Emergent perceptual biases from state-space geometry in spiking recurrent neural networks trained to discriminate time intervals

Luis Serrano-Fernandez[1,2], Manuel Beiran[3] and Nestor Parga[1,2]

1: Departamento de Física Teórica, Universidad Autónoma de Madrid, Cantoblanco 28049, Madrid, Spain;

2: Centro de Investigación Avanzada en Física Fundamental, Universidad Autónoma de Madrid, Cantoblanco 28049, Madrid, Spain;

3: Center for Theoretical Neuroscience, Zuckerman Institute, Columbia University, New York, NY, USA.

Abstract

The contraction bias has been extensively studied in psychophysics, but only recently has its origin begun to be investigated with neural activity recordings. Leveraging the fact that it is currently possible to train recurrent networks of spiking neurons to perform cognitive tasks, we investigated the causes of the contraction bias and how behavior relates to population firing activity in networks trained to discriminate temporal intervals. A novel normative model containing potential sources of the bias guided us in investigating how it emerged in the networks. Various geometric features of state-space trajectories encoded compressed representations of the durations of the first interval, which were modulated by sensory history. Furthermore, these compressions conveyed a Bayesian estimate of the durations, thus relating activity and behavior. We conjecture that this occurs in areas of the brain where information about the current and preceding stimuli converges, and that it holds generally in delayed comparison tasks.

INTRODUCTION

Perception is not based purely on the current stimulus but is affected by other factors such as recent stimulation, prior knowledge and context, giving rise to perceptual biases (Bausenhart et al. 2014; Shi and Burr 2016). Examples of these biases appear in magnitude estimation (Shi et al., 2013; Jazayeri et al., 2010; Petzschner et al., 2015; Thurley, 2016) and delayed comparison tasks (Hollingworth, 1910; Ashourian et al., 2011; Dyjas et al., 2012, 2014), in which a stimulus parameter is perceived as shifted toward the mean of its distribution. These phenomena, known respectively as the central tendency and the contraction bias, have been extensively investigated in psychophysics (Hollingworth, 1910, Jazayeri et al., 2010; Ashourian et al., 2011; Dyjas et al., 2012, 2014; Shi et al., 2013; Petzschner et al., 2015; Thurley, 2016) and, more recently and to a lesser extent, with neural activity recordings in temporal reproduction tasks (Jazayeri and Shadlen, 2015; Remington et al., 2018; Egger et al., 2019; Sohn et al., 2019; Meirhaeghe et al., 2021; Henke et al., 2021) and delayed comparison tasks (Akrami et al., 2018; Sarno et al., 2022).

The contraction bias is affected by the stimulation history (Raviv et al. 2012, Akrami et al., 2018). A remarkable study found that in delayed comparison tasks the activity of posterior parietal cortex (PPC) neurons carries information about past stimuli, but not about the current stimulus (Akrami et al., 2018). Furthermore, that activity was causally related to the contraction bias. In addition to recent sensory history, neural activity in PPC also reflected longer-term sensory statistics, in the form of the mean of the stimulus distribution.

Within a Bayesian framework, prior experience is integrated with the noisy observation of the current stimulus, generating the contraction bias in delayed comparison tasks (Preuschhof et al., 2010; Ashourian et al., 2011; Raviv et al., 2012; Sarno et al., 2022). This integration would result in a sensory representation biased to the most reliable or likely stimuli (Preuschhof et al., 2010; Ashourian et al., 2011; Papadimitriou et al., 2015). However, despite their relevance, no study has addressed the fundamental questions of how the integration of those relevant factors takes place in neural circuits and what features of neural population activity relate to behavior. Moreover, it is not known how the neural information about the preceding stimuli relates to Bayesian computations.

New computational techniques enable recurrent neural networks to be trained on tasks similar to those used in experiments (Mante et al., 2013; Carnevale et al., 2015; Song et al., 2016; Song et al. 2017; Chaisangmongkon et al., 2017, Wang et al., 2018; Russo et al., 2018; Yang et al., 2019; Dubreuil et al., 2022). This provides us with another valuable tool with which to investigate the network mechanisms responsible for perceptual biases. This approach enables to formulate hypotheses, draw predictions and devise analysis methods that could help to guide experiments. In addition, trained networks allow full access to the

state-space dynamics and geometry of the neural trajectories and thus to the study of the relationship between population activity and behavior (Egger et al., 2019; Sohn et al., 2019).

In this work, to further the understanding of the causes of the contraction bias and its connection with neural population activity, we trained spiking recurrent neural networks (sRNNs) with the full-FORCE algorithm (DePasquale et al., 2016; Abbott et al., 2016; see also DePasquale et al., 2018) to compare the durations of two temporal intervals (d1 and d2), spaced by a delay interval. The trained networks did exhibit the bias. The investigation of the networks' behavior allowed us to identify potential sources of the contraction bias: short- and long-term sensory history and the integration of noisy observations and prior knowledge. The analysis of the neural firing activity indicated that these factors were present; then, an observer with access to the population activity would have to consider all these factors. This led us to propose a novel normative model that could be used to guide the analysis of the data and help identify the network computations responsible for the emergence of the contraction bias.

The activity of the neural population described trajectories in a low-dimensional state space. The geometry of the trajectories during the task encoded the durations of the first interval but did so by compressing them with respect to their true values. These compressions were hypothesized to convey a Bayesian estimate of the interval, thus relating population activity and behavior. Indeed, the Bayesian estimator showed similar compressions as those of the state-space trajectories, thereby supporting a Bayesian explanation of the contraction bias.

Our work leads to testable predictions about the causes of the contraction bias and how they are interrelated. It predicts a relationship between geometric properties of state-space trajectories and behavior. Moreover the normative model predicts that the sensory history contributes not only to the mean of the observations of the first interval but also modulates their variance; this modulation has behavioral consequences, since an increase in observation variance implies a larger contraction bias and a larger compression in the geometric properties of the trajectories. These findings are a consequence of the recurrence of the network and the existence of a delay period between the two presented stimuli. More generally, then, one might expect that the contraction bias in other discrimination tasks can be explained in a similar way.

RESULTS

Network training and numerical simulations

We trained sRNNs to decide which of two time intervals, presented sequentially and separated by a delay interval, was longer (Fig. 1a). Networks were trained with a set of duration pairs (d_1, d_2) (Fig. 1b), selected randomly and independently in each trial, using the full-FORCE algorithm for sRNNs (DePasquale et al., 2016; DePasquale et al., 2018; Abbott et al., 2016; Supplementary Fig. S1-1). Trials were presented continuously, without altering the activity at the end of each trial. Once a network was trained, a large number of test trials were obtained for further analyses of task performance and population activity (Methods). We considered three different trial blocks: a block in which only diagonal classes (classes labeled from 1 to 10, Fig. 1b) were tested, a block where only horizontal classes (d_2=370 ms) were tested and a block where all classes were tested (Methods). These will be referred to as the diagonal, horizontal and full blocks (Fig. 1b). Each block defined a context with a distinct prior distribution of d_1 (Fig. 1c); note that some classes appeared in more than one context.

In the main numerical experiment we trained a total of 35 networks (obtained from random initializations of the network parameters), with a fixed delay interval of 1500 ms. Unless otherwise specified, the analyses were performed on the set of 35 networks. In a separate numerical experiment, to examine the effect of the delay period duration on the contraction bias, we trained 15 sRNNs by selecting its value on each trial randomly from a set of four possible values (Methods). In another experiment, we trained seven networks resetting the population activity at the end of every trial.

The trained networks exhibit the contraction bias

The psychometric curves were computed using test trials from the horizontal classes in the full block. Figs. 1d (left) and 1d (right) show this curve for their mean over all trained networks and for example network 1, respectively, and Supplementary Fig. S2-1a shows this curve for example network 2. Psychometric curves were heterogeneous across the 35 trained networks (Supplementary Fig. S1-2). Next we analyzed the discrimination performance using test trials from the diagonal block. Classes in this block had a fixed duration difference $\Delta d=|d_1-d_2|$ (60 ms; Fig. 1b). However, contrary to what would be expected for classes with the same duration differences, accuracy was not uniform (Figs. 1e (right) and 1e (left) show the accuracy of example network 1 and the mean over the networks, respectively, and Supplementary Fig. S2-1b shows the accuracy of example network 2). The contraction bias could explain this disparity. This bias is usually described as a shift of the

encoding of duration of the first stimulus towards the mean of its distribution (i.e., $<d_1>=370$ ms, Fig. 1f). This means that a short duration (i.e., $d_1=250$ ms) is encoded as longer, while a long duration is encoded as shorter. Therefore, correct discriminations for classes with $d_1<d_2$ are increasingly impaired by the bias as classes are visited from right to left (upper diagonal, Fig. 1f), while classes with $d_1>d_2$ are increasingly facilitated by the bias (lower diagonal, Fig. 1f). To make this bias structure explicit, we labeled the classes such that those with favorable bias effects are near the extremes, while classes with disfavorable effects are around the center (Fig. 1f). With this arrangement of classes the decision accuracy was higher for classes near the two extremes (Fig. 1e). Note that since the networks were trained with sequences of temporally uncorrelated d_1 values, this result implies that correlations in the training stimuli are not crucial for generating the bias.

The contraction bias of a network was quantified as the root mean square deviation of the performance relative to the middle classes, a quantity denoted as **B** (see Methods). The distribution of this bias metric spanned approximately one order of magnitude across the networks (Fig. 1g).

Behavior and possible causes of the contraction bias

One way in which behavior in the current trial can be altered is through the persistence of the neural representation of the stimuli applied in previous trials. This would produce a contraction driven by the mean of the stimulus parameter or by the most recently presented stimuli (Raviv et al., 2012; Akrami et al., 2018). On the other hand, it has been proposed that the bias can be generated by a Bayesian mechanism that occurs if, at the time of the comparison of the two stimuli, the uncertainty in the observation of the first stimulus is greater than the uncertainty in the observation of the second one -which creates a tendency to rely more on the *a priori* knowledge on the first stimulus (Ashourian and Loewenstein, 2011). In support of this idea, behavioral data from monkeys discriminating tactile frequencies were well fit with a Bayesian model (Sarno et al., 2022). In addition, the stimulus statistics could be stored in the network synapses during the learning stage, then affecting behavior with a contraction of the intervals toward the learned mean (Preuschhof et al., 2010). We describe below the results of numerical experiments performed to examine these three possible causes of contraction bias.

To check that the previous class can modulate behavior we sorted a large number of test trials by the preceding class and calculated the choice accuracy for each case (Methods). The analysis confirmed that the preceding class modulates performance (see Fig. 2a for the example network 1 and Supplementary Fig. S2-1c for a second example network). In addition, we observed the presence of a choice bias, which depended on whether the preceding choice had been rewarded.

If the contraction bias were due solely to the action of preceding stimulation then it should remain constant or weaken as the duration of the delay period is increased (Papadimitriou et al., 2015). However, if a Bayesian mechanism were operating in our networks, we would expect that as the duration of the delay epoch increases, the noise of the representation of the first interval would also increase, resulting in a larger contraction bias (Papadimitriou et al., 2015). To compare the bias for different durations of the delay period, we trained 15 networks to perform time interval discrimination for four durations, randomly interleaving these values during training (Methods). Then, blocks of test trials with fixed delay period duration were collected and analyzed. The bias **B** had an increasing trend; Fig. 2b shows **B** vs delay duration for an example network.

To gain further insight into the mechanisms that could account for the contraction bias we trained networks by resetting the network activity at the end of each trial, rather than maintaining continuity of activity during the inter-trial epoch (Methods). In principle, if the bias only depended on the information about past trials, then this bias would disappear in the absence of this information. In case this bias persisted, then we would have to look for some other mechanism capable of producing it. To answer this question we investigated the seven networks trained under this condition. The accuracy of six of them showed a clear bias, as can be observed in the network mean of the psychometric function (Fig. 2c, left) and the mean accuracy (Fig. 2c, middle). It is noteworthy that the bias **B** turns out to be weaker than in the networks trained with no reset (Fig. 2c, right).

So far we have analyzed behavioral data from trained networks with an eye to potential sources of the contraction bias: sensory history, noisy observations and prior knowledge and memorized stationary statistics. We will return to these results later, to examine them in light of a normative model that incorporates the presumed causes of contraction bias. But before that we now move on to study the neuronal activity to see if these factors are really present.

Population Analyses - State space analysis

We first searched for those neurons that encoded the value of the first duration (d_1) during the delay period. In each network, we found a large number of neurons that encoded this interval in a monotonic manner . Most analyses on neural firing activity were performed at the population level, the population being all those neurons encoding d_1. Next, we adopted the view that neural circuits represent and process information through the state-space dynamics of neural populations (Vyas et al., 2020; Saxena and Cunningham, 2019; Ebitz and Hayden, 2021; Carnevale et al., 2015; Chaisangmongkon et al., 2017; Mante et al., 2013; Russo et al., 2018; Wang et al., 2018). We then used principal component analysis

(PCA) to determine the directions in state space responsible for more than 90% of the data variance (z-score) and projected the population activity onto that subspace (Methods).

As expected, during the presentation of the first interval the population z-score followed a common path until the end of the duration of the respective d_1, defining a curved manifold (Fig. 3a). This feature also appeared in recordings of monkeys performing time reproduction tasks, during the presentation of the sample interval (Sohn et al., 2019, Remington et al., 2018; Egger et al., 2019). This similarity is not surprising, since both the first interval in the discrimination and the sample interval in the reproduction task must be represented in the network activity to complete the corresponding task. The trajectories started the delay period at different speeds, so that the longer the duration of their interval d_1, the higher the speed (Fig. 3b). During the delay period, the velocities of the trajectories decreased as they approached the end of the interval (Fig. 3c), reaching states that depended on the value of d_1, thus defining a pseudo-attractor region (Fig. 3c, inset). Correspondingly, at the end of the delay period the percentage of variance explained by the first principal component approaches 100%. This result shows that before the presentation of the d2 the duration of the first interval is encoded in the position on the one-dimensional quasi-attractor where the trajectory arrives. However, it tells us nothing about which estimator of d1 is maintained by the network. This question, as well as what geometrical properties of the trajectories encode the estimator, requires connecting behavior with population activity and will be addressed later.

If the networks were to implement a Bayesian computation, integrating the noisy measure of the current d1 with prior knowledge about the stimuli then, when d_1 occurs in two different contexts, the activity of the population should follow distinct trajectories. Since classes 5 and 6 are shared by the horizontal and diagonal blocks, we wondered whether the networks learnt that these blocks define different contexts. To verify this we first defined relative distances following the Kinematic analysis of Neural Trajectory (KiNeT) methodology (Remington et al., 2018). Briefly, for each common class (j=5,6) we considered its trajectories in the two contexts (trajectories j_h and j_d), then we selected one of them (e.g., j_d) as a reference and defined corresponding states between the reference and the other trajectory (j_h). If at time t after the alignment event (SO1 or SF1) the reference trajectory was in the state $s_{ref}(t)$, the corresponding state in trajectory j_h was taken as the state s_j closest to $s_{ref}(t)$. Then, for each time t, we computed the Euclidean distance D(t) between the pair of corresponding states. Distances between neural states were computed in a PC space which accounted for more than 90% of the total variance during the investigated epoch. Fig. 3d (left) shows the trajectories of classes 5 and 6, in the horizontal and diagonal blocks, during the presentation of the first time interval (example network 1). Common trajectories evolved disjointly, keeping a significantly non-zero relative distance between them (Fig. 3d, right).

During the delay period, trajectories common to the two contexts (Fig. 3e, left) also maintained significantly non-zero distances between them (Fig. 3e, right). These results indicate that the networks acquired knowledge about the context during the training stage. It should be noted that the networks were not trained specifically on the horizontal and diagonal contexts, since all the classes of the set in Fig. 1b, chosen randomly, were used for that purpose (Methods).

Neural activity is affected by sensory history

We now turn to check whether the neural activity in the current trial maintains information not only about the interval d_1 presented in this trial but also about those in the preceding ones. To do so, we begin with an analysis of the trajectories in state space, and then perform a more quantitative analysis of mutual information. Based on the results described above, it can be expected that if the neural population coded the value of the current d_1 throughout the delay period, then trajectories corresponding to consecutive durations should stay at significantly non-zero relative distances. Likewise, if the representation of the first interval persisted to the next trial, trajectories should still maintain non-zero distances during the inter-trial-interval (ITI). To verify if this is so, we calculated the relative distances between pairs of consecutive trajectories and verified whether they were significant. Again, we employed the KiNeT methodology (Remington et al., 2018); for each pair of trajectories (j and j+1) we selected trajectory j as a reference and, for each time t, we computed the Euclidean distance $D_j(t)$ between the pair of corresponding states (Methods). All pairs of consecutive trajectories in the diagonal set maintained non-zero relative distances during the delay period (Fig. 4a) and ITI (Fig. 4b). This implies that the neural representation of the first interval is maintained beyond the epoch in which it is presented.

To further support this analysis we analyzed the mutual information (MI) that the activity has on the presented time intervals. If the network was maintaining the value of the interval durations applied in preceding trials, then in the current trial there should be a fraction of neurons showing significant MI about those stimuli. Fig. 4c (left) shows the fraction of neurons with significant MI about the first interval in the current trial (red lines), the preceding trial (blue) and the second previous trial (green). Only correct and d_1-shorter choice trials have been used. Fig. 4c (right) presents the result of a similar analysis for d_1-longer choice trials. Soon after SO1 (after a time equivalent to the shortest first interval), the fraction of neurons with MI about the current d_1 increases rapidly and remains high during both the entire delay period and the ITI. The population of neurons with significant MI on the first interval of the preceding trial (blue), is quite large before SO1, during the delay period it stays at a similar level as the population with MI on the current d_1, although it becomes smaller during the ITI. In contrast, the fraction of neurons with MI on the second

previous d_1 decreases rapidly throughout the current trial. Fig. 4d exhibits the results of similar analyses for the fraction of neurons with significant MI about the second interval (d_2) in the current (red), the previous (blue) and the second previous (green) trials. Notice that during the ITI the fraction of neurons with MI on the second previous d_2 becomes close to significance level.

Note that these results differ from the conclusions obtained by Akrami et al. (2018). Recordings in posterior parietal cortex showed that neurons in this area only had information about the value of stimuli in previous trials; information about the current one supposedly is processed in other areas. Our model assumes an area in which information about the current and preceding stimuli converge.

Models. **Ideal observer of the delay-period activity**

In the above analyses of neuronal activity we verified that the factors suspected for being responsible for the contraction bias are indeed present in the trained networks. Now, taking advantage that normative models can help identify which computations are involved in producing the observed behavior, we will formulate a model containing the possible causes of the bias. The expectation in doing so is that such a model may guide us in understanding how these factors operate and interact to generate the bias. During the delay period, the activity of the trained networks integrates information about the current trial d_1 with information about past intervals (Fig. 4). To model behavior, we took the viewpoint of an ideal observer who has access to the delay-period activity, has knowledge about the stimulus set and can construct a belief about the duration of the presented interval. The activity observation made at trial n (o_n) combines the noisy measurement of the current interval ($d'_{1,n}$, which is assumed to be Gaussian with mean d_1 and standard deviation $\sigma_1=W_1d_1$. See Methods) with the representations of a few recent past intervals (m), and with the observations made at a longer temporal scale (o_{n-m-1}). That is,

$$o_n = (1-\eta)\, d'_{1,n} + (1-\eta)\eta\, d'_{1,n-1} + \ldots + (1-\eta)\eta^{m+1}\, d'_{1,n-m} + \eta^m\, o_{n-m-1} \qquad (1)$$

where the parameter η weights the contribution of preceding intervals (see Methods for further details) and $d'_{1,j}$ (j=n-1,…,n-m) are the Gaussian noisy measurements of the past intervals $d_{1,j}$. We assumed that the long-term history is in a Gaussian stationary regime, with the mean (M_{st}) and variance ($\Sigma_{1,st}^2$) obtained self-consistently (Methods). With these assumptions, the observation of the interval d_1 turns out to also be a Gaussian variable (o_1), with mean M and variance Σ_1^2. The observation of the second interval is simply the noisy measurement $d'_{2,n}$, also assumed to be Gaussian, with mean $d_{2,n}$ and standard deviation $\sigma_2=W_2d_2$. The prior probabilities of d_1 and d_2 were assumed to be discrete and uniform (Fig.

1b). Decisions about the states, "$d_1>d_2$" and "$d_1<d_2$", were made from the posterior probabilities using the *maximum a posteriori* decision rule (Methods).

We start by the simple case in which the delay period activity only combines the representation of the present first interval with the long-term sensory history (i.e., m=0). In this situation, the mean M and variance Σ_1^2 of the observation probability are

$$M = d_{1,n} + \eta\,(\langle d_1\rangle - d_{1,n}) \qquad (2a)$$
$$\Sigma_1^2 = \Sigma_1^2(W_1,\eta,\langle d_1\rangle,\sigma_{d1}^2,d_{1,n}), \qquad (2b)$$

where $\langle d_1\rangle$ is the first moment of d_1 in the stimulus set and σ_{d1}^2 is its variance (see Methods). Note that both M and Σ_1 could contribute to the contraction bias: The mean indicates explicitly that the first interval is contracted toward the interval mean, by an amount proportional to the difference between the true value of the interval and the d_1 mean. The variance provides a Bayesian contribution to the contraction bias, which operates when the uncertainty in the first interval exceeds the uncertainty in the second interval, i.e. when $\Sigma_1>\sigma_2$ (Ashourian et al., 2011). For each network, the model parameters (η, W_1 and W_2) were fitted to optimize the similarity between the model and the network performances (Methods). For this purpose we considered the accuracy curve of each network. The result for the example network 1 is shown in Fig. 5a (see Supplementary Fig. S5-1a for example network 2) and the RMSE distribution for the fits of all the networks appears in Fig. 5b. Analogous to the calculation of the bias **B** that we performed using the simulation data, we can now obtain a measure of the contraction bias, $\mathbf{B_{fit}}$, using the fit of the discrimination curves (Fig. 5c).

However, this simple version of the model neglects a possible role of the sensory history on the contraction bias. We now turn to the more relevant case in which the delay period activity integrates the representations of the current and previous first intervals and the long-term sensory history (i.e., the case m=1). Now, observations of the network activity have mean and variance

$$M = d_{1,n} + \eta\,(d_{1,n-1} - d_{1,n}) + \eta^2\,(\langle d_1\rangle - d_{1,n-1}) \qquad (3a)$$
$$\Sigma_1^2 = \Sigma_1^2(W_1,\eta,d_{1,n},d_{1,n-1},\langle d_1^2\rangle,\sigma_{d1}^2), \qquad (3b)$$

where $\langle d_1^2\rangle$ is the second moment of d_1 and σ_{d1}^2 is its variance (see Methods). Note that the observation of the current interval contracts towards $d_{1,n-1}$ which, in turn, appears contracted towards $\langle d_1\rangle$. Again, there is a Bayesian source for the bias when $\Sigma_1>\sigma_2$. Interestingly, the preceding interval affects the observation variance and, hence, the sensory history may modulate the bias generated by the Bayesian mechanism. Thus, in the model the two

mechanisms appear to be intricately related. The observation probability can be obtained for any m (Methods).

For each network, the parameters η, W_1 and W_2 were fitted to optimize the similarity between the model and the network performance (Methods). For this purpose, we considered the 10 accuracy curves, each modulated by one of the possible preceding trial classes (Fig. 2a for the example network 1), and used them all to fit the model (Methods). The corresponding fit is shown in Fig. 5d. (for the example network 2 see Supplementary Fig. S5-1b). The preceding class modulates the accuracy of the two example networks. The distribution of the RMSEs of the fits over all the networks is in Fig. 5e and the relationship between the bias **B** and the bias quantified with these fits appears in Fig. 5f. Fig. 5g shows the marginal accuracy curve, obtained by marginalizing over the first interval in the preceding trial (Methods. For example network 2 see Supplementary Fig. S5-1c).

Up to now, we studied the effect on network behavior of $d_{1,n-1}$. However, the activity during the delay period also has information about the second interval in the preceding trial, $d_{2,n-1}$ (Fig. 4d). This implies that the activity seen by the ideal observer (Eq. 1) should also contain a contribution from the previous d_2. If this is considered, the mean and variance of the observation probability are (see Methods)

$$M = d_{12,n} + \eta^2 (d_{12,n-1} - d_{12,n}) + \eta^4 (\langle d_{12}\rangle - d_{12,n-1}), \quad (4a)$$

$$\Sigma_1^2 = \Sigma_1^2(W_1,W_2,\eta,d_{1,n},d_{1,n-1},d_{2,n-1},d_{2,n-2},\langle d_1^2\rangle,\langle d_2^2\rangle,\sigma_{d1}^2,\sigma_{d2}^2), \quad (4b)$$

where $d_{12,n} = (d_{1,n} + \eta d_{2,n-1})/(1+\eta)$ and $\langle d_{12}\rangle = (\langle d_1\rangle + \eta \langle d_2\rangle)/(1+\eta)$ have weights normalized to one. In turn, the sum of all the weights in M is also one. This resembles the best multilinear regression fit obtained for magnitude discrimination (Akrami et al., 2018). Surprisingly, the quality of fit with this model is identical to that obtained with the model in which the contribution of the second interval is not considered (for network 1, RMSE=0.0012 versus 0.0011. See Supplementary Fig. S5-2 and Supplementary Table S1). For the sake of simplicity, we will disregard the contribution to the stimulation history of the second interval since its inclusion did not improve the quality of the fit. More specifically, in the following we will consider the version of the model whose mean and variance are given by Eqs. (3a,b). In addition, for illustrative purposes, in some topics we will also use the simpler version given in Eqs. (2a,b).

Above we saw that a weaker contraction bias is still present in networks trained resetting the network activity at the end of each trial (Fig. 2c). We now wonder about a model that describes this situation, and about the origin of this bias. Since the current trial is not affected by the stimulation history, the observation model in Eq. (1) should be excluded.

Instead, we considered the possibility that neural circuits may have knowledge of the stimulus statistics (e.g., knowledge of $<d_1>$), which presumably could have been learned during training trials and kept stored in the network synapses. We then contrasted the hypothesis that the circuits have knowledge about $<d_1>$ with the hypothesis that they do not. In the first case the observations have mean $M = d_{1,n} + \mu(<d_1> - d'_{1,n})$ and $\Sigma_1^2 = \sigma_1^2$ while in the second $M = d_{1,n}$ and variance $\Sigma_1^2 = \sigma_1^2$. We fitted the behavioral data for each network with both models obtaining comparable results in terms of the fit quality (Supplementary Fig. S5-3). Comparison of the two models using BIC or AIC tends to favor the model that does not require knowledge of $<d_1>$ (Supplementary Table S2). This means that the Bayesian mechanism could be enough to explain the origin of the contraction bias in networks trained using activity reset. Then, in these networks there is little evidence that synapses store the mean of the stimuli. If we were to translate this claim to networks trained without resetting the population activity, it would have to be explained how such networks estimate the moments from the distribution of d_1 (see, e.g., Eqs. 3). However, this does not pose a problem, since if the sensory history is accessible the moments can be estimated, for example, by means of exponential moving averages.

How the bias correlates with the model parameters

We will now leverage the normative model to understand how the sensory history and the uncertainty in the observations operate and interact to generate the bias. The dependence of the variance of the observation of d_1 on sensory history makes the analysis of these issues in the set of trained networks not immediate. Then, to investigate the relevance of the model parameters in the generation of the contraction bias, we first noted that the condition $\Sigma_1>\sigma_2$ determines whether the uncertainty in the observation of the first interval contributes to the bias. We will say that in networks where this condition is satisfied the Bayesian mechanism for generating the bias is operative. We then sorted the 35 networks into two groups: the group in which that condition was satisfied (the positive group) and the group in which it was not (the negative group). To obtain further insight into the role of each parameter, for each network group we evaluated the correlation between the phenomenological measure of the bias (**B)** and the quantities which are relevant for its generation, the history parameter (η) and the uncertainties in the observations of the two intervals (Σ_1 and W_2). Note the use of Σ_1 instead of W_1 in this analysis, this is because the quantity relevant for determining the bias is the total variance of the observation probability.

We start with the simplest version of the model (m=0, Eq. (2)). About half of the networks (18/35) belonged to the positive group, that is, the noise affecting the observation of the network activity was greater than that of d_2, signaling a contribution from the observation uncertainty to the contraction bias. As expected, in positive networks **B** was

significantly and positively correlated with Σ_1 (Fig. 6a and Supplementary Fig. S6-1a). Instead, its correlation with η was not significant (Fig. 6b). This is surprising, given that the mean M explicitly describes the mechanism by which the stimulation history contributes to the formation of the contraction bias (Eq. (3)). The explanation for this paradox is that the stimulation history also contributes to Σ_1, and does so in such a way that η is negatively correlated with the variance (Fig. 6c and Supplementary Fig. S6-2). This reduces the overall effect of η on the bias. Finally, **B** was not significantly correlated with W_2 (Fig. 6d), a parameter that only appears after the end of the delay period. We observed that in most networks the difference between the noise parameters, W_1-W_2, was positive (Fig. 6e). In negative networks the situation was the other way around: bias and variance were not significantly correlated (Fig. 6f and Supplementary Fig. S6-1b), in contrast, η and **B** were positively correlated (Fig. 6g). In negative networks **B** and W_2 are also uncorrelated, Fig. 6h.

A similar pattern of correlations held for the model version with m=1, Eq. (3). We again sorted the 35 networks into two groups, according to the sign of the difference (Σ_1-σ_2). For most networks (28/35), the fit satisfied that $\Sigma_1>\sigma_2$, meaning that the Bayesian mechanism was present more often than in the simpler model that did not consider transient past sensory contributions. As for the simpler model, in positive networks the bias **B** was favorably correlated with Σ_1 (Fig. 6i and Supplementary Fig. S6-3). Instead, its correlation with η was not significant (Fig. 6j). The reason for this feature was again the competition between the contributions of the sensory history to the mean M and to the variance Σ_1^2 (Fig. 6k and Supplementary Fig. S6-5). Again, **B** was not significantly correlated with W_2 (Fig. 6l). Finally, the difference W_1-W_2 was still positive (Fig. 6m). As for the negative networks, this model gives predictions similar to those found with the simpler model (Figs. 6n-q and Supplementary Fig. S6-4), although the positive correlation between **B** and η was somewhat below significance level (Fig. 6o).

In summary, inspired by our simulated data, we began by proposing a normative model that included two factors that supposedly produce the contraction bias, namely, the stimulation history and the observation uncertainty. Fitting the simulation data with this model confirmed that in most networks both mechanisms were necessary. Although the measurement of d_1 is affected by a fixed noise, given by the parameter W_1, when the ideal observer looks at the network activity, the uncertainty of his observation is affected by the stimulation history. We then conclude that the contraction bias results from the stimulation history, the uncertainty in interval measurements, and an interaction between these two factors. So far we did not mention how a network obtains the mean and variance of the distribution of d_1. One possibility is that it simply accumulates all interval measurements using, e.g., an exponential moving average. From this point of view, Eq. (1) can be taken as a useful Gaussian approximation, which in practice works quite well.

State space geometry is related to behavior

So far we have analyzed behavior and neural activity separately. Guided by a normative model of behavior we have 1) distinguished two groups of networks in which the Bayesian mechanism would have different relevance and 2) noted that sensory history affects the integration of the current interval with a priori knowledge about the stimuli. Nevertheless, these analyses do not inform us about how the computations suggested by the model might be implemented in the networks. As a step in that direction, we will now investigate how state-space geometry relates to behavior. The task requires that during the presentation of d_1 the neural activity represents an estimate of its duration and then maintains it during the delay period. Accordingly, we expect that over these two epochs an estimate of d_1 may be extracted from geometric properties of the trajectories (Sohn et al. 2019, Egger et al., 2019). We have seen that during interval presentation the trajectories define a one-dimensional manifold characterized by its curvature (Fig. 3a) while after the completion of each interval (SF1) the trajectories remain distant, reaching a linear attractor at the end of the delay epoch (Fig. 3b). So, we will now assume that estimations of the duration of the first interval are represented first in the curvature of the manifold (the "curved manifold hypothesis"; Sohn et al., 2019) and then, during the delay period, in the relative distances between each trajectory and a reference one. However, the nature of the estimator is unknown. In delayed comparison tasks the estimate of the first stimulus is not reported, only the comparison choices are available. This contrasts with magnitude reproduction tasks, in which the reproduced magnitude can be taken as a noisy version of the estimate (Jazayeri and Shadlen, 2010). Given the limited information about what estimator of d_1 used the networks, we investigated, apart from the true d_1, two other possible estimations, the Bayesian and Maximum Likelihood (ML) estimators (Methods), and studied how they related with the curvature and relative distances of the neural trajectories.

To find out which estimator best explained the curvature data, we projected the neural states in the curved manifold onto an encoding vector **u**, which we took as the vector pointing from the states associated with the shortest to the longest d_1 in the set (Sohn et al. 2019; see Methods). These projections (denoted as P_j, j=1-5) plotted vs d_1 appear in Fig. 7a for example network 1 (triangles). We then asked whether the P_j more closely reflected the true d_1 values or one of the two estimates, $d_{1,Bayes}$ or $d_{1,ML}$. To confront these possibilities, we first separately fitted a linear expression ($aP_j + b$) to the two estimators and to the true d_1. Then, we compared the corresponding goodness of fit using their RMSEs. Fig. 7b shows the regressions for the true d_1 (left), the Bayesian estimator (middle) and the ML estimator (right). For this example network the test favored the Bayesian estimator hypothesis (Fig. 7c). The same happened for all positive networks. In fact, the distributions over networks of

the difference in RMSE between any of the two estimators and the true d_1 are concentrated in positive values of those differences (Supplementary Figure S7-1, upper). Importantly, for the Bayesian estimator (left) the mean of this distribution was twice the mean of the distribution of the ML estimator (middle), indicating that the hypothesis that the curvature encodes the Bayesian estimator is better than the hypothesis that it encodes the ML estimator.

To determine which estimator of d_1 best accounted for the relative distances between the neural trajectories during the delay epoch, we followed a similar approach. During that epoch the duration of the first interval could be coded in the relative distances between the trajectories in state space. To check it we used the KiNeT method (Remington et al, 2018) to compute the relative distance $D_j(t)$ between each trajectory j=1-5 and the reference trajectory d_1=370 ms, over the delay period (Methods). The time courses of these distances for the example network 1 are shown in Fig. 7d, and their temporal means (D_j=<$D_j(t)$>), as a function of d_1, appear in the inset. We then asked whether the mean relative distances D_j coded better the true d_1 values or their Bayesian or ML estimators. To confront those alternatives, we first separately fitted a linear expression ($aD_j + b$) to the two estimates and to the true d_1. Then, we compared the goodness of fit using their RMSEs (Egger et al., 2019). Fig. 7e shows the regressions for the true d_1 (left), the Bayesian estimator (middle) and the ML estimator (right). For the example network the test favored the Bayesian estimator hypothesis (Fig. 7f). The same conclusion held for all the networks, except for one that preferred ML estimator. Again, the distributions over networks of the difference in RMSE between any of the two estimators and the true d_1 are concentrated in positive values of those differences (Supplementary Fig. S7-1, bottom). Notably, for the Bayesian estimator, the mean of the distribution of the difference in RMSE (left) was two times larger than the mean of the distribution for the ML estimator (middle), Showing that the hypothesis that the relative distances encode the Bayesian estimator is much better than the hypothesis that it encodes the ML estimator. The results of these tests provide us with a link between the behavioral fit and the geometry of state space trajectories.

Since the example network 2 fulfills $\Sigma_1<\sigma_2$, we do not expect it to pass either test well. To verify this expectation, we performed an analysis on network 2 similar to that on network 1. The result is shown in Supplementary Fig. S7-2. Although the 1-dimensional manifold did present curvature, it failed the test; the mean of the difference ($RMSE_{d1}$-$RMSE_{d1,Bayes}$) was approximately 2.5 times larger for network 1 than for network 2. As for the distance test, for network 2 the mean of the corresponding difference of RMSEs was negative.

Earlier, we quantified the contraction bias in terms of the discrimination curves (**B**). Now, we can quantify it using the estimator $d_{1,Bayes}$, which resulted from the behavioral fit. For this purpose, for each duration d_1, we first obtained the average difference between its

Bayesian estimate ($d_{1,Bayes}$) and its true value, and then computed the root mean square of the differences (Methods). This gave us a measure for the bias ($\mathbf{B}_{Bayes}$), derived from the Bayesian model, that correlated well with the empirical measure **B** (C=0.96, $p<10^{-6}$). In addition, $\mathbf{B}_{Bayes}$ had the same pattern of correlations with the model parameters as the phenomenological bias **B** (Supplementary Fig. S7-3, Fig. S7-4 and Fig. S7-5).

Effect of stimulation history on state space geometry

The previous study did not consider the effect of sensory history. In the following we show that the previous class modulates the geometry in state space and that it does so in a way that reflects the bias change that it generates. If the curved manifold hypothesis were correct then a higher contraction bias should be related to a higher compression on the encoding axis. To verify if this is true we leveraged the fact that the bias is modulated by the previous class. For example, in the example network 1, when varying the previous class along the set's upper diagonal (i.e., from class 1 to 5) the bias decreases approximately by half (**B**=0.12 vs. 0.06. Fig. 2a). Therefore, it is to be expected that the compression obtained using trials preceded by class 1 will be higher than the compression resulting from trials preceded by class 5. Likewise, the compression of the mean relative distances should be higher in trials preceded by class 1. We performed this analysis using the same compression measure for curvature and relative distances (Methods). Since curvature could be sensitive to the size of the analysis time window we considered various sizes of it. Fig. 8a shows the projection on the encoding axis versus d_1 (upper) and the relative distances versus d_1 (lower), for the two conditions. The results for the corresponding compressions appear in Fig. 8b; in agreement with our expectations, for both curvature (right) and relative distances (bottom left), compression was higher when conditioned to the previous class 1.

Previously, when studying the marginal case (i.e., without conditioning on the stimulation history), we found that the geometry of the trajectories encoded the Bayesian estimator of d1 better. So, it is pertinent to ask whether this property holds when conditioning on the stimulus history. The results of the curvature and relative distance tests, conditioning on the previous classes 1 and 5 are shown in Supplementary Fig. S8-1. As in the marginal case, the preferred estimator is the Bayesian one.

Network 2 has a similar bias **B** a s network 1 and undergoes a similar change in bias when conditioned on prior classes 1 and 5 (B=0.14 vs. 0.06. Supplementary Fig. S2-1c). However, since the condition $\Sigma_1<\sigma_2$ is not met by this network, no clear correspondence with the compression of geometrical properties of the trajectories can be expected. In fact, the compression of the projections on the encoding axis show a rather erratic behavior with the size of the analysis window, and the compression of the relative distances is null or small,

contradicting the existence of an appreciable contraction bias in the two conditions (Supplementary Fig. S8-2).

These analyses favor the hypothesis that the contraction bias manifests itself in state space as a compression induced by the curvature of a one-dimensional manifold during the presentation of the first interval and as a compression of relative distances during the memory period. In turn, these compressions can be explained by a Bayesian mechanism that is modulated by sensory history.

DISCUSSION

The contraction bias is influenced by the stimulation history (Raviv et al. 2012, Akrami et al., 2018). But the action of preceding stimuli is not the only possible explanation for this bias, which could also be generated from the integration of prior knowledge and noisy observations of the stimuli (Ashourian and Loewenstein, 2011; Lieder et al., 2019; Sarno et al., 2022). How the contraction bias results from the joint action of these factors has not been investigated. Here we have addressed this issue by training sRNNs on a delayed comparison task. The analysis of the behavioral and firing activity data allowed us to identify the sources of the contraction bias and to propose a normative Bayesian model that assisted us in determining the network computations responsible for the occurrence of the contraction bias.

The Bayesian explanation of the contraction bias operated in most trained networks (80%). In those networks the Bayesian explanation played a preponderant role, with sensory history modulating uncertainty in the observation of the first stimulus. Although the sensory history was always present, in practice it was correlated with the contraction bias in a low fraction of the networks (the remaining 20%).

During the execution of the task, successive geometric properties of the trajectories in the state space showed compressed representations of the real values of the first interval. During the presentation of that stimulus this happened on (a linear readout of) the curvature of a one-dimensional manifold; then, during the delay period, on the relative distances of the trajectories. These compressions are matched by the biased Bayesian estimator resulting from the normative model, indicating that the contraction bias emerges from the geometry of the trajectories in state space.

Several aspects of our results compare favorably with recent experimental studies of neural activity in both delayed comparison tasks and reproduction tasks. Akrami et al. (2018) found that during a discrimination task neurons in PPC carried information about recent sensory history. However, this area did not present information about the current trial stimulus, so it does not correspond to the model considered in our work, which assumes an area that has information about current and preceding stimuli. On the other hand, in a

temporal reproduction task Sohn et al. (2019) found that a one-dimensional manifold exhibited a compressed representation of the time interval, that was matched by the compression predicted by the Bayesian estimator (Sohn et al., 2019; see also Sohn and Narain, 2021). Our results showed that a similar relationship between population activity and behavior may occur even when an estimate of the interval is not reported in this task. Furthermore, other geometrical features that occur while the stimulus is being kept in short-term memory showed similar effects.

Our work advances the understanding of the origin of the contraction bias by proposing a unified description of the potential sources of this bias and showing that the Bayesian explanation is prevalent in recurrent networks trained to discriminate temporal intervals. The results raise the question of whether they occur in the brain; in this regard, one of the most pressing questions is the elucidation of in which areas and in what way the contraction bias is generated.

CODE AVAILABILITY. Custom MATLAB scripts used to implement the model and produce the figures will be available upon publication.

DATA AVAILABILITY. The simulated data that support the findings in this study are available from the corresponding author upon reasonable request

ACKNOWLEDGMENTS

This work was supported by the grant PGC2018-101992-B-I00 from the Spanish Ministry of Science, Innovation and Universities.

REFERENCES

Abbott, L. F., DePasquale, B., & Memmesheimer, R. M. (2016). Building functional networks of spiking model neurons. Nature neuroscience, 19(3), 350-355.

Akrami, A., Kopec, C. D., Diamond, M. E., & Brody, C. D. (2018). Posterior parietal cortex represents sensory history and mediates its effects on behaviour. *Nature*, *554*(7692), 368-372.

Ashourian, P., & Loewenstein, Y. (2011). Bayesian inference underlies the contraction bias in delayed comparison tasks. *PloS one*, *6*(5), e19551.

Bausenhart, K. M., Dyjas, O., & Ulrich, R. (2014). Temporal reproductions are influenced by an internal reference: Explaining the Vierordt effect. *Acta psychologica*, *147*, 60-67.

Carnevale, F., de Lafuente, V., Romo, R., Barak, O., & Parga, N. (2015). Dynamic control of response criterion in premotor cortex during perceptual detection under temporal uncertainty. *Neuron*, *86*(4), 1067-1077.

Chaisangmongkon, W., Swaminathan, S. K., Freedman, D. J., & Wang, X. J. (2017). Computing by robust transience: how the fronto-parietal network performs sequential, category-based decisions. Neuron, 93(6), 1504-1517.

DePasquale, B., Churchland, M. M., & Abbott, L. F. (2016). Using firing-rate dynamics to train recurrent networks of spiking model neurons. arXiv preprint arXiv:1601.07620.

DePasquale, B., Cueva, C. J., Rajan, K., Escola, G. S., & Abbott, L. F. (2018). full-FORCE: A target-based method for training recurrent networks. PloS one, 13(2), e0191527.

Dubreuil, A., Valente, A., Beiran, M., Mastrogiuseppe, F., & Ostojic, S. (2022). The role of population structure in computations through neural dynamics. *Nature Neuroscience*, 1-12.

Dyjas, O., & Ulrich, R. (2014). Effects of stimulus order on discrimination processes in comparative and equality judgements: Data and models. *Quarterly Journal of Experimental Psychology*, *67*(6), 1121-1150.

Dyjas, O., Bausenhart, K. M., & Ulrich, R. (2012). Trial-by-trial updating of an internal reference in discrimination tasks: Evidence from effects of stimulus order and trial sequence. *Attention, Perception, & Psychophysics*, *74*(8), 1819-1841.

Ebitz, R. B., & Hayden, B. Y. (2021). The population doctrine in cognitive neuroscience. *Neuron*, *109*(19), 3055-3068.

Egger, S. W., Remington, E. D., Chang, C. J., & Jazayeri, M. (2019). Internal models of sensorimotor integration regulate cortical dynamics. *Nature neuroscience*, *22*(11), 1871-1882.

Henke, J., Bunk, D., von Werder, D., Häusler, S., Flanagin, V. L., & Thurley, K. (2021). Distributed coding of duration in rodent prefrontal cortex during time reproduction. *Elife*, *10*, e71612.

Hollingworth, H. L. (1910). The central tendency of judgment. *The Journal of Philosophy, Psychology and Scientific Methods*, *7*(17), 461-469.

Jazayeri, M., & Shadlen, M. N. (2010). Temporal context calibrates interval timing. *Nature neuroscience*, *13*(8), 1020-1026.

Jazayeri, M., & Shadlen, M. N. (2015). A neural mechanism for sensing and reproducing a time interval. *Current Biology*, *25*(20), 2599-2609.

Lieder, I., Adam, V., Frenkel, O., Jaffe-Dax, S., Sahani, M., & Ahissar, M. (2019). Perceptual bias reveals slow-updating in autism and fast-forgetting in dyslexia. *Nature neuroscience*, *22*(2), 256-264.

Mante, V., Sussillo, D., Shenoy, K. V., & Newsome, W. T. (2013). Context-dependent computation by recurrent dynamics in prefrontal cortex. *nature*, *503*(7474), 78-84.

Meirhaeghe, N., Sohn, H., & Jazayeri, M. (2021). A precise and adaptive neural mechanism for predictive temporal processing in the frontal cortex. *Neuron*, *109*(18), 2995-3011.

Papadimitriou, C., Ferdoash, A., & Snyder, L. H. (2015). Ghosts in the machine: memory interference from the previous trial. *Journal of neurophysiology*, *113*(2), 567-577.

Petzschner, F. H., Glasauer, S., & Stephan, K. E. (2015). A Bayesian perspective on magnitude estimation. *Trends in cognitive sciences*, *19*(5), 285-293.

Preuschhof, C., Schubert, T., Villringer, A., & Heekeren, H. R. (2010). Prior Information biases stimulus representations during vibrotactile decision making. *Journal of Cognitive Neuroscience*, *22*(5), 875-887.

Raviv, O., Ahissar, M., & Loewenstein, Y. (2012). How recent history affects perception: the normative approach and its heuristic approximation. *PLoS computational biology*, *8*(10), e1002731.

Remington, E. D., Narain, D., Hosseini, E. A., & Jazayeri, M. (2018). Flexible sensorimotor computations through rapid reconfiguration of cortical dynamics. *Neuron*, *98*(5), 1005-1019.

Russo, A. A., Bittner, S. R., Perkins, S. M., Seely, J. S., London, B. M., Lara, A. H., ... & Churchland, M. M. (2018). Motor cortex embeds muscle-like commands in an untangled population response. *Neuron*, *97*(4), 953-966.

Sarno, S., Beirán, M., Falcó-Roget, J., Diaz-deLeon, G., Rossi-Pool, R., Romo, R., & Parga, N. (2022). Dopamine firing plays a dual role in coding reward prediction errors and signaling motivation in a working memory task. *Proceedings of the National Academy of Sciences*, *119*(2), e2113311119.

Saxena, S., & Cunningham, J. P. (2019). Towards the neural population doctrine. *Current opinion in neurobiology*, *55*, 103-111.

Shi, Z., & Burr, D. (2016). Predictive coding of multisensory timing. *Current Opinion in Behavioral Sciences*, *8*, 200-206.

Shi, Z., Church, R. M., & Meck, W. H. (2013). Bayesian optimization of time perception. *Trends in cognitive sciences*, *17*(11), 556-564.

Sohn, H., Narain, D., Meirhaeghe, N., & Jazayeri, M. (2019). Bayesian computation through cortical latent dynamics. Neuron, 103(5), 934-947.

Sohn, H., & Narain, D. (2021). Neural implementations of Bayesian inference. Current Opinion in Neurobiology, 70, 121-129.

Song, H. F., Yang, G. R., & Wang, X. J. (2016). Training excitatory-inhibitory recurrent neural networks for cognitive tasks: a simple and flexible framework. PLoS computational biology, 12(2), e1004792.

Song, H. F., Yang, G. R., & Wang, X. J. (2017). Reward-based training of recurrent neural networks for cognitive and value-based tasks. Elife, 6, e21492.

Thurley, K. (2016). Magnitude estimation with noisy integrators linked by an adaptive reference. *Frontiers in Integrative Neuroscience*, *10*, 6.

Vyas, S., Golub, M. D., Sussillo, D., & Shenoy, K. V. (2020). Computation through neural population dynamics. *Annual Review of Neuroscience*, *43*, 249.

Wang, J., Narain, D., Hosseini, E. A., & Jazayeri, M. (2018). Flexible timing by temporal scaling of cortical responses. Nature neuroscience, 21(1), 102-110.

Yang, G. R., Joglekar, M. R., Song, H. F., Newsome, W. T., & Wang, X. J. (2019). Task representations in neural networks trained to perform many cognitive tasks. *Nature neuroscience*, *22*(2), 297-306.

METHODS

Temporal Interval Discrimination (TID) task and simulations. Networks had to decide which of two time intervals, presented sequentially separated by a delay epoch, was the shortest (Fig. 1a). Each pair of intervals, with duration (d_1,d_2), is denoted as a class (Fig. 1b).

In our main simulated experiment, 35 networks were trained to perform task with the training set in Fig. 1b. A training trial began with the Start Cue event. After this event, a delay of 1s was presented until the onset (SO1) of the first time interval (d_1), which had variable duration (Fig. 1b). A second time interval (d_2) of variable duration (Fig. 1b) was presented after a delay period. The duration of this delay interval was 1.5s. One second after the offset of d_2 (SF2) a target signal, F_{out}, whose sign indicated the correct decision (positive for decision "$d_2 > d_1$", negative for "$d_2 < d_1$"), was presented. A beta distribution was used as target signal. These networks were trained with a block of 5000 training trials presented in a continuous manner, without perturbing the network activity at the end of the trial. During the training stage, the class used in a trial was selected randomly from the set in Fig. 1b, excluding the central class (370,370) ms.

Once a network had been trained, test trials were built in the same way as the training trials, except that the target was not presented. During the testing stage, we considered three different blocks of trials: the full block, the horizontal block and the diagonal block. Trials in the full block used all the classes in the set of Fig. 1b (including the central class). Trials in the horizontal block only considered classes with $d_2 = 370$ ms. Finally, trials in the diagonal block only used the diagonal classes (classes labeled from 1 to 10, Fig. 1b). From the full block, only its horizontal classes were used to ensure that the network learned the task. The horizontal block was only used, together with the diagonal block, to study whether the networks learned that both blocks define different contexts. In the rest of the analyses, models and tests, only the diagonal block was used. In terms of number of test trials, behavior was analyzed with 50.000 test trials per network. This behavioral data of 50.000 trials was fitted using Bayesian models. Analyses of neural activity were done with 4.000 test trials for studies in which the sensory history was not considered and with 20.000 test trials for studies conditioning on the preceding class.

Simulation with several delays period values. In addition, we considered a second numerical experiment in which the networks were trained to perform the task with several values of the delay period. In each training trial, one of four possible values of the delay period was selected randomly: 1.000 ms, 1.500 ms, 2.000 ms and 2500 ms. Once a network was trained, a block of 4.000 consecutive test trials was generated for each delay period. Each test trial randomly took a class from the diagonal block. 15 networks were trained and tested using this simulation.

Simulation with neural activity reset. In a third numerical experiment, 2.000 ms before the Start Cue of the following trial, the neural activity was reset to a random initial state, both on training and test trials. 7 networks were trained and tested with this simulation, obtaining separately two dataset per network of 4.000 test trials with diagonal block classes and with full block classes.

Network architecture and model neurons. We considered spiking recurrent neural networks (sRNNs) of 500 leaky integrate-and-fire (LIF) neurons. The dynamics of the membrane potential, V, is

$$\tau_m \frac{dV}{dt} = V_{rest} - V(t) + g[J \cdot s(t) + J^f \cdot f(t) + U \cdot F_{in}(t)] + I_{ext}, \tag{5}$$

where $\tau_m = 10$ ms is the membrane time constant and $V_{rest} = -55$ mV is the rest potential. U are the input weights (defined from an uniform distribution) through which both stimuli (d_1, d_2) are introduced into the sRNN as external input signals, F_{in}. J and J^f are plastic hidden recurrent synapses and non-plastic hidden recurrent connections, respectively. The corresponding synaptic currents $s(t)$ and $f(t)$ are

$$\tau_s \frac{ds}{dt} = -s(t) \quad \text{and} \quad \tau_f \frac{df}{dt} = -f(t), \tag{6}$$

where $\tau_s = 100$ ms for the slow current and $\tau_s = 5$ ms for the fast current. Finally, $g = 1.4$ is the gain factor.

Full-FORCE training algorithm. The full-FORCE algorithm was used as target-based supervised learning method to train sRNNs (dePasquale et al., 2016; dePasquale et al., 2018). The full-FORCE scheme trains a network to generate the desired target output, F_{out}, using target functions for the recurrent currents, F_J, obtained from an auxiliary RNN of rate neurons (target-generating network). These target functions provide the necessary information about the currents to train the recurrent connections of the sRNN.

Target-generating network. As target-generating network we used the continuous firing-rate model of neuron. The objective of this network is to provide a basis on target to sRNN. Therefore, target-generating network must be driven by desired target signals, F_{out}, in addition to the series of input events, F_{in}. Consequently, the dynamic of the target-generating network with $\tilde{N} = 500$ firing-rate units recurrently connected through the $\tilde{J}$ is governed by the following equation:

$$\tau_x \frac{dx}{dt} = -x(t) + \tilde{g}[\tilde{J} \cdot H(x) + \tilde{U} \cdot F_{in}(t) + \tilde{u} \cdot F_{out}(t)], \tag{7}$$

where the non-plastic connection matrix, $\tilde{J}$, is built from a Gaussian distribution of mean zero and variance $1/\tilde{N}$. H(x) = tanh(x) is the rate activity of a unit. $\tilde{U}$ and $\tilde{u}$ are the non-plastic synapses that introduce the input signal, F_{in}, and the target signal, F_{out}, respectively, into the target-generating network. Both $\tilde{U}$ and $\tilde{u}$ are built from an uniform distribution. Finally, $\tilde{g} = 1.4$ is the gain factor.

The auxiliary functions targets, F_J, are given as output of this target-generating network to sRNN through the u weights:

$$F_J(t) = u[\tilde{g}\tilde{J} \cdot H(x) + \tilde{u} \cdot F_{out}(t)]. \tag{8}$$

Training method of sRNN. For each training trial, both the plastic recurrent hidden weights, J, and output weights, W, are adjusted to generate an optimal network response. Considering both auxiliary functions, F_J, and the internal plastic current, $s(t)$, the recurrent hidden weights, J, are modified as follows:

$$J \cdot s(t) \approx F_J(t). \tag{9}$$

On the other hand, output weights, W, gets updated according to:

$$z(t) = W \cdot s(t) \approx F_{out}(t), \tag{10}$$

where $z(t)$ is considered as the output signal or readout produced by the sRNN.

Learning algorithm. Recursive Least Square (RLS) was used to minimize the cost function (Sussillo and Abbott, 2009) described as:

$$C_J =< J \cdot s(t) - \tilde{J} \cdot H(x) - \tilde{u} \cdot F_{out}(t) >^2 . \tag{11}$$

Testing. When collecting test trials, the targets were not presented. Thus, a network output was considered as correct if both F_{out} and the readout, $z(t)$, fulfilled the following condition during the target epoch (Ingrosso et al., 2019)

$$\frac{\sum_t z(t) \cdot F_{out}}{\sqrt{\sum_t z(t)^2}\sqrt{\sum_t (F_{out})^2}} > 0.5. \tag{12}$$

Conversely, a trial was taken as wrong if $z(t)$ satisfied Eq. (12) with $F_{out}(t)$ replaced by $-F_{out}(t)$. When the readout did not satisfy any of these two conditions it was considered as a noisy response. These trials were a small fraction of the test trials and were not used in the analyses.

Data analyses: Behavior. Performance curves. We can study the network behaviour through two performance curves:

Psychometric curve: This curve plots the probability to answer d_1 shorter than d_2 as a function of the values of first interval, d_1. It is built with only those test trials with horizontal classes when test block used was full block. It is used to study the network discrimination threshold. This curve shows a sigmoidal behavior. Therefore, it can be adjusted by the following nonlinear fit:

$$P(S) = \frac{1}{1 + e^{A_1 + A_2 \cdot d_1}}, \tag{13}$$

where P(S) is the probability that network chooses $d_1 < d_2$, i.e. S refers to decision shorter. The opposite choice, longer ($d_1 > d_2$), would be indicated by L. A_1 and A_2 are the adjusted parameters.

Discrimination curve (accuracy curve): This curve is plotted with the percentage of trials whose response was correct as a function of the first interval. It uses those test trials with diagonal classes when the chosen test block was diagonal block. Its purpose is to analyze the network learning properties.

Data analyses: Behavior. Contraction bias. Denoting the probability of decision S when the class was c as $P(S|c)$, the contraction bias B was quantified as the mean square deviation of $P(S|c)$ on diagonal classes relative to the middle classes

$$B^2 = \frac{1}{10}\left[\sum_{uc=1}^{5}\Big(P(S|uc) - P(S|3)\Big)^2 + \sum_{lc=6}^{10}\Big(P(S|lc) - P(S|8)\Big)^2\right], \tag{14}$$

where uc and lc denote classes on the upper and the lower diagonals, respectively.

Data analyses: Activity. Firing Rates. Firing rates are a useful tool to analyse the neural activity. The firing rate is computed as the number of spikes produced by a neuron for a specific time interval, denoted as analysis time window. Using a time window, we binned spike times for each correct trial either aligned to

SO1 (the analysis was during the presentation of the first stimulus) or to SF1 (the analysis was during the delay-period) or to SF2 (the analysis was during the ITI-period, i.e, inter-trial-interval) in order to estimate the instantaneous firing rate. Finally, we estimated the trial-averaged firing rates conditioning over the value of d_1.

The type of time window used depends on the desired analysis. We have focused on two types of window.

Anti-causal time window: It is used to count the future spikes, i.e. from the present, t_0, to the end of the time window, $t_0 + T$.

$$\rho = \frac{1}{T} \int_{t_0}^{t_0+T} S(t)dt, \tag{15}$$

where $S(t)$ is the spike counting function through time.

Half non-causal half causal time window: It is centered on the present and counts both past and future spikes:

$$\rho = \frac{1}{T} \int_{t_0-T/2}^{t_0+T/2} S(t)dt. \tag{16}$$

Data analyses: Activity. Mutual Information. We wanted to know how the neural response was modulated by each stimulus. In other words, we wanted to study the evolution across the current, previous and second previous trials of the number of neurons whose neural response had significant information about the first stimulus set ($d_1 = \{d_1^1, ..., d_1^{N_{d_1}}\}$). Also, we studied the information about the second stimulus. For this, it was necessary to quantify the spike timing response neuron per neuron. The trial was divided into L anti-causal bins of width $\Delta t = 0.250s$ that slides $0.025s$. Consequently, the single neuron response was quantified as a discrete array $r = \{r_1, ..., r_L\}$, where each r_i indicated the number of spikes fired in the ith time bin. R collected all possible values of r. The information about a time interval parameter d_1^i carried by a neuron was quantified with the mutual information $I(d_1|R)$ (Shannon, 1948):

$$I(d_1|R) = \sum_{d_1^i} P(d_1^i) \sum_r P(r|d_1^i) \log_2 \frac{P(r|d_1^i)}{P(r)}, \tag{17}$$

where $P(d_1^i)$ is the probability of a time interval d_1^i and $P(r|d_1^i)$ is the conditional probability of observing a response r given that the presented interval was d_1^i. To compute Eq. (17) we used a Matlab-based toolbox (Magri et al., 2009) that includes corrections for an upward bias in the estimate of the mutual information due to limited-sampling of the neural response (Treves and Panzeri, 1995). This bias was estimated using an extrapolation quadratic in the number of trials, N_{trial},

$$bias = \frac{a}{N_{trial}} + \frac{b}{(N_{trial})^2}, \tag{18}$$

where a and b are free parameters that depend on the stimulus-response probabilities.

Once percentage of neurons with mutual information has been computed, we used a shuffling procedure in order to destroy the history and, therefore, the causality between the responses of consecutive trials. This shuffling consisted of putting together all firing rates of all trials of all neurons, performing a certain number of permutations without replacement (25 in our case) and returning to each trial of each neuron a firing rate

different from the original. With this shuffled dataset, we calculated, bin-by-bin, the percentage of neurons with significant mutual information and its mean and standard deviation were plotted as horizontal shadings in Fig. 4c-d.

Data analyses. Unit selection and classification. Except in the analyzes with mutual information, where we have used the entire population of 500 neurons, in the rest of the analyzes and models we have used a sub-population of neurons. The particularity of these neurons is that their firing rate increases or decreases monotonically with the stimulus during the delay period. In other words, these neurons are coded the perceived stimulus. For this classification, we fitted the firing rate of each neurons with a linear function. If the slope of this linear fit was significant and positive, then the neuron was positively tuned and it was classified as *plus*. If the slope of this linear fit was significant and negative, then the neuron was negatively tuned and it was classified as *minus*.

Data analyses: Activity. Principal Component Analysis (PCA). In order to analyze the population activity we used PCA to examine the trial-averaged firing rates and to identify the low-dimensional space that captured as much information as was feasible. For this, firing rate matrices of all neurons with tuning conditioning over d_1 were built. We then concatenated these matrices across the five values of d_1 along the time dimension in order to apply PCA and find the required number of principal components (PCs). We used as many PCs (typically 4-6) as necessary to explain around 90%-95% of variance in the data. Neural trajectories were analyzed within the subspace spanned by these top PCs.

Data analyses: Activity. KiNeT. It is possible to estimate the relative distances between a pair of trajectories (i and j) using the Kinematic analysis of Neural Trajectory (KiNeT) methodology introduced in (Remington et al., 2018). For this, we selected one of the two trajectories as a reference (e.g., ref=i) and we defined the corresponding states between the reference and the other trajectory (j). If at time t after the aligning event (SO1 for the presentation epoch, SF1 for the delay period and SF2 for the ITI) the reference trajectory was in the state $s_{ref}(t)$, then we looked for the nearest state in trajectory j, $s_j(t)$, to $s_{ref}(t)$, i.e. we looked for the minimum Euclidean distance between trajectory j and the state $s_{ref}(t)$. This is the relative distance between the non-reference trajectory and the reference trajectory at time t, $D_j(t)$. Distances between neural trajectories were computed in a PC space which accounted for more than 90% of the total variance during the investigated epoch. This analysis has been used in the following situations.

State-space trajectories depend on the context. To verify that classes common to horizontal and diagonal blocks (classes 5 and 6) define different trajectories, we computed their relative distances in both contexts taking the diagonal context as reference. Principal Components and the corresponding KiNeT analysis were made for each one of the 100 resamples of a bootstrap procedure with replacement.

State-space trajectories depend on d_1 until the next trial. To check whether the duration of the first interval was transmitted to the delay interval or the ITI we checked whether the relative distances between the trajectories corresponding to different values of d_1 were significantly nonzero.

Relative distances during the delay epoch encode an estimate of d_1. We used KiNeT with the purpose of checking

if an estimate of d_1 may be extracted from the relative distances between the neural trajectories during the delay epoch (see Fig. 7d) (Sohn et al. 2019, Egger et al., 2019).

Data analyses: Activity. State-space Geometry and Compression. *Curvature.* To investigate the representation of the first interval in state space during the presentation epoch we analyzed the curvature of the nonlinear manifold developed during that period. We proceded as in Sohn et al., 2019. One way to measure the curvature is by projecting the neural trajectories onto an "encoding axis", $\vec{u}$. This axis is a unit vector defined between the state associated with the shortest SF1, $X(t = SF1_1)$, and the longest SF1, $X(t = SF1_5)$. The projected state within PC space at time $t = SF1_j$, $X(SF1_j)$, onto the encoding axis, $\vec{u}$, is denoted as P_j. It can be computed as the projection of a point on a given line in Euclidean space if we consider a second line perpendicular to the first one and containing both the projected point and the desired projection:

$$P_j = \frac{\overrightarrow{X(SF1_1)X(SF1_j)} \cdot \vec{u}}{||\vec{u}||^2}\vec{u} + X(SF1_1). \tag{19}$$

Distances relative to the central trajectory. To study the representations of the first interval during the delay period we analyzed the relative distances between the trajectories for every value of d_1, $D_j(t)$, and a reference trajectory taken as that corresponding to $d_1 = 370$ ms.

Compression. Compression of projections onto encoding axis during representation of the first interval can be computed as follow:

$$C[P] = 1 - min\left(\frac{P_3 - P_1}{P_5 - P_3}, \frac{P_5 - P_3}{P_3 - P_1}\right). \tag{20}$$

Compression of the mean of relative distances, $D_j =< D_j(t) >$, over time during delay period can be computed in the same way as Eq. 20:

$$C[D] = 1 - min\left(\frac{D_3 - D_1}{D_5 - D_3}, \frac{D_5 - D_3}{D_3 - D_1}\right). \tag{21}$$

Bayesian Model. Formulation and Likelihood. We assumed that at the current trial n the Bayesian observer makes an observation of the first interval ($o_{1,n}$) given by a combination of the current-trial noisy measurement of d_1 ($d'_{1,n}$) and the observation in the preceding trial

$$o_{1,n} = (1 - \eta)\, d'_{1,n} + \eta o_{1,n-1}. \tag{22}$$

Here η is a parameter that measures the contribution of the observation in the preceding trial. To work with analytical expressions, instead of iterating this equation we expressed $o_{1,n}$ by extracting the interval measurements made in the m past trials (Eq. (1) in the main text)

$$o_{1,n} = \sum_{j=0}^{m} \eta^j\, (1 - \eta)\, d'_{1,n-j} + \eta^{m+1} o_{1,n-(m+1)}. \tag{23}$$

We assumed that the noisy measurements of the intervals are described by normal distributions ($j = 0, 1, ..., m$)

$$d'_{1,n-j} \sim \mathcal{N}\left(d_{1,n-j}, W_1^2 d_{1,n-j}^2\right), \tag{24}$$

where W_1 is the Weber fraction, a parameter of the Bayesian model to be determined. In addition, we assumed that trials prior to the last m constitute a Gaussian stationary regime (the long-term sensory history). The mean and variance of the stationary variable are obtained using self-consistency

$$M_{1,st} = \langle d_1 \rangle \tag{25}$$

$$\Sigma_{1,st}^2 = \frac{1-\eta}{1+\eta}\left(W_1^2 \langle d_1^2 \rangle + \sigma_{d_1}^2\right), \tag{26}$$

where $\langle d_1 \rangle$ and $\sigma_{d_1}^2$ are the mean and variance of the prior probability of d_1. Then, the observation in the current trial, described by the variable o_1, is also a Gaussian variable with mean $\mathcal{M}$ and variance Σ_1^2

$$o_1 \sim \mathcal{N}\left(\mathcal{M}, \Sigma_1^2\right). \tag{27}$$

The distribution of this variable defines the likelihood of an observation o_1, given the short-term sensory history, that we denote by $P\left(o_1 | d_{1,n}, \ldots, d_{1,n-m}\right)$.

We considered two cases of this model. In the first case all past trials contribute to the stationary regime (i.e., $m = 0$). The likelihood $P\left(o_1 | d_{1,n}\right)$ has mean and variance

$$\mathcal{M} = d_1 + \eta(\langle d_1 \rangle - d_1) \tag{28}$$

$$\Sigma_1^2 = (1-\eta)^2 W_1^2 d_1^2 + \eta^2 \Sigma_{1,st}^2. \tag{29}$$

In the second case the transient regime contains only the preceding trial (i.e., $m = 1$). The likelihood $P\left(o_1 | d_{1,n}, d_{1,n-1}\right)$ has mean and variance

$$\mathcal{M} = d_{1,n} + \eta\Big[d_{1,n-1} - d_{1,n} + \eta\left(\langle d_1 \rangle - d_{1,n-1}\right)\Big] \tag{30}$$

$$\Sigma_1^2 = (1-\eta)^2 W_1^2 d_{1,n}^2 + \eta^2 (1-\eta)^2 W_1^2 d_{1,n-1}^2 + \eta^4 \Sigma_{1,st}^2. \tag{31}$$

The mean and variance for a more general case, in which the transient consists of the previous m trials, is given in the Supplementary Text 1.

This model does not consider the contribution of the noisy measurements of the second interval in the previous trial ($d'_{2,n-1}$). When it is taken into account, the mean and variance of the stationary variable are

$$M_{12,st} = \frac{\langle d_1 \rangle + \eta \langle d_2 \rangle}{1+\eta} \tag{32}$$

$$\Sigma_{12,st}^2 = \frac{1-\eta}{1+\eta}\Big[\frac{W_1^2 \langle d_1^2 \rangle + \sigma_{d_1}^2 + \eta^2\left(W_2^2 \langle d_2^2 \rangle + \sigma_{d_2}^2\right)}{1+\eta^2}\Big], \tag{33}$$

where $\langle d_2 \rangle$ and $\sigma_{d_2}^2$ are the mean and variance of the prior probability of d_2. The mean and variance of the observation probability are

$$\mathcal{M} = d_{12,n} + \eta^2 \Big[d_{12,n-1} - d_{12,n} + \eta^2 \left(\langle d_{12} \rangle - d_{12,n-1} \right) \Big] \tag{34}$$

$$\Sigma_1^2 = (1-\eta)^2 \left[W_1^2 d_{1,n}^2 + \eta^4 W_1^2 d_{1,n-1}^2 \right] + \eta^2 (1-\eta)^2 \left[W_2^2 d_{2,n-1}^2 + \eta^4 W_2^2 d_{2,n-2}^2 \right] + \eta^8 \Sigma_{12,st}^2. \tag{35}$$

The mean was expressed in terms of the normalized quantities

$$d_{12,n} = \frac{d_{1,n} + \eta d_{2,n-1}}{1+\eta}$$

$$d_{12,n-1} = \frac{d_{1,n-1} + \eta d_{2,n-2}}{1+\eta}$$

$$\langle d_{12} \rangle = \frac{\langle d_1 \rangle + \eta \langle d_2 \rangle}{1+\eta}.$$

In addition, the weights of the various contributions to the mean are also normalized. This is reminiscent of a result by Akrami et al. (2018) who found that the weights in their best multilinear regression model appear normalized.

Finally, the noisy observation of d_2, denoted as o_2, was taken as a Gaussian variable with mean d_2 and standard deviation $\sigma_2 = W_2 d_2$,

$$o_2 \sim \mathcal{N}\left(d_2, W_2^2 d_2^2 \right). \tag{36}$$

The distribution of this variable defines the likelihood of an observation o_2, given the current trial second interval, that we denote by $P\left(o_2 | d_{2,n}\right)$.

Bayesian Model. Beliefs. The computation of posterior probabilities closely follows a Bayesian model for tactile frequency discrimination (Sarno et al., 2022). For the sake of completeness, here we adapt it to temporal interval discrimination. The main difference is that the likelihoods are given by the expressions derived above. We will present the equations only for the case $m = 1$. The equations for the case in which the previous second interval is also given, or the case in which none of the previous intervals is fixed, are derived in a similar way.

The discrete prior probabilities of the first interval are denoted as $P(d_1^i)$ ($i = 1, \ldots, 5$ labels the possible values of d_1 in the stimulus set, Fig. 1b). Bayes theorem establishes that the posterior distribution $P(d_1^i | o_1, d_{1,-1}^j)$, conditioned to the preceding interval $d_{1,-1}^j$ ($j = 1, \ldots, 5$), results from combining the noisy observation with the prior distribution,

$$b_1(d_{1,0}^i | d_{1,-1}^j) \equiv P(d_{1,0}^i | o_1, d_{1,-1}^j) \propto P(o_1 | d_{1,0}^i, d_{1,-1}^j) P(d_{1,0}^i). \tag{37}$$

Notice that the notation was simplified denoting $d_{1,n}$ and $d_{1,n-1}$ as $d_{1,0}$ and $d_{1,-1}$, respectively. The left hand side indicates that this probability is also known as the belief $b_1(d_{1,0}^j | d_{1,-1}^j)$ about the duration of the first time interval, conditioned to the duration in the previous trial. One may need to marginalize over $d_{1,-1}^j$, summing over all values of j.

Denoting the discrete prior probabilities of the second time interval as $P(d_2^l)$ ($l = 1, \ldots, 9$ labels the possible values of d_2 in the stimulus set, Fig. 1b), the posterior probability $P(d_2^l | o_2)$, or belief $b_2(d_2^l)$ about the duration of the second time interval, is

$$b_2(d_2^l) \equiv P(d_2^l|o_2) \propto P(o_2|d_2^l)P(d_2^l). \tag{38}$$

Let us denote class k as $c_k = (d_1^k, d_2^k)$, where $k = 1, \ldots, 10$ is a class label (Fig. 1b). The joint posterior probability or joint belief that class c_k had been presented in the current trial, conditioned to the interval in the previous trial (the observations were o_1 and o_2) (Sarno et al., 2022)

$$B(c_k|o_1, o_2, d_{1,-1}^j) = \frac{P(o_2)}{P(o_2|o_1)} \frac{P(d_2 = d_2^k|d_1 = d_1^k)}{P(d_2^k)} b_1(d_1^k|d_{1,-1}^j) b_2(d_2^k). \tag{39}$$

The first quotient is a normalization factor. $P(d_2 = d_2^k|d_1 = d_1^k)$ is the transition matrix relating the probability of presenting d_2 after d_1. If we assume that network knows perfectly the structure of the set, then the only non-zero matrix elements correspond to the 10 classes in the stimulus set. Note that this depends on the block under consideration (full, diagonal or horizontal blocks, Fig. 1b). For example, in the full block, for a given value of the first interval, the second interval can take either one or two or three different values. Thus, the non-zero transition probabilities could be 1, 1/2 or 1/3, respectively.

The Bayesian observer determines his choice based on about which of the states, $d_1 < d_2$ or $d_1 > d_2$, he has a higher belief. These beliefs are computed by adding the class beliefs $B(k|o_1^*, o_2)$'s over classes k for a specific choice

$$\begin{cases} b(L|d_{1,-1}^j) = \sum_{d_1 > d_2} B(k|o_1, o_2, d_{1,-1}^j) & \text{choice } d_1 \text{ Longer, } d_1 > d_2 \\ b(S|d_{1,-1}^j) = 1 - b(L|d_{1,-1}^j) & \text{choice } d_1 \text{ Shorter, } d_1 < d_2. \end{cases} \tag{40}$$

Bayesian Model. Probability of deciding $d_1 < d_2$. The conditional probability of an ideal Bayesian observer, with knowledge of o_1 and o_2, to choose d_1 shorter when the presented class was $(d_{1,0}^k, d_{2,0}^k)$ is

$$P(d_1 < d_2|d_{1,-1}^j) = \sum_{o_1, o_2} P(o_1|d_{1,0}^k, d_{1,-1}^j) P(o_2|d_{2,0}^k) G(b(S|d_{1,-1}^j)), \tag{41}$$

where $G(b(S|d_{1,-1}^j))$ is a sigmoidal function centered at $b(S|d_{1,-1}^j) = 0.5$, which slope tends to 0 if $b(S|d_{1,-1}^j) \ll 0.5$ and tends to 1 in the opposite limit (Sarno et al., 2022; see also Ashourian et al., 2011).

At this point, we decided to include a win-stay and loss-shift strategy (WSLS) in the decision making. This strategy consists of the temptation to change the answer after a previous trials that was incorrect (a loss) and to maintain the same answer that was given in the previous trial if said choice was correct (a win). In order to model this effect, we included two more parameters to the m=1 model: γ_1 (used if the previous class was from the upper diagonal) and γ_2 (used if previous class belonged to the lower diagonal). Both parameters allow the model to reinforce or not the choices $d_1 < d_2$ and $d_1 > d_2$ following the WSLS strategy. Thus, we rewrite the argument of G in Eq. (41) as follows:

$$P(d_1 < d_2|d_{1,-1}^j) = \sum_{o_1, o_2} P(o_1|d_{1,0}^k, d_{1,-1}^j) P(o_2|d_{2,0}^k) G(arg), \tag{42}$$

where arg depends on the previous outcome and the origin (upper or lower diagonal) of the previous class:

- Previous trial was correct with a class from upper diagonal: $arg = (1 - \gamma_1)\ b(S|d_{1,-1}^j) + \gamma_1$.

– Previous trial was correct with a class from lower diagonal: $arg = (1-\gamma_2)\ b(S|d^j_{1,-1})$.

– Previous trial was incorrect with a class from upper diagonal: $arg = (1-\gamma_1)\ b(S|d^j_{1,-1})$.

– Previous trial was incorrect with a class from lower diagonal: $arg = (1-\gamma_2)\ b(S|d^j_{1,-1}) + \gamma_2$.

Bayesian Model. Correlations. In order to find correlations between the model parameters with each other and with the bias, B, we used the MATLAB function *corrcoef* to obtain the correlation coefficients and the p-value for testing the null hypothesis that there is no relationship between the chosen parameters. A correlation was considered as significant if p-value was smaller than the significance level, 0.05.

Bayesian and Maximum Likelihood estimators. For a given trial n, the Bayesian estimator ($d^{(e)}_{1,n} = d^{Bayes}_{1,n}$) results from a loss function. Taking it as the squared error

$$(d^{Bayes}_{1,n} - d_{1,n})^2,$$

the Bayesian estimator is the mean of d_1 weighted with the conditional posterior probability. Marginalizing on the previous trial first interval,

$$d^{Bayes}_{1,n} = \sum_i d^i_1 \cdot b_1(d^i_{1,n}). \tag{43}$$

Likewise, the ML estimator ($d^{(e)}_{1,n} = d^{ML}_{1,n}$) results from a loss function. Taking it as the squared error

$$(d^{ML}_{1,n} - d_{1,n})^2,$$

the Maximum Likelihood estimator is associated with the first stimulus that maximizes the likelihood function. The marginalized case is given by:

$$d^{ML}_{1,n} = \arg\max_{d_1} P(o_1|d^i_{1,n}). \tag{44}$$

However, equations Eq. (43) and (44) only estimate a given first stimulus observation on a given trial n. Once the model parameters have been determined, we have carried out a simulation with 10.000 trials in which we estimated, with previous equations, the Bayesian and ML estimation of each of the 10.000 observations. Once simulation finished, we grouped trials based on the real d_1 and averaged the Bayesian/ML estimate for each group. In this way we obtained an averaged Bayesian ($d_{1,Bayes}$) and ML ($d_{1,ML}$) estimates for each value of d^i_1.

The same simulations could be made conditioning in the preceding trial first interval. But with the conditioned case we used a greater number of trials, 100.000. The Bayesian estimator conditioning in the previous trial first interval is given by:

$$d^{Bayes}_{1,n}(d^j_{1,n-1}) = \sum_i d^i_1 \cdot b_1(d^i_{1,n}|d^j_{1,n-1}), \tag{45}$$

and the conditioned ML estimator is given by

$$d^{ML}_{1,n}(d^j_{1,n-1}) = \arg\max_{d_1} P(o_1|d^i_{1,n}, d^j_{1,n-1}). \tag{46}$$

Fitting the Model to Data. The model parameters (σ_1, σ_2, η, γ_1 and γ_2) were fitted by minimizing the mean squared difference between the model and the network performances. This minimization was done using the *simulannealbnd* command in MATLAB (Mathworks) which finds the minimum using a simulated annealing algorithm, setting of lower and upper bounds on the model parameters. For the case $m = 1$ (conditioning over the previous class), the minimized error resulted from summing over the ten differences between the model and the network performances (one difference per previous class). Thus, with the version m=1 we obtained a model accuracy modulated by both the preceding class and the previous outcome. In order to obtain the marginal accuracy, we marginalized over both the class and outcome in the preceding trial.

Marginalize over previous class and outcome. From model version m=1 we obtained the accuracy conditioned on the current class, the previous class and the previous outcome. We marginalized on the previous outcome and on the previous class in order to obtain the marginal discrimination curve. Let us denote current class r as $C_{0,r} = (d_1^r, d_2^r)$, where $r = 1, \ldots, 10$ is a class label (Fig. 1b). In the same way we denote the previous class s as $C_{-1,s} = (d_1^s, d_2^s)$, where $s = 1, \ldots, 10$ is a class label (Fig. 1b). Also, the previous outcome, U_{-1}, is denoted as $U_{-1} = 1$ if it was correct and as $U_{-1} = 0$ if it was incorrect. The probability of answering Shorter with knowledge of both the previous class and the current class can be decomposed as follows

$$\begin{aligned} P(S|C_0,\, C_{-1}) = P(S|C_{0,r},\, C_{-1,s},\, U_{-1} = 1) \cdot P(C_{-1},\, U_{-1} = 1) + \\ + P(S|C_{0,r},\, C_{-1,s},\, U_{-1} = 0) \cdot P(C_{-1},\, U_{-1} = 0). \end{aligned} \tag{47}$$

For a given current class, r, we extract the previous class from the previous equation:

$$\begin{aligned} \sum_s \frac{1}{10} P(S|C_0) = \sum_s \frac{1}{10} \Big[P(S|C_{0,r},\, C_{-1,s},\, U_{-1} = 1) \cdot P(C_{-1},\, U_{-1} = 1) + \\ + P(S|C_{0,r},\, C_{-1,s},\, U_{-1} = 0) \cdot P(C_{-1},\, U_{-1} = 0) \Big]. \end{aligned} \tag{48}$$

If the current class belongs to the upper diagonal, then the probability of hit is equal to the probability $d_1 < d_2$. In other words, if the current class is on the upper diagonal, then $P(S|C_0) = P(C_0,\, U_0 = 1)$:

$$\begin{aligned} 10P(C_0,\, U_0 = 1) = \sum_s \Big[P(S|C_{0,r},\, C_{-1,s},\, U_{-1} = 1) \cdot P(C_{-1},\, U_{-1} = 1) + \\ + P(S|C_{0,r},\, C_{-1,s},\, U_{-1} = 0) \cdot P(C_{-1},\, U_{-1} = 0) \Big]. \end{aligned} \tag{49}$$

Also we have $P(C_{-1},\, U_{-1} = 0) = 1 - P(C_{-1},\, U_{-1} = 1)$:

$$\begin{aligned} 10P(C_0,\, U_0 = 1) = \sum_s \Big[P(S|C_{0,r},\, C_{-1,s},\, U_{-1} = 1) \cdot P(C_{-1},\, U_{-1} = 1) + \\ + P(S|C_{0,r},\, C_{-1,s},\, U_{-1} = 0) \cdot \Big(1 - P(C_{-1},\, U_{-1} = 1)\Big) \Big]. \end{aligned} \tag{50}$$

Reorganizing the previous equation we have, for a given current class, r:

$$\sum_s [P(S|C_{0,r},\, C_{-1,s},\, U_{-1}=1) - P(S|C_{0,r},\, C_{-1,s},\, U_{-1}=0)] \cdot\, P(C_{-1,s},\, U_{-1}=1) - 10P(C_{0,r},\, U_0=1) = \\ -\sum_s P(S|C_{0,r},\, C_{-1,s},\, U_{-1}=0).$$

If the current class is the same as the previous class, r=s, then the probability of hit of the current class is equal to the probability of hit of the previous class $P(C_{0,r},\, U_0=1) = P(C_{-1,r},\, U_{-1}=1)$:

$$\sum_s [P(S|C_{0,r},\, C_{-1,s},\, U_{-1}=1) - P(S|C_{0,r},\, C_{-1,s},\, U_{-1}=0)] \cdot\, P(C_{-1,s},\, U_{-1}=1) - 10P(C_{-1,r},\, U_{-1}=1) = \\ -\sum_s P(S|C_{0,r},\, C_{-1,s},\, U_{-1}=0). \quad (51)$$

In short, we have 5 equations with 5 unknowns, one for each current class located on the upper diagonal. On the contrary, if the current class is on the bottom diagonal, then we would obtain the following 5 equations another 5 unknowns, one for each current class located on the bottom diagonal:

$$\sum_s [P(S|C_{0,r},\, C_{-1,s},\, U_{-1}=1) - P(S|C_{0,r},\, C_{-1,s},\, U_{-1}=0)] \cdot\, P(C_{-1,s},\, U_{-1}=1) + 10P(C_{-1,r},\, U_{-1}=1) = \\ 10 - \sum_s P(S|C_{0,r},\, C_{-1,s},\, U_{-1}=0). \quad (52)$$

Solving this system of 10 equations we obtain the accuracy, $P(C_0,\, U_0 = 1)$, for each current class (10 unknowns). With this, it is possible to compute the probability $d_1 < d_2$ conditioned over current class and previous class, $P(S|C_0,\, C_{-1})$ using Eq. (47).

Model comparison. Akaike Information Criteria (AIC):

$$AIC = -2log(L) + 2N, \quad (53)$$

where N is the number of model parameters and L is the likelihood of obtaining the observed choice ($d_1 < d_2$ or $d_1 > d_2$) across all trials. For given class and choice, the likelihood function takes the values $l(x = d_1 < d_2) = P(d_1 < d_2)$ or $l(x = d_1 > d_2) = P(d_1 > d_2)$. Then, $log(L)$ is the sum over all trials:

$$log(L) = \sum_{n=1}^{N_{trials}} log(l(x_i)). \quad (54)$$

Bayesian (Schwarz) Information Criteria (BIC):

$$BIC = -2log(L) + Nlog(N_{Obs}), \quad (55)$$

where N_{Obs} is the sample size (number of observations).

Relationship between geometry and behavior. We asked whether Bayesian estimates of d_1 were being encoded by state space geometry. For this, we proposed three hypotheses: State space geometry codes the true

value of d_1 or it codes a combination of current d_1 and prior knowledge of d_1 as a Bayesian estimator or as a Maximum Likelihood (ML) estimator. Depending on the trial epoch, we evaluated these hypotheses as follows.

Testing during the presentation of the first stimulus (Sohn et al., 2019): We wanted to check which hypothesis best explained the curvature data during the presentation of the first interval. We did it fitting, separately, these two estimates of d_1 and the true value of d_1 with a linear regression of the projections, P_j,

$$d_1;\ d_{1,Bayes};\ d_{1,ML} = aP_j + b. \tag{56}$$

The goodness of the fits was compared using their Root Mean Square Errors (RMSEs). In order to obtain greater precision in the test, the size of the analysis window used to compute the firing rates was reduced to 100 ms with a slide of 10 ms throughout the presentation of the first interval. The nature of this filter was half non-causal half causal. We used the number of PCs needed to explain, at least, 95% of the variance (between 3 and 5 PCs).

Testing during the delay-period (Egger et al. 2019): To check whether relative distances between neural trajectories during delay period are encoding Bayesian estimates we repeated the previous test but using the mean of relative distances through time, D_j, instead of projections.

$$d_1;\ d_{1,Bayes};\ d_{1,ML} = aD_j + b. \tag{57}$$

In this test we used a totally non-causal filter whose size was 100 ms with a slide of 10 ms throughout the delay period. We used the number of PCs needed to explain, at least, 90% of the variance (between 5 and 11 PCs).

Finally, we used, in both tests, a bootstrap procedure in order to evaluate its significance. From the original sample of neurons population we created 100 resamples by repeatedly sampling with replacement. We made both tests with each resample and the 100 test results were used as a dispersion of the test itself.

REFERENCES

Ingrosso, A., & Abbott, L. F. (2019). Training dynamically balanced excitatory-inhibitory networks. PloS one, 14(8), e0220547.

Magri, C., Whittingstall, K., Singh, V., Logothetis, N. K., and Panzeri, S. (2009). A toolbox for the fast information analysis of multiple-site LFP, EEG and spike train recordings. BMC neuroscience, 10(1), 1-24.

Shannon, C. E. A mathematical theory of communication. Bell Syst. Tech. J. 27, 379–423 (1948).

Sussillo, D., & Abbott, L. F. (2009). Generating coherent patterns of activity from chaotic neural networks. Neuron, 63(4), 544-557.

Treves, A. and Panzeri, S. The upward bias in measures of information derived from limited data samples. Neural Comput. 7, 399–407 (1995).

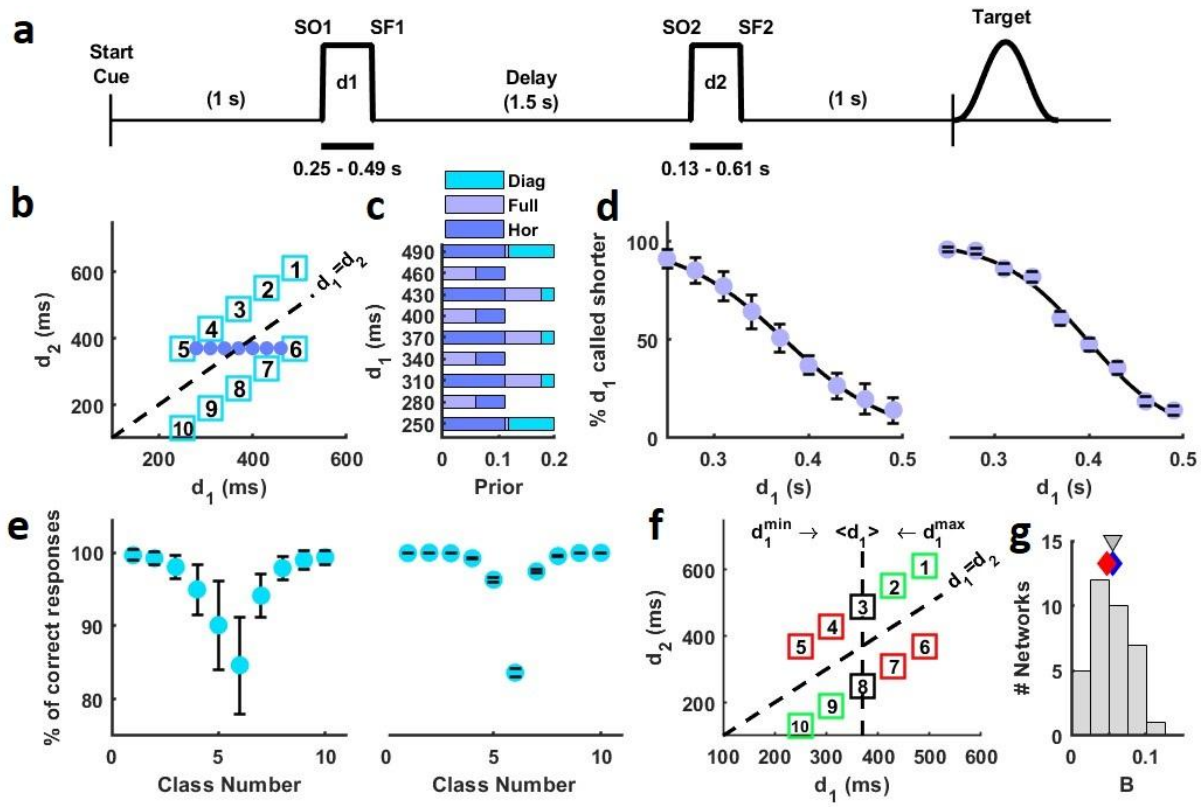

Fig. 1 | Task, set and curves. a. Timeline of the temporal interval discrimination task. An event called Start Cue indicates the beginning of the trial. A second later, two time intervals (d_1 and d_2) are presented sequentially separated by a delay period of 1500 ms. A second after d_2, a target (given by a beta distribution) is presented. **b**. Distribution set of possible pairs (d_1 and d_2). All classes of this distribution set, excluding the central class (370,370) ms, were used in network training. There are three block types: horizontal block (d_2=370 ms), diagonal block (classes labeled from 1 to 10) and full block formed by the previous two. Note that classes number 5 and 6 are present in the three blocks. Network test was carried out by choosing one of these three blocks. **c**. Prior distribution of d_1 per block. **d. Left**. Averaged psychometric curve over 35 networks. It shows that networks learned the task well. This curve was built using the horizontal classes of the full block. Error bars are the standard deviation over networks. **d. Right**. Psychometric curve for example network 1. Error bars were computed with 1000 bootstrap resamples. Black lines in both psychometric curves are the sigmoidal fits of the behavioral data. **e. Left**. Averaged discrimination curve over 35 networks. It shows the percentage of correct test trials with classes of the diagonal block. Its purpose is to analyze the contraction bias. Classes number 4, 5, 6 and 7 are handicapped by contraction bias. Error bars are the standard deviation over the 35 networks. **e. Right**. Discrimination curve for example network 1. Error bars were computed with 1000 resamples in a bootstrap process. **f**. Due to the contraction bias, the first stimulus is perceived contracted towards the mean of its distribution. This bias increases the difficulty of red classes while green classes are favored. Black classes (3 and 8) are not affected by contraction bias. **g**. Histogram of networks phenomenological bias (**B**) computed using Eq. 14. (Methods). Gray triangle represents the mean of the distribution (0.055). Blue diamond symbolizes the **B** of the example network 1 while red diamond refers to the **B** of the example network 2.

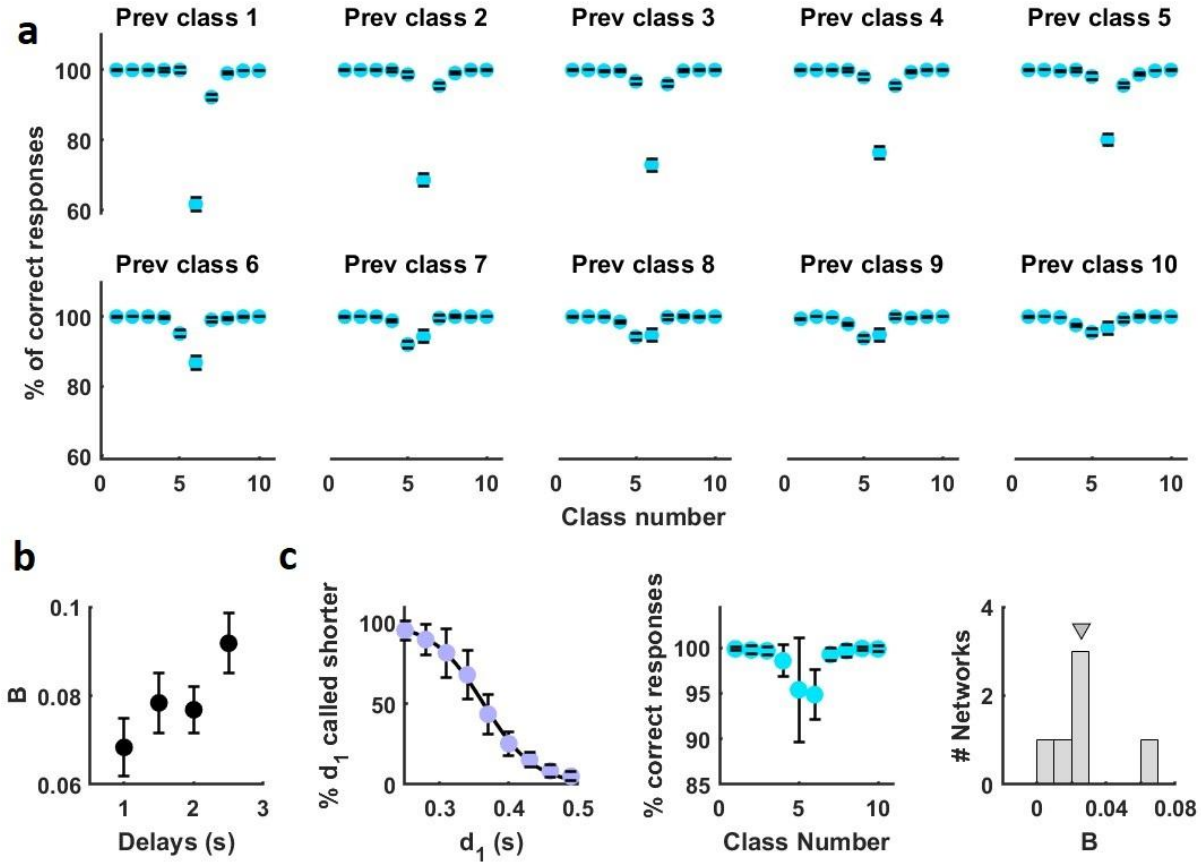

Fig. 2 | Effect of previous class on performance, simulation training with several delay-period durations and simulation resetting the neural activity. a. Accuracy conditioned over each of the ten previous classes. These discrimination curves correspond to example network 1 using the diagonal block. Error bars were computed with 1000 bootstrap resamples. **b**. Phenomenological bias (**B**) vs. delay-period duration for an example network trained by randomly interleaving four different delay durations (1000, 1500, 2000 and 2500 ms) and tested with four blocks of test trials, each one with a fixed delay duration, using the classes of diagonal block. Error bars were computed with a bootstrap procedure of 100 resamples. **c**. Simulation with a neural activity reset to a random initial state in all training and test trials (see Methods). 7 networks were trained and tested with this simulation. **Left**: Averaged psychometric curve over networks (see Fig. S2-2a for an example network). The classes used in the test trials were horizontal and collected from the full block. Black line refers to a sigmoidal fit. **Middle**: Averaged discrimination curve over networks (see Fig. S2-2b for an example network) using classes from the diagonal block. Error bars of both curves refer to the standard deviation over networks. **Right**: Histogram with the value of B for all 7 reset networks. Gray triangle indicates the histogram's mean (0.0253).

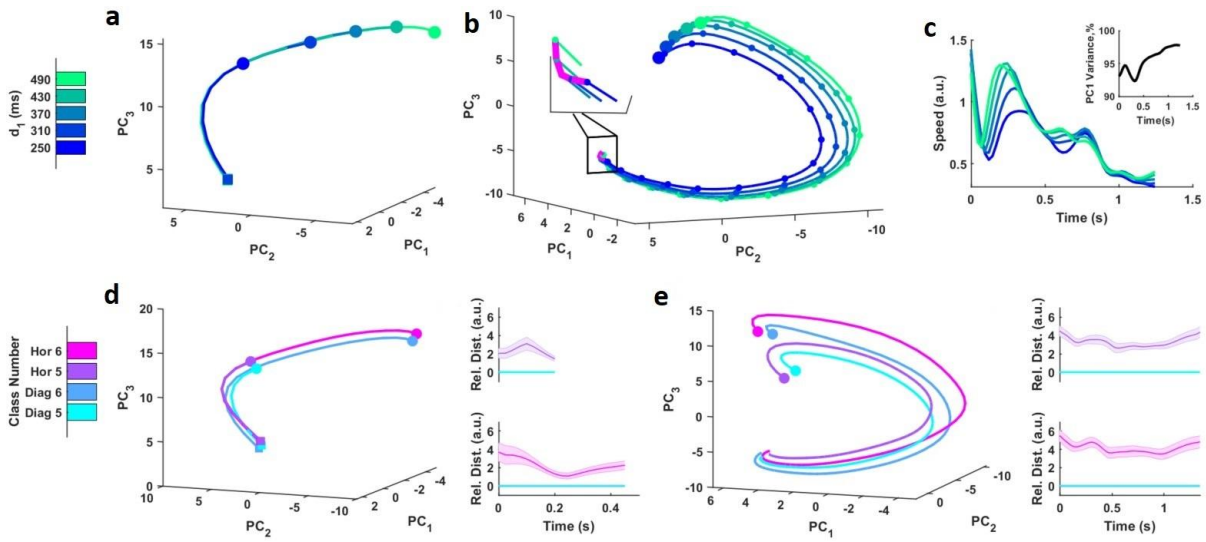

Fig. 3 | State space and contexts for example network 1. **a-c**. State space built with the first three PCs computed from the trial averaged firing rates using the diagonal block. These firing rates were calculated by sliding a 150 ms time window at 25 ms intervals and using a sub-population of 234 neurons with tuning during the delay period (see Methods). **a**. Neural trajectories of each d_1 value during the presentation of the first stimulus (firing rates were aligned to SO1). Squares indicate SO1. Circles indicate each SF1. **b**. Neural trajectories during the delay period (firing rates were aligned to SF1). Big circles indicate each SF1. Small circles indicate neural states at 100 ms increments. Note that the distance between consecutive circles reflects speed. **Inset** shows a storage quasi-attractor (magenta line) formed at the end of the delay period. **c**. Speed of neural trajectories in state space versus time during the delay period. Speed is the norm of the velocity vector. This vector is computed as the covered distance in 150 ms time samples separated 25 ms for each of the space components. **Inset** shows the percentage of variance explained by the first PC sliding a 150 ms window at 25 ms intervals. The variance approaches 100% as the delay ends. **d-g**. Comparison of diagonal and horizontal contexts using the classes 5 (d_1=250 ms) and 6 (d_1=490 ms), which are present in both blocks. Both context firing rates were computed sliding a 150 ms time window at 25 ms intervals and using 225 neurons. These neurons correspond to those ones with tuning during the delay period using the full context, which contains the diagonal and horizontal contexts. **d**. **Left**. State space trajectories of classes 5 and 6 during the presentation of the first stimulus. Squares indicate SO1. Circles indicate each SF1. **Right**. Relative distances, during the presentation of the first interval, of the classes 5 (**upper**) and 6 (**bottom**). PCA was made in order to explain at least 90% of variance. Thus, in class 5 panel the number of PCs used was 2 while in class 6 panel was 3 PCs. Shadings indicate the median ± 95% CIs computed from a bootstrap procedure with 100 resamples. **e. Left**. State space of classes number 5 and 6, during the delay period. Circles indicate each SF1. **Right**. Relative distances, during the delay period, of classes 5 (**upper**) and 6 (**bottom**). 4 PCs needed to explain more than 90% of variance. Shadings computed as in **d, right**.

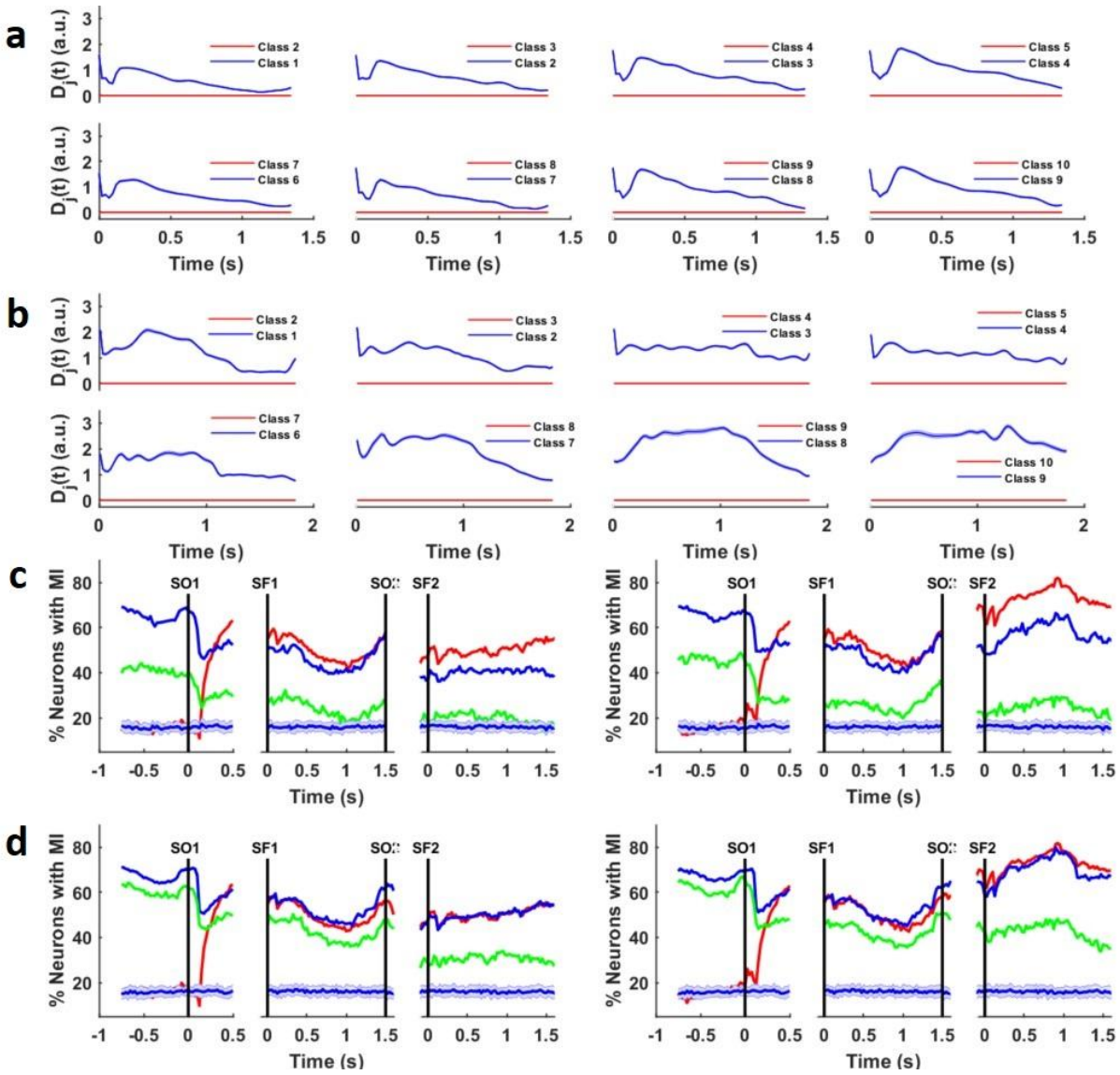

Fig. 4 | Mutual information. a-b. Relative distances, $D_j(t)$, between pairs of consecutive classes. The index j refers to the pair of classes of a given panel. Shadings indicate the median ± 95% CIs computed from a bootstrap procedure with 100 resamples. **a**. Relative distances during the delay period (aligned to SF1). **b**. Relative distances during the ITI period (aligned to SF2). **c-d**. Mutual information from trial n (red), from trial n-1 (blue) and from trial n-2 (green). **c.** Percentage of neurons with significant mutual information about the first stimulus conditioning on correct trials and on either choice d_1 shorter (**left**) or choice d1 longer (**right**). **d.** Percentage of neurons with significant mutual information about the second stimulus, conditioning on correct trials and on either choice d_1 shorter (**left**) or choice d1 longer (**right**). Horizontal shadings refers to the mean and standard deviation of the percentage of neurons with significant information computed from shuffled data (Methods).

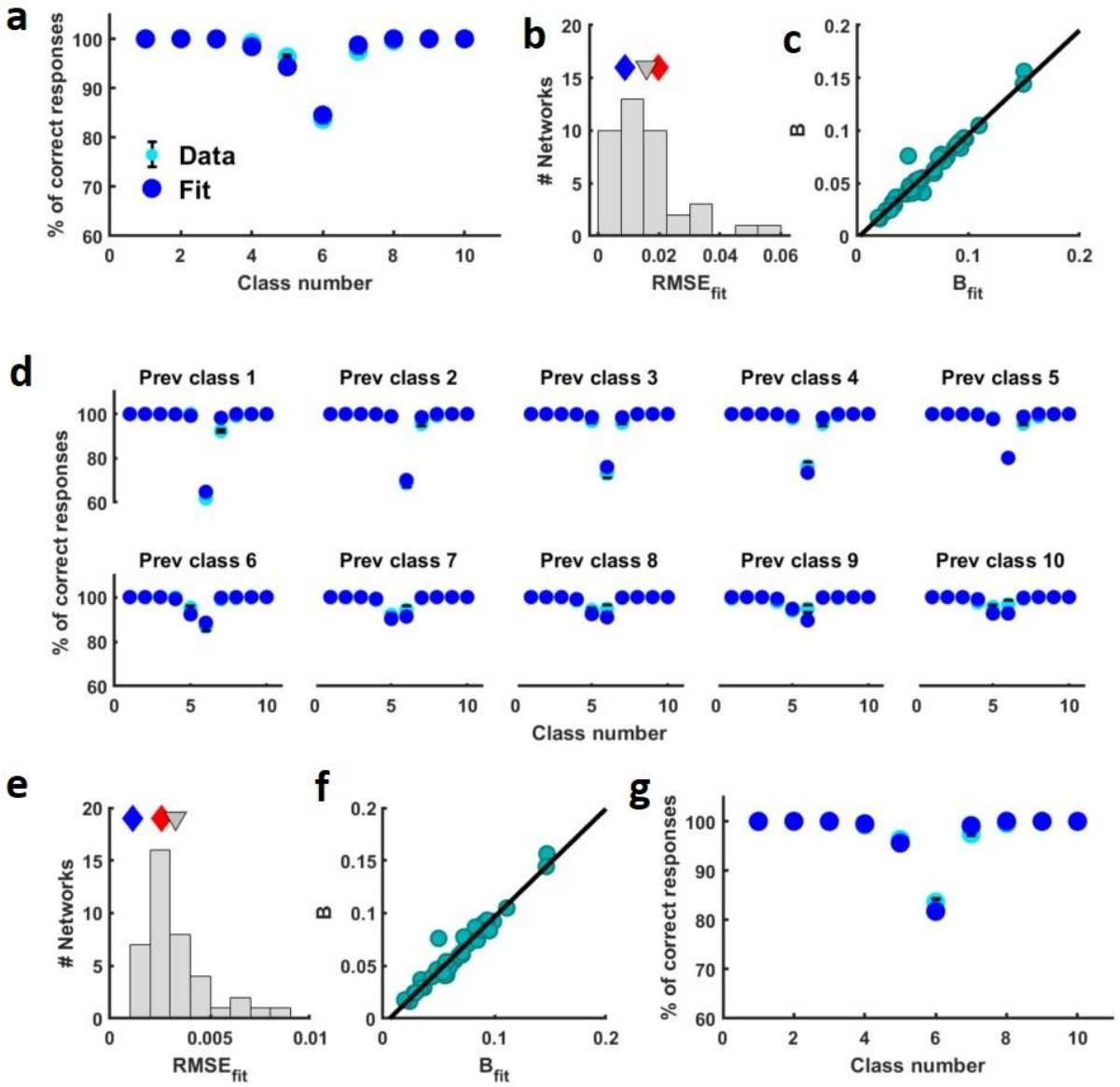

Fig. 5 | Fits with models. a-c. Fit with the simplest model version, m=0. **a**. Fit of the discrimination curve. Cyan dots with error bars correspond to the behavioral data of example network 1 in Fig. 1e right. Dark blue circles are the fit. **b**. Histogram of RMSEs (root mean square error from the difference between the behavioral data performance and the model-fitted performance) over 35 networks. Gray triangle indicates the histogram's mean (0.016). Blue diamond indicates the RMSE value of network 1 and red diamond refers to network 2. **c**. Correlation over 35 networks between the phenomenological bias (**B**) and the bias computed from the model fit ($\mathbf{B_{fit}}$) (correlation=0.98; p-value=5.6e-27). Both **B** and $\mathbf{B_{fit}}$ were computed with Eq. 14 (Methods). **d-g.** Fit with model version m=1. **d**. Fit of the ten accuracy curves conditioned over each possible preceding diagonal class. Cyan circles correspond to behavioral data of example network 1 in Fig. 2a. **e**. Histogram of RMSEs of performance fits over 35 networks. The histogram's mean is equal to 0.0033. **f**. Correlation between **B** and $\mathbf{B_{fit}}$ (correlation=0.97; p-value=4.2e-26). **g.** Marginal discrimination curve fitted for network 1, obtained by marginalizing over the first interval in the previous trial.

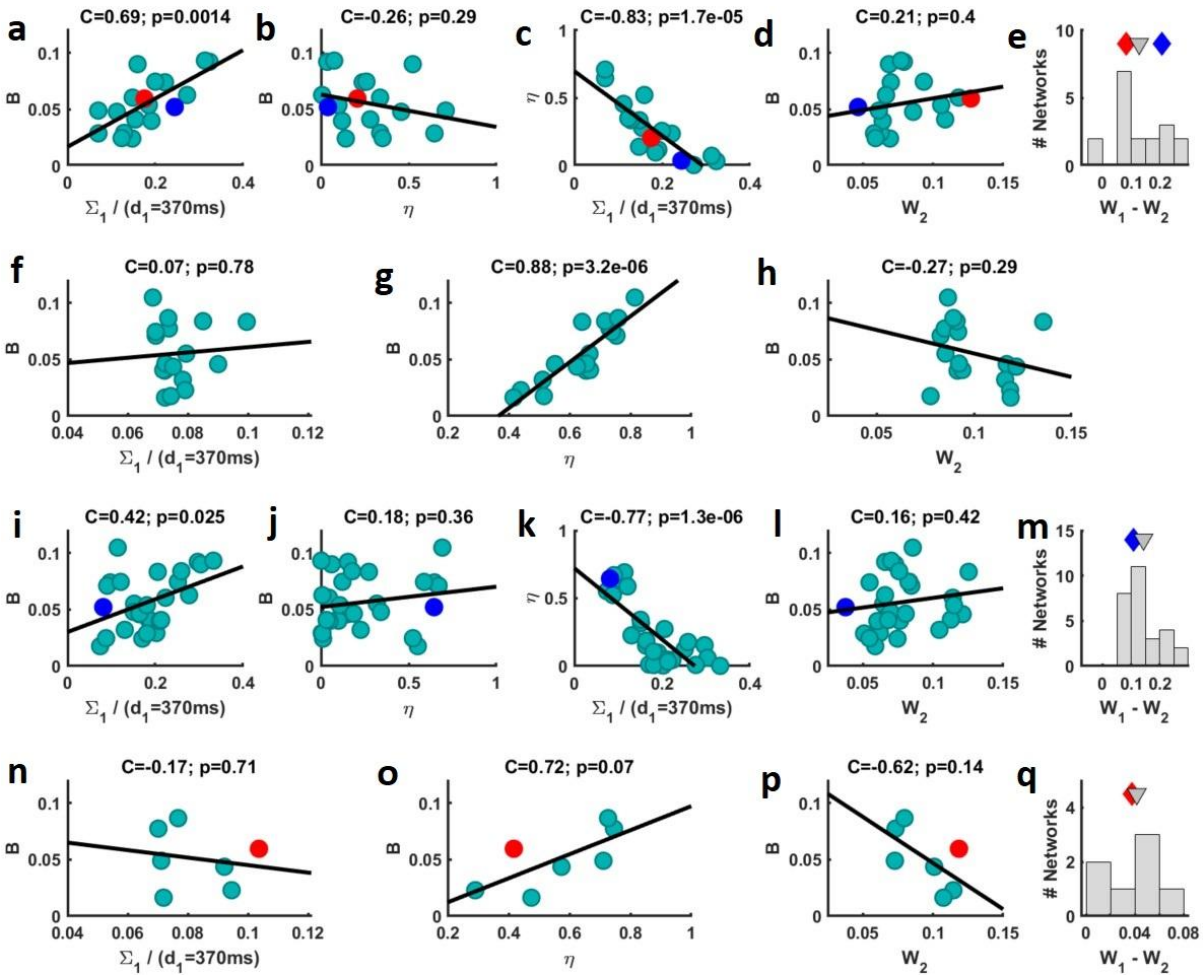

Fig. 6 | Correlations between phenomenological bias (B) and parameters of model versions m=0 and m=1. a-e. Correlations of 18 networks classified as positive by model version m=0. **a**. Correlations between **B** and the total variance of observation probability (Σ_1) for the central first stimulus (see Fig. S6-1a). **b**. Correlation between **B** and the history parameter (η). **c**. Correlation between η and Σ_1 for the central first stimulus (see Fig. S6-2). **d**. Correlation between **B** and the uncertainty in the observations of the second stimulus (W_2). **e.** Histogram of the difference between both uncertainty (W_1 - W_2). Gray triangle indicates the histogram's mean (0.1281). **f-h.** Correlations of 17 networks classified as negative by model version m=0. **f**. Correlation between **B** and Σ_1 for the central first stimulus (see Fig. S6-1b). **g**. Correlation between **B** and η. **h**. Correlation between **B** and W_2. **i-m.** Correlations of 28 networks classified as positive by model version m=1. **i**. Correlations between **B** and Σ_1 only for the central value of current and previous first stimulus (see Fig. S6-3). **j**. Correlation between **B** and η. **k**. Correlation between η and Σ_1 only for the central value of current and previous first stimulus (see Fig. S6-5). **l**. Correlation between **B** and W_2. **m**. Histogram of the difference between both uncertainty (W_1 - W_2). The histogram's mean is equal to 0.1433. **n-q.** Correlations of 7 networks classified as negative by model version m=1. **n**. Correlation between **B** and Σ_1 for both the current and the previous central first stimulus (see Fig. S6-4). **o**. Correlation between **B** and η. **p**. Correlation between **B** and W_2. **q.** Histogram of the difference between both uncertainty (W_1 - W_2). The histogram's mean is equal to 0.0417. Blue markers refer to the example network 1 and red markers refer to the example network 2. C indicates the correlation value and p is the p-value.

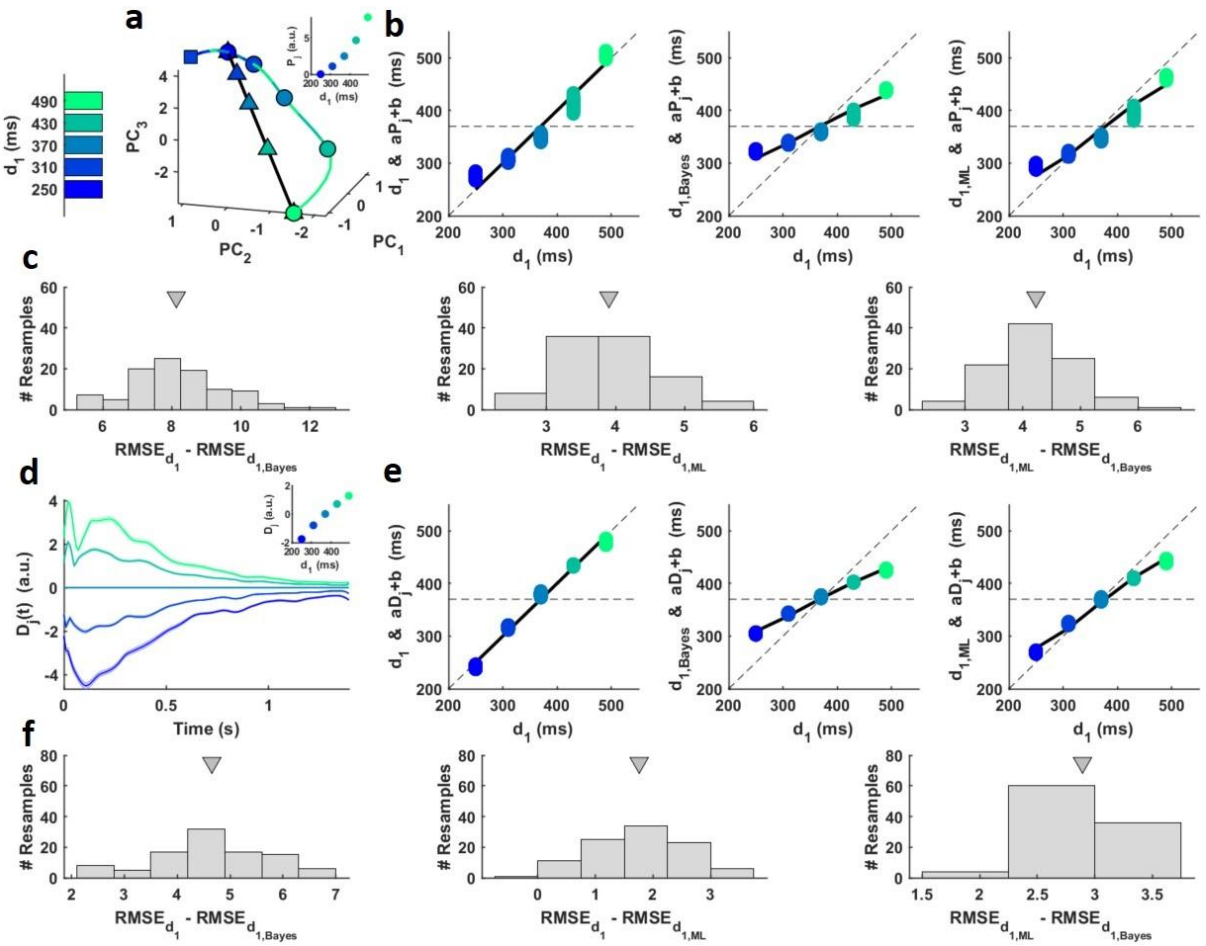

Fig. 7 | Comparison of state space geometry with behavior using model version m=1 for example network 1. a. Neural trajectories during the presentation of the first stimulus and projections, P_j, (triangles) of each SF1 state (circles) onto the encoding axis, **u** (black line). State space was built with the first 4 PCs which explain 97% of variance. Squares represent the beginning of the figure, located within the first stimulus. **Inset** shows the euclidean distances between consecutive projections as a function of d_1. **b**. Fit of a linear regression of the projections (aP_j + b) to the true d_1 (**left**), the Bayesian estimator (**middle**) and the ML estimator (**right**) as a function of d_1. The colored dots show multiple estimates of the regression derived from bootstrapping (n=100). The solid lines indicate the true d_1 (**left**), the Bayesian estimator (**middle**) and the ML estimator (**right**) as a function of d_1. **c**. Histograms on shufflings of the difference in RMSE between the real d1 hypothesis and both estimated d1 hypotheses. Gray triangle indicates the mean of each histogram **d**. Relative distance, $D_j(t)$, between each trajectory j=1-5 and the reference trajectory d_1=370 ms, during the delay period. Neural trajectories embedded in a state space built with 6 PCs which explain 92% of variance. Shadings indicate the median ± 95% CIs computed from a bootstrap procedure with 100 resamples. **Inset** shows the temporal means of distances as a function of d_1. **e**. Fit of a linear regression of the mean relative distances (aD_j + b) to the true d_1 (**left**), the Bayesian estimator (**middle**) and the ML estimator (**right**) as a function of d_1 (same format as in **b**). **f.** Histograms on shufflings of the difference in RMSE between the real d_1 hypothesis and both estimated d_1 hypotheses. Firing rates, averaged over test trials with classes from diagonal block, of 234 tuning neurons were calculated aligning to SO1 and using an analysis time window of 100 ms and 10 ms of slide.

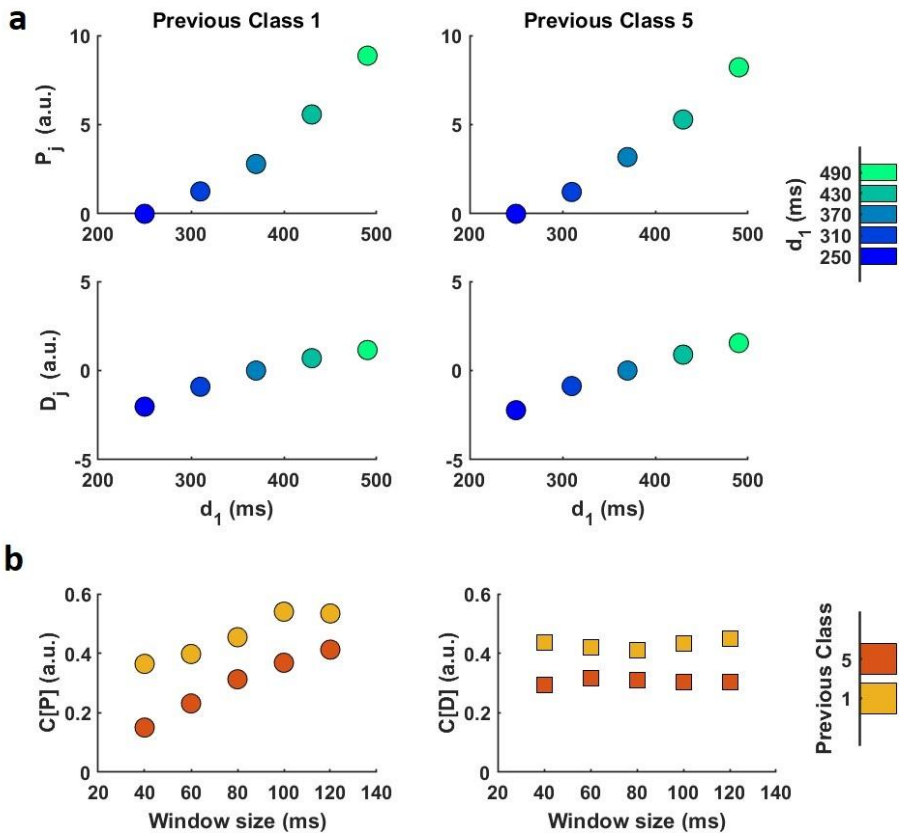

Fig. 8 | Effect of the previous class on the geometry in state space for example network 1. a. Projections, P_j, of each SF1 state onto the encoding axis as function of d_1 (**upper**) conditioning on previous class 1 (**left**) and previous class 5 (**right**). Temporal mean of relative distances, D_j, as function of d_1 (**bottom**) conditioning over previous class 1 (**left**) and previous class 5 (**right**). The analysis time window used had a size of 100 ms and a slide of 10 ms. 6 PCs were used in order to build the state space, explaining the 97% (projections conditioning on previous class 1), 98% (projections, previous class 5), 86% (distances, previous class 1) and 89% (distances, previous class 5) of variance. **b**. Compression of projections, C[P], (**left**) and mean relative distances, C[D], (**right**) as function of size of the analysis time window (40, 60, 80, 100 and 120 ms). These compressions were computed using Eqs. 20-21 (Methods). If compression between the shortest d_1's (P_3 - P_1) was higher than the compression between the highest d_1's (P_5 - P_3), then C[P] was indicated as a square. Otherwise C[P] was indicated as a circle. Same in case of C[D].

Supplemental Information

Emergent perceptual biases from state-space geometry in spiking recurrent neural networks trained to discriminate time intervals

Luis Serrano-Fernandez, Manuel Beiran and Nestor Parga

Supplementary Texts

Supplementary Text 1: General version of normative Bayesian model. Both the mean and variance for a general version of the used Bayesian model, in which the transient consists of the previous m trials (j=0,1, …, m) is given by the following equations:

$$\mathcal{M} = \sum_{j=0}^{m} \eta^j d_{1,n-j} + \eta\left[\eta^j \langle d_1 \rangle - \sum_{j=0}^{m} \eta^j d_{1,n-j}\right] \tag{58}$$

$$\Sigma_1^2 = \sum_{j=0}^{m} \eta^{2j} (1-\eta)^2 W_1^2 d_{1,n-j}^2 + \eta^{2(m+1)} \Sigma_{1,st}^2 \tag{59}$$

where

$$\Sigma_{1,st}^2 = \frac{1-\eta}{1+\eta}\left(W_1^2 \langle d_1^2 \rangle + \sigma_{d_1}^2\right). \tag{60}$$

Supplementary Tables

Model	W_1	W_2	η	γ_1	γ_2	RMSE
m=1 without $d_{2,n-1}$	0.15	0.038	0.64	0.45	0.084	0.0011
m=1 with $d_{2,n-1}$	0.19	0.048	0.57	0.46	0.22	0.0012

Supplementary Table S1: Fitted parameter and RMSE values from the fit of behavioral data of the example network 1 using the version model m=1 including and not including the previous second interval, $d_{2,n-1}$. W_1 and W_2 are the uncertainties in the observations of the first and second stimulus, respectively. η is the history parameter. γ_1 and γ_2 are the two last parameters used in model version m=1 to include a win-stay and loss-shift strategy (WSLS) in the decision making (see Methods). RMSE is the root mean square error from the difference between the behavioral data performance and the model performance so it reflects the quality of the model in order to fit the data.

$Network$	$BIC\ with\ \mu$	$BIC\ without\ \mu$	ΔBIC	$AIC\ with\ \mu$	$AIC\ without\ \mu$	ΔAIC
1	4224	4171	53	4205	4159	46
2	1047	1039	8	1028	1026	2
3	900	872	28	881	859	22
4	2160	2157	3	2141	2145	-4
5	1523	1515	8	1504	1502	2
6	6981	6969	12	6963	6956	7
7	2129	2033	96	2110	2021	89

Supplementary Table S2: Comparison of the model that requires knowledge of the mean of the intervals with the model that does not require it. The weight of this knowledge is given by the parameter μ**.** For this, 7 networks trained and tested with reset were used. The criteria AIC and BIC tend to favor the model that does not require knowledge of the mean of the intervals.

Supplementary Figures

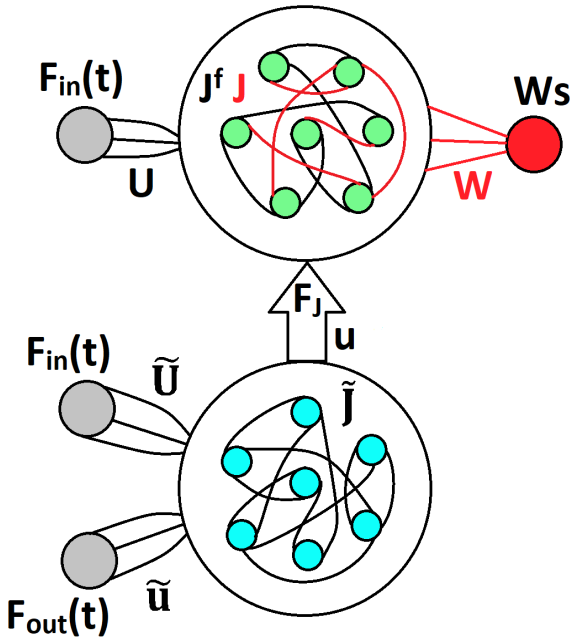

Fig. S1-1 | Full-FORCE scheme. The top network is a spiking RNN of N recurrently connected LIF neurons (green circles) that receives an external input, F_{in}, (gray circle) through the U synapses and generates an output signal, Ws, (red circle) which is returned through the synapses W. Black connections (U and J^f) are non-plastic recurrent synapses chosen randomly while red connections (hidden weights J and output weights W) are plastic recurrent synapses which are adjusted during training epoch. The bottom network, called target-generating network, is a continuous network of Ñ firing-rate neurons (cyan circles) recurrently connected through non-plastic synapses, $\tilde{J}$ (black lines). This network generates the auxiliary function , F_J, needed to train the intrinsic parameters in J of sRNN. Thus, this network receives as inputs both F_{in} and F_{out} through the non-plastic synapses $\tilde{U}$ and $\tilde{u}$, respectively, and sends the auxiliary function to sRNN through the u connections.

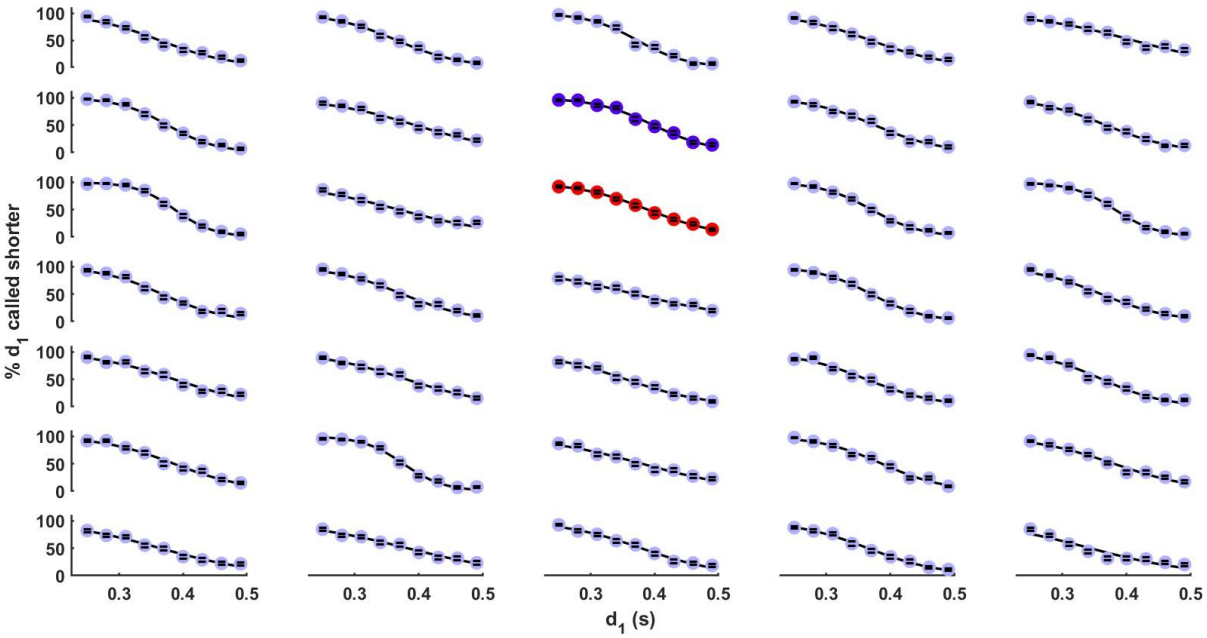

Fig. S1-2 | Psychometric curves are heterogeneous across the 35 trained networks. This type of curve is built using the horizontal classes of the full block. Error bars were computed with 1000 bootstrap resamples. Black lines in psychometric curves refer to the sigmoidal fits of the simulated behavioral data. Panel with dark blue circles is associated with the behavioral data of example network 1. Panel with red circles corresponds to example network 2.

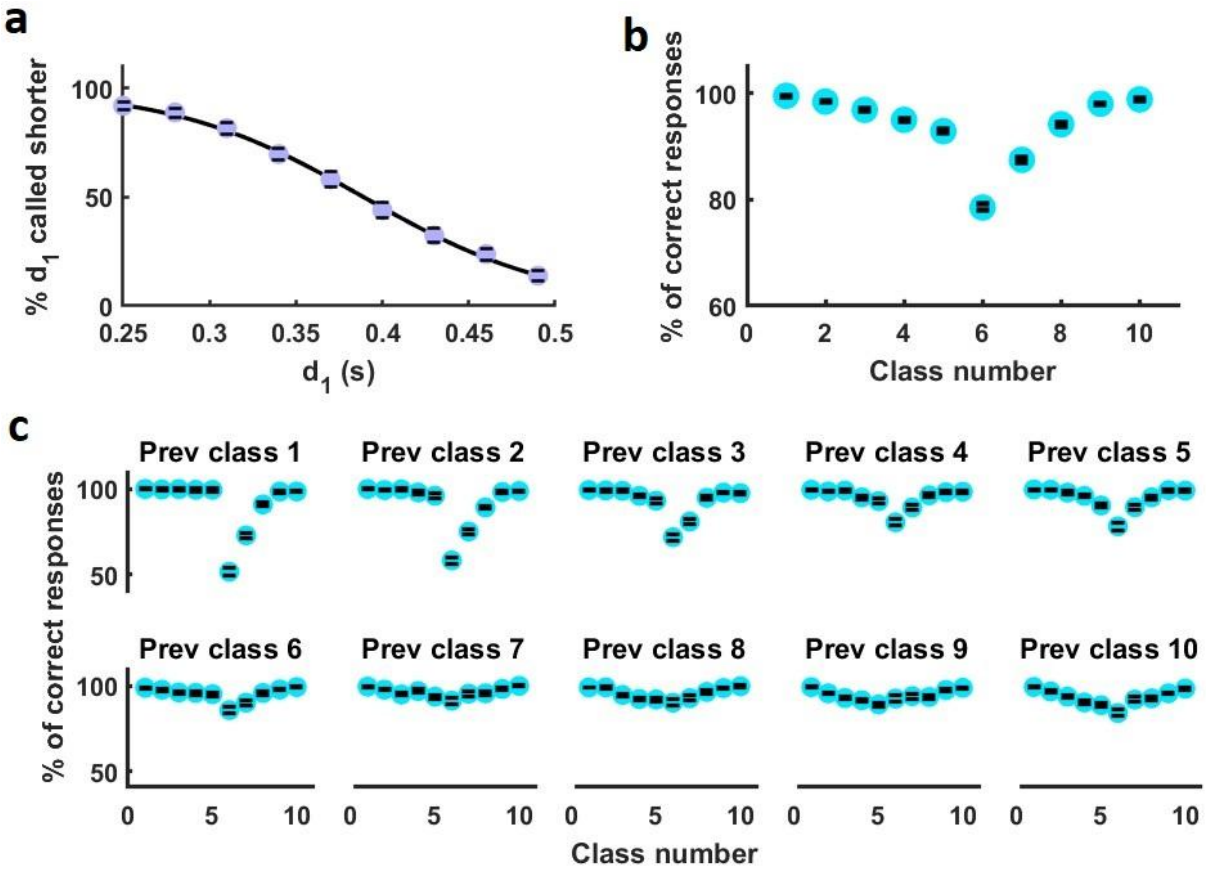

Fig. S2-1 | Simulated behavioral data from the example network 2. a. Psychometric curve using the horizontal classes of full block. Black line refers to the sigmoidal fit of the simulated data. **b**. Discrimination (accuracy) curve which shows the percentage of correct test trials with classes of the diagonal block. **c**. Ten accuracy curves, each one conditioned over one of the ten previous classes. Error bars were computed with 1000 bootstrap resamples.

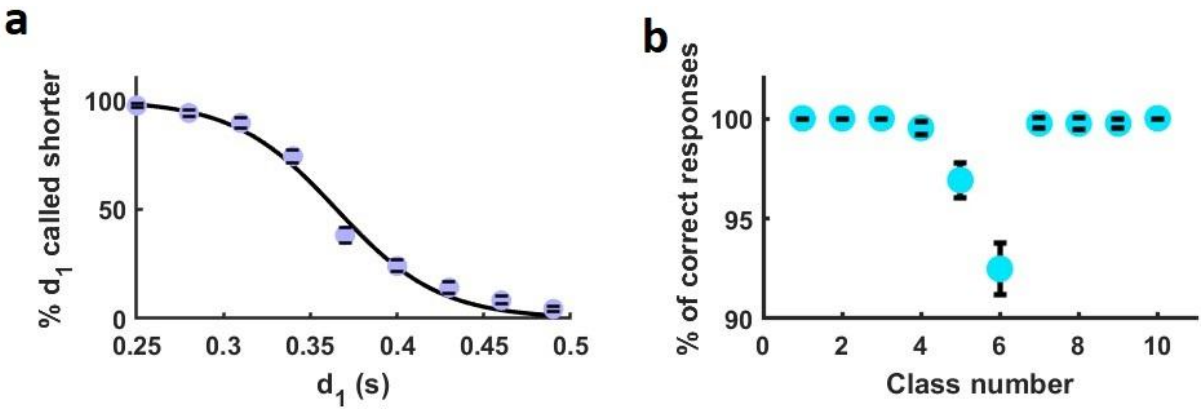

Fig. S2-2 | Simulated behavioral data from an example network trained and tested resetting the neural activity at the end of the trial. This reset takes place 2000 ms before the end of each training and test trial. **a**. Psychometric curve. Black line refers to a sigmoidal fit. **b**. Discrimination curve. Error bars were computed with 1000 bootstrap resamples.

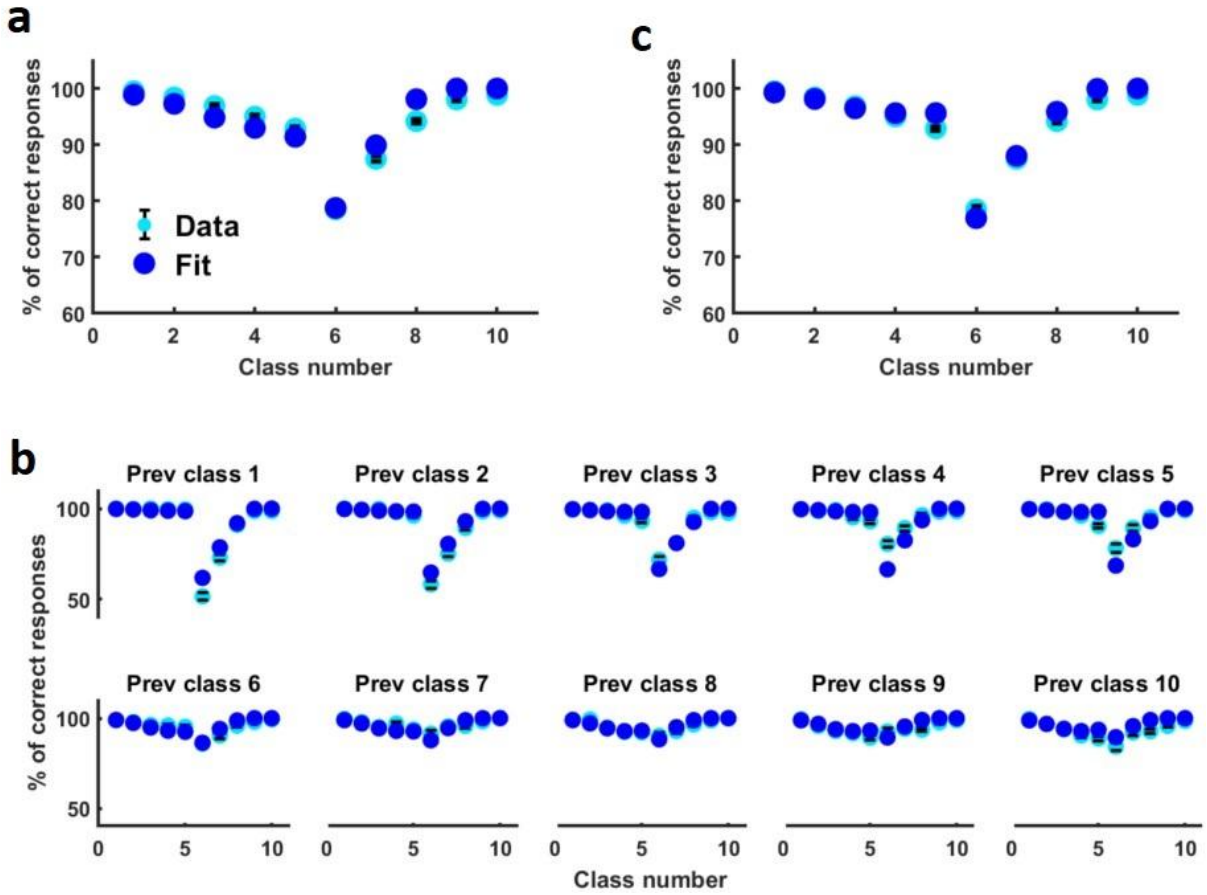

Fig. S5-1 | Fitting the simulated behavioral data of example network 2. a. Fit of the discrimination curve using the model version m=0. Cyan circles with error bars correspond to the simulated behavioral data of example network 2 in Fig. S2-1b. Dark blue circles are the fit. **b**. Fit of the ten accuracy curves conditioned over each possible preceding diagonal class and using the model version m=1. Cyan circles correspond to behavioral data of example network 2 in Fig. S2-1c. **c.** Marginal discrimination curve fitted for example network 2 with model version m=1, obtained by marginalizing over the class in the previous trial.

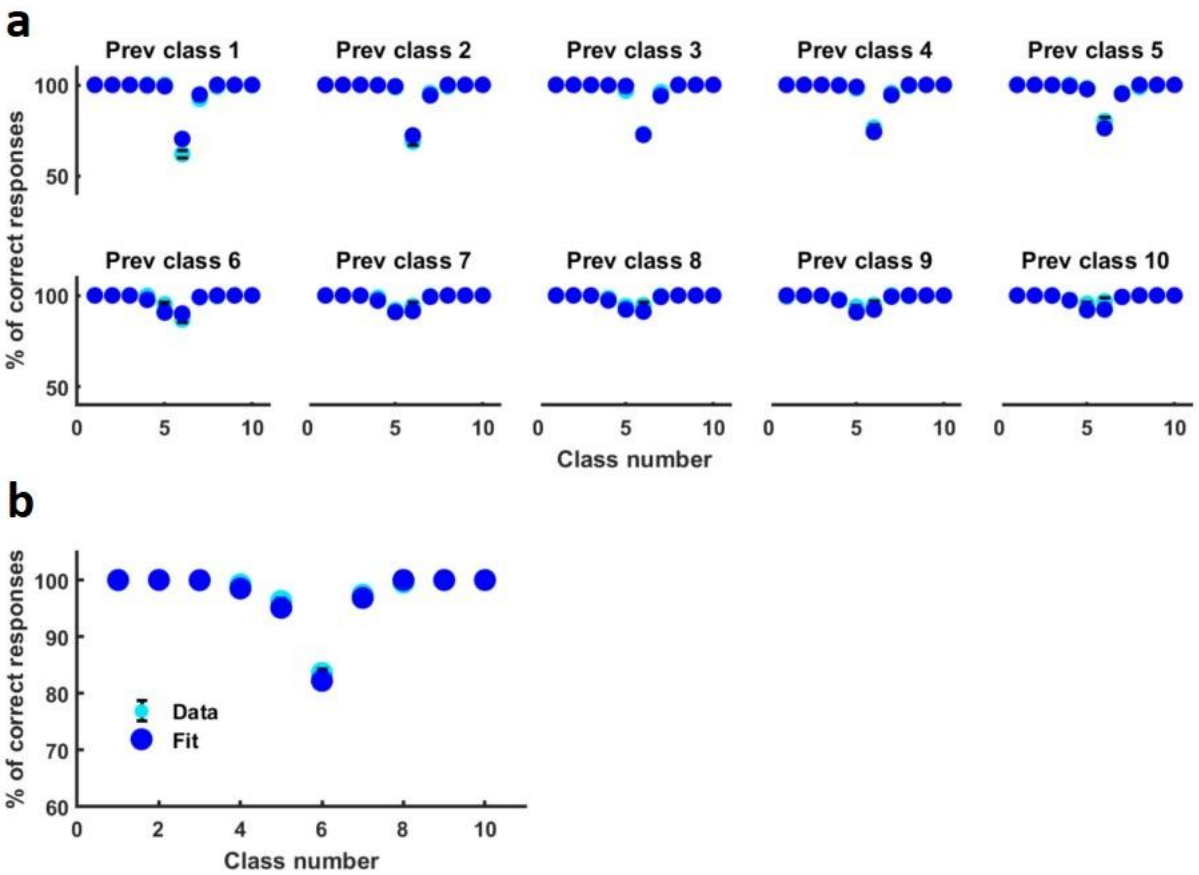

Fig. S5-2 | Fitting the behavioral data of example network 1 with the version model which includes both the extraction of the previous first interval and the contribution of the previous second interval. a. Fit of the ten accuracy curves conditioned over each possible preceding diagonal class. Cyan dots with error bars correspond to the behavioral data of example network 1 in Fig. 2a. Dark blue circles refer to the fit. **b.** Marginal discrimination curve fitted, obtained by marginalizing over the class in the previous trial. Cyan circles correspond to behavioral data of example network 1 in Fig. 1e Right.

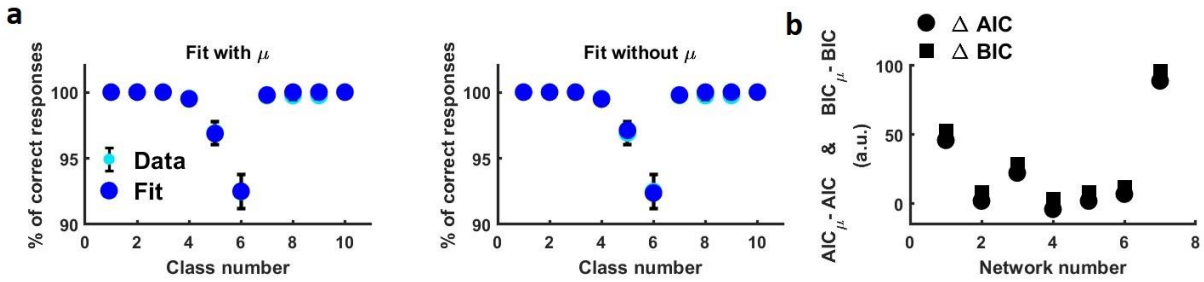

Fig. S5-3 | Evaluation of the hypothesis that the neural circuits have knowledge about stimulus statistics. a. Fitting the simulated behavioral data of an example network trained and tested resetting the neural activity (see Fig. S2-2b) using a model that includes a weight of the $<d_1>$, μ, as parameter (**left**) or using a model that does not include it (**right**). **b**. Difference between the AIC measure for the model with μ parameter and the AIC for the model without this parameter, versus network number (circles). Id. for the BIC measures (squares). Comparison of the two models tends to favor the model that does not require knowledge of $<d_1>$.

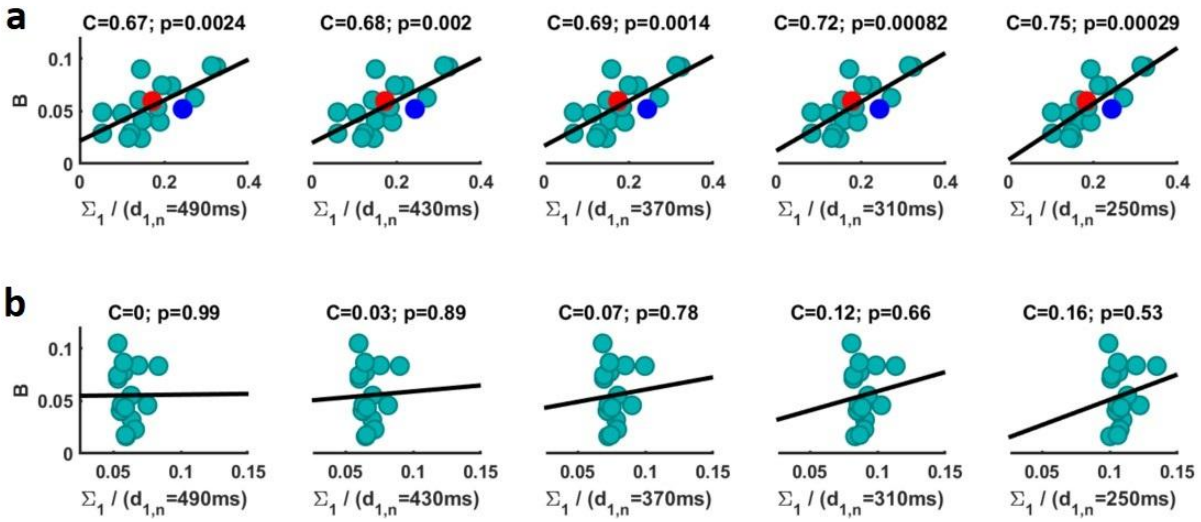

Fig. S6-1 | Correlations between phenomenological bias (B) and the total variance of observation probability (Σ_1) for all values of the first stimulus using model version m=0. a. Correlations correspond to those networks classified as positive by model version m=0. Blue circle indicates the example network 1 correlation. Red circle refers to example network 2. **b**. Correlations correspond to those networks classified as negative by model version m=0.

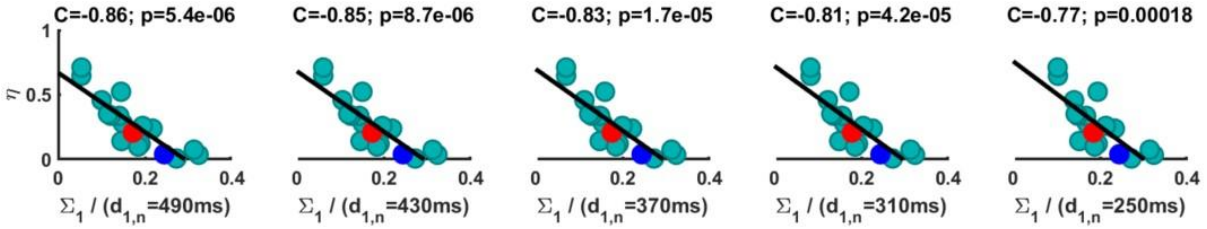

Fig. S6-2 | Correlations between the history parameter (η) and the total variance of observation probability (Σ_1) for the all values of the first stimulus using those networks classified as positive by model version m=0. Blue circle indicates the example network 1 correlation. Red circle refers to example network 2.

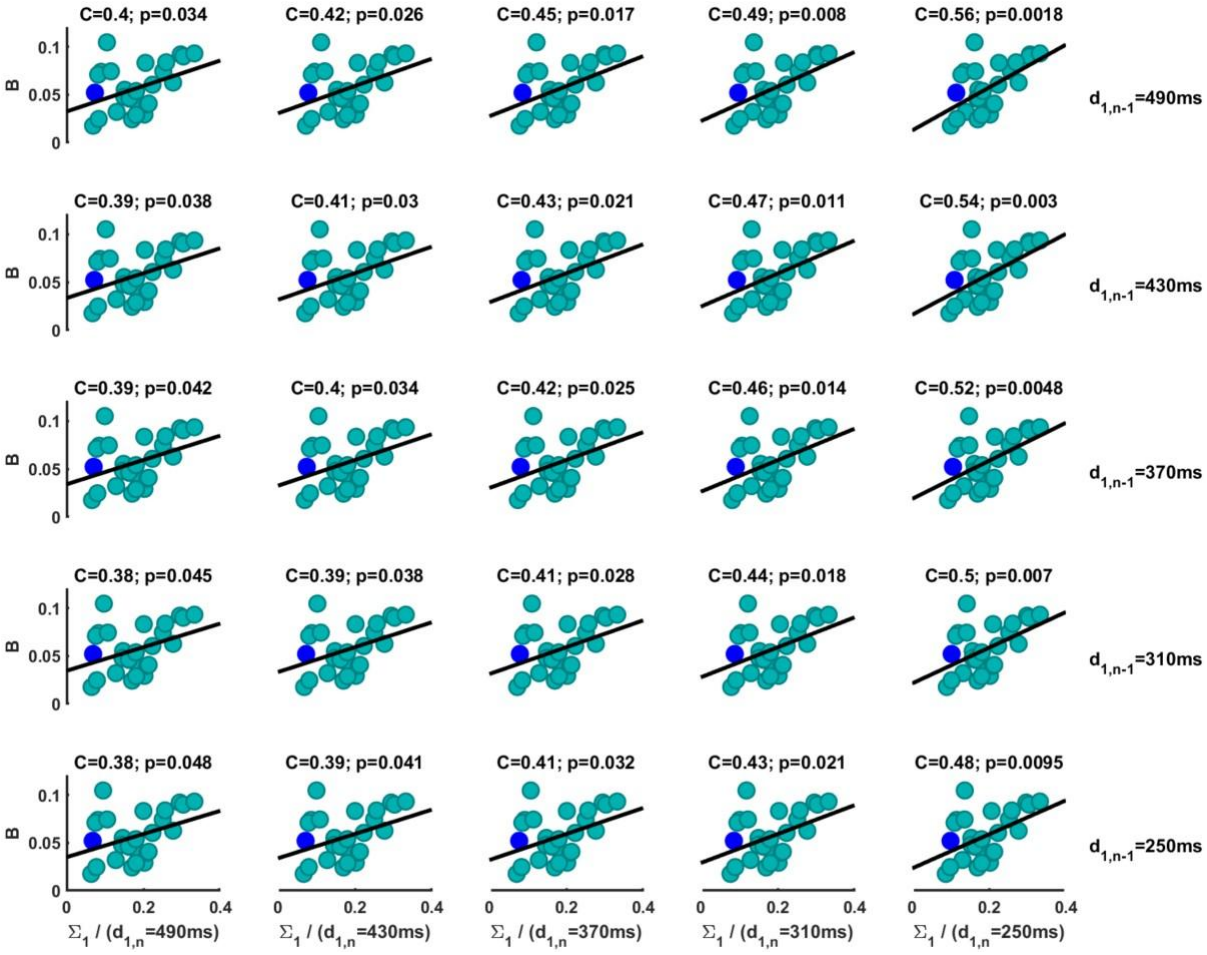

Fig. S6-3 | Correlations between B and the total variance of observation probability (Σ_1) conditioning over the first stimulus on the previous (n-1) and current (n) trial, using those networks classified as positive by model version m=1. Blue circles indicate the example network 1 correlation.

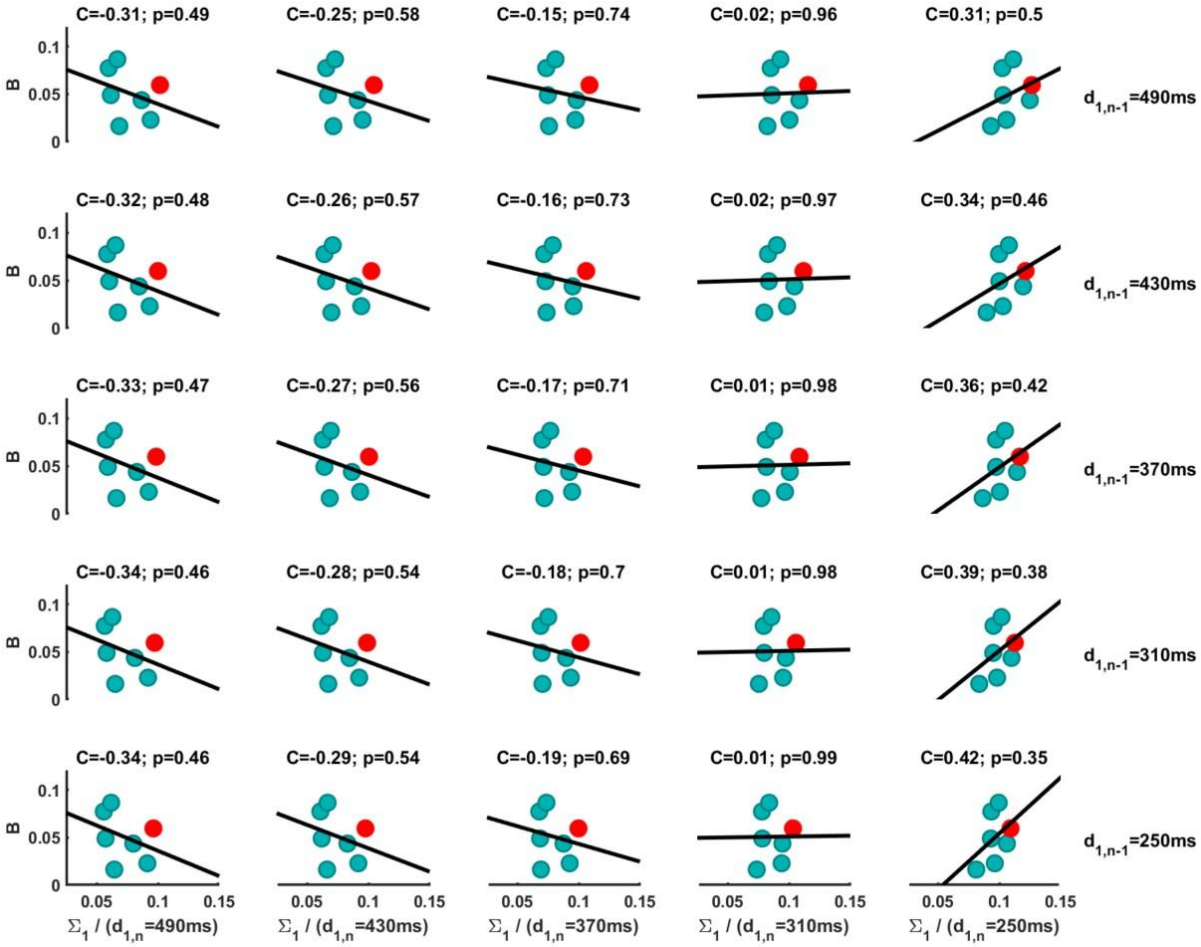

Fig. S6-4 | Correlations between B and the total variance of observation probability (Σ_1) conditioning over the first stimulus on the previous (n-1) and current (n) trial, using those networks classified as negative by model version m=1. Red circles indicate the example network 2 correlation.

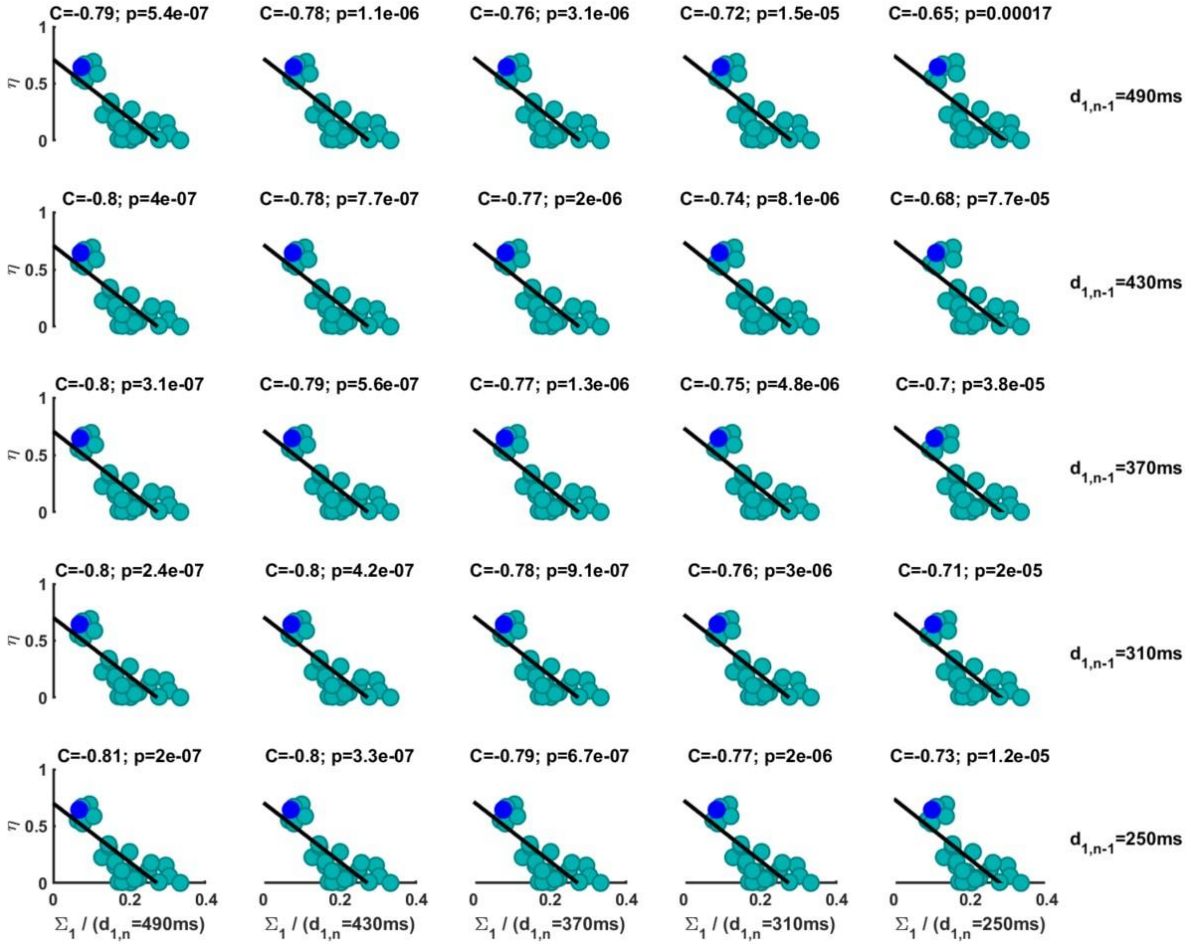

Fig. S6-5 | Correlations between the history parameter (η) and the total variance of observation probability (Σ_1) conditioning over the first stimulus on the previous (n-1) and current (n) trial, using those networks classified as positive by model version m=1. Blue circles indicate the example network 1 correlation.

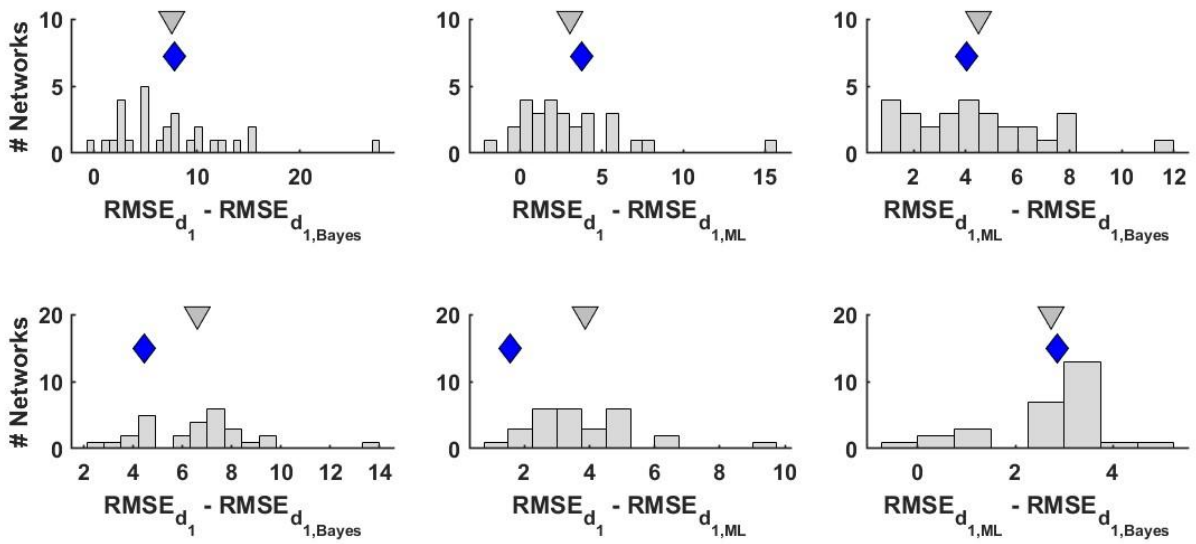

Fig. S7-1 | Histograms over networks classified as positive by model version m=1 from the comparison of state space geometry with behavior. Upper. Histograms of the difference in RMSE resulting from the fit of the true d_1 and both Bayesian and ML estimators with the linear expression that includes the projections ($aP_j + b$). Gray triangle indicates the mean of each histogram. Blue diamond refers to example network 1. **Bottom**. Histograms of the difference in RMSE resulting from the fit of the true d_1 and both Bayesian and ML estimators with the linear expression that includes the temporal mean of the relative distances ($aD_j + b$).

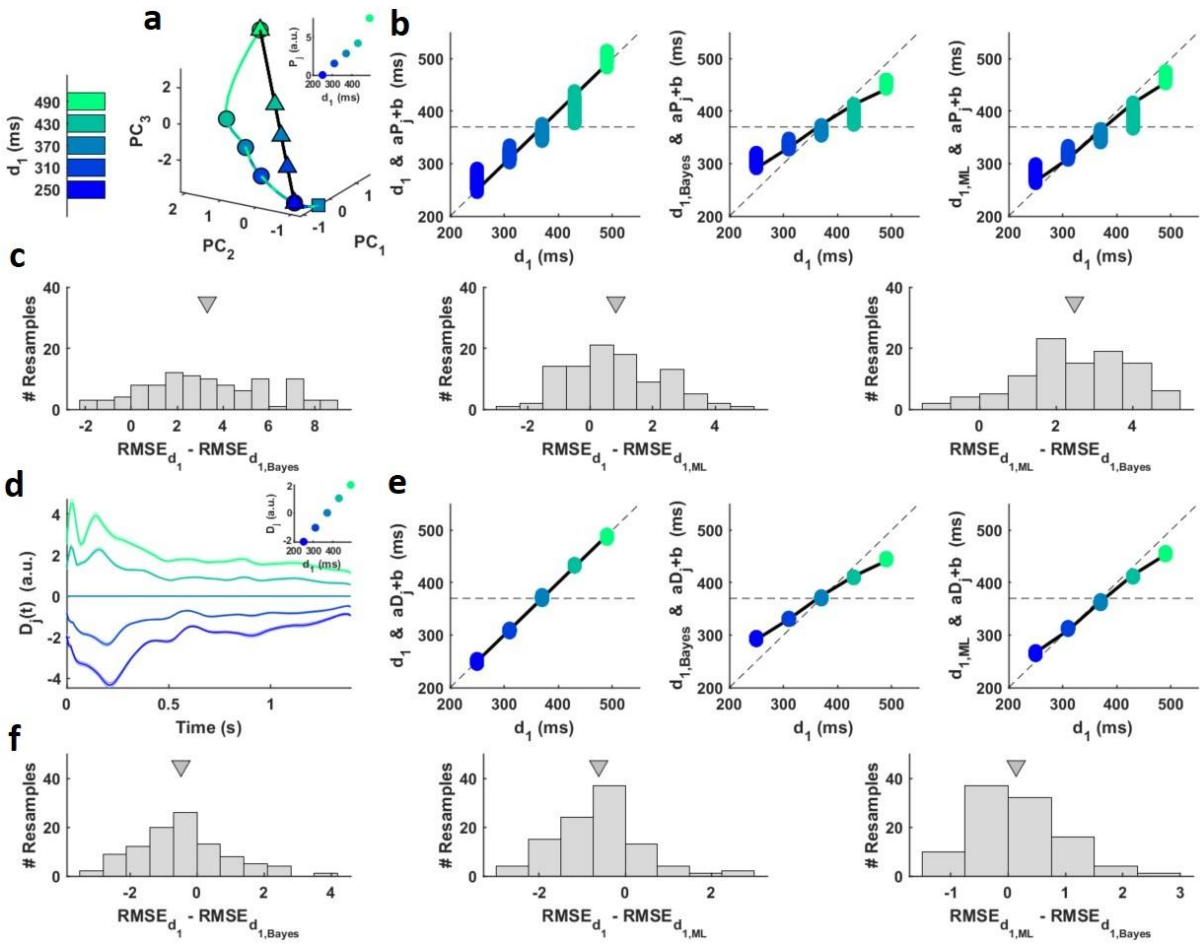

Fig. S7-2 | Comparison of state space geometry with behavior using model version m=1 for example network 2. a. Neural trajectories during the presentation of the first stimulus and projections, P_j, (triangles) of each SF1 state (circles) onto the encoding axis, **u** (black line). State space was built with the first 4 PCs which explain 95% of variance. Squares represent the beginning of the figure, located within the first stimulus. **Inset** shows the euclidean distances between consecutive projections as a function of d_1. **b**. Fit of a linear regression of the projections (aP_j + b) to the true d_1 (**left**), the Bayesian estimator (**middle**) and the ML estimator (**right**) as a function of d_1. The colored dots show multiple estimates of the regression derived from bootstrapping (n=100). The solid lines indicate the true d_1 (**left**), the Bayesian estimator (**middle**) and the ML estimator (**right**) as a function of d_1. **c**. Histograms on shufflings of the difference in RMSE between the real d1 hypothesis and both estimated d1 hypotheses. Gray triangle indicates the mean of each histogram **d**. Relative distance, $D_j(t)$, between each trajectory j=1-5 and the reference trajectory d_1=370 ms, during the delay period. Neural trajectories embedded in a state space built with 7 PCs which explain 91% of variance. Shadings indicate the median ± 95% CIs computed from a bootstrap procedure with 100 resamples. **Inset** shows the temporal means of distances as a function of d_1. **e**. Fit of a linear regression of the mean relative distances (aD_j + b) to the true d_1 (**left**), the Bayesian estimator (**middle**) and the ML estimator (**right**) as a function of d_1 (same format as in **b**). **f.** Histograms on shufflings of the difference in RMSE between the real d_1 hypothesis and both estimated d_1 hypotheses. Firing rates, averaged over test trials with classes from diagonal block, of 234 tuning neurons were calculated aligning to SO1 and using an analysis time window of 100 ms and 10 ms of slide.

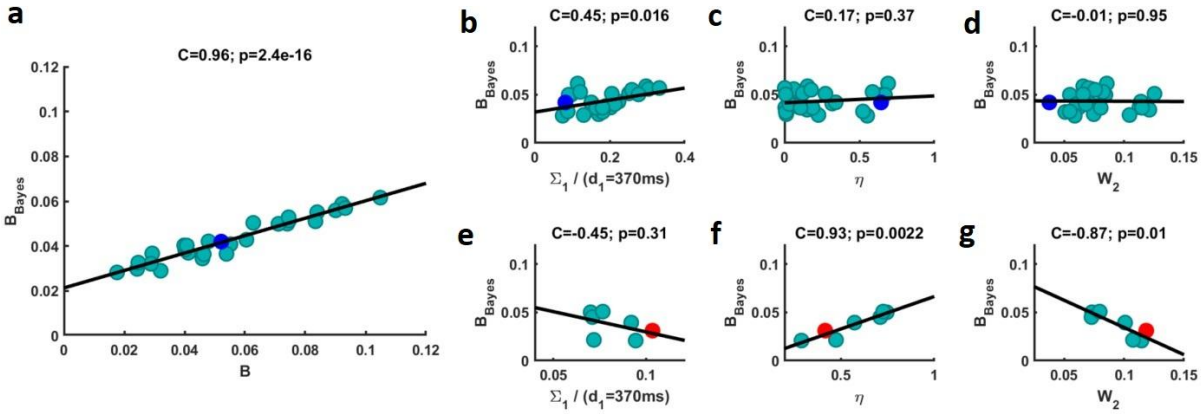

Fig. S7-3 | Correlations between the estimated Bayesian bias ($\mathbf{B_{Bayes}}$) derived from Bayesian model and parameters of model version m=1. a-d. Correlations of 28 networks classified as positive by model version m=1. **a.** Correlation between $\mathbf{B_{Bayes}}$ and the empirical measure **B**. **b**. Correlations between $\mathbf{B_{Bayes}}$ and the total variance of observation probability (Σ_1) only for the central value of current and previous first stimulus (see Fig. S7-4). **c**. Correlation between $\mathbf{B_{Bayes}}$ and the history parameter (η). **d**. Correlation between $\mathbf{B_{Bayes}}$ and the uncertainty in the observations of the second stimulus (W_2). **e-g.** Correlations of 7 networks classified as negative by model version m=1. **e**. Correlation between $\mathbf{B_{Bayes}}$ and Σ_1 only for the central value of current and previous first stimulus (see Fig. S7-5). **f.** Correlation between $\mathbf{B_{Bayes}}$ and η. **g**. Correlation between $\mathbf{B_{Bayes}}$ and W_2. C indicates the correlation value and p is the p-value.

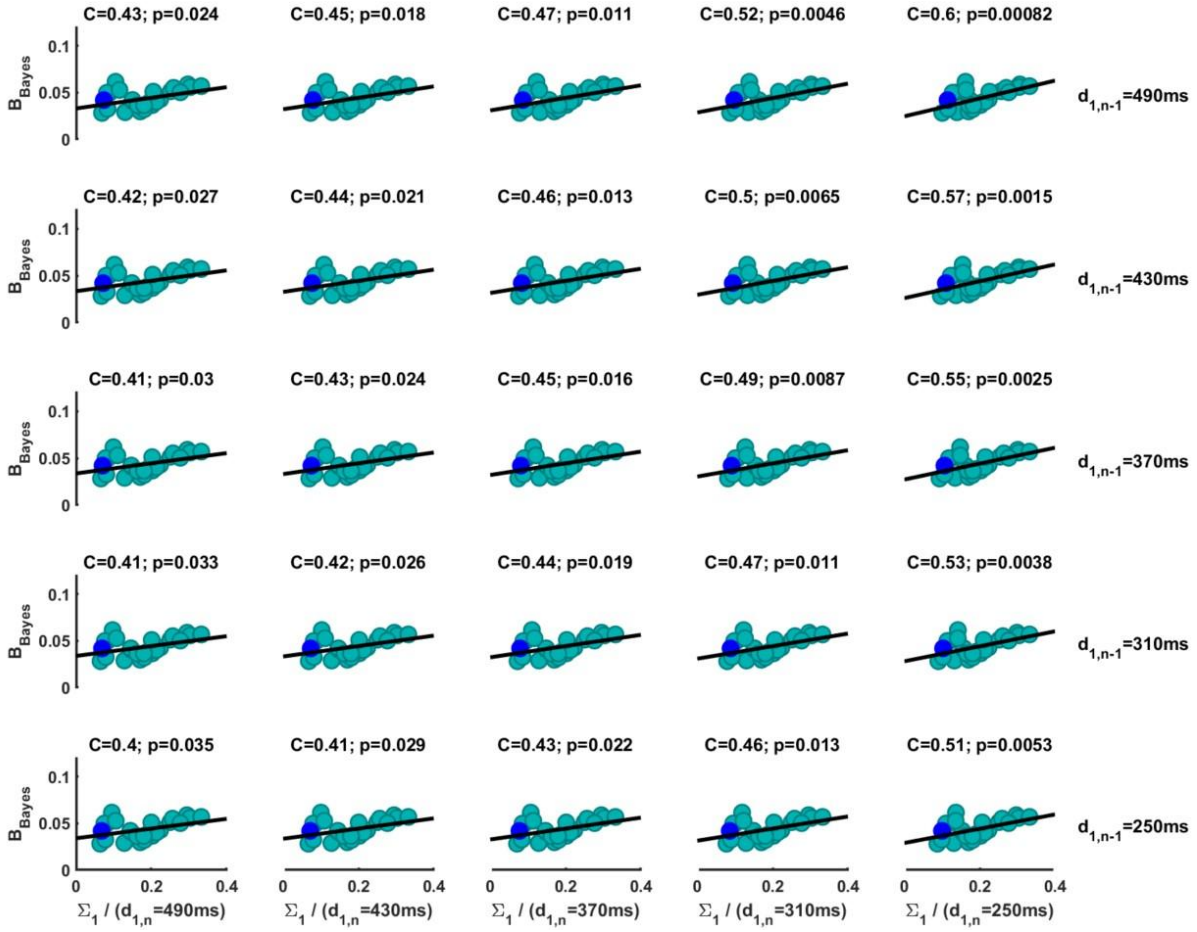

Fig. S7-4 | Correlations between B_{Bayes} and the total variance of observation probability (Σ_1) conditioning over the first stimulus on the previous (n-1) and current (n) trial, using those networks classified as positive by model version m=1. Blue circles indicate the example network 1 correlation.

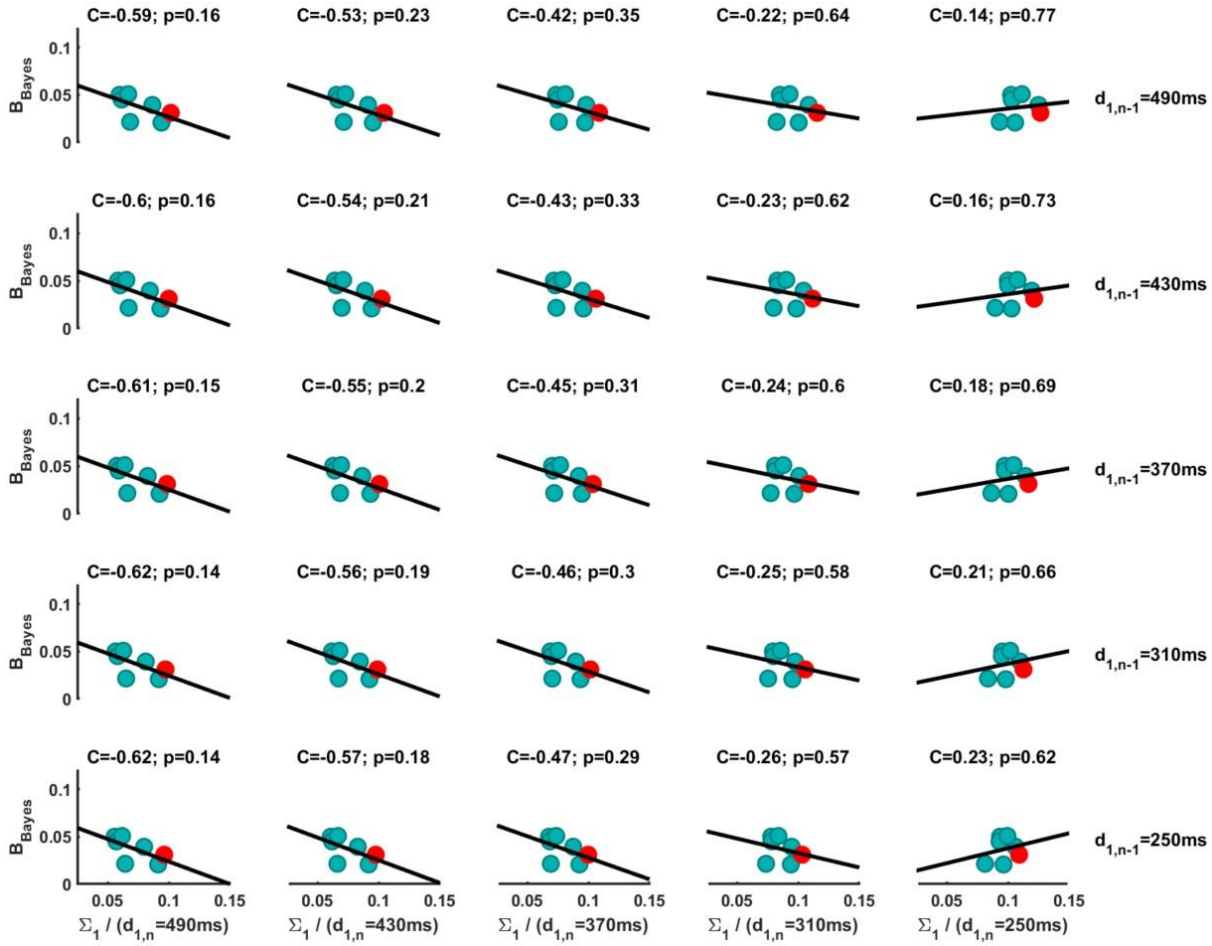

Fig. S7-5 | Correlations between B_{Bayes} and the total variance of observation probability (Σ_1) conditioning over the first stimulus on the previous (n-1) and current (n) trial, using those networks classified as negative by model version m=1. Red circles indicate the example network 2 correlation.

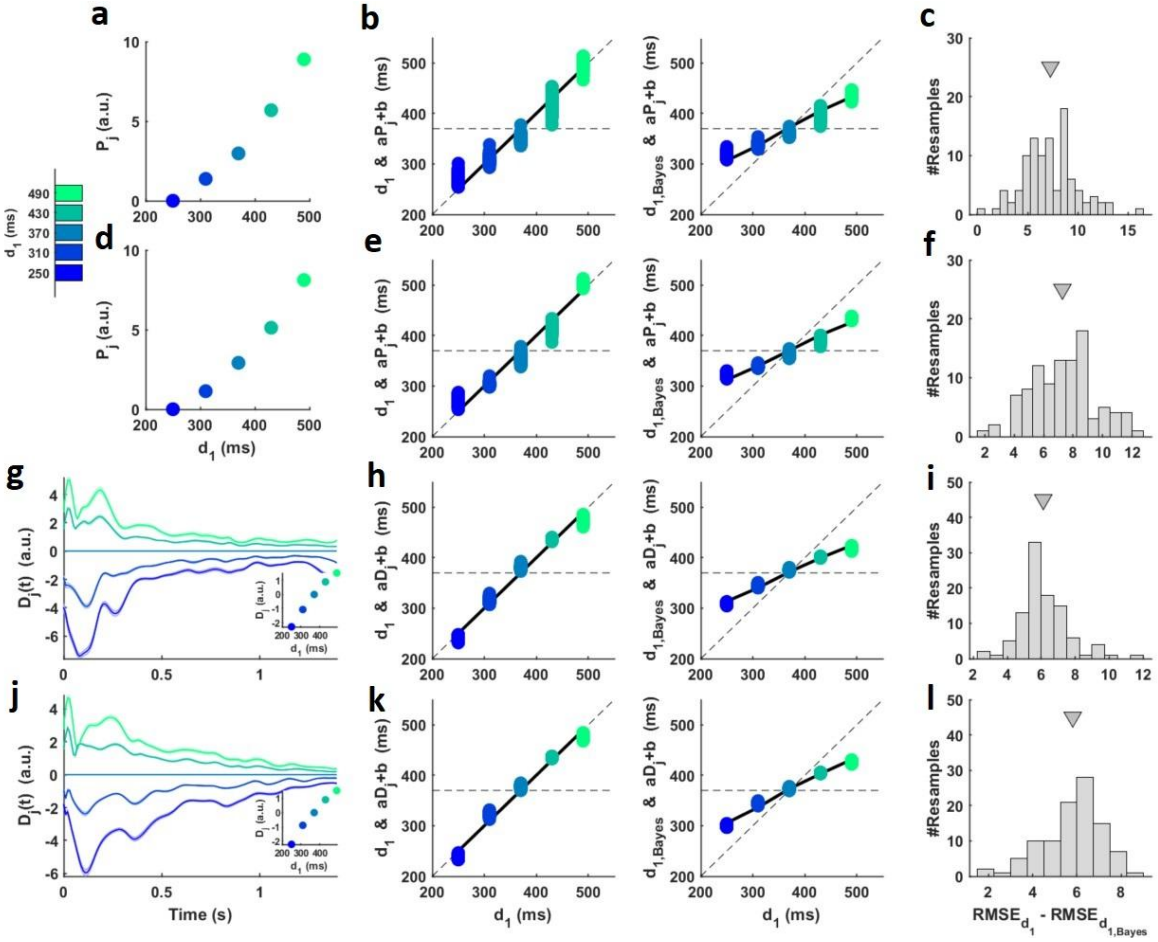

Fig. S8-1 | Effect of stimulation history on state space geometry and comparison with behavior for example network 1. a-c. State space geometry and test during the presentation of the first interval and conditioning over the previous class 1. **a**. Projections, P_j, of each SF1 state onto the encoding axis, u, as a function of d_1. **b**. Fit of a linear regression of the projections (aP_j + b) to the true d_1 (**left**) and the Bayesian estimator (**right**) as a function of d_1. The colored circles show multiple estimates of the regression derived from bootstrapping (n=100). The solid lines indicate the true d_1 (**left**) and the Bayesian estimator (**right**) as a function of d_1. **c**. Histogram on shufflings of the difference in RMSE between the real d1 hypothesis and the Bayesian estimate hypothesis. Gray triangle indicates the mean of each histogram. **d-f**. State space geometry and test during the presentation of the first interval and conditioning over the previous class 5 (same format as in **a-b**). **g-i**. State space geometry and test during the delay-period and conditioning over the previous class 1. **g**. Relative distance, $D_j(t)$, between each trajectory j=1-5 and the reference trajectory d_1=370 ms. Shadings indicate the median ± 95% CIs computed from a bootstrap procedure with 100 resamples. **Inset** shows the temporal means of distances as a function of d_1. **h**. Fit of a linear regression of the projections (aD_j + b) to the true d_1 (**left**) and the Bayesian estimator (**right**) as a function of d_1. The colored dots show multiple estimates of the regression derived from bootstrapping (n=100). **i**. Histogram on shufflings of the RMSE difference. **j-l**. State space geometry and test during the delay-period and conditioning over the previous class 5 (same format as in **g-i**). Firing rates, averaged over test trials with classes from diagonal block, of 234 tuning neurons were calculated aligning to SO1 and using an analysis time window of 100 ms and 10 ms of slide. Each of the previous four state spaces was constructed with 5-9 PCs in order to explain more than 90% of the variance.

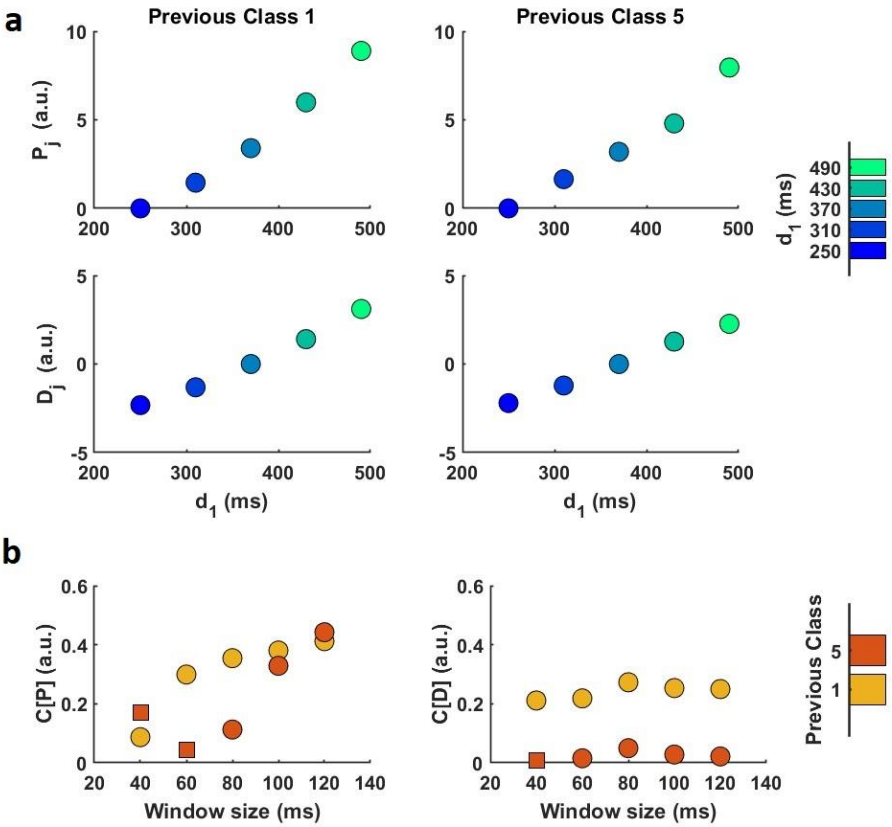

Fig. S8-2 | Effect of the previous class on the geometry in state space for example network 2 which has been classified as negative by model version m=1. a. Projections, P_j, of each SF1 state onto the encoding axis as function of d_1 (**upper**) conditioning on previous class 1 (**left**) and previous class 5 (**right**). Temporal mean of relative distances, D_j, as function of d_1 (**bottom**) conditioning over previous class 1 (**left**) and previous class 5 (**right**). The analysis time window used had a size of 100 ms and a slide of 10 ms. 6 PCs were used in order to build the state space, explaining the 97% (projections conditioning on previous class 1), 97% (projections, previous class 5), 81% (distances, previous class 1) and 84% (distances, previous class 5) of variance. **b**. **Left**. Compression of projections, C[P], shows a rather erratic behavior with the size of the analysis time window (40, 60, 80, 100 and 120 ms). **b. Right**. Compression of the mean relative distances, C[D], as a function of size of the analysis time window is null or small. The behavior of both compressions contradicts the existence of an appreciable contraction bias in this negative network. These compressions were computed using Eqs. 20-21 (Methods). If compression between the shortest d_1's (P_3 - P_1) was higher than the compression between the highest d_1's (P_5 - P_3), then C[P] was indicated as a square. Otherwise C[P] was indicated as a circle. Same in case of C[D].