Flexible inference for animal learning rules using neural networks

Yuhan Helena Liu[1,*], Victor Geadah[1], and Jonathan Pillow[1,*]

[1]Princeton University, Princeton, NJ, USA
[*]Correspondence: hl7582@princeton.edu, pillow@princeton.edu

Abstract

Understanding how animals learn is a central challenge in neuroscience, with growing relevance to the development of animal- or human-aligned artificial intelligence. However, existing approaches tend to assume fixed parametric forms for the learning rule (e.g., Q-learning, policy gradient), which may not accurately describe the complex forms of learning employed by animals in realistic settings. Here we address this gap by developing a framework to infer learning rules directly from behavioral data collected during *de novo* task learning. We assume that animals follow a decision policy parameterized by a generalized linear model (GLM), and we model their learning rule—the mapping from task covariates to per-trial weight updates—using a deep neural network (DNN). This formulation allows flexible, data-driven inference of learning rules while maintaining an interpretable form of the decision policy itself. To capture more complex learning dynamics, we introduce a recurrent neural network (RNN) variant that relaxes the Markovian assumption that learning depends solely on covariates of the current trial, allowing for learning rules that integrate information over multiple trials. Simulations demonstrate that the framework can recover ground-truth learning rules. We applied our DNN and RNN-based methods to a large behavioral dataset from mice learning to perform a sensory decision-making task and found that they outperformed traditional RL learning rules at predicting the learning trajectories of held-out mice. The inferred learning rules exhibited reward-history–dependent learning dynamics, with larger updates following sequences of rewarded trials. Overall, these methods provide a flexible framework for inferring learning rules from behavioral data in *de novo* learning tasks, setting the stage for improved animal training protocols and the development of behavioral digital twins.

1 Introduction

The study of animal learning has a long history, beginning with foundational work such as Pavlov's classical conditioning and the Rescorla–Wagner model of associative learning [1, 2]. Uncovering the mechanisms that underlie learning and decision-making is crucial for understanding and predicting animal behavior [3]. This knowledge has wide-reaching applications, including benefits for human health and medicine [4–7], informing conservation efforts by anticipating how animals respond to environmental change [8–12], and potentially guiding the development of biologically aligned AI systems, which have garnered increasing interest in recent years [13, 14].

While inference of learning rules is gaining traction, prior studies have key limitations. For example, many leverage reinforcement learning (RL) paradigms with normative approaches that assume reward maximization [15–17] or weight update under a parametric learning rule (e.g., Q-learning [18], REINFORCE [19, 20]). Although these approaches have provided clear insights into animal learning,

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

they may lack the flexibility to capture the detailed structure of real learning rules, which are often unknown and may not align with parametric forms conceived by human researchers. As a result, existing models may span only a small subset of the space of possible learning strategies animals actually use. Recent work has proposed more flexible quantitative models of learning based on recurrent neural networks (RNNs) [21–24] or mechanisms proposed by large language models [25]. However, these studies have generally focused on "bandit" style tasks, in which animals adapt their behavior to time-varying rewards in a fixed task structure, as opposed to learning a novel task or behavior from scratch [26, 27]. De novo learning is common in experimental neuroscience, as animals are routinely trained to acquire novel input–output mappings from scratch, yet comparatively understudied. Finally, a variety of descriptive models have been developed to characterize trial-by-trial changes in weights that govern an animal's policy as it evolves over learning, but do not identify any rules governing these weight updates [28, 29].

To address these shortcomings, we propose a new approach with the following **main contributions**. To our knowledge, this is the first approach to infer nonparametric, non-Markovian learning rules directly from behavior in de novo tasks.

1. **Inferring flexible learning rules using DNNs.** We develop a deep neural network (DNN) modeling approach to infer flexible animal learning rules from de novo task learning data (Figs. 1 and 2).
2. **Non-Markovian learning rules.** Many classic learning rules are Markovian or "memoryless", meaning that the learning update depends only on the input, action, and reward on the current time step. Such rules, including our own DNN approach, thus fail to capture "non-Markovian" learning, where updates depend on multiple time steps of trial history. To overcome this limitation, we incorporate a recurrent neural network (RNN) to capture arbitrary dependencies of learning on trial history (Fig. 3).
3. **Improved held-out prediction on real data.** We apply our methods to mouse training data from the International Brain Laboratory (IBL) and observe significantly higher log-likelihood on held-out data compared to existing approaches (Fig. 4).
4. **Insights into animal learning strategies.** We analyze the inferred learning rules and identify several departures from classic policy gradient learning, including non-Markovian reward history dependencies lasting multiple trials (Fig. 5).

2 Model and methods

Model of decision-making. We begin with a dynamic Bernoulli generalized linear model (GLM) of an animal's time-varying behavior during learning [19, 20, 28, 30]. This provides an interpretable model of an animal's decision-making policy in terms of a set of weights that evolve dynamically over the course of learning. This differs from recent work in which the animal's policy is directly parametrized by an RNN [21, 22], which offers increased flexibility but may be challenging to interpret.

On trial t, a sensory stimulus $s_t \in [-1, 1]$ appears on the left or right side of the screen, where $s_t > 0$ indicates the fractional contrast of a right-side stimulus and $s_t < 0$ indicates the (negative) fractional contrast of a left-side stimulus. The mouse makes a binary choice $y_t \in \{0, 1\}$ by turning a wheel, where 0 corresponds to a leftward and 1 to a rightward choice, respectively (Fig. 1A). The true label of the stimulus is denoted as $z_t = \mathbb{1}[s_t > 0]$, representing the correct response according to the task.

We assume the animal's policy on trial t is governed by a weight vector $\mathbf{w}_t \in \mathbb{R}^d$, which describes how task covariates, denoted $\mathbf{x}_t \in \mathbb{R}^d$, influence the animal's choice. Unless otherwise specified, we will define the input vector to be $\mathbf{x} = [s_t, 1]^\top$, consisting of the signed stimulus intensity s_t and a constant or "bias" term [30]. This input interacts linearly with the weight vector $\mathbf{w}_t$, determining the probability that the animal selects the "rightward" choice on trial t:

$$p(y_t = 1 \mid \mathbf{x}_t, \mathbf{w}_t) = \frac{1}{1 + e^{-\mathbf{x}_t^\top \mathbf{w}_t}}. \tag{1}$$

The reward function is given by $r_t = \mathbb{1}[y_t = z_t]$, which corresponds to a positive reward of 1 if the animal makes a correct decision ($y_t = z_t$) and a reward of 0 otherwise.

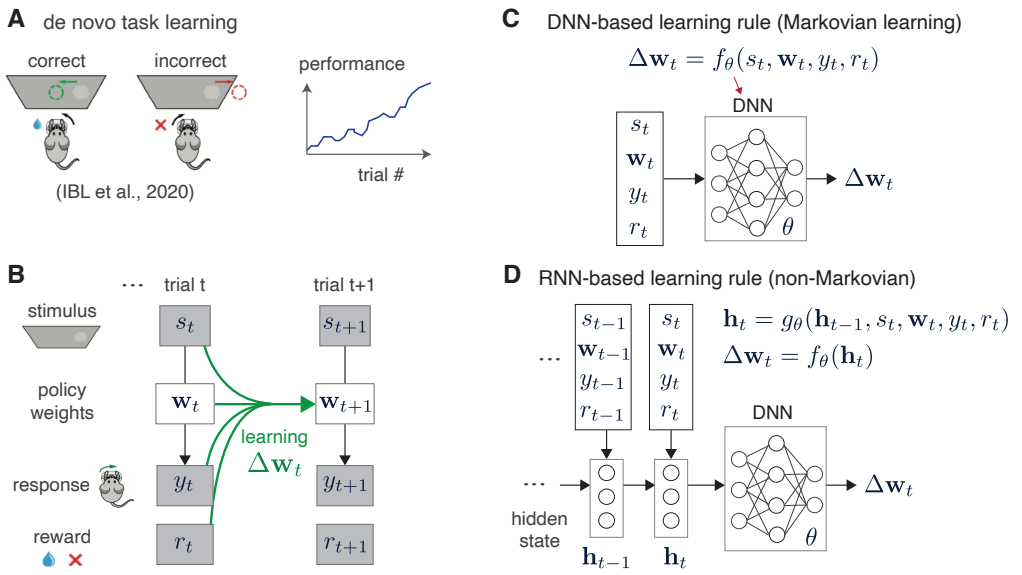

Figure 1: **Task schematic and learning rule inference methods**. **(A)** We examine learning of a sensory decision-making task, in which mice must learn to report which side of the screen contains a visual stimulus by turning a wheel [31]. **(B)** In our framework, decision-making is governed by the weights of a Bernoulli generalized linear model (GLM), which evolve across trials according to an unknown learning rule. **(C-D)** To infer the learning rule, we approximate the weight update function $\Delta \mathbf{w}_t$ using either (C) a deep neural network (DNN) that maps the current trial covariates to a weight change; or (D) a recurrent neural network (RNN) that integrates information from previous trials before feeding into the DNN. We optimized the neural network model parameters θ by maximizing the log-probability of the animal choice data under the dynamic Bernoulli GLM (see Methods and Algorithm 1). We refer to our approaches as **DNNGLM** or **RNNGLM** depending on the model used to parametrize the learning rule.

DNNs for learning rule inference. Learning in this paradigm is captured by updates to the decision weights $\mathbf{w}$. We first model the weight update function using a feedforward neural network (Fig. 1C):

$$\Delta \mathbf{w}_t = f_\theta(\mathbf{w}_t, \mathbf{x}_t, y_t, r_t), \tag{2}$$

where f_θ is a feedforward network with two hidden layers and trainable parameters θ (see Supp. C for architecture details). The next weight is given by:

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \Delta \mathbf{w}_t. \tag{3}$$

Initial weights $\mathbf{w}_0$ can be treated as trainable parameters, initialized via psychometric curve estimates (Supp. C). The inferred weights $\mathbf{w}_t$ parameterize the GLM response distribution (Eq. 1), and network parameters θ are optimized to maximize the log-likelihood of observed choices:

$$\ell_t(\mathbf{w}_t; \mathbf{x}_t) = y_t \log p(y_t = 1 \mid \mathbf{x}_t, \mathbf{w}_t) + (1 - y_t) \log(1 - p(y_t = 1 \mid \mathbf{x}_t, \mathbf{w}_t)), \tag{4}$$

which corresponds to minimizing binary cross-entropy loss.

RNNs for learning rule inference. Since feedforward networks are memoryless, we also consider a recurrent parameterization using a GRU [32] to encode history. The same inputs $\{\mathbf{w}_t, \mathbf{x}_t, y_t, r_t\}$ are passed into a GRU g_θ to produce a hidden state $\mathbf{h}_t$, which is then passed to the feedforward network:

$$\mathbf{h}_t = g_\theta(\mathbf{h}_{t-1}, \mathbf{w}_t, \mathbf{x}_t, y_t, r_t), \tag{5}$$
$$\Delta \mathbf{w}_t = f_\theta(\mathbf{h}_t). \tag{6}$$

The model is trained using the same objective and procedure as above (see also Supp. C).

By combining a GLM for choice modeling with neural network–based learning rules, we balance interpretability and flexibility. We refer to the two models as **DNNGLM** and **RNNGLM**, depending

on whether the learning rule is parameterized with a feedforward or recurrent network. Cross-validation, together with the implicit regularization properties of neural networks, helps mitigate overfitting; further details regarding our model and data processing can be found in Supp. C. The full procedure is outlined in Algorithm 1.

Algorithm 1 Learning Rule Inference from Trial-by-Trial Behavior

Inputs: Trial sequence $\{s_t, y_t, r_t\}_{t=1}^{T}$
for trial $t = 1$ to T **do**
  Construct input vector: $u_t = [s_t, y_t, r_t, \mathbf{w}_t]$
  Compute weight updates: $\Delta \mathbf{w}_t \leftarrow \mathrm{NN}(u_t)$
  Update the inferred GLM weights: $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + \Delta \mathbf{w}_t$
  Predict choice probability: $P_{y_t} = \sigma(\mathbf{w}_t^\top x_t)$
  Accumulate binary cross-entropy loss: $\mathcal{L}_t = -\left[y_t \log P_{y_t} + (1 - y_t) \log(1 - P_{y_t})\right]$
end for
Objective: Minimize total loss $\mathcal{L} = \sum_{t=1}^{T} \mathcal{L}_t$ using gradient descent

3 Simulation results: recovery of ground-truth learning rules

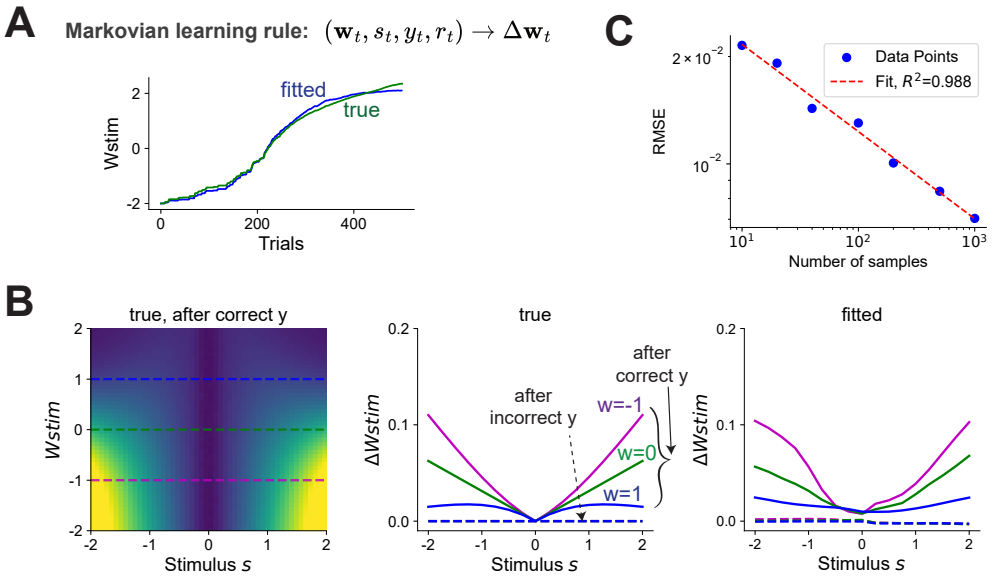

Figure 2: **Recovering ground-truth learning rules in simulated data.** We simulated animal learning using a policy gradient method known as "REINFORCE" [33]. (See Supp. Fig. 6 for an alternative ground-truth learning rule). This rule is Markovian in that the weight update $\Delta \mathbf{w}_t$ depends only on current-trial variables. **(A)** Simulated trajectory for stimulus weight w_{stim} for an example simulated mouse (green) and the inferred weights using the DNNGLM (blue). **(B)** Left: Heatmap showing the ground-truth stimulus weight change Δw_{stim} following a correct choice, as a function of w_{stim} and stimulus s. Horizontal dashed lines indicate w_{stim} or w slices shown in middle and right panels. Middle: slices showing stimulus weight change Δw_{stim} as a function of the stimulus s, for different values of the true weight w_{stim}, after both correct (solid) and incorrect (dashed) decisions. (The REINFORCE algorithm exhibits no learning after incorrect trials for this setting of rule parameters). Right: corresponding slices through the weight update function inferred using the DNNGLM. Note that the model successfully captures key characteristics of the learning rule, such as the slowing of learning at higher weights, increased learning with stimulus amplitude, and the asymmetry in learning after correct and incorrect choices. (See Supp. Fig. 7 for analogous plots for bias parameter updates and Supp. Fig. 6 for a comparison to other learning rules.) **(C)** Error in recovered learning rule as a function of dataset size, indicating that the DNNGLM converges to the true REINFORCE learning rule with increasing amounts of data.

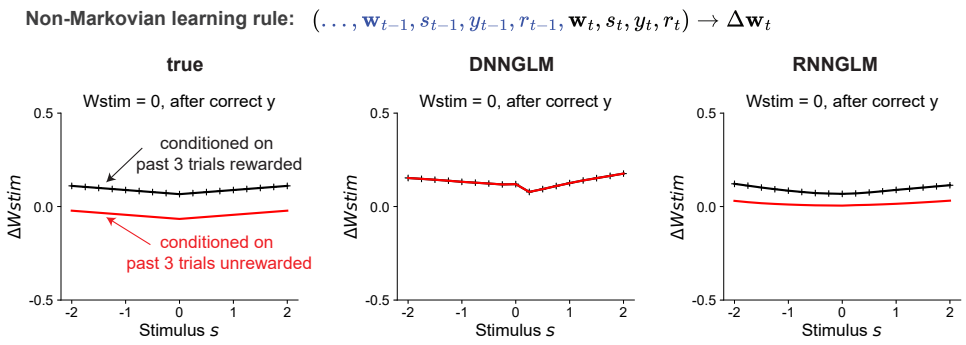

Figure 3: **Simulated example with a non-Markovian learning rule.** The RNNGLM recovered the reward history dependence present in the ground truth: larger weight updates occur when the past 3 trials were rewarded (black) versus unrewarded (red). Such differences were absent in DNNGLM, as it can only account for Markovian learning rules, where weight updates depend on current trial information. The ability of RNNGLM to capture these effects reflects its greater flexibility. Minor deviations from the ground truth are likely due to finite data (Fig. 2C). To keep the figure simple, we only plotted with the learning rule given stimulus weight $w_{stim} = 0$ and after correct choices, but we observed similar phenomena for other parameter settings.

To begin our investigation, we first consider simulated data where the ground truth is known. We consider two types of learning rules: **Markovian** and **non-Markovian**. By Markovian learning rules, we mean that weight updates depend only on current-trial variables and are thus conditionally independent of stimuli, choices, and rewards on previous trials. In contrast, non-Markovian learning rules allow weight updates on the current trial to exhibit arbitrary dependencies on task history.

For Markovian learning rules, we considered a classic policy gradient learning rule known as REINFORCE [33], which is defined by:

$$\begin{aligned}\Delta\mathbf{w_t} &\propto r_t \nabla_{\mathbf{w}} \log p(y_t \mid \mathbf{x}_t, \mathbf{w_t}) \\ &= r_t \epsilon_{y_t}(1 - p_{y_t}) x_t, \quad \epsilon_{y_t=R} = +1, \quad \epsilon_{y_t=L} = -1,\end{aligned} \tag{7}$$

where $p_{y_t} := p(y_t \mid \mathbf{x}_t, \mathbf{w_t})$, the probability of the choice y_t given input $\mathbf{x}_t$ under the animal's current policy. REINFORCE is a policy gradient method that uses Monte Carlo integration to evaluate the gradient of the policy with respect to reward (details in Supp. B.1).

We first validated our framework on synthetic data where the ground truth learning rule is known (Fig. 2). When simulating animal learning under REINFORCE [33], our method (DNNGLM) recovered the weight trajectory and key properties of the learning rule, including dependencies on decision outcome, stimulus contrast, and current weight. Reconstruction accuracy—measured as the log base 10 of RMSE between the ground-truth $\Delta\mathbf{w}$ and the model-predicted $\Delta\mathbf{w}$ on the training data—also improves with increasing data (Fig. 2C).

We then simulated a non-Markovian learning rule in which weight updates depended on trial history (Fig. 3). Specifically, we consider a modified REINFORCE rule with an eligibility-trace–like factor derived from a non-Markovian task setting (see Supp. B.2 for details):

$$\Delta\mathbf{w}_t \propto r_t \sum_s \epsilon_{y_{t-s}}(1 - p_{y_{t-s}})\mathbf{x}_{t-s}. \tag{8}$$

A key property of this rule is that a correct and rewarded past trial contributes a nonnegative term to the sum — since $\epsilon_{y_{t-s}}$ and $\mathbf{x}_{t-s}$ share the same sign and the remaining terms are nonnegative — while an incorrect trial contributes a nonpositive term. Thus, conditioning on several past trials being rewarded (versus unrewarded) leads to larger weight updates under the ground truth learning rule. This history-dependent effect is successfully captured by the fitted RNNGLM (Fig. 3).

In this case, the DNNGLM model fails to recover the correct learning dynamics; RNNGLM recovers the reward-history dependence in the learning rule. This demonstrates the need for recurrent architectures in modeling non-Markovian, history-dependent learning processes. In principle, the weights $\mathbf{w}$ could carry some history even in DNNs, since $\mathbf{w} \leftarrow \mathbf{w} + \Delta\mathbf{w}$ integrates past updates. However,

the learned update function $\Delta \mathbf{w}$ itself does not explicitly depend on past trial information and thus cannot capture history-dependent learning dynamics (see Fig. 3). While expanding the DNN input to include historical information is possible, this would require manual feature engineering and reduce the model's ability to automatically determine which past inputs are relevant.

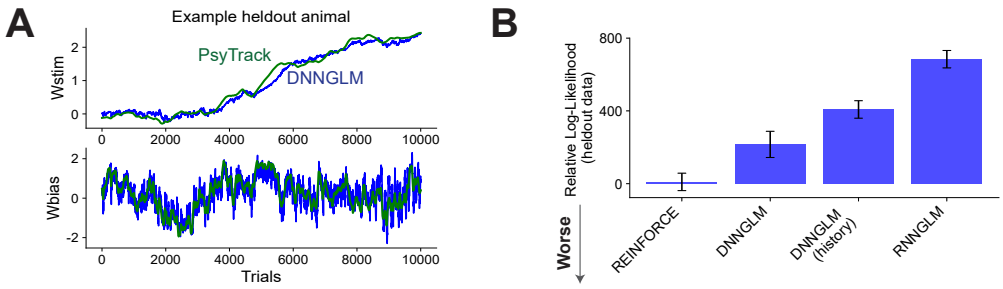

Figure 4: **Application to IBL mouse learning data. (A)** Predicted stimulus (top) and bias (bottom) weight trajectories for a held-out animal — using DNNGLM trained on other animals and applied to this animal's stimulus and choice sequence — plotted along the weight trajectories inferred by PsyTrack [28], which is a learning-agnostic and reliable method for tracking psychometric weights from behavioral time series. (Supp. Fig. 15 shows additional weight trajectories.) Similar traces were observed for RNNGLM as well. **(B)** Our methods (DNNGLM and RNNGLM) achieved significantly higher test log-likelihood (LL) on held-out data. Here we plot LL relative to the REINFORCE model. Notably, the RNNGLM model also out-performed the DNNGLM extended to include the previous trial input ("DNNGLM- history"). Error bars reflect standard deviation across cross-validation seeds.

We also add more results in the Supp. Material demonstrating that our main findings are robust across a variety of conditions: (i) a learning rule without the decision outcome assymetry (Supp. Fig. 6); (ii) recovery of an in-between learning rule when mixing update functions across individuals (Supp. Fig. 11); (iii) different initial stimulus weights (Supp. Fig. 9); (iv) longer trial lengths (Supp. Fig. 10). While the initial policy weights are typically unknown, we estimated them from the psychometric curve (Supp. C) and demonstrated recovery in Supplementary Fig. 10. For faster simulations in the main text, and to allow multiple repetitions for robustness testing, we used shorter trial sequences. These shorter sequences make it difficult to estimate the initial weight due to fewer samples at the beginning, so we initialized with a known initial weight in this case. Robustness tests showing recovery with incorrect W_0 are provided in Supplementary Table 5. We also show sensitivity to the initial weight estimate (Supp. Table 3) and degradation of recovery under added noise (Supp. Table 4).

4 Application to mouse learning data

For real data, we used the publicly available International Brain Laboratory (IBL) dataset [31], following the same preprocessing as in [19]. We first fit the learning rule on a pool of training animals. We then took the stimulus and choice sequence from a held-out test animal and applied the fitted DNNGLM or RNNGLM to predict the evolution of its stimulus and bias weights. We compared the predicted weight trajectories to those inferred by PsyTrack [28], a learning-agnostic and reliable method for tracking psychometric weights from behavior, and observed close agreement (Fig. 4A). Furthermore, using the predicted weights to compute the choice log-likelihood on held-out animals, both DNNGLM and RNNGLM outperformed REINFORCE-based models (Fig. 4B and Table 1). We also assessed temporal generalization on future held-out data and observed similar trends (Supp. 2). Notably, RNNGLM also outperformed history-augmented DNNGLM, in which previous-trial information was added as an additional regressor to the GLM.

Moreover, our methods revealed empirical features of mouse learning that deviate from classical learning rules (Fig. 5A). These include the presence of a "negative baseline" (coined in [20] but also observed in our flexible framework), where weight updates following incorrect trials actually *decreased* task accuracy — even though a positive update would aid learning in this particular task. We also observed asymmetries in update magnitude based on decision correctness, with distinct updates following correct (solid lines) versus incorrect (dashed lines) choices, as well as asymmetries

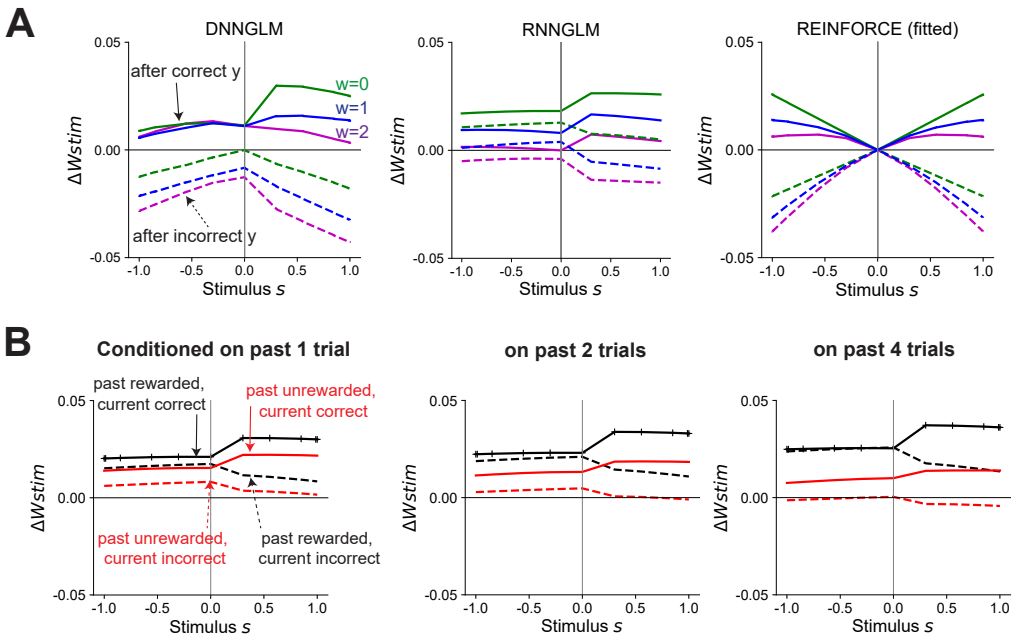

Figure 5: **Properties of inferred learning rules. (A)** For the dataset in Fig. 4, DNNGLM reveals negative weight updates following errors (coined as 'negative baseline' in [20], which was also observed in our flexible framework). Bias weight plot is provided in Supp. Fig. 8; note $\mathbf{w}$ is fixed at positive weights because the training data mainly involved positive $\mathbf{w}$. We follow the same plotting convention in Fig. 2; since RNNGLM depends on additional historical variables, we plot the mean Δw_{stim} averaged across these historical dimensions. **(B)** RNNGLM suggests non-Markovity (history dependence) in learning: larger weight updates are observed when all past trials were rewarded (black) versus unrewarded (red). This gap widens as we condition on history beyond just the previous trial, **suggesting that the history dependency extends beyond the most recent past trial**. We note that plotting up to four past trials was intended only as an illustrative example, not a fundamental limit of RNNGLM; as shown in Supp. Fig. 13, the model suggests longer history dependencies. To keep the figure simple, we only plotted with stimulus w fixed at 0, but similar trends are observed elsewhere.

across stimulus sides (i.e., different update functions for left vs. right choices). Notably, such side asymmetries have also been reported in prior work [34].

In addition, the fitted RNNGLM — achieving the best held-out log-likelihood — showed higher updates following sequences of rewarded trials (Fig. 5B), suggesting sensitivity to reward history spanning multiple trials (see also Supp. Fig. 11B). Together, these findings highlight the framework's ability to improve predictive accuracy while uncovering mechanistic insights into animal learning.

Because inferring flexible learning rules from binary choices is an under-constrained inverse problem, multiple parameterizations can fit the data similarly well. We therefore repeated inference for each model (DNNGLM, RNNGLM) across multiple random seeds, holding data splits and hyperparameters fixed. Across seeds, log-likelihoods were similar, while the recovered update functions showed modest variation. To visualize this, we now plot mean and standard deviation across of weight updates across seeds (Supp. Fig. 12). Despite some variability, **the key characteristics are consistent across seeds**—a negative baseline, side asymmetry, and multi-trial (non-Markovian) dependence on reward/stimulus history. Moreover, the elimination of these features suggests they are functionally meaningful: enforcing a non-negativity constraint on the baseline reduces held-out predictive performance (Table 1). Additionally, to assess fairness, we matched parameter counts between DNNGLM and RNNGLM, and observed that RNNGLM still outperformed ($p = 5.53e-4$), indicating the gap is not due to just model size.

Comparison	Heldout LL	p-val
REINFORCE vs. **DNNGLM**	-60576 vs. $\mathbf{-60360}$	$1.57\text{e}{-2}$
DNNGLM vs. **RNNGLM**	-60360 vs. $\mathbf{-59892}$	$1.79\text{e}{-8}$
REINFORCE vs. nonnegative base	$\mathbf{-60576}$ vs. -68532	$1.25\text{e}{-5}$
tinyRNN vs. **RNNGLM**	-60564 vs. $\mathbf{-59892}$	$2.12\text{e}{-3}$
REINFORCE (history) vs. **RNNGLM**	-60504 vs. $\mathbf{-59892}$	$7.73\text{e}{-7}$
DNNGLM (history) vs. **RNNGLM**	-60168 vs. $\mathbf{-59892}$	$3.84\text{e}{-6}$

Table 1: Animal-level held-out log-likelihoods (higher is better) for Fig. 4B, except that we report total log-likelihood (summed across animals) instead of relative log-likelihood. We show p-values from paired t-tests across mice, comparing different learning rule models on the IBL dataset. Each entry shows two values (one per model in the comparison), reported as mean ± standard error across cross-validation seeds. For the interpretable RNN baseline — which predicts behavior directly rather than parameterizing weights or weight updates — we used tinyRNN [21] with 8 GRUs, more than the 1–4 GRUs used in their original paper, which performed even worse in our tests. Results for held-out future data also show consistent trends (Supp. Table 2).

Finally, **Proposition 1** in Supp. A provides theoretical support for our approach in highly simplified settings. A full investigation of identifiability and convergence is left for future work.

5 Related Work

Previous studies have shown that artificial neural networks provide a more accurate description of animal learning behavior than classical learning models [35–52]. Several recent works have taken steps to enhance the interpretability of inferred learning strategies — for example, by regularizing the latent space or parameterizing in low-dimensional or human-interpretable forms [21–23, 25], but they have not addressed *de novo* learning, i.e., learning to perform a new sensory-motor task behavior structure (input–output mappings) from scratch.

Methods for inferring learning rules during the acquisition of new task structures remain sparse. Those that do exist typically assume a predefined functional form [18–20, 34], limiting their ability to capture learning strategies that differ from hand-designed models. In addition, many are memoryless or 'Markovian', assuming that weight updates depend only on the task variables encountered on the current trial; this stands in contrast to biological learning, where history plays a prominent role — such as through eligibility traces [53, 54] or the influence of trial history on decision-making [30, 55–57]. In this work, we focus on inferring learning rules from behavioral data, motivated by the broad availability of behavioral datasets and the potential for large-scale inference without requiring simultaneous neural recordings. In parallel, several studies have proposed approaches to infer synaptic-level plasticity rules [58–66], which offer detailed insights into neural mechanisms. While some of these approaches can also be applied to behavioral data (e.g., [62]), their focus has largely been on neural-level inference and often requires neural activity recordings or handcrafted functional bases for interpretation. Our method instead emphasizes flexible inference directly from behavior, improving weight recovery via psychometric-curve initialization and extending flexibility with recurrent networks to capture multi-trial history beyond bandit tasks.

Beyond neuroscience, our work is also related to the meta-learning literature. A prominent line of work learns optimizers or update rules via gradient-based meta-learning [67], with extensions to reinforcement learning [44, 68]; several computational neuroscience studies likewise meta-learn synaptic plasticity rules within interpretable frameworks [41, 63, 69, 70]. In contrast to meta-learning approaches that optimize agents to learn well across tasks, we address a fundamentally different goal: inferring, from behavior alone and in de novo tasks, the learning rule that a biological agent used to learn from scratch. This poses distinct challenges: (i) animals often exhibit suboptimal or biased strategies that violate normative reward-maximization assumptions [57, 71], enlarging the solution space and introducing history-dependence and other idiosyncrasies; (ii) updates must integrate external stimuli—not just reward histories—going beyond classic bandit settings; and (iii) many prior approaches fix feature representations, use static policy forms, or rely on hand-designed rules that do not adapt during training. These factors call for flexible yet interpretable models with minimal structural priors; under these conditions, prior baselines underperform relative to ours (Table 1). Our

approach thus helps bridge structured neuroscience models with black-box meta-learning in this de novo inference setting.

6 Discussion

Understanding how animals learn from experience—particularly in de novo settings where input–output mappings must be acquired from scratch—remains a central challenge. Prior work has largely relied on predefined parametric models or focused on simpler settings like bandit tasks that do not require learning a new task structure from scratch. Here we instead infer, from behavior alone, the trial-by-trial learning rule used by the animal, confronting three challenges highlighted in Related Work: suboptimal or biased updates, the need to integrate external stimuli and multi-trial history, and the limits of fixed rule forms or static policy features. We demonstrate that our framework recovers ground-truth rules in simulation, extends to non-Markovian dynamics via recurrent architectures, improves held-out prediction accuracy on real behavioral data, and yields analyzable insights into animal learning. Together, these results support a data-driven framework for uncovering learning processes.

To our knowledge, this is the first study to infer the learning rule function in *de novo* learning settings without assuming a specific model class. While neural network–based approaches have been proposed in non-*de novo* settings such as bandit tasks [21–25], structured sensory inputs and the need to learn new input–output mappings introduce additional challenges. Prior work has also modeled behavior by having RNNs directly output cognitive states or choices, but such models often act as black boxes that are difficult to interpret [21, 22]. Although recent studies have sought to balance flexibility and interpretability by using RNNs to represent classical model parameters [52], we find that directly modeling parameter updates $\Delta \mathbf{w}$ leads to better ground truth reconstruction (Supp. Fig. 14). This advantage likely stems from the implicit regularization of neural networks, which biases learning dynamics toward smoother functions first in the feature learning regime [72–84]. That said, it remains unclear whether the inductive bias of our setup generalizes across various learning systems — an open question we highlight as a future direction (see below). One promising result supporting our approach is that, in a sense, implicitly regularizing weight updates aligns with the philosophy behind PsyTrack, which also acts at the level of weight changes [28].

Broader outlook for neuroscience and AI. By inferring learning rules directly from behavior without assuming a fixed model structure, our approach offers a data-driven framework for understanding how animals learn de novo. This has several implications. First, interpreting the inferred learning rules could offer insights into learning behavior (e.g., suboptimal or history-dependent patterns), which may point to additional variables—such as trial history, internal states, or reward timing—that are worth tracking in future behavioral learning studies. In turn, more accurate behavioral learning models could inform applications in health and conservation—e.g., anticipating how animals respond to environmental changes or interventions [4–12]. This also lays a foundation for more challenging settings with richer task structures and scaling to more expressive architectures. Second, de novo learning is ubiquitous in experimental neuroscience, where animals are often trained from scratch on novel tasks, yet remains understudied. By revealing structure in how animals learn during this phase, our method can inform experimental design and training protocols, supporting more efficient task training [19, 20].

From an AI perspective, our work contributes to the growing interest in biologically-aligned learning systems [13, 14]. By uncovering suboptimality and history dependence in animal learning, we reveal deviations from standard artificial agents that could inform the design of more human- or animal-aligned AI. Relatedly, our approach may support the development of behavioral digital twins—AI models that mimic individual learning processes for simulation, personalization, or real-time prediction [85]. Finally, by leveraging the implicit regularization of neural networks to infer learning rules, our work raises new questions about how such inductive biases interact with neural network–based system identification (see below).

Limitations and future directions. *Pooling across animals).* Neural network–based approaches are often data hungry, and reconstruction accuracy improves with more data (Fig. 2C). To address this, we pool across animals rather than fitting individual subjects. While pooling is common in neuroscience [86–88], it can obscure individual variability and inherit known limitations [89]. Nonetheless, learning rule inference from behavior alone is underconstrained in low-data regimes,

and pooling offers a practical mitigation strategy. This population-level approach aligns with broader trends in neuroscience and AI toward large-scale behavioral datasets.

Static vs. dynamic learning rules). Our current model assumes a static weight update function. While RNNs may implicitly capture temporal changes—perhaps explaining why fitting separate RNNGLMs to the first and second halves of training did not yield improvement—future work could explicitly model dynamic learning rules to better disentangle non-Markovian dependencies from nonstationarity. An important direction is to investigate whether the observed suboptimal negative baseline arises from specific cognitive mechanisms — such as choice perseverance or forgetting [90] — or from model mismatch. More broadly, suboptimal learning is common in the literature [91, 92], especially in pathological populations [93, 94], where applying our approach could be interesting.

Stochasticity). Our current learning model is deterministic. Capturing stochasticity in learning would be a valuable extension.

Generalization across datasets, tasks, and architectures). We evaluated our method using the same preprocessed IBL dataset as [19]. This focus on a relatively simple task was intentional: it mirrors widely used de novo learning paradigms in laboratory neuroscience, where animals are introduced to novel tasks with minimal prior structure. Such simplified tasks capture key hallmarks of biological learning—history dependence, biases, and suboptimal strategies—while providing a tractable platform for reverse-engineering learning dynamics. In this sense, our work should be viewed as a stepping stone toward broader generalization: establishing validity in a controlled setting before extending to richer tasks and datasets. Indeed, to demonstrate potential scalability, we tested higher-dimensional inputs and found that our approach continues to recover core qualitative features of learning rules, with performance improving as data availability increases (Supp. Table 6). These results indicate that our framework can extend beyond the simple tasks considered here.

Beyond extending to diverse tasks and datasets, another future avenue would be to investigate how different architectures—and their associated inductive biases—affect inference accuracy, particularly under data limitations. Our framework currently fixes the decision model (e.g., standard models such as GLMs), which aids interpretability and follows common neuroscience practice but limits expressivity. In fact, for multiplicative decision processes, this assumption leads to a significant performance degradation (Supp. Table 7). We view this modular approach—using a standard decision model while focusing on trial-by-trial weight updates—as a principled first step, consistent with strategies in neuroscience and machine learning where components are studied separately before joint inference. Extending the framework to jointly infer both the decision model and the learning rule is an important direction for future work.

Overall, we introduce a flexible framework for inferring learning rules directly from behavior, without assuming a predefined form. By capturing non-Markovian dynamics and uncovering insights into de novo learning, we hope this approach opens new avenues for studying how animals learn in rich, structured environments.

Acknowledgement

The authors thank the International Brain Laboratory for data availability, Orren Karniol-Tambour for feedback on the manuscript, and Alex Riordan for helpful discussions. YHL was partially supported by the Fonds de recherche du Québec – Nature et technologies (FRQNT). VG was supported the Porter Ogden Jacobus Fellowship at Princeton University, and by doctoral scholarships from the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Fonds de recherche du Québec – Nature et technologies (FRQNT). JWP was supported by grants from the Simons Collaboration on the Global Brain (SCGB AWD543027), the NIH BRAIN initiative (9R01DA056404-04), an NIH R01 (NIH 1R01EY033064), and a U19 NIH-NINDS BRAIN Initiative Award (U19NS104648).

References

[1] Ivan Pavlov. *Conditioned Reflexes*. Oxford University Press, 1927.

[2] Robert A. Rescorla and Allan R. Wagner. A theory of pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement. *Classical Conditioning II: Current Research and Theory*, page 64–99, 1972.

[3] Edoardo Fazzari, Donato Romano, Fabrizio Falchi, and Cesare Stefanini. Animal behavior analysis methods using deep learning: A survey. *arXiv preprint arXiv:2405.14002*, 2024.

[4] Sudarsini Tekkam Gnanasekar, Svetlana Yanushkevich, Nynke J Van den Hoogen, and Tuan Trang. Rodent tracking and abnormal behavior classification in live video using deep neural networks. In *2022 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 830–837. IEEE, 2022.

[5] Julie K Shaw and Sarah Lahrman. The human–animal bond–a brief look at its richness and complexities. *Canine and feline behavior for veterinary technicians and nurses*, pages 88–105, 2023.

[6] Gianluca Manduca, Valeria Zeni, Sara Moccia, Beatrice A Milano, Angelo Canale, Giovanni Benelli, Cesare Stefanini, and Donato Romano. Learning algorithms estimate pose and detect motor anomalies in flies exposed to minimal doses of a toxicant. *Iscience*, 26(12), 2023.

[7] Benjamin L Hart. Behavioural defences in animals against pathogens and parasites: parallels with the pillars of medicine in humans. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 366(1583):3406–3417, 2011.

[8] Dhanushi A Wijeyakulasuriya, Elizabeth W Eisenhauer, Benjamin A Shaby, and Ephraim M Hanks. Machine learning for modeling animal movement. *PloS one*, 15(7):e0235750, 2020.

[9] Jin Hou, Yuxin He, Hongbo Yang, Thomas Connor, Jie Gao, Yujun Wang, Yichao Zeng, Jindong Zhang, Jinyan Huang, Bochuan Zheng, et al. Identification of animal individuals using deep learning: A case study of giant panda. *Biological Conservation*, 242:108414, 2020.

[10] Ellen M Ditria, Sebastian Lopez-Marcano, Michael Sievers, Eric L Jinks, Christopher J Brown, and Rod M Connolly. Automating the analysis of fish abundance using object detection: optimizing animal ecology with deep learning. *Frontiers in Marine Science*, 7:429, 2020.

[11] Rudresh Pillai, Rupesh Gupta, Neha Sharma, and Rajesh Kumar Bansal. A deep learning approach for detection and classification of ten species of monkeys. In *2023 International Conference on Smart Systems for applications in Electrical Sciences (ICSSES)*, pages 1–6. IEEE, 2023.

[12] Simon RO Nilsson, Nastacia L Goodwin, Jia Jie Choong, Sophia Hwang, Hayden R Wright, Zane C Norville, Xiaoyu Tong, Dayu Lin, Brandon S Bentzley, Neir Eshel, et al. Simple behavioral analysis (simba)–an open source toolkit for computer classification of complex social behaviors in experimental animals. *BioRxiv*, pages 2020–04, 2020.

[13] Ilia Sucholutsky, Lukas Muttenthaler, Adrian Weller, Andi Peng, Andreea Bobu, Been Kim, Bradley C Love, Erin Grant, Iris Groen, Jascha Achterberg, et al. Getting aligned on representational alignment. *arXiv preprint arXiv:2310.13018*, 2023.

[14] Anthony Zador, Sean Escola, Blake Richards, Bence Ölveczky, Yoshua Bengio, Kwabena Boahen, Matthew Botvinick, Dmitri Chklovskii, Anne Churchland, Claudia Clopath, et al. Catalyzing next-generation artificial intelligence through neuroai. *Nature communications*, 14(1):1597, 2023.

[15] Peter Dayan and Nathaniel D Daw. Decision theory, reinforcement learning, and the brain. *Cognitive, Affective, & Behavioral Neuroscience*, 8(4):429–453, 2008.

[16] Yael Niv. Reinforcement learning in the brain. *Journal of Mathematical Psychology*, 53(3):139–154, 2009.

[17] Richard S Sutton and Andrew G Barto. Time-derivative models of pavlovian reinforcement. *Learning and computational neuroscience: Foundations of adaptive networks*, 1988.

[18] Armin Lak, Emily Hueske, Junya Hirokawa, Paul Masset, Torben Ott, Anne E Urai, Tobias H Donner, Matteo Carandini, Susumu Tonegawa, Naoshige Uchida, et al. Reinforcement biases subsequent perceptual decisions when confidence is low, a widespread behavioral phenomenon. *ELife*, 9:e49834, 2020.

[19] Zoe Ashwood, Nicholas A Roy, Ji Hyun Bak, and Jonathan W Pillow. Inferring learning rules from animal decision-making. *Advances in Neural Information Processing Systems*, 33:3442–3453, 2020.

[20] Victor Geadah and Jonathan W Pillow. Inferring learning rules during de novo task learning. *bioRxiv*, pages 2025–09, 2025.

[21] Li Ji-An, Marcus K Benna, and Marcelo G Mattar. Automatic discovery of cognitive strategies with tiny recurrent neural networks. *bioRxiv*, pages 2023–04, 2023.

[22] Kevin Miller, Maria Eckstein, Matt Botvinick, and Zeb Kurth-Nelson. Cognitive model discovery via disentangled rnns. *Advances in Neural Information Processing Systems*, 36:61377–61394, 2023.

[23] Maria K Eckstein, Christopher Summerfield, Nathaniel D Daw, and Kevin J Miller. Predictive and interpretable: Combining artificial neural networks and classic cognitive models to understand human learning and decision making. *BioRxiv*, pages 2023–05, 2023.

[24] Amir Dezfouli, Hassan Ashtiani, Omar Ghattas, Richard Nock, Peter Dayan, and Cheng Soon Ong. Disentangled behavioural representations. *Advances in neural information processing systems*, 32, 2019.

[25] Pablo Samuel Castro, Nenad Tomasev, Ankit Anand, Navodita Sharma, Rishika Mohanta, Aparna Dev, Kuba Perlin, Siddhant Jain, Kyle Levin, Noémi Éltető, et al. Discovering symbolic cognitive models from human and animal behavior. *bioRxiv*, pages 2025–02, 2025.

[26] Joanna C Chang, Matthew G Perich, Lee E Miller, Juan A Gallego, and Claudia Clopath. De novo motor learning creates structure in neural activity space that shapes adaptation. *bioRxiv*, 2023.

[27] Davin Greenwell, Samia Vanderkolff, and Jacob Feigh. Understanding de novo learning for brain-machine interfaces. *Journal of Neurophysiology*, 129(4):749–750, 2023.

[28] Nicholas A Roy, Ji Hyun Bak, Athena Akrami, Carlos Brody, and Jonathan W Pillow. Efficient inference for time-varying behavior during learning. *Advances in neural information processing systems*, 31, 2018.

[29] Sebastian A. Bruijns, Kcénia Bougrova, Inês C. Laranjeira, Petrina Y. P. Lau, Guido T. Meijer, Nathaniel J. Miska, Jean-Paul Noel, Alejandro Pan-Vazquez, Noam Roth, Karolina Z. Socha, Anne E. Urai, and Peter Dayan. Dissecting the complexities of learning with infinite hidden markov models. bioRxiv 2023.12.22.573001, December 2023.

[30] Nicholas A Roy, Ji Hyun Bak, Athena Akrami, Carlos D Brody, and Jonathan W Pillow. Extracting the dynamics of behavior in sensory decision-making experiments. *Neuron*, 109(4):597–610, 2021.

[31] International Brain Laboratory. A standardized and reproducible method to measure decision-making in mice. *BioRxiv*, page 909838, 2020.

[32] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[33] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8:229–256, 1992.

[34] Samuel Liebana Garcia, Aeron Laffere, Chiara Toschi, Louisa Schilling, Jacek Podlaski, Matthias Fritsche, Peter Zatka-Haas, Yulong Li, Rafal Bogacz, Andrew Saxe, et al. Striatal dopamine reflects individual long-term learning trajectories. *BioRxiv*, pages 2023–12, 2023.

[35] Blake A Richards, Timothy P Lillicrap, Philippe Beaudoin, Yoshua Bengio, Rafal Bogacz, Amelia Christensen, Claudia Clopath, Rui Ponte Costa, Archy de Berker, Surya Ganguli, et al. A deep learning framework for neuroscience. *Nature neuroscience*, 22(11):1761–1770, 2019.

[36] Guangyu Robert Yang and Xiao-Jing Wang. Artificial neural networks for neuroscientists: a primer. *Neuron*, 107(6):1048–1070, 2020.

[37] Daniel LK Yamins, Ha Hong, Charles F Cadieu, Ethan A Solomon, Darren Seibert, and James J DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the national academy of sciences*, 111(23):8619–8624, 2014.

[38] Andrea Banino, Caswell Barry, Benigno Uria, Charles Blundell, Timothy Lillicrap, Piotr Mirowski, Alexander Pritzel, Martin J Chadwick, Thomas Degris, Joseph Modayil, et al. Vector-based navigation using grid-like representations in artificial agents. *Nature*, 557(7705):429–433, 2018.

[39] Christopher J Cueva and Xue-Xin Wei. Emergence of grid-like representations by training recurrent neural networks to perform spatial localization. *arXiv preprint arXiv:1803.07770*, 2018.

[40] James CR Whittington, Timothy H Muller, Shirley Mark, Guifen Chen, Caswell Barry, Neil Burgess, and Timothy EJ Behrens. The tolman-eichenbaum machine: unifying space and relational memory through generalization in the hippocampal formation. *Cell*, 183(5):1249–1263, 2020.

[41] Danil Tyulmankov, Guangyu Robert Yang, and LF Abbott. Meta-learning synaptic plasticity and memory addressing for continual familiarity detection. *Neuron*, 110(3):544–557, 2022.

[42] Valerio Mante, David Sussillo, Krishna V Shenoy, and William T Newsome. Context-dependent computation by recurrent dynamics in prefrontal cortex. *nature*, 503(7474):78–84, 2013.

[43] Marino Pagan, Vincent D Tang, Mikio C Aoi, Jonathan W Pillow, Valerio Mante, David Sussillo, and Carlos D Brody. Individual variability of neural computations underlying flexible decisions. *Nature*, 639(8054):421–429, 2025.

[44] Jane X Wang, Zeb Kurth-Nelson, Dharshan Kumaran, Dhruva Tirumala, Hubert Soyer, Joel Z Leibo, Demis Hassabis, and Matthew Botvinick. Prefrontal cortex as a meta-reinforcement learning system. *Nature neuroscience*, 21(6):860–868, 2018.

[45] Valeria Fascianelli, Fabio Stefanini, Satoshi Tsujimoto, Aldo Genovesio, and Stefano Fusi. Neural representational geometry correlates with behavioral differences between monkeys. *bioRxiv*, pages 2022–10, 2022.

[46] Kristopher T Jensen, Guillaume Hennequin, and Marcelo G Mattar. A recurrent network model of planning explains hippocampal replay and human behavior. *Nature neuroscience*, 27(7):1340–1348, 2024.

[47] Joshua C Peterson, David D Bourgin, Mayank Agrawal, Daniel Reichman, and Thomas L Griffiths. Using large-scale experiments and machine learning to discover theories of human decision-making. *Science*, 372(6547):1209–1214, 2021.

[48] Mingyu Song, Yael Niv, and Mingbo Cai. Using recurrent neural networks to understand human reward learning. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 43, 2021.

[49] Paul I Jaffe, Russell A Poldrack, Robert J Schafer, and Patrick G Bissett. Modelling human behaviour in cognitive tasks with latent dynamical systems. *Nature Human Behaviour*, 7(6):986–1000, 2023.

[50] Ionatan Kuperwajs, Heiko H Schütt, and Wei Ji Ma. Using deep neural networks as a guide for modeling human planning. *Scientific reports*, 13(1):20269, 2023.

[51] Milena Rmus, Ti-Fen Pan, Liyu Xia, and Anne GE Collins. Artificial neural networks for model identification and parameter estimation in computational cognitive models. *PLOS Computational Biology*, 20(5):e1012119, 2024.

[52] Yoav Ger, Eliya Nachmani, Lior Wolf, and Nitzan Shahar. Harnessing the flexibility of neural networks to predict dynamic theoretical parameters underlying human choice behavior. *PLoS Computational Biology*, 20(1):e1011678, 2024.

[53] Jeffrey C. Magee and Christine Grienberger. Synaptic Plasticity Forms and Functions. *Annual Review of Neuroscience*, 43(1):95–117, jul 2020.

[54] Wulfram Gerstner, Marco Lehmann, Vasiliki Liakoni, Dane Corneil, and Johanni Brea. Eligibility Traces and Plasticity on Behavioral Time Scales: Experimental Support of NeoHebbian Three-Factor Learning Rules. *Frontiers in Neural Circuits*, 12:53, jul 2018.

[55] Anne E Urai, Anke Braun, and Tobias H Donner. Pupil-linked arousal is driven by decision uncertainty and alters serial choice bias. *Nature communications*, 8(1):14637, 2017.

[56] Ari S Morcos and Christopher D Harvey. History-dependent variability in population dynamics during evidence accumulation in cortex. *Nature neuroscience*, 19(12):1672–1681, 2016.

[57] Athena Akrami, Charles D Kopec, Mathew E Diamond, and Carlos D Brody. Posterior parietal cortex represents sensory history and mediates its effects on behaviour. *Nature*, 554(7692):368–372, 2018.

[58] Sukbin Lim, Jillian L McKee, Luke Woloszyn, Yali Amit, David J Freedman, David L Sheinberg, and Nicolas Brunel. Inferring learning rules from distributions of firing rates in cortical neurons. *Nature neuroscience*, 18(12):1804–1810, 2015.

[59] Jacob Portes, Christian Schmid, and James M Murray. Distinguishing learning rules with brain machine interfaces. *Advances in neural information processing systems*, 35:25937–25950, 2022.

[60] Daniel R Kepple, Rainer Engelken, and Kanaka Rajan. Curriculum learning as a tool to uncover learning principles in the brain. In *International Conference on Learning Representations*, 2021.

[61] Aran Nayebi, Sanjana Srivastava, Surya Ganguli, and Daniel L Yamins. Identifying learning rules from neural network observables. *Advances in Neural Information Processing Systems*, 33:2639–2650, 2020.

[62] Yash Mehta, Danil Tyulmankov, Adithya Rajagopalan, Glenn Turner, James Fitzgerald, and Jan Funke. Model based inference of synaptic plasticity rules. *Advances in Neural Information Processing Systems*, 37:48519–48540, 2024.

[63] Basile Confavreux, Friedemann Zenke, Everton Agnes, Timothy Lillicrap, and Tim Vogels. A meta-learning approach to (re) discover plasticity rules that carve a desired function into a neural network. *Advances in Neural Information Processing Systems*, 33:16398–16408, 2020.

[64] Basile Confavreux, Poornima Ramesh, Pedro J Goncalves, Jakob H Macke, and Tim Vogels. Meta-learning families of plasticity rules in recurrent spiking networks using simulation-based inference. *Advances in Neural Information Processing Systems*, 36:13545–13558, 2023.

[65] Poornima Ramesh, Basile Confavreux, Pedro J Goncalves, Tim P Vogels, and Jakob H Macke. Indistinguishable network dynamics can emerge from unalike plasticity rules. *bioRxiv*, pages 2023–11, 2023.

[66] David Bell, Alison Duffy, and Adrienne Fairhall. Discovering plasticity rules that organize and maintain neural circuits. *Advances in Neural Information Processing Systems*, 37:40732–40751, 2024.

[67] Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando De Freitas. Learning to learn by gradient descent by gradient descent. *Advances in neural information processing systems*, 29, 2016.

[68] Yan Duan, John Schulman, Xi Chen, Peter L Bartlett, Ilya Sutskever, and Pieter Abbeel. Rl2: Fast reinforcement learning via slow reinforcement learning. 2016. *URL http://arxiv.org/abs/1611.02779*, 2016.

[69] Thomas Miconi, Kenneth Stanley, and Jeff Clune. Differentiable plasticity: training plastic neural networks with backpropagation. In *International Conference on Machine Learning*, pages 3559–3568. PMLR, 2018.

[70] Navid Shervani-Tabar and Robert Rosenbaum. Meta-learning biologically plausible plasticity rules with random feedback pathways. *Nature Communications*, 14(1):1805, 2023.

[71] David B Kastner, Eric A Miller, Zhuonan Yang, Demetris K Roumis, Daniel F Liu, Loren M Frank, and Peter Dayan. Spatial preferences account for inter-animal variability during the continual learning of a dynamic cognitive task. *Cell reports*, 39(3), 2022.

[72] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *Advances in neural information processing systems*, 31, 2018.

[73] Lenaic Chizat, Edouard Oyallon, and Francis Bach. On lazy training in differentiable programming. *Advances in neural information processing systems*, 32, 2019.

[74] Timo Flesch, Keno Juechems, Tsvetomira Dumbalska, Andrew Saxe, and Christopher Summerfield. Rich and lazy learning of task representations in brains and neural networks. *BioRxiv*, pages 2021–04, 2021.

[75] Thomas George, Guillaume Lajoie, and Aristide Baratin. Lazy vs hasty: linearization in deep networks impacts learning schedule based on example difficulty. *TMLR*, 2022.

[76] Lukas Braun, Clémentine Dominé, James Fitzgerald, and Andrew Saxe. Exact learning dynamics of deep linear networks with prior knowledge. *Advances in Neural Information Processing Systems*, 35:6615–6629, 2022.

[77] Mohammad Pezeshki, Oumar Kaba, Yoshua Bengio, Aaron C Courville, Doina Precup, and Guillaume Lajoie. Gradient starvation: A learning proclivity in neural networks. *Advances in Neural Information Processing Systems*, 34, 2021.

[78] Aristide Baratin, Thomas George, César Laurent, R Devon Hjelm, Guillaume Lajoie, Pascal Vincent, and Simon Lacoste-Julien. Implicit regularization via neural feature alignment. In *International Conference on Artificial Intelligence and Statistics*, pages 2269–2277. PMLR, 2021.

[79] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *International conference on machine learning*, pages 5301–5310. PMLR, 2019.

[80] Yuan Cao, Zhiying Fang, Yue Wu, Ding-Xuan Zhou, and Quanquan Gu. Towards understanding the spectral bias of deep learning. *arXiv preprint arXiv:1912.01198*, 2019.

[81] Alexander Atanasov, Blake Bordelon, and Cengiz Pehlevan. Neural networks as kernel learners: The silent alignment effect. *arXiv preprint arXiv:2111.00034*, 2021.

[82] Abdulkadir Canatar, Blake Bordelon, and Cengiz Pehlevan. Spectral bias and task-model alignment explain generalization in kernel regression and infinitely wide neural networks. *Nature communications*, 12(1):2914, 2021.

[83] Melikasadat Emami, Mojtaba Sahraee-Ardakan, Parthe Pandit, Sundeep Rangan, and Alyson K Fletcher. Implicit bias of linear rnns. In *International Conference on Machine Learning*, pages 2982–2992. PMLR, 2021.

[84] Andrew Gordon Wilson. Deep learning is not so mysterious or different. *arXiv preprint arXiv:2503.02113*, 2025.

[85] Evangelia Katsoulakis, Qi Wang, Huanmei Wu, Leili Shahriyari, Richard Fletcher, Jinwei Liu, Luke Achenie, Hongfang Liu, Pamela Jackson, Ying Xiao, et al. Digital twins for health: a scoping review. *NPJ digital medicine*, 7(1):77, 2024.

[86] Zoe Ashwood, Aditi Jha, and Jonathan W Pillow. Dynamic inverse reinforcement learning for characterizing animal behavior. *Advances in neural information processing systems*, 35:29663–29676, 2022.

[87] Zoe C Ashwood, Nicholas A Roy, Iris R Stone, International Brain Laboratory, Anne E Urai, Anne K Churchland, Alexandre Pouget, and Jonathan W Pillow. Mice alternate between discrete strategies during perceptual decision-making. *Nature Neuroscience*, 25(2):201–212, 2022.

[88] Scott S Bolkan, Iris R Stone, Lucas Pinto, Zoe C Ashwood, Jorge M Iravedra Garcia, Alison L Herman, Priyanka Singh, Akhil Bandi, Julia Cox, Christopher A Zimmerman, et al. Opponent control of behavior by dorsomedial striatal pathways depends on task demands and internal state. *Nature neuroscience*, 25(3):345–357, 2022.

[89] Sharlen Moore and Kishore V Kuchibhotla. Slow or sudden: Re-interpreting the learning curve for modern systems neuroscience. *IBRO Neuroscience Reports*, 13:9–14, 2022.

[90] Ronald L Davis and Yi Zhong. The biology of forgetting—a perspective. *Neuron*, 95(3):490–503, 2017.

[91] Manuel Molano-Mazón, Yuxiu Shao, Daniel Duque, Guangyu Robert Yang, Srdjan Ostojic, and Jaime De La Rocha. Recurrent networks endowed with structural priors explain suboptimal animal behavior. *Current Biology*, 33(4):622–638, 2023.

[92] Tzuhsuan Ma and Ann M Hermundstad. A vast space of compact strategies for effective decisions. *Science Advances*, 10(25):eadj4064, 2024.

[93] Amir Dezfouli, Kristi Griffiths, Fabio Ramos, Peter Dayan, and Bernard W Balleine. Models that learn how humans learn: The case of decision-making and its disorders. *PLoS computational biology*, 15(6):e1006903, 2019.

[94] M Ganesh Kumar, Adam Manoogian, Billy Qian, Cengiz Pehlevan, and Shawn A Rhoads. Neurocomputational underpinnings of suboptimal beliefs in recurrent neural network-based agents. *bioRxiv*, pages 2025–03, 2025.

[95] George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.

[96] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural networks*, 4(2):251–257, 1991.

[97] Allan Pinkus. Approximation theory of the mlp model in neural networks. *Acta numerica*, 8:143–195, 1999.

[98] MH Hassoun. *Fundamentals of Artificial Neural Networks*. The MIT Press, 1995.

[99] Simon Haykin. *Neural networks: a comprehensive foundation*. Prentice Hall PTR, 1998.

[100] Zhou Lu, Hongming Pu, Feicheng Wang, Zhiqiang Hu, and Liwei Wang. The expressive power of neural networks: A view from the width. *Advances in neural information processing systems*, 30, 2017.

[101] Ken-ichi Funahashi and Yuichi Nakamura. Approximation of dynamical systems by continuous time recurrent neural networks. *Neural networks*, 6(6):801–806, 1993.

[102] Aad W Van der Vaart. *Asymptotic statistics*, volume 3. Cambridge university press, 2000.

[103] Rob J Hyndman and George Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2018.

[104] Mario Geiger, Stefano Spigler, Arthur Jacot, and Matthieu Wyart. Disentangling feature and lazy training in deep neural networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2020(11):113301, 2020.

[105] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[106] Pauli Virtanen, Ralf Gommers, Travis E Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, et al. Scipy 1.0: fundamental algorithms for scientific computing in python. *Nature methods*, 17(3):261–272, 2020.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: To make this easier for the readers, we have referred to the pertinent figures and sections under "Main contributions" in Introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Details on limitations and future work are discussed in the 'Limitations and future works' subsection in Discussion.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: We provide detailed assumption and proof for Proposition 1 in Supp. A.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Training details are provided in Supp. C. Moreover, our code is available `https://github.com/Helena-Yuhan-Liu/InferLearningANNGLM`

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We use IBL datasets, which are publicly available. Our code is available `https://github.com/Helena-Yuhan-Liu/InferLearningANNGLM`.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Details are provided in Supp. C.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We have included this information in all applicable figures and tables. In particular, p-values for log-likelihood comparisons are provided in Table 1, with details of the t-test and cross-validation procedure described in Supp. C.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Information pertaining to computing resources and simulation time can be found in Supp. C.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have carefully read the NeurIPS Code of Ethics and attest that the research conforms.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: This research advances our understanding of animal learning mechanisms and contributes to the development of interpretable models for behavior. We do not anticipate any immediate ethical or societal impacts. However, over time, these findings could influence related fields such as neuroscience and machine learning, which may indirectly affect society depending on how these technologies are applied.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: No specific safeguards were implemented, since this work is focused on basic research to better understand animal learning, as explained above. The model is not intended for deployment or real-world decision-making. As such, no foreseeable misuse or ethical risks require mitigation at this stage.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Please see Supp. C.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: LLMs were used solely for writing, editing, and formatting assistance in this work. We also briefly used Deep Research to assess the coverage of our related work discussion and found that our initial literature review was already comprehensive.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

A Theoretical justifications

To gain theoretical insights into why this approach could work, we consider two key questions: (1) Do neural networks have the capacity to approximate the learning rule, i.e., the weight update function? (2) Since weight updates are not directly observable, can they be identified from behavioral data? Question (1), assuming well-behaved weight update functions, is addressed by universal approximation theorems [95–101]. For question (2), we start by considering a very simple idealized setting in the proposition below and leave comprehensive theoretical investigations into identifiability for future work.

Proposition 1. *For simplicity, let* $\mathbf{w}_t \in \mathbb{R}$ *for* $t \in \{0, 1\}$*, and assume* $y_t \sim \text{Bernoulli}(P_{y_t})$*, where*

$$P_{y_t} := \mathbb{P}[y_t = 1 \mid \mathbf{w}_t, x_t] = \sigma(\mathbf{w}_t x_t), \quad \text{with } \sigma(z) = \frac{1}{1 + e^{-z}} \text{ denoting the sigmoid function.}$$

We further make the following simplifying assumptions: (1) $\Delta \mathbf{w}_t = \mathbf{w}_{t+1} - \mathbf{w}_t$ *depends only on* $(\mathbf{w}_t, x_t, y_t)$*. (2) For each* t*, inputs* $x_t^{(i)} \in \{-1, 1\}$ *are drawn i.i.d. across samples from the uniform distribution over this set, i.e.,* $\mathbb{P}(x_t = -1) = \mathbb{P}(x_t = 1) = \frac{1}{2}$*. (3) The same initial weight* $\mathbf{w}_0$ *is used across* N *independent repetitions. (4) For all sample* i*, we assume* $P_{y_t}^{(i)} \in [\epsilon, 1 - \epsilon]$ *for some arbitrary* $0 < \epsilon < 1$*. (5) The prior on* $\mathbf{w}_t$ *is smooth, strictly positive, and does not concentrate near the boundary of the parameter space. Then, as* N *approaches infinity, the standard error (SE) of* $\Delta \mathbf{w}_0 = w_1 - w_0$ *across samples approaches* 0*.*

Proof. The log-likelihood for a single observation at time t is:

$$\ell_t(\mathbf{w}_t) = y_t \log P_{y_t} + (1 - y_t) \log(1 - P_{y_t}).$$

By taking the second derivative, we arrive at:

$$\begin{aligned} \frac{\partial^2 \ell_t(\mathbf{w}_t)}{\partial \mathbf{w}_t^2} &= -P_{y_t}(1 - P_{y_t}) x_t^2 \\ &= -P_{y_t}(1 - P_{y_t}), \end{aligned}$$

where x_t^2 is dropped due to the assumption that $x \in \{-1, 1\}$.

Summing over N independent observations, the Fisher information for $\mathbf{w}_t$ is:

$$I(\mathbf{w}_t) = \sum_{i=1}^{N} P_{y_t}^{(i)}(1 - P_{y_t}^{(i)}).$$

By the Bernstein-von Mises (BvM) theorem (see [102]), the posterior distribution of $\mathbf{w}_t$ given the data is asymptotically normal:

$$\pi(\mathbf{w}_t \mid \text{data}) \xrightarrow{d} \mathcal{N}\left(\hat{w}_t, \frac{1}{I(\mathbf{w}_t)}\right),$$

where $\hat{w}_t$ is the MLE of $\mathbf{w}_t$.

Thus, by BvM theorem, in the limit as $N \to \infty$, the posterior variance is asymptotically equal to:

$$\text{Var}(\mathbf{w}_t) = \frac{1}{\sum_{i=1}^{N} P_{y_t}^{(i)}(1 - P_{y_t}^{(i)})}.$$

and applying assumption (4):

$$\text{Var}(\mathbf{w}_t) \leq \frac{1}{N\epsilon(1 - \epsilon)},$$

and

$$\text{SE}(\mathbf{w}_t) \leq \sqrt{\frac{1}{N\epsilon(1 - \epsilon)}}.$$

Since $\Delta \mathbf{w}_0 = w_1 - w_0$, the variance of $\Delta \mathbf{w}_0$ can be expressed as:

$$\text{Var}(\Delta \mathbf{w}_0) = \text{Var}(w_1) + \text{Var}(w_0) - 2 \cdot \text{Cov}(w_1, w_0).$$

Using the Cauchy-Schwarz inequality, the covariance term is bounded:

$$|\text{Cov}(w_1, w_0)| \leq \sqrt{\text{Var}(w_1) \cdot \text{Var}(w_0)}.$$

Thus, we have:

$$\text{Var}(\Delta \mathbf{w}_0) \leq \text{Var}(w_1) + \text{Var}(w_0) + 2\sqrt{\text{Var}(w_1) \cdot \text{Var}(w_0)}.$$

Substituting the variance bounds:

$$\text{Var}(\Delta \mathbf{w}_0) \leq \frac{1}{N\epsilon(1-\epsilon)} + \frac{1}{N\epsilon(1-\epsilon)} + 2\sqrt{\frac{1}{N\epsilon(1-\epsilon)}}.$$

This simplifies to:

$$\text{Var}(\Delta \mathbf{w}_0) \leq \left(\sqrt{\frac{1}{N\epsilon(1-\epsilon)}} + \sqrt{\frac{1}{N\epsilon(1-\epsilon)}} \right)^2$$

Taking the square root to get the SE:

$$\text{SE}(\Delta \mathbf{w}_0) \leq \sqrt{\frac{1}{N\epsilon(1-\epsilon)}} + \sqrt{\frac{1}{N\epsilon(1-\epsilon)}}.$$

As $N \to \infty$, both terms approach 0, so $\text{SE}(\Delta \mathbf{w}_0) \to 0$.

□

Despite the restrictive assumptions, this proposition demonstrates (1) identifiability in highly simple settings to motivate future theoretical work and (2) identifies factors that contribute to the Fisher information. These factors include the number of animals pooled, N (see Fig. 2C) and initial weights, $\mathbf{w}_0$ (see Supp. Table 3). We also note that the third assumption could be approximately achieved by pretraining animals to a common performance level or by pooling animals that exhibit similar initial behavior.

B Learning rules considered for simulated data

In simulated data, we model weight updates as follows:

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \Delta\mathbf{w}_t, \tag{9}$$

where $\Delta\mathbf{w}_t$ is determined by a specific ground truth learning rule. We remind readers that variables were defined in Section 2.

For **Markovian learning rules**, we consider the classical REINFORCE [33] (more details in Supp. B.1):

$$\begin{aligned}\Delta\mathbf{w_t} &\propto r(y_t, z_t)\nabla_{\mathbf{w}} \log p(y_t \mid \mathbf{x}_t, \mathbf{w_t}) \\ &= r(y_t, z_t)\epsilon_{y_t}(1 - p_{y_t})x_t, \quad \epsilon_{y_t=R} = +1, \quad \epsilon_{y_t=L} = -1,\end{aligned} \tag{10}$$

where $p_{y_t} := p(y_t \mid \mathbf{x}_t, \mathbf{w_t})$.

Results for simulated data using the classical REINFORCE rule were shown in Fig. 2. A key property of this rule—captured well by our method—is the clear difference in weight updates following correct versus incorrect choices. To further test our approach, we considered an alternative rule that lacks this characteristic: a maximum likelihood rule (Supp. Fig. 6). This can be viewed as analogous to supervised learning, where the correct label is known and the weight update aims to increase the probability of choosing the correct action:

$$\begin{aligned}\Delta\mathbf{w_t} &\propto \nabla_{\mathbf{w_t}} \log p(z_t \mid \mathbf{x}_t, \mathbf{w}) \\ &= \epsilon_{z_t}(1 - p_{z_t})x_t, \quad \epsilon_{z_t=R} = +1, \quad \epsilon_{z_t=L} = -1,\end{aligned} \tag{11}$$

which doesn't depend on the action and reward.

For **non-Markovian learning rule** in Fig. 3, we consider a modified REINFORCE with an "elibility-trace-like" factor (see Supp. B.2):

$$\Delta\mathbf{w_t} \propto r(y_t, z_t) \sum_s \epsilon_{y_{t-s}}(1 - p_{y_{t-s}})x_{t-s}. \tag{12}$$

B.1 REINFORCE

We present the classical REINFORCE [33]:

$$\Delta\mathbf{w}_t \propto r_t(y_t, z_t)\nabla_{\mathbf{w}_t} log P(y_t|x_t, \mathbf{w}_t), \tag{13}$$

which is derived as follows.

We want to maximize the expected reward (expectation across the agent's stochastic behavior) by computing its gradient:

$$\begin{aligned}\nabla_{\mathbf{w}_t}\mathbb{E}_{P(y_t|x_t,\mathbf{w}_t)}[r(y_t, z_t)] &= \nabla_{\mathbf{w}_t} \int r(y_t, z_t)P(y_t|x_t, \mathbf{w}_t)dy_t \\ &\overset{(a)}{=} \int r(y_t, z_t)\nabla_{\mathbf{w}_t} P(y_t|x_t, \mathbf{w}_t)dy_t \\ &\overset{(b)}{=} \int r(y_t, z_t)P(y_t|x_t, \mathbf{w}_t)\nabla_{\mathbf{w}_t} log P(y_t|x_t, \mathbf{w}_t)dy_t \\ &= \mathbb{E}_{P(y_t|x_t,\mathbf{w}_t)}[r(y_t, z_t)\nabla_{\mathbf{w}_t} log P(y_t|x_t, \mathbf{w}_t)],\end{aligned} \tag{14}$$

where (a) is because $r(y_t, z_t)$ is independent of $\mathbf{w}_t$ once y_t is given, and (b) is from the log-derivative trick. We can evaluate the right-hand-side with a Monte Carlo integral by assuming the animal's choice y_t is sampled according to its policy $P(y_t|x_t, \mathbf{w}_t)$, which results in Eq. 13.

B.2 REINFORCE with "eligibility trace"

Suppose we change the task setting and the reward r_t now depends up to past S data points, i.e. $r_t = r_t(y_{t-S:t}, z_{t-S:t})$, then:

$$
\begin{aligned}
&\nabla_{\mathbf{w}_t} \mathbb{E}_{P(y_{t-S:t}|x_{t-S:t},\mathbf{w}_{t-S:t})}[r(y_{t-S:t}, z_{t-S:t})] \\
&= \nabla_{\mathbf{w}_t} \int r(y_{t-S:t}, z_{t-S:t}) P(y_{t-S:t}|x_{t-S:t}, \mathbf{w}_{t-S:t}) dy_{t-S:t} \\
&\overset{(a)}{=} \int r(y_{t-S:t}, z_{t-S:t}) \nabla_{\mathbf{w}_t} P(y_{t-S:t}|x_{t-S:t}, \mathbf{w}_{t-S:t}) dy_{t-S:t} \\
&\overset{(b)}{=} \int r(y_{t-S:t}, z_{t-S:t}) P(y_{t-S:t}|x_{t-S:t}, \mathbf{w}_{t-S:t}) \nabla_{\mathbf{w}_t} logP(y_{t-S:t}|x_{t-S:t}, \mathbf{w}_{t-S:t}) dy_{t-S:t} \\
&= \mathbb{E}_{P(y_{t-S:t}|x_{t-S:t},\mathbf{w}_{t-S:t})}[r(y_{t-S:t}, z_{t-S:t}) \nabla_{\mathbf{w}_t} logP(y_{t-S:t}|x_{t-S:t}, \mathbf{w}_{t-S:t})], \\
&\overset{(c)}{\approx} \mathbb{E}_{P(y_{t-S:t}|x_{t-S:t},\mathbf{w}_{t-S:t})}[r(y_{t-S:t}, z_{t-S:t}) \sum_{s=0}^{S} \nabla_{\mathbf{w}_{t-s}} logP(y_{t-s}|x_{t-s}, \mathbf{w}_{t-s})],
\end{aligned} \tag{15}
$$

where (a) is because $r(y_{t-S:t}, z_{t-S:t})$ is independent of $\mathbf{w}$ once $y_{t-S:t}$ is given, and (b) is from the log-derivative trick; (c) we assume $\mathbf{w}_{t-s} \approx \mathbf{w}_t$ and also $P(y_{t-S:t}|x_{t-S:t}, \mathbf{w}_{t-S:t}) = \Pi_{s=0}^{S} P(y_{t-s}|x_{t-s}, \mathbf{w}_{t-s})$ from the task setup. Again, we apply Monte-Carlo approximation:

$$
\Delta \mathbf{w}_t \propto r_t \sum_{s=0}^{S} \nabla_{\mathbf{w}_{t-s}} logP(y_{t-s}|x_{t-s}, \mathbf{w}_{t-s}). \tag{16}
$$

For Fig. 3, we used $S = 10$ and defined $r_t = r_t(y_{t-S:t}, z_{t-S:t})$ as 1 if at least half of the recent S trials are correct and 0 otherwise; we found similar results for other choices too. We refer to it as 'eligibility-trace-like', although it may not arise from eligibility traces, since the reward is combined with an input/output-dependent factor that accumulates over multiple time steps.

C Additional details on model and data

Simulated data. We simulate a pool of animals, each following the same underlying learning rule. In Supp. Fig. 11, we also explore populations with mixed weight update functions, where pooling tends to recover an in-between weight update function. Even when animals follow the same rule, their learning trajectories differ due to the stochasticity of received inputs and behavior.

For each simulated animal, the learning process unfolds over trials $t = 0, 1, \ldots, T$ as follows. Each stimulus s_t is drawn uniformly from discrete values in $[-2, 2]$ with 0.25 increments, excluding 0 (except for Fig. 2, which includes 0); varying the increment size did not affect our conclusions. The correct label or answer is defined as $z_t = \mathbb{1}[s_t > 0]$. Decision weights are updated according to $\mathbf{w}_{t+1} = \mathbf{w}_t + \Delta\mathbf{w}_t$, where $\Delta\mathbf{w}_t$ is computed from the ground-truth learning rule being tested (see Supp. B). The agent's binary choice $y_t \in \{0, 1\}$ is sampled from a Bernoulli GLM parameterized by $\mathbf{w}_t$, as described in Section 2. A reward is given if the choice matches the correct answer: $r_t = \mathbb{1}[y_t = z_t]$.

For the total number of trials T, we use $T = 500$ by default (to speed up simulations), but also demonstrate that our conclusions hold for longer T ($T = 8000$) in Supp. Fig. 10. We set the learning rate α such that the animal's performance reaches a saturation regime — typically when the stimulus weight approaches a value of 3 for this task. The GLM includes a stimulus weight that is updated over trials, and a fixed bias weight sampled randomly at the start of each simulation from $\{-1, 0, 1\}$. We simulate a pool of 10 to 1000 animals to observe the influence of data availability on reconstruction accuracy in Fig. 2C. We evaluate model performance using 5-fold cross-validation, where each fold holds out a subset of animals entirely. This ensures generalization across individuals, consistent with the underlying assumption of pooling — that animals share commonalities which can be learned from a subset and applied to held-out animals.

IBL data. We use the publicly available International Brain Laboratory (IBL) dataset [31], and follow the same preprocessing pipeline as [19], with details available in their accompanying code. For dataset details and learning curves, refer to Supp. C and Fig. S1 in [19]. In other words, we used the same preprocessed dataset as [19] ($N = 12$), for benchmarking purposes. For evaluation, we adopt two cross-validation strategies, and both led to similar trends (Tables 1 and 2). First, under the pooling assumption that a shared learning model generalizes across animals, we use K-fold cross-validation with animal-held-out splits. We fit the model using the first 10000 trials per animal to capture the full learning trajectory, including the saturation phase. For each fold, we report the average validation log-likelihood across four random seeds and conduct paired t-tests across matched animals to obtain the p-values reported in Table 1. The paired design accounts for variability in individual animals' performance: animals who learn well are easier to predict, while animals who have not learned the task are harder to model (see Fig. S1 in [19]). Second, to assess temporal generalization, we adopt a future-data holdout strategy [103]. We train up to the point where the learning curve inflects (around trial 5500), ensuring that the evaluation data reflect ongoing learning rather than the saturation regime. This also avoids cutting off training too early, where little learning signals are present. We then evaluate performance on the next 500 trials beyond that point, although similar trends were observed when using 200 or 1000 trials instead. As above, we compute paired t-tests on animal-matched log-likelihoods to assess significance.

Learning model, initial GLM weights, and hyperparameters. Our learning rule model is depicted in Fig. 1C. The network takes as input the current trial stimulus s_t, choice y_t, reward r_t, and the current weight $\mathbf{w}_t$. For IBL data, we also include an additional binary input indicating daybreak. The network then outputs inferred weight update. For choice and reward, we input the animal's actual choice and reward rather than those generated by the model, since these are the signals the animal would have access to when updating its weights; this approach is also consistent with previous studies. The DNN architecture consists of two hidden layers with 32 units each, while the RNN uses a single recurrent layer with 32 hidden units. All architectural components and training settings are treated as hyperparameters and tuned via cross-validation by animal held-out. All neural network weights and biases were treated as trainable parameters.

To estimate the initial decision weights $\mathbf{w}_0$, we treat them as trainable parameters but initialize them using a psychometric curve fit to the first 100 trials. Specifically, we compute the empirical probability of choosing $y = 1$ as a function of stimulus value, which defines a psychometric curve. Since we model the choice probability as $\sigma(\mathbf{w}^\top x)$, where σ is the logistic sigmoid, we apply

the inverse-sigmoid (logit) transformation to the empirical probabilities, yielding a linear target $b = \text{logit}(p(y = 1|x))$. We then fit $\mathbf{w}_0$ via a least-squares regression to the equation $\mathbf{w}^\top x = b$ using the stimulus values. We investigate the sensitivity of performance to incorrect $\mathbf{w}_0$ in Table 5. As mentioned, to enable faster simulations and repeated robustness tests, we used shorter trial sequences. Because these provide fewer early samples, we initialized with a known initial weight. That said, applying our method to unknown initial weight still led to accurate recovery (Supp. Fig. 10). For IBL data, however, we instead initialize $\mathbf{w}_0 = \vec{0}$, following [19], since animals typically start at chance-level performance; we also verified that training $\mathbf{w}_0$ did not significantly impact our results on IBL data.

We tune hyperparameters, including the number of hidden units, number of layers, learning rate, and number of training epochs. We used a batch size of 1 for simulated data. For real IBL data, we trained on all animals jointly; using a batch size of 1 instead did not affect our conclusions. Cross-validation ensures generalization and helps mitigate overfitting, in addition to the fact that neural networks in feature learning regime inherently favor smooth function approximators via implicit regularization (see Discussion). These implicit biases are themselves controlled by architectural choices and training regimes (e.g., depth, width, and training duration) [84, 104], which justifies the need for hyperparameter tuning. While we do not apply any explicit regularization in this work, exploring the interplay between explicit and implicit regularization on learning rule identifiability remains an interesting direction for future research. Validation-driven model selection is crucial to avoid recovering an incorrect learning rule that merely overfits the training data — achieving high training log-likelihood but poor generalization to held-out data.

Computational details. Our code is available `https://github.com/Helena-Yuhan-Liu/InferLearningANNGLM`. We implemented our models in PyTorch v1.13.1 [105]. Simulations were run on a machine equipped with a 12th Gen Intel(R) Core(TM) i7-12700H processor (14 cores, 20 threads) with a maximum clock speed of 2.30GHz. Training averages to around 2 hours per run. We used the Adam optimizer with a default learning rate of $1e-3$, and binary cross-entropy loss, which corresponds to the negative log-likelihood under a Bernoulli model. For statistical analysis and significance testing, we used tools from the SciPy package [106].

D Additional results

Fig. 7 repeats Fig. 2 for additional bias weight values. Fig. 8 shows the fitted bias weight updates for the IBL dataset. Table 2 echoes the trends in Table 1 with future data heldout instead of animal-level heldout. Additionally, we also demonstrate our main findings are robust across a variety of conditions: (i) a learning rule without the decision outcome assymetry (Supp. Fig. 6); (ii) recovery of an in-between learning rule when mixing update functions across individuals (Supp. Fig. 11); (iii) different initial stimulus weights (Supp. Fig. 9); (iv) longer trial lengths (Supp. Fig. 10). We also show sensitivity to the initial weight estimate (Supp. Table 3) and degradation of recovery under added noise and incorrect initial weight estimates (Supp. Table 4 and 5). Comparisons for parameterizing the weights $\mathbf{w}$ versus the updates $\Delta\mathbf{w}$ are provided in Fig. 14. Finally, we show the uncovered weight trajectories in Fig. 15 and 16.

During initial rebuttal experiments, we observed slightly higher performance for the Transformer variant compared to the RNN under limited training epochs. In the final analysis with full training epochs, this trend no longer holds — performance across architectures is comparable ($p = 7.44e-1$). We note that RNN-Transformer comparison is not central to our main contribution. We hypothesize that the comparable performance may be due to the small sample size (12 animals, as in [19]). We anticipate that the scalability advantages of Transformer and state-space models (e.g. Mamba) will become increasingly important as we extend to larger datasets.

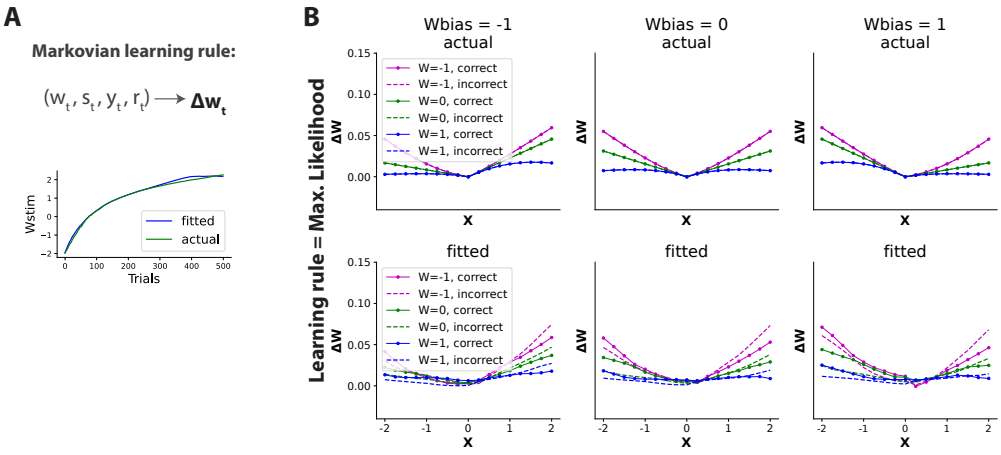

Figure 6: Similar to Fig. 2 but repeated for Max. Likelihood as the ground truth in simulated data. Again, we see that the reconstruction successfully captures key characteristics: the slowing of learning at higher weights, the increase with stimulus amplitude, and the lack of clear separation between correct versus incorrect decisions.

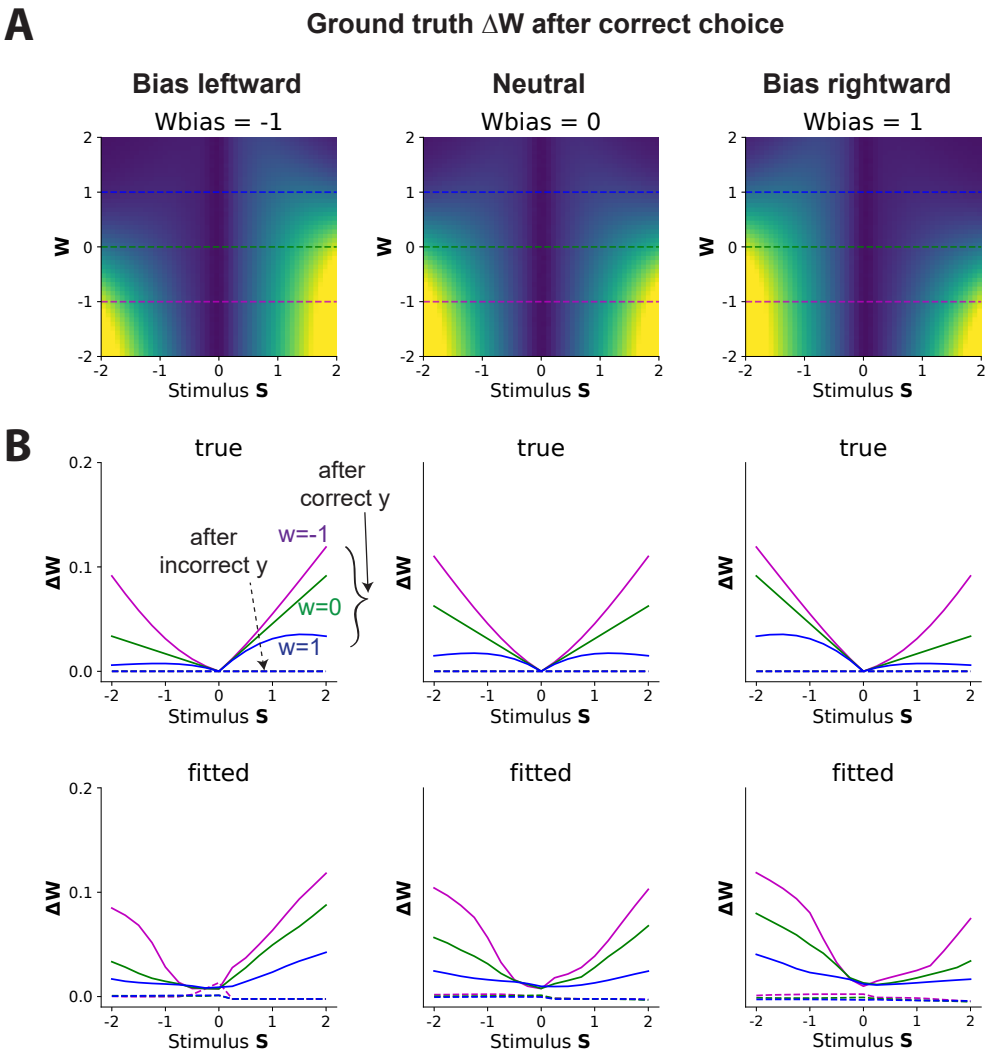

Figure 7: Similar to Fig. 2, but repeated while holding at other bias weight values in simulated data. The trends in Fig. 2 hold, and our reconstruction also captures the side asymmetry when the bias weight is nonzero.

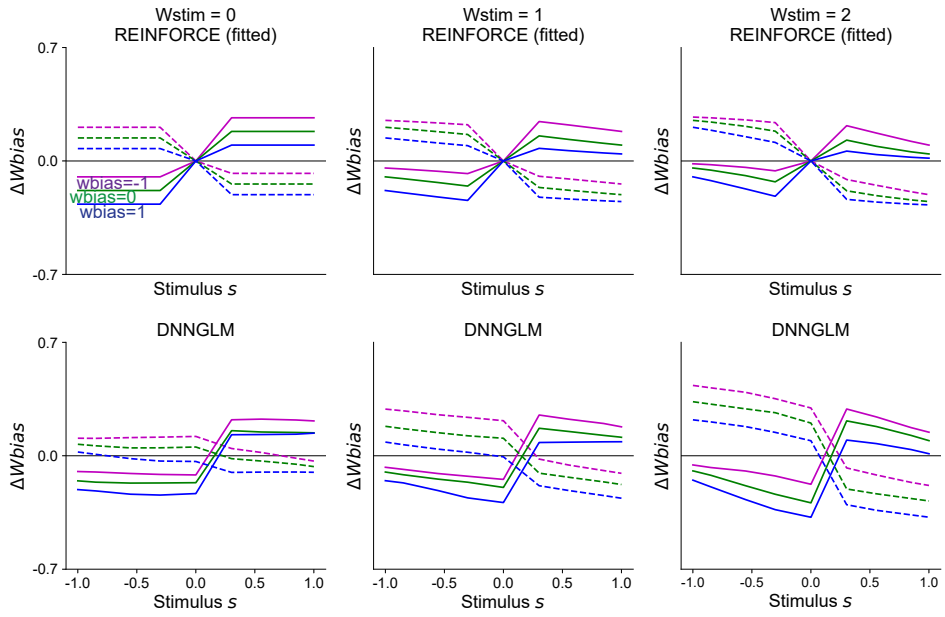

Figure 8: **Bias weight update function closely resembles REINFORCE.** Similar to Fig. 5, which visualizes the stimulus weight update function, we plot here the inferred update function for the bias weight. We plot REINFORCE (fitted on the DNNGLM) in the top row and DNNGLM on the bottom row. The fitted neural network model captures key qualitative features of the REINFORCE update rule for the bias weight.

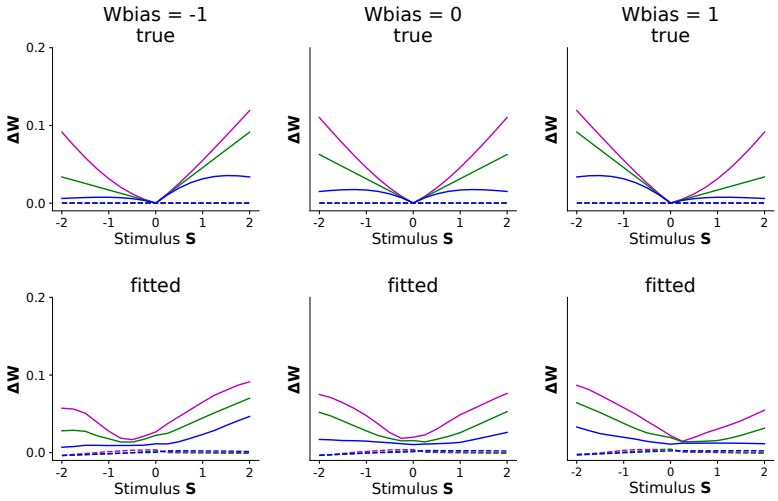

Figure 9: **Different initial stimulus weight $\mathbf{w}_0$. (A)** In our main results for simulated data, we initialized with stimulus weight $\mathbf{w}_0 = -2$ so that the animal starts with poor performance and learns over time, thereby covering a wide range of stimulus weights during training. Here, we instead randomly initialize $\mathbf{w}_0 \sim \mathcal{U}[-2, 2]$ (so different animals begin with different initial weights) and observe similar overall trends in the learned weight update function. Plotting convention follows that of Fig. 2.

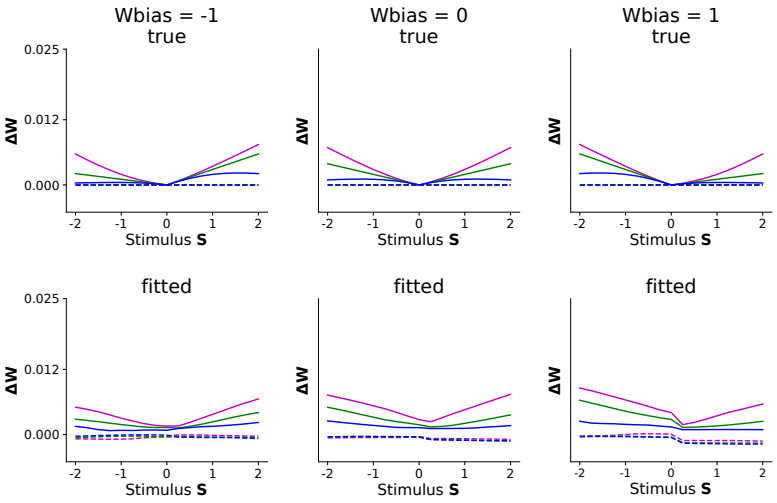

Figure 10: Trends observed in Fig. 2 hold for longer trial sequences ($T = 8000$). Plotting conventions are identical to those in Fig. 2 for simulated data.

Comparison	**Heldout LL**	p**-val**
REINFORCE vs. **DNNGLM**	-2924.4 vs. $\mathbf{-2905.8}$	$1.41\text{e}{-2}$
DNNGLM vs. **RNNGLM**	-2905.8 vs. $\mathbf{-2881.2}$	$2.24\text{e}{-3}$
REINFORCE vs. nonnegative base	$\mathbf{-2924.4}$ vs. -3272.4	$1.08\text{e}{-9}$
REINFORCE (history) vs. **RNNGLM**	-2914.8 vs. $\mathbf{-2881.2}$	$5.61\text{e}{-4}$
DNNGLM (history) vs. **RNNGLM**	-2890.2 vs. $\mathbf{-2881.2}$	$3.52\text{e}{-2}$
tinyRNN vs. **RNNGLM**	-2893.8 vs. $\mathbf{-2881.2}$	$3.38\text{e}{-2}$

Table 2: Future-data prediction log-likelihoods (higher is better) for different learning models, along with p-values from paired t-tests matched across mice. These results mirror the trends observed in Table 1.

Initial stimulus weight $\mathbf{w}_0$	RMSE of reconstructed $\Delta\mathbf{w}$
-2	0.0130
0	0.0244
2	0.0287

Table 3: **Impact of initial stimulus weight $\mathbf{w}_0$ on reconstruction accuracy.** We measure reconstruction RMSE of the inferred $\Delta\mathbf{w}$ when varying the ground truth initial weight $\mathbf{w}_0$ in simulated data, keeping all other factors fixed. Initializing with $\mathbf{w}_0 = -2$ yields the lowest error, while $\mathbf{w}_0 = 2$ leads to saturation and poor identifiability. This pattern is consistent with Fisher Information analysis (Supp. A): having more datapoints around weight values near $\mathbf{w} = 0$ correspond to choice probabilities near 0.5, which maximize Fisher Information and enhance identifiability of learning rules. Starting from $\mathbf{w}_0 = -2$ ensures the learning trajectory passes through this region, enabling more accurate recovery of the update function. This trend holds even in light of the fact that the value of $\Delta\mathbf{w}$ would be smaller for higher $\mathbf{w}$ in REINFORCE. This analysis suggests that having animals initially perform poorly on the task, even below chance level, could potentially provide more information for learning rule identification.

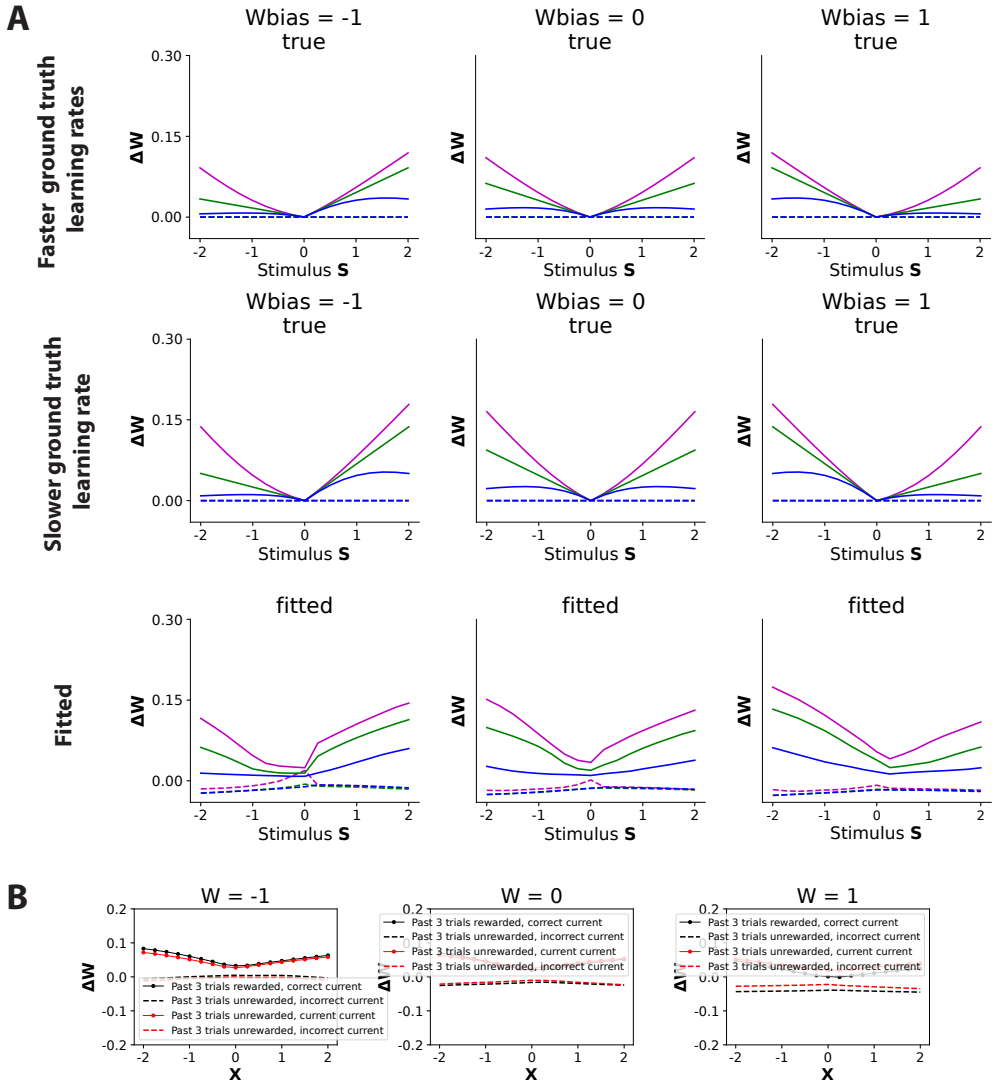

Figure 11: **Pooling with mixed weight update functions. (A)** Following the plotting convention in Fig. 2, we simulate a mixed population where half the animals learn with REINFORCE using a higher learning rate (top row) and the other half with a lower learning rate (middle row). The fitted function (bottom row) recovers an intermediate update rule that lies between the two subpopulations. **(B)** To test whether the reward-history dependence observed in Fig. 5 is merely a result of faster-learning animals in the pool contributing more recently rewarded trials, we train RNNGLM on a population with mixed learning rates. Reward-conditioned weight update plots — where past trial rewarded (black) and unrewarded (red) — do not consistently show the reward history dependence in this setting. This suggests that the history dependence learned by RNNGLM in IBL data is not merely due to faster learners receiving more recent rewards; this makes sense as in both this plot and Fig. 5, the learning stage is controlled by fixing the stimulus weight $\mathbf{w}$.

Noise Level	RMSE (Reconstructed Δw vs. Ground Truth)
$\sigma = \frac{1}{4}\alpha$	0.0216
$\sigma = \frac{1}{2}\alpha$	0.0349
$\sigma = 1\alpha$	0.0652

Table 4: Reconstruction accuracy degrades with increasing white Gaussian noise in simulated data. We add zero-mean white Gaussian noise to the ground truth REINFORCE learning rule for simulated data, with the noise standard deviation σ expressed as a fraction of the learning rate α. In other words, the ground truth weight changes are now driven by two components: a learning component and a noise component, as in [19], whereas our fitted model remains deterministic. We report RMSE between the reconstructed and true $\Delta\mathbf{w}$. As expected, higher noise leads to worse recovery accuracy.

$\mathbf{w}_0$ correctness	RMSE
Default	0.0130
+0.5 perturb	0.0270
+1.0 perturb	0.0453
+2.0 perturb	0.1519

Table 5: Effect of initial weight $\mathbf{w}_0$ correctness on reconstruction accuracy in simulated data. We test the following scenarios: default, and $\mathbf{w}_0$ offset by +0.5, +1.0, +2.0 after the estimation. Deviations from the true $\mathbf{w}_0$ can substantially degrade the accuracy of learning rule recovery, underscoring the importance of accurately estimating $\mathbf{w}_0$.

Additional Input Dim	Default Sample Size	3× Samples
1	1.00	0.59
3	1.09	0.62
10	1.35	0.7
30	1.58	1.20
100	X	1.67

Table 6: **Recovery performance (relative RMSE) of the inferred learning rule across increasing input dimensionality.** Each row corresponds to an input dimensionality setting; additional inputs are distractor streams added to the default task in Fig. 2. An "X" marks failure to capture the qualitative features in Fig. 2; features remain captured at higher-D task settings, and reconstruction accuracy improves with more data. Relative RMSE is normalized to the baseline (1 additional input, default sample size).

Condition	Relative RMSE
Decision model known	1.00
Model mismatch	1.62

Table 7: **Performance degradation with decision model mismatch.** We examine a multiplicative decision structure as the ground-truth model. The first row corresponds to the case where the inference framework knows the true multiplicative decision structure, $\text{logit}(P(\text{right})) = w_L \cdot c \cdot x_L + w_R \cdot (1 - c) \cdot x_R$, for rightward cumulative cue and context variable c between 0 and 1, and leftward cumulative cue between 0 and -1. The second row represents model mismatch, where the inference framework still assumes a GLM model, $\text{logit}(P(\text{right})) = \hat{w}_L \cdot x_L + \hat{w}_R \cdot x_R + \hat{w}_c \cdot c$.

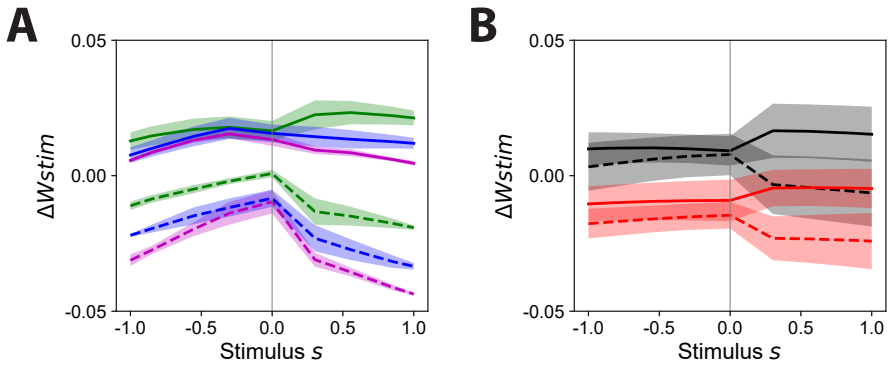

Figure 12: **Robustness of inferred learning rules across random seeds.** We ran all inference procedures with three different random seeds, each achieving similar test log-likelihoods, and show the mean $\pm$ standard deviation across seeds as shaded regions. (A) follows the plotting convention of Fig. 5A, showing stimulus- and choice-dependent weight updates. (B) follows the plotting convention of Fig. 5B, showing reward-history–conditioned updates. While some variation is present, the essential qualitative features—such as side bias, negative baseline, and reward-history dependence—are consistently recovered across seeds.

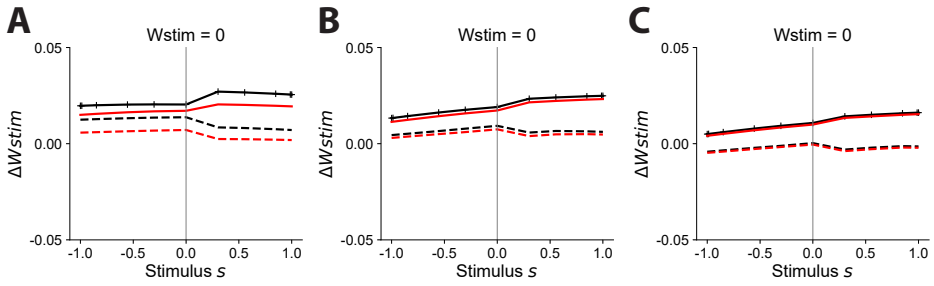

Figure 13: **Longer history effects observed in the model.** In Fig. 5B, we conditioned on up to four consecutive rewarded or unrewarded trials, since more consecutive trials risk extrapolation. Here we bypass this issue by still conditioning on four consecutive rewarded/unrewarded trials, but taken from further back in the trial history, to test whether the model can capture dependencies beyond the immediate past. (A) conditioned on trials 7 to 10 trials ago, (B) 17 to 20 trials ago, and (C) 27–30 trials ago. History effects are still detectable beyond just four trials, but decay with distance. Plotting conventions are the same as in Fig. 5B.

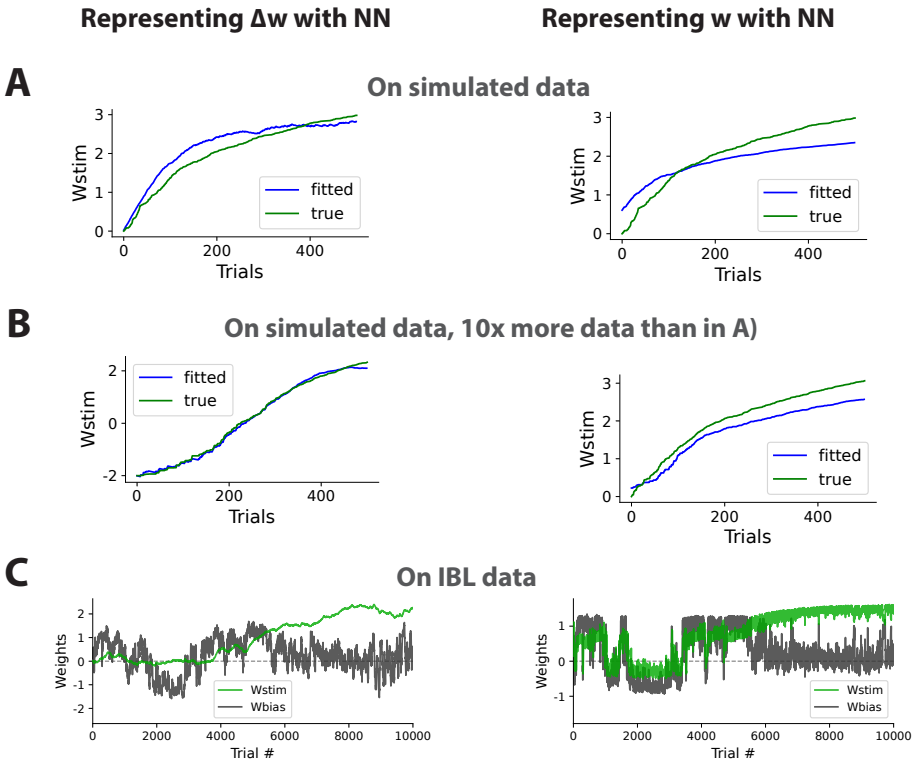

Figure 14: **Parameterizing $\Delta \mathbf{w}$ directly (as opposed to $\mathbf{w}$) improves weight trajectory recovery.** **(A)** On simulated data, using a neural network to directly represent the weight update $\Delta \mathbf{w}$ yields more accurate recovery of the weight trajectory compared to directly parameterizing $\mathbf{w}$ using an RNN. The RNN-based $\mathbf{w}$ trajectory is overly smooth, likely due to implicit regularization that biases the model toward simpler solutions in the weight trajectory rather than the weight update function; in contrast, regularizing the weight changes (via $\Delta \mathbf{w}$) provides a more appropriate inductive bias (see Discussion). **(B)** As expected, the effect of implicit regularization is more pronounced in underconstrained settings (e.g., low-data regimes), and becomes less critical as more data or constraints are introduced; since with sufficient data, both models should approximate the learning trajectory well as per the universal approximation results. **(C)** On IBL data, this distinction is also evident: when training on just one example animal, parameterizing $\Delta \mathbf{w}$ leads to more realistic weight trajectories than using an RNN to represent $\mathbf{w}$ directly.

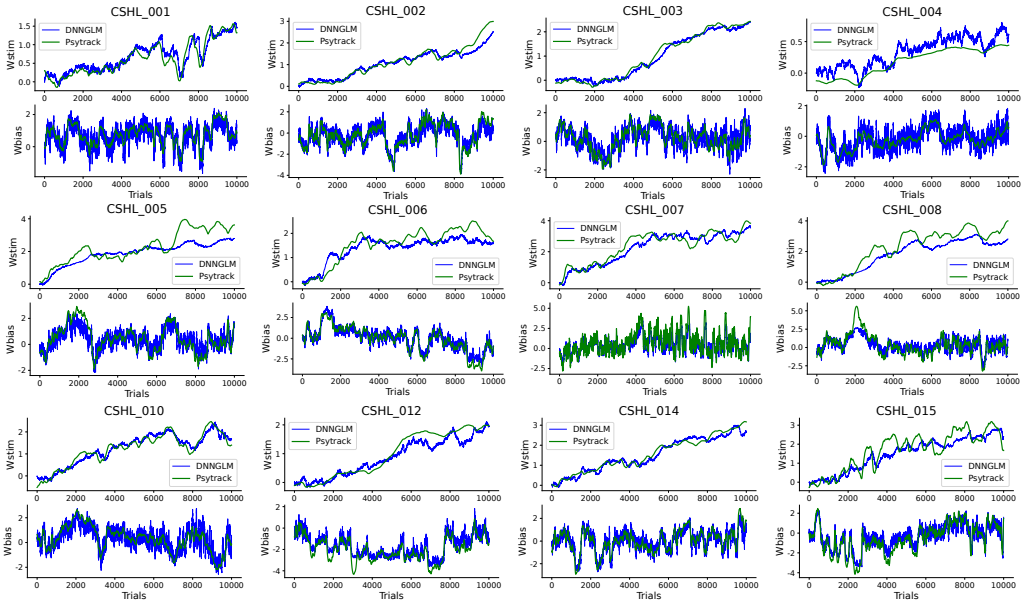

Figure 15: Extended version of Fig. 4A, showing both stimulus and bias weight trajectories across more animals. The decision weights recovered by DNNGLM closely resemble those inferred by PsyTrack.

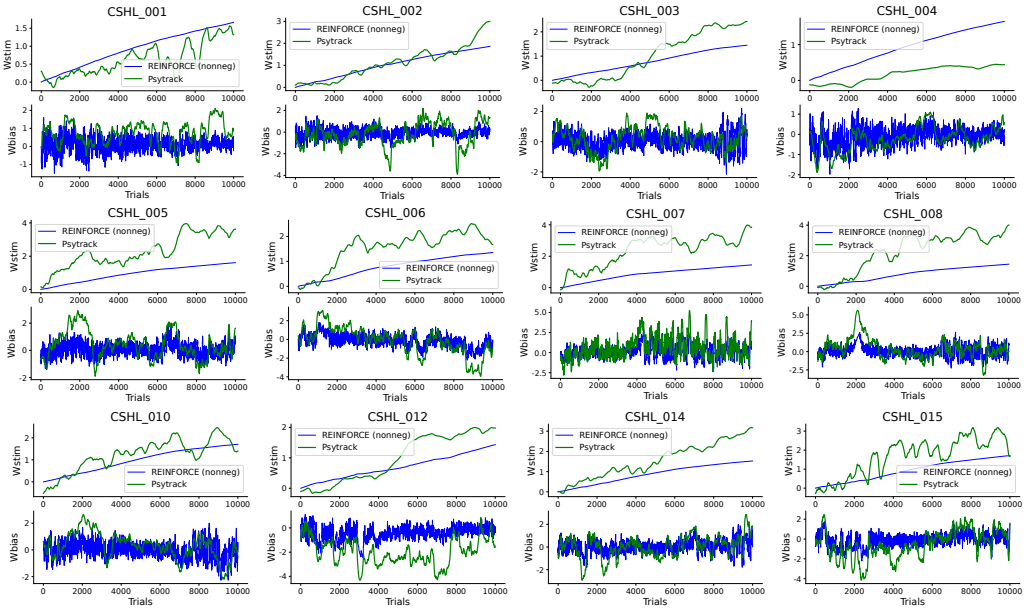

Figure 16: Similar to Supp. Fig. 15, but showing the decision weights recovered by REINFORCE (with the nonnegative baseline constraint), which was previously shown to yield worse log-likelihoods (Tables 1 and 2). We see here that it leads to greater discrepancies from the weights inferred by PsyTrack.

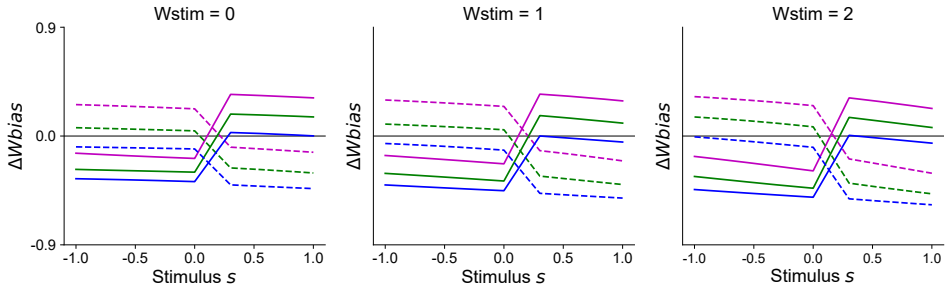

Figure 17: The inferred update function for the bias weight, using RNNGLM, qualitatively matches the fitting in Fig. 8.

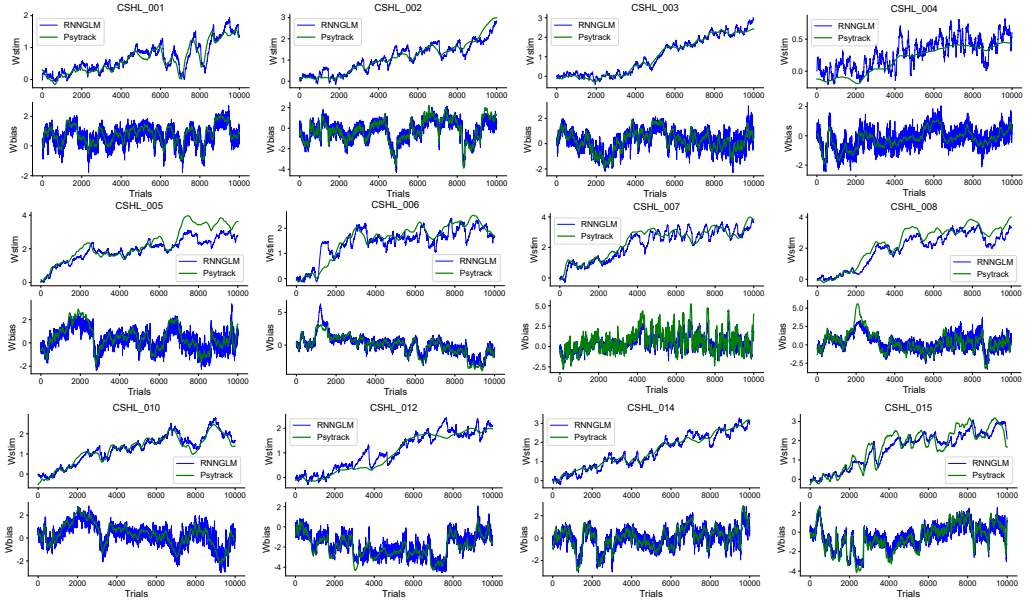

Figure 18: Similar to Fig. 15 but for RNNGLM, where recovered weights also closely resemble those inferred by PsyTrack.