

# Representational similarity modulates neural and behavioral signatures of novelty

## Highlights

- We model how stimulus similarities modulate novelty computation
- The model explains existing experimental data and makes new predictions
- Low-level feature similarity modulates V1 novelty responses in mice
- Spatial similarity modulates novelty-driven exploration in mice

## Authors

Sophia Becker, Alireza Modirshanechi, Wulfram Gerstner

## Correspondence

sophia.becker@epfl.ch

## In brief

The brain generalizes across similar stimuli, allowing for more efficient processing—yet it is unclear how such generalization affects the brain's responses to novel stimuli. Becker et al. propose a model quantifying the effect of stimulus similarities on novelty computation and use it to explain V1 novelty responses and exploration in mice.

Article

# Representational similarity modulates neural and behavioral signatures of novelty

Sophia Becker,<sup>1,2,5,\*</sup> Alireza Modirshanechi,<sup>1,2,3,4</sup> and Wulfram Gerstner<sup>1,2</sup>

<sup>1</sup>School of Computer and Communication Sciences, EPFL, Lausanne, Switzerland

<sup>2</sup>School of Life Sciences, EPFL, Lausanne, Switzerland

<sup>3</sup>Helmholtz Munich, Munich, Germany

<sup>4</sup>Max-Planck-Institute for Biological Cybernetics, Tübingen, Germany

<sup>5</sup>Lead contact

\*Correspondence: [sophia.becker@epfl.ch](mailto:sophia.becker@epfl.ch)

<https://doi.org/10.1016/j.neuron.2026.01.007>

## SUMMARY

**Novelty signals in the brain drive exploration and learning. While the perceived novelty of a stimulus is known to depend on previous experience, it remains elusive how generalization between familiar and novel stimuli impacts novelty computation. Specifically, existing models of novelty computation fail to account for the effects of stimulus similarities that are abundant in naturalistic tasks. Here, we present a biologically plausible model that captures how stimulus similarities modulate novelty signals in the brain and influence novelty-driven exploration. By applying our model to two publicly available datasets, we show (1) how generalization across similar visual stimuli affects novelty responses in the mouse visual cortex and (2) how generalization across nearby locations impacts mouse exploration in an unfamiliar environment. Our model explains distinct neural and behavioral signatures of novelty, makes mechanistic predictions about synaptic plasticity rules in novelty-computing circuits, and enables theory-driven experiment design.**

## INTRODUCTION

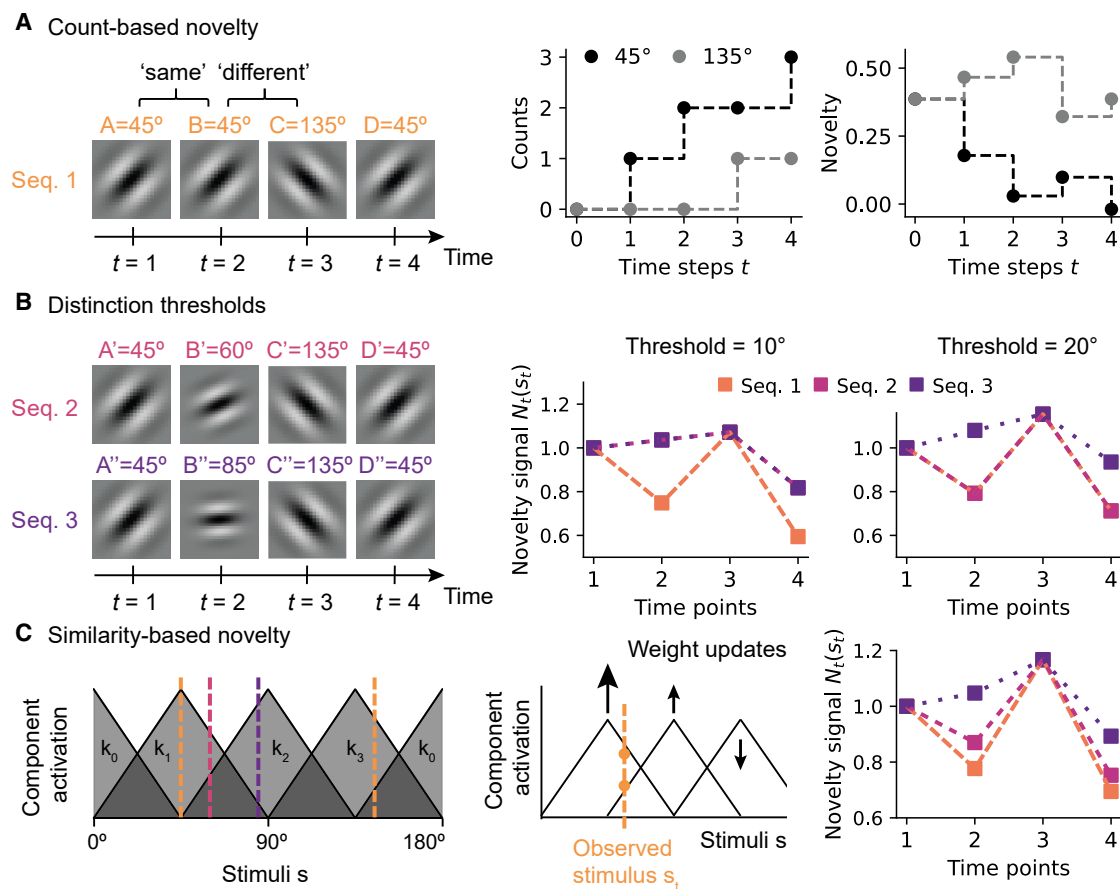
Novelty signaling in the brain is vital to facilitate learning,<sup>1–9</sup> enhance sensory processing,<sup>10–15</sup> and drive behavior.<sup>16–28</sup> Many experimental studies use computational models of novelty to explain and predict the effects of novelty on neural activity and behavior.<sup>20–24,29–31</sup>

However, the concept of novelty and its quantification in these studies relies on the assumption that stimuli are either “identical” or “different.” This approach neglects any generalization between stimuli and causes substantial limitations in environments that are continuous or exhibit stimulus similarities. For example, consider two previously unobserved stimuli that are similar to each other but not identical, e.g., two paintings by the same artist that exhibit a similar painting technique and style. If the two stimuli are presented sequentially, then how “novel” is the second painting after observing the first one? Classic novelty models force us to either (1) consider the two stimuli as entirely separate, implying that the second painting is completely novel despite its similarity to the first or (2) consider them as identical, predicting that the second painting and a repetition of the first painting would be experienced as equally familiar. Both options are at odds with what we would expect from human and animal perception.<sup>32</sup> In the same way, classic novelty models fail to address novelty computation in continuous environments, e.g., during spatial exploration. Indeed, several experiments suggest that the brain’s novelty response to a given stimulus does not

merely depend on the number of exposures to the exact stimulus but is also modulated by exposure to specific features of the stimulus<sup>16,33–35</sup> (discussion). These observations raise the question of how computational models of novelty can account for generalization across similar stimuli.

In the field of machine learning (ML), novelty-like signals are used to guide artificial agents during exploration of unfamiliar environments with sparse or no rewards.<sup>36–42</sup> These novelty-like signals are often computed as decreasing functions of the observed frequency of a given stimulus (“state”). The observed frequency is computed by neural networks that, through training, generalize across similar states.<sup>36–42</sup> The success of such novelty-guided artificial agents in exploring complex environments highlights the functional relevance of generalization across similar stimuli and locations in the environment. However, ML models rely on extensive training to estimate novelty, as well as architecture and learning rules with limited biological plausibility and interpretability.

Here, we propose a biologically plausible modeling framework for novelty computation (“similarity-based novelty”) that combines the strengths of classic and ML-inspired novelty models to capture the effects of generalization on novelty computation. Our model is consistent with classic novelty models<sup>20–24,29–31,43</sup> in environments with discrete stimuli but extends to continuous spaces and environments with similarity structure. We test our similarity-based novelty model on two datasets and derive new experimental predictions.



**Figure 1. From count-based to similarity-based novelty**

(A) Stimulus counts as in Equation 1 (middle) and count-based novelty responses as in Equations 2 and 3 (right) for an example sequence of four Gabors, characterized by their angular orientations between 0° and 180° (left). Leakiness is set to  $\alpha = 0$ . Each time we observe a stimulus, its count increases by one unit. Different stimuli (e.g., here: 45° vs. 135°) are counted separately. Stimulus novelty is a decreasing function of the stimulus count.

(B) Count-based novelty responses (middle and right), as on the right in (A), for two additional example sequences of four Gabors (left). The three sequences in (A) and (B) are identical except for the second stimulus. Novelty predictions for the three Gabor sequences in (A) and (B) depend on the chosen distinction threshold between stimuli. Stimuli are counted as different if they differ by more than 10° (middle) or 20° (right), respectively.

(C) Similarity-based novelty as defined in Equations 4, 5, and 6, using triangular components centered at reference orientations 0°, 45°, 90° and 135° (left; see STAR Methods). Component weights encode the empirical frequency of each component, and they change relative to the activation of their associated component (middle). Vertical dashed lines in (C) (left, middle) indicate stimulus locations in the interval [0°, 180°]. Similarity-based novelty predictions (right) reflect the different levels of similarity between the first two stimuli across the three sequences. See also Figure S1.

## RESULTS

### Count-based novelty models fail to account for similarity modulation of novelty responses

Most algorithmic models of novelty computation in the brain<sup>20–24,29–31,44,45</sup> are based on estimating the count-based frequency at which stimuli (e.g., sensory stimuli, spatial locations, etc.) have been observed: the more often a stimulus is observed, the less novel it is. These “count-based” novelty models usually consist of four building blocks: (1) a stimulus representation, (2) stimulus counters, (3) an empirical frequency function that measures the familiarity of stimuli, and (4) a novelty function that computes the novelty of stimuli as a decreasing function of their familiarity.<sup>20–24,29–31</sup>

The “stimulus representation” specifies how stimuli from the environment are represented. For example, Gabor-like visual stimuli could be characterized by their angular orientation (Figure 1A, left). The “stimulus counters”  $C^{(t)}(s)$  keep track of how often a stimulus  $s$  has been observed up to time  $t$ : every time the stimulus  $s$  is observed, its counter  $C^{(t)}(s)$  is increased by one unit (Figure 1A, middle). In the following, we consider “leaky” count-based novelty, where the stimulus counters  $C^{(t)}(s)$  decay with a factor  $(1 - \alpha)$ ,  $\alpha \in [0, 1]$ , in each time step:

$$C^{(t)}(s) = (1 - \alpha)C^{(t-1)}(s) + \delta(s_t, s), \quad (\text{Equation 1})$$

where  $\delta(s_t, s)$  is 1 if the current observation  $s_t$  is equal to stimulus  $s$  and 0 otherwise. The leakiness of the counters captures the effect of forgetful stimulus memory (see STAR Methods for

details). A value of  $\alpha = 0$  corresponds to non-leaky count-based novelty. Count-based novelty models use separate counters for different stimuli. Examples where stimuli can be clearly identified as distinct or identical are human studies in abstract sequential decision-making tasks.<sup>20,46</sup> However, as we illustrate below, counting, be it leaky or non-leaky, is not well-defined in continuous stimulus spaces, causing problems with count-based novelty predictions.

The third element of count-based novelty models is the "empirical frequency" function  $p$  that measures the familiarity of a given stimulus  $s$  at time  $t + 1$ :

$$p^{(t+1)}(s) = \frac{C^{(t)}(s) + \varepsilon}{T_t + |S|\varepsilon}, \quad (\text{Equation 2})$$

where  $C^{(t)}(s)$  are the stimulus counters,  $T_t = (1 - \alpha)T_{t-1} + 1$  is the leaky time count, and  $|S|$  is the number of different stimuli. At time  $t = 0$ , all stimuli start with the same prior frequency  $p^{(0)}(s) = 1/|S|$ . The frequency increases if a stimulus  $s$  is observed (by increasing the stimulus count  $C^{(t)}(s)$  and time count  $T_t$ ), and decreases if it is not observed (by increasing only the time count  $T_t$ ). The parameter  $\varepsilon$  regulates how much the frequency estimate  $p^{(t)}$  is influenced by the (uniform) prior  $p^{(0)}$ .

The last element of count-based novelty models is the "novelty" function  $N$ . The novelty of a stimulus  $s$  decreases nonlinearly with its familiarity:

$$N^{(t)}(s) = -\log p^{(t)}(s), \quad (\text{Equation 3})$$

where the negative logarithm is a standard choice of nonlinearity in the novelty literature.<sup>43,45</sup> The combination of empirical count-based frequency (Equation 2) and logarithm (Equation 3) makes the novelty decrease rapidly during the first few observations of a stimulus  $s$  (Figure 1A, right), in line with experimental observations.<sup>47</sup>

While count-based novelty predicts the novelty of stimuli that can be clearly identified as distinct or identical, it is not well-defined for stimuli with varying degrees of similarity. To illustrate this, we consider a toy example: we present three stimulus sequences with varying levels of stimulus similarities to two count-based novelty models with different distinction thresholds between stimuli (Figures 1A and 1B, left). Each sequence consists of four Gabor stimuli, characterized by their angular orientation. The three sequences are identical except for the second stimulus, which varies in similarity to the first (seq. 1: identical, seq. 2: 15° difference, seq. 3: 40° difference). We evaluate the novelty response to each stimulus as predicted by two count-based novelty models (without leak,  $\alpha = 0$ ): the first model considers Gabors as distinct if they differ by more than 10° (Figure 1B, middle) and the second one if they differ by more than 20° (Figure 1B, right). These threshold values are consistent with the range of psychometric thresholds for orientation differences in mice.<sup>48</sup> For sequence 1 and sequence 3 (orange and purple lines in Figure 1B), both count-based novelty models predict qualitatively similar novelty responses: every time we observe the first stimulus, its novelty decreases (seq. 1: at  $t = 2, t = 4$ ; seq. 3: at  $t = 4$ ), while previously unseen stimuli show high novelty responses (seq. 1:  $t = 1, t = 3$ ; seq. 3:  $t = 1, t = 2, t = 3$ ). Since we are measuring frequencies, and the

initial counts  $C^{(0)}$  in Equation 2 are zero for all stimuli, the responses to completely novel stimuli increase during the observation of the sequence (with different slopes, depending on the distinction threshold, i.e., the number of stimulus counters). Importantly, for the second sequence, the two count-based models predict novelty responses that are either identical to the third sequence (low distinction threshold, *distinct* first two stimuli; Figure 1B, middle) or the first sequence (high distinction threshold, *identical* first two stimuli; Figure 1B, right). This exemplifies that count-based novelty models do not account for stimulus similarities beyond a simple distinction between same and different, contradicting both common-sense intuition and experimental evidence<sup>16,33–35</sup> (discussion). A more intuitive notion of novelty would allow novelty to be modulated by similarities between stimuli.

### Similarity-based novelty

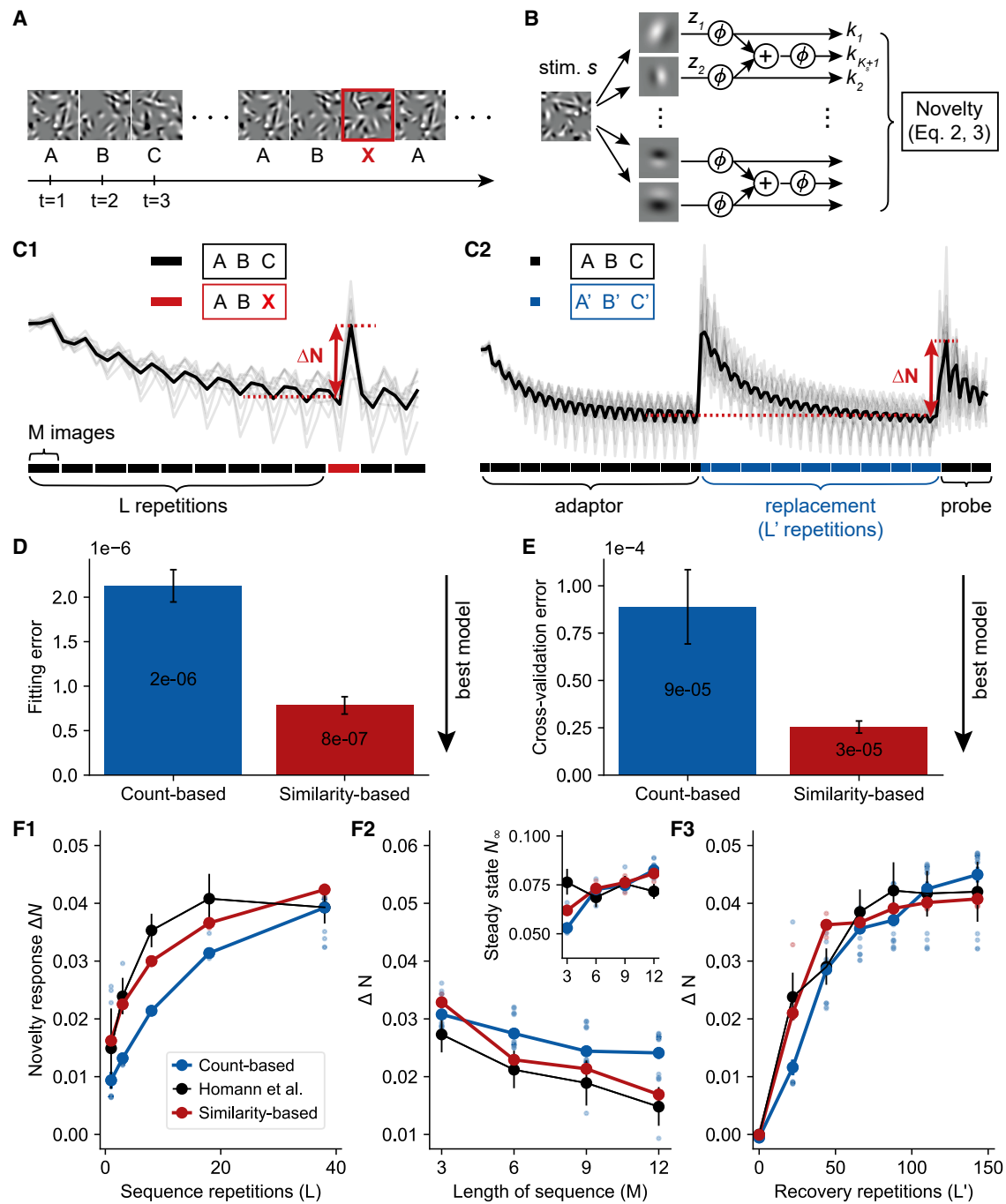
To address the limitations of count-based novelty models, we propose a generalized novelty model ("similarity-based novelty") that accounts for the effect of stimulus similarities and allows us to model novelty in continuous stimulus spaces. Its core difference to count-based novelty models is that, instead of computing the familiarity of stimuli as a function of their *discrete* empirical frequency (Equation 2), our model uses probabilistic mixture models to compute a *continuous* empirical frequency density across the stimulus space  $S$ .

The idea of mixture models is to represent a complex probability density as the weighted sum of simpler probability densities (mixture "components") over the stimulus space.<sup>49</sup> The mixture components can have arbitrary shapes, as long as they are non-negative and their integral over the stimulus space is one. In general, each component can overlap with an arbitrary number of other components. In the context of our problem, mixture components can serve as a flexible way to express stimulus similarities in both discrete and continuous spaces. For example, in our toy example, we can choose overlapping triangular components to account for orientation similarities (Figure 1C). Each component encodes the similarity of any given stimulus to a fixed reference stimulus, i.e., the orientation at the center of the triangular component. The pairwise similarity between two arbitrary stimuli can thus be described by the overlap in components they activate.

Based on the components, we define the familiarity  $p$  of a stimulus  $s$  at time  $t + 1$  as a mixture model that generalizes the empirical frequency in Equation 2:

$$p^{(t+1)}(s) = \sum_{j=1}^N w_j^{(t)} k_j(s), \quad (\text{Equation 4})$$

where  $k_1, \dots, k_N$  are the components that account for similarities in the stimulus space, and  $w_1^{(t)}, \dots, w_N^{(t)}$ , with  $\sum_{j=1}^N w_j^{(t)} = 1$  are the weights with which each component contributes to the empirical frequency  $p^{(t)}$  at a given time point  $t$ . In order for  $p$  to represent a meaningful and consistent notion of familiarity across space and time, we must choose the weights  $w_j^{(t)}$  appropriately. We do so using statistical inference: after a sequence of observations  $s_1, \dots, s_t$ , we choose the weights so as to maximize the likelihood



**Figure 2. Similarity-based novelty robustly explains novelty responses in mouse V1**

(A) Passive viewing task of Homann et al.<sup>51</sup> A sequence of images is presented repeatedly, except that the last image in the last sequence repetition is replaced by a novel, i.e., previously unseen, image (X). Familiar images in the sequence and novel images X each consist of a linear superposition of 40 Gabor filters with random parameters.

(B) The components of similarity-based novelty ( $k_j$  in Equation 4) emulate V1 simple and complex cells.<sup>57,58</sup> “Simple cell”-like components compute the cosine similarity between the stimulus image  $s$  and their reference Gabor (defined by its preferred orientation, phase, frequency, and location) and pass it through a non-negative function  $\phi$  (here: ReLU for all components; see STAR Methods) to compute their component activation. The activation of a “complex cell”-like component is computed by applying the same function  $\phi$  to the sum of 2 simple cell-like component activations with opposite phase.

(C) Three variations of the passive viewing experiment<sup>51</sup> in (A), with example population traces predicted by similarity-based novelty simulations (black line: average across 10 individual simulations [gray lines] for different image sequences). We measure responses  $\Delta N$  to the novel image X for (1) different number  $L$  of sequence repetitions ( $M = 3$  fixed; C1) and (2) different number  $M$  of familiar images in a sequence ( $L = 18$  fixed; C1). We further assess the recovery from

(legend continued on next page)

of the observed sequence. The resulting update for the similarity-based novelty weights  $w_j^{(t)}$  can be reformulated as an iterative learning rule that updates the weights in a *multiplicative* fashion (STAR Methods). Under the biologically motivated assumption that the similarity-based weight update uses a constant learning rate, its learning rule can further be linked to existing<sup>33,47,50–56</sup> or new (STAR Methods) circuit models of novelty computation.

Analogously to leaky count-based novelty, we define “leaky” similarity-based novelty by accounting for gradual forgetting of past stimulus observations. The weights  $w_j^{(t)}$  of leaky similarity-based novelty are given as

$$w_j^{(t)} = \frac{R_j^{(t)} + \varepsilon}{T_t + N\varepsilon}, \quad (\text{Equation 5})$$

where  $R_j^{(t)}$  represents a leaky “soft counter” of past component observations  $j$ :

$$R_j^{(t)} = (1 - \alpha)R_j^{(t-1)} + \gamma_j^{(t)} \quad (\text{Equation 6})$$

The soft counter  $R_j^{(t)}$  is counting “responsibilities”  $\gamma_j^{(t)}$  that represent how well a given component of the mixture distribution is explaining the current stimulus observation (STAR Methods). Analogously to the leaky stimulus counters in count-based novelty, the forgetfulness of the soft counter  $R_j^{(t)}$  is regulated by the leakiness parameter  $\alpha$ . Using the component weights defined in Equation 5 along with suitable stimulus and component representations, we estimate the familiarity of a stimulus  $s$  at time  $t$  by its empirical frequency  $p^{(t)}(s)$ . The definition of familiarity in Equations 4 and 5 implies that similar stimuli influence each other’s familiarity via the components they share. For example, if a stimulus activates two (overlapping) components, its familiarity will be influenced by other stimuli that share any of the two components. Analogously to count-based novelty (Equation 3), we then apply the negative logarithm to  $p^{(t)}(s)$  to compute the novelty  $N^{(t)}(s)$ . The resulting novelty model (similarity-based novelty) is (1) well defined in both discrete and continuous stimulus spaces, (2) accounts for similarities between stimuli via its components, and (3) is equivalent to classical count-based novelty in discrete stimulus spaces without similarity structure (STAR Methods).

To illustrate the differences between count-based and similarity-based novelty, we compare their novelty predictions in our toy example (Figures 1A and 1B, left). Analogous to the count-based model (Figure 1B, middle and right), we predict the novelty responses to each stimulus in each sequence using similarity-based novelty (Figure 1C). As components, we choose equidistantly placed, triangular functions that encode orientation similarities as linearly decreasing functions of their angular difference (Figure 1C, left and middle). This resembles a piecewise-linear

approximation to the tuning curves of orientation-selective cells in V1.<sup>57,58</sup> In contrast to count-based novelty, similarity-based novelty predictions for the second stimulus increase with the angular difference to the first stimulus (Figure 1C, right), reflecting the varying degree of similarity between the first two stimuli across the three sequences. An analogous similarity-driven decrease of novelty responses is predicted for the stimulus at time  $t = 4$ . Note that the novelty predictions for the third stimulus in each sequence are not influenced by similarity because it is so dissimilar to other stimuli in the sequence ( $>50^\circ$  difference) that it does not share any components. For wider components, generalization across stimuli also affects the third stimulus (Figure S1). This toy example illustrates that, in contrast to count-based novelty, similarity-based novelty generalizes familiarity across stimuli based on their similarity, as defined by the underlying components.

### Similarity-based novelty explains novelty responses in mouse V1

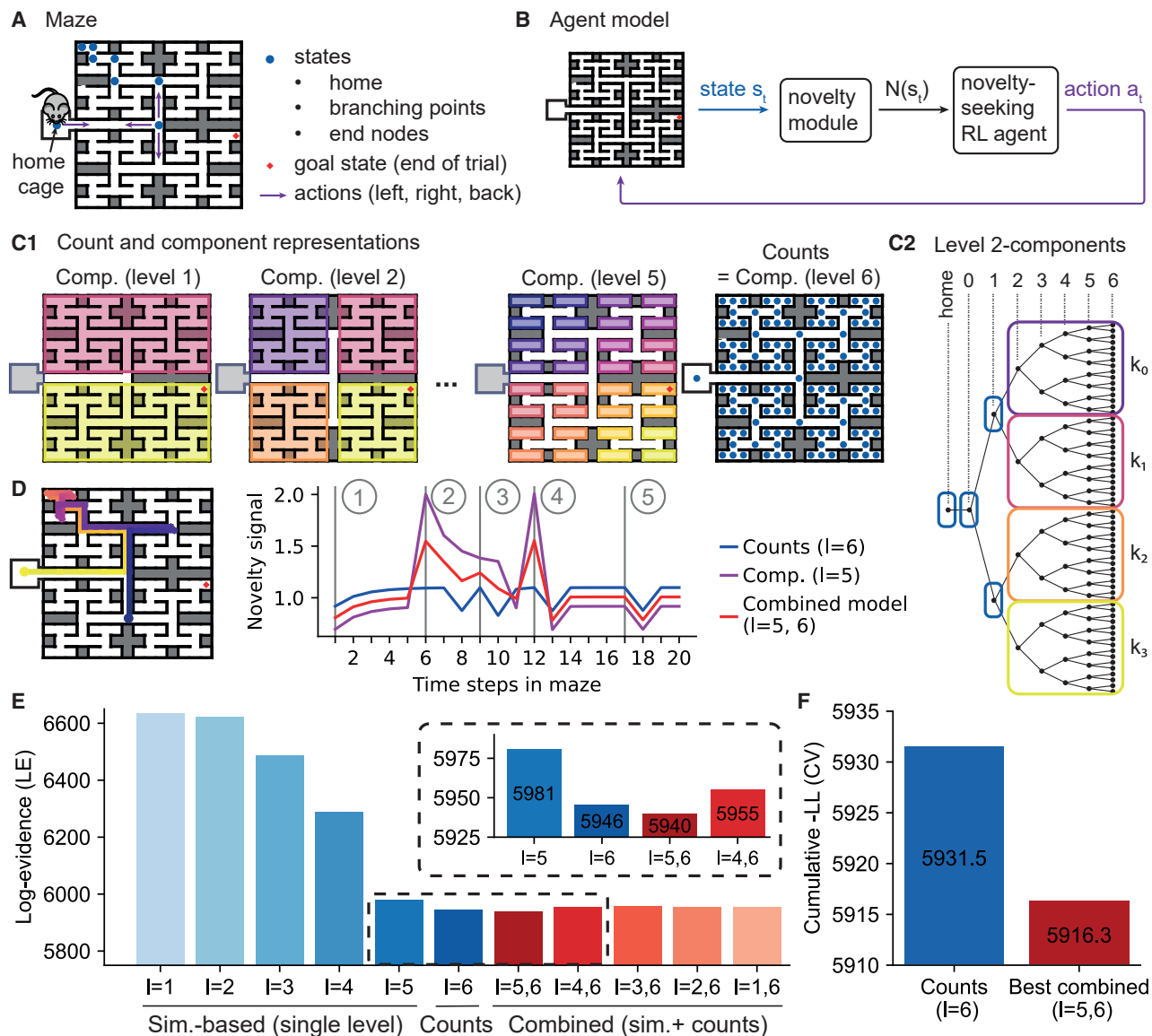
We wondered whether similarity-based novelty can capture novelty responses in the mouse primary visual cortex (V1) during passive viewing of familiar and novel images. The visual stimuli in the passive viewing experiment of Homann et al.<sup>51</sup> are images that each consist of multiple randomly placed and randomly oriented Gabor filters (Figure 2A). Homann et al. study V1 novelty responses in mice during three variations of their experiment (Figures 2C1 and 2C2). In the first variant (“ $L$ -experiment”), a sequence of three images ( $M = 3$ ) is presented for  $L$  repetitions ( $L = 1, 3, 8, 18, 38$ ) before a “novel” image ( $X$ ) is shown in the subsequent repetition (Figure 2C1). Here, a novel image is defined as an image that is not in the set of familiar images that are shown as part of the repeating image sequence. The neural population response to the novel image increases with  $L$  (Figure 2F1, black line). In the second variation of the experiment (“ $M$ -experiment”), the number  $M$  of familiar images in the sequence is varied ( $M = 3, 6, 9, 12$ ; Figure 2C1), while the number of sequence repetitions  $L$  remains constant ( $L = 18$ ). Increasing  $M$  decreases the neural responses to the novel image (Figure 2F2, black line) but does not significantly affect the steady-state population activity after habituation to the familiar sequence (Figure 2F2, black line in inset). The third variation (“ $L'$ -experiment”) investigates the recovery from familiarity: after neural responses to a familiar sequence (“adaptor,”  $M = 3$ ) have converged, a novel sequence (“replacement,”  $M = 3$ ) is presented for  $L'$  repetitions, before the initial sequence is shown again and the neural population response  $\Delta N$  is measured (Figure 2C2). The recovered novelty response to the formerly familiar sequence increases with  $L'$  (Figure 2F3, black line).

familiarity (C2): after novelty responses to an initial sequence (adaptor) have converged, a different sequence (replacement) is presented for  $L'$  repetitions. Subsequently, the recovered novelty response  $\Delta N$  to the initial sequence (probe) is measured ( $M = 3$  fixed).

(D and E) Fitted (D) and leave-one-out cross-validated (E) mean squared error (MSE) between neural novelty responses<sup>51</sup> and novelty responses predicted by fitted count-based and similarity-based novelty, combined across the three experiments. Error bars show the SEM, estimated with jackknife resampling of the fitting and cross-validation process, respectively (STAR Methods).

(F) Average neural novelty responses<sup>51</sup> (black line, error bars: SEM across  $n = 5$  mice) and average novelty predictions  $\Delta N$  (blue and red lines) by fitted models on the held-out data (here: single data point) as a function of sequence repetitions  $L$  (F1), sequence length  $M$  (F2), and recovery repetitions  $L'$  (F3). Inset in F2: steady-state responses  $N_\infty$  to the familiar sequence. Individual blue and red data points show model predictions obtained by jackknife resampling of the cross-validation process (as in E).

See also Figures S3 and S4.



**Figure 3. Similarity-based novelty explains mouse exploration in an unfamiliar maze**

(A) The maze of Rosenberg et al.<sup>59</sup> We define states (blue circles, example states on a short trajectory) with respective actions (purple arrows): (1) a single action in the home cage (go into the maze), (2) three actions at branching points (turn left, right, or backward), and (3) one action in the end nodes (turn back). To investigate intrinsically driven exploration, we consider mouse behavior before their first encounter with the goal state (red square).

(B) In each time step  $t$ , a novelty-seeking RL agent takes its current position (state  $s_t$ ) as input and computes the state novelty using its internal novelty model. Then it updates its internal value function and policy based on model-based RL (STAR Methods) and chooses its next action,  $a_t$  from its policy.

(C1 and C2) Counts and components used for the RL agents' internal novelty models. (C1) Components of different granularity that tile the state space into "areas" of the maze, e.g., into quadrants in the level-2 representation. Each component is active in a given area of the maze (marked in different colors). For example, the purple level-2 component  $k_0$  is active in all states within the purple quadrant, while states further up in the tree have separate counters (marked in blue in C2). Level-6 components correspond to count-based novelty, with separate counters for each state (C1, right). For the computation of novelty, we consider either similarity-based novelty with level- $l$  components,  $l = 1, \dots, 5$ ; count-based novelty (corresponding to level-6 components); or a combined count- and similarity-based model (level-6 components and level- $l$ ,  $l = 1, \dots, 5$  components).

(D) Novelty signals predicted by a count-based novelty agent (blue), a level-5 similarity-based novelty agent (purple), and a combined count- and level-5 similarity-based novelty agent (red) on an example path through the maze. Numbers indicate relevant behavioral events along the path: first entry into (1) the maze, (2) novel level-5 component, (3) novel level-6 component, (4) novel level-5 component, and (5) novel level-2 component (quadrant).

(E) Model fit of novelty-seeking RL models to mouse exploration behavior, as indicated by the model log-evidence (LE, log-likelihood augmented by penalty for number of parameters; see STAR Methods), for model-based RL agents seeking either similarity-based or count-based novelty (blue), and model-based RL agents seeking combined count- and similarity-based novelty (red).

(legend continued on next page)

To test whether these qualitative features of novelty responses in V1 can be captured by leaky count-based and similarity-based novelty, respectively, we fit each novelty model to the combined experimental data<sup>51</sup> from all three experiments (STAR Methods). The fitted parameters of each model minimize the mean squared error (MSE) between the statistics of predicted and measured neural novelty responses in all three experiments. For the count-based novelty model, all images are considered as distinct. For the similarity-based novelty model, we define components that emulate either V1 simple cells<sup>57,58</sup> with specific orientation, phase, frequency, and location preference, or V1 complex cells<sup>57,58</sup> that combine the feature preferences of two V1 simple cells with opposite phase (Figure 2B, see STAR Methods). The number of bins (for count-based novelty) and components (for similarity-based novelty) as well as the leakiness parameter  $\alpha$  are free parameters that are fitted jointly with the remaining parameters of each model using grid search (STAR Methods).

Novelty responses as predicted by the fitted similarity-based novelty model capture data substantially better than count-based novelty (Figure 2D). To control for the possibility of overfitting by the more complex similarity-based model, we cross-validate this finding by leave-one-out cross-validation (Figure 2E). Moreover, on the held-out data (Figure 2F), similarity-based novelty (red) explains the features of V1 novelty responses across all three Homann experiments. By contrast, count-based novelty (blue) deviates from V1 novelty responses, most notably in the *L*- and *M*-experiments. To investigate why count-based novelty fit V1 novelty responses less well and less robustly, we perturbed single parameters from their fitted values (Figure S3). We found that the fit of count-based novelty is highly susceptible to changes of the leakiness parameter  $\alpha$ . In particular, slight deviations from the fitted  $\alpha$ -value led to instant saturation of the novelty responses predicted by count-based novelty (Figure S3D). This indicates that count-based novelty needs to trade off the timescales required for each experiment: a higher-than-fitted  $\alpha$  in the *L*-experiment causes the predicted novelty responses to reach their steady state in very few repetitions, preventing the gradual saturation of novelty responses observed in the neural data. Simultaneously, a lower-than-fitted  $\alpha$  prevents the decline of novelty responses with increasing stimulus sequence length *M*. By contrast, similarity-based novelty maintains its good fit across the majority of  $\alpha$  values (Figure S3E). Moreover, for other model parameters, the fit of similarity-based novelty is stable across the entire parameter range (Figures S3A and S3C). The better and more robust fit of similarity-based novelty is based on the model's capacity to generalize familiarity across similar stimuli. For example, in the *M*-experiment, generalization leads to a natural decrease of novelty responses as the number *M* of familiar stimuli and, thus, the feature overlap with the novel stimulus increases. As a result, similarity-based novelty avoids a trade-off of leakiness time-

scales and achieves a better and more robust fit to neural responses across all three experiments.

### Similarity-based novelty explains mouse exploration in an unfamiliar maze

Going beyond the passive viewing task, we now ask whether similarity-based novelty also helps explain behavioral signatures of novelty processing during active exploration. Behavioral data by Rosenberg et al.<sup>59</sup> show exploration of mice in an unfamiliar, freely accessible maze (Figure 3A). The maze has the structure of a binary tree (Figure 3C2): mice enter the maze through a single corridor that branches into two corridors, each of which branches again into two corridors, etc. After six branching points, each corridor ends in a wall ("end node"), where mice can only turn around. Half of the mice do not receive any reward, and the other half receive a drop of water upon licking a reward port in one of the maze's end nodes ("goal state"). To only include novelty-driven behavior in our analysis, we consider mice behavior before the first encounter with the goal state in both the unrewarded and the rewarded groups. During this initial exploration phase, mice from the rewarded and the unrewarded groups have access to identical information and do not show significant behavioral differences.<sup>59</sup>

We model mouse behavior using novelty-seeking reinforcement learning (RL) models that maximize intrinsically computed novelty instead of extrinsic rewards from the environment (Figure 3B, see STAR Methods). To compare the exploration behavior of different "agents," i.e., mice and novelty-seeking RL models, we define "states" and "actions" in the maze. States include the home cage, the branching points, and the end nodes (blue circles in Figure 3A). In each of the branching points, mice or RL agents can take one of three actions (purple arrows in Figure 3A): (1) going down the *left* corridor until the next branching point or end node, (2) going down the *right* corridor until the next branching point or end node, or (3) going *back* to the previous branching point. In the home cage and the end nodes, agents only have one available action, i.e., going back.

We compare mouse behavior to model-based novelty-seeking RL models and ask whether seeking leaky count-based novelty, leaky similarity-based novelty, or a combination of both best explains mouse behavior. For similarity-based novelty, we define components as abstract "place fields" that encode spatial similarity in the binary tree by considering localized "areas" of the maze (e.g., a given quadrant of the maze). The size of the area encoded by a given component defines the spatial scale ("granularity level") across which it generalizes familiarity. For example, a "level-2" component (Figures 3C1 and 3C2) generalizes across all states in one quadrant, while a "level-5" component (Figure 3C1) generalizes across two neighboring end nodes and their closest branching point. The home cage as well as each branching point outside of the set of areas covered at a given level are treated as separate states

(F) As a further approximation of the model LE, we calculate the cumulative negative LL across held-out sets in 5-fold cross-validation ("CV-LL," based on splits of the set of mice) for the best count-based and best combined count- and similarity-based RL agents from (E) (all agents: see Figure S5). The difference in LE and CV-LL can be interpreted as log-Bayes factors, respectively; differences greater than  $\log(20) = 3$  and  $\log(150) = 5$  are considered as significant and strongly significant.<sup>60,61</sup>

See also Figure S5.

(Figure 3C2). For count-based novelty, we count each state in the maze separately—this is equivalent to using level-6 similarity-based components (i.e., abstract place fields of smaller size). Agents seeking a combination of count-based and similarity-based novelty use two sets of components: (1) counts, i.e., level-6 components, and (2) components of one additional level  $l < 6$ . Novelty signals are computed separately for each set of components and combined into a final novelty signal through a weighted sum with fitted weight (STAR Methods). Using a combination of smaller components ( $l = 6$ ) and larger components ( $l < 6$ ) reflects findings that hippocampal place cells have place fields of different sizes.<sup>62</sup>

Count-based, similarity-based, and combined count-similarity-based novelty models produce qualitatively different novelty signals along a fixed example path (Figure 3D). For example, level-5 components (purple line) generate strong novelty responses upon encountering an unseen group of neighboring end nodes and associated branching points (events 2 and 4) that gradually decay while staying within that group of states. By contrast, count-based (level-6) novelty (blue line) reacts with approximately equal novelty to the two novel end nodes and branching point (events 2 and 3). Both level-5 and count (level-6) components generate similar novelty predictions for states higher up in the binary tree (events 1 and 5). The combined model is a mixture of both signals (red line). To test how the spatial scale of generalization affects novelty-driven exploration, we compare (1) count-based novelty, (2) similarity-based novelty models with components of different granularity levels (levels 1–5), and (3) combined count-similarity-based novelty models with count components and one additional set of components (levels 1–5) to mouse behavior.

We fit all model variants described above, i.e., model-based RL with different novelty types (count-based, similarity-based, and combined count-similarity-based) for all levels of granularity, to mouse behavior, using maximum likelihood estimation of the parameters (STAR Methods). Bayesian model comparison between all fitted models (STAR Methods) shows that exploration driven by combined count-similarity-based novelty with best-fitting granularity explains mouse behavior significantly better than exploration driven by count-based novelty (Figures 3E and 3F), despite higher model complexity (accounted for through a penalty in the model log-evidence, Figure 3E, and by cross-validation, Figure 3F). In particular, the combined novelty model with level-5 components and count (level-6) components best explains the exploration behavior of mice. Components of granularity  $l = 5$  generalize across two immediately neighboring end nodes and their shared branching point and correspond to place fields that cover two neighboring end nodes as well as the corridor connecting these. Components of granularity  $l = 6$  correspond to localized place fields restricted to a single endpoint or corridor. Hence, the combination of "place fields" of two different sizes leads to the best prediction of behavior.

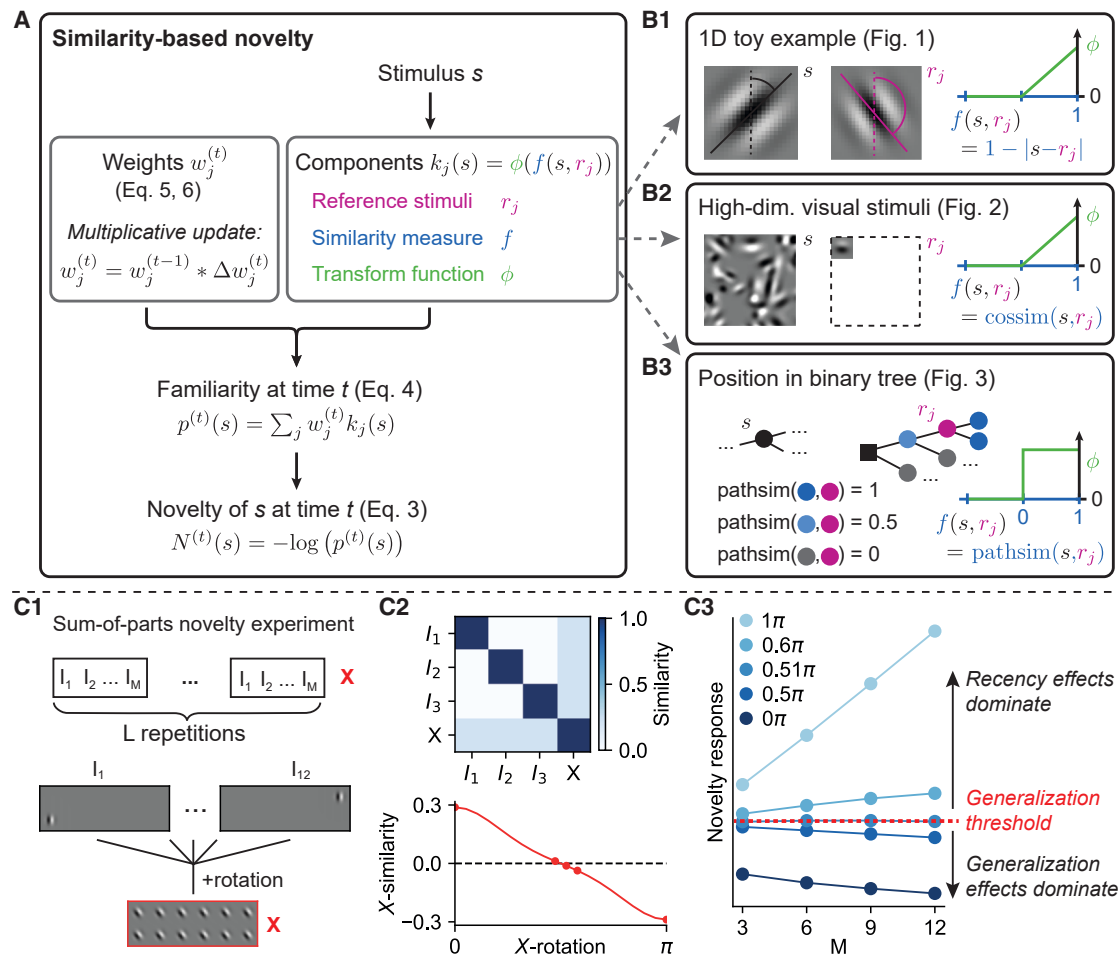
### A framework for new theory-driven experiments

In the previous sections, we showed that similarity-based novelty captures V1 novelty responses and novelty-driven exploration in mice. In this section, we illustrate why similarity-based novelty presents a flexible modeling framework for novelty

computation across tasks and ask whether we can use similarity-based novelty to make testable predictions for new experimental protocols.

The core elements of similarity-based novelty are (1) the non-negative component functions  $k_j$  that encode stimulus similarities in the representational space and (2) the history- and time-dependent weights  $w_j^{(t)}$  that encode the frequency of each component (Figure 4A). The weights are updated with a bio-plausible, multiplicative plasticity rule that is directly derived from the normative approach and remains the same across different choices of component functions. The component functions, on the other hand, can be flexibly adapted to model different datasets. In this work, we presented three types of components: (1) simple, one-dimensional components that capture orientation similarity (Figure 4B1), (2) V1-like components that capture similarities in high-dimensional image space (Figure 4B2), and (3) abstract place field-like components that capture spatial similarities in a binary tree maze (Figure 4B3). Importantly, these choices of component functions all share three basic elements (Figure 4A). The first is a set of reference stimuli  $r_j$  that form "representative samples" for similarity computation, e.g., equidistantly spaced orientations (Figure 4B1), the preferred inputs of V1 simple or complex cells (Figure 4B2), or specific branching points in a maze (Figure 4B3). The second element is a similarity measure  $f$  between pairs of stimuli, e.g., the Euclidean distance of angles (Figure 4B1), cosine similarity in pixel space (Figure 4B2), or a shortest-path distance (Figure 4B3). The third element is a non-negative function  $\phi$  that transforms the similarity between an input stimulus and the reference stimuli into component activations, capturing how stimulus similarities influence the stimulus familiarity (green line in Figures 4B1–4B3). Thus, with a single probabilistic framework for novelty, we can model novelty computation and its modulation by stimulus similarities across different tasks and datasets.

While several studies<sup>23,25,31,51</sup> have tested the impact of recency and repetition on the novelty of a stimulus, systematic work on how generalization across stimuli impacts novelty signals is absent. We propose a new experimental protocol ("sum-of-parts" experiment, Figures 4C1–4C3) to close this gap. Like in the  $M$ -experiment of Homann et al.,<sup>51</sup> a sequence of  $M$  familiar images ( $M = 3, 6, 9, 12$ ) is presented for  $L$  repetitions before a novel image  $X$  is shown. In contrast to the previous experiments, however, the  $M_{\max}$  familiar stimuli are chosen such that, after a controlled *transformation*, they make up equal parts of the novel image  $X$  (Figure 4C1;  $M_{\max} = 12$ ). The degree of similarity between the familiar sequence and the image  $X$  is controlled by the amplitude of the transformation (here: rotation of the Gabors in the image  $X$ ). Importantly, the stimuli in the familiar sequence are fully distinct (zero pairwise similarity), but each of them has equal similarity to the novel image  $X$  (Figure 4C2, top). The higher the rotation of the novel image  $X$ , the lower the similarity between familiar images in the sequence and  $X$  (Figure 4C2, bottom). The same experiment can also be performed, e.g., with more naturalistic images, as long as pairwise similarities between the familiar parts of the novel image are sufficiently low. Using similarity-based novelty, we predict the novelty responses for different lengths  $M$  of the familiar



**Figure 4. A framework for new theory-driven experiments**

(A) Similarity-based novelty represents the familiarity of stimuli as a probabilistic mixture model, using (1) non-negative component functions that capture similarities between stimuli, and (2) component weights encoding the empirical frequency of each component in past observations.

(B1–B3) The similarity-based novelty framework can be adapted to modeling neural data and behavior across different tasks, enabling task- and stimulus-specific novelty predictions with a single probabilistic framework.

(C1–C3) Schematic of a proposed experimental protocol. (C1) A sequence of  $M$  familiar images ( $M = 3, 6, 9, 12$ ) is presented for  $L$  repetitions, before a novel image  $X$  is shown. All 12 familiar stimuli are chosen such that, after a controlled transformation (here: rotation of the Gabors), they contribute equal parts to the novel image  $X$ . The degree of similarity between familiar images and the image  $X$  is controlled through the amplitude of the transformation (here: rotation angle). (C2, top) Equal pairwise similarity between the  $M$  images in the familiar sequence (here: 0 similarity, shown for  $M = 3$ ), and between each familiar image and  $X$ . (C2, bottom) The larger the rotation angle of the Gabors in  $X$ , the lower the pairwise similarity between any familiar image and  $X$ . Circles denote the rotation values for which novelty predictions were computed in C3. (C3) Novelty signals as predicted by similarity-based novelty for different values of  $M$  and different angles of rotation of  $X$ . For high similarity (low rotation) between familiar and novel images, novelty responses decrease with  $M$  (dark blue, "generalization-dominant" regime). For low similarity (high rotation), novelty responses increase with  $M$  (light blue, "recency-dominant" regime).

See also Figure S6.

sequence ( $M = 3, 6, 9, 12$ ) and for different levels of similarity between the familiar and novel images (Figure 4C3). Similarity-based novelty predicts that, for high similarity between familiar and novel images, novelty responses *decrease with*  $M$  due to similarity-based generalization of familiarity. However, for low similarity between familiar and novel images, similarity-based novelty predicts the opposite trend, i.e., that novelty responses *increase with*  $M$  due to the memory leakage in longer sequences. This suggests that, as the similarity between familiar and novel images increases, the novelty responses gradually shift from

the "generalization"-dominant regime (high similarity, dark blue in Figure 4C3) to the "recency"-dominant regime (low similarity, light blue in Figure 4C3). These predictions of similarity-based novelty are robust with respect to its model parameters (Figure S6). Our sum-of-parts protocol exemplifies how similarity-based novelty enables new theory-driven experiment design. When used with appropriate naturalistic stimuli, our protocol can also be extended to brain areas beyond V1 and facilitate insights into novelty computation across the visual processing hierarchy.

## DISCUSSION

We developed a model of novelty computation that accounts for the impact of stimulus similarities on novelty computation. We showed that similarity-based novelty successfully captures both novelty responses in mouse V1<sup>51</sup> and novelty-driven exploration of mice.<sup>59</sup> We further used similarity-based novelty to make testable predictions in a novel experimental protocol.

Our results suggest that similarities between unfamiliar and familiar stimuli significantly influence both neural and behavioral signatures of novelty-related processing in the brain. Specifically, (1) similarity to familiar stimuli attenuates the novelty response to a previously unobserved stimulus, and (2) generalization of familiarity across nearby spatial locations makes exploration more efficient by reducing the time spent exploring “similar” (= close-by) regions in the maze. Our model unifies these findings by showing that they can both arise due to the modulation of novelty signals by stimulus similarities. Building on this, we predict that novelty responses in other sensory modalities<sup>4,12,34,63</sup> and in the hippocampus<sup>6,8,11</sup> as well as downstream novelty signaling in salience-related regions<sup>5,24,25,35,64</sup> could also be influenced by generalization across stimuli. If this prediction is true, it would have important implications for the experimental study of novelty: novel input stimuli should either be chosen as almost perfectly distinct (e.g., Ghazizadeh and Hikosaka<sup>7</sup>, Xu et al.<sup>20</sup>, and Zhang et al.<sup>35</sup>) to eliminate the impact of stimulus similarities on novelty signals, or their similarity should be controlled for as an experimental variable.

Our similarity-based novelty model also allows us to reinterpret existing experimental findings. For example, a study by Montgomery<sup>16</sup> investigating the impact of luminance on mice exploration in otherwise identical mazes finds that the degree to which mice explore a new maze increases as a function of luminosity difference (i.e., dissimilarity) between the new maze and the previously explored one. Along the same lines, DeBaene and Vogels<sup>33</sup> report that neuronal adaptation in the monkey inferior temporal (IT) cortex significantly depends on the difference between adaptor and test stimulus with respect to features such as object shape and location. This suggests possible circuit implementations through which stimulus similarities could modulate novelty signals (STAR Methods).

Insights from ML support the functional need to generalize novelty across similar stimuli,<sup>32</sup> because two problems need to be solved: (1) mapping a diversity of high-dimensional inputs to reasonable abstract states (“view invariance”), and (2) grouping states together based on their shared characteristics (“state invariance”). Similarity-based novelty solves the second problem by expressing similarities in a given representation space by (potentially overlapping) component functions. Importantly, similarity-based novelty does not aim to solve the problem of learning a suitable stimulus representation, but it provides a flexible framework to compute novelty signals *given* an abstract feature representation. In our work, we constructed stimulus representations based on experimental knowledge and hypotheses. In general, however, similarity-based novelty is not limited to handcrafted representations but can also be used, e.g., in the latent space of trained deep networks with “brain-like” representations,<sup>65,66</sup> or on representations learned with biologically

plausible learning rules.<sup>67–69</sup> Using learned representations will remove design choices that we had to make in our handcrafted representation. For example, while our V1 results are stable with respect to a change in component width (Figure S3), our behavioral exploration results do depend on the granularity level, i.e., the width of the spatial components. In this case, optimal granularity could arise naturally from a learned model of place cells.

Some ML algorithms solve both the view invariance problem (i.e., finding a good representation) and the state invariance problem (i.e., grouping similar representational states together) simultaneously by training deep networks to estimate the state familiarity, e.g., via pseudo-counts,<sup>36,37</sup> hashing to discrete bins,<sup>38,70</sup> or construction of meaningful latent spaces.<sup>41</sup> The fact that most of these methods are trained “end to end” is both an advantage, because they learn an appropriate state representation simultaneously with the familiarity density, and a disadvantage, because they require learning a separate density estimation network using backpropagation instead of computing novelty from existing representations, such as place cells.

One of the central advantages of similarity-based novelty is its biologically interpretable (local) learning rule. Specifically, the weights that capture the empirical frequency of each component are updated with a multiplicative learning rule. While similarity-based novelty does not make explicit predictions about which circuit structure has to underlie novelty computation, its update rule for the novelty weights is consistent (STAR Methods) with several existing circuit mechanisms proposed for novelty computation, including input adaptation and short-term synaptic depression,<sup>33,50–52</sup> inhibitory circuits,<sup>55</sup> or Hebbian/anti-Hebbian multiplicative plasticity,<sup>53,54</sup> and inspires new mechanistic hypotheses (STAR Methods). As such, similarity-based novelty proposes a normative view on a wide range of mechanistic novelty models that provide area- and modality-specific mechanistic implementations of novelty computation.

Novelty computation in the brain is closely intertwined with and influenced by other cognitive computations such as attention,<sup>12,15,71</sup> contextual processing,<sup>72</sup> or the computation of other perceptual saliency signals, e.g., surprise.<sup>35,73</sup> In particular, while Homann et al. performed control experiments to exclude an influence of other intrinsic motivations such as surprise on the recorded V1 responses,<sup>51</sup> surprise and novelty influence brain responses jointly in many tasks,<sup>20,35</sup> and even modulate each other.<sup>35,73</sup> Such a modulatory effect of surprise on novelty could be included in the similarity-based novelty framework, e.g., via a modulation of the novelty update rate (similar to Liakoni et al.<sup>74</sup>).

While novelty has been shown to be a dominant drive of human exploration in unfamiliar environments,<sup>20,22</sup> a diversity of other intrinsic reward-like signals are involved in driving exploration<sup>11,16–18,20–22,25,75,76</sup> (also see several recent reviews<sup>64,77–81</sup>), especially when extrinsic rewards such as food and money are absent. The relative importance of different motivational signals in explaining behavior depends on multiple factors, including the temporal stages of exploration<sup>20</sup> as well as the exploration objectives and features of the task and environment.<sup>82</sup> Our findings for novelty suggest that stimulus similarities could be another aspect that substantially influences the computation and behavioral impact of intrinsic motivational signals and that is currently

understudied in neuroscience. Extending our similarity-based approach to other intrinsic motivational signals, such as surprise, could thus provide a valuable tool to quantify and model brain signals and behavior in more complex naturalistic tasks.

In conclusion, we propose a computational framework to model novelty computation in the brain (similarity-based novelty) that allows us to assess the impact of similarity-driven generalization on neural novelty responses and novelty-driven behavior. Our approach opens new possibilities for future theory-driven experiment design.

### RESOURCE AVAILABILITY

#### Lead contact

Requests for further information and resources should be directed to and will be fulfilled by the lead contact, Sophia Becker ([sophia.becker@epfl.ch](mailto:sophia.becker@epfl.ch)).

#### Materials availability

This study did not generate new, unique reagents.

#### Data and code availability

- This paper analyzes existing, publicly available experimental data. The behavior dataset published by Rosenberg et al.<sup>59</sup> is accessible under the [10.22002/D1.2031](https://doi.org/10.22002/D1.2031).<sup>83</sup> The V1 dataset was extracted from the figures in Homann et al.<sup>51</sup> and has been deposited under the DOI [10.5281/zenodo.18092959](https://doi.org/10.5281/zenodo.18092959), together with all original code.
- All original code for model simulations and analysis has been deposited at Zenodo under the DOI [10.5281/zenodo.18092959](https://doi.org/10.5281/zenodo.18092959) and is publicly available as of the date of publication.<sup>84</sup> Model simulation data have been deposited at Zenodo under the DOI [10.5281/zenodo.18117791](https://doi.org/10.5281/zenodo.18117791).
- Any additional information required to reanalyze the data reported in this work paper is available from the [lead contact](#) upon request.

### ACKNOWLEDGMENTS

This research was supported by Swiss National Science Foundation grant nos. 200020\_184615, 200020\_207426, and 200021\_236436. The authors would like to thank Johanni Brea, Martin Barry, Flavio Martinelli, and all lab members for insightful discussions.

### AUTHOR CONTRIBUTIONS

Conceptualization, S.B., A.M., and W.G.; methodology, S.B., A.M., and W.G.; software, investigation, visualization, and validation, S.B.; writing – original draft, S.B.; writing – review & editing, S.B., A.M., and W.G.; supervision, A.M. and W.G.; funding acquisition, W.G.

### DECLARATION OF INTERESTS

The authors declare no competing interests.

### STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- [KEY RESOURCES TABLE](#)
- [EXPERIMENTAL MODEL AND STUDY PARTICIPANT DETAILS](#)
  - Mouse data
- [METHOD DETAILS](#)
  - Models of novelty computation
  - Count-based novelty w/wo leaky stimulus memory
  - Similarity-based novelty w/wo leaky stimulus memory
  - Count-based as a special case of similarity-based novelty

- Links of similarity-based novelty to circuit implementations of novelty
- Toy experiment: Gabors sequences of different similarity
- Passive viewing task
- Active exploration task
- Sum-of-parts experiment
- Derivations

### ● QUANTIFICATION AND STATISTICAL ANALYSIS

### SUPPLEMENTAL INFORMATION

Supplemental information can be found online at <https://doi.org/10.1016/j.neuron.2026.01.007>.

Received: February 26, 2025

Revised: October 21, 2025

Accepted: January 7, 2026

### REFERENCES

1. Rainer, G., and Miller, E.K. (2002). Timecourse of object-related neural activity in the primate prefrontal cortex during a short-term memory task. *Eur. J. Neurosci.* *15*, 1244–1254. <https://doi.org/10.1046/j.1460-9568.2002.01958.x>.
2. Lisman, J.E., and Grace, A.A. (2005). The hippocampal-VTA loop: controlling the entry of information into long-term memory. *Neuron* *46*, 703–713. <https://doi.org/10.1016/j.neuron.2005.05.002>.
3. Takeuchi, T., Duszkiwicz, A.J., Sonneborn, A., Spooner, P.A., Yamasaki, M., Watanabe, M., Smith, C.C., Fernández, G., Deisseroth, K., Greene, R.W., and Morris, R.G.M. (2016). Locus coeruleus and dopaminergic consolidation of everyday memory. *Nature* *537*, 357–362. <https://doi.org/10.1038/nature19325>.
4. Morrens, J., Aydin, Ç., Janse van Rensburg, A., Esquívelzeta Rabell, J., and Haesler, S. (2020). Cue-evoked dopamine promotes conditioned responding during learning. *Neuron* *106*, 142–153.e7. <https://doi.org/10.1016/j.neuron.2020.01.012>.
5. Kutlu, M.G., Zachry, J.E., Melugin, P.R., Tat, J., Cajigas, S., Isiktas, A.U., Patel, D.D., Siciliano, C.A., Schoenbaum, G., Sharpe, M.J., et al. (2022). Dopamine signaling in the nucleus accumbens core mediates latent inhibition. *Nat. Neurosci.* *25*, 1071–1081. <https://doi.org/10.1038/s41593-022-01126-1>.
6. Priestley, J.B., Bowler, J.C., Rolotti, S.V., Fusi, S., and Losonczy, A. (2022). Signatures of rapid plasticity in hippocampal CA1 representations during novel experiences. *Neuron* *110*, 1978–1992.e6. <https://doi.org/10.1016/j.neuron.2022.03.026>.
7. Ghazizadeh, A., and Hikosaka, O. (2022). Saliency memories formed by value, novelty and aversiveness jointly shape object responses in the prefrontal cortex and basal ganglia. *Nat. Commun.* *13*, 6338. <https://doi.org/10.1038/s41467-022-33514-3>.
8. Gómez-Ocádiz, R., Trippa, M., Zhang, C.L., Posani, L., Cocco, S., Monasson, R., and Schmidt-Hieber, C. (2022). A synaptic signal for novelty processing in the hippocampus. *Nat. Commun.* *13*, 4122. <https://doi.org/10.1038/s41467-022-31775-6>.
9. Tamboli, S., Singh, S., Topolnik, D., El Amine Barkat, M.E.A., Radhakrishnan, R., Guet-McCreight, A., and Topolnik, L. (2024). Mouse hippocampal CA1 VIP interneurons detect novelty in the environment and support recognition memory. *Cell Rep.* *43*, 114115. <https://doi.org/10.1016/j.celrep.2024.114115>.
10. Bunzeck, N., and Düzél, E. (2006). Absolute coding of stimulus novelty in the human substantia nigra/VTA. *Neuron* *51*, 369–379. <https://doi.org/10.1016/j.neuron.2006.06.021>.
11. Düzél, E., Bunzeck, N., Guitart-Masip, M., and Düzél, S. (2010). Novelty-related motivation of anticipation and exploration by dopamine

- (NOMAD): implications for healthy aging. *Neurosci. Biobehav. Rev.* 34, 660–669. <https://doi.org/10.1016/j.neubiorev.2009.08.006>.
12. Foley, N.C., Jangraw, D.C., Peck, C., and Gottlieb, J. (2014). Novelty enhances visual salience independently of reward in the parietal lobe. *J. Neurosci.* 34, 7947–7957. <https://doi.org/10.1523/JNEUROSCI.4171-13.2014>.
  13. Ghazizadeh, A., Griggs, W., and Hikosaka, O. (2016). Ecological origins of object salience: Reward, uncertainty, aversiveness, and novelty. *Front. Neurosci.* 10, 378. <https://doi.org/10.3389/fnins.2016.00378>.
  14. Tamura, K., Takeda, M., Setsuie, R., Tsubota, T., Hirabayashi, T., Miyamoto, K., and Miyashita, Y. (2017). Conversion of object identity to object-general semantic value in the primate temporal cortex. *Science* 357, 687–692. <https://doi.org/10.1126/science.aan4800>.
  15. Ernst, D., Becker, S., and Horstmann, G. (2020). Novelty competes with saliency for attention. *Vis. Res.* 168, 42–52. <https://doi.org/10.1016/j.visres.2020.01.004>.
  16. Montgomery, K.C. (1953). Exploratory behavior as a function of ‘similarity’ of stimulus situations. *J. Comp. Physiol. Psychol.* 46, 129–133. <https://doi.org/10.1037/h0055101>.
  17. Berlyne, D.E. (1966). Curiosity and exploration. *Science* 153, 25–33. <https://doi.org/10.1126/science.153.3731.25>.
  18. Berlyne, D.E. (1950). Novelty and curiosity as determinants of exploratory behaviour. *Br. J. Psychol. Gen. Sect.* 41, 68–80. <https://doi.org/10.1111/j.2044-8295.1950.tb00262.x>.
  19. Menegas, W., Babayan, B.M., Uchida, N., and Watabe-Uchida, M. (2017). Opposite initialization to novel cues in dopamine signaling in ventral and posterior striatum in mice. *eLife* 6, e21886. <https://doi.org/10.7554/eLife.21886>.
  20. Xu, H.A., Modirshanechi, A., Lehmann, M.P., Gerstner, W., and Herzog, M.H. (2021). Novelty is not Surprise: Human exploratory and adaptive behavior in sequential decision-making. *PLoS Comput. Biol.* 17, e1009070. <https://doi.org/10.1371/journal.pcbi.1009070>.
  21. Dezza, I.C., Noel, X., Cleeremans, A., and Yu, A.J. (2021). Distinct motivations to seek out information in healthy individuals and problem gamblers. *Transl. Psychiatry* 11, 408. <https://doi.org/10.1038/s41398-021-01523-3>.
  22. Modirshanechi, A., Lin, W.H., Xu, H.A., Herzog, M.H., and Gerstner, W. (2025). Novelty as a drive of human exploration in complex stochastic environments. *Proc. Natl. Acad. Sci. USA* 122, e2502193122. <https://doi.org/10.1073/pnas.2502193122>.
  23. Dubey, R., and Griffiths, T.L. (2020). Reconciling novelty and complexity through a rational analysis of curiosity. *Psychol. Rev.* 127, 455–476. <https://doi.org/10.1037/rev0000175>.
  24. Akiti, K., Tsutsui-Kimura, I., Xie, Y., Mathis, A., Markowitz, J.E., Anyoha, R., Datta, S.R., Mathis, M.W., Uchida, N., and Watabe-Uchida, M. (2022). Striatal dopamine explains novelty-induced behavioral dynamics and individual variability in threat prediction. *Neuron* 110, 3789–3804.e9. <https://doi.org/10.1016/j.neuron.2022.08.022>.
  25. Ogasawara, T., Sogukpinar, F., Zhang, K., Feng, Y.Y., Pai, J., Jezzini, A., and Monosov, I.E. (2022). A primate temporal cortex–zona incerta pathway for novelty seeking. *Nat. Neurosci.* 25, 50–60. <https://doi.org/10.1038/s41593-021-00950-1>.
  26. Cockburn, J., Man, V., Cunningham, W.A., and O’Doherty, J.P. (2022). Novelty and uncertainty regulate the balance between exploration and exploitation through distinct mechanisms in the human brain. *Neuron* 110, 2691–2702.e8. <https://doi.org/10.1016/j.neuron.2022.05.025>.
  27. Molas, S., Freels, T.G., Zhao-Shea, R., Lee, T., Gimenez-Gomez, P., Barbini, M., Martin, G.E., and Tapper, A.R. (2024). Dopamine control of social novelty preference is constrained by an interpeduncular–tegmentum circuit. *Nat. Commun.* 15, 2891. <https://doi.org/10.1038/s41467-024-47255-y>.
  28. Ahmadi, M., Houba, J.H.W., van Vierbergen, J.F.M., Giannouli, M., Gimenez, G.A., van Weeghel, C., Darbanfouladi, M., Shirazi, M.Y., Dziubek, J., Kacem, M., et al. (2021). A cell type-specific cortico-subcor-tical brain circuit for investigatory and novelty-seeking behavior. *Science* 372, eabe9681. <https://doi.org/10.1126/science.abe9681>.
  29. Gershman, S.J., and Niv, Y. (2015). Novelty and inductive generalization in human reinforcement learning. *Top. Cogn. Sci.* 7, 391–415. <https://doi.org/10.1111/tops.12138>.
  30. Gershman, S.J., Monfils, M.H., Norman, K.A., and Niv, Y. (2017). The computational nature of memory modification. *eLife* 6, e23763. <https://doi.org/10.7554/eLife.23763>.
  31. Maheu, M., Dehaene, S., and Meyniel, F. (2019). Brain signatures of a multiscale process of sequence learning in humans. *eLife* 8, e41541. <https://doi.org/10.7554/eLife.41541>.
  32. Jaegle, A., Mehrpour, V., and Rust, N. (2019). Visual novelty, curiosity, and intrinsic reward in machine learning and the brain. *Curr. Opin. Neurobiol.* 58, 167–174. <https://doi.org/10.1016/j.conb.2019.08.004>.
  33. De Baene, W.D., and Vogels, R. (2009). Effects of adaptation on the stimulus selectivity of macaque inferior temporal spiking activity and local field potentials. *Cereb. Cortex* 20, 2145–2165. <https://doi.org/10.1093/cercor/bhp277>.
  34. Meyer, T., and Rust, N.C. (2018). Single-exposure visual memory judgments are reflected in inferotemporal cortex. *eLife* 7, e32259. <https://doi.org/10.7554/eLife.32259>.
  35. Zhang, K., Bromberg-Martin, E.S., Sogukpinar, F., Kocher, K., and Monosov, I.E. (2022). Surprise and recency in novelty detection in the primate brain. *Curr. Biol.* 32, 2160–2173.e6. <https://doi.org/10.1016/j.cub.2022.03.064>.
  36. Bellemare, M., Srinivasan, S., Ostrovski, G., Schaul, T., Saxton, D., and Munos, R. (2016). Unifying count-based exploration and intrinsic motivation. In *NIPS’16: Proceedings of the 30th International Conference on Neural Information Processing Systems*, pp. 1471–1479. <https://doi.org/10.5555/3157096.3157262>.
  37. Ostrovski, G., Bellemare, M.G., van den Oord, A., and Munos, R. (2017). Count-based exploration with neural density models. In *Proceedings of the 34th International Conference on Machine Learning*, pp. 2721–2730.
  38. Tang, H., Houthoofd, R., Foote, D., Stooke, A., Chen, X., Duan, Y., Schulman, J., De Turck, F., and Abbeel, P. (2017). #Exploration: A Study of Count-Based Exploration for Deep Reinforcement Learning. In *NIPS’17: Proceedings of the 31st International Conference on Neural Information Processing Systems*, 30, pp. 2753–2762. <https://doi.org/10.5555/3294996.3295035>.
  39. Burda, Y., Edwards, H., Storkey, A., and Klimov, O. (2018). Exploration by random network distillation. Preprint at arXiv. <https://doi.org/10.48550/arXiv.1810.12894>.
  40. Savinov, N., Raichuk, A., Marinier, R., Vincent, D., Pollefeys, M., Lillicipap, T., and Gelly, S. (2019). Episodic curiosity through reachability. In *International Conference on Learning Representations (ICLR)* <https://openreview.net/pdf?id=SkeK3s0qKQ>.
  41. Tao, R.Y., François-Lavet, V., and Pineau, J. (2020). Novelty search in representational space for sample efficient exploration. In *34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, 33, pp. 8114–8126. <https://proceedings.neurips.cc/paper/2020/file/5ca41a86596a5ed567d15af0be224952-Paper.pdf>.
  42. Machado, M.C., Bellemare, M.G., and Bowling, M. (2020). Count-based exploration with the successor representation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 34, pp. 5125–5133. <https://doi.org/10.1609/aaai.v34i04.5955>.
  43. Modirshanechi, A., Becker, S., Brea, J., and Gerstner, W. (2023). Surprise and novelty in the brain. *Curr. Opin. Neurobiol.* 82, 102758. <https://doi.org/10.1016/j.conb.2023.102758>.
  44. Kakade, S., and Dayan, P. (2002). Dopamine: generalization and bonuses. *Neural Netw.* 15, 549–559. [https://doi.org/10.1016/S0893-6080\(02\)00048-5](https://doi.org/10.1016/S0893-6080(02)00048-5).
  45. Barto, A., Mirolli, M., and Baldassarre, G. (2013). Novelty or surprise? *Front. Psychol.* 4, 907. <https://doi.org/10.3389/fpsyg.2013.00907>.

46. Daw, N.D., Gershman, S.J., Seymour, B., Dayan, P., and Dolan, R.J. (2011). Model-based influences on humans' choices and striatal prediction errors. *Neuron* 69, 1204–1215. <https://doi.org/10.1016/j.neuron.2011.02.027>.
47. Bogacz, R., and Brown, M.W. (2003). Comparison of computational models of familiarity discrimination in the perirhinal cortex. *Hippocampus* 13, 494–524. <https://doi.org/10.1002/hipo.10093>.
48. Wang, L., McAlonan, K., Goldstein, S., Gerfen, C.R., and Krauzlis, R.J. (2020). A causal role for mouse superior colliculus in visual perceptual decision-making. *J. Neurosci.* 40, 3768–3782. <https://doi.org/10.1523/JNEUROSCI.2642-19.2020>.
49. Bishop, C.M., and Nasrabadi, N.M. (2006). *Pattern Recognition and Machine Learning* (Springer).
50. Mill, R., Coath, M., Wennekers, T., and Denham, S.L. (2011). A neurocomputational model of stimulus-specific adaptation to oddball and markov sequences. *PLoS Comput. Biol.* 7, e1002117. <https://doi.org/10.1371/journal.pcbi.1002117>.
51. Homann, J., Koay, S.A., Chen, K.S., Tank, D.W., and Berry, M.J. (2022). Novel stimuli evoke excess activity in the mouse primary visual cortex. *Proc. Natl. Acad. Sci. USA* 119, e2108882119. <https://doi.org/10.1073/pnas.2108882119>.
52. Aitken, K., Campagnola, L., Garrett, M.E., Olsen, S.R., and Mihalas, S. (2024). Simple synaptic modulations implement diverse novelty computations. *Cell Rep.* 43, 114188. <https://doi.org/10.1016/j.celrep.2024.114188>.
53. Lim, S., McKee, J.L., Woloszyn, L., Amit, Y., Freedman, D.J., Sheinberg, D.L., and Brunel, N. (2015). Inferring learning rules from distributions of firing rates in cortical neurons. *Nat. Neurosci.* 18, 1804–1810. <https://doi.org/10.1038/nn.4158>.
54. Mohan, K., Pereira-Obilinovic, U., Srednyak, S., Amit, Y., Brunel, N., and Freedman, D. (2024). Visual familiarity learning at multiple timescales in the primate inferotemporal cortex. Preprint at bioRxiv. <https://doi.org/10.1101/2024.01.05.574412>.
55. Schulz, A., Miehl, C., Berry, M.J., and Gjorgjieva, J. (2021). The generation of cortical novelty responses through inhibitory plasticity. *eLife* 10, e65309. <https://doi.org/10.7554/eLife.65309>.
56. Tyulmankov, D., Yang, G.R., and Abbott, L.F. (2022). Meta-learning synaptic plasticity and memory addressing for continual familiarity detection. *Neuron* 110, 544–557.e8. <https://doi.org/10.1016/j.neuron.2021.11.009>.
57. Hubel, D.H., and Wiesel, T.N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J. Physiol.* 160, 106–154. <https://doi.org/10.1113/jphysiol.1962.sp006837>.
58. Priebe, N.J. (2016). Mechanisms of orientation selectivity in the primary visual cortex. *Annu. Rev. Vis. Sci.* 2, 85–107. <https://doi.org/10.1146/annurev-vision-111815-114456>.
59. Rosenberg, M., Zhang, T., Perona, P., and Meister, M. (2021). Mice in a labyrinth show rapid learning, sudden insight, and efficient exploration. *eLife* 10, e66175. <https://doi.org/10.7554/eLife.66175>.
60. Kass, R.E., and Raftery, A.E. (1995). Bayes factors. *J. Am. Stat. Assoc.* 90, 773–795. <https://doi.org/10.1080/01621459.1995.10476572>.
61. Jeffreys, H. (1998). *The Theory of Probability* (Oxford University Press).
62. Eliav, T., Maimon, S.R., Aljadeff, J., Tsodyks, M., Ginosar, G., Las, L., and Ulanovsky, N. (2021). Multiscale representation of very large environments in the hippocampus of flying bats. *Science* 372, eabg4020. <https://doi.org/10.1126/science.abg4020>.
63. Hershenhoren, I., Taaseh, N., Antunes, F.M., and Nelken, I. (2014). Intracellular correlates of stimulus-specific adaptation. *J. Neurosci.* 34, 3303–3319. <https://doi.org/10.1523/JNEUROSCI.2166-13.2014>.
64. Monosov, I.E. (2024). Curiosity: primate neural circuits for novelty and information seeking. *Nat. Rev. Neurosci.* 25, 195–208. <https://doi.org/10.1038/s41583-023-00784-9>.
65. Yamins, D.L.K., and DiCarlo, J.J. (2016). Using goal-driven deep learning models to understand sensory cortex. *Nat. Neurosci.* 19, 356–365. <https://doi.org/10.1038/nn.4244>.
66. Schrimpf, M., Kubilius, J., Lee, M.J., Ratan Murty, N.A.R., Ajemian, R., and DiCarlo, J.J. (2020). Integrative benchmarking to advance neurally mechanistic models of human intelligence. *Neuron* 108, 413–423. <https://doi.org/10.1016/j.neuron.2020.07.040>.
67. Illing, B., Ventura, J., Bellec, G., and Gerstner, W. (2021). Local plasticity rules can learn deep representations using self-supervised contrastive predictions. In 35th Conference on Neural Information Processing Systems (NeurIPS 2021) <https://proceedings.neurips.cc/paper/2021/file/feade1d2047977cd0cefdafc40175a99-Paper.pdf>.
68. Halvagal, M.S., and Zenke, F. (2023). The combination of hebbian and predictive plasticity learns invariant object representations in deep sensory networks. *Nat. Neurosci.* 26, 1906–1915. <https://doi.org/10.1038/s41593-023-01460-y>.
69. Strösslin, T., Sheynikhovich, D., Chavarriga, R., and Gerstner, W. (2005). Robust self-localisation and navigation based on hippocampal place cells. *Neural Netw.* 18, 1125–1140. <https://doi.org/10.1016/j.neunet.2005.08.012>.
70. Dasgupta, S., Hattori, D., and Navlakha, S. (2022). A neural theory for counting memories. *Nat. Commun.* 13, 5961. <https://doi.org/10.1038/s41467-022-33577-2>.
71. Gottlieb, J. (2012). Attention, learning, and the value of information. *Neuron* 76, 281–295. <https://doi.org/10.1016/j.neuron.2012.09.034>.
72. Nitzan, N., Bennett, C., Movshon, J.A., Olsen, S.R., and Buzsáki, G. (2024). Mixing novel and familiar cues modifies representations of familiar visual images and affects behavior. *Cell Rep.* 43, 114521. <https://doi.org/10.1016/j.celrep.2024.114521>.
73. Summerfield, C., Trittschuh, E.H., Monti, J.M., Mesulam, M.M., and Egner, T. (2008). Neural repetition suppression reflects fulfilled perceptual expectations. *Nat. Neurosci.* 11, 1004–1006. <https://doi.org/10.1038/nn.2163>.
74. Liakoni, V., Modirshanechi, A., Gerstner, W., and Brea, J. (2021). Learning in volatile environments with the Bayes factor surprise. *Neural Comput.* 33, 269–340. [https://doi.org/10.1162/neco\\_a\\_01352](https://doi.org/10.1162/neco_a_01352).
75. Brändle, F., Stocks, L.J., Tenenbaum, J.B., Gershman, S.J., and Schulz, E. (2023). Empowerment contributes to exploration behaviour in a creative video game. *Nat. Hum. Behav.* 7, 1481–1489. <https://doi.org/10.1038/s41562-023-01661-2>.
76. Ramírez-Ruiz, J., Grytskyy, D., Mastrogiuseppe, C., Habib, Y., and Moreno-Bote, R. (2024). Complex behavior from intrinsic motivation to occupy future action-state path space. *Nat. Commun.* 15, 6368. <https://doi.org/10.1038/s41467-024-49711-1>.
77. Gottlieb, J., and Oudeyer, P.Y. (2018). Towards a neuroscience of active sampling and curiosity. *Nat. Rev. Neurosci.* 19, 758–770. <https://doi.org/10.1038/s41583-018-0078-0>.
78. Dubey, R., and Griffiths, T.L. (2020). Understanding exploration in humans and machines by formalizing the function of curiosity. *Curr. Opin. Behav. Sci.* 35, 118–124. <https://doi.org/10.1016/j.cobeha.2020.07.008>.
79. Karayanni, M., and Nelken, I. (2022). Extrinsic rewards, intrinsic rewards, and non-optimal behavior. *J. Comput. Neurosci.* 50, 139–143. <https://doi.org/10.1007/s10827-022-00813-z>.
80. Modirshanechi, A., Kondrakiewicz, K., Gerstner, W., and Haesler, S. (2023). Curiosity-driven exploration: foundations in neuroscience and computational modeling. *Trends Neurosci.* 46, 1054–1066. <https://doi.org/10.1016/j.tins.2023.10.002>.
81. Poli, F., O'Reilly, J.X., Mars, R.B., and Hunnius, S. (2024). Curiosity and the dynamics of optimal exploration. *Trends Cogn. Sci.* 28, 441–453. <https://doi.org/10.1016/j.tics.2024.02.001>.
82. Gruaz, L., Modirshanechi, A., Becker, S., and Brea, J. (2025). Merits of curiosity: A simulation study. *Open Mind* 9, 1037–1065. <https://doi.org/10.1162/opmi.a.9>.
83. Meister, M. (2021). markusmeister/rosenberg-2021-repository 1.0.1 (1.0.1). CaltechDATA. <https://doi.org/10.22002/D1.2031>.
84. sobecker. (2025). sobecker/sim\_nov: v1.0 (v1.0). Zenodo. <https://doi.org/10.5281/zenodo.18092959>.

85. Nowlan, S.J. (1991). Soft competitive adaptation: Neural network learning algorithms based on fitting statistical mixtures. Ph.D. thesis (Carnegie Mellon University).
86. McLachlan, G.J., and Krishnan, T. (2008). The EM Algorithm and Extensions, 2E (John Wiley & Sons, Inc.). <https://doi.org/10.1002/9780470191613>.
87. Sutton, R.S., and Barto, A.G. (2018). *Reinforcement Learning: an Introduction* (MIT Press).
88. Kempster, R., Gerstner, W., and van Hemmen, J.L. (2001). Intrinsic stabilization of output rates by spike-based hebbian learning. *Neural Comput.* *13*, 2709–2741. <https://doi.org/10.1162/089976601317098501>.
89. Carandini, M., and Heeger, D.J. (2011). Normalization as a canonical neural computation. *Nat. Rev. Neurosci.* *13*, 51–62. <https://doi.org/10.1038/nrn3136>.
90. Williams, L.E., and Holtmaat, A. (2019). Higher-order thalamocortical inputs gate synaptic long-term potentiation via disinhibition. *Neuron* *101*, 91–102.e4. <https://doi.org/10.1016/j.neuron.2018.10.049>.
91. Canto-Bustos, M., Friason, F.K., Bassi, C., and Oswald, A.M.M. (2022). Disinhibitory circuitry gates associative synaptic plasticity in olfactory cortex. *J. Neurosci.* *42*, 2942–2950. <https://doi.org/10.1523/JNEUROSCI.1369-21.2021>.
92. Aceituno, P.V., de Haan, S., Loidl, R., Beumer, L., and Grewe, B.F. (2024). Challenging backpropagation: Evidence for target learning in the neocortex. Preprint at bioRxiv. <https://doi.org/10.1101/2024.04.10.588837>.
93. Niell, C.M., and Stryker, M.P. (2008). Highly selective receptive fields in mouse visual cortex. *J. Neurosci.* *28*, 7520–7536. <https://doi.org/10.1523/JNEUROSCI.0623-08.2008>.
94. Efron, B. (1982). Introduction. In *The Jackknife, the Bootstrap and Other Resampling Plans* (Society for Industrial and Applied Mathematics), pp. 1–3. <https://doi.org/10.1137/1.9781611970319.ch1>.
95. Rosenberg, M., Zhang, T., Perona, P., and Meister, M. (2021). Rosenberg-2021-repository. <https://github.com/markusmeister/Rosenberg-2021-Repository>.
96. Chentanez, N., Barto, A.G., and Singh, S.P. (2004). Intrinsically motivated reinforcement learning. *Advances in Neural Information Processing* *17*.
97. Strehl, A.L., and Littman, M.L. (2008). An analysis of model-based interval estimation for Markov decision processes. *J. Comput. Syst. Sci.* *74*, 1309–1331. <https://doi.org/10.1016/j.jcss.2007.08.009>.
98. Dempster, A.P., Laird, N.M., and Rubin, D.B. (1977). Maximum likelihood from incomplete data via the em algorithm. *J. R. Stat. Soc. B* *39*, 1–22. <https://doi.org/10.1111/j.2517-6161.1977.tb01600.x>.
99. Daw, N.D. (2011). Trial-by-trial data analysis using computational models. Decision making, affect, and learning. *Atten. Perform.* *XXIII*, 3–38. <https://doi.org/10.1093/acprof:oso/9780199600434.003.0001>.

## STAR★METHODS

### KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
<b>Deposited data</b>		
Model simulations for grid search (fit to neural data) and model robustness analysis	This paper	<a href="https://doi.org/10.5281/zenodo.18117791">https://doi.org/10.5281/zenodo.18117791</a>
Simulations of best models (fit to behavior data)	This paper	<a href="https://doi.org/10.5281/zenodo.18117791">https://doi.org/10.5281/zenodo.18117791</a>
V1 recordings	Homann et al. <sup>51</sup>	<a href="https://doi.org/10.1073/pnas.2108882119">https://doi.org/10.1073/pnas.2108882119</a> (figures) and <a href="https://doi.org/10.5281/zenodo.18092959">https://doi.org/10.5281/zenodo.18092959</a> (extracted data)
Behavior recordings	Rosenberg et al. <sup>59</sup>	<a href="https://github.com/markusmeister/Rosenberg-2021-Repository">https://github.com/markusmeister/Rosenberg-2021-Repository</a> and <a href="https://doi.org/10.22002/D1.2031">https://doi.org/10.22002/D1.2031</a>
<b>Software and algorithms</b>		
Similarity-based novelty	This paper	<a href="https://github.com/sobecker/sim_nov">https://github.com/sobecker/sim_nov</a> and <a href="https://doi.org/10.5281/zenodo.18092959">https://doi.org/10.5281/zenodo.18092959</a>
Python	Python Software Foundation	<a href="https://www.python.org/">https://www.python.org/</a>

### EXPERIMENTAL MODEL AND STUDY PARTICIPANT DETAILS

#### Mouse data

We reused published data from experiments with mice performing (i) a passive viewing task with familiar and novel stimuli (see Homann et al.<sup>51</sup> for detailed experimental background), and (ii) a free exploration task in a novel maze (see Rosenberg et al.<sup>59</sup> for detailed experimental background).

### METHOD DETAILS

#### Models of novelty computation

As described in the main text (Equation 3), we define the novelty  $N^{(t+1)}(s)$  of a stimulus  $s$  at time  $t + 1$  as the negative logarithm of the "familiarity"  $p^{(t+1)}(s)$ :

$$N^{(t+1)}(s) = -\log p^{(t+1)}(s), \quad (\text{Equation 7})$$

where the familiarity density  $p^{(t+1)}$  is a parametric or non-parametric estimate of the "true" frequency density  $p_*^{(t+1)}$  at time point  $t + 1$ . The true frequency density is based on all observations that were made prior to time point  $t + 1$  (i.e. excluding the observation  $s_{t+1}$ ):

$$p_*^{(t+1)}(s) = \Pr(s_{t+1} = s | s_{1:t}). \quad (\text{Equation 8})$$

Both count-based and similarity-based familiarity are defined as maximum-likelihood (or maximum-a-posteriori, MAP) estimators of  $p_*^{(t+1)}$ , i.e., they assume a given parametric shape of the familiarity,  $p^{(t+1)} = p(\cdot | \theta^{(t)})$ , whose parameters  $\theta^{(t)}$  at time point  $t + 1$  are chosen to maximize the log-likelihood of past stimulus observations  $s_{1:t}$ , given the prior  $p(\theta)$  over the parameters (see [derivations](#) for details):

$$\theta^{(t+1)} = \underset{\theta}{\operatorname{argmax}} \log p(s_{1:t} | \theta) p(\theta). \quad (\text{Equation 9})$$

The difference between count-based and similarity-based novelty are the assumptions each model makes about the space in which the observable stimuli  $s$  live, and the assumed parametric shape of the familiarity  $p$ . Count-based novelty assumes that all stimulus observations are drawn from a finite set of discrete stimuli,  $S = \{x_1, \dots, x_N\}$ , and that the probability of observing a given stimulus is captured by a categorical distribution. In contrast, similarity-based novelty allows a more general notion of stimulus space that includes both discrete and continuous stimulus spaces, and assumes that stimulus observations can be described by a mixture distribution (Equation 4).

### Count-based novelty w/wo leaky stimulus memory Non-leaky count-based novelty

Based on these assumptions, the basic count-based novelty estimator is given as

$$p^{(t+1)}(s) = \frac{C^{(t)}(s) + \varepsilon}{t + |S|\varepsilon}, \quad (\text{Equation 10})$$

where  $C^{(t)}(s)$  counts the number of times that the stimulus  $s \in S = \{x_1, \dots, x_N\}$  has been observed up to time  $t$ , and the terms  $\varepsilon$  and  $|S|\varepsilon$  arise from the prior distribution (see [derivations](#), also see Xu et al.<sup>20</sup>).

The expression for the familiarity  $p^{(t+1)}$  in [Equation 10](#) gives rise to the following *iterative* familiarity update (see [derivations](#)):

$$p^{(t+1)}(s) = (1 - \alpha_t)p^{(t)}(s) + \alpha_t \delta(s_t, s), \quad (\text{Equation 11})$$

for  $s \in S = \{x_1, \dots, x_N\}$  and with time-dependent learning rate

$$\alpha_t = \frac{1}{t + |S|\varepsilon}. \quad (\text{Equation 12})$$

### Leakiness

The definition of count-based familiarity in [Equation 10](#) does not yet account for the possibility of leaky memory, where long-ago stimulus observations are gradually forgotten. We can account for such leakiness in two different ways.

The first one ("leaky" count-based novelty) assumes that the stimulus counts  $C^{(t)}(s)$  *decay with a fixed factor*  $\alpha$  between two time steps ([Equation 1](#) in the main text), yielding the familiarity (also see [Equation 2](#) in the main text):

$$p^{(t+1)}(s) = \frac{C^{(t)}(s) + \varepsilon}{T_t + |S|\varepsilon}, \quad (\text{Equation 13})$$

where

$$T_t = \sum_{s \in S} C^{(t)}(s) = \sum_{t'=1}^t (1 - \alpha)^{t-t'} \quad (\text{Equation 14})$$

is the leaky time count normalizing the stimulus counts to obtain a frequency (see [derivations](#) for details). The decay parameter  $\alpha \in (0, 1)$  regulates the speed of memory decay: the larger  $\alpha$ , the faster the stimulus counts decay over time.

The second one ("fixed-rate" count-based novelty) arises from the iterative familiarity update in [Equation 11](#), where leaky stimulus memory is introduced by replacing the time-dependent learning rate  $\alpha_t$  with a *fixed learning rate*  $\eta \in (0, 1)$ . Note that the time-dependent learning rate  $\alpha_t$  decreases to zero over time, such that late stimulus observations do not 'overwrite' early stimulus observations in the empirical frequency estimate. When we choose a constant learning rate  $\eta$ , these early observations are gradually overwritten by later observations as the empirical frequency is continuously updated. Hence, the larger the fixed learning rate  $\eta$ , the faster the stimulus memory decays over time.

In the main text, we focus on the *leaky* count-based novelty model (see [Figure S4](#) for a comparison of the two leakiness types in the passive viewing task).

### Similarity-based novelty w/wo leaky stimulus memory Non-leaky similarity-based novelty

The central element of similarity-based novelty is the mixture model that defines the familiarity density  $p$  ([Equation 4](#) in the main text). Its weights at time point  $t$  are chosen as the maximum-a-posteriori (MAP) estimate of the sequence  $s_{1:t}$  of stimulus observations up to  $t$ :

$$\mathbf{w}^{(t)} = \underset{\mathbf{w}}{\operatorname{argmax}} \log \Pr(s_{1:t} | \mathbf{w}) \Pr(\mathbf{w}), \quad (\text{Equation 15})$$

where  $\mathbf{w}^{(t)} = (w_1^{(t)}, \dots, w_N^{(t)})^T$  is the vector of weights at time  $t$ , and  $\Pr(\mathbf{w})$  is a Dirichlet prior over the weights ([derivations](#)). We solve [Equation 15](#) using the incremental expectation-maximization (EM) algorithm,<sup>49,85,86</sup> resulting in a closed-form solution for the weights  $\mathbf{w}^{(t)}$ . The closed-form ("batch") solution for the weights can be reformulated as an online learning rule. The two weight update rules (batch and online) derived from the EM algorithms are given as follows.

- (i) **EM-derived weights (batch).** For each time point  $t$  at which we observe a stimulus  $s_t$ , we compute the E-step (estimation step) of the incremental EM algorithm, i.e. we compute "responsibilities"  $y_j^{(t)}$  that estimate how well a given component  $k_j$  of the mixture distribution at time  $t - 1$  captures the observation  $s_t$ :

$$y_j^{(t)} = \frac{w_j^{(t-1)} k_j(s_t)}{\sum_{j'=1}^N w_{j'}^{(t-1)} k_{j'}(s_t)} \quad (\text{Equation 16})$$

The initial weights are chosen uniformly as  $w_j^{(0)} = \frac{1}{N}$  for all  $j = 1, \dots, N$ . In the subsequent M-step (maximization step) for time point  $t$ , we compute the new weights  $w_j^{(t)}$  based on the responsibilities  $\gamma_j^{(t)}$ ,  $j = 1, \dots, N$ ,  $t' = 1, \dots, t$  that were estimated in all previous E-steps (Equation 16):

$$w_j^{(t)} = \frac{R_j^{(t)} + \varepsilon}{t + N\varepsilon} = \frac{1}{t + N\varepsilon} \left[ \sum_{t'=1}^t \gamma_j^{(t')} + \varepsilon \right], \quad (\text{Equation 17})$$

where  $\varepsilon$  is the uniform concentration parameter of the Dirichlet prior over the weights, and  $R_j^{(t)}$  denotes the (non-leaky) "soft" counter of the responsibilities of component  $j$  up to time  $t$ . The main difference to textbook incremental EM algorithms<sup>49</sup> is that at each iteration, the amount of data in the batch increases since an additional data point has been added to the sequence  $s_{1:t}$  (see [derivations](#) for details).

- (ii) **Online weight update.** The expression for the weights at time  $t$  (Equation 17) still depends on the information from previous E-steps through the responsibilities  $\gamma_j^{(t')}$ ,  $t' = 1, \dots, t - 1$ . However, we can reformulate Equation 17 to obtain an online weight update rule that computes  $w^{(t)}$  only based on the responsibilities from the current E-step at time  $t$  and the previous weights  $w^{(t-1)}$ :

$$w_j^{(t)} = \frac{1}{t + N\varepsilon} \left( \sum_{t'=1}^t \gamma_{j,t'}^{(t')} + \varepsilon \right) = w_j^{(t-1)} + \frac{1}{t + N\varepsilon} \left( \gamma_{j,t}^{(t)} - w_j^{(t-1)} \right), \quad (\text{Equation 18})$$

for all  $j = 1, \dots, N$ . The iterative update rule for the weights in Equation 18 gives rise to an iterative update of the familiarity  $p$ , analogous to count-based novelty (Equation 11):

$$p^{(t+1)}(s) = (1 - \alpha_t) p^{(t)}(s) + \alpha_t \sum_{j=1}^N \gamma_j^{(t)} k_j(s_t), \quad (\text{Equation 19})$$

where  $s_t$  is the current stimulus observation and  $\alpha_t = \frac{1}{t + N\varepsilon}$  is the time-dependent learning rate.

### Multiplicative weight updates

The weight update rule in Equation 18 can be reformulated as the following delta rule:

$$w_j^{(t)} = w_j^{(t-1)} + \alpha_t \delta_j^{(t)}, \quad (\text{Equation 20})$$

with error

$$\delta_j^{(t)} = \gamma_j^{(t)} - w_j^{(t-1)} = \left( \frac{k_j(s_t)}{p^{(t-1)}(s_t)} - 1 \right) w_j^{(t-1)} \quad (\text{Equation 21})$$

If we pull out a factor  $1/p^{(t-1)}(s_t)$  in front of the brackets, we see that the error in Equation 21 is proportional to the difference between the component activation and the familiarity of the current observation  $s_t$ , respectively. This means that the update rule increases (decreases) a given weight when its contribution to the current observation is higher (lower) than estimated by the current familiarity. Note that, while the weight update looks similar to the value updates of reinforcement learning algorithms,<sup>87</sup> the function implemented by novelty weights is fundamentally different: Reward values in RL estimate the *future expected* reward in a given state, while we estimate the *current* empirical stimulus frequency under a specific parametric assumption about the frequency density (mixture model). Furthermore, the weight update rule in Equation 20 is *multiplicative* since the error (Equation 21) depends on the previous weights  $w_j^{(t-1)}$  through a multiplication factor.

### Leakiness

Analogously to count-based novelty, we introduce two alternative notions of leakiness into similarity-based novelty to account for gradual forgetting of past stimulus observations.

The first one ("leaky" similarity-based novelty) assumes that, in the M-step of the EM-algorithm, the weights are computed as the *leaky* sum of past responsibilities (also see Equation 5 in the main text):

$$w_j^{(t)} = \frac{R_j^{(t)} + \varepsilon}{T_t + N\varepsilon} = \frac{1}{T_t + N\varepsilon} \left[ \sum_{t'=1}^t (1 - \alpha)^{t-t'} \gamma_j^{(t')} + \varepsilon \right], \quad (\text{Equation 22})$$

where  $R_j^{(t)}$  are the leaky soft counts of past responsibilities as described in Equation 6 in the main text. The leaky time count  $T_t$  is given as

$$T_t = \sum_{j=1}^N R_j^{(t)} = \sum_{t'=1}^t (1 - \alpha)^{t-t'}. \quad (\text{Equation 23})$$

The second notion of leakiness ("fixed-rate" similarity-based novelty) assumes a *fixed learning rate*  $\eta \in (0, 1)$  instead of the time-dependent learning rate  $\alpha_t$  in Equation 19, i.e.

$$\rho^{(t+1)}(s) = (1 - \eta)\rho^{(t)}(s) + \eta \sum_{j=1}^N \gamma_j^{(t)} k_j(s). \quad (\text{Equation 24})$$

Analogously to fixed-rate count-based novelty, the constant learning rate in Equation 24 causes early component observations (responsibilities) to be overwritten by late ones.

We focus on *leaky* similarity-based novelty in the main text, while comparisons with fixed-rate similarity-based novelty for the passive viewing task are included in Figure S4. Below, we show that the fixed-rate variant of similarity-based novelty gives rise to possible circuit implementations of similarity-based novelty.

### Count-based as a special case of similarity-based novelty

The table below summarizes all variants of count-based and similarity-based familiarity discussed above. By comparing the equations in the table, we see that count-based and similarity-based novelty are equivalent if either

- (i) all observations  $s_{1:t}$  originate from a discrete set of distinct stimuli,  $s_{t'} \in S = \{x_1, \dots, x_N\}$ ; or
- (ii) similarity-based novelty uses box-shaped or dirac delta components centered at the stimuli  $x_1, \dots, x_N$ , such that each stimulus counted by count-based novelty is covered by exactly one separate component.

Then the responsibilities of similarity based familiarity are given as

$$\gamma_j^{(t)} = \frac{w_j^{(t)} \delta(s_t, x_j)}{\sum_{j'=1}^N w_{j'}^{(t)} \delta(s_t, x_{j'})} = \delta(s_t, x_j) \quad (\text{Equation 25})$$

such that both the initial familiarities and the familiarity updates of count-based and similarity-based familiarities are equal, yielding, for each of the three model variants (non-leaky, leaky, and fixed-rate), identical predictions of both models at all time points and for all stimuli.

### Familiarity update rules

Leakiness	Count/"soft" count notation	Delta-rule notation
<b>Count-based novelty</b>		
<b>Non-leaky</b>	$\rho^{(t+1)}(s) = \frac{C^{(t)}(s) + \epsilon}{t +  S \epsilon}$	$\rho^{(t+1)}(s) = (1 - \alpha_t)\rho^{(t)}(s) + \alpha_t \delta(s_t, s)$
Parameters: $ S , \epsilon$	$C^{(t)}(s) = C^{(t-1)}(s) + \delta(s_t, s)$	$\alpha_t = \frac{1}{t +  S \epsilon}$
<b>Leaky</b>	$\rho^{(t+1)}(s) = \frac{C^{(t)}(s) + \epsilon}{T_t +  S \epsilon}$	see Equation 55 in derivations
Parameters: $ S , \epsilon, \alpha$	$C^{(t)}(s) = (1 - \alpha)C^{(t-1)}(s) + \delta(s_t, s)$	
<b>Fixed rate</b>	see Equation 58 in derivations	$\rho^{(t+1)}(s) = (1 - \eta)\rho^{(t)}(s) + \eta \delta(s_t, s)$
Parameters: $ S , \alpha$		$\eta \in (0, 1)$
<b>Similarity-based novelty</b>		
<b>Non-leaky</b>	$\rho^{(t+1)}(s) = \sum_j^N w_j^{(t)} k_j(s)$	$\rho^{(t+1)}(s) = (1 - \alpha_t)\rho^{(t)}(s) + \alpha_t \sum_j^N \gamma_j^{(t)} k_j(s_t)$
Parameters: $N, \epsilon$	$w_j^{(t)} = \frac{R_j^{(t)} + \epsilon}{t + N\epsilon}$ $R_j^{(t)} = R_j^{(t-1)} + \gamma_j^{(t)}$	$\alpha_t = \frac{1}{t + N\epsilon}$
<b>Leaky</b>	$\rho^{(t+1)}(s) = \sum_j^N w_j^{(t)} k_j(s)$	see Equation 93 in derivations
Parameters: $N, \epsilon, \alpha$	$w_j^{(t)} = \frac{R_j^{(t)} + \epsilon}{T_t + N\epsilon}$ $R_j^{(t)} = (1 - \alpha)R_j^{(t-1)} + \gamma_j^{(t)}$	
<b>Fixed-rate</b>	see Equation 94 in derivations	$\rho^{(t+1)}(s) = (1 - \eta)\rho^{(t)}(s) + \eta \sum_j^N \gamma_j^{(t)} k_j(s_t)$
Parameters: $N, \alpha$		$\eta \in (0, 1)$

Update rules for count-based (upper three rows) and similarity-based familiarity (lower three rows), for three model variants each: (1) non-leaky model, (2) leaky model, (3) fixed-rate model. Counters and soft counters are initialized at zero:  $C^{(0)}(s) = 0$  for all  $s \in S$ ,  $R_j^{(0)} = 0$  for all components  $j = 1, \dots, N$ .

## Links of similarity-based novelty to circuit implementations of novelty

### Links to existing models

As summarized in [Table S1](#), mechanistic novelty models provide tools to implement familiarity- and novelty-computation based on gain adaptation,<sup>33,51</sup> short-term synaptic depression,<sup>50</sup> co-tuned inhibitory plasticity,<sup>55</sup> anti-Hebbian plasticity,<sup>47,54,56</sup> or – more generally – *multiplicative modulation* of existing weights.<sup>52</sup> Some of these models also show co-familiarization of ‘similar’ stimuli, mostly due to shared synaptic weights that lead to generalization across stimuli. In most models, this generalization arises from randomly initialized weight matrices,<sup>51,52</sup> but it can also emerge (though to a potentially lesser extent) in networks that have been pretrained.<sup>55,56</sup> Importantly, the normalization of familiarization effects that is a crucial part of similarity-based novelty is not, or at most implicitly, included in existing mechanistic modeling approaches, e.g. through anti-Hebbian learning rules<sup>47,54,56</sup> that lead to normalization of firing rates.<sup>88</sup>

### Normalization of familiarity effects in a circuit model

To see how we can implement similarity-based novelty in a circuit, we reformulate the update of *fixed-rate* similarity-based novelty. The update of the familiarity  $p^{(t)}(s)$  ([Equation 24](#)) is given as.

$$p^{(t+1)}(s) = (1 - \eta)p^{(t)}(s) + \eta \sum_{j=1}^N \gamma_j^{(t)} k_j(s). \quad (\text{Equation 26})$$

Because  $p$  is a mixture model, the above update of  $p$  is fully determined by the corresponding update of the familiarity weights  $w_j^{(t)}$ :

$$w_j^{(t+1)} = (1 - \eta)w_j^{(t)} + \eta \gamma_j^{(t)} = (1 - \eta)w_j^{(t)} + \eta \frac{w_j^{(t)} k_j(s_t)}{\sum_{j'} w_{j'}^{(t)} k_{j'}(s_t)} \quad (\text{Equation 27})$$

where we used the definition of the responsibilities  $\gamma_j^{(t)}$  to rewrite the equation. Note that all elements of the update rule in [Equation 27](#) are available locally in time and space. They can be interpreted as elements of a circuit as follows. The component activation  $k_j(s_t)$  corresponds to the presynaptic activation of a neuron  $j$  in response to the current observation  $s_t$ , which projects to a postsynaptic target with weight  $w_j^{(t)}$ . The first term in [Equation 27](#),  $(1 - \eta)w_j^{(t)}$ , corresponds to the leaky decay of this pre-to-post weight. The normalization factor  $\frac{1}{\sum_{j'} w_{j'}^{(t)} k_{j'}(s_t)}$  can further be interpreted as the (slightly delayed) activation of the postsynaptic target in the presence of shunting inhibition.<sup>89</sup>

A circuit model ([Figure S2](#)) could implement this computation using concepts of divisive normalization,<sup>89</sup> and learning modulated by dendritic disinhibition.<sup>90–92</sup> For simplicity, we assume in the following that neurons transmit their inputs approximately linearly, or that nonlinear transformations of their input can be decoded by the receiving neurons or neural populations. The details of how these mechanisms should be implemented in a network model remains subject of future work. Presynaptic activations take the role of individual component activations  $k_j(s)$  in response to a stimulus  $s$  from the environment ((1) in [Figure S2A](#)). Upon synaptic transmission (with synaptic weight  $w_j^{(t)}$ ), they evoke postsynaptic potentials of strength  $w_j^{(t)} k_j(s)$  in postsynaptic neurons  $j$ . The activations from the population of postsynaptic neurons are integrated by a shared inhibitory population into a combined familiarity signal (corresponding to  $\sum_{j=1}^N w_j^{(t)} k_j(s)$ , (2) in [Figure S2A](#)). The shared inhibitory population innervates the dendritic compartments of the postsynaptic neurons  $j$ , where shunting inhibition leads to divisive normalization. The resulting dendritic activity, corresponding to the normalization factor  $\frac{1}{\sum_{j=1}^N w_j^{(t)} k_j(s)}$ , is propagated down to the soma, where a multiplicative learning rule<sup>92</sup> leads to desired update of the pre-to-post weights between  $k_j(s)$  and the postsynaptic neuron  $j$ , respectively ((4) in [Figure S2A](#)). Interestingly, this mechanistic implementation predicts a *distributional familiarity code* in the activity of the postsynaptic population.

### Toy experiment: Gabor sequences of different similarity

Novelty predictions of count-based and one similarity-based novelty models were computed for three sequences of four Gabor stimuli each. Gabor stimuli are characterized by their angular orientation. Across all sequences  $i = 1, 2, 3$ , all except the second stimulus  $s_2$  are identical. The second stimulus in each sequence shows varying levels of similarity to the first stimulus:  $s_2^{(1)} = 45^\circ$  (identical to  $s_1^{(1)}$ ),  $s_2^{(2)} = 60^\circ$  (15° difference to  $s_1^{(2)}$ ),  $s_2^{(3)} = 85^\circ$  (40° difference to  $s_1^{(3)}$ ).

Countable states for the count-based novelty models are obtained by binning the space of Gabor orientations  $[0, 180^\circ]$  into 9 bins of width  $20^\circ$  with centers  $0^\circ, 20^\circ, 40^\circ$ , etc. (model variant 1) or into four bins of  $45^\circ$  with centers  $0^\circ, 45^\circ, 90^\circ$ , and  $135^\circ$  (model variant 2). The size of the state space,  $|S|$ , is equal to the number of bins, respectively; counts at  $t=0$  were initialized at zero, and leakiness was set to  $\alpha = 0$  (non-leaky).

The component representation of similarity-based novelty is defined on the 1D torus of Gabor orientations  $[0, 180^\circ]$ . We use four equidistantly placed components  $k_j$ ,  $j = 1, \dots, 4$ , centered at  $c_1 = 0^\circ, c_2 = 45^\circ, c_3 = 90^\circ$  and  $c_4 = 135^\circ$ . The components are defined by the following triangular function:

$$k_j(\mathbf{s}) = \frac{1}{\sigma_j} \left( 1 - \frac{|\mathbf{s} - \mathbf{c}_j|}{\sigma_j} \right) \theta \left( 1 - \frac{|\mathbf{s} - \mathbf{c}_j|}{\sigma_j} \right), \quad (\text{Equation 28})$$

where  $\mathbf{s} \in [0^\circ, 180^\circ]$  is the orientation of the Gabor stimulus,  $\mathbf{c}_j$  as defined above are the centers of the components,  $\sigma_j$  are the component widths, and  $\theta$  is the Heaviside function. We choose  $\sigma_j = 45^\circ$  for all components, creating the overlapping components depicted in Figure 1C (left). The component weights at  $t = 0$  were initialized uniformly to  $\frac{1}{N}$ , where  $N = 4$  is the number of components. For the toy example simulations depicted in Figure 1C (right), we used non-leaky similarity-based novelty ( $\alpha = 0$ ), with prior  $\varepsilon = 1$ .

### Component width variation

For the novelty signals depicted in Figure S1, we simulated the same similarity-based novelty model as above, but with (i) component width  $\sigma_j = 52.5^\circ$  ( $N = 4$ , same component centers), (ii) component width  $\sigma_j = 60^\circ$  ( $N = 4$ , same component centers), (iii) component width  $\sigma_j = 60^\circ$  and  $N = 3$  components centered equidistantly at  $c_1 = 0^\circ$ ,  $c_2 = 60^\circ$  and  $c_3 = 120^\circ$ .

## Passive viewing task

### Experimental data

Homann et al.<sup>51</sup> used two-photon calcium imaging to record the neural activity of layer 2/3 neurons in the primary visual cortex (V1) of 5 GCaMP6f-expressing mice during passive viewing of familiar and "novel" visual stimuli. During the recording, mice were head-fixed but could run freely on an air suspended Styrofoam ball in front of a toroidal screen, onto which the visual stimuli were projected. Homann et al. extracted the peak neural responses (trial- and population-averaged) to the novel image ( $\Delta N$ ) and to the familiar sequence (steady-state activity  $N_\infty$ ) during the "variable repetition experiment" ( $L$ -experiment) and the "variable image number experiment" ( $M$ -experiment), as well as the transient population response ( $\Delta N$ ) to the formerly familiar sequence after the recovery interval in the "repeated image set experiment" ( $L'$ -experiment).

### Model simulations and fitting

**Stimuli.** All images for our simulation of the passive viewing experiment were sampled from a single distribution, designed to match that of Homann et al.<sup>51</sup> Each image is the linear superposition of 40 Gabor filters with randomly chosen orientation (uniformly sampled from  $[0, \pi]$ ), phase (unif. sampled from  $[-\pi, \pi]$ ), frequency (unif. sampled from  $[0.04, 0.08]$ ), and location ( $x$  and  $y$  coordinate sampled uniformly from  $[-130, 130]$  and  $[-20, 90]$ , in units of visual degrees) on a grey background. The width of each Gabor filter was set to  $\frac{1}{3f}$ , where  $f$  is the spatial frequency of the Gabor filter.

**Simulation protocol.** Model simulations followed the same protocol as the mouse experiment by Homann et al.<sup>51</sup> Each simulation step corresponds to 300 ms in the experiment, i.e. in each simulation step, a single stimulus is presented.

**Component definition.** The input stimuli for similarity-based novelty are  $100 \times 200$  pixel images as defined above. Each image is represented by a pixel vector  $\mathbf{s} \in \mathbb{R}^{20,000}$  with entries specifying the intensity of each image pixel (grey-scale; the lower the pixel value, the darker the color; grey background has pixel value 0). We define two types of component functions that emulate the receptive fields of V1 simple and complex cells, respectively.

- (i) "Simple cell"-like components. Analogously to the component centers in the toy example, each component function  $k_j$  is defined relative to a 'reference Gabor filter'  $\mathbf{r}_j \in \mathbb{R}^{20,000}$ , i.e. the pixel vector of a single Gabor filter that is chosen equidistantly from the same parameter space (orientation, phase, frequency,  $x$ - $y$ -location) as the Gabor filters that form the stimulus images. The value of component  $k_j$  with reference Gabor  $\mathbf{r}_j$  in response to a stimulus image  $\mathbf{s}$  is defined as

$$k_j(\mathbf{s}) = \varphi(f(\mathbf{s}, \mathbf{r}_j)), \quad (\text{Equation 29})$$

where  $f$  is the cosine distance between the pixel vectors  $\mathbf{s}$  and  $\mathbf{r}_j$ :

$$z_j = f(\mathbf{s}, \mathbf{r}_j) = \frac{\mathbf{s} \cdot \mathbf{r}_j}{\|\mathbf{s}\| \|\mathbf{r}_j\|} \in [-1, 1], \quad (\text{Equation 30})$$

and  $\varphi$  is a non-negative transform function that maps the similarity value  $z_j$  between the stimulus and the reference Gabor (in the interval  $[-1, 1]$ ) to the component activation. We choose the function  $\varphi$  to be triangular with width parameter  $\sigma_j = 1$  and centered at  $z = 1$  (maximum similarity):

$$\varphi^\Delta(z_j) = \begin{cases} 1 - \frac{1|z - 1|}{\sigma_j} & \text{if } z_j \in [1 - \sigma_j, 1], \\ 0 & \text{otherwise.} \end{cases} \quad (\text{Equation 31})$$

- (ii) "Complex cell"-like components. Complex cell-like components are defined relative to a reference set of two simple cell-like components of opposite phase. The activations from the set of reference components are added, and then passed through the same function  $\varphi$  as for the simple cell-like components. Complex cell-like components can be thought of as encoding an OR relation between the reference Gabors of the simple cell-like components in their reference set.

We define  $K_s$  simple cell-like components and  $K_c$  complex cell-like components. The number of simple cell-like components  $K_s$  is determined by two free parameters: the convolutional density (or stride length)  $c$  that determines the distance between two component centers in units of image pixels, and the number  $N$  of Gabor filters with different combinations of orientation, phase and frequency. The number of complex cell-like components  $K_c$  is chosen to maintain a ratio of  $K_s : K_c = 3 : 1$ , in alignment with experimental results for layer 2/3 of V1 in mice,<sup>93</sup> where Homann et al. recorded. The two parameters  $c$  and  $N$  determining the number of components as well as the novelty parameters  $\alpha$  and  $\epsilon$  are fitted jointly to the neural data by Homann et al. as described below.

**Comparison with count-based novelty.** The count-based model receives the same inputs but does not use receptive fields. Instead, separate counts are used for different images. The parameter  $|S|$ , specifying the total number of stimulus counters ( $|S| > 13$ , i.e. larger than the number of stimuli presented in the protocol), and the novelty parameters  $\alpha$  and  $\epsilon$  are fitted to the neural data.

**Fit to neural data.** To fit each novelty model to the neural data, we use grid search in combination with linear regression. We simulate a given novelty model  $\mathcal{M} \in \{\text{count-based novelty, similarity-based novelty}\}$  across the full range of model parameters

$$\theta_{\text{count}} = (\alpha, \epsilon, |S|), \theta_{\text{similarity}} = (\alpha, \epsilon, c, N),$$

with  $n = 50$  model simulations for each parameter set. We then fit the statistics of the simulated novelty responses,  $X$ , to the statistics of the neural data,  $Y$ , using linear regression. Statistics from all three Homann experiments are fitted jointly, with equal weighting for each experimental condition. The parameter set that minimizes the mean-squared error (MSE) between neural data  $Y$  and the fitted model predictions  $\beta^* X$  is chosen as the best-fit parameter set for model  $\mathcal{M}$ . The error of the fitted MSE as reported in Figure 2 D is computed as the standard error based on jackknife re-sampling<sup>94</sup> of the experimental data from all experiments.

**Cross-validation.** To control for overfitting, we compute the cross-validated MSE for each fitted model. We used leave-one-out cross-validation, where a single experimental condition is dropped for each training set (yielding 20 cross-validation folds for each of the experimental conditions). For Figures 2E–FFigures S3S4, we compute the average test error across cross-validation folds as well as the trained model predictions on their respective held-out experimental condition. We report the standard errors of cross-validated MSE and test predictions across jackknife-resampled repetitions of the cross-validation procedure.

**Parameter robustness (Figure S3).** We test parameter sensitivity by systematically varying a single parameter across its allowed range, while keeping all other model parameter fixed at their fitted values. For each perturbation, we refit the linear regression to the experimental data, and report the resulting (i) model fit to the experimental data and (ii) model predictions. We perform this sensitivity analysis for all fitted model parameters (Figures S3A–S3E), i.e.

- Count-based novelty:
  - leakiness  $\alpha$ ,
  - prior  $\epsilon$ ,
  - number of states  $|S|$ ;
- Similarity-based novelty:
  - leakiness  $\alpha$ ,
  - prior  $\epsilon$ ,
  - components' convolutional density  $c$ ,
  - number of convolution components  $N$  (i.e. different Gabor filters that are repeated in each center location).

For similarity-based novelty, we also test robustness to changes in component width  $\sigma$  in an analogous way (Figure S3F), but allowing either refitting of (i) both the novelty parameters  $\alpha$  and  $\epsilon$  as well as the linear regression, or (ii) only the linear regression, as for the other parameters.

## Active exploration task

### Behavioral data

**Task.** Rosenberg et al.<sup>59</sup> track the behaviour of 20 mice (10 of them water-deprived) during free exploration of an unfamiliar maze (maze cleaned with ethanol between mice). Each mouse was video-recorded during its first 7 hours of access to the maze. The maze was constructed as a 6-level binary tree, with branching points and end points of corridors as nodes. One single end node in the maze contained a water delivery port, which delivered a water reward to water-deprived mice (rewarded group) but was inactive during the experiment with non-deprived mice (unrewarded group). The  $x$ - $y$ -coordinates of the animal's nose in each video frame were used as the mouse position for subsequent behavioral analysis (both by Rosenberg et al.,<sup>59</sup> and in our study).

**Environment formalization.** To enable direct comparison between the behavior of mice and reinforcement learning (RL) models, we formalize the maze environment using "states" and "actions." States  $s$  characterize the possible locations of agents (mice or RL models) in the maze, and actions  $a$  characterize the possible next direction the agent can take from a given state. Analogously to Rosenberg et al.,<sup>59</sup> the home cage and all branching points and end points of the maze corridors (i.e. all nodes in the binary tree) were defined as states. We define four different actions: (i) go forward into maze (available in the home cage); (ii) go back (available in all states except the home cage); (iii) turn and go into the left corridor (available in all branching points) and (iv) turn and go into the right corridor (available in all branching points). For simplicity of notation, our notion of left and right is based on the egocentric view of

the agent during an outward path. Based on the definition of states in the maze, mouse behavior over time is discretized into ‘time points’, where each time point marks the transition to a new state in the maze.

**Preprocessing.** For each mouse in Rosenberg et al.’s data set,<sup>95</sup> we extract the sequence of states and actions taken at each time point between its first entry into the maze and its first encounter with the goal state. To exclude any reward-related behaviors from the analysis, we disregard mouse behavior after the first encounter with the goal state, independently of whether the mouse received a reward in the goal state (rewarded group) or not (unrewarded group). Since there are no significant behavioral differences between rewarded and unrewarded mice during the time before the first reward encounter,<sup>59</sup> we treat them as equivalent.

### **Novelty-Seeking reinforcement learning (N-RL) models**

We adapt classical reinforcement learning (RL) algorithms to model novelty-seeking by replacing the extrinsic reward signal by an intrinsically computed novelty signal (“novelty-seeking RL agents,” see e.g. Xu et al.,<sup>20</sup> Gershman and Niv,<sup>29</sup> Jaegle et al.,<sup>32</sup> Belle-mare et al.,<sup>36</sup> and Chentanez et al.<sup>96</sup> for similar approaches).

Like classical RL agents, novelty-seeking RL (N-RL) agents move in an environment which is defined by states  $s$ , and actions  $a$  that allow the agent to transition between states. In general, the environment can be probabilistic in the sense that a given action  $a$  in a given state  $s$  can lead the agent into different states  $s'$ . The probabilities for each such state transition  $(s, a, s')$  are called transition probabilities and summarized in the transition matrix  $p \in \mathbb{R}^{|\mathcal{S}| \times |\mathcal{A}| \times |\mathcal{S}|}$ , where  $|\mathcal{S}|$  is the size of the state space  $\mathcal{S}$ , and  $|\mathcal{A}|$  is the size of the action space  $\mathcal{A}$ . In our model, the environment is deterministic, i.e., the correct transition matrix contains only binary entries.

The main difference between novelty-seeking and classical RL algorithms is that, instead of maximizing the discounted future rewards, N-RL algorithms maximize the discounted future novelty:

$$G_{N,t} = \sum_{t'=t}^{T_{\max}} \gamma^{t'-t} N^{(t)}(s_{t'}), \quad (\text{Equation 32})$$

where  $\gamma$  is the discount factor, and the novelty  $N^{(t)}(s_{t'})$  is computed using any suitable novelty model, e.g. count-based or similarity-based novelty. Importantly, the novelty  $N^{(t)}$  is updated only based on “real” state observations sampled from the environment, and remains fixed during planning. This still means that the N-RL agent’s reward function  $N^{(t)}(s)$  changes in every time step during exploration, in contrast to classical RL-agents that usually maximize a reward function that is constant in time. Analogous to reward-based V-values and Q-values, N-RL algorithms compute ‘novelty-based’ V-values and Q-values:

$$V_N^{(t)}(s) = \mathbb{E}_{a_{t'} \sim \pi^{(t')}(\cdot|s_{t'}), t'=1, \dots, t} \left[ \sum_{t'=1}^{\infty} \gamma^{t'-t} N^{(t)}(s_{t'}) \mid s_1 = s \right], \quad (\text{Equation 33})$$

$$Q_N^{(t)}(s, a) = \mathbb{E}_{a_{t'} \sim \pi^{(t')}(\cdot|s_{t'}), t'=1, \dots, t} \left[ \sum_{t'=1}^{\infty} \gamma^{t'-t} N^{(t)}(s_{t'}) \mid s_1 = s, a_1 = a \right]. \quad (\text{Equation 34})$$

Similar to the novelty that is estimated only based on ‘real’ state observations, the estimated transition probabilities remain fixed during planning. This aligns with established practice in the field of intrinsically motivated RL.<sup>97</sup> As a result, the agent iterates between *sampling from the environment*, using the current action policy (based on the current value estimates) and leading to updates in the novelty function  $N^{(t)}(s)$  and the estimated transition probabilities – and *planning*, where the Bellman equation is solved based on the current estimates of novelty and transition probabilities and the values and action policy are updated accordingly.

### **Model-based N-RL algorithm**

We use a model-based (MB) N-RL algorithm (Algorithm 1) that updates its novelty-based Q-values through prioritized sweeping (e.g., see Sutton and Barto<sup>87</sup>) and applies a softmax policy on the novelty-based Q-values to determine its actions. The MB N-RL algorithm is characterized by the parameters of the novelty model ( $\epsilon_c$  for count-based novelty;  $\epsilon_k$  and  $\mathbf{k}$  for similarity-based novelty), as well as five additional parameters (see line 2 in Algorithm 1). The inverse temperature  $\beta$  determines the noise level of the agent’s softmax policy over the Q-values (low  $\beta$  leads to more noisy actions). The leak factor  $k_{\text{leak}}$  and the prior  $\epsilon_{\text{env}}$  define the update of the belief counters  $\alpha$ . The belief counters  $\alpha$  help the agent form its world model by keeping track of how often the agent has encountered a given transition  $(s, a, s')$  in the environment. Specifically,  $k_{\text{leak}}$  determines how much the counter  $\alpha$  for a given transition decays between two observations of that transition – the higher  $k_{\text{leak}}$ , the more influenced are the  $\alpha$  values by recently observed transitions, and the lower the influence of the prior  $\epsilon_{\text{env}}$ . A lower  $\epsilon_{\text{env}}$ , in turn, causes the  $\alpha$  values to adjust more quickly to the observed transition counts. The more deterministic our environment is, the lower our  $\epsilon_{\text{env}}$  should thus be, since every observation contains reliable information about the underlying state transition. The parameter  $\lambda$  and  $T_{PS}$  characterize the Q-value updates:  $T_{PS}$  determines how many sweeps, i.e. Q-value updates are performed in each step (the higher  $T_{PS}$ , the more precise the model-based Q-values update), while  $\lambda$  has a similar role to the discount factor  $\gamma$  that discounts the influence of future expected novelty on the state value of a given state  $s$  (the higher  $\lambda$ , the more important the value of future novelty).

### Algorithm 1. Model-based N-RL algorithm

#### # Parameter specification

- 1: Specify environment parameters  $S, A, p$ , and  $s_G$  (goal state),  $T_{\max}$  (max. simulation time steps).
- 2: Specify algorithm parameters  $\{\lambda, T_{PS}, k_{\text{leak}}, \epsilon_{\text{env}}, \beta\}$ .
- 3: Specify novelty parameters  $\{\alpha, \epsilon_c\}$  (count-based novelty) or  $\{\alpha, \epsilon_k, \mathbf{k}\}$  (similarity-based novelty).

#### # Variable initialization

- 4: Initialize novelty variables (counts for count-based novelty; soft counts for similarity-based novelty).
- 5: Initialize  $U^{(1)}(s) = \frac{N^{(1)}(s)}{1-\lambda}$  for all  $s \in S$ .
- 6: Initialize Q-values as  $Q_N^{(1)}(s, a) = U^{(1)}(s)$  for all  $(s, a) \in S \times A$ .
- 7: Initialize  $\alpha_{s,a}^{(1)}(s') = \begin{cases} \epsilon_{\text{env}} & \text{if } p(s, a, s') > 0 \\ \text{nan} & \text{if } p(s, a, s') = 0 \end{cases}$  for all  $(s, a, s') \in S \times A \times S$ .

#### # Initialize agent.

- 8: Set  $t \leftarrow 0$  and initialize state  $s_1 \leftarrow s_{\text{home}}$  (home cage).
- 9: Update novelty variables using leaky count-based novelty or leaky similarity-based novelty.
- 10: Initialize time  $t \leftarrow 1$ .

#### # Simulate agent.

- 11: **while**  $s_t \neq s_G$  and  $t < T_{\max}$  **do**

#### # Take action

- 12: Sample action  $a_t$  from  $\pi(a|s_t) \propto \exp(\beta Q_N^{(t)}(s_t, a))$ .
- 13: Evaluate action  $a_t$  in environment and observe next state  $s_{t+1}$ .

#### # Update internal variables

- 14: Update novelty variables using leaky count-based novelty or leaky similarity-based novelty.
- 15: Update  $\alpha_{s,a}^{(t+1)}(s') = k_{\text{leak}} \alpha_{s,a}^{(t)}(s') + (1 - k_{\text{leak}}) \epsilon_{\text{env}}$  for all  $(s, a, s') \in S \times A \times S$ .
- 16: Update  $\alpha_{s_t, a_t}^{(t+1)}(s') = \alpha_{s_t, a_t}^{(t)}(s') + \delta(s', s_{t+1})$  for all  $s' \in S$ .
- 17: Update beliefs  $\hat{\theta}^{(t+1)}(s') = \alpha_{s,a}^{(t+1)}(s') / \sum_{\tilde{s}' \in S} \alpha_{s,a}^{(t+1)}(\tilde{s}')$ .
- 18: Compute  $N^{(t+1)}(s)$  as leaky count-based novelty or leaky similarity-based novelty.
- 19: Compute new Q-values  $Q_N^{(t+1)}(s, a)$  and new U-values  $U^{(t+1)}(s)$  for all  $(s, a) \in S \times A$ , using  $N^{(t+1)}(s)$  and Algorithm 2

#### # Update time step

- 20:  $t \leftarrow t + 1$ .
- 21: **end while**

### Algorithm 2. Prioritized Sweeping with target signal $N^{(t+1)}$

- 1: Initialize next Q-values  $Q^{(t+1)}(s, a) = \sum_{s' \in S} \hat{\theta}_{s,a}^{(t+1)}(s') (N^{(t+1)}(s') + \lambda U^{(t)}(s'))$  for all  $(s, a) \in S \times A$ .
- 2: Initialize next U-values  $U^{(t+1)}(s) = U^{(t)}(s)$  for all  $s \in S$ .
- 3: Initialize priority queue  $q = \left\{ \left| \max_{a \in A} Q^{(t+1)}(s, a) - U^{(t+1)}(s) \right| \right\}_{s \in S}$ .

#### # Perform prioritized Q-value update

- 4: **for**  $\tau = 1, \dots, T_{PS}$  **do**
- 5: Determine state with largest U-value update as  $s_\tau = \text{argmax}(q)$ .
- 6: Compute U-value update for  $s_\tau$  as  $\Delta U_\tau^{(t+1)} = \max_{a \in A} Q^{(t+1)}(s_\tau, a) - U^{(t+1)}(s_\tau)$ .
- 7: Update  $U^{(t+1)}(s_\tau) \leftarrow U^{(t+1)}(s_\tau) + \Delta U_\tau^{(t+1)} = \max_{a \in A} Q(s_\tau, a)$ .
- 8: Update next Q-values  $Q^{(t+1)}(s, a) \leftarrow Q^{(t+1)}(s, a) + \lambda \hat{\theta}^{(t+1)}(s, a, s_\tau) \Delta U_\tau^{(t+1)}$  for all  $(s, a) \in S \times A$ .
- 9: Recompute priority queue  $q \leftarrow \left\{ \left| \max_{a \in A} Q^{(t+1)}(s, a) - U^{(t+1)}(s) \right| \right\}_{s \in S}$
- 10: **end for**

### Model simulations and fitting

We simulate and fit a total of 11 MB N-RL models to the exploration behavior of the mice in the Rosenberg maze, and compare the fitted models with respect to their log-evidence: (i) five agents seeking similarity-based novelty, (ii) one agent seeking count-based novelty, and (iii) five agents seeking a combination of count-based and similarity-based novelty (see below). Our model-based algorithm implements Prioritized Sweeping<sup>87</sup> (Algorithm 2).

**Count-based novelty.** Count-based novelty in the Rosenberg maze uses a separate count for each state as defined above, i.e. each node in the binary tree representation of the maze. State counts are updated in a leaky fashion, with leakiness  $\alpha$  and prior  $\epsilon$  as free parameters that are fitted to data. Using Equations 1, 2, and 3, the novelty reward at each time step  $t$  is computed as the novelty of the current state  $s_t$  in the environment.

**Similarity-based novelty.** We define similarity-based novelty in the Rosenberg maze with respect to six sets of components that differ in the degree of granularity with which they encode the maze. The component representations are based on the idea that mice could generalize across states within a given "area" of the maze, e.g. the left or right half of the maze (allo-centric perspective). The size of the area covered by each components determines the "granularity" of the representation, i.e. the larger the area, the lower the granularity of the representation. To formally write the corresponding component functions, we consider the binary-tree structure of the maze (Figure 3C2 [also see Figure S5]). We define level  $\ell = 0, \dots, 6$  in the binary tree, where level  $\ell = 0$  contains the first branching point in the maze and level  $\ell = 6$  contains all end nodes. We define the "(sub-)tree"  $\text{Tree}(s)$  for any state  $s$  in the maze ( $s \in S$ ) as

$$\text{Tree}(s) = \{s' \in S : s' \text{ is part of a binary tree with root state } s\}. \quad (\text{Equation 35})$$

Hence, the entire maze (excluding the home state) can be written as  $\text{Tree}(s_0)$  where  $s_0$  is the first branching point of the maze. We further define the sets  $S_\ell$ ,  $\ell = \{0, 1, \dots, 6\}$ , where each set  $S_\ell$  contains all  $2^\ell$  states in a given level  $\ell$  of the binary tree. For example, the set  $S_0$  contains only the first branching point of the maze, while  $S_6$  contains all 64 end nodes.

We define the set of components for a given granularity level  $\ell$ ,  $\ell = 1, \dots, 6$  as follows: For each of the  $2^\ell$  states  $s_j \in S_\ell$ , we define an area component:

$$k_j^{(\ell)} = \begin{cases} \frac{1}{K^{(\ell)}} & \text{if } s \in \text{Tree}(s_j) \\ 0 & \text{otherwise,} \end{cases} \quad (\text{Equation 36})$$

where the factor  $K^{(\ell)} = K_j^{(\ell)} = \sum_{s \in S} k_j^{(\ell)}(s)$  is added to normalize the components to 1 over the state space. For all other  $1 + \sum_{\ell'=0}^{\ell-1} 0$  states  $s_j \notin \text{Tree}(S_\ell)$ , we define single state components:

$$k_j(s) = \begin{cases} 1 & \text{if } s = s_j \\ 0 & \text{otherwise.} \end{cases} \quad (\text{Equation 37})$$

This means that for components of granularity level  $\ell = 6$ , each state in the maze is covered by a separate component – similarity-based novelty with this choice of components is equivalent to count-based novelty.

**Combined count- and similarity-based novelty.** Combined count- and similarity-based novelty agents seek a novelty signal

$$N^{(t)}(s) = w_{cnov} N_{cnov}^{(t)}(s) + (1 - w_{cnov}) N_{snov}^{(t)}(s) \quad (\text{Equation 38})$$

that is the weighted sum of count-based novelty  $N_{cnov}^{(t)}$  and similarity-based novelty  $N_{snov}^{(t)}$ . Both count-based and similarity-based novelty are defined separately as described above. They share the free parameter  $\alpha$  (leakiness), but have separate free parameters  $\epsilon_{cnov}$  and  $\epsilon_{snov}$  instead of a shared parameter  $\epsilon$  (prior).

**Model fitting.** We fit the parameters of each MB N-RL model to mouse behavior using maximum-likelihood estimation (MLE):

$$\hat{\theta} = \underset{\theta}{\text{argmax}} \text{LL}(\mathcal{D}, \mathcal{M}, \theta) = \underset{\theta}{\text{argmax}} \log P(\mathcal{D} | \mathcal{M}, \theta), \quad (\text{Equation 39})$$

where  $\mathcal{M}$  is the MB N-RL model that is being fitted,  $\theta$  are its model parameters, and  $\mathcal{D}$  is the mouse data. The mouse data  $\mathcal{D}$  consists of the appended state-action sequences of all 20 mice during their exploration of the maze:

$$\mathcal{D} = \left( s_0^{(1)}, a_0^{(1)}, \dots, a_{T_1-1}^{(1)}, s_{T_1}^{(1)}, \dots, s_0^{(20)}, a_0^{(20)}, \dots, a_{T_{20}-1}^{(20)}, s_{T_{20}}^{(20)} \right), \quad (\text{Equation 40})$$

where  $s_t^{(i)}$  and  $a_t^{(i)}$  denote the state and action of mouse  $i$  at time point  $t$ , and  $T_i$  denotes the number of exploration steps of mouse  $i$  until its first encounter of the goal state.

We maximize the log-likelihood for a given model by minimizing its negative log-likelihood using the `scipy.optimize` implementation of the Nelder-Mead algorithm (best performing among relevant optimization algorithms, including L-BFGS-B, SLSQP). In each minimization step, the log-likelihood for the current parameter set  $\theta$  is computed (see [derivations](#)), and its negative is used as input to the minimizer of the `scipy.optimize` package to compute the next candidate parameter set. To improve optimization performance by the use of unbounded optimization variables, we optimize model parameters in a transformed space (sigmoid function for MB parameters  $\lambda$ ,  $k_{leak}$  and novelty parameters  $\alpha$ ,  $w_{cnov}$ ; soft-plus function for MB parameters  $\beta$ ,  $\epsilon_{env}$  and novelty parameter  $\epsilon$ ). For each model, we run the optimization until convergence from 5 different initial conditions, and use the best-fitting parameter set across initial conditions as the fitted parameters for the model.

**Bayesian model comparison.** To compare fitted models, we approximate their log-evidence (LE) in two ways. First, we compute the model LE based on the Schwartz approximation:

$$\text{LE}(\mathcal{D}, \mathcal{M}) \approx \frac{1}{2} \text{BIC}(\mathcal{D}, \mathcal{M}) = \text{LL}(\mathcal{D}, \mathcal{M}, \theta) - \frac{|\theta|}{2} \log(T), \quad (\text{Equation 41})$$

where the log-likelihood (LL) term accounts for the fit of the model. The remaining term introduces a penalty for model complexity, which is a function of the number of free parameters  $|\theta|$  of the model, and the number of data points  $T = \sum_{i=1}^{20} T_i$  in the fitted data set  $\mathcal{D}$ .

Second, we approximate the model LE as the cumulative negative log-likelihood of the model on the test sets of  $k$ -fold cross-validation:

$$\text{LE}(\mathcal{D}, \mathcal{M}) \approx \sum_{i=1}^k \text{LL}(\mathcal{D}_i^{(\text{test})}, \mathcal{M}, \theta_i), \quad (\text{Equation 42})$$

where  $\mathcal{D}_i^{(\text{test})}$  is the held-out set of the  $i$ th fold in  $k$ -fold cross-validation, and  $\theta_i$  are the parameters of the model fitted on the train data  $\mathcal{D}_i^{(\text{train})}$  of the corresponding fold. Here, we perform 5-fold cross-validation by sub-sampling the set of mice (Figures 3 and S5).

**Posterior predictive checks.** As a posterior predictive check, we compute the average steps until agents reach the rewarded goal state in the maze (Figure S5B), comparing (i) mice, (ii) fitted leaky count-based agents, and (iii) fitted combined agents (level-5-6 count-similarity-based). For each model, statistics are computed based on  $20 \times 20$  simulations with different initial seeds.

## Sum-of-parts experiment

### Experimental protocol

**Stimuli.** The ‘novel’ stimulus  $X$  consists of 12 identical Gabors ( $0^\circ$  phase:  $-0.5\pi$ , frequency:  $0.06/^\circ$ , width:  $5.56^\circ$ ) that are positioned equidistantly in two rows of six Gabors each (center distance in  $x$ -direction:  $43.3^\circ$ , center distance in  $y$ -direction:  $45^\circ$ ) on rectangular grey background (dimensions:  $[-130, 130^\circ]$  in  $x$ -direction (pixel resolution: 200),  $[-20^\circ, 70^\circ]$  in  $y$ -direction (pixel resolution: 100)). Each of the 12 familiar stimuli  $I_m$ ,  $m = 1, \dots, 12$  consists of one of the 12 Gabors used in the novel image  $X$ , positioned in the same location as in  $X$ . Relative to the familiar stimuli, the Gabors in the novel image  $X$  are rotated by an angle  $\vartheta$  between  $0^\circ$  and  $180^\circ$ .

**Protocol.** The first  $M$  familiar stimuli ( $M = 3, 6, 9, 12$ ) are presented for  $L = 17$  repetitions. In the subsequent repetition  $L + 1$ , the last image of the familiar sequence is replaced by the novel image  $X$ . The (normalized) novelty response  $\Delta N(X)$  to the novel image  $X$  is computed as the difference of the raw novelty response to  $X$  and the average raw novelty response to the familiar stimuli during the last repetition of the sequence:

$$\Delta N(X) = N^{(M(L+1))}(X) - N_\infty = N^{(M(L+1))}(X) - \frac{1}{M} \sum_{m=1}^M N^{(M(L+1)-m)}(I_m), \quad (\text{Equation 43})$$

where  $N^{(t)}(s)$  is the raw novelty response to a stimulus  $s$  at time step  $t$ , as predicted by similarity-based novelty model.

### Model simulations

We simulate similarity-based novelty during the sum-of-parts experiment. Components are identical to the ‘simple cell’-like components used to model the passive viewing experiment (see above). Model parameters are equivalent to the fitted parameters from the passive viewing experiment (in particular: leakiness parameter  $\alpha = 0.1$ ). We test the robustness of the experimental predictions to changes in  $\alpha$  by repeating the model simulations for grid values of  $\alpha \in [0.1, 0.9]$  (discretization step: 0.1) (Figure S6).

## Derivations

### Derivation of count-based novelty

Derivations of non-leaky count-based novelty are based on Xu et al.<sup>20</sup> Count-based novelty computes the familiarity  $\rho^{(t+1)}(s)$  for any stimulus  $s \in \mathcal{S} = \{X_1, \dots, X_N\}$  as the maximum-a-posteriori (MAP) estimate of previously observed stimuli  $s_{1:t}$ :

$$\Pr(s_{t+1} = X_i | s_{1:t}) = \rho^{(t+1)}(X_i) = \theta_i^{(t+1)} = \underset{\theta}{\text{argmax}} \log \Pr(s_{1:t} | \theta) \Pr(\theta). \quad (\text{Equation 44})$$

Under the iid assumption for the observations  $s_{1:t}$  and for a conjugate Dirichlet prior with concentration parameter  $\varepsilon + 1 = (\varepsilon_1 + 1, \dots, \varepsilon_N + 1)$ , Equation 44 reduces to:

$$\theta_i^{(t+1)} = \underset{\theta}{\text{argmax}} \sum_{i=1}^N \left( \sum_{t'=1}^t \delta(s_{t'}, X_i) + \varepsilon_i \right) \log \theta_i + \log B(\varepsilon + 1). \quad (\text{Equation 45})$$

where  $B(\varepsilon + 1)$  denotes the beta function that is independent of the maximization parameter  $\theta$ , and  $\delta$  is the Kronecker delta function. Since the  $\theta_i^{(t+1)}$  estimate the parameters of a categorical distribution, they need to sum to one. To maximize the log-likelihood in Equation 45 under this constraint, we solve

$$0 = \frac{d}{d\theta_j^{(t+1)}} \left[ \sum_{j=1}^N \left( \sum_{t'=1}^t \delta(s_{t'}, x_j) + \varepsilon_j \right) \log \theta_j^{(t+1)} - \lambda \left( \sum_{j=1}^N \theta_j^{(t+1)} - 1 \right) \right] = \frac{C^{(t)}(x_j) + \varepsilon_j}{\theta_j^{(t+1)}} - \lambda, \quad (\text{Equation 46})$$

where  $\lambda$  is the Lagrange multiplier, and  $C^{(t)}(x_i) = \sum_{t'=1}^t \delta(s_{t'}, x_i)$  denotes the counts of stimulus  $x_i$  at time  $t$  (after observing  $s_t$ ). Solving Equation 46, we obtain the relationship

$$\frac{C^{(t)}(x_i) + \varepsilon_i}{\theta_i^{(t+1)}} = \frac{1}{N-1} \sum_{j=1, j \neq i}^N \frac{C^{(t)}(x_j) + \varepsilon_j}{\theta_j^{(t+1)}}, \quad (\text{Equation 47})$$

which is satisfied whenever  $\theta_i^{(t+1)} = r_t (C^{(t)}(x_i) + \varepsilon_i)$  for some  $r_t \in \mathbb{R}$ . Due to the normalization condition on the  $\theta_i^{(t+1)}$ , we therefore obtain the MLE estimators

$$p^{(t+1)}(x_i) = \theta_i^{(t+1)} = \frac{C^{(t)}(x_i) + \varepsilon_i}{\sum_{j=1}^N C^{(t)}(x_j) + \varepsilon_j} = \frac{C^{(t)}(x_i) + \varepsilon_i}{t + \sum_{j=1}^N \varepsilon_j} \quad (\text{Equation 48})$$

For symmetric Dirichlet prior, i.e.  $\varepsilon_i = \varepsilon$  for all  $i = 1, \dots, |S| = N$ , this reduces to Equation 10.

*Delta-rule formulation.* The count-based familiarity in Equation 10 can be written as the delta rule in Equation 11 as follows:

$$p^{(t+1)}(x_i) = \frac{C^{(t-1)}(x_i) + \delta(s_t, x_i) + \varepsilon}{t + N\varepsilon} = \underbrace{\frac{C^{(t-1)}(x_i) + \varepsilon}{t - 1 + N\varepsilon}}_{p^{(t)}(x_i)} \underbrace{\left( 1 - \frac{1}{t + N\varepsilon} \right)}_{1 - \alpha_t} + \underbrace{\frac{1}{t + N\varepsilon}}_{\alpha_t} \delta(s_t, x_i). \quad (\text{Equation 49})$$

*Leakiness.* As described previously, we can define leaky count-based novelty in two different ways: (i) by assuming that the counts  $C^{(t)}(x_i)$  in Equation 10 decay with a fixed factor  $\alpha$  between two time steps (Equation 1), or (ii) by assuming a fixed learning rate  $\eta$  for the delta rule in Equation 11. In the following, we derive the closed-form expression and delta rule update for both types of leaky count-based novelty.

*Decaying counts.* Defining decaying counts (Equation 1) also affects the denominator of  $\theta_i^{(t+1)}$  in Equation 10:

$$p^{(t+1)}(x_i) = \theta_i^{(t+1)} = \frac{C^{(t)}(x_i) + \varepsilon}{T_t + N\varepsilon}, \quad (\text{Equation 50})$$

where, due to leaky counting,

$$T_t = \sum_{j=1}^N C^{(t)}(x_j) = (1 - \alpha) \sum_{j=1}^N C^{(t-1)}(x_j) + 1 = (1 - \alpha) T_{t-1} + 1 = \frac{1 - (1 - \alpha)^t}{\alpha}, \quad (\text{Equation 51})$$

which converges to  $1/\alpha$  as  $t$  goes to infinity.

Analogously to the non-leaky case, we can derive an iterative update rule for the familiarity  $p$ :

$$p^{(t+1)}(x_i) = \frac{C^{(t)}(x_i) + \varepsilon}{T_t + N\varepsilon} \quad (\text{Equation 52})$$

$$= (1 - \alpha) \frac{C^{(t-1)}(x_i) + \varepsilon}{T_t + N\varepsilon} + \frac{\alpha \varepsilon + \delta(s_t, x_i)}{T_t + N\varepsilon} \quad (\text{Equation 53})$$

$$= (1 - \alpha) p^{(t)}(x_i) \left( 1 - \frac{1 - \alpha T_{t-1}}{T_t + N\varepsilon} \right) + \frac{\alpha \varepsilon + \delta(s_t, x_i)}{T_t + N\varepsilon} \quad (\text{Equation 54})$$

$$= (1 - \alpha) (1 - \alpha_t (1 - \alpha T_{t-1})) p^{(t)}(x_i) + \alpha_t (\delta(s_t, x_i) + \varepsilon), \quad (\text{Equation 55})$$

with  $\alpha_t = \frac{1}{T_t + N\varepsilon}$ .

For flat prior ( $\varepsilon = 0$ ), this reduces to

$$p^{(t+1)}(x_i) = (1 - \alpha_t) p^{(t)}(x_i) + \alpha_t \delta(s_t, x_i), \quad (\text{Equation 56})$$

with

$$\alpha_t = \frac{\alpha}{1 - (1 - \alpha)^t} \xrightarrow{t \rightarrow \infty} \alpha \quad (\text{Equation 57})$$

**Fixed learning rate.** Updating the familiarity in Equation 49 with a fixed learning rate  $\eta$  instead of time-varying learning rate  $\alpha_t$  gives the following closed-form expression for  $\rho^{(t+1)}(x_i)$ :

$$\rho^{(t+1)}(x_i) = (1 - \eta)\rho^{(t)}(x_i) + \eta\delta(s_t, x_i) = (1 - \eta)^{t+1} \underbrace{\rho^{(1)}(x_i)}_{1/N} + \eta \sum_{t'=1}^t (1 - \eta)^{t-t'} \delta(s_{t'}, x_i) \quad (\text{Equation 58})$$

Note that, in contrast to decaying counts-familiarity, the fixed learning rate-familiarity is not influenced by the prior  $\epsilon$  as long as it is symmetric and the initial familiarity is thus  $\rho^{(1)}(x_i) = 1/N$  for all stimuli. By comparing Equations 56 and 58, however, we can see that the updates of decaying-counts familiarity and fixed learning rate-familiarity are *asymptotically equivalent* if we choose a flat prior  $\epsilon = 0$  and a fixed learning rate equal to the leak parameter,  $\eta = \alpha$ . In contrast, the learning rate in the non-leaky model converges to zero.

### Derivation of similarity-based novelty

We estimate the familiarity  $\rho^{(t+1)}$  at time point  $t + 1$ :

$$\rho^{(t+1)}(s) = \sum_{j=1}^N w_j^{(t)} k_j(s), \quad (\text{Equation 59})$$

by choosing the mixture weights at time  $t$ ,  $\mathbf{w}^{(t)} = (w_1^{(t)}, \dots, w_N^{(t)})^T$ , as the maximum-a-posteriori (MAP) estimate of the sequence  $s_{1:t}$  of observations up to  $t$ :

$$\mathbf{w}^{(t)} = \underset{\mathbf{w}}{\operatorname{argmax}} \log \Pr(s_{1:t} | \mathbf{w}) \Pr(\mathbf{w}), \quad (\text{Equation 60})$$

where  $\Pr(\mathbf{w})$  is a Dirichlet prior over the weights, i.e.

$$\Pr(\mathbf{w}) = c(\alpha) \prod_{j=1}^N (w_j)^{\alpha_j - 1}, \quad (\text{Equation 61})$$

with concentration parameters  $\alpha = (\alpha_1, \dots, \alpha_N)^T$  and

$$c(\alpha) = \frac{\Gamma\left(\sum_{j=1}^N \alpha_j\right)}{\prod_{j=1}^N \Gamma(\alpha_j)} \quad (\text{Equation 62})$$

**Classical EM for similarity-based novelty.** Building on Chapter 9 in Bishop and Nasrabadi<sup>49</sup>, we provide a condensed derivation of the classical EM update for the settings of similarity-based novelty (i.e. arbitrary non-negative mixture components).

To apply the EM algorithm to our mixture model in Equation 59, we define vectors of binary latent random variables  $\mathbf{z}_{t'} = (z_{1,t'}, \dots, z_{N,t'})^T$  for each time step  $t' = 1, \dots, t$ , such that the following three conditions hold:

$$\Pr(z_{j,t'} = 1 | \mathbf{w}) = w_j \text{ for all } j = 1, \dots, N \text{ and } t' = 1, \dots, t \quad (\text{Equation 63})$$

$$\Pr(s_{t'} | z_{j,t'} = 1) = k_j(s_{t'}) \text{ for all } j = 1, \dots, N \text{ and } t' = 1, \dots, t \quad (\text{Equation 64})$$

$$\sum_{j=1}^N z_{j,t'} = 1 \text{ for all } t' = 1, \dots, t. \quad (\text{Equation 65})$$

Note that the last condition, Equation 65, implies that each vector  $\mathbf{z}_{t'}$  is a one-hot-coding vector of the  $N$  component indices. The probability of a given  $\mathbf{z}_{t'}$  to be a one-hot-coding vector of index  $j$  is given by the component weight  $w_j^{(t)}$  (Equation 63), while the probability of observing stimulus  $s_{t'}$  when  $\mathbf{z}_{t'}$  is encoding index  $j$  is the value of component  $j$  at stimulus  $s_{t'}$  (Equation 64). The latent variables thus link the component activations for a given stimulus and the component weights.

Equations 63 and 64 allow us to write conditional distributions for the stimulus and latent variable at time  $t'$ ,  $s_{t'}$  and  $\mathbf{z}_{t'}$ :

$$\rho(s_{t'} | \mathbf{z}_{t'}) = \prod_{j=1}^N (k_j(s_{t'}))^{z_{j,t'}} \quad (\text{Equation 66})$$

$$\rho(\mathbf{z}_{t'} | \mathbf{w}) = \prod_{j=1}^N (w_j)^{z_{j,t'}}. \quad (\text{Equation 67})$$

We consider the observations  $s_{t'}$  to be independent (see discussion below). That allows us to further write the joint distributions

$$p(s_{1:t}|\mathbf{z}_{1:t}) = \prod_{t'=1}^t P(s_{t'}|\mathbf{z}_{t'}) = \prod_{t'=1}^t \prod_{j=1}^N (k_j(s_{t'}))^{z_{j,t'}} \quad (\text{Equation 68})$$

$$p(\mathbf{z}_{1:t}|\mathbf{w}) = \prod_{t'=1}^t P(\mathbf{z}_{t'}|\mathbf{w}) = \prod_{t'=1}^t \prod_{j=1}^N (w_j)^{z_{j,t'}}. \quad (\text{Equation 69})$$

*Log-likelihood decomposition.* The latent variables  $\mathbf{z}_t$  allow for a decomposition of the log-likelihood in Equation 60. Let  $q(\mathbf{z}_{1:t})$  be an arbitrary distribution over the latent variables  $\mathbf{z}_{1:t}$ , then we can write the log-likelihood as

$$\log \Pr(s_{1:t}|\mathbf{w})\Pr(\mathbf{w}) \stackrel{\text{law of total prob.}}{=} \sum_{\mathbf{z}_{1:t}} q(\mathbf{z}_{1:t}|\mathbf{w}) \log \Pr(s_{1:t}|\mathbf{w}) + \log \Pr(\mathbf{w}) \quad (\text{Equation 70})$$

$$= \sum_{\mathbf{z}_{1:t}} q(\mathbf{z}_{1:t}|\mathbf{w}) \log \left( \frac{\Pr(s_{1:t}, \mathbf{z}_{1:t}|\mathbf{w})}{q(\mathbf{z}_{1:t}|\mathbf{w})} \right) + \log \Pr(\mathbf{w}) - \sum_{\mathbf{z}_{1:t}} q(\mathbf{z}_{1:t}|\mathbf{w}) \log \left( \frac{\Pr(\mathbf{z}_{1:t}|s_{1:t}, \mathbf{w})}{q(\mathbf{z}_{1:t}|\mathbf{w})} \right) \quad (\text{Equation 71})$$

$$= \mathcal{L}(q(\mathbf{z}_{1:t}), p(s_{1:t}, \mathbf{z}_{1:t})) + \text{KL}(q(\mathbf{z}_{1:t})||p(\mathbf{z}_{1:t}|s_{1:t})), \quad (\text{Equation 72})$$

where  $p(\mathbf{z}_{1:t}|s_{1:t}) = \Pr(\mathbf{z}_{1:t}|s_{1:t}, \mathbf{w})$ . Based on this decomposition, the log-likelihood can be maximized by iteratively (i) minimizing the KL divergence (E-step) while keeping the weights  $\mathbf{w}$  fixed at their prior values, and (ii) maximizing the Q term with respect to  $\mathbf{w}$  while keeping the latent distribution  $q$  fixed (M-step). For the proof that this iterative procedure indeed finds a local maximum of the log-likelihood, see e.g. Bishop and Nasrabadi<sup>49</sup> and McLachlan and Krishnan.<sup>86</sup>

*E-step.* The KL-divergence of  $q$  and  $p$  becomes minimal when the two distributions are equal, so in the E-step, we set

$$q(\mathbf{z}_{1:t}|\mathbf{w}) = p(\mathbf{z}_{1:t}|s_{1:t}, \mathbf{w}) = \frac{\Pr(s_{1:t}|\mathbf{z}_{1:t}, \mathbf{w})\Pr(\mathbf{z}_{1:t}|\mathbf{w})}{\Pr(s_{1:t}|\mathbf{w})} = \prod_{t'=1}^t \prod_{j=1}^N \underbrace{\left( \frac{w_j k_j(s_{t'})}{\sum_{j'=1}^N w_{j'} k_{j'}(s_{t'})} \right)}_{\gamma_j^{(t')}}^{z_{j,t'}} \quad (\text{Equation 73})$$

where we have used the mutual independence of any two observations  $s_{t'}$ ,  $s_{t''}$  and any two latent variables  $\mathbf{z}_{t'}$  and  $\mathbf{z}_{t''}$ , respectively to obtain Equation 73, and substituted the distributions from Equations 59, 66, and 67. To compute the distribution  $q$  in the E-step, it is thus sufficient to compute the terms

$$\gamma_j^{(t')} = \frac{w_j k_j(s_{t'})}{\sum_{j'=1}^N w_{j'} k_{j'}(s_{t'})} \quad (\text{Equation 74})$$

also called ‘responsibilities’, for every  $j = 1, \dots, N$  and every observed stimulus  $s_{t'}$ ,  $t' = 1, \dots, t$ .

*M-step.* To maximize the term  $\mathcal{L}(q(\mathbf{z}_{1:t}), p(s_{1:t}, \mathbf{z}_{1:t}))$  with respect to  $\mathbf{w}$ , we keep the latent distribution  $q$  fixed (denoted by fixed weights  $\mathbf{w}_{E\text{-step}}$  that parametrize  $q$ ). Note that

$$\mathcal{L}(q(\mathbf{z}_{1:t}), p(s_{1:t}, \mathbf{z}_{1:t})) = \sum_{\mathbf{z}_{1:t}} q(\mathbf{z}_{1:t}|\mathbf{w}_{E\text{-step}}) \log \left( \frac{p(s_{1:t}, \mathbf{z}_{1:t}|\mathbf{w})}{q(\mathbf{z}_{1:t}|\mathbf{w}_{E\text{-step}})} \right) + \log \Pr(\mathbf{w}) \quad (\text{Equation 75})$$

$$= \sum_{\mathbf{z}_{1:t}} q(\mathbf{z}_{1:t}|\mathbf{w}_{E\text{-step}}) \log p(s_{1:t}, \mathbf{z}_{1:t}|\mathbf{w}) + \log \Pr(\mathbf{w}) - \sum_{\mathbf{z}_{1:t}} q(\mathbf{z}_{1:t}|\mathbf{w}_{E\text{-step}}) \log q(\mathbf{z}_{1:t}|\mathbf{w}_{E\text{-step}}). \quad (\text{Equation 76})$$

Since we keep the latent distribution  $q$  fixed throughout the maximization, the last term in Equation 76 is constant in  $\mathbf{w}$ . It is thus sufficient to maximize

$$Q(\mathbf{w}, \mathbf{w}_{E\text{-step}}) = \sum_{\mathbf{z}_{1:t}} q(\mathbf{z}_{1:t}|\mathbf{w}_{E\text{-step}}) \log p(s_{1:t}, \mathbf{z}_{1:t}|\mathbf{w}) + \log \Pr(\mathbf{w}) \quad (\text{Equation 77})$$

$$= \sum_{\mathbf{z}_{1:t}} q(\mathbf{z}_{1:t}|\mathbf{w}_{E\text{-step}}) \log \Pr(s_{1:t}|\mathbf{z}_{1:t}, \mathbf{w}) \Pr(\mathbf{z}_{1:t}|\mathbf{w}) + \log \Pr(\mathbf{w}) \quad (\text{Equation 78})$$

$$= \sum_{\mathbf{z}_{1:t}} \prod_{t'=1}^t \prod_{j=1}^N (\gamma_j^{(t')})^{z_{j,t'}} \sum_{t'=1}^t \sum_{j=1}^N z_{j,t'} \log(w_j k_j(s_{t'})) + \log \Pr(\mathbf{w}), \quad (\text{Equation 79})$$

where we have substituted the distributions in Equations 66 and 67 and the expression for  $q$  from the E-step. Since all  $\mathbf{z}_{t'}$  are one-hot coding vectors, exactly one  $z_{t',j}$  is non-zero for each  $t'$ ; and since we are summing over all possible one-hot coding sequences of

vectors  $\mathbf{z}_{1:t}$ , each  $z_{j,t'}$  will be non-zero exactly once. Equation 79 therefore reduces to

$$Q(\mathbf{w}, \mathbf{w}_{E\text{-step}}) = \sum_{t'=1}^t \sum_{j=1}^N \gamma_j^{(t')} (\log k_j(s_{t'}) + \log w_j) + \log \Pr(\mathbf{w}). \quad (\text{Equation 80})$$

We maximize  $Q$  as in Equation 80 under the constraint that all weights sum to one using Lagrange multipliers, by solving

$$0 = \frac{d}{dw_j} \left( Q(\mathbf{w}, \mathbf{w}_{E\text{-step}}) - \lambda \left( \sum_{j=1}^N w_j - 1 \right) \right) = \frac{1}{w_j} \sum_{t'=1}^t \gamma_j^{(t')} + \frac{d}{dw_j} \log \Pr(\mathbf{w}) - \lambda = \frac{1}{w_j} \sum_{t'=1}^t \gamma_j^{(t')} + \frac{\alpha_j - 1}{w_j} - \lambda \quad (\text{Equation 81})$$

Solving Equation 81 gives

$$\lambda = t + \sum_{j=1}^N (\alpha_j - 1), \quad (\text{Equation 82})$$

yielding the following solution for the new mixture weights (i.e. the M-step update):

$$w_j = \frac{1}{t + \sum_{j=1}^N (\alpha_j - 1)} \left[ \sum_{t'=1}^t \gamma_j^{(t')} + (\alpha_j - 1) \right] \quad (\text{Equation 83})$$

For a symmetric Dirichlet prior, i.e.  $\alpha_j = \varepsilon + 1$ , for all  $j = 1, \dots, N$ , the M-step update reduces to

$$w_j = \frac{1}{t + N\varepsilon} \left[ \sum_{t'=1}^t \gamma_j^{(t')} + \varepsilon \right]. \quad (\text{Equation 84})$$

*Online Incremental EM for similarity-based novelty.* Classical EM<sup>86,98</sup> as above estimates the mixture weights by iteratively applying E-step and M-step update until convergence, using all observations  $s_{1:t}$  in each single update. When modeling human and animals, however, it is more realistic to assume that the brain updates its estimate of the novelty of a stimulus immediately after its observation – even if that leads to a less precise estimate of the overall statistics of stimuli in the environment. This is captured by incremental EM, an approximation of EM that uses only a single observation in each update and iterates repeatedly through the sequence of observations  $s_{1:t}$  one-by-one until the algorithm has converged. For our case of humans and animals experiencing an unfamiliar environment, we further assume that the incremental EM algorithms uses a continuing sequence of observations for its estimates, such that each observation is used only once for the update of the algorithm. The resulting weight estimates are in that sense ‘approximate’ estimates of the component weights but also the best estimates that are available for a potentially infinite sequence of observations. To distinguish our specific use of the incremental EM algorithm from its standard ‘finite stimulus sequence’ usage, we also refer to it as ‘online incremental EM algorithm’ (OI-EM).

In OI-EM, each observation  $s_t$  is followed by a single iteration of E- and M-step, and each E-step only recomputes the responsibilities for the current observation  $s_t$  (all other responsibilities are kept the same):

$$\gamma_{j,t}^{(t)} = \begin{cases} \frac{w_j^{(t-1)} k_j(s_t)}{\sum_{j=1}^N w_j^{(t-1)} k_j(s_t)} & \text{if } t' = t \\ \gamma_{j,t}^{(t-1)} & \text{else.} \end{cases} \quad (\text{Equation 85})$$

The M-step of the algorithm is the same as in the full EM algorithm (Equation 84) – just the underlying responsibilities are different due to the incremental E-step:

$$w_j^{(t)} = \frac{1}{t + N\varepsilon} \left[ \sum_{t'=1}^t \gamma_{j,t'}^{(t)} + \varepsilon \right] = \frac{1}{t + N\varepsilon} \left[ \sum_{t'=1}^t \gamma_{j,t'}^{(t')} + \varepsilon \right]. \quad (\text{Equation 86})$$

Equation 86 gives rise to an iterative update rule (also see first section of the STAR Methods, where we dropped the iteration index  $^{(t')}$  for better readability):

$$w_j^{(t)} = \frac{t - 1 + N\varepsilon}{t + N\varepsilon} w_j^{(t-1)} + \frac{1}{t + N\varepsilon} \gamma_{j,t}^{(t)} = (1 - \alpha_t) w_j^{(t-1)} + \alpha_t \gamma_{j,t}^{(t)}, \quad (\text{Equation 87})$$

with time-dependent learning rate  $\alpha_t = \frac{1}{t + N\varepsilon}$ ,  $\gamma_{j,t}^{(t)}$  as in Equation 85, and initial weights  $w_j^{(0)} = \frac{1}{N}$ .

*Leaky similarity-based novelty.* As for count-based novelty, leaky similarity-based novelty can be implemented in two ways: (i) by assuming that, in the M-step of the EM-algorithm, the weights are computed as the *leaky* sum of past responsibilities (see Equation 22), i.e.

$$w_j = \frac{1}{T_t + N\epsilon} \left[ \sum_{t'=1}^t (1 - \alpha)^{t-t'} \gamma_j^{(t')} + \epsilon \right], \quad (\text{Equation 88})$$

with

$$T_t \stackrel{\text{Eq. 23}}{=} \sum_{t'=1}^t \sum_{j=1}^N (1 - \alpha)^{t-t'} \gamma_j^{(t')} = \frac{1 - (1 - \alpha)^t}{\alpha} \xrightarrow{t \rightarrow \infty} \frac{1}{\alpha}; \quad (\text{Equation 89})$$

or (ii) by assuming a *fixed learning rate*  $\eta$  instead of the time-dependent learning rate  $\alpha_t$  (see Equation 24).

*Decaying responsibilities.* For decaying responsibilities as defined in Equation 22, the iterative weight update rule is given by:

$$w_j^{(t)} = \frac{1}{T_t + N\epsilon} \left[ \sum_{t'=1}^t (1 - \alpha)^{t-t'} \gamma_j^{(t')} + \epsilon \right] \quad (\text{Equation 90})$$

$$= \frac{1}{T_t + N\epsilon} (\gamma_j^{(t)} + \alpha\epsilon) + (1 - \alpha) \frac{T_{t-1} + N\epsilon}{T_t + N\epsilon} w_j^{(t-1)} \quad (\text{Equation 91})$$

$$= (1 - \alpha)(1 - \alpha_t(1 - \alpha T_{t-1})) w_j^{(t-1)} + \alpha_t (\gamma_j^{(t)} + \alpha\epsilon), \quad (\text{Equation 92})$$

with  $\alpha_t = \frac{1}{T_t + N\epsilon}$  and  $T_t$  as in Equation 23. This gives the familiarity update:

$$p^{(t+1)}(s) = (1 - \alpha)(1 - \alpha_t(1 - \alpha T_{t-1})) p^{(t)}(s) + \alpha_t \sum_{j=1}^N (\gamma_j^{(t)} + \alpha\epsilon) k_j(s) \quad (\text{Equation 93})$$

*Fixed learning rate.* A fixed learning rate in the familiarity update (Equation 19) is equivalent to using a fixed learning rate in the weight update (Equation 87). Analogously to the count-based fixed learning rate-estimator (see above), the weights at a given time  $t$  are then the weighted average between the initial weight and the discounted sum of responsibilities up to time  $t$ :

$$w_j^{(t)} = (1 - \eta) w_j^{(t-1)} + \eta \gamma_j^{(t)} = (1 - \eta)^t w_j^{(0)} + \underbrace{\eta}_{1/N} \sum_{t'=1}^t (1 - \eta)^{t-t'} \gamma_j^{(t')}. \quad (\text{Equation 94})$$

### Behavioral fit: MLE for N-RL models

The following calculations are based on Daw et al.<sup>99</sup> To compute the log-likelihood (Equation 39) for a given N-RL model  $\mathcal{M}$  to explain the data  $\mathcal{D}$ , we use the Markov property of N-RL models:

$$\log P(\mathcal{D} | \mathcal{M}, \theta) = \log \prod_{i=1}^{20} P(s_0^{(i)}, a_0^{(i)}, \dots, a_{T_i-1}^{(i)}, s_{T_i}^{(i)} | \mathcal{M}, \theta) \quad (\text{Equation 95})$$

$$= \sum_{i=1}^{20} \left( \log P(s_0^{(i)} | \mathcal{M}, \theta) + \sum_{t=1}^{T_i} \log P(a_{t-1}^{(i)} | s_{t-1}^{(i)}; \mathcal{M}, \theta) + \sum_{t=1}^{T_i} \log P(s_t^{(i)} | a_{t-1}^{(i)}; \mathcal{M}, \theta) \right). \quad (\text{Equation 96})$$

Since all mice and agents start in the home cage by experiment design, we find  $P(s_0^{(i)} | \mathcal{M}, \theta) = 1$ ; and since the environment is deterministic, we can further simplify  $P(s_{t+1}^{(i)} | a_t^{(i)}; \mathcal{M}, \theta) = 1$ . Therefore, Equation 96 simplifies to

$$\log P(\mathcal{D} | \mathcal{M}, \theta) = \sum_{i=1}^{20} \sum_{t=1}^{T_i} \log P(a_{t-1}^{(i)} | s_{t-1}^{(i)}; \mathcal{M}, \theta). \quad (\text{Equation 97})$$

Since our N-RL models use a softmax policy, each individual log-likelihood is given as

$$\log P(a_{t-1}^{(i)} | s_{t-1}^{(i)}; \mathcal{M}, \theta) = \beta Q_{\mathcal{M}, \theta}^{(t-1)}(s_{t-1}^{(i)}, a_{t-1}^{(i)}) - \log \sum_{a'} \exp(\beta Q_{\mathcal{M}, \theta}^{(t-1)}(s_{t-1}^{(i)}, a')), \quad (\text{Equation 98})$$

with inverse temperature  $\beta$ .

### QUANTIFICATION AND STATISTICAL ANALYSIS

All analyses were performed using Python. Standard errors are computed either (i) using jackknife resampling on the available experimental data points (passive viewing task) or (ii) using bootstrapping on the number of mice or agents (behavior statistics in active exploration task). Details are specified in the respective figure legends and the method details. Significance for the model comparison in the active exploration task is assessed using the established interpretation scale for log-evidence, i.e. a log-evidence difference  $\geq 3$  is significant.<sup>60,61</sup> Leave-one-out cross-validation with jackknife-resampled SE (passive viewing task) and 5-fold cross-validation (active exploration task) are performed to control for overfitting.