

Interrogating theoretical models of neural computation with deep inference
Sean R. Bittner¹, Agostina Palmigiano¹, Alex T. Piet^{2,3,4}, Chunyu A. Duan⁵, Carlos D. Brody^{2,3,6},
Kenneth D. Miller¹, and John P. Cunningham⁷.

¹Department of Neuroscience, Columbia University,

²Princeton Neuroscience Institute,

³Princeton University,

⁴Allen Institute for Brain Science,

⁵Institute of Neuroscience, Chinese Academy of Sciences,

⁶Howard Hughes Medical Institute,

⁷Department of Statistics, Columbia University

¹ 1 Abstract

² A 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 experi-
⁴ mentally observed phenomenon – whether behavioral or a pattern of neural activity – and thus
⁵ can offer insights into neural computation. The operation of these mechanistic circuits, like all
⁶ models, critically depends on the choices of model parameters. A key process in circuit modeling
⁷ is then to identify the model parameters consistent with observed phenomena: to solve the inverse
⁸ problem. To solve challenging inverse problems modeling neural datasets, neuroscientists have used
⁹ statistical inference techniques to much success. However, most research in theoretical neuroscience
¹⁰ focuses on how computation emerges in biologically interpretable circuit models, and how the model
¹¹ parameters govern computation; it is not focused on the latent structure of empirical models of
¹² noisy experimental datasets. In this work, we present a novel technique that brings the power
¹³ and versatility of the probabilistic modeling toolkit to theoretical inverse problems. Our method
¹⁴ uses deep neural networks to learn parameter distributions with rich structure that have specific
¹⁵ computational properties in biologically relevant models. This methodology is explained through
¹⁶ a motivational example inferring conductance parameters in an STG subcircuit model. Then, with
¹⁷ RNNs of increasing size, we show that only EPI allows precise control over the behavior of inferred
¹⁸ parameters, and that EPI scales better in parameter dimension than alternative techniques. In the
¹⁹ remainder of this work, we explain novel theoretical insights through the examination of intricate
²⁰ parametric structure in complex circuit models. In a model of primary visual cortex with multiple

21 neuron-types, where analysis becomes untenable with each additional neuron-type, we discovered
22 how noise distributed across neuron-types governs the excitatory population. Finally, in a model
23 of superior colliculus, we identified and characterized two distinct regimes of connectivity that
24 facilitate switching between opposite tasks amidst interleaved trials. We also found that all task-
25 switching connectivities in this model reproduce behaviors from inactivation experiments, further
26 establishing this hypothesized circuit model. Beyond its scientific contribution, this work illustrates
27 the variety of analyses possible once deep learning is harnessed towards solving theoretical inverse
28 problems.

29 2 Introduction

30 The fundamental practice of theoretical neuroscience is to use a mathematical model to understand
31 neural computation, whether that computation enables perception, action, or some intermediate
32 processing. A neural circuit is systematized with a set of equations – the mechanistic model – and
33 these equations are motivated by biophysics, neurophysiology, and other conceptual considerations
34 [1–4]. The function of this system is governed by the choice of model *parameters*, which when
35 configured in a particular way, give rise to a measurable signature of a computation. The work
36 of analyzing a model then requires solving the inverse problem: given a computation of interest,
37 how can we reason about particular parameter configurations? The inverse problem is crucial for
38 reasoning about likely parameter values, uniquenesses and degeneracies, and predictions made by
39 the model [5, 6].

40 Consider the idealized practice: one carefully designs a model and analytically derives how compu-
41 tational properties determine model parameters. Seminal examples of this gold standard include
42 our field’s understanding of memory capacity in associative neural networks [7], chaos and au-
43 tocorrelation timescales in random neural networks [8], the paradoxical effect [9], and decision
44 making [10]. Unfortunately, as circuit models include more biological realism, theory via analytical
45 derivation becomes intractable. Still, we can gain insight into these complex models by identifying
46 the distribution of parameters that produce computations. By solving the inverse problem in this
47 way, scientific analysis of biologically realistic models is made possible [6, 11–14].

48 While theoretical neuroscience is concerned with how model parameters govern computational
49 properties, existing methodology for statistical inference in neuroscience [15–36] (see review, [37])
50 requires that parameters be conditioned on an explicit dataset. The scientific insight for a model

51 of computation is then limited by the quantity and quality of available neural data. Even with a
52 vast amount of high-quality recordings, neural data often reflect uninstructed behaviors [38–40],
53 and thus may only reflect the computation of interest amidst a sea of task-irrelevant factors. A
54 common alternative is to synthesize an explicit dataset that is exemplary of that computation, so
55 that the framework of statistical inference can be applied for parameter identification. In this case,
56 well-defined computational properties are being shoehorned into artificial datasets for the purpose
57 of methodological compatibility.

58 Another key challenge is that as models of computation become more complex, statistical inference
59 becomes intractable. Such mechanistic models in theoretical neuroscience are noisy systems of
60 differential equations that can only be sampled or realized through forward simulation [41, 42];
61 they lack a tractable likelihood function, which is necessary for statistical inference. Therefore, the
62 most popular approaches to parameter inference in mechanistic models have been simulation-based
63 inference methods [43, 44], in which reasonable parameters are obtained via simulation and rejection.
64 A new class of techniques [45–47] use deep learning to improve upon traditional simulation-based
65 inference approaches. However, to use these methods in theoretical neuroscience, we must represent
66 computation with an explicit dataset in some way. Theorists are therefore barred from using the
67 probabilistic modeling toolkit for science with circuit models, unless they reformulate their inverse
68 problem into a framework for observational datasets.

69 To address the methodological incongruity between explicit datasets and emergent properties, we
70 present a statistical inference method for conditioning parameters of neural circuit models directly
71 on computation. In this work, we define computation by an emergent property, which is a statistical
72 description of the phenomena to be produced by the neural circuit model. In emergent property
73 inference (EPI), we infer the distribution of model parameters that produce this emergent property.
74 With EPI, parameters are conditioned directly on an implicit dataset defined by the computation
75 of interest. By using recent optimization techniques [48], EPI uses deep learning to make rich,
76 flexible approximations to the parameter distributions [49], the structure of which reveals scientific
77 insight about how parameters govern the emergent property.

78 Equipped with this method, we prove out the potential of EPI by demonstrating its capabilities and
79 presenting novel theoretical findings borne from its analysis. First, we show EPI’s ability to handle
80 mechanistic models using a classic model of parametric degeneracy in biology: the stomatogastric
81 ganglion [50, 51]. Then, we show EPI’s scalability to high dimensional parameter distributions by
82 inferring connectivities of recurrent neural networks (RNNs) that exhibit stable, yet amplified re-

sponses – a hallmark of neural responses throughout the brain [52–54]. In a model of primary visual cortex (V1) [55, 56] with different neuron-types, we show that the equation for excitatory variability become analytically intractable as more populations are added. Strikingly, the way in which noisy inputs across neuron-types governs excitatory variability is salient in the visualized structure of the EPI inferred parameter distribution. Finally, we investigated the possible connectivities of superior colliculus (SC) that allow execution of different tasks on interleaved trials [57]. EPI discovered a rich distribution containing two connectivity regimes with different solution classes. We queried the deep probability distribution learned by EPI to produce a mechanistic understanding of cortical responses in each regime. Intriguingly, all inferred connectivities reproduced results from optogenetic inactivation experiments in this behavioral paradigm – emergent phenomena that EPI was not conditioned upon. These theoretical insights afforded by EPI illustrate the value of deep inference for the interrogation of neural circuit models.

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 [58]. 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 computation [12, 50].

A subcircuit model of the STG [51] is shown schematically in Figure 1A. The fast population (f_1 and f_2) represents the subnetwork generating the pyloric rhythm and the slow population (s_1 and s_2) represents the subnetwork of the gastric mill rhythm. The two fast neurons mutually inhibit one another, and spike at a greater frequency than the mutually inhibiting slow neurons. The hub neuron (hub) couples with either the fast or slow population, or both depending on modulatory conditions. The jagged connections indicate electrical coupling having electrical conductance g_{el} , smooth connections in the diagram are inhibitory synaptic projections having strength g_{synA} onto the hub neuron, and $g_{synB} = 5\text{nS}$ for mutual inhibitory connections. Note that the behavior of this model will be critically dependent on its parameterization – the choices of conductance parameters

113 $\mathbf{z} = [g_{el}, g_{synA}]$.

114 Second, once the model is selected, one must specify what the model should produce. In this STG
115 model, we are concerned with neural spiking frequency, which emerges from the dynamics of the
116 circuit model 1B. An emergent property studied by Gutierrez et al. of this stochastic model is the
117 hub neuron firing at an intermediate frequency between the intrinsic spiking rates of the fast and
118 slow populations. This emergent property is shown in Figure 1C at an average frequency of 0.55Hz.
119 Our notion of intermediate hub frequency is not strictly 0.55Hz, but also moderate deviations of
120 this frequency between the fast (.35Hz) and slow (.68Hz) frequencies, which are quantified in
121 the emergent property with variance 0.025^2Hz^2 .

122 Third, the model parameters producing these outputs are inferred. To infer the STG parameters of
123 intermediate hub frequency with existing methodology, we need an explicit dataset: experimentally
124 recorded or synthesized. By precisely quantifying the emergent property of interest as a statistical
125 feature of the model, we use EPI to condition directly on this emergent property. EPI learns a
126 probability distribution of model parameters constrained to produce the emergent property. In
127 this last step lies the opportunity for a shift away from a dataset-oriented representation of model
128 output towards that of an implicit dataset, where the only structure is the emergent property of
129 interest.

130 Before presenting technical details (in the following section), let us understand emergent property
131 inference schematically. EPI (Fig. 1D) takes, as input, the model and the specified emergent
132 property, and as its output, produces the parameter distribution EPI (Fig. 1E). This distribution –
133 represented for clarity as samples from the distribution – is a parameter distribution that produces
134 the emergent property. Scientifically, we can use this parameter distribution to efficiently generate
135 many parameters producing the emergent property or analyze the structure of the distribution
136 which informs how model parameters govern the emergent property.

137 3.2 A deep generative modeling approach to emergent property inference

138 Emergent property inference (EPI) formalizes the three-step procedure of the previous section with
139 deep probability distributions. First, as is typical, we consider the model as a coupled set of
140 differential equations. In this STG example, the model activity $\mathbf{x} = [x_{f1}, x_{f2}, x_{hub}, x_{s1}, x_{s2}]$ is the
141 membrane potential for each neuron, which evolves according to the biophysical conductance-based



Figure 1: Emergent property inference (EPI) in the stomatogastric ganglion. **A.** Conductance-based biophysical model of the STG subcircuit. **B.** Spiking frequency $\omega(\mathbf{x}; \mathbf{z})$ is an emergent property statistic. Simulated at $g_{el} = 4.5\text{nS}$ and $g_{synA} = 3\text{nS}$. **C.** The emergent property of intermediate hub frequency. Simulated activity traces are colored by $\log q_\theta(\mathbf{z} | \mathcal{X})$ of generating parameters. (Panel E). **D.** For a choice of model and emergent property, emergent property inference (EPI) learns a deep probability distribution of parameters \mathbf{z} . **E.** The EPI distribution producing intermediate hub frequency. Samples are colored by log probability density. Contours of hub neuron frequency error are shown at levels of .525, .53,575 Hz (dark to light gray away from mean). Dimension of sensitivity \mathbf{v}_1 (solid) and degeneracy \mathbf{v}_2 (dashed). **F** (Top) The predictive distribution of EPI. The black and gray dashed lines show the mean and two standard deviations according the emergent property. (Bottom) Simulations at the starred parameter values.

142 equation:

$$C_m \frac{d\mathbf{x}(t)}{dt} = -h(\mathbf{x}(t); \mathbf{z}) + d\mathbf{B} \quad (1)$$

143 where $C_m = 1\text{nF}$, and \mathbf{h} is a sum of the leak, calcium, potassium, hyperpolarization, electrical, and
144 synaptic currents, all of which have their own complicated dependence on activity \mathbf{x} and parameters
145 $\mathbf{z} = [g_{el}, g_{synA}]$, and $d\mathbf{B}$ is white gaussian noise [51, 59] (see Section 5.2.1 for more detail).

146 Second, we stipulate that our model should produce the emergent property of “intermediate hub
147 frequency” (Figure 1C). We stipulate that the hub neuron’s spiking frequency – denoted $\omega_{\text{hub}}(\mathbf{x})$
148 is close to a frequency of 0.55Hz, between that of the slow and fast frequencies. Mathematically,
149 we define this emergent property with two statistical constraints: that the mean hub frequency is
150 0.55Hz,

$$\mathbb{E}_{\mathbf{z}, \mathbf{x}} [\omega_{\text{hub}}(\mathbf{x}; \mathbf{z})] = 0.55 \quad (2)$$

151 and that the variance of the hub frequency is moderate

$$\text{Var}_{\mathbf{z}, \mathbf{x}} [\omega_{\text{hub}}(\mathbf{x}; \mathbf{z})] = 0.025^2. \quad (3)$$

152 The hub neuron frequency is constrained over the distribution of parameters \mathbf{z} and the distribution
153 of the data \mathbf{x} that those parameters produce. Formally, the emergent property is the collection of
154 these two constraints

$$\mathcal{X} : \mathbb{E}_{\mathbf{z}, \mathbf{x}} [\omega_{\text{hub}}(\mathbf{x}; \mathbf{z})] = 0.55, \quad \text{Var}_{\mathbf{z}, \mathbf{x}} [\omega_{\text{hub}}(\mathbf{x}; \mathbf{z})] = 0.025^2. \quad (4)$$

155 In general, an emergent property is a collection of first-, second- and higher moments of statistics
156 that together define the phenomena.

157 Third, we perform emergent property inference: we find a distribution over parameter configu-
158 rations \mathbf{z} that produces the emergent property; in other words, they obey the constraints intro-
159 duced in Equation 4. This distribution will be chosen from a family of probability distributions
160 $\mathcal{Q} = \{q_{\boldsymbol{\theta}}(\mathbf{z}) : \boldsymbol{\theta} \in \Theta\}$, defined by a deep neural network [49, 60, 61] (Figure 1D, EPI box). Deep
161 probability distributions map a simple random variable \mathbf{z}_0 through a deep neural network with
162 weights and biases $\boldsymbol{\theta}$ to parameters $\mathbf{z} = g_{\boldsymbol{\theta}}(\mathbf{z}_0)$ to a suitably complicated distribution (see Section
163 5.1.2 for more details). Many distributions in \mathcal{Q} will respect the emergent property constraints,
164 so we select the most random (highest entropy) distribution, which is the same choice made in
165 variational bayesian methods (see Section 5.1.6). In EPI optimization, stochastic gradient steps in
166 $\boldsymbol{\theta}$ are taken such that entropy is maximized, and the emergent property \mathcal{X} is produced (see Section

167 5.1) The inferred EPI distribution is denoted $q_{\theta}(\mathbf{z} \mid \mathcal{X})$, since it is conditioned upon emergent
168 property \mathcal{X} . This is meant to share the same notation as a posterior distribution $q_{\theta}(\mathbf{z} \mid \mathbf{x})$ that is
169 conditioned upon an explicit dataset.

170 EPI produces parameter distributions that can be queried for scientific insight. The modes of
171 $q_{\theta}(\mathbf{z} \mid \mathcal{X})$ indicate parameter choices exemplary of the emergent property (Fig. 1E yellow star). As
172 probability in the EPI inferred distribution decreases, the emergent property deteriorates. Perturb-
173 ing \mathbf{z} along a dimension in which $q_{\theta}(\mathbf{z} \mid \mathcal{X})$ does not change will not disturb the emergent property,
174 making this parameter combination *degenerate* with respect to the emergent property. In contrast,
175 if \mathbf{z} is perturbed along a dimension that strongly decreases $q_{\theta}(\mathbf{z} \mid \mathcal{X})$, we call that parameter com-
176 bination *sensitive*. By querying the second order derivative (Hessian) of $\log q_{\theta}(\mathbf{z} \mid \mathcal{X})$ at a mode,
177 we can quantitatively identify how sensitive (or robust) each eigenvector is by its eigenvalue; the
178 more negative the eigenvalue, the more sensitive. Indeed, samples equidistant from the mode along
179 these EPI-identified dimensions of sensitivity (v_1 , smaller eigenvalue) and robustness (v_2 , greater
180 eigenvalue) (Fig. 1E, arrows) agree with error contours (Fig. 1E contours) and have diminished or
181 preserved hub frequency, respectively (Fig. 1F activity traces). Once an EPI distribution has been
182 inferred, this Hessian calculation requires trivial computation (see Section 5.2.4).

183 In the following sections, we demonstrate EPI on three neural circuit models across ranges of
184 biological realism, neural system function, and network scale. First, we demonstrate the superior
185 scalability of EPI compared to alternative techniques by inferring high-dimensional distributions of
186 recurrent neural network (RNN) connectivities that exhibit amplified, yet stable responses. Also
187 in this RNN example, we emphasize that EPI is the only technique that controls the predictions
188 made by the inferred parameter distribution. Next, in a model of primary visual cortex [55,56], we
189 show how EPI captures a curved manifold of parametric degeneracy, revealing how input variability
190 across neuron types affects the excitatory population. Finally, in a model of superior colliculus [57],
191 we used EPI to capture multiple parametric regimes of task switching, and queried the dimensions
192 of sensitivity ($\mathbf{v}_1(\mathbf{z})$) to mechanistically characterize each regime.

193 3.3 Scaling inference of RNN connectivity with EPI

194 Transient amplification is a hallmark of neural activity throughout cortex, and is often thought to be
195 intrinsically generated by recurrent connectivity in the responding cortical area [52–54]. It has been
196 shown that to generate such amplified, yet stabilized responses, the connectivity of RNNs must be
197 non-normal [52,62], and satisfy additional constraints [63]. In theoretical neuroscience, RNNs are

198 optimized and then examined to show how dynamical systems could execute a given computation
 199 [64, 65], but such biologically realistic constraints on connectivity are ignored during optimization
 200 for practical reasons. In general, access to distributions of connectivity adhering to theoretical
 201 criteria like stable amplification, chaotic fluctuations [8], or low tangling [66] would add scientific
 202 value and context to existing research with RNNs. Here, we use EPI to learn RNN connectivities
 203 producing stable amplification, and demonstrate the superior scalability and efficiency of EPI to
 204 alternative approaches.

205 We consider a rank-2 RNN with N neurons having connectivity $W = UV^\top$ and dynamics

$$\tau \dot{\mathbf{x}} = -\mathbf{x} + W\mathbf{x}, \quad (5)$$

206 where $U = [\mathbf{u}_1 \ \mathbf{u}_2] + g\chi^{(U)}$, $V = [\mathbf{v}_1 \ \mathbf{v}_2] + g\chi^{(V)}$, $\mathbf{u}_1, \mathbf{u}_2, \mathbf{v}_1, \mathbf{v}_2 \in [-1, 1]^N$, and $\chi_{i,j}^{(U)}, \chi_{i,j}^{(V)} \sim$
 207 $\mathcal{N}(0, 1)$. We infer connectivity parameterizations $\mathbf{z} = [\mathbf{u}_1^\top, \mathbf{u}_2^\top, \mathbf{v}_1^\top, \mathbf{v}_2^\top]^\top$ that produce stable ampli-
 208 fication. Two conditions are necessary and sufficient for RNNs to exhibit stable amplification [63]:
 209 $\text{real}(\lambda_1) < 1$ and $\lambda_1^s > 1$, where λ_1 is the eigenvalue of W with greatest real part and λ^s is the max-
 210 imum eigenvalue of $W^s = \frac{W+W^\top}{2}$. RNNs with $\text{real}(\lambda_1) = 0.5 \pm 0.5$ and $\lambda_1^s = 1.5 \pm 0.5$ will be stable
 211 with modest decay rate ($\text{real}(\lambda_1)$ close to its upper bound of 1) and exhibit modest amplification
 212 (λ_1^s close to its lower bound of 1). EPI can naturally condition on this emergent property

$$\begin{aligned} \mathcal{X} : \mathbb{E}_{\mathbf{z}, \mathbf{x}} \begin{bmatrix} \text{real}(\lambda_1) \\ \lambda_1^s \end{bmatrix} &= \begin{bmatrix} 0.5 \\ 1.5 \end{bmatrix} \\ \text{Var}_{\mathbf{z}, \mathbf{x}} \begin{bmatrix} \text{real}(\lambda_1) \\ \lambda_1^s \end{bmatrix} &= \begin{bmatrix} 0.25^2 \\ 0.25^2 \end{bmatrix}, \end{aligned} \quad (6)$$

213 under the notion that variance constraints with standard deviation 0.25 predicate that the vast
 214 majority of samples (those within two standard deviations) are within the specified ranges.

215 For comparison, we infer the parameters \mathbf{z} likely to produce stable amplification using two al-
 216 ternative simulation-based inference approaches. We ran sequential Monte Carlo approximate
 217 Bayesian computation (SMC-ABC) [43] and sequential neural posterior estimation (SNPE) [45]
 218 with observation $\mathbf{x}_0 = \boldsymbol{\mu}$. SMC-ABC is a rejection sampling approach that SMC techniques to
 219 improve efficiency, and SNPE approximates posteriors with deep probability distributions using
 220 a two-network architecture (see Section 5.1.1). Unlike EPI, these statistical inference techniques
 221 do not constrain the statistics of the predictive distribution, and these predictions of the inferred
 222 posteriors are typically affected by model characteristics (e.g. N and g , Fig. 11). To compare the



Figure 2: **A.** Wall time of EPI (blue), SNPE (orange), and SMC-ABC (green) to converge on RNN connectivities producing stable amplification. Each dot shows convergence time for an individual random seed. For reference, the mean wall time for EPI to achieve its full constraint convergence (means and variances) is shown (blue line). **B.** Simulation count of each algorithm to achieve convergence. Same conventions as A. **C.** The predictive distributions of connectivities inferred by EPI (blue), SNPE (orange), and SMC-ABC (green), with reference to $\mathbf{x}_0 = \mu$ (gray star). **D.** Simulations of networks inferred by each method ($\tau = 100ms$). Each trace (15 per algorithm) corresponds to simulation of one z . (Below) Ratio of obtained samples producing stable amplification, monotonic decay, and instability.

efficiency of these different techniques, we measured the time and number of simulations necessary for the distance of the predictive mean to be less than 0.5 from $\mu = \mathbf{x}_0$ (see Section 5.3).

As the number of neurons N in the RNN is scaled, and thus the dimension of the parameter space $\mathbf{z} \in [-1, 1]^{4N}$, we see that EPI converges at greater speed and at greater dimension than SMC-ABC and SNPE (Fig. 2A). It also becomes most efficient to use EPI in terms of simulation count at $N = 50$ (Fig. 2B). It is well known that ABC techniques struggle in dimensions greater than about 30 [67], yet we were careful to assess the scalability of the more comparable approach SNPE. Between EPI and SNPE, we closely controlled the number of parameters in deep probability distributions by dimensionality (Fig. 10), and tested more aggressive SNPE hyperparameterizations when SNPE failed to converge (Fig. 12). From this analysis, we see that deep inference techniques EPI and SNPE are far more amenable to inference of high dimensional parameter distributions than rejection sampling techniques like SMC-ABC, and that EPI outperforms SNPE in both criteria in high dimensions.

No matter the number of neurons, EPI always produces connectivity distributions with mean and variance of $\text{real}(\lambda_1)$ and λ_1^s according to \mathcal{X} (Fig. 2C, blue). For the dimensionalities in which SMC-ABC is tractable, the inferred parameters are concentrated and offset from \mathbf{x}_0 (Fig. 2C, green). When using SNPE the predictions of the inferred parameters are highly concentrated at some RNN sizes and widely varied in others (Fig. 2C, orange). We see these properties reflected in simulations from the inferred distributions: EPI produces a consistent variety of stable, amplified activity norms $|r(t)|$, SMC-ABC produces a limited variety of responses, and the changing variety of responses from SNPE emphasizes the control of EPI on parameter predictions.

EPI outperforms SNPE in high dimensions by using gradient information (from $\nabla_{\mathbf{z}} f(\mathbf{x}; \mathbf{z}) = \nabla_{\mathbf{z}} [\text{real}(\lambda_1), \lambda_1^s]^{\top}$) on each iteration of optimization. This agrees with recent speculation that such gradient information could improve the efficiency of simulation-based inference techniques [68]. Since gradients of the emergent property statistics are necessary in EPI optimization, gradient tractability is a key criteria when determining the suitability of a simulation-based inference technique. Evidenced by this analysis, EPI is a clear choice for inferring high dimensional parameter distributions when the emergent property gradient is efficiently calculated. This can be invaluable for understanding how RNNs produce complex emergent phenomena. Even with a high degree of biophysical realism and expensive emergent property gradients, EPI was run successfully on intermediate hub frequency in a 5-neuron subcircuit model of the STG (Section 3.1). However, conditioning on the pyloric rhythm [69] in a model of the pyloric subnetwork model [12] proved to

255 be prohibitive with EPI. The pyloric subnetwork requires many time steps for simulation and many
 256 key emergent property statistics (e.g. burst duration and phase gap) are not calculable or easily
 257 approximated with differentiable functions. In such cases, gradient-free approaches like SNPE have
 258 proved to be a powerful option [45]. In the next two sections, we use EPI for novel scientific insight
 259 by examining the structure of inferred distributions.

260 **3.4 EPI reveals how noisy input across neuron-types governs excitatory vari-
 261 ability in a V1 model**

262 Dynamical models of excitatory (E) and inhibitory (I) populations with supralinear input-output
 263 function have succeeded in explaining a host of experimentally documented phenomena. In a regime
 264 characterized by inhibitory stabilization of strong recurrent excitation, these models give rise to
 265 paradoxical responses [9], selective amplification [52, 62], surround suppression [70] and normaliza-
 266 tion [71]. Despite their strong predictive power, E-I circuit models rely on the assumption that
 267 inhibition can be studied as an indivisible unit. However, experimental evidence shows that inhibi-
 268 tion is composed of distinct elements – parvalbumin (P), somatostatin (S), VIP (V) – composing
 269 80% of GABAergic interneurons in V1 [72–74], and that these inhibitory cell types follow specific
 270 connectivity patterns (Fig. 3A) [75]. While research has shown that V1 only shares specific dimen-
 271 sions of neuronal variability with downstream areas [76], the role played by recurrent dynamics and
 272 the connectivity across neuron-type populations is not understood. Here, in a model of V1 with
 273 biologically realistic connectivity, we use EPI to show how the structure of input across neuron
 274 types affects the variability of the excitatory population – the population largely responsible for
 275 projecting to other brain areas [77].

276 We considered response variability of a nonlinear dynamical V1 circuit model (Fig. 3A) with a
 277 state comprised of each neuron-type population’s rate $\mathbf{x} = [x_E, x_P, x_S, x_V]^\top$. Each population
 278 receives recurrent input $W\mathbf{x}$, where W is the effective connectivity matrix (see Section 5.4). Each
 279 population also experiences an external input \mathbf{h} , which determines population rate via supralinear
 280 nonlinearity $\phi(\cdot) = [\cdot]_+^2$. There is also an additive noisy input ϵ parameterized by variances for
 281 each neuron type population $\sigma^2 = [\sigma_E^2, \sigma_P^2, \sigma_S^2, \sigma_V^2]$. This noise has a slower dynamical timescale
 282 $\tau_{\text{noise}} > \tau$ than the population rate, allowing fluctuations around a stimulus-dependent steady-state
 283 (Fig. 3B). This model is the stochastic stabilized supralinear network (SSSN) [78]

$$\tau \frac{d\mathbf{x}}{dt} = -\mathbf{x} + \phi(W\mathbf{x} + \mathbf{h} + \epsilon). \quad (7)$$

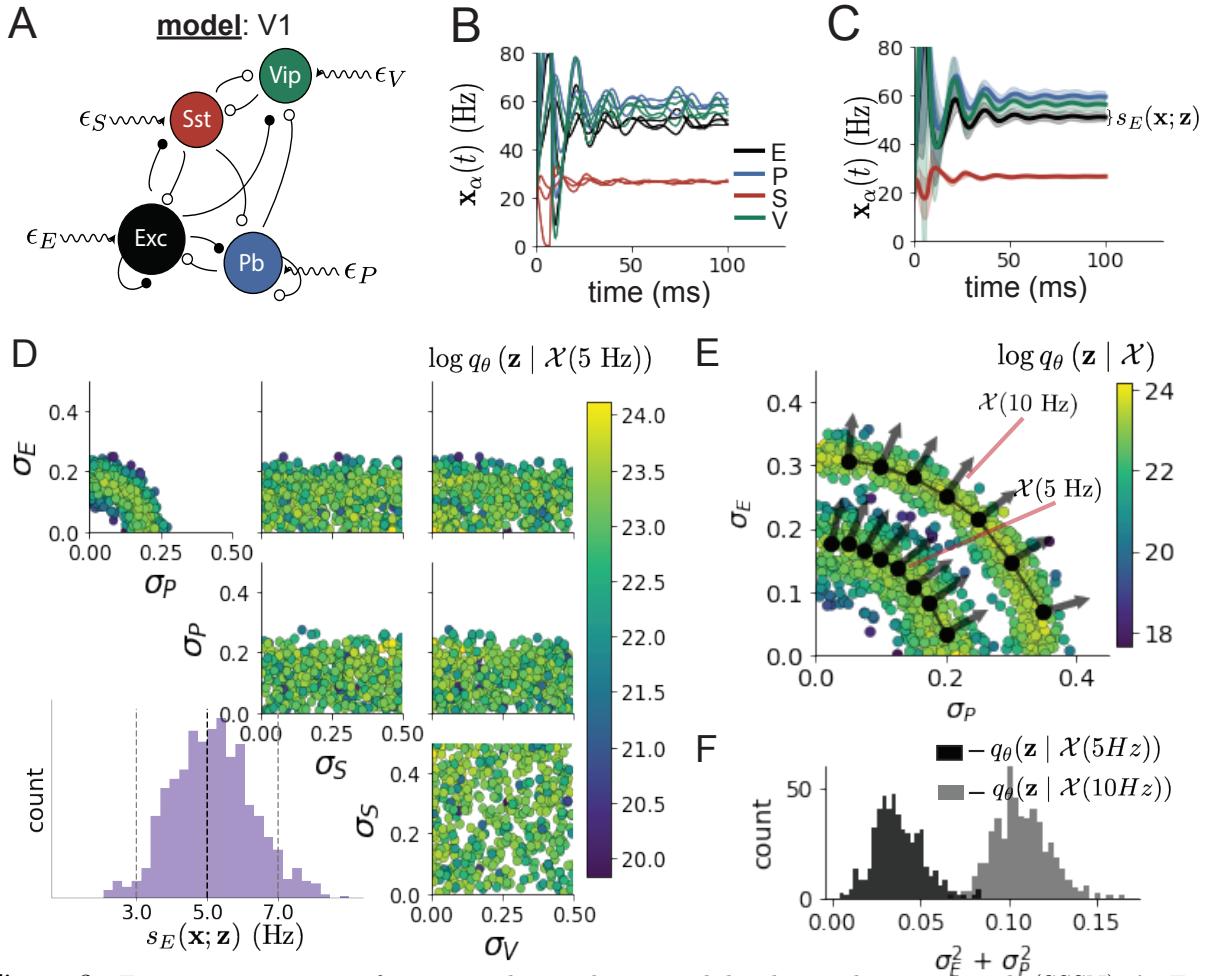


Figure 3: Emergent property inference in the stochastic stabilized supralinear network (SSSN). **A.** Four-population model of primary visual cortex with excitatory (black), parvalbumin (blue), somatostatin (red), and VIP (green) neurons (excitatory and inhibitory projections filled and unfilled, respectively). Some neuron-types largely do not form synaptic projections to others ($|W_{\alpha_1, \alpha_2}| < 0.025$). Each neural population receives a baseline input \mathbf{h}_b , and the E- and P-populations also receive a contrast-dependent input \mathbf{h}_c . Additionally, each neural population receives a slow noisy input ϵ . **B.** Transient network responses of the SSSN model. Traces are independent trials with varying initialization $\mathbf{x}(0)$ and noise ϵ . **C.** Mean (solid line) and standard deviation $s_E(\mathbf{x}; \mathbf{z})$ (shading) across 100 trials. **D.** EPI distribution of noise parameters \mathbf{z} conditioned on E-population variability. The EPI predictive distribution of $s_E(\mathbf{x}; \mathbf{z})$ is shown on the bottom-left. **E.** (Top) Enlarged visualization of the σ_E - σ_P marginal distribution of EPI $q_\theta(\mathbf{z} | \mathcal{X}(5 \text{ Hz}))$ and $q_\theta(\mathbf{z} | \mathcal{X}(10 \text{ Hz}))$. Each black dot shows the mode at each σ_P . The arrows show the most sensitive dimensions of the Hessian evaluated at these modes. **F.** The predictive distributions of $\sigma_E^2 + \sigma_P^2$ of each inferred distribution $q_\theta(\mathbf{z} | \mathcal{X}(5 \text{ Hz}))$ and $q_\theta(\mathbf{z} | \mathcal{X}(10 \text{ Hz}))$.

generalized to have multiple inhibitory neuron types, and introduces stochasticity to previous four neuron-type models of V1 [55]. Stochasticity and inhibitory multiplicity introduce substantial complexity to mathematical derivations (see Section 5.4.4) motivating the treatment of this model with EPI. Here, we consider fixed weights W and input \mathbf{h} [56], and study the effect of input variability $\mathbf{z} = [\sigma_E, \sigma_P, \sigma_S, \sigma_V]^\top$ on excitatory variability.

We quantify levels y of E-population variability with the emergent property

$$\begin{aligned}\mathcal{X}(y) &: \mathbb{E}_{\mathbf{z}} [s_E(\mathbf{x}; \mathbf{z})] = y \\ \text{Var}_{\mathbf{z}} [s_E(\mathbf{x}; \mathbf{z})] &= 1\text{Hz}^2,\end{aligned}\tag{8}$$

where $s_E(\mathbf{x}; \mathbf{z})$ is the standard deviation of the stochastic E-population response about its steady state (Fig. 3C). In the following analyses, we compare levels of 5Hz and 10Hz, and select 1 Hz² variance such that the two emergent properties do not overlap in $s_E(\mathbf{z}; \mathbf{x})$.

First, we ran EPI to obtain parameter distribution $q_{\theta}(\mathbf{z} | \mathcal{X}(5 \text{ Hz}))$ producing E-population variability around 5 Hz (Fig. 3D). From the marginal distribution of σ_E and σ_P (Fig. 3D, top-left), we can see that $s_E(\mathbf{x}; \mathbf{z})$ is sensitive to various combinations of σ_E and σ_P . Alternatively, both σ_S and σ_V are degenerate with respect to $s_E(\mathbf{x}; \mathbf{z})$ evidenced by the high variability in those dimensions (Fig. 3D, bottom-right). Together, these observations imply a curved path with respect to $s_E(\mathbf{x}; \mathbf{z})$ of 5 Hz, which is indicated by the modes along σ_P (Fig. 3E).

Figure 3E suggests a quadratic relationship in E-population fluctuations and the standard deviation of E- and P-population input; as the square of either σ_E or σ_P increases, the other compensatorily decreases to preserve the level of $s_E(\mathbf{x}; \mathbf{z})$. This quadratic relationship is preserved at greater level of E-population variability $\mathcal{X}(10 \text{ Hz})$ (Fig. 3E). Indeed, the sum of squares of σ_E and σ_P is larger in $q_{\theta}(\mathbf{z} | \mathcal{X}(10 \text{ Hz}))$ than $q_{\theta}(\mathbf{z} | \mathcal{X}(5 \text{ Hz}))$ (Fig 3F, $p < 1 \times 10^{-10}$), while the sum of squares of σ_S and σ_V are not significantly different in the two EPI distributions (Fig. 15, $p = .40$), in which parameters were bounded from 0 to 0.5. The strong interactive influence of E- and P-population input variability on excitatory variability is intriguing, since this circuit exhibits a paradoxical effect in the P-population (and no other inhibitory types) (Fig. 15), meaning that the E-population is P-stabilized. Future research may uncover a link between the population of network stabilization and compensatory interactions governing excitatory variability.

EPI revealed the quadratic dependence of excitatory variability on input variability to the E- and P-populations, as well as its independence to input from the other two inhibitory populations. We show that with each neuron-type population added to this E-I model, calculations of excitatory

313 variability with respect to noise parameters become unruly and challenging to work with (see
 314 Section 5.4.4). This emphasizes the value of streamlined methods for gaining understanding about
 315 theoretical models when mathematical analysis becomes onerous or impractical. While EPI can
 316 be used to investigate fundamental aspects of sensory processing, in the next section, we use the
 317 probabilistic tools of EPI to identify and characterize two distinct parametric regimes of a neural
 318 circuit executing a computation, and then relate these insights to behavioral experiments.

319 3.5 EPI identifies two regimes of rapid task switching

320 It has been shown that rats can learn to switch from one behavioral task to the next on randomly
 321 interleaved trials [79], and an important question is what types of neural connectivity allow this
 322 ability. In this experimental setup, rats were explicitly cued on each trial to either orient towards
 323 a visual stimulus in the Pro (P) task or orient away from a visual stimulus in the Anti (A) task
 324 (Fig. 4A). Neural recordings in superior colliculus (SC) exhibited two populations of neurons that
 325 represented task context (Pro or Anti). Furthermore, Pro/Anti neurons in each hemisphere were
 326 strongly correlated with the animal’s decision [57]. These results motivated a model of SC that
 327 is a four-population dynamical system with functionally-defined neuron-types. Here, our goal is
 328 to understand how connectivity in this circuit model governs the ability to perform rapid task
 329 switching: to respond with satisfactory accuracy in both tasks on randomly interleaved trials.

330 In this SC model, there are Pro- and Anti-populations in each hemisphere (left (L) and right
 331 (R)) with activity variables $\mathbf{x} = [x_{LP}, x_{LA}, x_{RP}, x_{RA}]^\top$. The connectivity of these populations is
 332 parameterized by self sW , vertical vW , diagonal dW and horizontal hW connections (Fig. 4B). The
 333 input \mathbf{h} is comprised of a positive cue-dependent signal to the Pro or Anti populations, a positive
 334 stimulus-dependent input to either the Left or Right populations, and a choice-period input to the
 335 entire network (see Section 5.5.1). Model responses are bounded from 0 to 1 as a function ϕ of an
 336 internal variable \mathbf{u}

$$\begin{aligned}\tau \frac{d\mathbf{u}}{dt} &= -\mathbf{u} + W\mathbf{x} + \mathbf{h} + d\mathbf{B} \\ \mathbf{x} &= \phi(\mathbf{u}).\end{aligned}\tag{9}$$

337 The model responds to the side with greater Pro neuron activation; e.g. the response is left if
 338 $x_{LP} > x_{RP}$ at the end of the trial. Here, we use EPI to determine the network connectivity
 339 $\mathbf{z} = [sW, vW, dW, hW]^\top$ that produces rapid task switching.
 340 Rapid task switching is formalized mathematically as an emergent property with two statistics:



Figure 4: **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 EPI inferred distribution of rapid task switching networks. Red and purple stars indicate modes \mathbf{z}^* of each connectivity regime. Sensitivity vectors $\mathbf{v}_1(\mathbf{z}^*)$ are shown by arrows. (Bottom-left) EPI predictive distribution of task accuracies. **D.** The connectivity regimes have different responses to perturbation. (Top) Mean and standard error ($N_{\text{test}} = 25$) of accuracy with respect to perturbation along the sensitivity dimension of each mode \mathbf{z}^* . (Middle) Same with perturbation in the dimension of increasing λ_{task} (\mathbf{v}_{task}). (Bottom) Same with perturbation in the dimension of increasing λ_{diag} (\mathbf{v}_{diag}).

accuracy in the Pro task $p_P(\mathbf{x}; \mathbf{z})$ and Anti task $p_A(\mathbf{x}; \mathbf{z})$. We stipulate that accuracy be on average .75 in each task with variance .075²

$$\begin{aligned} \mathcal{X} : \mathbb{E}_{\mathbf{z}} \begin{bmatrix} p_P(\mathbf{x}; \mathbf{z}) \\ p_A(\mathbf{x}; \mathbf{z}) \end{bmatrix} &= \begin{bmatrix} .75 \\ .75 \end{bmatrix} \\ \text{Var}_{\mathbf{z}} \begin{bmatrix} p_P(\mathbf{x}; \mathbf{z}) \\ p_A(\mathbf{x}; \mathbf{z}) \end{bmatrix} &= \begin{bmatrix} .075^2 \\ .075^2 \end{bmatrix}. \end{aligned} \quad (10)$$

75% accuracy is a realistic level of performance in each task, and with the chosen variance, inferred models will not exhibit fully random responses (50%), nor perfect performance (100%).

The EPI inferred distribution (Fig. 4C) produces Pro and Anti task accuracies (Fig. 4C, middle-left) consistent with rapid task switching (Equation 10). This parameter distribution has intricate structure, that is not captured well by simple linear correlations (Fig. 17). Specifically, the shape of the EPI distribution changes dramatically on different sides of parameter space. This is most saliently pointed out in the marginal distribution of $sW-hW$ (Fig. 4C top-right), where anticorrelation between sW and hW switches to correlation with decreasing sW . The two regimes produce different types of responses in the Pro and Anti tasks (Fig. SC2). Not only has EPI captured this complicated distribution of connectivities producing rapid task switching, we can query the EPI distribution $q_{\theta}(\mathbf{z} | \mathcal{X})$ to understand these two parametric regimes of SC connectivity.

To distinguish these two regimes, we use the EPI distribution to identify two sets of modes. By fixing hW to different values and doing gradient ascent on $\log q_{\theta}(\mathbf{z} | \mathcal{X})$, we arrive at two solutions $\mathbf{z}^*(hW_{\text{fixed}}, r)$ where regime $r \in [1, 2]$, and regime 1 is that of greater sW (see Section 5.5.4). As hW_{fixed} increases, the modes coalesce to intermediate parameters reflecting a transition between the two sets of modes (Fig. 20 top). By using EPI to connect these two regimes through this transitional region of parameter space, we can explore what distinguishes the two regimes by stepping from the prototypical connectivity of regime 1 to that of regime 2.

While the connectivities gradually coalesce to the transitional part of parameter space, the sensitivity dimensions $\mathbf{v}_1(\mathbf{z})$ are categorically different across regimes (Fig. 20 bottom). The sensitivity dimension identifies the parameter combination which causes the emergent property to diminish with the shortest perturbation. Since the two regimes have different $\mathbf{v}_1(\mathbf{z})$, this suggests they have different pathologies in their connectivity. By perturbing connectivity in each regime along the sensitivity dimension, we can get a sense of the differing nature of these pathologies.

When perturbing connectivity along the sensitivity dimension, Pro accuracy monotonically increases in both regimes (Fig. 4D, top-left). However, there is a stark difference between regimes in

369 Anti accuracy. Anti accuracy falls in either direction of \mathbf{v}_1 in regime 1, yet monotonically increases
370 along with Pro accuracy in regime 2 (Fig. 4D, top-right). These distinct pathologies of rapid task
371 switching are caused by distinct connectivity changes ($\mathbf{v}_1(\mathbf{z}^*(r = 1))$ vs $\mathbf{v}_1(\mathbf{z}^*(r = 2))$) and explain
372 the sharp change in local structure of the EPI distribution.

373 To further examine the two regimes, we can perturb \mathbf{z} in the same way along dimensions that inde-
374 pendently change the eigenvalues of the connectivity matrix (which has constant eigenvectors with
375 respect to \mathbf{z}). These eigenvalues λ_{all} , λ_{side} , λ_{task} , and λ_{diag} correspond to connectivity eigenmodes
376 with intuitive roles in processing in this task (Fig. 19A). For example, greater λ_{task} will strengthen
377 internal representations of task, while greater λ_{diag} will amplify dominance of Pro and Anti pairs in
378 opposite hemispheres (Section 5.5.6). Perturbation analyses reveal that decreasing λ_{task} has close
379 to the same effect on Anti accuracy as perturbations along the sensitivity dimension (Fig. 4D,
380 middle). This suggests that there is a carefully tuned strength of task representation in regime
381 1, which if disturbed results in random Anti trial responses. Finally, we recognize that increasing
382 λ_{diag} has opposite effects on Anti accuracy in each regime (Fig. 4D, bottom). In the next section,
383 we build on these mechanistic characterizations of each regime by examining their resilience to
384 optogenetic silencing.

385 **3.6 EPI inferred SC connectivities reproduce results from optogenetic inacti-
386 vation experiments**

387 During the delay period of this task, the circuit must prepare to execute the correct task based
388 on the cue input. Experimental results from Duan et al. found that optogenetic inactivation of
389 SC during the delay period consistently decreased performance in the Anti task, but had no effect
390 on the Pro task (Fig. 5A). This suggests that SC maintains a representation of task throughout
391 the delay period, which is important for correct execution of the Anti task. Network connectivities
392 inferred by EPI exhibited this same effect in simulation at high optogenetic strength $\gamma \in [0, 1]$ (Fig.
393 5B) (see Section 5.5.7). To emphasize, EPI inferred connectivities were only conditioned upon the
394 emergent property of rapid task switching, not on Anti task failure during delay period silencing.

395 The mean increase in Anti error is closest to the experimentally measured value of 7% at $\gamma = 0.675$
396 (Fig. 5B, black dot). At this level of optogenetic strength, only regime 1 exhibits an increase in
397 Anti error with delay period silencing (Fig. 5C, left). The connectivities in regime 2 are thus more
398 resilient in the Anti task to delay period silencing than regime 1. In regime 1, greater λ_{task} and
399 λ_{diag} decrease Anti error (Fig. 5C, right). In other words, these anticorrelations show that stronger

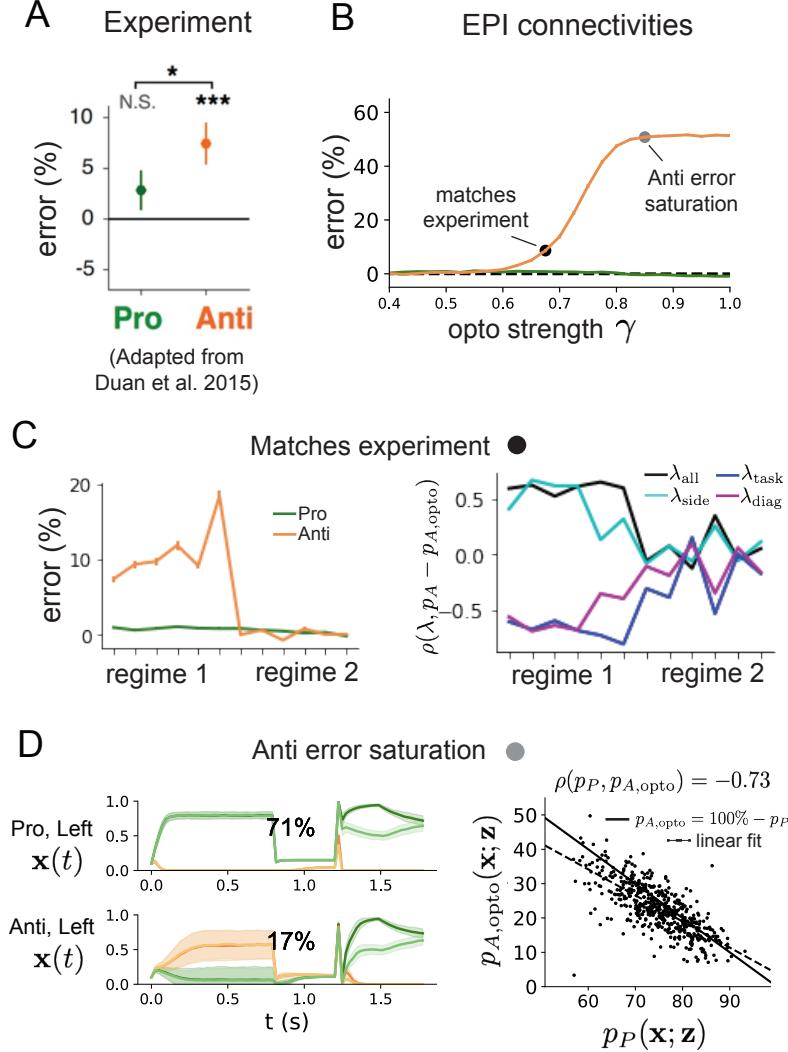


Figure 5: **A.** Experimental effect of delay period silencing on Pro and Anti task accuracy in rats. **B.** Mean and standard error (bars) across EPI distribution of Pro and Anti error induced by delay period inactivation of varying optogenetic strength. **C.** (Left) Mean and standard error of Pro and Anti error from regime 1 to regime 2 at $\gamma = 0.675$. (Right) Correlations of connectivity eigenvalues with Anti error from regime 1 to regime 2 at $\gamma = 0.675$. **D.** (Left) Responses of the SC model at the mode of the EPI distribution to delay period inactivation at $\gamma = 0.85$. (Right) Anti accuracy following delay period inactivation at $\gamma = 0.85$ versus accuracy in the Pro task across connectivities in the EPI distribution.

400 task representations and diagonal amplification make the SC model more resilient to delay period
401 silencing in the Anti task. All correlations of connectivity eigenvalue with Anti error degrade in
402 regime 2, where there is no effect of delay period silencing on Anti error (Fig. 5C, right).

403 At about $\gamma = 0.85$ (Fig. 5B, gray dot), the Anti error saturates, while Pro error remains at zero
404 Following delay period inactivation at this optogenetic strength, there are strong similarities in
405 the responses of Pro and Anti trials during the choice period (Fig. 5D, left). We interpreted
406 these similarities to suggest that delay period inactivation at this saturated level flips the internal
407 representation of task (from Anti to Pro) in the model. This would explain why the Anti error
408 saturates at 50%: the average Anti accuracy in EPI inferred connectivities is 75%, but is 25% when
409 the internal representation is flipped during delay period silencing. This hypothesis prescribes a
410 model of Anti accuracy during delay period silencing of $p_{A,\text{opto}} = 100\% - p_P$, which is fit closely
411 across both regimes of the EPI inferred connectivities (Fig. 5D, right). Similarities between Pro
412 and Anti trial responses were not present at the experiment-matching level of $\gamma = 0.675$ (Fig. 22
413 left) and neither was anti-correlation in p_P and $p_{A,\text{opto}}$ (Fig. 22 right).

414 In summary, the connectivity inferred by EPI to perform rapid task switching replicated results
415 from optogenetic silencing experiments. We found that at levels of optogenetic strength matching
416 experimental levels of Anti error, only one regime actually exhibited the effect. This suggests that
417 one regime is less resilient to optogenetic perturbation, and perhaps more biologically realistic.
418 Finally, we mechanistically characterized the pathology in Anti error that occurs in both regimes
419 when optogenetic strength is increased to high levels. This insight was made possible by using the
420 probabilistic tools afforded by EPI: we identified two regimes and the continuum of connectivities
421 between them, and we obtained many parameter samples at each step along the continuum (in 7.36
422 seconds rather than 4.2 days with brute force methods, see Section 5.5).

423 4 Discussion

424 In neuroscience, machine learning has primarily been used to reveal structure in neural datasets [37].
425 Careful inference procedures are developed for these statistical models allowing precise, quantita-
426 tive reasoning, which clarifies the way data informs beliefs about the model parameters. However,
427 these statistical models often lack resemblance to the underlying biology, making it unclear how
428 to go from the structure revealed by these methods, to the neural mechanisms giving rise to it. In
429 contrast, theoretical neuroscience has focused on careful mechanistic modeling and the production

430 of emergent properties of computation, rather than measuring structure in some noisy observed
431 dataset. In this work, we improve upon parameter inference techniques in theoretical neuroscience
432 with emergent property inference, harnessing deep learning towards parameter inference with re-
433 spect to emergent phenomena in interpretable models of neural computation (see Section 5.1.1).

434 Methodology for statistical inference in mechanistic models of neural circuits has evolved consider-
435 ably in recent years. Early work used rejection sampling techniques [43, 80, 81], but more recently
436 developed methodology employs deep learning to improve efficiency or provide flexible distribution
437 approximations. SNPE [45] and other sequential techniques for inference in mechanistic models
438 developed along with EPI (see Section 5.1.1) have been used for posterior inference with noisy
439 experimental datasets. On the other hand, EPI is a deep inference technique designed to condition
440 directly on emergent properties, such that the parameter distribution only produces the computa-
441 tion of interest. EPI is thus ideally suited for questions in theoretical neuroscience, and we show
442 that it has superior scaling properties to these other inference techniques (see Section 3.3).

443 **Acknowledgements:**

444 This work was funded by NSF Graduate Research Fellowship, DGE-1644869, McKnight Endow-
445 ment Fund, NIH NINDS 5R01NS100066, Simons Foundation 542963, NSF NeuroNex Award, DBI-
446 1707398, The Gatsby Charitable Foundation, Simons Collaboration on the Global Brain Postdoc-
447 toral Fellowship, Chinese Postdoctoral Science Foundation, and International Exchange Program
448 Fellowship. Helpful conversations were had with Francesca Mastrogiuseppe, Srdjan Ostojic, James
449 Fitzgerald, Stephen Baccus, Dhruva Raman, Liam Paninski, and Larry Abbott.

450 **Data availability statement:**

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

453 **Code availability statement:**

454 All software written for the current study is available at <https://github.com/cunningham-lab/epi>.

455 **References**

- 456 [1] Nancy Kopell and G Bard Ermentrout. Coupled oscillators and the design of central pattern
457 generators. *Mathematical biosciences*, 90(1-2):87–109, 1988.

- 458 [2] Eve Marder. From biophysics to models of network function. *Annual review of neuroscience*,
459 21(1):25–45, 1998.
- 460 [3] Larry F Abbott. Theoretical neuroscience rising. *Neuron*, 60(3):489–495, 2008.
- 461 [4] Xiao-Jing Wang. Neurophysiological and computational principles of cortical rhythms in
462 cognition. *Physiological reviews*, 90(3):1195–1268, 2010.
- 463 [5] Ryan N Gutenkunst, Joshua J Waterfall, Fergal P Casey, Kevin S Brown, Christopher R
464 Myers, and James P Sethna. Universally sloppy parameter sensitivities in systems biology
465 models. *PLoS Comput Biol*, 3(10):e189, 2007.
- 466 [6] Timothy O’Leary, Alex H Williams, Alessio Franci, and Eve Marder. Cell types, network
467 homeostasis, and pathological compensation from a biologically plausible ion channel expres-
468 sion model. *Neuron*, 82(4):809–821, 2014.
- 469 [7] John J Hopfield. Neural networks and physical systems with emergent collective computa-
470 tional abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558, 1982.
- 471 [8] Haim Sompolinsky, Andrea Crisanti, and Hans-Jurgen Sommers. Chaos in random neural
472 networks. *Physical review letters*, 61(3):259, 1988.
- 473 [9] Misha V Tsodyks, William E Skaggs, Terrence J Sejnowski, and Bruce L McNaughton. Para-
474 doxical effects of external modulation of inhibitory interneurons. *Journal of neuroscience*,
475 17(11):4382–4388, 1997.
- 476 [10] Kong-Fatt Wong and Xiao-Jing Wang. A recurrent network mechanism of time integration
477 in perceptual decisions. *Journal of Neuroscience*, 26(4):1314–1328, 2006.
- 478 [11] WR Foster, LH Ungar, and JS Schwaber. Significance of conductances in hodgkin-huxley
479 models. *Journal of neurophysiology*, 70(6):2502–2518, 1993.
- 480 [12] Astrid A Prinz, Dirk Bucher, and Eve Marder. Similar network activity from disparate circuit
481 parameters. *Nature neuroscience*, 7(12):1345–1352, 2004.
- 482 [13] Pablo Achard and Erik De Schutter. Complex parameter landscape for a complex neuron
483 model. *PLoS computational biology*, 2(7):e94, 2006.
- 484 [14] Leandro M Alonso and Eve Marder. Visualization of currents in neural models with similar
485 behavior and different conductance densities. *Elife*, 8:e42722, 2019.

- 486 [15] Robert E Kass and Valérie Ventura. A spike-train probability model. *Neural computation*,
487 13(8):1713–1720, 2001.
- 488 [16] Emery N Brown, Loren M Frank, Dengda Tang, Michael C Quirk, and Matthew A Wilson.
489 A statistical paradigm for neural spike train decoding applied to position prediction from
490 ensemble firing patterns of rat hippocampal place cells. *Journal of Neuroscience*, 18(18):7411–
491 7425, 1998.
- 492 [17] Liam Paninski. Maximum likelihood estimation of cascade point-process neural encoding
493 models. *Network: Computation in Neural Systems*, 15(4):243–262, 2004.
- 494 [18] Wilson Truccolo, Uri T Eden, Matthew R Fellows, John P Donoghue, and Emery N Brown.
495 A point process framework for relating neural spiking activity to spiking history, neural
496 ensemble, and extrinsic covariate effects. *Journal of neurophysiology*, 93(2):1074–1089, 2005.
- 497 [19] Elad Schneidman, Michael J Berry, Ronen Segev, and William Bialek. Weak pairwise correlations
498 imply strongly correlated network states in a neural population. *Nature*, 440(7087):1007–
499 1012, 2006.
- 500 [20] Shaul Druckmann, Yoav Banitt, Albert A Gidon, Felix Schürmann, Henry Markram, and Idan
501 Segev. A novel multiple objective optimization framework for constraining conductance-based
502 neuron models by experimental data. *Frontiers in neuroscience*, 1:1, 2007.
- 503 [21] Richard Turner and Maneesh Sahani. A maximum-likelihood interpretation for slow feature
504 analysis. *Neural computation*, 19(4):1022–1038, 2007.
- 505 [22] M Yu Byron, John P Cunningham, Gopal Santhanam, Stephen I Ryu, Krishna V Shenoy, and
506 Maneesh Sahani. Gaussian-process factor analysis for low-dimensional single-trial analysis of
507 neural population activity. In *Advances in neural information processing systems*, pages
508 1881–1888, 2009.
- 509 [23] Jakob H Macke, Lars Buesing, John P Cunningham, Byron M Yu, Krishna V Shenoy, and
510 Maneesh Sahani. Empirical models of spiking in neural populations. *Advances in neural
511 information processing systems*, 24:1350–1358, 2011.
- 512 [24] Il Memming Park and Jonathan W Pillow. Bayesian spike-triggered covariance analysis. In
513 *Advances in neural information processing systems*, pages 1692–1700, 2011.

- 514 [25] Einat Granot-Atedgi, Gašper Tkačik, Ronen Segev, and Elad Schneidman. Stimulus-
515 dependent maximum entropy models of neural population codes. *PLoS Comput Biol*,
516 9(3):e1002922, 2013.
- 517 [26] Kenneth W Latimer, Jacob L Yates, Miriam LR Meister, Alexander C Huk, and Jonathan W
518 Pillow. Single-trial spike trains in parietal cortex reveal discrete steps during decision-making.
519 *Science*, 349(6244):184–187, 2015.
- 520 [27] Kaushik J Lakshminarasimhan, Marina Petsalis, Hyeshin Park, Gregory C DeAngelis, Xaq
521 Pitkow, and Dora E Angelaki. A dynamic bayesian observer model reveals origins of bias in
522 visual path integration. *Neuron*, 99(1):194–206, 2018.
- 523 [28] Lea Duncker, Gergo Bohner, Julien Boussard, and Maneesh Sahani. Learning interpretable
524 continuous-time models of latent stochastic dynamical systems. *Proceedings of the 36th In-*
525 *ternational Conference on Machine Learning*, 2019.
- 526 [29] Josef Ladenbauer, Sam McKenzie, Daniel Fine English, Olivier Hagens, and Srdjan Ostojic.
527 Inferring and validating mechanistic models of neural microcircuits based on spike-train data.
528 *Nature Communications*, 10(4933), 2019.
- 529 [30] Yuanjun Gao, Evan W Archer, Liam Paninski, and John P Cunningham. Linear dynamical
530 neural population models through nonlinear embeddings. In *Advances in neural information*
531 *processing systems*, pages 163–171, 2016.
- 532 [31] Yuan Zhao and Il Memming Park. Recursive variational bayesian dual estimation for non-
533 linear dynamics and non-gaussian observations. *stat*, 1050:27, 2017.
- 534 [32] Gabriel Barello, Adam Charles, and Jonathan Pillow. Sparse-coding variational auto-
535 encoders. *bioRxiv*, page 399246, 2018.
- 536 [33] Chethan Pandarinath, Daniel J O’Shea, Jasmine Collins, Rafal Jozefowicz, Sergey D Stavisky,
537 Jonathan C Kao, Eric M Trautmann, Matthew T Kaufman, Stephen I Ryu, Leigh R
538 Hochberg, et al. Inferring single-trial neural population dynamics using sequential auto-
539 encoders. *Nature methods*, page 1, 2018.
- 540 [34] Alexander B Wiltschko, Matthew J Johnson, Giuliano Iurilli, Ralph E Peterson, Jesse M
541 Katon, Stan L Pashkovski, Victoria E Abraira, Ryan P Adams, and Sandeep Robert Datta.
542 Mapping sub-second structure in mouse behavior. *Neuron*, 88(6):1121–1135, 2015.

- 543 [35] Matthew J Johnson, David K Duvenaud, Alex Wiltschko, Ryan P Adams, and Sandeep R
544 Datta. Composing graphical models with neural networks for structured representations and
545 fast inference. In *Advances in neural information processing systems*, pages 2946–2954, 2016.
- 546 [36] Eleanor Batty, Matthew Whiteway, Shreya Saxena, Dan Biderman, Taiga Abe, Simon Musall,
547 Winthrop Gillis, Jeffrey Markowitz, Anne Churchland, John Cunningham, et al. Behavenet:
548 nonlinear embedding and bayesian neural decoding of behavioral videos. *Advances in Neural
549 Information Processing Systems*, 2019.
- 550 [37] Liam Paninski and John P Cunningham. Neural data science: accelerating the experiment-
551 analysis-theory cycle in large-scale neuroscience. *Current opinion in neurobiology*, 50:232–241,
552 2018.
- 553 [38] Christopher M Niell and Michael P Stryker. Modulation of visual responses by behavioral
554 state in mouse visual cortex. *Neuron*, 65(4):472–479, 2010.
- 555 [39] Aman B Saleem, Ash Ayaz, Kathryn J Jeffery, Kenneth D Harris, and Matteo Carandini.
556 Integration of visual motion and locomotion in mouse visual cortex. *Nature neuroscience*,
557 16(12):1864–1869, 2013.
- 558 [40] Simon Musall, Matthew T Kaufman, Ashley L Juavinett, Steven Gluf, and Anne K Church-
559 land. Single-trial neural dynamics are dominated by richly varied movements. *Nature neuro-
560 science*, 22(10):1677–1686, 2019.
- 561 [41] Peter Dayan, Laurence F Abbott, et al. Theoretical neuroscience: computational and mathe-
562 matical modeling of neural systems. *Journal of Cognitive Neuroscience*, 15(1):154–155, 2003.
- 563 [42] Eugene M Izhikevich. *Dynamical systems in neuroscience*. MIT press, 2007.
- 564 [43] Scott A Sisson, Yanan Fan, and Mark M Tanaka. Sequential monte carlo without likelihoods.
565 *Proceedings of the National Academy of Sciences*, 104(6):1760–1765, 2007.
- 566 [44] Juliane Liepe, Paul Kirk, Sarah Filippi, Tina Toni, Chris P Barnes, and Michael PH Stumpf.
567 A framework for parameter estimation and model selection from experimental data in systems
568 biology using approximate bayesian computation. *Nature protocols*, 9(2):439–456, 2014.
- 569 [45] Pedro J Gonçalves, Jan-Matthis Lueckmann, Michael Deistler, Marcel Nonnenmacher, Kaan
570 Öcal, Giacomo Bassetto, Chaitanya Chintaluri, William F Podlaski, Sara A Haddad, Tim P

- 571 Vogels, et al. Training deep neural density estimators to identify mechanistic models of neural
572 dynamics. *bioRxiv*, page 838383, 2019.
- 573 [46] George Papamakarios, David Sterratt, and Iain Murray. Sequential neural likelihood: Fast
574 likelihood-free inference with autoregressive flows. In *The 22nd International Conference on*
575 *Artificial Intelligence and Statistics*, pages 837–848. PMLR, 2019.
- 576 [47] Joeri Hermans, Volodimir Begy, and Gilles Louppe. Likelihood-free mcmc with amortized
577 approximate ratio estimators. In *International Conference on Machine Learning*, pages 4239–
578 4248. PMLR, 2020.
- 579 [48] Gabriel Loaiza-Ganem, Yuanjun Gao, and John P Cunningham. Maximum entropy flow
580 networks. *International Conference on Learning Representations*, 2017.
- 581 [49] Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows.
582 *International Conference on Machine Learning*, 2015.
- 583 [50] Mark S Goldman, Jorge Golowasch, Eve Marder, and LF Abbott. Global structure, ro-
584 bustness, and modulation of neuronal models. *Journal of Neuroscience*, 21(14):5229–5238,
585 2001.
- 586 [51] Gabrielle J Gutierrez, Timothy O’Leary, and Eve Marder. Multiple mechanisms switch an
587 electrically coupled, synaptically inhibited neuron between competing rhythmic oscillators.
588 *Neuron*, 77(5):845–858, 2013.
- 589 [52] Brendan K Murphy and Kenneth D Miller. Balanced amplification: a new mechanism of
590 selective amplification of neural activity patterns. *Neuron*, 61(4):635–648, 2009.
- 591 [53] Guillaume Hennequin, Tim P Vogels, and Wulfram Gerstner. Optimal control of transient dy-
592 namics in balanced networks supports generation of complex movements. *Neuron*, 82(6):1394–
593 1406, 2014.
- 594 [54] Giulio Bondanelli, Thomas Deneux, Brice Bathellier, and Srdjan Ostojic. Population coding
595 and network dynamics during off responses in auditory cortex. *BioRxiv*, page 810655, 2019.
- 596 [55] Ashok Litwin-Kumar, Robert Rosenbaum, and Brent Doiron. Inhibitory stabilization and
597 visual coding in cortical circuits with multiple interneuron subtypes. *Journal of neurophysi-
598 ology*, 115(3):1399–1409, 2016.

- 599 [56] Agostina Palmigiano, Francesco Fumarola, Daniel P Mossing, Nataliya Kraynyukova, Hillel
600 Adesnik, and Kenneth Miller. Structure and variability of optogenetic responses identify the
601 operating regime of cortex. *bioRxiv*, 2020.
- 602 [57] Chunyu A Duan, Marino Pagan, Alex T Piet, Charles D Kopec, Athena Akrami, Alexander J
603 Riordan, Jeffrey C Erlich, and Carlos D Brody. Collicular circuits for flexible sensorimotor
604 routing. *bioRxiv*, page 245613, 2018.
- 605 [58] Eve Marder and Vatsala Thirumalai. Cellular, synaptic and network effects of neuromodula-
606 tion. *Neural Networks*, 15(4-6):479–493, 2002.
- 607 [59] Catherine Morris and Harold Lecar. Voltage oscillations in the barnacle giant muscle fiber.
608 *Biophysical journal*, 35(1):193–213, 1981.
- 609 [60] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp.
610 *Proceedings of the 5th International Conference on Learning Representations*, 2017.
- 611 [61] George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for
612 density estimation. In *Advances in Neural Information Processing Systems*, pages 2338–2347,
613 2017.
- 614 [62] Mark S Goldman. Memory without feedback in a neural network. *Neuron*, 61(4):621–634,
615 2009.
- 616 [63] Giulio Bondanelli and Srdjan Ostojic. Coding with transient trajectories in recurrent neural
617 networks. *PLoS computational biology*, 16(2):e1007655, 2020.
- 618 [64] David Sussillo. Neural circuits as computational dynamical systems. *Current opinion in
619 neurobiology*, 25:156–163, 2014.
- 620 [65] Omri Barak. Recurrent neural networks as versatile tools of neuroscience research. *Current
621 opinion in neurobiology*, 46:1–6, 2017.
- 622 [66] Abigail A Russo, Sean R Bittner, Sean M Perkins, Jeffrey S Seely, Brian M London, Antonio H
623 Lara, Andrew Miri, Najja J Marshall, Adam Kohn, Thomas M Jessell, et al. Motor cortex
624 embeds muscle-like commands in an untangled population response. *Neuron*, 97(4):953–966,
625 2018.
- 626 [67] Scott A Sisson, Yanan Fan, and Mark Beaumont. *Handbook of approximate Bayesian com-
627 putation*. CRC Press, 2018.

- 628 [68] Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference.
629 *Proceedings of the National Academy of Sciences*, 2020.
- 630 [69] Eve Marder and Allen I Selverston. *Dynamic biological networks: the stomatogastric nervous*
631 *system*. MIT press, 1992.
- 632 [70] Hirofumi Ozeki, Ian M Finn, Evan S Schaffer, Kenneth D Miller, and David Ferster. Inhibitory
633 stabilization of the cortical network underlies visual surround suppression. *Neuron*, 62(4):578–
634 592, 2009.
- 635 [71] Daniel B Rubin, Stephen D Van Hooser, and Kenneth D Miller. The stabilized supralinear
636 network: a unifying circuit motif underlying multi-input integration in sensory cortex.
637 *Neuron*, 85(2):402–417, 2015.
- 638 [72] Henry Markram, Maria Toledo-Rodriguez, Yun Wang, Anirudh Gupta, Gilad Silberberg, and
639 Caizhi Wu. Interneurons of the neocortical inhibitory system. *Nature reviews neuroscience*,
640 5(10):793, 2004.
- 641 [73] Bernardo Rudy, Gordon Fishell, SooHyun Lee, and Jens Hjerling-Leffler. Three groups of
642 interneurons account for nearly 100% of neocortical gabaergic neurons. *Developmental neu-*
643 *robiology*, 71(1):45–61, 2011.
- 644 [74] Robin Tremblay, Soohyun Lee, and Bernardo Rudy. GABAergic Interneurons in the Neocor-
645 tex: From Cellular Properties to Circuits. *Neuron*, 91(2):260–292, 2016.
- 646 [75] Carsten K Pfeffer, Mingshan Xue, Miao He, Z Josh Huang, and Massimo Scanziani. Inhi-
647 bition of inhibition in visual cortex: the logic of connections between molecularly distinct
648 interneurons. *Nature Neuroscience*, 16(8):1068, 2013.
- 649 [76] João D Semedo, Amin Zandvakili, Christian K Machens, M Yu Byron, and Adam Kohn.
650 Cortical areas interact through a communication subspace. *Neuron*, 102(1):249–259, 2019.
- 651 [77] Daniel J Felleman and David C Van Essen. Distributed hierarchical processing in the primate
652 cerebral cortex. *Cerebral cortex (New York, NY: 1991)*, 1(1):1–47, 1991.
- 653 [78] Guillaume Hennequin, Yashar Ahmadian, Daniel B Rubin, Máté Lengyel, and Kenneth D
654 Miller. The dynamical regime of sensory cortex: stable dynamics around a single stimulus-
655 tuned attractor account for patterns of noise variability. *Neuron*, 98(4):846–860, 2018.

- 656 [79] Chunyu A Duan, Jeffrey C Erlich, and Carlos D Brody. Requirement of prefrontal and
657 midbrain regions for rapid executive control of behavior in the rat. *Neuron*, 86(6):1491–1503,
658 2015.
- 659 [80] Mark A Beaumont, Wenyang Zhang, and David J Balding. Approximate bayesian computa-
660 tion in population genetics. *Genetics*, 162(4):2025–2035, 2002.
- 661 [81] Paul Marjoram, John Molitor, Vincent Plagnol, and Simon Tavaré. Markov chain monte carlo
662 without likelihoods. *Proceedings of the National Academy of Sciences*, 100(26):15324–15328,
663 2003.
- 664 [82] Lawrence Saul and Michael Jordan. A mean field learning algorithm for unsupervised neural
665 networks. In *Learning in graphical models*, pages 541–554. Springer, 1998.
- 666 [83] Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and
667 Edward Teller. Equation of state calculations by fast computing machines. *The journal of
668 chemical physics*, 21(6):1087–1092, 1953.
- 669 [84] W Keith Hastings. Monte carlo sampling methods using markov chains and their applications.
670 1970.
- 671 [85] Mark Girolami and Ben Calderhead. Riemann manifold langevin and hamiltonian monte
672 carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*,
673 73(2):123–214, 2011.
- 674 [86] Andrew Golightly and Darren J Wilkinson. Bayesian parameter inference for stochastic bio-
675 chemical network models using particle markov chain monte carlo. *Interface focus*, 1(6):807–
676 820, 2011.
- 677 [87] Dustin Tran, Rajesh Ranganath, and David Blei. Hierarchical implicit models and likelihood-
678 free variational inference. In *Advances in Neural Information Processing Systems*, pages
679 5523–5533, 2017.
- 680 [88] Sean R Bittner, Agostina Palmigiano, Kenneth D Miller, and John P Cunningham. Degener-
681 ate solution networks for theoretical neuroscience. *Computational and Systems Neuroscience
682 Meeting (COSYNE), Lisbon, Portugal*, 2019.

- 683 [89] Sean R Bittner, Alex T Piet, Chunyu A Duan, Agostina Palmigiano, Kenneth D Miller,
684 Carlos D Brody, and John P Cunningham. Examining models in theoretical neuroscience
685 with degenerate solution networks. *Bernstein Conference 2019, Berlin, Germany*, 2019.
- 686 [90] Marcel Nonnenmacher, Pedro J Goncalves, Giacomo Bassetto, Jan-Matthis Lueckmann, and
687 Jakob H Macke. Robust statistical inference for simulation-based models in neuroscience. In
688 *Bernstein Conference 2018, Berlin, Germany*, 2018.
- 689 [91] Deistler Michael, , Pedro J Goncalves, Kaan Oecal, and Jakob H Macke. Statistical infer-
690 ence for analyzing sloppiness in neuroscience models. In *Bernstein Conference 2019, Berlin,*
691 *Germany*, 2019.
- 692 [92] Jan-Matthis Lueckmann, Pedro J Goncalves, Giacomo Bassetto, Kaan Öcal, Marcel Nonnen-
693 macher, and Jakob H Macke. Flexible statistical inference for mechanistic models of neural
694 dynamics. In *Advances in Neural Information Processing Systems*, pages 1289–1299, 2017.
- 695 [93] Martin J Wainwright, Michael I Jordan, et al. Graphical models, exponential families, and
696 variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.
- 697 [94] Sean R Bittner and John P Cunningham. Approximating exponential family models (not
698 single distributions) with a two-network architecture. *arXiv preprint arXiv:1903.07515*, 2019.
- 699 [95] Johan Karlsson, Milena Anguelova, and Mats Jirstrand. An efficient method for structural
700 identifiability analysis of large dynamic systems. *IFAC Proceedings Volumes*, 45(16):941–946,
701 2012.
- 702 [96] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary
703 differential equations. In *Advances in neural information processing systems*, pages 6571–6583,
704 2018.
- 705 [97] Xuechen Li, Ting-Kam Leonard Wong, Ricky TQ Chen, and David Duvenaud. Scalable
706 gradients for stochastic differential equations. *arXiv preprint arXiv:2001.01328*, 2020.
- 707 [98] Andreas Raue, Clemens Kreutz, Thomas Maiwald, Julie Bachmann, Marcel Schilling, Ursula
708 Klingmüller, and Jens Timmer. Structural and practical identifiability analysis of partially
709 observed dynamical models by exploiting the profile likelihood. *Bioinformatics*, 25(15):1923–
710 1929, 2009.

- 711 [99] Dhruva V Raman, James Anderson, and Antonis Papachristodoulou. Delineating parameter
712 unidentifiabilities in complex models. *Physical Review E*, 95(3):032314, 2017.
- 713 [100] Maria Pia Saccomani, Stefania Audoly, and Leontina D’Angiò. Parameter identifiability of
714 nonlinear systems: the role of initial conditions. *Automatica*, 39(4):619–632, 2003.
- 715 [101] George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Bal-
716 aji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *arXiv*
717 preprint arXiv:1912.02762, 2019.
- 718 [102] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolu-
719 tions. In *Advances in neural information processing systems*, pages 10215–10224, 2018.
- 720 [103] Durk P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling.
721 Improved variational inference with inverse autoregressive flow. *Advances in neural informa-*
722 *tion processing systems*, 29:4743–4751, 2016.
- 723 [104] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *Internation-
724 al Conference on Learning Representations*, 2015.
- 725 [105] Emmanuel Klinger, Dennis Rickert, and Jan Hasenauer. pyabc: distributed, likelihood-free
726 inference. *Bioinformatics*, 34(20):3591–3593, 2018.
- 727 [106] David S Greenberg, Marcel Nonnenmacher, and Jakob H Macke. Automatic posterior trans-
728 formation for likelihood-free inference. *International Conference on Machine Learning*, 2019.

729 **5 Methods**

730 **5.1 Emergent property inference (EPI)**

731 Determining the combinations of model parameters that can produce observed data or a desired
732 output is a key part of scientific practice. Solving inverse problems is especially important in
733 neuroscience, since we require complex models to describe the complex phenomena of neural com-
734 putations. While much machine learning research has focused on how to find latent structure
735 in large-scale neural datasets, less has focused on inverting theoretical circuit models conditioned
736 upon the emergent phenomena they produce. Here, we introduce a novel method for statistical
737 inference, which finds distributions of parameter solutions that only produce the desired emer-
738 gent property. This method seamlessly handles neural circuit models with stochastic nonlinear
739 dynamical generative processes, which are predominant in theoretical neuroscience.

740 Consider model parameterization \mathbf{z} , which is a collection of scientifically interesting variables that
741 govern the complex simulation of data \mathbf{x} . For example (see Section 3.1), \mathbf{z} may be the electrical
742 conductance parameters of an STG subcircuit, and \mathbf{x} the evolving membrane potentials of the five
743 neurons. In terms of statistical modeling, this circuit model has an intractable likelihood $p(\mathbf{x} | \mathbf{z})$,
744 which is predicated by the stochastic differential equations that define the model. Even so, we do
745 not scientifically reason about how \mathbf{z} governs all of \mathbf{x} , but rather specific phenomena that are a
746 function of the data $f(\mathbf{x}; \mathbf{z})$. In the STG example, $f(\mathbf{x}; \mathbf{z})$ measures hub neuron frequency from the
747 evolution of \mathbf{x} governed by \mathbf{z} . With EPI, we learn distributions of \mathbf{z} that results in an average and
748 variance of $f(\mathbf{x}; \mathbf{z})$, denoted $\boldsymbol{\mu}$ and σ^2 . We refer to the collection of these statistical moments as an
749 emergent property. Such emergent properties \mathcal{X} are defined through choice of $f(\mathbf{x}; \mathbf{z})$ (which may
750 be one or multiple statistics), $\boldsymbol{\mu}$, and σ^2

$$\mathcal{X} : \mathbb{E}_{\mathbf{z}, \mathbf{x}} [f(\mathbf{x}; \mathbf{z})] = \boldsymbol{\mu}, \text{Var}_{\mathbf{z}, \mathbf{x}} [f(\mathbf{x}; \mathbf{z})] = \sigma^2. \quad (11)$$

751 Precisely, the emergent property statistics $f(\mathbf{x}; \mathbf{z})$ must have means $\boldsymbol{\mu}$ and variances σ^2 over the
752 EPI distribution of parameters and stochasticity of the data given the parameters. By defining
753 these means and variances over both levels of stochasticity – the inferred distribution and that of
754 the model – there is a fine degree of control over predictions made by the inferred parameters.

755 In EPI, deep probability distributions are optimized to learn the inferred distribution. In deep
756 probability distributions, a simple random variable $\mathbf{z}_0 \sim q_0(\mathbf{z}_0)$ is mapped deterministically via a
757 sequence of deep neural network layers (g_1, \dots, g_l) parameterized by weights and biases $\boldsymbol{\theta}$ to the

758 support of the distribution of interest:

$$\mathbf{z} = g_{\theta}(\mathbf{z}_0) = g_l(\dots g_1(\mathbf{z}_0)) \sim q_{\theta}(\mathbf{z}). \quad (12)$$

759 Such deep probability distributions embed the inferred distribution in a deep network. Once opti-
760 mized, this deep network representation has remarkably useful properties: fast sampling, probability
761 evaluations, and also first- and second-order probability gradient evaluations.

762 By choosing a neural circuit model, often represented as a system of differential equations, we
763 implicitly define a model likelihood $p(\mathbf{x} | \mathbf{z})$, which may be unknown or intractable for our purposes.
764 Given this model choice and that of an emergent property \mathcal{X} , $q_{\theta}(\mathbf{z})$ is optimized via the neural
765 network parameters θ to find a maximally entropic distribution q_{θ}^* within the deep variational
766 family \mathcal{Q} producing the emergent property \mathcal{X} :

$$\begin{aligned} q_{\theta}(\mathbf{z} | \mathcal{X}) &= q_{\theta}^*(\mathbf{z}) = \operatorname{argmax}_{q_{\theta} \in \mathcal{Q}} H(q_{\theta}(\mathbf{z})) \\ \text{s.t. } \mathcal{X} &: \mathbb{E}_{\mathbf{z}, \mathbf{x}} [f(\mathbf{x}; \mathbf{z})] = \boldsymbol{\mu}, \operatorname{Var}_{\mathbf{z}, \mathbf{x}} [f(\mathbf{x}; \mathbf{z})] = \boldsymbol{\sigma}^2. \end{aligned} \quad (13)$$

767 Entropy is chosen as the normative selection principle to match that of variational Bayesian methods
768 (see Section 5.1.5). However, a key difference is that variational Bayesian methods do not constrain
769 the predictions of their inferred parameter distribution. This optimization is executed using the
770 algorithm of Maximum Entropy Flow Networks (MEFNs) [48].

771 In the remainder of Section 5.1, we will explain the finer details and motivation of the EPI method.
772 First, we explain related approaches and what EPI introduces to this domain (Section 5.1.1). Sec-
773 ond, we describe the special class of deep probability distributions used in EPI called normalizing
774 flows (Section 5.1.2). Next, we explain the constrained optimization technique used to solve Equa-
775 tion 13 (Section 5.1.3). Then, we demonstrate the details of this optimization in a toy example
776 (Section 5.1.4). Finally, we establish the known relationship between maximum entropy distribu-
777 tions and exponential families (Section 5.1.5), which is used to explain how EPI can be viewed as
778 a form of variational inference (Section 5.1.6).

779 5.1.1 Related approaches

780 When Bayesian inference problems lack conjugacy, scientists use approximate inference methods like
781 variational inference (VI) [82] and Markov chain Monte Carlo (MCMC) [83, 84]. After optimization,
782 variational methods return a parameterized posterior distribution, which we can analyze. Also, the
783 variational approximating distribution class is often chosen such that it permits fast sampling. In

784 contrast MCMC methods only produce samples from the approximated posterior distribution. No
785 parameterized distribution is estimated, and additional samples are always generated with the same
786 sampling complexity. Inference in models defined by systems of differential has been demonstrated
787 with MCMC [85], although this approach requires tractable likelihoods. Advancements have lever-
788 aged structure in stochastic differential equation models to improve likelihood approximations, thus
789 expanding the domain of applicable models [86].

790 Simulation-based inference [68] is model parameter inference in the absence of a tractable likelihood
791 function. The most prevalent approach to simulation-based inference is approximate Bayesian
792 computation [80], in which satisfactory parameter samples are kept from random prior sampling
793 according to a rejection heuristic. The obtained set of parameters do not have a probabilities,
794 and further insight about the model must be gained from examination of the parameter set and
795 their generated activity. Methodological advances to ABC methods have come through the use
796 of Markov chain Monte Carlo (MCMC-ABC) [81] and sequential Monte Carlo (SMC-ABC) [43]
797 sampling techniques. SMC-ABC is considered state-of-the-art ABC, yet this approach still struggles
798 to scale in dimensionality (cf. Fig. 2). Furthermore, once a parameter set has been obtained by
799 SMC-ABC from a finite set of particles, the SMC-ABC algorithm must be run again from scratch
800 with a new population of initialized particles to obtain additional samples.

801 For scientific model analysis, we seek a parameter distribution exhibiting the properties of a well-
802 chosen variational approximation: a parametric form conferring analytic calculations, and trivial
803 sampling time. For this reason, ABC and MCMC techniques are unattractive, since they only
804 produce a set of parameter samples and have unchanging sampling rate. EPI infers parameters
805 in mechanistic models using the MEFN [48] algorithm using a deep variational approximation.
806 The deep neural network of EPI defines the parametric form of the distribution approximation.
807 Furthermore, the EPI distribution is constrained to produce an emergent property. In other words,
808 the summary statistics of the posterior predictive distribution are fixed to have certain first and
809 second moments. EPI optimization is enabled using stochastic gradient techniques in the spirit
810 of likelihood-free variational inference [87]. The analytic relationship between EPI and variational
811 inference is explained in Secton 5.1.6.

812 We note that, during our preparation and early presentation of this work [88, 89], another work
813 has arisen with broadly similar goals: bringing statistical inference to mechanistic models of neural
814 circuits ([45, 90, 91]). We are encouraged by this general problem being recognized by others in the
815 community, and we emphasize that these works offer complementary neuroscientific contributions

816 (different theoretical models of focus) and use different technical methodologies (ours is built on
817 our prior work [48], theirs similarly [92]).

818 The method EPI differs from SNPE in some key ways. SNPE belongs to a “sequential” class
819 of recently developed simulation-based inference methods in which two neural networks are used
820 for posterior inference. This first neural network is a deep probability distribution (normalizing
821 flow) used to estimate the posterior $p(\mathbf{z} | \mathbf{x})$ (SNPE) or the likelihood $p(\mathbf{x} | \mathbf{z})$ (sequential neural
822 likelihood (SNL [46])). A recent advance uses an unconstrained neural network to estimate the
823 likelihood ratio (sequential neural ratio estimation (SNRE [47])). In SNL and SNRE, MCMC
824 sampling techniques are used to obtain samples from the approximated posterior. This contrasts
825 with EPI and SNPE, which use deep probability distributions to model parameters, which facilitates
826 immediate measurements of sample probability, gradient, or Hessian for system analysis. The
827 second neural network in this sequential class of methods is the amortizer. This unconstrained
828 deep network maps data \mathbf{x} (or statistics $f(\mathbf{x}; \mathbf{z})$) or model parameters \mathbf{z} to the weights and biases of
829 the first neural network. These methods are optimized on a conditional density (or ratio) estimation
830 objective. The data used to optimize this objective are generated via an adaptive procedure, in
831 which training data pairs $(\mathbf{x}_i, \mathbf{z}_i)$ become sequentially closer to the true data and posterior.

832 The approximating fidelity of the deep probability distribution in sequential approaches is opti-
833 mized to generalize across the training distribution of the conditioning variable. This generalization
834 property of the sequential methods can reduce the accuracy at the singular posterior of interest.
835 Whereas in EPI, the entire expressivity of the deep probability distribution is dedicated to learning
836 a single distribution as well as possible. Amortization is not possible in EPI, since EPI learns
837 an exponential family distribution parameterized by its mean (see Section 5.1.5). Since EPI dis-
838 tributions are defined by the mean $\boldsymbol{\mu}$ of their statistics, there is the well-known inverse mapping
839 problem of exponential families [93] that prohibits an amortization based approach. However, we
840 have shown that the same two-network architecture of the sequential simulation-based inference
841 methods can be used for amortized inference in intractable exponential family posteriors using their
842 natural parameterization [94].

843 Finally, one important differentiating factor between EPI and sequential simulation-based infer-
844 ence methods is that EPI leverages gradients $\nabla_{\mathbf{z}} f(\mathbf{x}; \mathbf{z})$ during optimization. These gradients can
845 improve convergence time and scalability, as we have shown on an example conditioning low-rank
846 RNN connectivity on the property of stable amplification (see Section 3.3). With EPI, we prove
847 out the suggestion that a deep inference technique can improve efficiency by leveraging these model

848 gradients when they are tractable. Sequential simulation-based inference techniques may be better
 849 suited for scientific problems where $\nabla_{\mathbf{z}} f(\mathbf{x}; \mathbf{z})$ is intractable or unavailable: when there is a non-
 850 differentiable model or it requires lengthy simulations. However, the sequential simulation-based
 851 inference techniques cannot constrain the predictions of the inferred distribution in the manner of
 852 EPI.

853 Structural identifiability analysis involves the measurement of sensitivity and unidentifiabilities in
 854 natural models. Around a point, one can measure the Jacobian. One approach that scales well is
 855 EAR [95]. A popular efficient approach for systems of ODEs has been neural ODE adjoint [96] and
 856 its stochastic adaptation [97]. Casting identifiability as a statistical estimation problem, the profile
 857 likelihood can assess via iterated optimization while holding parameters fixed [98]. An exciting
 858 recent method is capable of recovering the functional form of such unidentifiabilities away from a
 859 point by following degenerate dimensions of the fisher information matrix [99]. Global structural
 860 non-identifiabilities can be found for models with polynomial or rational dynamics equations using
 861 DAISY [100]. With EPI, we have all the benefits given by a statistical inference method plus the
 862 ability to query the first- or second-order gradient of the probability of the inferred distribution at
 863 any chosen parameter value. The second-order gradient of the log probability (the Hessian), which
 864 is directly afforded by EPI distributions, produces salient information about parametric sensitivity
 865 of the emergent property. For example, the eigenvector with most negative eigenvalue of the Hessian
 866 shows parametric combinations away from a parameter choice that decrease the in EPI distribution
 867 probability the fastest. We refer to this eigenvector as the sensitivity dimension, and it is used to
 868 generate scientific insight about a model of superior colliculus connectivity (see Section 3.5).

869 5.1.2 Deep probability distributions and normalizing flows

870 Deep probability distributions are comprised of multiple layers of fully connected neural networks
 871 (Equation 12). When each neural network layer is restricted to be a bijective function, the sample
 872 density can be calculated using the change of variables formula at each layer of the network. For
 873 $\mathbf{z}_i = g_i(\mathbf{z}_{i-1})$,

$$p(\mathbf{z}_i) = p(g_i^{-1}(\mathbf{z}_i)) \left| \det \frac{\partial g_i^{-1}(\mathbf{z}_i)}{\partial \mathbf{z}_i} \right| = p(\mathbf{z}_{i-1}) \left| \det \frac{\partial g_i(\mathbf{z}_{i-1})}{\partial \mathbf{z}_{i-1}} \right|^{-1}. \quad (14)$$

874 However, this computation has cubic complexity in dimensionality for fully connected layers. By
 875 restricting our layers to normalizing flows [49, 101] – bijective functions with fast log determinant
 876 Jacobian computations, which confer a fast calculation of the sample log probability. Fast log

877 probability calculation confers efficient optimization of the maximum entropy objective (see Section
878 5.1.3).

879 We use the Real NVP [60] normalizing flow class, because its coupling architecture confers both
880 fast sampling (forward) and fast log probability evaluation (backward). Fast probability evaluation
881 facilitates fast gradient and Hessian evaluation of log probability throughout parameter space.
882 Glow permutations were used in between coupling stages [102]. This is in contrast to autoregressive
883 architectures [61, 103], in which only one of the forward or backward passes can be efficient. In this
884 work, normalizing flows are used as flexible parameter distribution approximations $q_{\theta}(\mathbf{z})$ having
885 weights and biases θ . We specify the architecture used in each application by the number of Real-
886 NVP affine coupling stages, and the number of neural network layers and units per layer of the
887 conditioning functions.

888 When calculating Hessians of log probabilities in deep probability distributions, it is important to
889 consider the normalizing flow architecture. With autoregressive architectures [61, 103], fast sam-
890 pling and fast log probability evaluations are mutually exclusive. That makes these architectures
891 undesirable for EPI, where efficient sampling is important for optimization, and log probability
892 evaluation speed predicates the efficiency of gradient and Hessian calculations. With Real NVP
893 coupling architectures, we get both fast sampling and fast Hessians making both optimization and
894 scientific analysis efficient.

895 5.1.3 Augmented Lagrangian optimization

896 To optimize $q_{\theta}(\mathbf{z})$ in Equation 13, the constrained maximum entropy optimization is executed using
897 the augmented Lagrangian method. The following objective is minimized:

$$L(\theta; \eta_{\text{opt}}, c) = -H(q_{\theta}) + \eta_{\text{opt}}^T R(\theta) + \frac{c}{2} \|R(\theta)\|^2 \quad (15)$$

898 where average constraint violations $R(\theta) = \mathbb{E}_{\mathbf{z} \sim q_{\theta}} [\mathbb{E}_{\mathbf{x} \sim p(\mathbf{x}|\mathbf{z})} [T(\mathbf{x}; \mathbf{z}) - \mu_{\text{opt}}]]$, $\eta_{\text{opt}} \in \mathbb{R}^m$ are the
899 Lagrange multipliers where $m = |\mu_{\text{opt}}| = |T(\mathbf{x}; \mathbf{z})| = 2|f(\mathbf{x}; \mathbf{z})|$, and c is the penalty coefficient. The
900 sufficient statistics $T(\mathbf{x}; \mathbf{z})$ and mean parameter μ_{opt} are determined by the means μ and variances
901 σ^2 of emergent property statistics $f(\mathbf{x}; \mathbf{z})$ defined in Equation 13 (see Section 5.1.6). Specifically,
902 $T(\mathbf{x}; \mathbf{z})$ is a concatenation of the first and second moments, μ_{opt} is a concatenation of μ and σ^2
903 (see section 5.1.5), and the Lagrange multipliers are closely related to the natural parameters η of
904 exponential families (see Section 5.1.5). Weights and biases θ of the deep probability distribution
905 are optimized according to Equation 15 using the Adam optimizer with learning rate 10^{-3} [104].

906 The gradient with respect to entropy $H(q_{\theta}(\mathbf{z}))$ can be expressed using the reparameterization trick
 907 as an expectation of the negative log density of parameter samples \mathbf{z} over the randomness in the
 908 parameterless initial distribution $q_0(\mathbf{z}_0)$:

$$H(q_{\theta}(\mathbf{z})) = \int -q_{\theta}(\mathbf{z}) \log(q_{\theta}(\mathbf{z})) d\mathbf{z} = \mathbb{E}_{\mathbf{z} \sim q_{\theta}} [-\log(q_{\theta}(\mathbf{z}))] = \mathbb{E}_{\mathbf{z}_0 \sim q_0} [-\log(q_{\theta}(g_{\theta}(\mathbf{z}_0)))]. \quad (16)$$

909 Thus, the gradient of the entropy of the deep probability distribution can be estimated as an
 910 average with respect to the base distribution \mathbf{z}_0 :

$$\nabla_{\theta} H(q_{\theta}(\mathbf{z})) = \mathbb{E}_{\mathbf{z}_0 \sim q_0} [-\nabla_{\theta} \log(q_{\theta}(g_{\theta}(\mathbf{z}_0)))]. \quad (17)$$

911 The lagrangian parameters η_{opt} are initialized to zero and adapted following each augmented
 912 Lagrangian epoch, which is a period of optimization with fixed (η_{opt}, c) for a given number of
 913 stochastic optimization iterations. A low value of c is used initially, and conditionally increased
 914 after each epoch based on constraint error reduction. The penalty coefficient is updated based
 915 on the result of a hypothesis test regarding the reduction in constraint violation. The p-value of
 916 $\mathbb{E}[|R(\theta_{k+1})|] > \gamma \mathbb{E}[|R(\theta_k)|]$ is computed, and c_{k+1} is updated to βc_k with probability $1 - p$. The
 917 other update rule is $\eta_{\text{opt},k+1} = \eta_{\text{opt},k} + c_k \frac{1}{n} \sum_{i=1}^n (T(\mathbf{x}^{(i)}) - \mu_{\text{opt}})$ given a batch size n . Throughout
 918 the study, $\gamma = 0.25$, while β was chosen to be either 2 or 4. The batch size of EPI also varied
 919 according to application.

920 The intention is that c and η_{opt} start at values encouraging entropic growth early in optimization.
 921 With each training epoch in which the update rule for c is invoked by unsatisfactory constraint
 922 error reduction, the constraint satisfaction terms are increasingly weighted, resulting in a decreased
 923 entropy. This encourages the discovery of suitable regions of parameter space, and the subsequent
 924 refinement of the distribution to produce the emergent property (see example in Section 5.1.4). The
 925 momentum parameters of the Adam optimizer are reset at the end of each augmented Lagrangian
 926 epoch.

927 Rather than starting optimization from some θ drawn from a randomized distribution, we found
 928 that initializing $q_{\theta}(\mathbf{z})$ to approximate an isotropic Gaussian distribution conferred more stable, con-
 929 sistent optimization. The parameters of the Gaussian initialization were chosen on an application-
 930 specific basis. Throughout the study, we chose isotropic Gaussian initializations with mean μ_{init}
 931 at the center of the distribution support and some standard deviation σ_{init} , except for one case,
 932 where an initialization informed by random search was used (see Section 5.2).

933 To assess whether the EPI distribution $q_{\theta}(\mathbf{z})$ produces the emergent property, we assess whether
 934 each individual constraint on the means and variances of $f(\mathbf{x}; \mathbf{z})$ is satisfied. We consider the EPI

935 to have converged when a null hypothesis test of constraint violations $R(\boldsymbol{\theta})_i$ being zero is accepted
 936 for all constraints $i \in \{1, \dots, m\}$ at a significance threshold $\alpha = 0.05$. This significance threshold is
 937 adjusted through Bonferroni correction according to the number of constraints m . The p-values for
 938 each constraint are calculated according to a two-tailed nonparametric test, where 200 estimations
 939 of the sample mean $R(\boldsymbol{\theta})^i$ are made using N_{test} samples of $\mathbf{z} \sim q_{\boldsymbol{\theta}}(\mathbf{z})$ at the end of the augmented
 940 Lagrangian epoch.

941 When assessing the suitability of EPI for a particular modeling question, there are some important
 942 technical considerations. First and foremost, as in any optimization problem, the defined emergent
 943 property should always be appropriately conditioned (constraints should not have wildly different
 944 units). Furthermore, if the program is underconstrained (not enough constraints), the distribution
 945 grows (in entropy) unstably unless mapped to a finite support. If overconstrained, there is no pa-
 946 rameter set producing the emergent property, and EPI optimization will fail (appropriately). Next,
 947 one should consider the computational cost of the gradient calculations. In the best circumstance,
 948 there is a simple, closed form expression (e.g. Section 5.3) for the emergent property statistic given
 949 the model parameters. On the other end of the spectrum, many forward simulation iterations
 950 may be required before a high quality measurement of the emergent property statistic is available
 951 (e.g. Section 5.2). In such cases, backpropagating gradients through the SDE evolution will be
 952 expensive.

953 5.1.4 Example: 2D LDS

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

$$\tau \frac{d\mathbf{x}}{dt} = A\mathbf{x} \quad (18)$$

956 with

$$A = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix}. \quad (19)$$

957 To run EPI with the dynamics matrix elements as the free parameters $\mathbf{z} = [a_1, a_2, a_3, a_4]$ (fix-
 958 ing $\tau = 1$), the emergent property statistics $T(\mathbf{x})$ were chosen to contain the first and second
 959 moments of the oscillatory frequency, $\frac{\text{imag}(\lambda_1)}{2\pi}$, and the growth/decay factor, $\text{real}(\lambda_1)$, of the oscil-
 960 lating system. λ_1 is the eigenvalue of greatest real part when the imaginary component is zero, and
 961 alternatively of positive imaginary component when the eigenvalues are complex conjugate pairs.
 962 To learn the distribution of real entries of A that produce a band of oscillating systems around

963 1Hz, we formalized this emergent property as $\text{real}(\lambda_1)$ having mean zero with variance 0.25^2 , and
 964 the oscillation frequency $2\pi\text{imag}(\lambda_1)$ having mean $\omega = 1$ Hz with variance $(0.1\text{Hz})^2$:

$$\mathbb{E}[T(\mathbf{x})] \triangleq \mathbb{E} \begin{bmatrix} \text{real}(\lambda_1) \\ \text{imag}(\lambda_1) \\ (\text{real}(\lambda_1) - 0)^2 \\ (\text{imag}(\lambda_1) - 2\pi\omega)^2 \end{bmatrix} = \begin{bmatrix} 0.0 \\ 2\pi\omega \\ 0.25^2 \\ (2\pi 0.1)^2 \end{bmatrix} \triangleq \boldsymbol{\mu}. \quad (20)$$

965

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

$$\lambda = \frac{\left(\frac{a_1+a_4}{\tau}\right) \pm \sqrt{\left(\frac{a_1+a_4}{\tau}\right)^2 + 4\left(\frac{a_2a_3-a_1a_4}{\tau}\right)}}{2}. \quad (21)$$

969 Importantly, even though $\mathbb{E}_{\mathbf{x} \sim p(\mathbf{x}|\mathbf{z})}[T(\mathbf{x})]$ is calculable directly via a closed form function and
 970 does not require simulation, we cannot derive the distribution $q_{\boldsymbol{\theta}}^*$ directly. This fact is due to the
 971 formally hard problem of the backward mapping: finding the natural parameters η from the mean
 972 parameters $\boldsymbol{\mu}$ of an exponential family distribution [93]. Instead, we used EPI to approximate this
 973 distribution (Fig. S1B). We used a real-NVP normalizing flow architecture with four masks, two
 974 neural network layers of 15 units per mask, with batch normalization momentum 0.99, mapped
 975 onto a support of $z_i \in [-10, 10]$. (see Section 5.1.2).

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

984 To explain the bimodality of the EPI distribution, we examined the imaginary component of λ_1 .

985 When $\text{real}(\lambda_1) = \frac{a_1+a_4}{2} = 0$, we have

$$\text{imag}(\lambda_1) = \begin{cases} \sqrt{\frac{a_1a_4-a_2a_3}{\tau}}, & \text{if } a_1a_4 < a_2a_3 \\ 0 & \text{otherwise} \end{cases}. \quad (22)$$

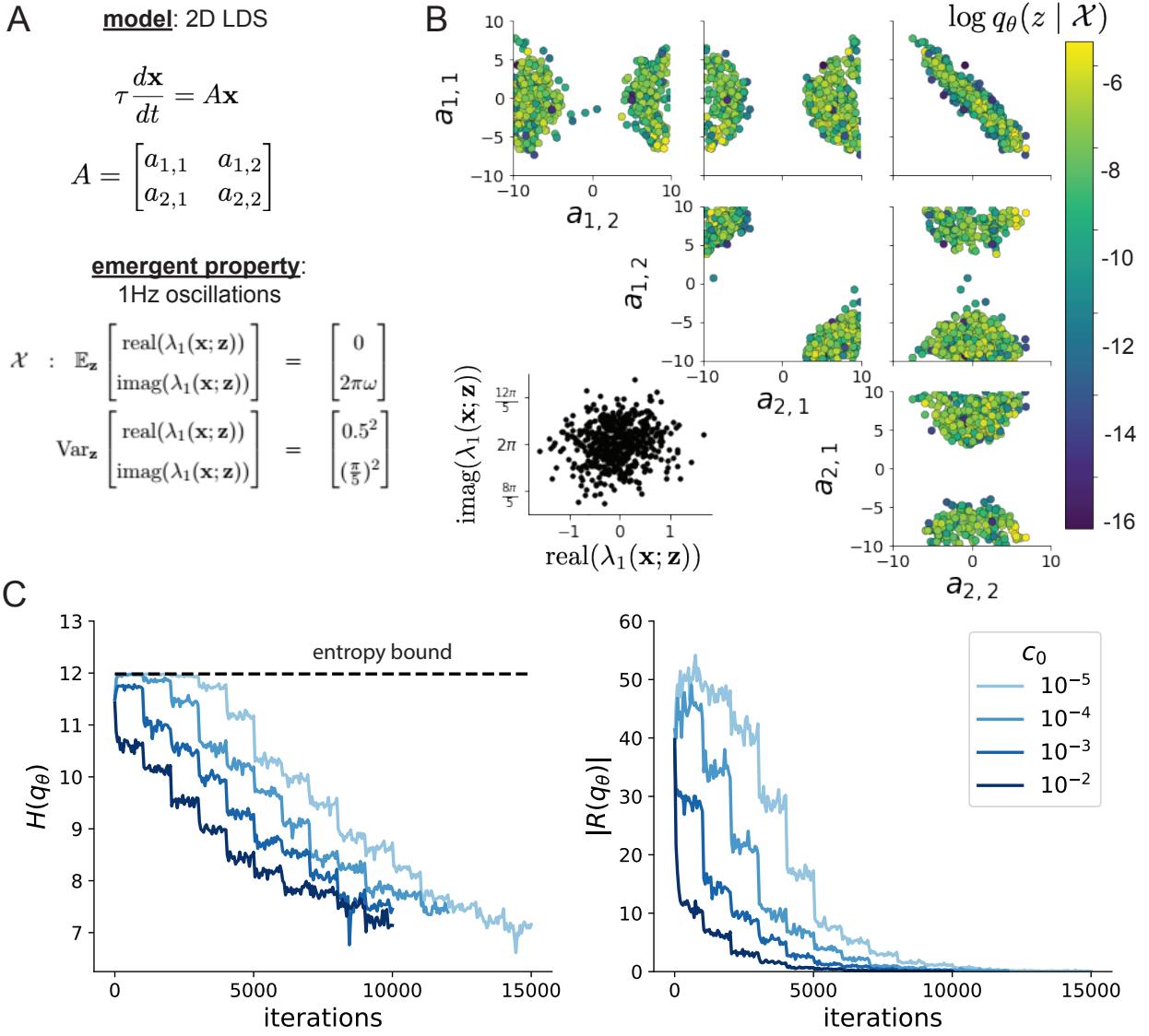


Figure 6: (LDS1): **A.** Two-dimensional linear dynamical system model, where real entries of the dynamics matrix A are the parameters. **B.** The EPI distribution for a two-dimensional linear dynamical system with $\tau = 1$ that produces an average of 1Hz oscillations with some small amount of variance. Dashed lines indicate the parameter axes. **C.** Entropy throughout the optimization. At the beginning of each augmented Lagrangian epoch (2,000 iterations), the entropy dipped due to the shifted optimization manifold where emergent property constraint satisfaction is increasingly weighted. **D.** Emergent property moments throughout optimization. At the beginning of each augmented Lagrangian epoch, the emergent property moments adjust closer to their constraints.

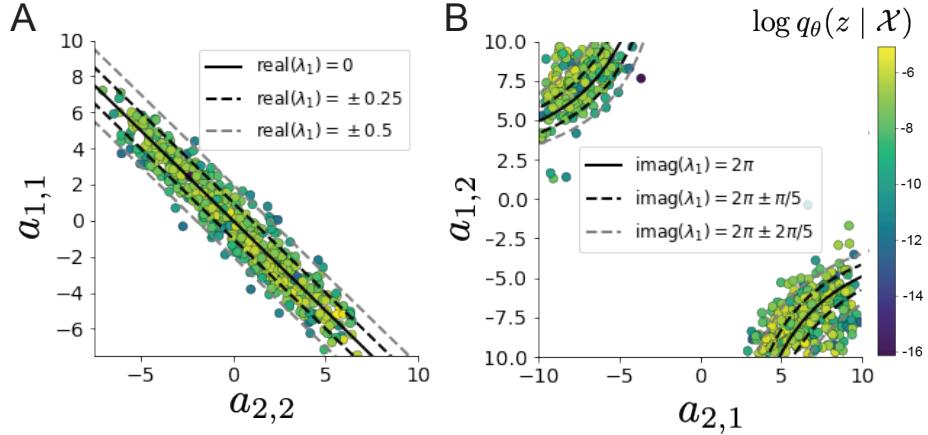


Figure 7: (LDS2): **A.** Probability contours in the a_1 - a_4 plane were derived from the relationship to emergent property statistic of growth/decay factor $\text{real}(\lambda_1)$. **B.** Probability contours in the a_2 - a_3 plane were derived from the emergent property statistic of oscillation frequency $2\pi\text{imag}(\lambda_1)$.

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

$$\text{imag}(\lambda_1)^2 = a_1a_4 - a_2a_3 \quad (23)$$

988 Since we constrained $\mathbb{E}_{z \sim q_\theta} [\text{imag}(\lambda)] = 2\pi$ (with $\omega = 1$), we can plot contours of the equation
 989 $\text{imag}(\lambda_1)^2 = a_1a_4 - a_2a_3 = (2\pi)^2$ for various a_1a_4 (Fig. S2B). With $\sigma_{1,4} = \mathbb{E}_{z \sim q_\theta} (|a_1a_4 - E_{q_\theta}[a_1a_4]|)$,
 990 we show the contours as $a_1a_4 = 0$ (black), $a_1a_4 = -\sigma_{1,4}$ (black dotted), and $a_1a_4 = -2\sigma_{1,4}$ (grey
 991 dotted). This validates the curved structure of the inferred distribution learned through EPI. We
 992 took steps in negative standard deviation of a_1a_4 (dotted and gray lines), since there are few positive
 993 values a_1a_4 in the learned distribution. Subtler combinations of model and emergent property will
 994 have more complexity, further motivating the use of EPI for understanding these systems. As we
 995 expect, the distribution results in samples of two-dimensional linear systems oscillating near 1Hz
 996 (Fig. S3).

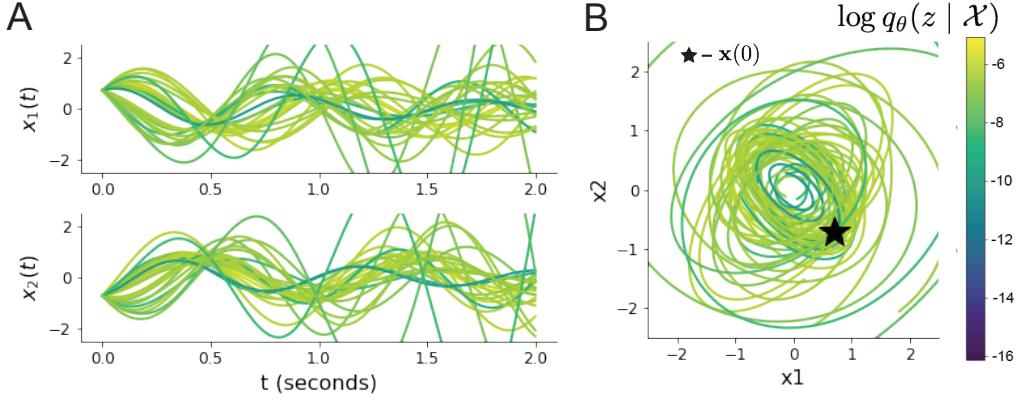


Figure 8: (LDS3): Sampled dynamical systems $\mathbf{z} \sim q_\theta(\mathbf{z})$ and their simulated activity from $\mathbf{x}(0) = [\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2}]$ colored by log probability. **A.** Each dimension of the simulated trajectories throughout time. **B.** The simulated trajectories in phase space.

997 5.1.5 Maximum entropy distributions and exponential families

998 EPI is a maximum entropy distribution, which have fundamental links to exponential family dis-
 999 tributions. A maximum entropy distribution of form:

$$p^*(\mathbf{z}) = \underset{p \in \mathcal{P}}{\operatorname{argmax}} H(p(\mathbf{z})) \quad (24)$$

s.t. $\mathbb{E}_{\mathbf{z} \sim p}[T(\mathbf{z})] = \boldsymbol{\mu}_{\text{opt.}}$

1000 will have probability density in the exponential family:

$$p^*(\mathbf{z}) \propto \exp(\boldsymbol{\eta}^\top T(\mathbf{z})). \quad (25)$$

1001 The mappings between the mean parameterization $\boldsymbol{\mu}_{\text{opt}}$ and the natural parameterization $\boldsymbol{\eta}$ are
 1002 formally hard to identify except in special cases [93].

1003 In EPI, emergent properties are defined as statistics having a fixed mean and variance as in Equation
 1004 4. The variance constraint is a second moment constraint on $f(\mathbf{x}; \mathbf{z})$

$$\operatorname{Var}_{\mathbf{z}, \mathbf{x}}[f(\mathbf{x}; \mathbf{z})] = \mathbb{E}_{\mathbf{z}, \mathbf{x}}[(f(\mathbf{x}; \mathbf{z}) - \boldsymbol{\mu})^2] \quad (26)$$

1005 As a general maximum entropy distribution (Equation 24), the sufficient statistics vector contains
 1006 both first and second order moments of $f(\mathbf{x}; \mathbf{z})$

$$T(\mathbf{x}; \mathbf{z}) = \begin{bmatrix} f(\mathbf{x}; \mathbf{z}) \\ (f(\mathbf{x}; \mathbf{z}) - \boldsymbol{\mu})^2 \end{bmatrix}, \quad (27)$$

1007 which are constrained to the chosen means and variances

$$\boldsymbol{\mu}_{\text{opt}} = \begin{bmatrix} \boldsymbol{\mu} \\ \sigma^2 \end{bmatrix}. \quad (28)$$

1008 5.1.6 EPI as variational inference

1009 In Bayesian inference a prior belief about model parameters \mathbf{z} is stated in a prior distribution $p(\mathbf{z})$,
 1010 and the statistical model capturing the effect of \mathbf{z} on observed data points \mathbf{x} is formalized in the
 1011 likelihood distribution $p(\mathbf{x} | \mathbf{z})$. In Bayesian inference, we obtain a posterior distribution $p(\mathbf{z} | \mathbf{x})$,
 1012 which captures how the data inform our knowledge of model parameters using Bayes' rule:

$$p(\mathbf{z} | \mathbf{x}) = \frac{p(\mathbf{x} | \mathbf{z})p(\mathbf{z})}{p(\mathbf{x})}. \quad (29)$$

1013 The posterior distribution is analytically available when the prior is conjugate with the likelihood.
 1014 However, conjugacy is rare in practice, and alternative methods, such as variational inference [82],
 1015 are utilized.

1016 In variational inference, a posterior approximation $q_{\boldsymbol{\theta}}^*$ is chosen from within some variational family
 1017 \mathcal{Q}

$$q_{\boldsymbol{\theta}}^*(\mathbf{z}) = \underset{q_{\boldsymbol{\theta}} \in \mathcal{Q}}{\operatorname{argmin}} KL(q_{\boldsymbol{\theta}}(\mathbf{z}) || p(\mathbf{z} | \mathbf{x})). \quad (30)$$

1018 The KL divergence can be written in terms of entropy of the variational approximation:

$$KL(q_{\boldsymbol{\theta}}(\mathbf{z}) || p(\mathbf{z} | \mathbf{x})) = \mathbb{E}_{\mathbf{z} \sim q_{\boldsymbol{\theta}}} [\log(q_{\boldsymbol{\theta}}(\mathbf{z}))] - \mathbb{E}_{\mathbf{z} \sim q_{\boldsymbol{\theta}}} [\log(p(\mathbf{z} | \mathbf{x}))] \quad (31)$$

$$= -H(q_{\boldsymbol{\theta}}) - \mathbb{E}_{\mathbf{z} \sim q_{\boldsymbol{\theta}}} [\log(p(\mathbf{x} | \mathbf{z})) + \log(p(\mathbf{z})) - \log(p(\mathbf{x}))] \quad (32)$$

1019 Since the marginal distribution of the data $p(\mathbf{x})$ (or ‘‘evidence’’) is independent of $\boldsymbol{\theta}$, variational
 1020 inference is executed by optimizing the remaining expression. This is usually framed as maximizing
 1021 the evidence lower bound (ELBO)

$$\underset{q_{\boldsymbol{\theta}} \in \mathcal{Q}}{\operatorname{argmin}} KL(q_{\boldsymbol{\theta}} || p(\mathbf{z} | \mathbf{x})) = \underset{q_{\boldsymbol{\theta}} \in \mathcal{Q}}{\operatorname{argmax}} H(q_{\boldsymbol{\theta}}) + \mathbb{E}_{\mathbf{z} \sim q_{\boldsymbol{\theta}}} [\log(p(\mathbf{x} | \mathbf{z})) + \log(p(\mathbf{z}))]. \quad (33)$$

1023 Now, consider the setting where we have chosen a uniform prior, and stipulate a mean-field gaussian
 1024 likelihood on a chosen statistic of the data $f(\mathbf{x}; \mathbf{z})$

$$p(\mathbf{x} | \mathbf{z}) = \mathcal{N}(f(\mathbf{x}; \mathbf{z}) | \boldsymbol{\mu}_f, \Sigma_f), \quad (34)$$

1025 where $\Sigma_f = \text{diag}(\boldsymbol{\sigma}_f^2)$. The log likelihood is then proportional to a dot product of the natural
 1026 parameter of this mean-field gaussian distribution and the first and second moment statistics.

$$\log p(\mathbf{x} | \mathbf{z}) \propto \boldsymbol{\eta}_f^\top T(\mathbf{x}, \mathbf{z}), \quad (35)$$

1027 where

$$\boldsymbol{\eta}_f = \begin{bmatrix} \frac{\boldsymbol{\mu}_f}{\sigma_f^2} \\ \frac{-1}{2\sigma_f^2} \end{bmatrix}, \text{ and} \quad (36)$$

1028

$$T(\mathbf{x}; \mathbf{z}) = \begin{bmatrix} f(\mathbf{x}; \mathbf{z}) \\ (f(\mathbf{x}; \mathbf{z}) - \boldsymbol{\mu}_f)^2 \end{bmatrix}. \quad (37)$$

1029 The variational objective is then

$$\operatorname{argmax}_{q_{\theta} \in Q} H(q_{\theta}) + \boldsymbol{\eta}_f^\top \mathbb{E}_{\mathbf{z} \sim q_{\theta}} [T(\mathbf{x}; \mathbf{z})] \quad (38)$$

1030 Comparing this to the Lagrangian objective (without augmentation) of EPI, we see they are the

1031 same

$$\begin{aligned} q_{\theta}^*(\mathbf{z}) &= \operatorname{argmin}_{q_{\theta} \in Q} -H(q_{\theta}) + \boldsymbol{\eta}_{\text{opt}}^\top (\mathbb{E}_{\mathbf{z}, \mathbf{x}} [T(\mathbf{x}; \mathbf{z})] - \boldsymbol{\mu}_{\text{opt}}) \\ &= \operatorname{argmin}_{q_{\theta} \in Q} -H(q_{\theta}) + \boldsymbol{\eta}_{\text{opt}}^\top \mathbb{E}_{\mathbf{z}, \mathbf{x}} [T(\mathbf{x}; \mathbf{z})]. \end{aligned} \quad (39)$$

1032 where $T(\mathbf{x}; \mathbf{z})$ consists of the first and second moments of the emergent property statistic $f(\mathbf{x}; \mathbf{z})$
1033 (Equation 27). Thus, EPI is implicitly executing variational inference with a uniform prior and a
1034 mean-field gaussian likelihood on the emergent property statistics. The mean and variances of the
1035 mean-field gaussian likelihood are predicated by $\boldsymbol{\eta}_{\text{opt}}$ (Equations 36 and 38), which is adapted after
1036 each EPI optimization epoch based on \mathcal{X} (see Section 5.1.3). In EPI, the inferred distribution is
1037 not conditioned on a finite dataset as in variational inference, but rather the emergent property
1038 \mathcal{X} dictates the likelihood parameterization such that the inferred distribution will produce the
1039 emergent property. As a note, we could not simply choose $\boldsymbol{\mu}_f$ and σ_f directly from the outset, since
1040 we do not know which of these choices will produce the emergent property \mathcal{X} , which necessitates
1041 the EPI optimization routine that adapts $\boldsymbol{\eta}_{\text{opt}}$. Accordingly, we replace the notation of $p(\mathbf{z} | \mathbf{x})$
1042 with $p(\mathbf{z} | \mathcal{X})$ conceptualizing an inferred distribution that obeys emergent property \mathcal{X} (see Section
1043 5.1).

1044 5.2 Stomatogastric ganglion

1045 In Section 3.1 and 3.2, we used EPI to infer conductance parameters in a model of the stomatogastric
1046 ganglion (STG) [51]. This 5-neuron circuit model represents two subcircuits: that generating the
1047 pyloric rhythm (fast population) and that generating the gastric mill rhythm (slow population).
1048 The additional neuron (the IC neuron of the STG) receives inhibitory synaptic input from both
1049 subcircuits, and can couple to either rhythm dependent on modulatory conditions. There is also

1050 a parametric regime in which this neuron fires at an intermediate frequency between that of the
 1051 fast and slow populations [51], which we infer with EPI as a motivational example. This model
 1052 is not to be confused with an STG subcircuit model of the pyloric rhythm [69], which has been
 1053 statistically inferred in other studies [12, 45].

1054 **5.2.1 STG model**

1055 We analyze how the parameters $\mathbf{z} = [g_{el}, g_{synA}]$ govern the emergent phenomena of intermediate
 1056 hub frequency in a model of the stomatogastric ganglion (STG) [51] shown in Figure 1A with
 1057 activity $\mathbf{x} = [x_{f1}, x_{f2}, x_{hub}, x_{s1}, x_{s2}]$, using the same hyperparameter choices as Gutierrez et al.
 1058 Each neuron's membrane potential $x_\alpha(t)$ for $\alpha \in \{f1, f2, hub, s1, s2\}$ is the solution of the following
 1059 stochastic differential equation:

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

1060 The input current of each neuron is the sum of the leak, calcium, potassium, hyperpolarization,
 1061 electrical and synaptic currents. Each current component is a function of all membrane potentials
 1062 and the conductance parameters \mathbf{z} . Finally, we include gaussian noise dB to the model of Gutierrez
 1063 et al. so that the model stochastic, although this is not required by EPI.

1064 The capacitance of the cell membrane was set to $C_m = 1nF$. Specifically, the currents are the
 1065 difference in the neuron's membrane potential and that current type's reversal potential multiplied
 1066 by a conductance:

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

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

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

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

$$h_K(\mathbf{x}; \mathbf{z}) = g_KN(x_\alpha - V_K) \quad (45)$$

$$h_{hyp}(\mathbf{x}; \mathbf{z}) = g_hH(x_\alpha - V_{hyp}). \quad (46)$$

1072 The reversal potentials were set to $V_{leak} = -40mV$, $V_{Ca} = 100mV$, $V_K = -80mV$, $V_{hyp} = -20mV$,
 1073 and $V_{syn} = -75mV$. The other conductance parameters were fixed to $g_{leak} = 1 \times 10^{-4}\mu S$, g_{Ca} ,
 1074 g_K , and g_{hyp} had different values based on fast, intermediate (hub) or slow neuron. The fast
 1075 conductances had values $g_{Ca} = 1.9 \times 10^{-2}$, $g_K = 3.9 \times 10^{-2}$, and $g_{hyp} = 2.5 \times 10^{-2}$. The intermediate

conductances had values $g_{Ca} = 1.7 \times 10^{-2}$, $g_K = 1.9 \times 10^{-2}$, and $g_{hyp} = 8.0 \times 10^{-3}$. Finally, the slow conductances had values $g_{Ca} = 8.5 \times 10^{-3}$, $g_K = 1.5 \times 10^{-2}$, and $g_{hyp} = 1.0 \times 10^{-2}$.

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

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

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

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

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

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

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

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

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

Finally, there is a synaptic gating variable as well:

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

When the dynamic gating variables are considered, this is actually a 15-dimensional nonlinear dynamical system. The gaussian noise $d\mathbf{B}$ has variance $(1 \times 10^{-12})^2$ A², and introduces variability in frequency at each parameterization \mathbf{z} .

5.2.2 Hub frequency calculation

In order to measure the frequency of the hub neuron during EPI, the STG model was simulated for $T = 300$ time steps of $dt = 25\text{ms}$. The chosen dt and T were the most computationally convenient choices yielding accurate frequency measurement. We used a basis of complex exponentials with frequencies from 0.0-1.0 Hz at 0.01Hz resolution to measure frequency from simulated time series

$$\Phi = [0.0, 0.01, \dots, 1.0]^\top \dots \quad (55)$$

1097 To measure spiking frequency, we processed simulated membrane potentials with a relu (spike
 1098 extraction) and low-pass filter with averaging window of size 20, then took the frequency with the
 1099 maximum absolute value of the complex exponential basis coefficients of the processed time-series.
 1100 The first 20 temporal samples of the simulation are ignored to account for initial transients.
 1101 To differentiate through the maximum frequency identification, we used a soft-argmax Let $X_\alpha \in$
 1102 $\mathcal{C}^{|\Phi|}$ be the complex exponential filter bank dot products with the signal $x_\alpha \in \mathbb{R}^N$, where $\alpha \in$
 1103 $\{f1, f2, \text{hub}, s1, s2\}$. The soft-argmax is then calculated using temperature parameter $\beta = 100$

$$\psi_\alpha = \text{softmax}(\beta |X_\alpha| \odot i), \quad (56)$$

1104 where $i = [0, 1, \dots, 100]$. The frequency is then calculated as

$$\omega_\alpha = 0.01\psi_\alpha \text{Hz}. \quad (57)$$

1105 Intermediate hub frequency, like all other emergent properties in this work, is defined by the mean
 1106 and variance of the emergent property statistics. In this case, we have one statistic, hub neuron
 1107 frequency, where the mean was chosen to be 0.55Hz, and variance was chosen to be $(0.025\text{Hz})^2$
 1108 (Equation 4).

1109 5.2.3 EPI details for the STG model

1110 As a maximum entropy distribution, $T(\mathbf{x}; \mathbf{z})$ is comprised of both these first and second moments
 1111 of the hub neuron frequency (as in Equations 27 and 28)

$$T(\mathbf{x}; \mathbf{z}) = \begin{bmatrix} \omega_{\text{hub}}(\mathbf{x}; \mathbf{z}) \\ (\omega_{\text{hub}}(\mathbf{x}; \mathbf{z}) - 0.55)^2 \end{bmatrix}, \quad (58)$$

$$\boldsymbol{\mu}_{\text{opt}} = \begin{bmatrix} 0.55 \\ 0.025^2 \end{bmatrix}. \quad (59)$$

1112 Throughout optimization, the augmented Lagrangian parameters η and c , were updated after each
 1113 epoch of 5,000 iterations(see Section 5.1.3). The optimization converged after five epochs (Fig. S4).

1114 For EPI in Fig 1E, we used a real NVP architecture with three Real NVP coupling layers and two-
 1115 layer neural networks of 25 units per layer. The normalizing flow architecture mapped $\mathbf{z}_0 \sim \mathcal{N}(\mathbf{0}, I)$
 1116 to a support of $\mathbf{z} = [g_{\text{el}}, g_{\text{synA}}] \in [4, 8] \times [0.01, 4]$, initialized to a gaussian approximation of samples
 1117 returned by a preliminary ABC search. We did not include $g_{\text{synA}} < 0.01$, for numerical stability.
 1118 EPI optimization was run using 5 different random seeds for architecture initialization $\boldsymbol{\theta}$ with an

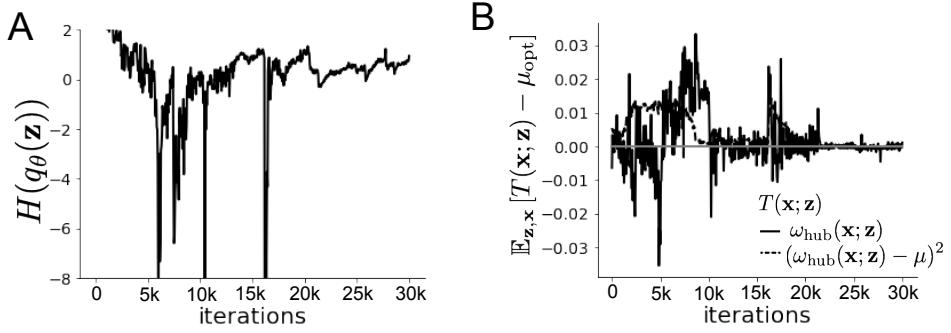


Figure 9: (STG1): EPI optimization of the STG model producing network syncing. **A.** Entropy throughout optimization. **B.** The emergent property statistic means and variances converge to their constraints at 25,000 iterations following the fifth augmented Lagrangian epoch.

augmented Lagrangian coefficient of $c_0 = 10^5$, a batch size $n = 400$, and $\beta = 2$. The architecture converged with criteria $N_{\text{test}} = 100$.

5.2.4 Hessian sensitivity vectors

To quantify the second-order structure of the EPI distribution, we evaluated the Hessian of the log probability $\frac{\partial^2 \log q(\mathbf{z}|\mathcal{X})}{\partial \mathbf{z} \partial \mathbf{z}^T}$. The eigenvector of this Hessian with most negative eigenvalue is defined as the sensitivity dimension \mathbf{v}_1 , and all subsequent eigenvectors are ordered by increasing eigenvalue. These eigenvalues are quantifications of how fast the emergent property deteriorates via the parameter combination of their associated eigenvector. In Figure 1D, the sensitivity dimension v_1 (solid) and the second eigenvector of the Hessian v_2 (dashed) are shown evaluated at the mode of the distribution. Since the Hessian eigenvectors have sign degeneracy, the visualized directions in 2-D parameter space were chosen to have positive g_{synA} . The length of the arrows is inversely proportional to the square root of the absolute value of their eigenvalues $\lambda_1 = -10.7$ and $\lambda_2 = -3.22$. For the same magnitude perturbation away from the mode, intermediate hub frequency only diminishes along the sensitivity dimension \mathbf{v}_1 (Fig. 1E-F).

₁₁₃₄ **5.3 Scaling EPI for stable amplification in RNNs**

₁₁₃₅ **5.3.1 Rank-2 RNN model**

₁₁₃₆ We examined the scaling properties of EPI by learning connectivities of RNNs of increasing size
₁₁₃₇ that exhibit stable amplification. Rank-2 RNN connectivity was modeled as $W = UV^\top$, where
₁₁₃₈ $U = [\mathbf{u}_1 \ \mathbf{u}_2] + g\chi^{(W)}$, $V = [\mathbf{v}_1 \ \mathbf{v}_2] + g\chi^{(V)}$, and $\chi_{i,j}^{(W)}, \chi_{i,j}^{(V)} \sim \mathcal{N}(0, 1)$. This RNN model has
₁₁₃₉ dynamics

$$\tau \dot{\mathbf{x}} = -\mathbf{x} + W\mathbf{x}. \quad (60)$$

₁₁₄₀ In this analysis, we inferred connectivity parameterizations $\mathbf{z} = [\mathbf{u}_1^\top, \mathbf{u}_2^\top, \mathbf{v}_1^\top, \mathbf{v}_2^\top]^\top \in [-1, 1]^{(4N)}$
₁₁₄₁ that produced stable amplification using EPI, SMC-ABC [43], and SNPE [45] (see Section Related
₁₁₄₂ Methods).

₁₁₄₃ **5.3.2 Stable amplification**

₁₁₄₄ For this RNN model to be stable, all real eigenvalues of W must be less than 1: $\text{real}(\lambda_1) < 1$,
₁₁₄₅ where λ_1 denotes the greatest real eigenvalue of W . For a stable RNN to amplify at least one input
₁₁₄₆ pattern, the symmetric connectivity $W^s = \frac{W+W^\top}{2}$ must have an eigenvalue greater than 1: $\lambda_1^s > 1$,
₁₁₄₇ where λ^s is the maximum eigenvalue of W^s . These two conditions are necessary and sufficient for
₁₁₄₈ stable amplification in RNNs [63].

₁₁₄₉ **5.3.3 EPI details for RNNs**

₁₁₅₀ We defined the emergent property of stable amplification with means of these eigenvalues (0.5
₁₁₅₁ and 1.5, respectively) that satisfy these conditions. To complete the emergent property definition,
₁₁₅₂ we chose variances (0.25^2) about those means such that samples rarely violate the eigenvalue
₁₁₅₃ constraints. In terms of the EPI optimization variables, this is written as

$$T(\mathbf{x}; \mathbf{z}) = \begin{bmatrix} \text{real}(\lambda_1)(\mathbf{x}; \mathbf{z}) \\ \lambda_1^s(\mathbf{x}; \mathbf{z}) \\ (\text{real}(\lambda_1)(\mathbf{x}; \mathbf{z}) - 0.5)^2 \\ (\lambda_1^s(\mathbf{x}; \mathbf{z}) - 1.5)^2 \end{bmatrix}, \quad (61)$$

$$\boldsymbol{\mu}_{\text{opt}} = \begin{bmatrix} 0.5 \\ 1.5 \\ 0.25^2 \\ 0.25^2 \end{bmatrix}. \quad (62)$$

1155 Gradients of maximum eigenvalues of Hermitian matrices like W^s are available with modern auto-
 1156 automatic differentiation tools. To differentiate through the $\text{real}(\lambda_1)$, we solved the following equation
 1157 for eigenvalues of rank-2 matrices using the rank reduced matrix $W^r = V^\top U$

$$\lambda_{\pm} = \frac{\text{Tr}(W^r) \pm \sqrt{\text{Tr}(W^r)^2 - 4\text{Det}(W^r)}}{2}. \quad (63)$$

1158 For EPI in Fig. 2, we used a real NVP architecture with three coupling layers of affine transfor-
 1159 mations parameterized by two-layer neural networks of 100 units per layer. The initial distribution
 1160 was a standard isotropic gaussian $\mathbf{z}_0 \sim \mathcal{N}(\mathbf{0}, I)$ mapped to the support of $\mathbf{z}_i \in [-1, 1]$. We used
 1161 an augmented Lagrangian coefficient of $c_0 = 10^3$, a batch size $n = 200$, $\beta = 4$, and chose to use
 1162 500 iterations per augmented Lagrangian epoch and emergent property constraint convergence was
 1163 evaluated at $N_{\text{test}} = 200$ (Fig. 2B blue line, and Fig. 2C-D blue).

1164 5.3.4 Methodological comparison

1165 We compared EPI to two alternative simulation-based inference techniques, since the likelihood
 1166 of these eigenvalues given \mathbf{z} is not available. Approximate Bayesian computation (ABC) [80] is a
 1167 rejection sampling technique for obtaining sets of parameters \mathbf{z} that produce activity \mathbf{x} close to some
 1168 observed data \mathbf{x}_0 . Sequential Monte Carlo approximate Bayesian computation (SMC-ABC) is the
 1169 state-of-the-art ABC method, which leverages SMC techniques to improve sampling speed. We ran
 1170 SMC-ABC with the pyABC package [105] to infer RNNs with stable amplification: connectivities
 1171 having eigenvalues within an ϵ -defined l_2 distance of

$$x_0 = \begin{bmatrix} \text{real}(\lambda_1) \\ \lambda_1^s \end{bmatrix} = \begin{bmatrix} 0.5 \\ 1.5 \end{bmatrix}. \quad (64)$$

1172 SMC-ABC was run with a uniform prior over $\mathbf{z} \in [-1, 1]^{(4N)}$, a population size of 1,000 particles
 1173 with simulations parallelized over 32 cores, and a multivariate normal transition model.

1174 SNPE, the next approach in our comparison, is far more similar to EPI. Like EPI, SNPE treats pa-
 1175 rameters in mechanistic models with deep probability distributions, yet the two learning algorithms

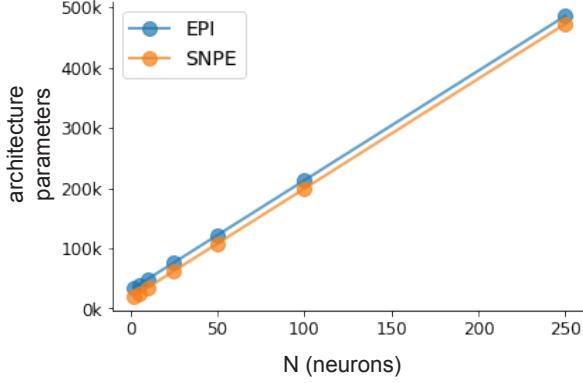


Figure 10: (RNN1): Number of parameters in deep probability distribution architectures of EPI (blue) and SNPE (orange) by RNN size (N).

are categorically different. SNPE uses a two-network architecture to approximate the posterior distribution of the model conditioned on observed data \mathbf{x}_0 . The amortizing network maps observations \mathbf{x}_i to the parameters of the deep probability distribution. The weights and biases of the parameter network are optimized by sequentially augmenting the training data with additional pairs $(\mathbf{z}_i, \mathbf{x}_i)$ based on the most recent posterior approximation. This sequential procedure is important to get training data \mathbf{z}_i to be closer to the true posterior, and \mathbf{x}_i to be closer to the observed data. For the deep probability distribution architecture, we chose a masked autoregressive flow with affine couplings (the default choice), three transforms, 50 hidden units, and a normalizing flow mapping to the support as in EPI. This architectural choice closely tracked the size of the architecture used by EPI (Fig. 10). As in SMC-ABC, we ran SNPE with $\mathbf{x}_0 = \mu$. All SNPE optimizations were run for a limit of 1.5 days on a Tesla V100 GPU, or until two consecutive rounds resulted in a validation log probability lower than the maximum observed for that random seed.

To clarify the difference in objectives of EPI and SNPE, we show their results on RNN models with different numbers of neurons N and random strength g . The parameters inferred by EPI consistently produces the same mean and variance of $\text{real}(\lambda_1)$ and λ_1^s , while those inferred by SNPE change according to the model definition (Fig. 11A). For $N = 2$ and $g = 0.01$, the SNPE posterior has greater concentration in eigenvalues around \mathbf{x}_0 than at $g = 0.1$, where the model has greater randomness (Fig. 11B top, orange). At both levels of g when $N = 2$, the posterior of SNPE has lower entropy than EPI at convergence (Fig. 11B top). However at $N = 10$, SNPE results in a predictive distribution of more widely dispersed eigenvalues (Fig. 11A bottom), and an inferred posterior with greater entropy than EPI (Fig. 11B bottom). We highlight these differences not

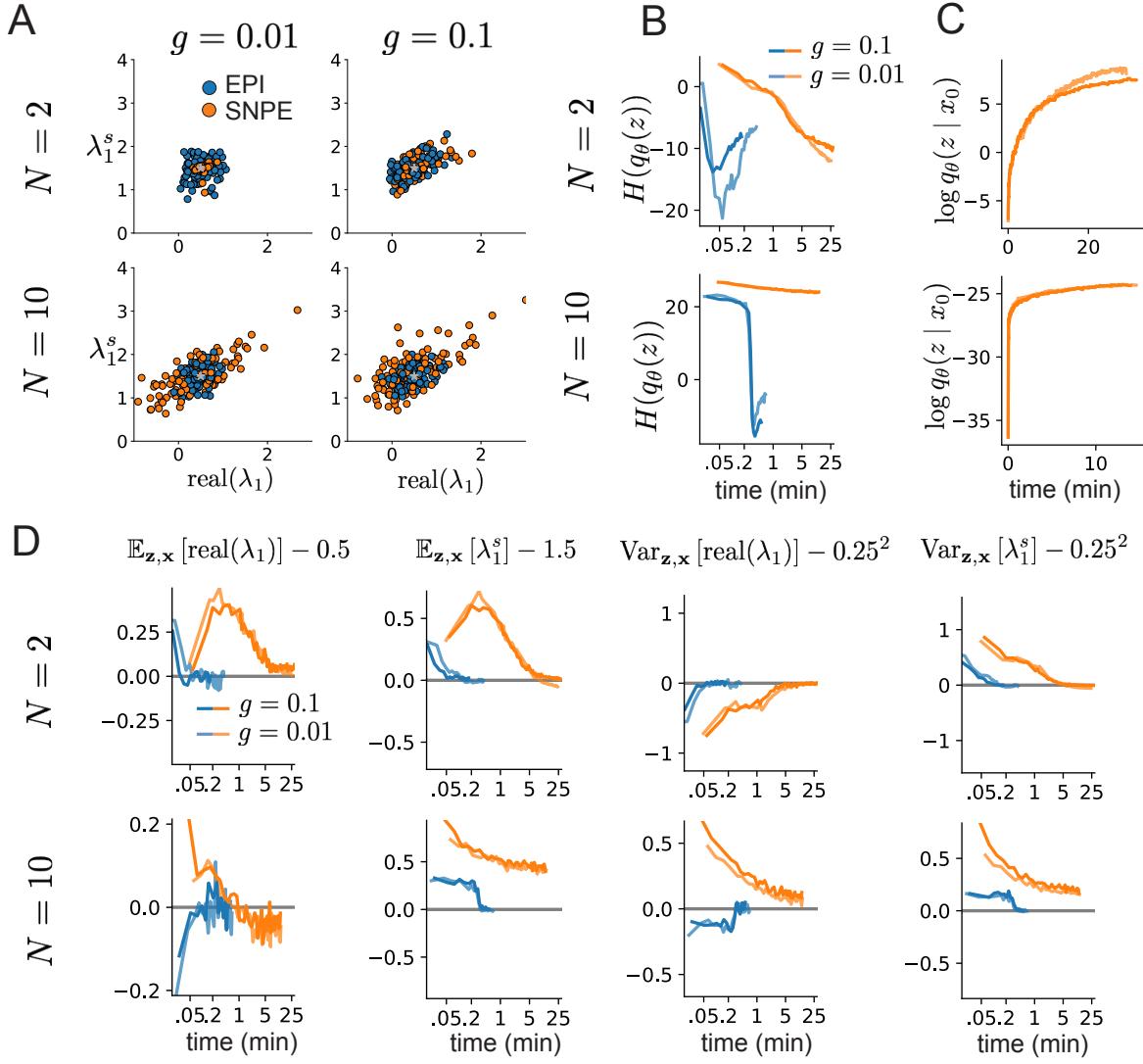


Figure 11: (RNN2): Model characteristics affect predictions of posteriors inferred by SNPE, while predictions of parameters inferred by EPI remain fixed. **A.** Predictive distribution of EPI (blue) and SNPE (orange) inferred connectivity of RNNs exhibiting stable amplification with $N = 2$ (top), $N = 10$ (bottom), $g = 0.01$ (left), and $g = 0.1$ (right). **B.** Entropy of parameter distribution approximations throughout optimization with $N = 2$ (top), $N = 10$ (bottom), $g = 0.1$ (dark shade), and $g = 0.01$ (light shade). **C.** Validation log probabilities throughout SNPE optimization. Same conventions as B. **D.** Adherence to EPI constraints. Same conventions as B.

1197 to focus on an insightful trend, but to emphasize that these methods optimize different objectives
1198 with different implications.

1199 Note that SNPE converges when it's validation log probability has saturated after several rounds
1200 of optimization (Fig. 11C), and that EPI converges after several epochs of its own optimization
1201 to enforce the emergent property constraints (Fig. 11D blue). Importantly, as SNPE optimizes
1202 its posterior approximation, the predictive means change, and at convergence may be different
1203 than \mathbf{x}_0 (Fig. 11D orange, left). It is sensible to assume that predictions of a well-approximated
1204 SNPE posterior should closely reflect the data on average (especially given a uniform prior and
1205 a low degree of stochasticity), however this is not a given. Furthermore, no aspect of the SNPE
1206 optimization controls the variance of the predictions (Fig. 11D orange, right).

1207 To compare the efficiency of these algorithms for inferring RNN connectivity distributions producing
1208 stable amplification, we develop a convergence criteria that can be used across methods. While EPI
1209 has its own hypothesis testing convergence criteria for the emergent property, it would not make
1210 sense to use this criteria on SNPE and SMC-ABC which do not constrain the means and variances
1211 of their predictions. Instead, we consider EPI and SNPE to have converged after completing its
1212 most recent optimization epoch (EPI) or round (SNPE) in which the distance

$$d(q_\theta(z)) = |\mathbb{E}_{\mathbf{z}, \mathbf{x}} [f(\mathbf{x}; \mathbf{z})] - \boldsymbol{\mu}|_2 \quad (65)$$

1213 is less than 0.5. We consider SMC-ABC to have converged once the population produces samples
1214 within the $\epsilon = 0.5$ ball ensuring stable amplification.

1215 When assessing the scalability of SNPE, it is important to check that alternative hyperparameter-
1216izations could not yield better performance. Key hyperparameters of the SNPE optimization are
1217 the number of simulations per round n_{round} , the number of atoms used in the atomic proposals of
1218 the SNPE-C algorithm [106], and the batch size n . To match EPI, we used a batch size of $n = 200$
1219 for $N \leq 25$, however we found $n = 1,000$ to be helpful for SNPE in higher dimensions. While
1220 $n_{\text{round}} = 1,000$ yielded SNPE convergence for $N \leq 25$, we found that a substantial increase to
1221 $n_{\text{round}} = 25,000$ yielded more consistent convergence at $N = 50$ (Fig. 12A). By increasing n_{round} ,
1222 we also necessarily increase the duration of each round. At $N = 100$, we tried two hyperparameter
1223 modifications. As suggested in [106], we increased n_{atom} by an order of magnitude to improve
1224 gradient quality, but this had little effect on the optimization (much overlap between same random
1225 seeds) (Fig. 12B). Finally, we increased n_{round} by an order of magnitude, which yielded convergence
1226 in one case, but no others. We found no way to improve the convergence rate of SNPE without
1227 making more aggressive hyperparameter choices requiring high numbers of simulations.

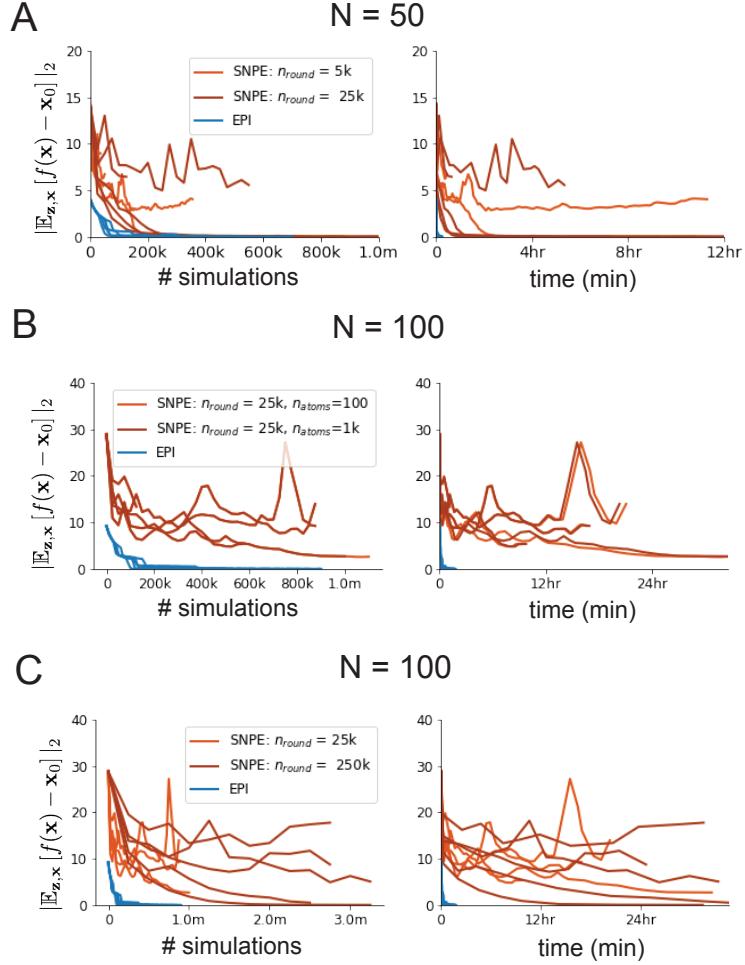


Figure 12: (RNN3): SNPE convergence was enabled by increasing n_{round} , not n_{atom} . **A.** Difference of mean predictions \mathbf{x}_0 throughout optimization at $N = 50$ with by simulation count (left) and wall time (right) of SNPE with $n_{\text{round}} = 5,000$ (light orange), SNPE with $n_{\text{round}} = 25,000$ (dark orange), and EPI (blue). Each line shows an individual random seed. **B.** Same conventions as A at $N = 100$ of SNPE with $n_{\text{atom}} = 100$ (light orange) and $n_{\text{atom}} = 1,000$ (dark orange). **C.** Same conventions as A at $N = 100$ of SNPE with $n_{\text{round}} = 25,000$ (light orange) and $n_{\text{round}} = 250,000$ (dark orange).

1228 In Figure 2C-D, we show samples from the random seed resulting in emergent property convergence
 1229 at greatest entropy (EPI), the random seed resulting in greatest validation log probability (SNPE),
 1230 and the result of all converged random seeds (SMC).

1231 **5.4 Primary visual cortex**

1232 **5.4.1 V1 model**

1233 In the stochastic stabilized supralinear network [78], population rate responses \mathbf{x} to input \mathbf{h} , recurrent input $W\mathbf{x}$ and slow noise ϵ are governed by
 1234

$$\tau \frac{d\mathbf{x}}{dt} = -\mathbf{x} + \phi(W\mathbf{x} + \mathbf{h} + \epsilon), \quad (66)$$

1235 where the noise is an Ornstein-Uhlenbeck process $\epsilon \sim OU(\tau_{\text{noise}}, \sigma)$

$$\tau_{\text{noise}} d\epsilon_\alpha = -\epsilon_\alpha dt + \sqrt{2\tau_{\text{noise}}} \tilde{\sigma}_\alpha dB \quad (67)$$

1236 with $\tau_{\text{noise}} = 5\text{ms} > \tau = 1\text{ms}$. The noisy process is parameterized as

$$\tilde{\sigma}_\alpha = \sigma_\alpha \sqrt{1 + \frac{\tau}{\tau_{\text{noise}}}}, \quad (68)$$

1237 so that σ parameterizes the variance of the noisy input in the absence of recurrent connectivity
 1238 ($W = \mathbf{0}$). As contrast $c \in [0, 1]$ increases, input to the E- and P-populations increases relative to
 1239 a baseline input $\mathbf{h} = \mathbf{h}_b + c\mathbf{h}_c$. Connectivity (W_{fit}) and input ($\mathbf{h}_{b,\text{fit}}$ and $\mathbf{h}_{c,\text{fit}}$) parameters were fit
 1240 using the deterministic V1 circuit model [56]

$$W_{\text{fit}} = \begin{bmatrix} W_{EE} & W_{EP} & W_{ES} & W_{EV} \\ W_{PE} & W_{PP} & W_{PS} & W_{PV} \\ W_{SE} & W_{SP} & W_{SS} & W_{SV} \\ W_{VE} & W_{VP} & W_{VS} & W_{VV} \end{bmatrix} = \begin{bmatrix} 2.18 & -1.19 & -.594 & -.229 \\ 1.66 & -.651 & -.680 & -.242 \\ .895 & -5.22 \times 10^{-3} & -1.51 \times 10^{-4} & -.761 \\ 3.34 & -2.31 & -.254 & -2.52 \times 10^{-4} \end{bmatrix}, \quad (69)$$

$$\mathbf{h}_{b,\text{fit}} = \begin{bmatrix} .416 \\ .429 \\ .491 \\ .486 \end{bmatrix}, \quad (70)$$

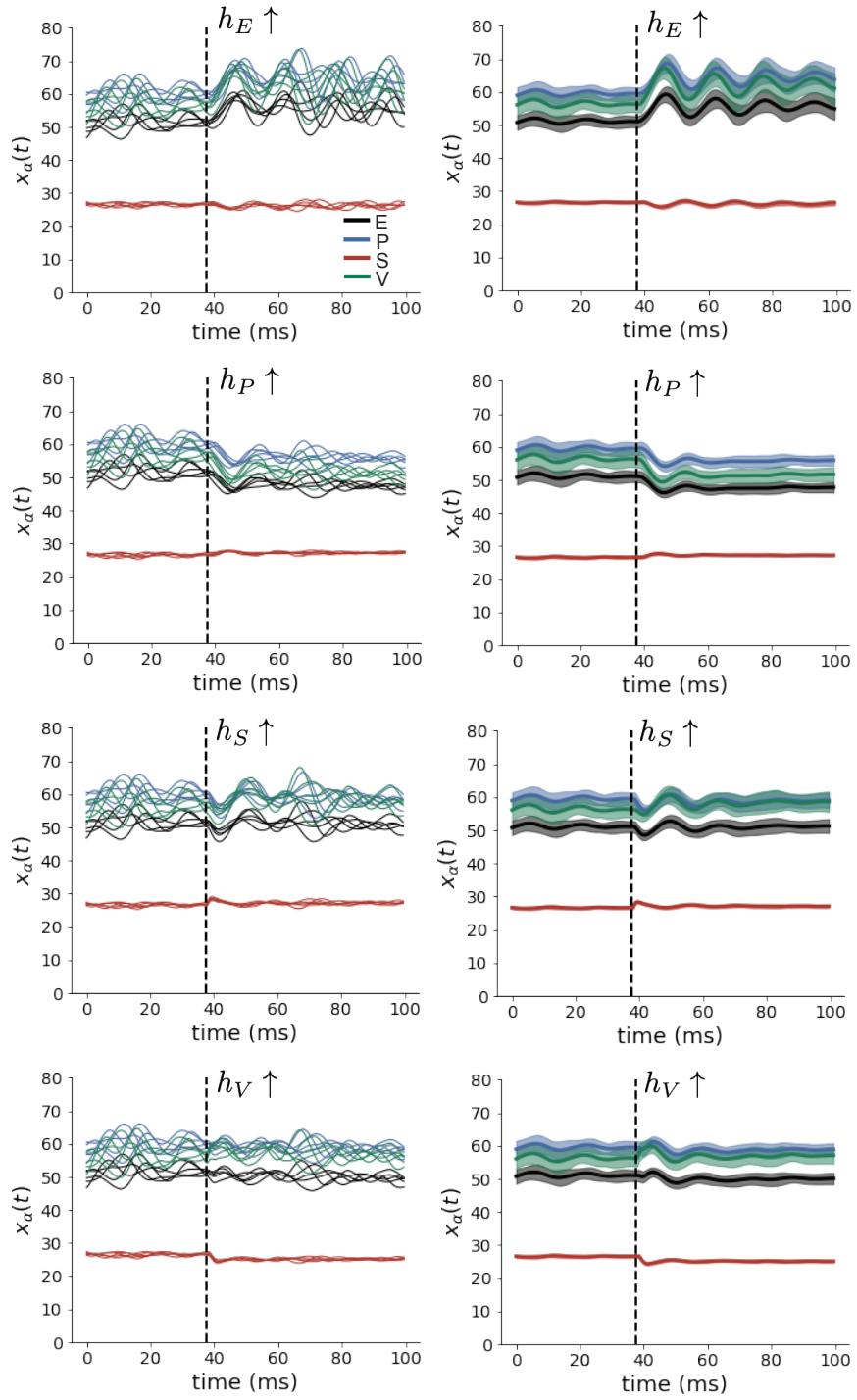


Figure 13: (V1 1) (Left) Simulations for small increases in neuron-type population input. Input magnitudes are chosen so that effect is salient (0.002 for E and P, but 0.02 for S and V). (Right) Average (solid) and standard deviation (shaded) of stochastic fluctuations of responses.

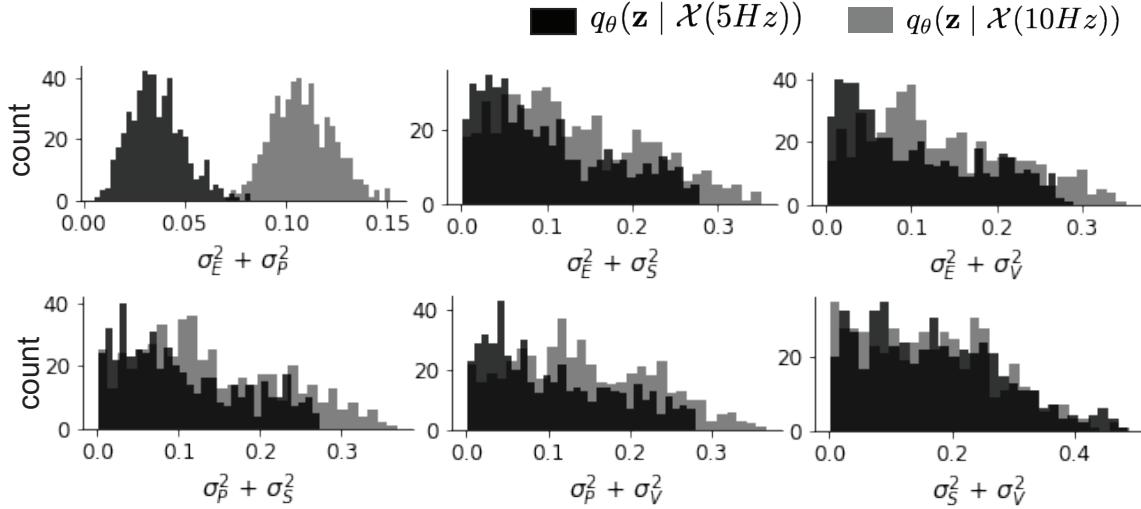


Figure 14: (V1 2) EPI predictive distributions of the sum of squares of each pair of noise parameters.

1241 and

$$\mathbf{h}_{c,\text{fit}} = \begin{bmatrix} .359 \\ .403 \\ 0 \\ 0 \end{bmatrix}. \quad (71)$$

1242 To obtain rates on a realistic scale (100-fold greater), we map these fitted parameters to an equiv-
 1243 alence class

$$W = \begin{bmatrix} W_{EE} & W_{EP} & W_{ES} & W_{EV} \\ W_{PE} & W_{PP} & W_{PS} & W_{PV} \\ W_{SE} & W_{SP} & W_{SS} & W_{SV} \\ W_{VE} & W_{VP} & W_{VS} & W_{VV} \end{bmatrix} = \begin{bmatrix} .218 & -.119 & -.0594 & -.0229 \\ .166 & -.0651 & -.068 & -.0242 \\ .0895 & -5.22 \times 10^{-4} & -1.51 \times 10^{-5} & -.0761 \\ .334 & -.231 & -.0254 & -2.52 \times 10^{-5} \end{bmatrix}, \quad (72)$$

$$\mathbf{h}_b = \begin{bmatrix} h_{b,E} \\ h_{b,P} \\ h_{b,S} \\ h_{b,V} \end{bmatrix} = \begin{bmatrix} 4.16 \\ 4.29 \\ 4.91 \\ 4.86 \end{bmatrix}, \quad (73)$$

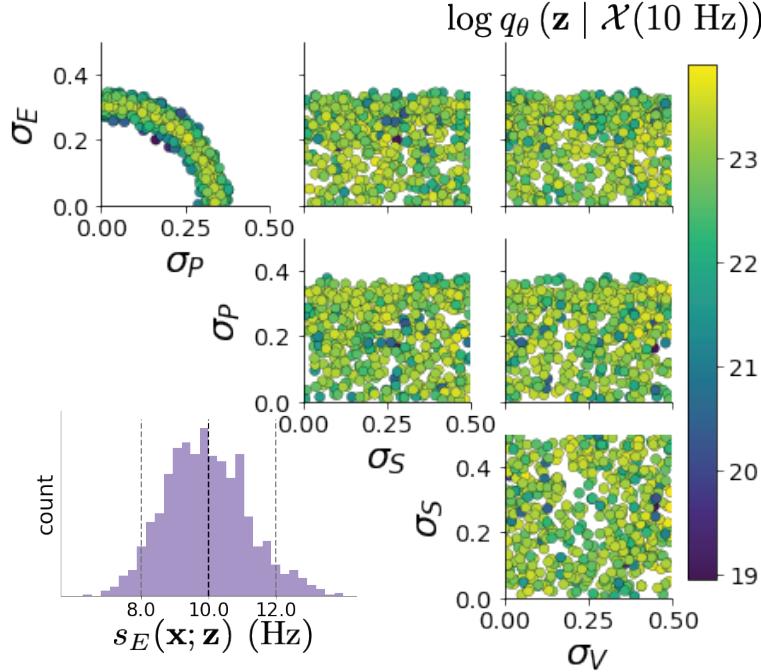


Figure 15: (V1 3) EPI inferred distribution for $\mathcal{X}(10 \text{ Hz})$.

1244 and

$$\mathbf{h}_c = \begin{bmatrix} h_{c,E} \\ h_{c,P} \\ h_{c,S} \\ h_{c,V} \end{bmatrix} = \begin{bmatrix} 3.59 \\ 4.03 \\ 0 \\ 0 \end{bmatrix}. \quad (74)$$

1245 Circuit responses are simulated using $T = 200$ time steps at $dt = 0.5\text{ms}$ from an initial condition
 1246 drawn from $\mathbf{x}(0) \sim U[10 \text{ Hz}, 25 \text{ Hz}]$. Standard deviation of the E-population $s_E(\mathbf{x}; \mathbf{z})$ is calculated
 1247 as the square root of the temporal variance from $t_{ss} = 75\text{ms}$ to $Tdt = 100\text{ms}$ averaged over 100
 1248 independent trials.

$$s_E(\mathbf{x}; \mathbf{z}) = \mathbb{E}_x \left[\sqrt{\mathbb{E}_{t > t_{ss}} [(x_E(t) - \mathbb{E}_{t > t_{ss}} [x_E(t)])^2]} \right] \quad (75)$$

1249 **5.4.2 EPI details for the V1 model**

1250 For EPI in Fig 3D-E, we used a real NVP architecture with three Real NVP coupling layers
 1251 and two-layer neural networks of 50 units per layer. The normalizing flow architecture mapped
 1252 $z_0 \sim \mathcal{N}(\mathbf{0}, I)$ to a support of $\mathbf{z} = [\sigma_E, \sigma_P, \sigma_S, \sigma_V] \in [0.0, 0.5]^4$. EPI optimization was run using three
 1253 different random seeds for architecture initialization $\boldsymbol{\theta}$ with an augmented Lagrangian coefficient of

1254 $c_0 = 10^{-1}$, a batch size $n = 100$, and $\beta = 2$. The distributions shown are those of the architectures
1255 converging with criteria $N_{\text{test}} = 100$ at greatest entropy across three random seeds.

1256 **5.4.3 Sensitivity analyses**

1257 In Fig. 3E, we visualize the modes of $q_{\theta}(\mathbf{z} \mid \mathcal{X})$ throughout the σ_E - σ_P marginal. Specifically, we
1258 calculated

$$\begin{aligned} \mathbf{z}^*(\sigma_{P,\text{fixed}}) &= \underset{\mathbf{z}}{\operatorname{argmax}} \log q_{\theta}(\mathbf{z} \mid \mathcal{X}) \\ \text{s.t. } \sigma_P &= \sigma_{P,\text{fixed}} \end{aligned} \quad (76)$$

1259 At each mode \mathbf{z}^* , we calculated the Hessian and visualized the sensitivity dimension in the direction
1260 of positive σ_E .

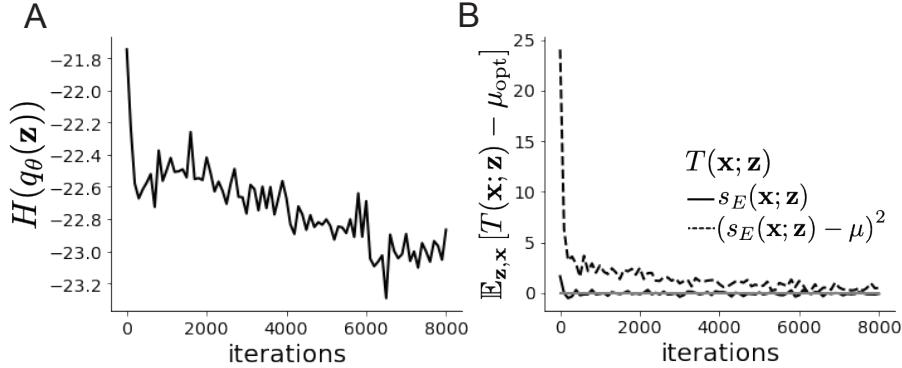


Figure 16: (V1 4) Optimization for V1

1261 **5.4.4 Primary visual cortex: challenges to analysis**

1262 TODO Agostina and I are putting this together now.

1263 **5.5 Superior colliculus**

1264 **5.5.1 SC model**

1265 The ability to switch between two separate tasks throughout randomly interleaved trials, or “rapid
1266 task switching,” has been studied in rats, and midbrain superior colliculus (SC) has been show to
1267 play an important in this computation [79]. Neural recordings in SC exhibited two populations of

1268 neurons that simultaneously represented both task context (Pro or Anti) and motor response (con-
 1269 tralateral or ipsilateral to the recorded side), which led to the distinction of two functional classes:
 1270 the Pro/Contra and Anti/Ipsi neurons [57]. Given this evidence, Duan et al. proposed a model
 1271 with four functionally-defined neuron-type populations: two in each hemisphere corresponding to
 1272 the Pro/Contra and Anti/Ipsi populations. We study how the connectivity of this neural circuit
 1273 governs rapid task switching ability.

1274 The four populations of this model are denoted as left Pro (LP), left Anti (LA), right Pro (RP)
 1275 and right Anti (RA). Each unit has an activity (x_α) and internal variable (u_α) related by

$$x_\alpha = \phi(u_\alpha) = \left(\frac{1}{2} \tanh\left(\frac{u_\alpha - a}{b}\right) + \frac{1}{2} \right), \quad (77)$$

1276 where $\alpha \in \{LP, LA, RA, RP\}$, $a = 0.05$ and $b = 0.5$ control the position and shape of the nonlin-
 1277 earity. We order the neural populations of x and u in the following manner

$$\mathbf{x} = \begin{bmatrix} x_{LP} \\ x_{LA} \\ x_{RP} \\ x_{RA} \end{bmatrix} \quad \mathbf{u} = \begin{bmatrix} u_{LP} \\ u_{LA} \\ u_{RP} \\ u_{RA} \end{bmatrix}, \quad (78)$$

1278 which evolve according to

$$\tau \frac{d\mathbf{u}}{dt} = -\mathbf{u} + W\mathbf{x} + \mathbf{h} + d\mathbf{B}. \quad (79)$$

1279 with time constant $\tau = 0.09s$, step size 24ms and Gaussian noise $d\mathbf{B}$ of variance 0.2^2 . These
 1280 hyperparameter values are motivated by modeling choices and results from [57].

1281 The weight matrix has 4 parameters for self sW , vertical vW , horizontal hW , and diagonal dW
 1282 connections:

$$W = \begin{bmatrix} sW & vW & hW & dW \\ vW & sW & dW & hW \\ hW & dW & sW & vW \\ dW & hW & vW & sW \end{bmatrix}. \quad (80)$$

1283 We study the role of parameters $\mathbf{z} = [sW, vW, hW, dW]^\top$ in rapid task switching.

1284 The circuit receives four different inputs throughout each trial, which has a total length of 1.8s.

$$\mathbf{h} = \mathbf{h}_{\text{constant}} + \mathbf{h}_{\text{P,bias}} + \mathbf{h}_{\text{rule}} + \mathbf{h}_{\text{choice-period}} + \mathbf{h}_{\text{light}}. \quad (81)$$

1285 There is a constant input to every population,

$$\mathbf{h}_{\text{constant}} = I_{\text{constant}}[1, 1, 1, 1]^\top, \quad (82)$$

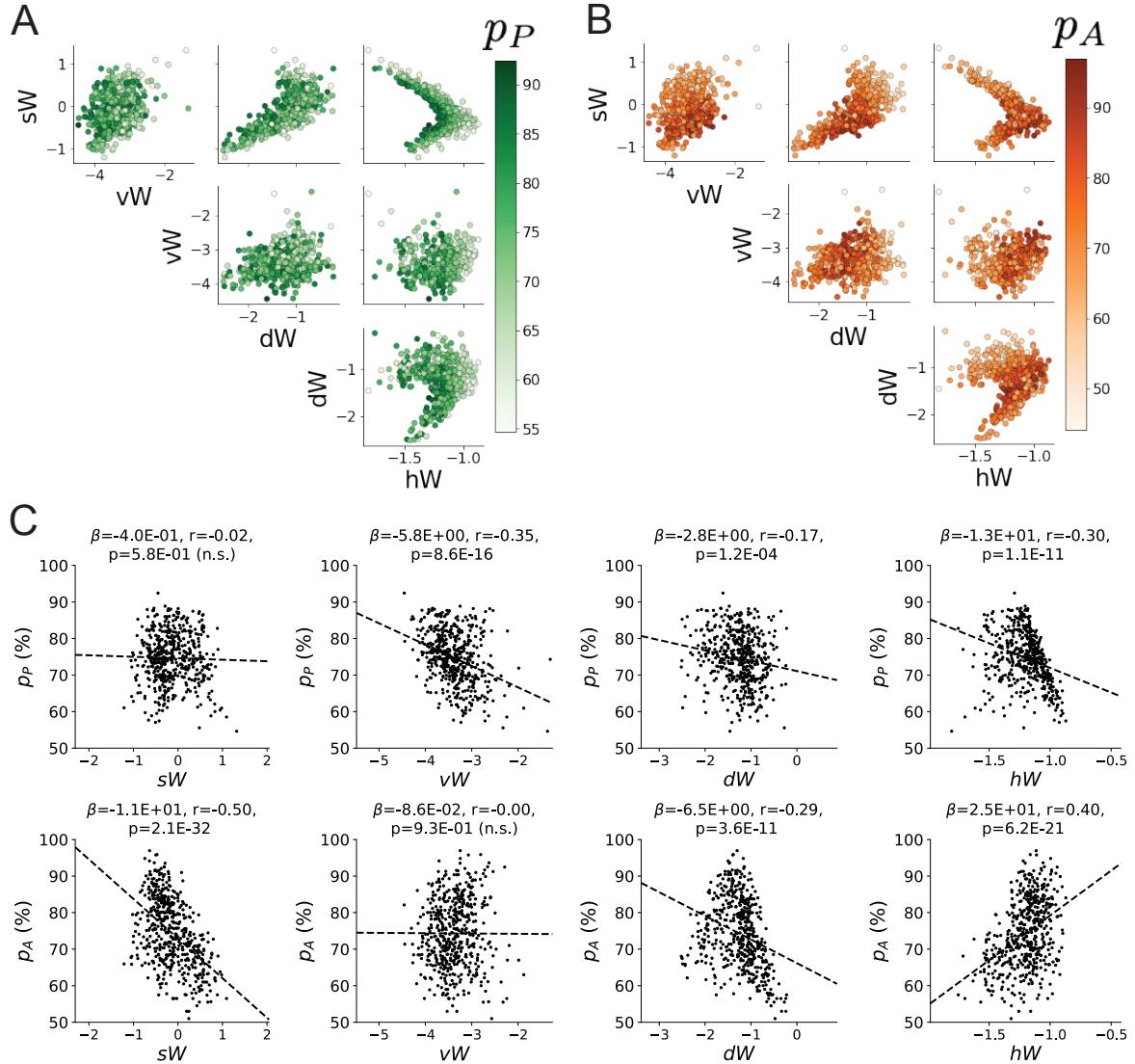


Figure 17: (SC1): **A.** Same pairplot as Fig. 4C colored by Pro task accuracy. **B.** Same as A colored by Anti task accuracy. **C.** Connectivity parameters of EPI distributions versus task accuracies. β is slope coefficient of linear regression, r is correlation, and p is the two-tailed p-value.

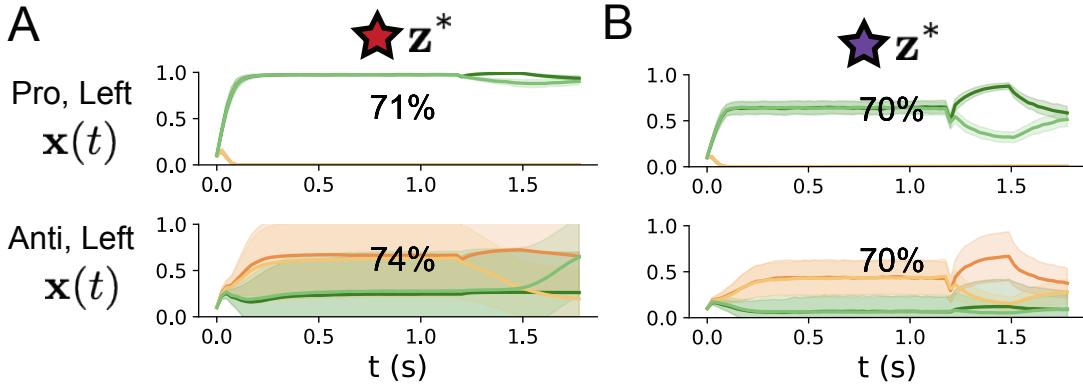


Figure 18: (SC2): **A.** Simulations in network regime 1 ($hW_{\text{fixed}} = -1.5$). **B.** Simulations in network regime 2 ($hW_{\text{fixed}} = -1.5$) .

1286 a bias to the Pro populations

$$\mathbf{h}_{P,\text{bias}} = I_{P,\text{bias}}[1, 0, 1, 0]^\top, \quad (83)$$

1287 rule-based input depending on the condition

$$\mathbf{h}_{P,\text{rule}}(t) = \begin{cases} I_{P,\text{rule}}[1, 0, 1, 0]^\top, & \text{if } t \leq 1.2s \\ 0, & \text{otherwise} \end{cases} \quad (84)$$

1288

$$\mathbf{h}_{A,\text{rule}}(t) = \begin{cases} I_{A,\text{rule}}[0, 1, 0, 1]^\top, & \text{if } t \leq 1.2s \\ 0, & \text{otherwise} \end{cases}, \quad (85)$$

1289 a choice-period input

$$\mathbf{h}_{\text{choice}}(t) = \begin{cases} I_{\text{choice}}[1, 1, 1, 1]^\top, & \text{if } t > 1.2s \\ 0, & \text{otherwise} \end{cases}, \quad (86)$$

1290 and an input to the right or left-side depending on where the light stimulus is delivered

$$\mathbf{h}_{\text{light}}(t) = \begin{cases} I_{\text{light}}[1, 1, 0, 0]^\top, & \text{if } 1.2s < t < 1.5s \text{ and Left} \\ I_{\text{light}}[0, 0, 1, 1]^\top, & \text{if } 1.2s < t < 1.5s \text{ and Right} \\ 0, & \text{otherwise} \end{cases}. \quad (87)$$

1291 The input parameterization was fixed to $I_{\text{constant}} = 0.75$, $I_{P,\text{bias}} = 0.5$, $I_{P,\text{rule}} = 0.6$, $I_{A,\text{rule}} = 0.6$,

1292 $I_{\text{choice}} = 0.25$, and $I_{\text{light}} = 0.5$.

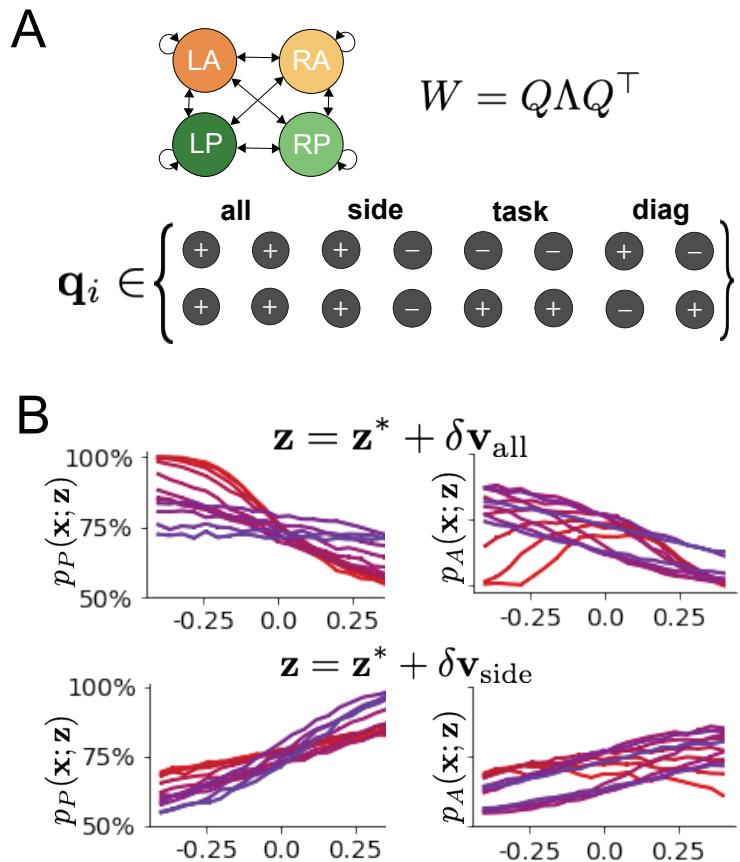


Figure 19: (SC3): **A.** Invariant eigenvectors of connectivity matrix W . **B.** Accuracies for connectivity perturbations for increasing λ_{all} and λ_{side} (rest shown in Fig. 4D).

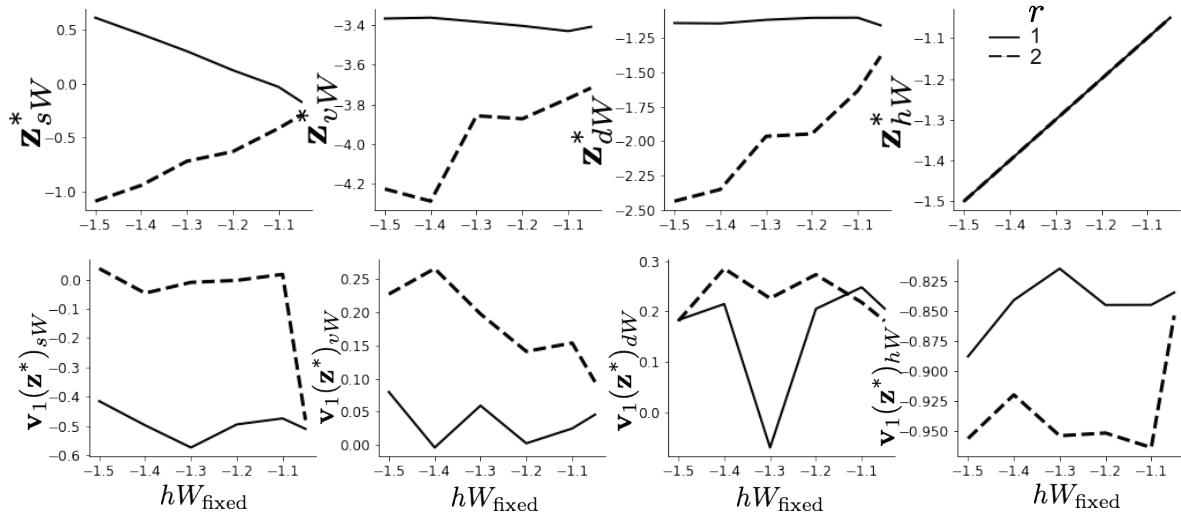


Figure 20: (SC4): **A.** The individual parameters of each mode throughout the two regimes. **B.** The individual sensitivities of parameters of each mode throughout the two regimes.

1293 5.5.2 Task accuracy calculation

1294 The accuracies of each task p_P and p_A are calculated as

$$p_P(\mathbf{x}; \mathbf{z}) = \mathbb{E}_{\mathbf{x}} [\Theta[x_{LP}(t = 1.8s) - x_{RP}(t = 1.8s)]] \quad (88)$$

1295 and

$$p_A(\mathbf{x}; \mathbf{z}) = \mathbb{E}_{\mathbf{x}} [\Theta[x_{RP}(t = 1.8s) - x_{LP}(t = 1.8s)]] \quad (89)$$

1296 given that the stimulus is on the left side, where Θ is the Heaviside step function, and the accuracy
1297 is averaged over 200 independent trials. The Heaviside step function is approximated as

$$\Theta(\mathbf{x}) = \text{sigmoid}(\beta \mathbf{x}), \quad (90)$$

1298 where $\beta = 100$.

1299 5.5.3 EPI details for the SC model

1300 Writing the EPI distribution as a maximum entropy distribution, $T(\mathbf{x}, \mathbf{z})$ is comprised of both these
1301 first and second moments of the accuracy in each task (as in Equations 27 and 28)

$$T(\mathbf{x}; \mathbf{z}) = \begin{bmatrix} p_P(\mathbf{x}; \mathbf{z}) \\ p_A(\mathbf{x}; \mathbf{z}) \\ (p_P(\mathbf{x}; \mathbf{z}) - 75\%)^2 \\ (p_A(\mathbf{x}; \mathbf{z}) - 75\%)^2 \end{bmatrix}, \quad (91)$$

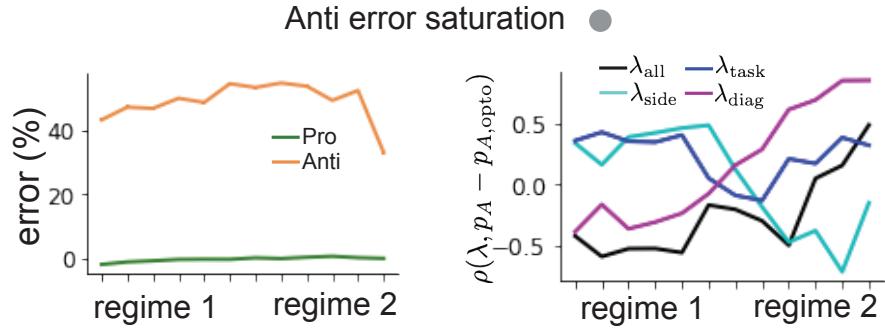


Figure 21: (SC5): (Left) Mean and standard error of Pro and Anti error from regime 1 to regime 2 at $\gamma = 0.85$. (Right) Correlations of connectivity eigenvalues with Anti error from regime 1 to regime 2 at $\gamma = 0.85$.

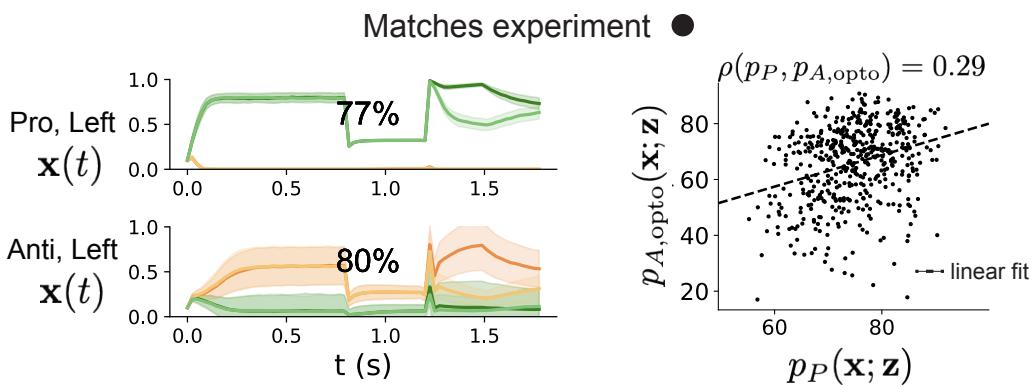


Figure 22: (SC6): (Left) Responses of the SC model at the mode of the EPI distribution to delay period inactivation at $\gamma = 0.675$. (Right) Anti accuracy following delay period inactivation at $\gamma = 0.675$ versus accuracy in the Pro task across connectivities in the EPI distribution.

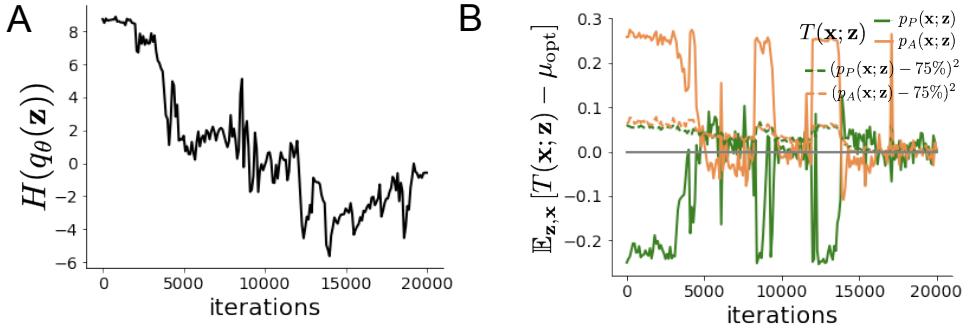


Figure 23: (SC7): **A.** Entropy throughout optimization. **B.** The emergent property statistic means and variances converge to their constraints at 20,000 iterations following the tenth augmented Lagrangian epoch.

$$\boldsymbol{\mu}_{\text{opt}} = \begin{bmatrix} 75\% \\ 75\% \\ 7.5\%^2 \\ 7.5\%^2 \end{bmatrix}. \quad (92)$$

1302 Throughout optimization, the augmented Lagrangian parameters η and c , were updated after each
 1303 epoch of 2,000 iterations (see Section 5.1.3). The optimization converged after ten epochs (Fig.
 1304 22).

1305 For EPI in Fig. 4C, we used a real NVP architecture with three coupling layers of affine transfor-
 1306 mations parameterized by two-layer neural networks of 50 units per layer. The initial distribution
 1307 was a standard isotropic gaussian $\mathbf{z}_0 \sim \mathcal{N}(\mathbf{0}, I)$ mapped to a support of $\mathbf{z}_i \in [-5, 5]$. We used an
 1308 augmented Lagrangian coefficient of $c_0 = 10^2$, a batch size $n = 100$, and $\beta = 2$. The distribution
 1309 was the greatest EPI distribution to converge across 5 random seeds with criteria $N_{\text{test}} = 25$.

1310 5.5.4 Regime identification with EPI

1311 We sought two sets of parameters from $q_\theta(\mathbf{z} | \mathcal{X})$ that were representative of each regime, so that we
 1312 could assess their implications on computation. For fixed values of hW , we hypothesized that there
 1313 are two modes: one in each regime of greater and lesser sW . To begin, we found one mode for each
 1314 regime at $hW_{\text{fixed}} = -1.5$ using 200 steps of gradient ascent of the deep probability distribution
 1315 $q_\theta(\mathbf{z} | \mathcal{X})$. In regime 1, the initialization had positive sW , and the initialization had negative sW
 1316 in regime 2, which led to disparate modes (Fig. 20 top). These modes were then used as the
 1317 initialization to find the next mode at $hW_{\text{fixed}} = -1.4$ and so on. 200 steps of gradient ascent
 1318 were always taken, and learning rates of 2.5×10^{-4} and 5×10^{-4} were used for regimes 1 and 2,

1319 respectively. Each of these modes is denoted $\mathbf{z}^*(hW_{\text{fixed}}, r)$ for regime $r \in \{1, 2\}$.

1320 For the analyses in Figure 5C and Figure 21, we obtained parameters for each step along the
1321 continuum between regimes 1 and 2 by sampling from the EPI distribution. Each sample was
1322 assigned to the closest mode $\mathbf{z}^*(hW_{\text{fixed}}, r)$. Sampling continued until 500 samples were assigned to
1323 each mode, which took 7.36 seconds. To obtain this many samples for each mode with brute force
1324 sampling over the chosen prior, this would take 4.20 days.

1325 5.5.5 Sensitivity analysis

1326 At each mode, we measure the sensitivity dimension (that of most negative eigenvalue in the Hessian
1327 of the EPI distribution) $\mathbf{v}_1(\mathbf{z}^*)$. To resolve sign degeneracy in eigenvectors, we chose $\mathbf{v}_1(\mathbf{z}^*)$ to have
1328 negative element in hW . This tells us what parameter combination rapid task switching is most
1329 sensitive to at this parameter choice in the regime. We see that while the modes of each regime
1330 gradually converge to similar connectivities at $hW_{\text{fixed}} = -1.05$ (Fig. 20 top), the sensitivity
1331 dimensions remain categorically different throughout the two regimes (Fig. 20 bottom). Only at
1332 $hW_{\text{fixed}} = -1.05$ is there a flip in sensitivity from regime 2 to regime 1 (in $\mathbf{v}_1(\mathbf{z}^*)_{sW}$ and $\mathbf{v}_1(\mathbf{z}^*)_{hW}$).
1333 There is thus some ambiguity regarding the “regime” of $\mathbf{z}^*(-1.05, 2)$, since the mode is derived
1334 from an initialization in regime 2, but has sensitivity like regime 1. We can consider this as an
1335 intermediate transitional region of parameter space between the two regimes. To emphasize this,
1336 $\mathbf{z}^*(-1.05, 1)$ and $\mathbf{z}^*(-1.05, 2)$ have the same color.

1337 5.5.6 Connectivity eigendecomposition and processing modes

1338 To understand the connectivity mechanisms governing task accuracy, we took the eigendecomposi-
1339 tion of the symmetric connectivity matrices $W = Q\Lambda Q^{-1}$, which results in the same basis vectors
1340 \mathbf{q}_i for all W parameterized by \mathbf{z} (Fig. 19A). These basis vectors have intuitive roles in processing for
1341 this task, and are accordingly named the *all* eigenmode - all neurons co-fluctuate, *side* eigenmode
1342 - one side dominates the other, *task* eigenmode - the Pro or Anti populations dominate the other,
1343 and *diag* mode - Pro- and Anti-populations of opposite hemispheres dominate the opposite pair.
1344 Due to the parametric structure of the connectivity matrix, the parameters \mathbf{z} are a linear function
1345 of the eigenvalues $\boldsymbol{\lambda} = [\lambda_{\text{all}}, \lambda_{\text{side}}, \lambda_{\text{task}}, \lambda_{\text{diag}}]^\top$ associated with these eigenmodes.

$$\mathbf{z} = A\boldsymbol{\lambda} \quad (93)$$

$$A = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \end{bmatrix}. \quad (94)$$

1347 We are interested in the effect of raising or lowering the amplification of each eigenmode in the
 1348 connectivity matrix. To test this, we calculate the unit vector of changes in the connectivity \mathbf{z} that
 1349 result from a change in the associated eigenvalues

$$\mathbf{v}_a = \frac{\frac{\partial \mathbf{z}}{\partial \lambda_a}}{\left\| \frac{\partial \mathbf{z}}{\partial \lambda_a} \right\|_2}, \quad (95)$$

1350 where

$$\frac{\partial \mathbf{z}}{\partial \lambda_a} = A \mathbf{e}_a, \quad (96)$$

1351 and e.g. $\mathbf{e}_{\text{all}} = [1, 0, 0, 0]^\top$. So \mathbf{v}_a is the normalized column of A corresponding to eigenmode a .
 1352 While perturbations in the sensitivity dimension $\mathbf{v}_1(\mathbf{z}^*)$ adapt with the mode \mathbf{z}^* chosen, perturba-
 1353 tions in \mathbf{v}_a for $a \in \{\text{all, side, text, diag}\}$ are invariant to \mathbf{z} (Equation 96).

1354 To understand the connectivity mechanisms that distinguish these two regimes, we perturb connec-
 1355 tivity at each mode in dimensions that have well defined roles in processing for the Pro and Anti
 1356 tasks. A convenient property of this connectivity parameterization is that there are \mathbf{z} -invariant
 1357 eigenmodes of connectivity, whose eigenvalues (or degree of amplification) change with \mathbf{z} . These
 1358 eigenmodes have intuitive roles in processing in each task, and are accordingly named the *all*,
 1359 *side*, *task*, and *diag* eigenmodes (see Section 5.5). Furthermore, the parameter dimension \mathbf{v}_a
 1360 ($a \in \{\text{all, side, task, and diag}\}$) that increases the eigenvalue of connectivity λ_a is \mathbf{z} -invariant (un-
 1361 like the sensitivity dimension $\mathbf{v}_1(\mathbf{z})$) and $\mathbf{v}_a \perp \mathbf{v}_{b \neq a}$. Thus, by changing the degree of amplification
 1362 of each processing mode by perturbing \mathbf{z} along \mathbf{v}_a , we can elicit the differentiating properties of
 1363 the two regimes.

1364 5.5.7 Optogenetic silencing modeling

1365 We tested whether the inferred SC model connectivities could reproduce experimental effects of
 1366 optogenetic inactivation in rats [79]. During periods of simulated optogenetic inactivation, activity
 1367 was decreased proportional to the optogenetic strength γ

$$x_\alpha = (1 - \gamma)\phi(u_\alpha). \quad (97)$$

1368 Delay period inactivation was from $0.8 < t < 1.2$.