-population model of primary visual cortex with excitatory (black), parvalbumin (blue), somatostatin (red), and vip (green) neurons. Some neuron-types largely do not form synaptic projections to others (excitatory and inhibitory projections filled and unfilled, respectively). B. Linear response predictions become inaccurate with greater input strength. V1 model simulations for input (b) solid and ($b + dh$) dashed. $b = [1, 1, 1, 1]^T$ and (left) $dh = [0.1, 0.1, 0.1, 0.1]^T$ (right) $dh = [0.5, 0.5, 0.5, 0.5]^T$. Stars indicate the linear response prediction. C. EPI distributions on differential input dh conditioned on differential response $\mathcal{B}(\alpha, y)$. Supporting evidence for the four generated hypotheses are indicated by gray boxes with labels H1, H2, H3, and H4. The linear prediction from two standard deviations away from y (from negative to positive) is overlaid in magenta (very small, near origin).

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

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

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

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

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

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

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

225

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

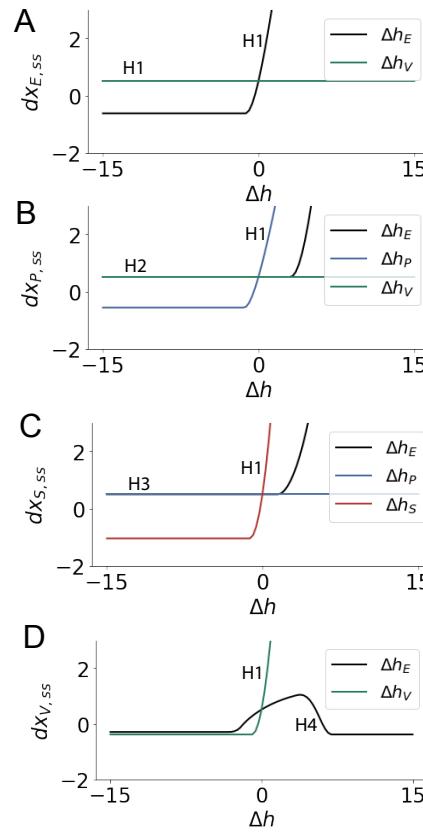


Figure 3: Confirming EPI generated hypotheses in V1. A. Differential responses by the E-population to changes in individual input $\Delta h_\alpha u_\alpha$ away from the mode of the EPI distribution dh^* . B-D Same plots for the P-, S-, and V-populations. Labels H1, H2, H3, and H4 indicate which curves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

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

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

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

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

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

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

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

251 Identifying measurable biological changes that result in improved behavior is important for neuro-
 252 science, since they may indicate how the learning brain adapts. In a rapid task switching experiment
 253 [41], rats were explicitly cued on each trial to either orient towards a visual stimulus in the Pro
 254 (P) task or orient away from a visual stimulus in the Anti (A) task (Fig. 3a). Neural recordings
 255 in the midbrain superior colliculus (SC) exhibited two populations of neurons that simultaneously
 256 represented both task context (Pro or Anti) and motor response (contralateral or ipsilateral to the
 257 recorded side): the Pro/Contra and Anti/Ipsi neurons [24]. Duan et al. proposed a model of SC
 258 that, like the V1 model analyzed in the previous section, is a four-population dynamical system.
 259 Here, the neuron-type populations are functionally-defined as the Pro- and Anti-populations in each
 260 hemisphere (left (L) and right (R)). The Pro- or Anti-populations receive an input determined by
 261 the cue, and then the left and right populations receive an input based on the side of the light
 262 stimulus. Activities were bounded between 0 and 1, so that a high output of the Pro population
 263 in a given hemisphere corresponds to the contralateral response. An additional stipulation is that
 264 when one Pro population responds with a high-output, the opposite Pro population must respond
 265 with a low output. Finally, this circuit operates in the presence of Gaussian noise resulting in trial-
 266 to-trial variability (see Section A.2.3). The connectivity matrix is parameterized by the geometry
 267 of the population arrangement (Fig. 3B).

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

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

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

²⁸⁹ 3.5 Linking RNN connectivity to computational error

²⁹⁰ So far, each model we have studied was designed from fundamental biophysical principles, genetically-
²⁹¹ or functionally-defined neuron types. At a more abstract level of modeling, recurrent neural net-
²⁹² works (RNNs) are high-dimensional dynamical models of computation that are becoming increas-
²⁹³ ingly popular in neuroscience research [42]. In theoretical neuroscience, RNN dynamics usually
²⁹⁴ follow the equation

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

²⁹⁵ where $x(t)$ is the network activity, W is the network connectivity, $\phi(\cdot) = \tanh(\cdot)$, and $I(t)$ is the
²⁹⁶ input to the system. Such RNNs are trained to do a task from a systems neuroscience experiment,
²⁹⁷ and then the unit activations of the trained RNN are compared to recorded neural activity. Fully-
²⁹⁸ connected RNNs with tens of thousands of parameters are challenging to characterize [43], especially
²⁹⁹ making statistical inferences about their parameterization. Alternatively, we consider a rank-1, N -
³⁰⁰ neuron RNN with connectivity

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

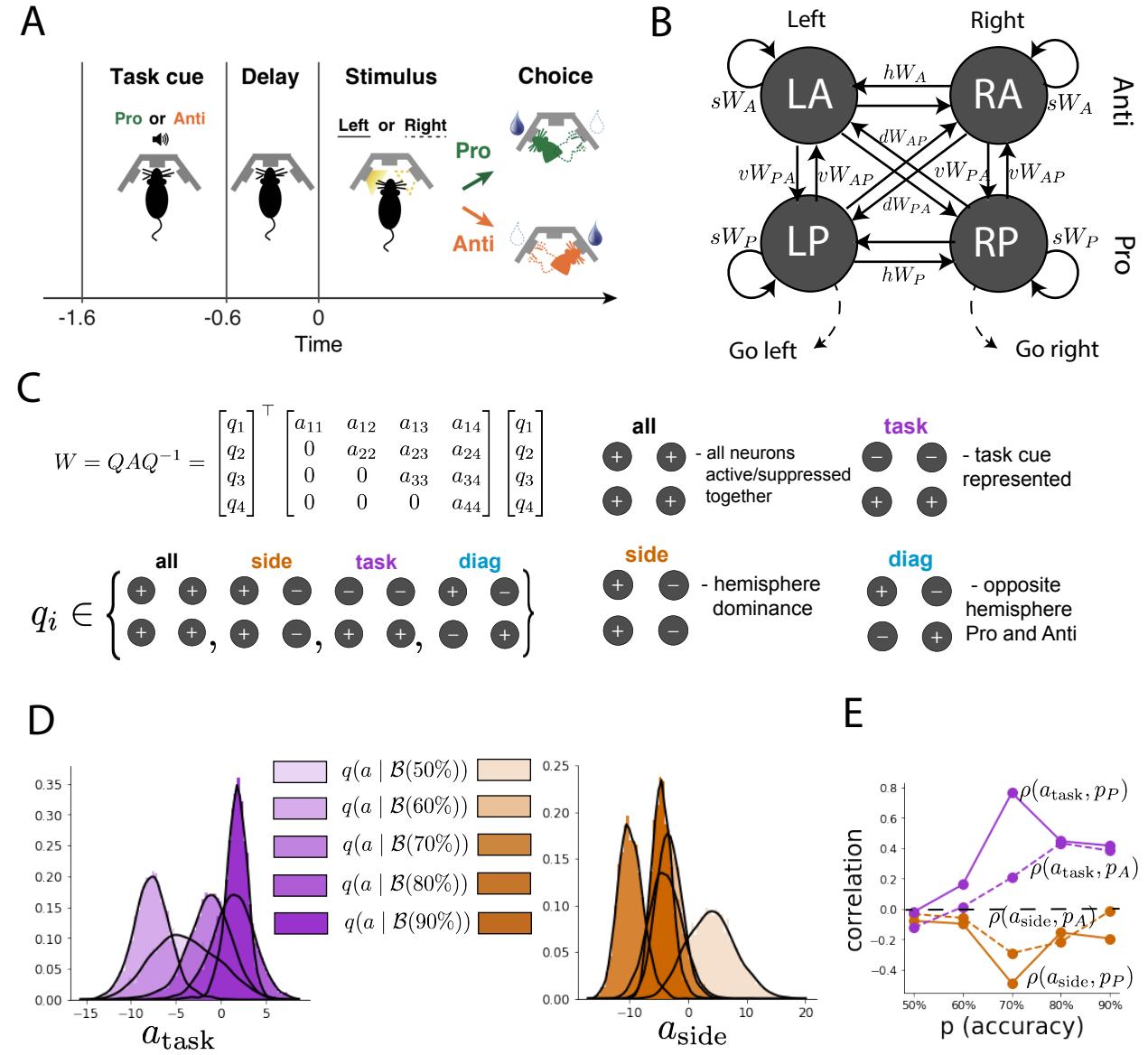


Figure 4: EPI reveals changes in SC [24] connectivity that control task accuracy. A. Rapid task switching behavioral paradigm (see text). B. Model of superior colliculus (SC). Neurons: LP - left pro, RP - right pro, LA - left anti, RA - right anti. Parameters: sW - self, hW - horizontal, vW - vertical, 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.

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

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

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

309 The RNN is trained to solve this task by producing readout activity that is on average the posterior
 310 mean μ_{post} , and activity whose variability is the posterior variance σ_{post}^2 (a setup inspired by
 311 [44]). To solve this Gaussian posterior conditioning task, the RNN response to a constant input
 312 $I(t) = yw + (n - M_n)$ must equal the posterior mean along readout vector w , where

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

313 Additionally, the amount of chaotic variance Δ_T must equal the posterior variance. κ_w and Δ_T can
 314 be expressed in terms of each other through a solvable system of nonlinear equations (see Section
 315 A.2.4) [25]. This theory allows us to mathematically formalize the execution of this task into an
 316 emergent property, where the emergent property statistics of the RNN activity are k_w and Δ_T and
 317 the emergent property values are the ground truth posterior mean μ_{post} and variance σ_{post}^2 :

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

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

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

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

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

H2: The posterior variance increases with g ;

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

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

4 Discussion

4.1 EPI is a general tool for theoretical neuroscience

Models of biological systems are often comprised of complex nonlinear differential equations, making traditional theoretical analysis and statistical inference intractable. In contrast, EPI is capable of learning distributions of parameters in such models producing measurable signatures of computation. We have demonstrated its utility on biological models (STG), intermediate-level models of interacting genetically- and functionally-defined neuron-types (V1, SC), and the most abstract of models (RNNs). We are able to condition both deterministic and stochastic models on low-level emergent properties like firing rates of membrane potentials, as well as high-level cognitive function like Gaussian posterior conditioning. Technically, EPI is tractable when the emergent property statistics are continuously differentiable with respect to the model parameters, which is very often the case; this emphasizes the general utility of EPI.

In this study, we have focused on applying EPI to low dimensional parameter spaces of models with low dimensional dynamical state. These choices were made to present the reader with a series of interpretable conclusions, which is more challenging in high dimensional spaces. In fact, EPI should scale reasonably to high dimensional parameter spaces, as the underlying technology has

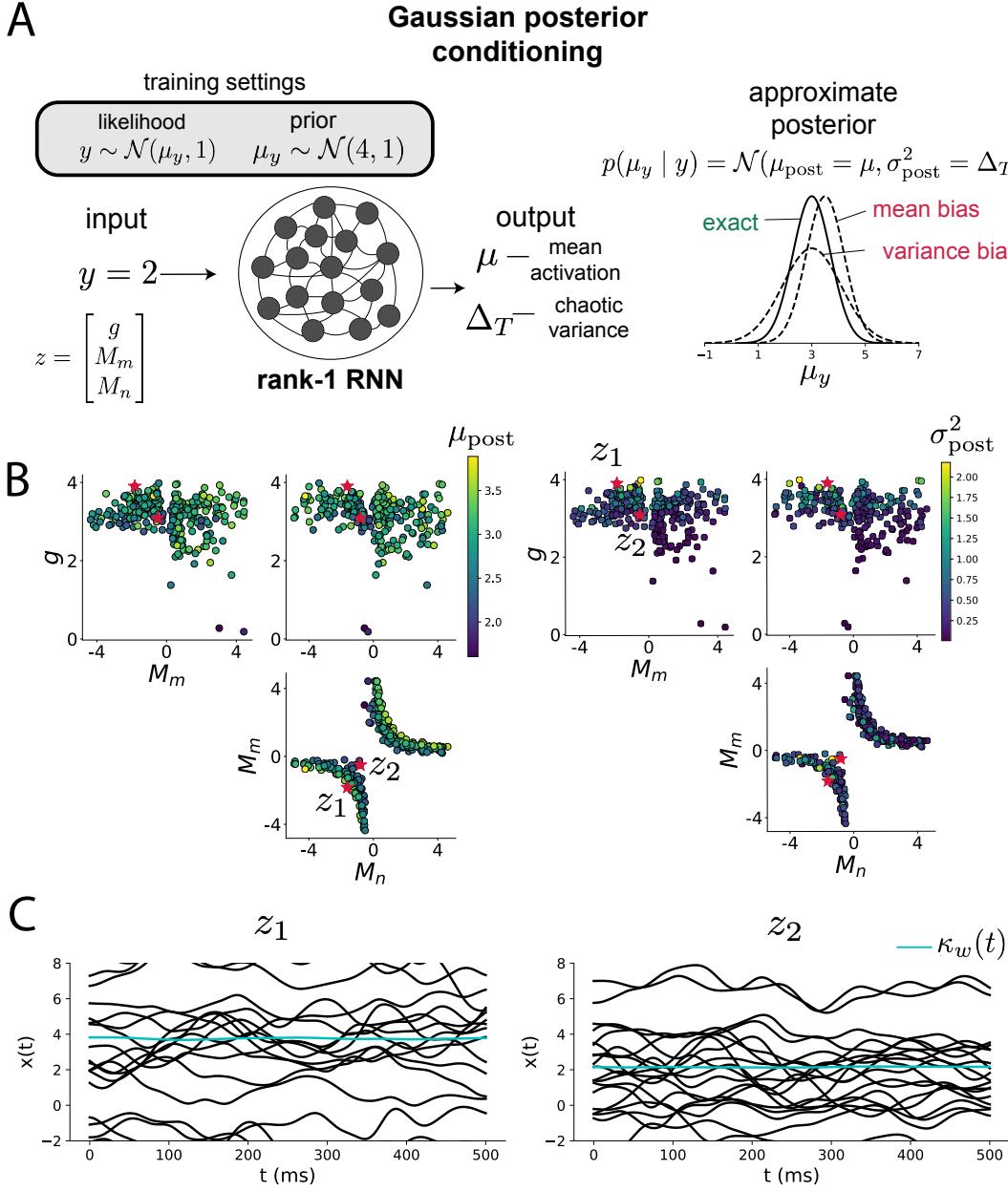


Figure 5: Sources of solution bias in an RNN computation. A. (left) A rank-1 RNN executing a Gaussian posterior conditioning computation on μ_y . (right) Bias in this computation can come from over- or under-estimating the posterior mean or variance. B. EPI distribution of rank-1 RNNs executing Gaussian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

354 produced state-of-the-art performance on high-dimensional tasks such as texture generation [18].
355 Of course, increasing the dimensionality of the dynamical state of the model makes optimization
356 more expensive, and there is a practical limit there as with any machine learning approach. For
357 systems with high dimensional state, we recommend using theoretical approaches (e.g. [25]) to
358 reason about reduced parameterizations of such high-dimensional systems.

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

370 **4.2 Novel hypotheses from EPI**

371 Machine learning has played an effective, multifaceted role in neuroscientific progress. Primarily,
372 it has revealed structure in large-scale neural datasets [45, 46, 47, 48, 49, 50] (see review, [14]).
373 Secondarily, trained algorithms of varying degrees of biological relevance are beginning to be viewed
374 as fully-observable computational systems comparable to the brain [43, 51].

375 For example, consider the fact that we do not fully understand the four-dimensional models of V1
376 [23]. Because analytical approaches to studying nonlinear dynamical systems become increasingly
377 complicated when stepping from two-dimensional to three- or four-dimensional systems in the
378 absence of restrictive simplifying assumptions [52], it is unsurprising that this model has been a
379 challenge. In Section 3.3, we showed that EPI was far more informative about neuron-type input
380 responsibility than the predictions afforded through analysis. By flexibly conditioning this V1 model
381 on different emergent properties, we performed an exploratory analysis of a *model* rather than a
382 dataset, which generated and proved out a set of testable predictions.

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

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

389 In our final analysis, we present a novel procedure for doing statistical inference on interpretable
390 parameterizations of RNNs executing simple tasks . This methodology relies on recently extended
391 theory of responses in random neural networks with minimal structure [25]. With this methodology,
392 we can finally open the probabilistic model selection toolkit reasoning about the connectivity of
393 RNNs solving tasks.

394 References

- 395 [1] Larry F Abbott. Theoretical neuroscience rising. *Neuron*, 60(3):489–495, 2008.
- 396 [2] John J Hopfield. Neural networks and physical systems with emergent collective computational
397 abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558, 1982.
- 398 [3] Haim Sompolinsky, Andrea Crisanti, and Hans-Jurgen Sommers. Chaos in random neural
399 networks. *Physical review letters*, 61(3):259, 1988.
- 400 [4] Misha V Tsodyks, William E Skaggs, Terrence J Sejnowski, and Bruce L McNaughton. Para-
401 doxical effects of external modulation of inhibitory interneurons. *Journal of neuroscience*,
402 17(11):4382–4388, 1997.
- 403 [5] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *International Confer-
404 ence on Learning Representations*, 2014.
- 405 [6] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation
406 and variational inference in deep latent gaussian models. *International Conference on Machine
407 Learning*, 2014.
- 408 [7] Yuanjun Gao, Evan W Archer, Liam Paninski, and John P Cunningham. Linear dynamical
409 neural population models through nonlinear embeddings. In *Advances in neural information
410 processing systems*, pages 163–171, 2016.

- 411 [8] Yuan Zhao and Il Memming Park. Recursive variational bayesian dual estimation for nonlinear
412 dynamics and non-gaussian observations. *stat*, 1050:27, 2017.
- 413 [9] Gabriel Barello, Adam Charles, and Jonathan Pillow. Sparse-coding variational auto-encoders.
414 *bioRxiv*, page 399246, 2018.
- 415 [10] Chethan Pandarinath, Daniel J O’Shea, Jasmine Collins, Rafal Jozefowicz, Sergey D Stavisky,
416 Jonathan C Kao, Eric M Trautmann, Matthew T Kaufman, Stephen I Ryu, Leigh R Hochberg,
417 et al. Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature
418 methods*, page 1, 2018.
- 419 [11] Alexander B Wiltschko, Matthew J Johnson, Giuliano Iurilli, Ralph E Peterson, Jesse M
420 Katon, Stan L Pashkovski, Victoria E Abraira, Ryan P Adams, and Sandeep Robert Datta.
421 Mapping sub-second structure in mouse behavior. *Neuron*, 88(6):1121–1135, 2015.
- 422 [12] Matthew J Johnson, David K Duvenaud, Alex Wiltschko, Ryan P Adams, and Sandeep R
423 Datta. Composing graphical models with neural networks for structured representations and
424 fast inference. In *Advances in neural information processing systems*, pages 2946–2954, 2016.
- 425 [13] Eleanor Batty, Matthew Whiteway, Shreya Saxena, Dan Biderman, Taiga Abe, Simon Musall,
426 Winthrop Gillis, Jeffrey Markowitz, Anne Churchland, John Cunningham, et al. Behavenet:
427 nonlinear embedding and bayesian neural decoding of behavioral videos. *Advances in Neural
428 Information Processing Systems*, 2019.
- 429 [14] Liam Paninski and John P Cunningham. Neural data science: accelerating the experiment-
430 analysis-theory cycle in large-scale neuroscience. *Current opinion in neurobiology*, 50:232–241,
431 2018.
- 432 [15] Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows.
433 *International Conference on Machine Learning*, 2015.
- 434 [16] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp.
435 *arXiv preprint arXiv:1605.08803*, 2016.
- 436 [17] George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density
437 estimation. In *Advances in Neural Information Processing Systems*, pages 2338–2347, 2017.
- 438 [18] Gabriel Loaiza-Ganem, Yuanjun Gao, and John P Cunningham. Maximum entropy flow
439 networks. *International Conference on Learning Representations*, 2017.

- [19] Dustin Tran, Rajesh Ranganath, and David Blei. Hierarchical implicit models and likelihood-free variational inference. In *Advances in Neural Information Processing Systems*, pages 5523–5533, 2017.
- [20] Mark K Transtrum, Benjamin B Machta, Kevin S Brown, Bryan C Daniels, Christopher R Myers, and James P Sethna. Perspective: Sloppiness and emergent theories in physics, biology, and beyond. *The Journal of chemical physics*, 143(1):07B201_1, 2015.
- [21] Mark S Goldman, Jorge Golowasch, Eve Marder, and LF Abbott. Global structure, robustness, and modulation of neuronal models. *Journal of Neuroscience*, 21(14):5229–5238, 2001.
- [22] Gabrielle J Gutierrez, Timothy O’Leary, and Eve Marder. Multiple mechanisms switch an electrically coupled, synaptically inhibited neuron between competing rhythmic oscillators. *Neuron*, 77(5):845–858, 2013.
- [23] Ashok Litwin-Kumar, Robert Rosenbaum, and Brent Doiron. Inhibitory stabilization and visual coding in cortical circuits with multiple interneuron subtypes. *Journal of neurophysiology*, 115(3):1399–1409, 2016.
- [24] Chunyu A Duan, Marino Pagan, Alex T Piet, Charles D Kopec, Athena Akrami, Alexander J Riordan, Jeffrey C Erlich, and Carlos D Brody. Collicular circuits for flexible sensorimotor routing. *bioRxiv*, page 245613, 2018.
- [25] Francesca Mastrogiovanni and Srdjan Ostojic. Linking connectivity, dynamics, and computations in low-rank recurrent neural networks. *Neuron*, 99(3):609–623, 2018.
- [26] Sean R Bittner, Agostina Palmigiano, Kenneth D Miller, and John P Cunningham. Degenerate solution networks for theoretical neuroscience. *Computational and Systems Neuroscience Meeting (COSYNE), Lisbon, Portugal*, 2019.
- [27] Sean R Bittner, Alex T Piet, Chunyu A Duan, Agostina Palmigiano, Kenneth D Miller, Carlos D Brody, and John P Cunningham. Examining models in theoretical neuroscience with degenerate solution networks. *Bernstein Conference*, 2019.
- [28] Jan-Matthis Lueckmann, Pedro Goncalves, Chaitanya Chintaluri, William F Podlaski, Giacomo Bassetto, Tim P Vogels, and Jakob H Macke. Amortised inference for mechanistic models of neural dynamics. In *Computational and Systems Neuroscience Meeting (COSYNE), Lisbon, Portugal*, 2019.

- 469 [29] Eve Marder and Vatsala Thirumalai. Cellular, synaptic and network effects of neuromodulation. *Neural Networks*, 15(4-6):479–493, 2002.
- 470
- 471 [30] Astrid A Prinz, Dirk Bucher, and Eve Marder. Similar network activity from disparate circuit
472 parameters. *Nature neuroscience*, 7(12):1345, 2004.
- 473 [31] Edwin T Jaynes. Information theory and statistical mechanics. *Physical review*, 106(4):620,
474 1957.
- 475 [32] Gamaleldin F Elsayed and John P Cunningham. Structure in neural population recordings:
476 an expected byproduct of simpler phenomena? *Nature neuroscience*, 20(9):1310, 2017.
- 477 [33] Cristina Savin and Gašper Tkačik. Maximum entropy models as a tool for building precise
478 neural controls. *Current opinion in neurobiology*, 46:120–126, 2017.
- 479 [34] Brendan K Murphy and Kenneth D Miller. Balanced amplification: a new mechanism of
480 selective amplification of neural activity patterns. *Neuron*, 61(4):635–648, 2009.
- 481 [35] Hirofumi Ozeki, Ian M Finn, Evan S Schaffer, Kenneth D Miller, and David Ferster. Inhibitory
482 stabilization of the cortical network underlies visual surround suppression. *Neuron*, 62(4):578–
483 592, 2009.
- 484 [36] Daniel B Rubin, Stephen D Van Hooser, and Kenneth D Miller. The stabilized supralinear
485 network: a unifying circuit motif underlying multi-input integration in sensory cortex. *Neuron*,
486 85(2):402–417, 2015.
- 487 [37] Henry Markram, Maria Toledo-Rodriguez, Yun Wang, Anirudh Gupta, Gilad Silberberg, and
488 Caizhi Wu. Interneurons of the neocortical inhibitory system. *Nature reviews neuroscience*,
489 5(10):793, 2004.
- 490 [38] Bernardo Rudy, Gordon Fishell, SooHyun Lee, and Jens Hjerling-Leffler. Three groups of
491 interneurons account for nearly 100% of neocortical gabaergic neurons. *Developmental neuro-
492 biology*, 71(1):45–61, 2011.
- 493 [39] (2018) Allen Institute for Brain Science. Layer 4 model of v1. available from:
494 <https://portal.brain-map.org/explore/models/l4-mv1>.
- 495 [40] Yazan N Billeh, Binghuang Cai, Sergey L Gratiy, Kael Dai, Ramakrishnan Iyer, Nathan W
496 Gouwens, Reza Abbasi-Asl, Xiaoxuan Jia, Joshua H Siegle, Shawn R Olsen, et al. Systematic

- 497 integration of structural and functional data into multi-scale models of mouse primary visual
498 cortex. *bioRxiv*, page 662189, 2019.
- 499 [41] Chunyu A Duan, Jeffrey C Erlich, and Carlos D Brody. Requirement of prefrontal and midbrain
500 regions for rapid executive control of behavior in the rat. *Neuron*, 86(6):1491–1503, 2015.
- 501 [42] Omri Barak. Recurrent neural networks as versatile tools of neuroscience research. *Current
502 opinion in neurobiology*, 46:1–6, 2017.
- 503 [43] David Sussillo and Omri Barak. Opening the black box: low-dimensional dynamics in high-
504 dimensional recurrent neural networks. *Neural computation*, 25(3):626–649, 2013.
- 505 [44] Rodrigo Echeveste, Laurence Aitchison, Guillaume Hennequin, and Máté Lengyel. Cortical-like
506 dynamics in recurrent circuits optimized for sampling-based probabilistic inference. *bioRxiv*,
507 page 696088, 2019.
- 508 [45] Robert E Kass and Valérie Ventura. A spike-train probability model. *Neural computation*,
509 13(8):1713–1720, 2001.
- 510 [46] Emery N Brown, Loren M Frank, Dengda Tang, Michael C Quirk, and Matthew A Wilson.
511 A statistical paradigm for neural spike train decoding applied to position prediction from
512 ensemble firing patterns of rat hippocampal place cells. *Journal of Neuroscience*, 18(18):7411–
513 7425, 1998.
- 514 [47] Liam Paninski. Maximum likelihood estimation of cascade point-process neural encoding
515 models. *Network: Computation in Neural Systems*, 15(4):243–262, 2004.
- 516 [48] M Yu Byron, John P Cunningham, Gopal Santhanam, Stephen I Ryu, Krishna V Shenoy, and
517 Maneesh Sahani. Gaussian-process factor analysis for low-dimensional single-trial analysis
518 of neural population activity. In *Advances in neural information processing systems*, pages
519 1881–1888, 2009.
- 520 [49] Kenneth W Latimer, Jacob L Yates, Miriam LR Meister, Alexander C Huk, and Jonathan W
521 Pillow. Single-trial spike trains in parietal cortex reveal discrete steps during decision-making.
522 *Science*, 349(6244):184–187, 2015.
- 523 [50] Lea Duncker, Gergo Bohner, Julien Boussard, and Maneesh Sahani. Learning interpretable
524 continuous-time models of latent stochastic dynamical systems. *Proceedings of the 36th Inter-
525 national Conference on Machine Learning*, 2019.

- 526 [51] Blake A Richards and et al. A deep learning framework for neuroscience. *Nature Neuroscience*,
 527 2019.
- 528 [52] Steven H Strogatz. Nonlinear dynamics and chaos: with applications to physics. *Biology, Chemistry, and Engineering (Studies in Nonlinearity)*, Perseus, Cambridge, UK, 1994.
- 530 [53] Rajesh Ranganath, Sean Gerrish, and David Blei. Black box variational inference. In *Artificial Intelligence and Statistics*, pages 814–822, 2014.
- 532 [54] Martin J Wainwright, Michael I Jordan, et al. Graphical models, exponential families, and
 533 variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.
- 534 [55] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp.
 535 *Proceedings of the 5th International Conference on Learning Representations*, 2017.
- 536 [56] Carsten K Pfeffer, Mingshan Xue, Miao He, Z Josh Huang, and Massimo Scanziani. Inhibition
 537 of inhibition in visual cortex: the logic of connections between molecularly distinct
 538 interneurons. *Nature Neuroscience*, 16(8):1068, 2013.

539 **A Methods**

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

541 Emergent property inference (EPI) learns distributions of theoretical model parameters that pro-
 542 duce emergent properties of interest. EPI combines ideas from likelihood-free variational inference
 543 [19] and maximum entropy flow networks [18]. A maximum entropy flow network is used as a deep
 544 probability distribution for the parameters, while these samples often parameterize a differentiable
 545 model simulator, which may lack a tractable likelihood function.

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

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

551 by making an approximation $q_\theta(z)$ to $p(z | \mathcal{B})$ (see Section A.1.5). So, over the DSN distribution
 552 $q_\theta(z)$ of model $p(x | z)$ for behavior \mathcal{B} , the emergent properties $T(x)$ are constrained in expectation
 553 to μ .

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

559 In most settings (especially those relevant to theoretical neuroscience) the likelihood of the behavior
 560 with respect to the model parameters $p(T(x) | z)$ is unknown or intractable, requiring an alternative
 561 to stochastic gradient variational Bayes [5] or black box variational inference[53]. These types
 562 of methods called likelihood-free variational inference (LFVI, [19]) skate around the intractable
 563 likelihood function in situations where there is a differentiable simulator. Akin to LFVI, DSNs are
 564 optimized with the following objective for a given theoretical model, emergent property statistics
 565 $T(x)$, and emergent property constraints μ :

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

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

569 A.1.1 Example: 2D LDS

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

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

571 that produce a band of oscillations. To do EPI with the dynamics matrix elements as the free
 572 parameters $z = [a_1, a_2, a_3, a_4]$, and fixing $\tau = 1$, such that the posterior yields a band of oscillations,
 573 the emergent property statistics $T(x)$ are chosen to contain the first- and second-moments of the
 574 oscillatory frequency Ω and the growth/decay factor d of the oscillating system. To learn the

575 distribution of real entries of A that yield a distribution of d with mean zero with variance 0.25^2 ,
 576 and oscillation frequency Ω with mean 1 Hz with variance $(0.1\text{Hz})^2$, then we would select the real
 577 part of the complex conjugate eigenvalues $\text{real}(\lambda_1) = d$ (via an arbitrary choice of eigenvalue of the
 578 dynamics matrix λ_1) and the positive imaginary component of one of the eigenvalues $\text{imag}(\lambda_1) =$
 579 $2\pi\Omega$ as the emergent property statistics. Those emergent property statistics are then constrained
 580 to

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

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

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

584 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
 585 directly via a closed form function and does not require simulation, we cannot derive the distribution
 586 q_θ^* directly. This is due to the formally hard problem of the backward mapping: finding the natural
 587 parameters η from the mean parameters μ of an exponential family distribution [54]. Instead, we
 588 can use EPI to learn the linear system parameters producing such a band of oscillations (Fig. S2B).

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

599 To explain the structure in the bimodality of the DSN posterior, we can look at the imaginary



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.

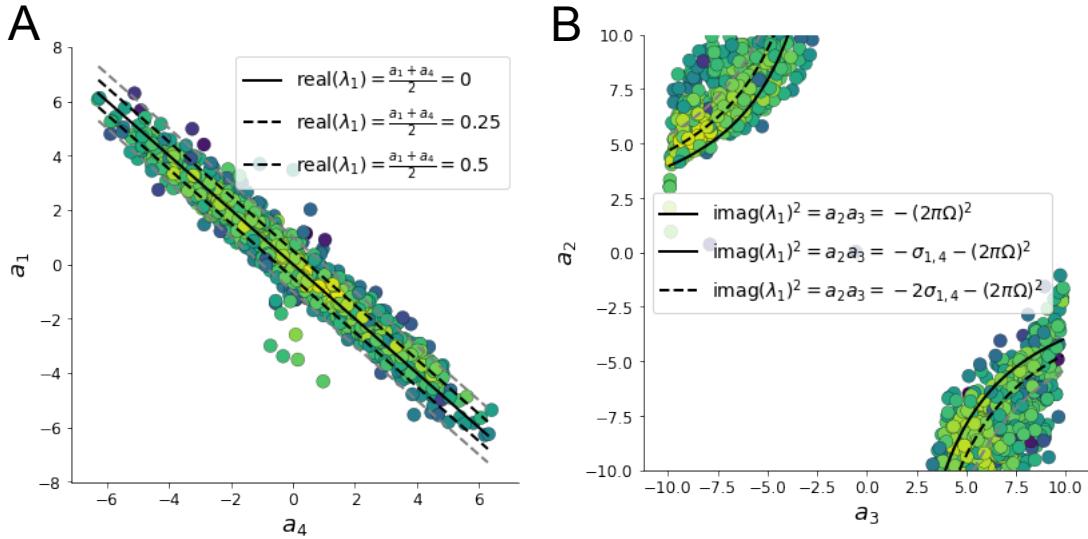


Fig. S3: A. Probability contours in the $a_1 - a_4$ plane can be derived from the relationship to emergent property statistic of growth/decay factor. B. Probability contours in the $a_2 - a_3$ plane can be derived from relationship to the emergent property statistic of oscillation frequency.

600 component of λ_1 . When $\text{real}(\lambda_1) = \frac{a_1 + a_4}{2} = 0$, we have

$$\text{imag}(\lambda_1) = \begin{cases} \sqrt{\frac{a_1 a_4 - a_2 a_3}{\tau}}, & \text{if } a_1 a_4 < a_2 a_3 \\ 0 & \text{otherwise} \end{cases} \quad (17)$$

601 When $\tau = 1$ and $a_1 a_4 > a_2 a_3$ (center of distribution above), we have the following equation for the
602 other two dimensions:

$$\text{imag}(\lambda_1)^2 = a_1 a_4 - a_2 a_3 \quad (18)$$

603 Since we constrained $E_{q_\theta} [\text{imag}(\lambda)] = 2\pi$ (with $\omega = 1$), we can plot contours of the equation
604 $\text{imag}(\lambda_1)^2 = a_1 a_4 - a_2 a_3 = (2\pi)^2$ for various $a_1 a_4$ (Fig. S3A). If $\sigma_{1,4} = E_{q_\theta} [|a_1 a_4 - E_{q_\theta}[a_1 a_4]|]$,
605 then we plot the contours as $a_1 a_4 = 0$ (black), $a_1 a_4 = -\sigma_{1,4}$ (black dotted), and $a_1 a_4 = -2\sigma_{1,4}$
606 (grey dotted) (Fig. S3B). This validates the curved structure of the inferred distribution learned
607 through EPI. We take steps in negative standard deviation of $a_1 a_4$ (dotted and gray lines), since
608 there are few positive values $a_1 a_4$ in the posterior. Subtler model-behavior combinations will have
609 even more complexity, further motivating the use of EPI for understanding these systems. Indeed,
610 we sample a distribution of systems oscillating near 1Hz (Fig. S4).

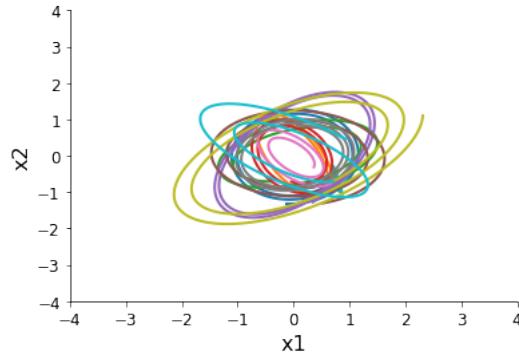


Fig. S4: Sampled dynamical system trajectories from the EPI distribution. Each trajectory is initialized at $x(0) = \begin{bmatrix} \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \end{bmatrix}$.

611 A.1.2 Augmented Lagrangian optimization

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

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

614 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
615 coefficient. For a fixed (α, c) , θ is optimized with stochastic gradient descent. A low value of c is
616 used initially, and increased during each augmented Lagrangian epoch – a period of optimization
617 with fixed α and c for a given number of stochastic optimization iterations. Similarly, α is tuned
618 each epoch based on the constraint violations. For the linear 2-dimensional system (Fig. S2C)
619 optimization hyperparameters are initialized to $c_1 = 10^{-4}$ and $\alpha_1 = 0$. The penalty coefficient
620 is updated based on a hypothesis test regarding the reduction in constraint violation. The p-
621 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
622 $1 - p$. Throughout the project, $\beta = 4.0$ and $\gamma = 0.25$ is used. The other update rule is $\alpha_{k+1} =$
623 $\alpha_k + c_k \frac{1}{n} \sum_{i=1}^n (T(x^{(i)}) - \mu)$. In this example, each augmented Lagrangian epoch ran for 2,000
624 iterations. We consider the optimization to have converged when a null hypothesis test of constraint
625 violations being zero is accepted for all constraints at a significance threshold 0.05. This is the dotted
626 line on the plots below depicting the optimization cutoff of EPI optimization for the 2-dimensional
627 linear system. If the optimization is left to continue running, entropy usually decreases, and
628 structural pathologies in the distribution may be introduced.

629 The intention is that c and α start at values encouraging entropic growth early in optimization.

630 Then, as they increase in magnitude with each training epoch, the constraint satisfaction terms are
 631 increasingly weighted, resulting in a decrease in entropy. Rather than using a naive initialization,
 632 before EPI, we optimize the deep probability distribution parameters to generate samples of an
 633 isotropic Gaussian of a selected variance, such as 1.0 for the 2D LDS example. This provides a
 634 convenient starting point, whose level of entropy is controlled by the user.

635 **A.1.3 Normalizing flows**

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

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

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

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

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

643 However, this computation has cubic complexity in dimensionality for fully connected layers. By
 644 restricting our layers to normalizing flows [15] – bijective functions with fast log determinant ja-
 645 cobian computations, we can tractably optimize deep generative models with objectives that are a
 646 function of sample density, like entropy. Most of our analyses use real NVP [55], which have proven
 647 effective in our architecture searches, and have the advantageous features of fast sampling and fast
 648 density evaluation.

649 **A.1.4 Related work**

650 (To come)

651

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

653 (To come)

654

655 **A.2 Theoretical models**

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

658 **A.2.1 Stomatogastric ganglion**

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

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

660 The membrane potential of each neuron is affected by the leak, calcium, potassium, hyperpolariza-
661 tion, electrical and synaptic currents, respectively. The capacitance of the cell membrane was set to
662 $C_m = 1nF$. Each current is a function of the neuron's membrane potential x_m and the parameters
663 of the circuit such as g_{el} and g_{syn} , whose effect on the circuit is considered in the motivational
664 example of EPI in Fig. 1. Specifically, the currents are the difference in the neuron's membrane
665 potential and that current type's reversal potential multiplied by a conductance:

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

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

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

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

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

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

671 The reversal potentials were set to $V_{leak} = -40mV$, $V_{Ca} = 100mV$, $V_K = -80mV$, $V_{hyp} = -20mV$,
672 and $V_{syn} = -75mV$. The other conductance parameters were fixed to $g_{leak} = 1 \times 10^{-4}\mu S$. g_{Ca} ,
673 g_K , and g_{hyp} had different values based on fast, intermediate (hub) or slow neuron. Fast: $g_{Ca} =$
674 1.9×10^{-2} , $g_K = 3.9 \times 10^{-2}$, and $g_{hyp} = 2.5 \times 10^{-2}$. Intermediate: $g_{Ca} = 1.7 \times 10^{-2}$, $g_K = 1.9 \times 10^{-2}$,
675 and $g_{hyp} = 8.0 \times 10^{-3}$. Intermediate: $g_{Ca} = 8.5 \times 10^{-3}$, $g_K = 1.5 \times 10^{-2}$, and $g_{hyp} = 1.0 \times 10^{-2}$.

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

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

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

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

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

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

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

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

684 where we set $v_1 = 0mV$, $v_2 = 20mV$, $v_3 = 0mV$, $v_4 = 15mV$, $v_5 = 78.3mV$, $v_6 = 10.5mV$,
 685 $v_7 = -42.2mV$, $v_8 = 87.3mV$, $v_9 = 5mV$, and $v_{th} = -25mV$. These are the same parameter
 686 values used in [22].

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

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

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

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

694 Our original approach to measuring frequency was to take the max of the fast Fourier transform
 695 (FFT) of the simulated time series. There are a few key considerations here. One is resolution
 696 in frequency space. Each FFT entry will correspond to a signal frequency of $\frac{F_s k}{N}$, where N is
 697 the number of samples used for the FFT, $F_s = \frac{1}{dt}$, and $k \in [0, 1, \dots, N - 1]$. Our resolution is
 698 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

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

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

Network syncing, like all other emergent properties in this work, are defined by the emergent property statistics and values. The emergent property statistics are the first- and second-moments of the firing frequencies. The first moments are set to 0.55Hz, while the second moments are set to 0.025Hz^2 .

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

720 For EPI in Fig 2C, we used a real NVP architecture with two coupling layers. Each coupling layer
 721 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
 722 have shown the EPI optimization that converged with maximum entropy across 2 random seeds
 723 and augmented Lagrangian coefficient initializations of $c_0=0, 2$, and 5.

724 **A.2.2 Primary visual cortex**

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

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

726 Some neuron-types largely lack synaptic projections to other neuron-types [56], and it is popular
 727 to only consider a subset of the effective connectivities [23].

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

728 By consolidating information from many experimental datasets, Billeh et al. [40] produce estimates
 729 of the synaptic strength (in mV)

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

730 and connection probability

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

⁷³¹ Multiplying these connection probabilities and synaptic efficacies gives us an effective connectivity
⁷³² matrix:

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

⁷³³ From use the entries of this full effective connectivity matrix that are not considered to be ineffectual.

⁷³⁵ We look at how this four-dimensional nonlinear dynamical model of V1 responds to different inputs,
⁷³⁶ and compare the predictions of the linear response to the approximate posteriors obtained through
⁷³⁷ EPI. The input to the system is the sum of a baseline input $b = [1 \ 1 \ 1 \ 1]^T$ and a differential
⁷³⁸ input dh :

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

⁷³⁹ All simulations of this system had $T = 100$ time points, a time step $dt = 5\text{ms}$, and time constant
⁷⁴⁰ $\tau = 20\text{ms}$. And the system was initialized to a random draw $x(0)_i \sim \mathcal{N}(1, 0.01)$.

⁷⁴¹ We can describe the dynamics of this system more generally by

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

⁷⁴² where the input to each neuron is

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

⁷⁴³ Let $F_{ij} = \gamma_i \delta(i, j)$, where $\gamma_i = f'(u_i)$. Then, the linear response is

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

⁷⁴⁴ which is calculable by

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

⁷⁴⁵ The emergent property we considered was the first and second moments of the change in rate dx
⁷⁴⁶ between the baseline input $h = b$ and $h = b + dh$. We use the following notation to indicate that
⁷⁴⁷ the emergent property statistics were set to the following values:

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

- 748 In the final analysis for this model, we sweep the input one neuron at a time away from the mode
 749 of each inferred distributions $dh^* = z^* = \operatorname{argmax}_z \log q_\theta(z \mid \mathcal{B}(\alpha, 0.1))$. The differential responses
 750 $dx_{\alpha,ss}$ are examined at perturbed inputs $h = b + dh^* + \Delta h_\alpha u_\alpha$ where u_α is a unit vector in the
 751 dimension of α and $\Delta h_\alpha \in [-15, 15]$.
- 752 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
 753 initial seeds using an architecture of four coupling layers, each with two hidden layers of 10 units.
 754 We set $c_0 = 10^5$. The support of the learned distribution was restricted to $z_i \in [-5, 5]$.

755 **A.2.3 Superior colliculus**

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

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

758 $\epsilon = 0.05$ and $\zeta = 0.5$ control the position and shape of the nonlinearity, respectively.

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

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

760 The internal variables follow dynamics:

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

761 with time constant $\tau = 0.09s$ and Gaussian noise σdB controlled by the magnitude of $\sigma = 1.0$. The
 762 weight matrix has 8 parameters sW_P , sW_A , vW_{PA} , vW_{AP} , hW_P , hW_A , dW_{PA} , and dW_{AP} (Fig.
 763 4B).

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

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

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

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

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

766

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

767 a choice-period input,

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

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

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

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

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

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

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

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

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

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

776

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

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

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

779

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

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

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

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

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

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

⁷⁹⁹ **A.2.4 Rank-1 RNN**

⁸⁰⁰ Recent work establishes a link between RNN connectivity weights and the resulting dynamical
⁸⁰¹ responses of the network, using dynamic mean field theory (DMFT) [25]. Specifically, DMFT
⁸⁰² describes the properties of activity in infinite-size neural networks given a distribution on the
⁸⁰³ connectivity weights. In such a model, the connectivity of a rank-1 RNN (which was sufficient for
⁸⁰⁴ our task), has weight matrix W , which is the sum of a random component with strength determined
⁸⁰⁵ by g and a structured component determined by the outer product of vectors m and n :

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

⁸⁰⁶ where the activity x evolves as $\dot{x} = I(t)$ and $I(t)$ is some input, ϕ is the tanh nonlinearity, and $\chi_{ij} \sim \mathcal{N}(0, \frac{1}{N})$.
⁸⁰⁷ The entries of m and n are drawn from Gaussian distributions $m_i \sim \mathcal{N}(M_m, 1)$ and $n_i \sim \mathcal{N}(M_n, 1)$.
⁸⁰⁸ From such a parameterization, this theory produces consistency equations for the dynamic mean
⁸⁰⁹ field variables in terms of parameters like g , M_m , and M_n , which we study in Section 3.5. That
⁸¹⁰ is the dynamic mean field variables (e.g. the activity along a vector κ_v , the total variance
⁸¹¹ Δ_0 , structured variance Δ_∞ , and the chaotic variance Δ_T) are written as functions of one another
⁸¹² in terms of connectivity parameters. The values of these variables can be obtained using a
⁸¹³ nonlinear system of equations solver. These dynamic mean field variables are then cast as task-
⁸¹⁴ relevant variables with respect to the context of the provided inputs. Mastrogiuseppe et al. designed
⁸¹⁵ low-rank RNN connectivities via minimalist connectivity parameters to solve canonical tasks from
⁸¹⁶ behavioral neuroscience.

⁸¹⁷ We consider the DMFT equation solver as a black box that takes in a low-rank parameterization z
⁸¹⁸ (e.g. $z = [g \ M_m \ M_n]$) and outputs the values of the dynamic mean field variables, of which we
⁸¹⁹ cast κ_w and Δ_T as task-relevant variables μ_{post} and σ_{post}^2 in the Gaussian posterior conditioning
⁸²⁰ toy example. Importantly, the solution produced by the solver is differentiable with respect to the
⁸²¹ input parameters, allowing us to use DMFT to calculate the emergent property statistics in EPI
⁸²² to learn distributions on such connectivity parameters of RNNs that execute tasks.

⁸²³ Specifically, we solve for the mean field variables κ_w , κ_n , Δ_0 and Δ_∞ , where the readout is nominally
⁸²⁴ chosen to point in the unit orthant $w = [1 \ \dots \ 1]^\top$. The consistency equations for these variables

825 in the presence of a constant input $I(t) = y - (n - M_n)$ can be derived following [25] are

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

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

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

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

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

830 A.3 Supplementary Figures

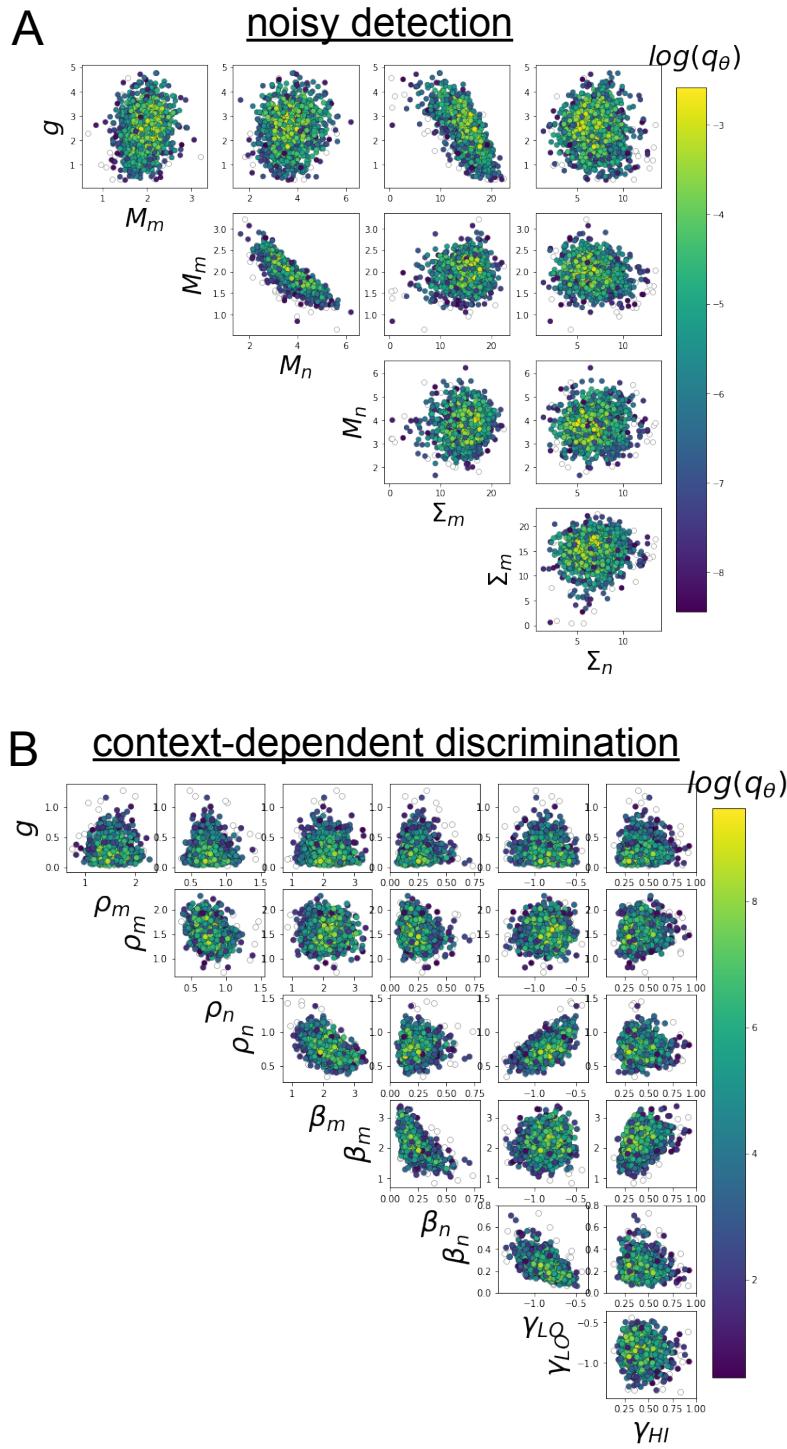


Fig. S1: A. EPI for rank-1 networks doing discrimination. B. EPI for rank-2 networks doing context-dependent discrimination.