Learning Efficiently in Uncertain and Structured Worlds

Simone D'Ambrogio

a dissertation
presented to the faculty
of the University of Oxford
in candidacy for the degree
of Doctor of Philosophy

recommended for acceptance
by the Department of Experimental Psychology
at Worcester College, Oxford

Advisor: Matthew F. S. Rushworth
Co-advisor: Laurence Hunt

April 2026

Abstract

Intelligence emerges from the interaction between learning mechanisms and the environments that shape them. This thesis investigates how biological and artificial systems navigate two fundamental challenges: learning under uncertainty when information is scarce or costly, and exploiting structured environments through abstraction and generalization.

The thesis develops both methodological innovations and empirical insights across three interconnected investigations. First, I introduce a hybrid modeling framework that combines theory-driven cognitive models with data-driven artificial neural networks, using symbolic regression to transform learned neural network functions into interpretable mathematical expressions. This approach bridges the flexibility of machine learning with the interpretability required for scientific understanding, providing a general method for discovering computational principles underlying cognition.

Second, I apply this framework to understand how humans compute the value of information during decision-making under uncertainty. Using ultra-high field 7T fMRI, I identify distinct neural signatures across neuromodulatory nuclei and cortical regions. The ventral tegmental area balances exploration versus exploitation by encoding opposing signals for information value and selection value. The anterior insula and anterior cingulate cortex guide the information sampling strategy. Symbolic regression reveals that the value of information follows exponential functions integrating evidence from both attended and unattended options, with parameters that capture individual differences in exploration strategies. These equations generalize to predict behavior in independent exploration-exploitation tasks.

Third, I investigate computational mechanisms underlying rapid generalization across sequential learning tasks. I develop computational models comparing baseline recurrent neural networks against architectures augmented with grid-hippocampal episodic memory systems. Models implementing abstract 2-dimensional maps through grid cell path integration achieve dramatic learning acceleration comparable to animal behavior. Critically, backward temporal credit assignment through causal attribution and episodic binding enables near-instantaneous transfer after initial learning. These computational findings generate specific neural predictions testable through fMRI pattern analysis and transcranial ultrasound stimulation.

Together, these investigations provide evidence consistent with the idea that intelligence may emerge from structured representations that facilitate rapid learning, effective information sampling, and flexible generalization across contexts.

Contents

Acknowledgments

This thesis would not have been possible without the support, encouragement, and patience of many people who have been part of this journey.

First and foremost, to Maria, my partner: you have been by my side through the easiest and hardest moments of this PhD. You never gave up on me, even when Zach and Zeno occupied way too much of the space between us. I owe you more than words can express. Your unwavering support has been my anchor.

To my parents, Ginetta and Piero: thank you for always putting my happiness first. Your constant encouragement and belief in me have been a source of strength throughout this journey. I am incredibly fortunate to have parents who support not just my ambitions, but my wellbeing above all else.

To my supervisor, Matthew: thank you for being there always. Our conversations have been some of the most exciting and intellectually stimulating experiences of my academic life. Your kindness, your infectious passion for this work, and your ability to energize those around you have made this journey truly enjoyable. I could not have asked for a better mentor!

Finally, to every member of the lab: thank you for creating an environment that felt less like a workplace and more like a family. Your support and friendship have made even the most challenging days manageable and the good days unforgettable.

1 Introduction

In the summer of 1961, a young chimpanzee named Jane sat at the edge of a termite mound in what is now Gombe National Park, Tanzania. She had been watching other chimpanzees for weeks, studying their behavior with intense curiosity. Suddenly, she selected a grass stem, carefully stripped off its leaves, and inserted it into the termite mound to fish for insects. When primatologist Jane Goodall witnessed this moment through her binoculars, she immediately sent a telegram to her mentor Louis Leakey: "Now we must redefine tool, redefine Man, or accept chimpanzees as beings into the community of Man."

This single act revealed intelligence operating across two fundamental dimensions that define this thesis. First, Jane had reduced uncertainty about termite fishing by observing other chimpanzees—learning from social information when direct experience was costly or unavailable. Second, she demonstrated remarkable abstraction: recognizing that a grass stem could serve as a tool, that its leaves needed removal, and that this solution could be applied across different termite mounds. She had extracted a structured pattern from her environment, the relationship between tool properties and foraging success, and generalized it to new situations.

These two aspects of Jane's behavior capture the essence of intelligence: navigating uncertain worlds by sampling information strategically, and exploiting structured environments through abstraction and generalization. This thesis explores how such intelligence emerges through the fundamental interplay between learning mechanisms and the environments that shape them, from the neural circuits that process social and environmental information to the algorithms that guide decisions when information is scarce or when patterns can be discovered and exploited.

1.1 Intelligence

Understanding intelligence is often regarded as one of the most profound and ambitious challenges in science. As the neuroscientist Tomaso Poggio eloquently said, "[the problem of intelligence] is not only one of the great problems in science, like the origin of the universe—it's actually the greatest of all, because it means understanding the very tool we use to understand everything else: our mind." [98].

Despite centuries of inquiry, the nature of intelligence remains one of the most contested topics among scholars. Researchers have proposed dozens of competing definitions, from Spearman's general intelligence factor to Gardner's multiple intelligences, from information processing approaches to embodied cognition theories. This diversity reflects not just academic disagreement, but the genuine complexity of the phenomenon itself.

The quest to understand intelligence draws from diverse fields. Neuroscientists seek to uncover the biological basis of intelligent behavior; computer scientists aim to replicate it through artificial systems; psychologists study its development and variability across individuals and species. Yet despite this diversity, a unifying question remains: How do systems (biological or artificial) acquire knowledge, generalize from experience, and make adaptive decisions?

Among the numerous theoretical perspectives, this thesis adopts the view that intelligence can be most productively understood as a fundamental mechanism that evolution has refined to solve the core challenge of adaptive behavior: the ability to *learn* about the *world* in order to act effectively within it. This perspective cuts through definitional debates by focusing on intelligence's primary evolutionary function: transforming experience into adaptive action across diverse and changing environments.

It is within this perspective that learning and decision-making take center stage. While intelligence is a vast construct, the capacity to learn from experience and to make context-sensitive decisions represents its most fundamental and functionally important expression. These processes lie at the intersection of multiple brain systems and cognitive functions and are increasingly tractable with the tools of modern cognitive and computational neuroscience.

In this first chapter, I will break down intelligence into two core components: *learning* —the mechanisms by which systems acquire and update knowledge from experience— and two key features of the *world* (or environment) that shape those experiences. Intelligent behavior is always defined in relation to the environment in which it unfolds, making the choice of environmental features critical. Among the many properties that characterize the world we live in, I will focus on two that are especially consequential for learning: its *uncertainty*, and the *structure* embedded in the experiences it generates. These features pro-

vide a useful starting point for understanding how intelligence emerges from the dynamic interplay between learning processes and the nature of the environment.

1.2 Learning

Imagine an alien visitor arriving on Earth for the first time. She is not the conquest-minded extraterrestrial of science fiction, but a curious being with a singular passion: discovering the finest cuisine our planet has to offer. This alien knows nothing about human customs, social structures, or the intricate rituals that govern our daily interactions. Every experience is novel, every situation unpredictable. She is called Hannah, and we will call her Ann for short.

On her first venture into a restaurant, the experience is overwhelming. She doesn't know what to do or in which order to do it. Should she wait outside for assistance, find her own seat, or perhaps sit at a table already occupied by other diners? Each decision presents a puzzle with no obvious solution. Without the ability to learn from experience, these seemingly simple questions would remain perpetually unanswered, making every interaction with the environment an exhausting exercise of trial and error.

After a few visits to the same Italian trattoria, she finally masters the local customs and behavioral sequences. But imagine if she had to learn everything from scratch at every new restaurant, or even with every new situation at the same restaurant (e.g. a different waiter, the tables set differently, etc.). Navigating the world would become a constant process of learning from scratch, consuming enormous amounts of time and cognitive energy that could otherwise be allocated to other tasks.

Ann needs the ability to extract general principles from specific experiences and generalize this knowledge to new situations. The temporal structure learned at the trattoria can be adapted to different restaurant types, even when the specific instances of the events in the sequence change dramatically.

This capacity for abstraction and generalization is not merely convenient, it is essential for navigating a world that is both structured and infinitely variable. Without it, every novel situation would demand complete reacquisition of behavioral strategies. Instead, intelligent systems, whether alien visitors, artificial, or biological agents, can leverage the inherent structure in their environments to build flexible, transferable knowledge that scales across contexts.

1.2.1 Reinforcement Learning as a Conceptual Framework

Before Ann can generalize from an Italian trattoria to a Japanese sushi bar, she must first learn the basic sequence of actions that leads to a satisfying meal in any single instance.

To dissect this fundamental learning process, we need a precise language to describe the interplay between an agent and its world. The framework of *reinforcement learning* (RL) provides exactly such a language. RL is not a specific algorithm, but rather a formal framework for understanding and automating goal-directed learning and decision-making [116]. It provides a precise mathematical language for describing how an agent can learn to act optimally in an environment by trial and error, guided only by a signal of success or failure. In RL, the agent must discover which actions yield the most reward by actively interacting with the world.

This learning-from-interaction loop is fundamental to how humans and other animals adapt to their surroundings. We try things, observe the consequences, and adjust our future behavior accordingly. RL formalizes this intuitive process, providing a powerful lens for modeling cognition [16, 43]. The framework is built around a few core components: an *agent*, an *environment*, and signals that pass between them: *actions*, *states*, and *rewards*.

The *agent* is the learner and decision-maker. In our example, this is Ann. The *environment* is everything the agent interacts with, comprising everything outside the agent. For Ann, this includes the restaurants, her phone, the city streets, and the passage of time. The agent selects *actions* (a), and the environment responds by presenting new situations, or *states* (s), to the agent. A state is a summary of the environment that provides the agent with all the necessary information to make a decision. For example, Ann might take the action of tapping on the "reviews" tab for a restaurant on her phone. The environment's response is a new state: the screen now displays a list of reviews. Crucially, a well-defined state possesses the *Markov property*: the future is independent of the past, given the present. This means that the current state encapsulates all relevant information from the history of interactions, and nothing more is needed to predict what will happen next. Formally, this is written as:

$$p(s_{t+1}, r_{t+1} \mid s_{0:t}, a_{0:t}) = p(s_{t+1}, r_{t+1} \mid s_t, a_t).$$

This property is a powerful simplification. The agent doesn't need to remember every single action it has ever taken; it only needs to know its current state to act optimally.

Finally, the environment provides *rewards*, which are numerical signals indicating the immediate value of a state transition. A high reward signals a good outcome (e.g., enjoying a delicious meal), while a low or negative reward signals a bad one (e.g., wasting time on a fruitless search, or arriving at a closed restaurant).

The agent's sole objective is to maximize the total reward it accumulates over the long run, a quantity known as the *return*. This simple, powerful objective drives the entire learning process. To achieve it, the agent must develop a *policy*, a function that takes state as input and outputs an action. A good policy, learned through experience, allows Ann

to navigate the complex world of human dining rituals to achieve her culinary goals efficiently.

Let's return to Ann, who is trying to decide where to dine. A useful state for her decision-making process might include: (i) the restaurant she is currently considering (e.g., Italian or Japanese); (ii) a summary of the evidence she has gathered for each option (e.g., the number of reviews read and their average rating); (iii) external constraints like the time remaining before the kitchens close; and (iv) her own internal state (e.g., how hungry she is). Her possible actions are intuitive: she can *stay* and gather more information about the current option, *switch* her attention to the other restaurant, or *select* an option and end her search making a reservation. The rewards she receives are a combination of the quality of her meal (positive reward) and the costs of her search in time and effort (negative reward). This entire sequence of states, actions, and rewards, is an instance of sequential decision-making under uncertainty, which can be formalized as a *Markov Decision Process*.

The Formalism of Markov Decision Processes. A reinforcement learning task that satisfies the Markov property is called a Markov Decision Process, or MDP. An MDP is defined by a tuple $(\mathcal{S}, \mathcal{A}, T, R, \gamma)$, where:

- $\mathcal{S}$ is the set of all possible states.
- $\mathcal{A}$ is the set of all possible actions.
- $T(s' \mid s, a)$ is the *transition function*, which gives the probability of moving to state s' after taking action a in state s.
- $R(s, a)$ is the *reward function*, which specifies the immediate reward received after taking action a in state s.
- γ is the *discount factor* ($0 \leq \gamma \leq 1$), which determines the present value of future rewards. A discount factor of 0 makes the agent "myopic," caring only about immediate rewards, while a factor approaching 1 makes it "farsighted," striving for long-term success.

The agent's goal is to maximize the *return*, G_t, which is the discounted sum of rewards from time step t onward:

$$G_t = \sum_{k=0}^{\infty} \gamma^k r_{t+1+k}. \tag{1.1}$$

The discount factor ensures that this sum is finite and gives more weight to immediate rewards than to distant ones, a feature that is both computationally convenient and psychologically plausible.

Value Functions and Optimal Policies. To maximize its return, the agent needs to learn a policy, $\pi(a|s)$, which is a strategy that specifies which action to take in each state. But how can the agent know which policy is best? This is where the concept of *value functions* becomes essential. A value function estimates "how good" it is to be in a particular state, or to take a particular action in a state.

The most fundamental value function is the *state-value function*, $V^\pi(s)$, which is the expected return when starting in state s and following policy π thereafter:

$$V^\pi(s) = \mathbb{E}_\pi[G_t \mid S_t = s]. \tag{1.2}$$

$V^\pi(s)$ satisfies the *Bellman expectation equation*, which decomposes the value of a state into the expected immediate reward plus the discounted value of the successor state:

$$V^\pi(s) = \sum_a \pi(a|s) \left(R(s,a) + \gamma \sum_{s'} T(s'|s,a) V^\pi(s') \right). \tag{1.3}$$

While state-values are useful for evaluating a policy, to choose an action an agent often needs to know the value of each specific action. This motivates the *action-value function*, $Q^\pi(s,a)$, defined as the expected return after taking action a in state s and thereafter following policy π:

$$Q^\pi(s,a) = \mathbb{E}_\pi[G_t \mid S_t = s, A_t = a]. \tag{1.4}$$

The Bellman expectation equation for the action-value function is:

$$Q^\pi(s,a) = R(s,a) + \gamma \sum_{s'} T(s' \mid s,a) \sum_{a'} \pi(a' \mid s') Q^\pi(s',a'). \tag{1.5}$$

This equation says that the value of taking action a in state s is the immediate reward plus the discounted value of the next state, averaged over all possibilities.

The ultimate goal is to find an *optimal policy*, π^*, that achieves the highest possible return from all states. An optimal policy is one that maximizes the value functions. The optimal action-value function, $Q^*(s,a)$, satisfies the *Bellman optimality equation*:

$$Q^*(s,a) = R(s,a) + \gamma \sum_{s'} T(s' \mid s,a) \max_{a'} Q^*(s',a'). \tag{1.6}$$

The immediate reward $R(s,a)$ is straightforward—it's what you get right away for taking action a in state s. The second term, $\gamma \sum_{s'} T(s' \mid s,a) \max_{a'} Q^*(s',a')$, captures the value of the future. Here's what's happening:

- $T(s' \mid s,a)$ gives the probability of transitioning to each possible next state s'.

- $\max_{a'} Q^*(s', a')$ identifies the value of the best action you could take from that next state.
- The summation averages this maximum value across all possible next states, weighted by their probabilities.
- γ discounts this future value to reflect that rewards now matter more than rewards later.

The expectation in this equation is taken over next states, not actions. Because the environment is stochastic, you don't know with certainty which state s' you'll end up in after taking action a. But once you arrive in any particular next state, an optimal agent will choose the action that yields the highest Q-value. This is the crucial insight: we average over the uncertainty in the world (which states might occur), but we maximize over our own choices (which actions to take).

Finally, if you know Q, finding the optimal policy becomes trivial. In any state s, simply compare $Q^*(s, a)$ for all available actions and select the one with the highest value: $\pi^*(s) = \arg\max_a Q^*(s, a)$. No complex computation is needed—the optimal decision is directly encoded in the Q-function itself. This is why Q-learning and related algorithms focus on learning Q: once you have accurate estimates of these action-values, optimal behavior follows immediately.

1.2.2 Reinforcement Learning in the Brain

While the mathematical framework we have described provides a complete theoretical foundation for reinforcement learning, real-world agents face a fundamental computational constraint. Computing the optimal policy via the Bellman optimality equations requires perfect knowledge of the environment's dynamics and enormous computational resources. For most realistic problems, this approach is computationally intractable. Instead, agents must learn to approximate optimal policies through direct interaction with the environment, updating their value estimates based on the rewards and state transitions they actually experience.

One of the most influential approaches to this problem is *temporal-difference (TD) learning*, which updates value estimates using the difference between predicted and observed outcomes. The core insight of TD learning is captured in the *temporal-difference error* or *reward prediction error (RPE)*:

$$\delta_t = R_{t+1} + \gamma V(s_{t+1}) - V(s_t) \tag{1.7}$$

This error signal represents the discrepancy between what the agent expected to receive (its current value estimate $V(s_t)$) and what it actually experienced (the immediate reward

R_{t+1} plus the discounted value of the next state). When $\delta_t > 0$, the outcome was better than expected; when $\delta_t < 0$, it was worse than expected; and when $\delta_t = 0$, the prediction was accurate. The agent can then use this error to incrementally update its value function, gradually improving its estimates through experience.

The abstract framework of reinforcement learning finds a striking parallel in the neurobiology of the brain. A wealth of evidence in neuroscience suggests that the brain implements a form of RL where the neurotransmitter dopamine plays a central role in computing and broadcasting a reward prediction error (RPE) signal — the discrepancy between an actual and an expected reward [103]. This RPE signal, representing the TD error δ_t from Equation 1.7, is thought to be the key teaching signal that drives trial-and-error learning in biological agents.

The foundational evidence for this hypothesis came from a series of seminal experiments by Schultz, Montague, and Dayan, who recorded the activity of individual dopamine neurons in the ventral tegmental area (VTA) and substantia nigra pars compacta (SNc) of the monkey midbrain during a classical conditioning task [103]. In the initial phase of the experiment, before any learning had occurred, a monkey would receive an unexpected drop of juice. At the moment the juice was delivered, its dopamine neurons responded with a sharp, phasic burst of activity. This response signaled a positive prediction error: the outcome was better than expected.

The experiment then entered a learning phase, where a neutral sensory cue, such as a light or a sound, was consistently presented a fixed time before the juice was delivered. Over many trials, the monkey learned to associate the cue with the impending reward. Once this association was established, the firing pattern of the dopamine neurons changed: they no longer fired when the juice reward was delivered; because the reward was now fully predicted by the cue, the prediction error at the time of reward was zero. Instead, the neurons now fired a phasic burst in response to the earliest reliable predictor of the reward—the sensory cue itself. The value signal had effectively been transferred from the outcome to the state that predicted it.

Crucially, the experiment included a third condition that sealed the interpretation. After the association was learned, the researchers would occasionally present the cue but then withhold the expected juice reward. At the precise moment the reward should have arrived, the dopamine neurons showed a brief, pronounced pause in their firing, dipping below their baseline activity level. This signaled a negative prediction error: the outcome was worse than expected. Taken together, these three conditions provided a stunning demonstration that dopamine neurons do not simply encode reward, but rather the error in the prediction of reward. This finding, later replicated in other animals [24, 59, 101] and in humans [5] using fMRI, provided a direct neural correlate of the temporal-difference

learning signal at the heart of RL models.

While these findings were compelling, little causal evidence was provided until a few years later, when Steinberg and colleagues (2013) used optogenetics to manipulate dopamine neurons in the VTA and SNc [112].

Optogenetics is a technique that allows for the precise control of neural activity with light [39]. The method involves genetically targeting specific populations of neurons to express light-sensitive proteins called opsins. By delivering targeted pulses of light, typically through an implanted fiber-optic cable, researchers can then activate or inhibit these specific neurons with millisecond precision. This ability to directly manipulate the activity of a defined set of neurons in a behaving animal provides a powerful tool for establishing causal links between neural activity and behavior, overcoming the correlational limitations of previous methods.

To establish this causal link, Steinberg and colleagues turned to a classic experimental design known as "blocking," which elegantly tests whether learning is driven by prediction error [112]. In a standard blocking experiment, an animal first learns that a cue (Cue A) predicts a reward. Then, a second cue (Cue B) is presented together with Cue A, followed by the same reward. Because the reward is already fully predicted by Cue A, there is no prediction error, and consequently, the animal fails to learn any association between the redundant Cue B and the reward.

The procedure unfolds in stages. First, a rat learns that a specific sensory cue, such as a sound (Cue A), reliably predicts a sugar water reward. The rat's expectation of the reward can be measured by its behavior: as it learns the association, it will spend an increasing amount of time with its head in the reward port during the sound, anticipating the delivery of the juice. Once this association is well-learned, and the rat reliably waits at the port, a second phase begins where a new cue, like a light (Cue B), is presented simultaneously with the original sound (Cue A), followed by the same reward. For this rat, the reward is now fully expected due to the presence of the sound, meaning there is no surprise or prediction error. In a final test phase, the rat is presented with only the light. As predicted by the blocking effect, the rat shows little to no conditioned response, spending minimal time at the reward port. The prior learning about the sound has "blocked" any new learning about the light. This behavioral effect is mirrored in the brain: dopamine neurons, which fire for unexpected rewards, show little to no response to the reward during the blocking phase [128].

Steinberg and colleagues reasoned that if the phasic dopamine signal is indeed the teaching signal, then artificially creating a prediction error during the blocking phase should "unblock" learning and allow the animal to learn about the light cue (Cue B). This is precisely what they found.

By using optogenetics to stimulate dopamine neurons at the exact moment the reward was delivered in the blocking phase, a moment when these neurons would normally be silent, they successfully induced learning about the otherwise blocked cue. When later presented with the light alone, these rats showed a robust conditioned response, spending significantly more time at the reward port than control animals that did not receive the stimulation. This elegant experiment provided powerful causal evidence that the phasic dopamine signal is not merely correlated with learning; it is the instructive signal that drives the updating of value predictions in the brain.

This dopamine-based learning system is orchestrated by a well-defined neural circuit [132]. To compute a prediction error, dopamine-releasing neurons in the ventral tegmental area (VTA) and substantia nigra pars compacta (SNc) must receive information about both expected and actual outcomes. These nuclei act as hubs, integrating inputs from cortical and subcortical areas that carry information about sensory cues, internal states, and value expectations. The resulting RPE signal is then broadcast widely, most notably to the striatum, a brain region critical for action selection. Here, dopamine release modulates synaptic plasticity, strengthening connections that led to positive surprises and weakening those that led to negative ones [11]. In essence, the dopamine system provides a continuous, real-time feedback signal that allows the brain to learn from its mistakes and successes, progressively refining its policies to maximize future rewards. This elegant biological mechanism provides a concrete implementation of the abstract principles of reinforcement learning, grounding the computational theory in the brain.

1.2.3 Uncertain World

In the previous section we have explored general principles of learning, abstracting away from the intricacies of particular environments. However, our account of intelligence remains incomplete without considering the *environment* in which learning unfolds. As we established earlier, intelligence can be defined as learning to act effectively in the world. This means that the properties of the environment are not merely background conditions, but active shapers of how intelligent behavior emerges and what forms it takes.

Let us return to our alien Ann, but now consider the profound challenge she faces not in learning the mechanics of restaurant dining, but in navigating the fundamental uncertainty that permeates every aspect of her new earthly experience. Each decision Ann makes, such as where to eat, how long to wait, or whether to try something new, must be made in the face of a pervasive uncertainty about the true state of the world.

This uncertainty is not unique to Ann's culinary adventures; it is a fundamental feature of virtually every environment in which intelligent agents must operate. Extensive evidence from psychology [45], economics [127], and behavioral ecology [113] demonstrates

that uncertainty plays a critical role in guiding behavior across species and contexts. Animals foraging in the wild face uncertainty about food availability, predator presence, and weather conditions. Humans making financial decisions confront uncertainty about market fluctuations, job security, and future needs. In each case, the agent must act without complete knowledge of the world's true state.

One particularly important form of uncertainty is *epistemic uncertainty*. The term comes from the Greek episteme, meaning knowledge, and refers to uncertainty that arises from an agent's lack of information about a phenomenon. Unlike the inherent randomness that characterizes some environmental processes, epistemic uncertainty can, in principle, be reduced through the accumulation of evidence. Ann's uncertainty about a restaurant's quality is epistemic; she could reduce it by reading reviews, asking locals, or simply visiting it multiple times. However, this reduction of uncertainty comes at a cost. Every moment Ann spends gathering information, she loses the opportunity to have a more rewarding experience. This creates a fundamental tension between the desire for accurate knowledge and the practical constraints of time and energy. This tension poses a *time/energy-accuracy tradeoff*.

This tradeoff is not merely a practical inconvenience; it represents one of the most fundamental computational challenges facing any intelligent system. The next section explores the specific computational approaches that agents (whether biological or artificial) can use to approximate optimal behavior when learning under uncertainty.

Algorithmic Approaches to Learning under Uncertainty

The reinforcement learning framework provides a powerful account of how an agent can learn to maximize reward. The central challenge is learning a function that maps each state to a scalar that represents the value of that state ($V(s)$). Once an agent knows the value of each state, making optimal decisions becomes straightforward. He could use the rule in Equation 1.8 to extract a policy from the value function:

$$\pi(s) = \arg\max_a (R(s,a) + \gamma \sum_{s'} P(s' \mid s,a) V(s')). \tag{1.8}$$

Concretely, consider the problem of learning a value function in a new, unfamiliar environment. Imagine an idealised environment made up of just two options, A and B. The agent must learn the value of each option to behave optimally in the environment. If the rewards are deterministic, the problem is trivial: the agent can simply try both options once, and subsequently always choose the one with highest reward. The situation becomes more complex when rewards are stochastic. In this situation, the solution would still be trivial if the agent knew the true expected value of each option, she could similarly always choose the

option with highest expected reward. In reality, however, the agent doesn't know the true expected value of each option, and the *RL* problem becomes how to learn these expected values.

To start, one might consider the simplest approach of sampling every option randomly with equal probability. This would clearly lead to suboptimal behavior: the agent should ideally select only the option with the highest expected reward, thus random sampling would lead to making the wrong choice half the time on average in this environment. Clearly, it is not a very good policy and it should be possible to identify a more effective strategy. Determining what such a strategy should look like lies at the heart of the *explore-exploit dilemma* in *RL*.

Employing a random policy is simple but not effective. Can we instead compute an optimal policy to solve the *explore-exploit dilemma*? Solving the Bellman equations (Eq. 1.3) would lead to optimal behaviour, however, we would need to know both the transition function $T(s'|s,a)$ and the reward function $R(s,a)$ to compute the value of each action. In our simple two-option scenario, the transition function is simply 1 (the probability of being presented with the same choice problem after any action is simply 1). However, the reward function poses a problem: the ground truth expected reward of each action is precisely what we are trying to learn. One might consider defining the reward function in terms of the agent's current Bayesian posterior beliefs about expected rewards, combined with the value of information that would be gained from each action. This approach would theoretically yield the optimal exploration strategy, but it comes with a critical limitation: optimal exploration is computationally intractable in all but the simplest cases. Optimal exploration requires combining immediate rewards with the value of information for each action, which necessitates thinking through all possible future action sequences and calculating how much future rewards could increase if more knowledge were collected. The value of information depends on how that information affects the agent's later choices, but those later choices may themselves result in new information, creating a recursive dependency. An optimal agent would need to consider the full "policy tree" describing all possible future trajectories, and because this tree grows exponentially with the planning horizon, it cannot be computed efficiently [105].

Fortunately, there is a way to handle problems where both the transition and reward functions are unknown, and the TD learning algorithm introduced in the previous section is one of them. TD learning can approximate value functions even with stochastic rewards by learning from experience through bootstrapping. Crucially, the quality of the approximation of the true value function depends on the policy used to generate that experience. For this reason, if we were to always select the action with the currently highest estimated value (*greedy* policy), we may get trapped in a suboptimal policy based on early, poten-

tially misleading experiences. This is problematic because TD learning can only learn about the value of actions that are actually tried. If the greedy policy causes the agent to stop exploring certain actions after a few unlucky early trials, those actions' values will never be properly estimated, regardless of how well the TD algorithm itself works.

A practical solution to this problem is to use heuristics that incentivize exploration during learning. A simple approach lies somewhere between the completely random policy mentioned earlier and the purely greedy policy. This strategy is known as the ε-greedy algorithm. The ε-greedy algorithm chooses the action with the highest estimated value most of the time, but with probability ε it selects a random action instead. This represents a reasonable compromise between exploration and exploitation, allowing the agent to explore the environment and reduce the risk of getting stuck in suboptimal policies, while still exploiting its current knowledge most of the time.

However, ε-greedy exploration does not take into account the uncertainty associated with different actions and treats all unexplored or poorly-explored actions equally. A more sophisticated approach uses *optimistic* action selection, embodied in algorithms like Upper Confidence Bound (UCB). The UCB algorithm augments each action's estimated value with an "exploration bonus" that is proportional to the uncertainty of that estimate. Actions that have been tried less frequently receive larger bonuses, making them more likely to be selected [105]. This approach follows the principle of "optimism in the face of uncertainty", which assumes that the uncertain value of an action might be better than current estimates suggest. UCB thus provides a more principled way to balance exploration and exploitation by directing exploration toward actions where more information could be most valuable.

While this theoretical framework provides a normative standard for optimal behavior under uncertainty, it raises a crucial question: How can such optimal strategies actually be implemented in the brain?

Neural Basis

The algorithmic approaches to learning under uncertainty described in the previous section, from simple ε-greedy to sophisticated UCB strategies, provide computational accounts of how agents should balance exploration and exploitation. Yet a fundamental question remains: How does the brain actually implement these computations? This is particularly challenging when we consider that strategies like UCB require the brain to maintain and update separate representations of expected value and uncertainty for each action, and then integrate these quantities to guide decision-making.

Consider the UCB algorithm's core computation: $a_t = \arg\max_k [Q_t(k) + U_t(k)]$, where $Q_t(k)$ represents the expected value of action k and $U_t(k)$ represents an uncertainty bonus.

For the brain to implement such a strategy, neural circuits must somehow compute both terms. The previous section established that dopaminergic neurons encode reward prediction errors (RPEs) that drive learning of action values through temporal-difference mechanisms. These RPE signals, transmitted from the VTA and SNc to the striatum, enable the gradual acquisition of accurate value estimates $Q_t(k)$ through synaptic plasticity. But where and how the brain might compute and represent the uncertainty associated with different actions is a hotly debated question.

Recent theoretical work has proposed an elegant solution to this challenge, suggesting that the basal ganglia, the same circuit that, receiving inputs from the VTA and SNc, is involved in value learning, might also encode and utilize uncertainty information [90, 97, 131]. The key insight comes from recognizing that the basal ganglia contain two major pathways with opposing effects on action selection, the direct pathway, which expresses D1 receptors (with synaptic weights G) facilitates actions, and the indirect pathway, which expresses D2 receptors (weights N) suppresses them. Through differential learning rules where positive prediction errors primarily strengthen G and negative errors strengthen N, these pathways naturally come to encode both expected value $Q = (G - N)/2$ and reward variability $S = (G + N)/2$. In this model, S can be explicitly used as an uncertainty bonus for exploration, with dopamine providing a novelty signal that decays with experience, effectively implementing the epistemic uncertainty $U(k)$ required for UCB-like strategies. This provides a biologically plausible mechanism whereby the same circuits that learn values through RPEs can also track uncertainty and use it to guide exploration, with the UCB computation $a_t = \arg\max_k[Q_t(k) + U_t(k)]$ approximated as selecting actions based on $(G - N)/2 + \theta\mathring{u}(G + N)/2 \times \text{novelty}$, where θ is modulated by motivational state via tonic dopamine levels [131].

Compelling empirical evidence supports the role of dopamine in uncertainty encoding. Using cell-type-specific optogenetics in rats performing a risky decision-making task, Zalocusky and colleagues (2016) [143] demonstrated that D2-expressing neurons in the nucleus accumbens causally control risk preference. These neurons showed elevated activity following loss outcomes, effectively encoding the negative prediction errors that would increase N in the model. Crucially, optogenetic stimulation of D2 neurons during decision-making converted risk-seeking rats into risk-averse animals with single-trial precision. Additional pharmacological evidence shows that D2 receptor agonists like pramipexole, which reduce the effective influence of the indirect pathway, increase risk-seeking behavior in both humans and other animals [31, 143].

While the basal ganglia model provides an elegant computational account of how uncertainty might be computed and utilized for exploration, the neural mechanisms underlying uncertainty-guided behavior are likely far more intricate. Neuroimaging studies consis-

tently reveal that decisions involving uncertainty activate a distributed network of brain regions extending well beyond the striatum [138]. Among these, the anterior cingulate cortex and anterior insula emerge as particularly reliable correlates of uncertainty processing and exploration-exploitation decisions across diverse experimental paradigms. Seminal work by Preuschoff and colleagues demonstrated a functional dissociation between striatal and insular processing: while the striatum encodes reward prediction errors, the anterior insula specifically represents both risk (variance in expected outcomes) and risk prediction errors [99, 100]. These regions, with their rich interconnections to both limbic and prefrontal areas, appear to play crucial roles in monitoring environmental volatility, evaluating action outcomes, and orchestrating adaptive responses to uncertainty.

The anterior insula appears to exert a direct influence on uncertainty-driven exploration through its anatomical and functional connections with the dopaminergic system. Recent evidence reveals that the anterior insula sends direct glutamatergic projections to the ventral tegmental area (VTA), where it modulates both local dopamine release and the activity of dopaminergic neurons projecting throughout the brain [57]. This top-down circuit provides a potential mechanism through which cortical computations of uncertainty could directly influence the dopaminergic signals that drive exploration.

Converging evidence from animal models and human lesion studies establishes that the anterior insula is necessary for computing and utilizing uncertainty signals that guide exploratory behavior. Inactivation of the insula in rats reduces preference for risky options while leaving risk-free choices intact [63], and patients with focal insula lesions lose gambling-related cognitive distortions that depend on uncertainty processing [23]. Furthermore, suppressing insular activity normalizes risk-taking in animals with pathologically elevated risk preference, while activating it increases risk-seeking in normal animals [93]. Together, these findings demonstrate that the anterior insula serves as a critical computational hub where uncertainty signals are represented and integrated into the valuation processes that drive exploration.

The anterior insula, however, does not operate in isolation. Resting-state functional connectivity analyses reveal that it forms part of a broader salience network with particularly strong connections to the anterior cingulate cortex [19]. This tight functional coupling between the anterior insula and ACC suggests they work in concert to process uncertainty, with each region contributing distinct computational components.

While the anterior insula computes uncertainty and biases exploration through its influence on dopaminergic circuits, the ACC serves a distinct computational role in regulating information-seeking and strategy switching. Recent evidence suggests that structured representations of task models and environmental contingencies are maintained in more anterior and ventral regions of prefrontal cortex, including ventromedial PFC and

orbitofrontal cortex [25, 81, 7, 126]. The ACC's role appears to be in monitoring the reliability of these model-based representations and controlling when to exploit them versus when to seek new information. The ACC continuously tracks environmental volatility and prediction errors [114, 60], signaling when evidence against current models accumulates. Rather than storing models themselves, the ACC computes decision-relevant quantities including the value of switching to alternative strategies, the average value of the broader environment, and the expected value of gathering additional information [83, 5]. Consistent with this information-seeking function, ACC activity ramps up before exploratory actions that might resolve uncertainty about environmental structure [133]. This suggests the ACC acts as a metacognitive controller that determines when exploitation of learned models is warranted and when information-seeking exploration is needed.

The mechanisms through which the ACC controls this explore-exploit balance appear to involve critical interactions with neuromodulatory systems, particularly the locus coeruleus-noradrenaline (LC-NE) system. Converging evidence from this lab demonstrates that the ACC plays an active role in controlling exploratory behavior. Early work showed that ACC activity is necessary for adaptive behavioral flexibility in response to changing contingencies [71], while more recent findings establish that ACC circuits actively control exploratory variability in decision-making [119]. The 2014 study revealed a critical mechanism: noradrenergic input from the locus coeruleus to the ACC modulates the balance between exploitation and exploration, with enhanced LC-NE input promoting behavioral variability and exploratory choice strategies [120]. This neuromodulatory gating aligns with theoretical proposals that norepinephrine signals unexpected uncertainty, the degree to which current observations violate model predictions, thereby triggering shifts from exploitation to exploration [141]. Taken together, these findings suggest that the LC-ACC circuit actively implements exploratory behavior: when volatility is high and the environment becomes unpredictable, increased noradrenergic signaling to ACC promotes the flexible, variable exploration needed to discover new environmental contingencies. The ACC is thus not simply a substrate for exploitation that becomes unnecessary during exploration; rather, the ACC-LC circuit provides a neurobiological mechanism for actively controlling the dynamic balance between exploiting known rewards and exploring alternatives based on environmental uncertainty.

The neural architecture of uncertainty-guided learning extends far beyond the circuits described here. Systematic reviews reveal that exploration-exploitation decisions engage distributed networks spanning frontopolar cortex, intraparietal sulcus, precuneus, and broader frontoparietal control systems, while exploitation recruits default network regions including angular gyrus and hippocampus [138]. However, the computational framework outlined in this section, from basal ganglia uncertainty encoding through cortical monitor-

ing and neuromodulatory control, provides a coherent account of how the brain might implement the core algorithmic components required for adaptive learning under uncertainty. The interplay between dopaminergic value learning, insular uncertainty computation, ACC model monitoring, and noradrenergic control suggests a multilevel architecture where uncertainty information flows from subcortical learning circuits to cortical control systems, enabling the flexible behavioral adaptations that characterize intelligent action in an uncertain world.

Hybrid modelling approaches to understanding information seeking

The preceding sections have outlined how uncertainty shapes learning and decision-making, from the algorithmic approaches that balance exploration and exploitation to the neural circuits that implement these computations. While significant progress has been made in understanding these mechanisms, a fundamental methodological challenge remains. Traditional approaches in cognitive neuroscience formalize hypotheses as mathematical models with fixed functional forms that offer clear interpretability but may oversimplify the complex computations underlying intelligent behavior. Conversely, recent data-driven approaches using artificial neural networks can capture complex, non-linear relationships and learn optimal strategies directly from data, but their distributed representations and high-dimensional parameter spaces often render them opaque and difficult to interpret. This trade-off between interpretability and expressiveness has limited our ability to discover novel computational principles that accurately describe how biological systems navigate uncertainty.

In this dissertation, I present a methodological innovation that bridges this gap: a hybrid modeling approach that combines the flexibility of artificial neural networks with the interpretability of mathematical models. The approach integrates data-driven components (ANNs) within established cognitive frameworks, allowing these networks to learn complex mappings between environmental states and behavioral outputs without imposing strong a priori constraints on their functional form. Crucially, we then apply symbolic regression to extract compact, interpretable mathematical equations from the trained networks, transforming opaque neural network computations into transparent symbolic representations. This two-stage process enables us to discover novel computational principles directly from data while maintaining the theoretical clarity essential for scientific understanding.

Chapter 2 of this dissertation presents the theoretical foundation and technical implementation of this hybrid modeling framework. I detail how artificial neural networks can be embedded within cognitive models to learn specific computational components—such as the value of information under uncertainty—while preserving the overall interpretability of the model architecture. I then demonstrate how symbolic regression techniques can distill

the complex functions learned by these networks into human-readable mathematical equations with just a few parameters. This methodological contribution extends beyond mere technical innovation; it offers a principled way to navigate the exploration-exploitation trade-off in scientific discovery itself, balancing the exploitation of established theories with the exploration of novel computational hypotheses.

Chapter 3 applies this framework to understand how humans solve a fundamental problem in learning under uncertainty: determining when to gather more information versus when to commit to a decision. We present data from an information sampling task where participants navigated these decisions while undergoing ultra-high field (7T) functional magnetic resonance imaging.

This work makes three primary contributions to our understanding of learning under uncertainty. First, it provides a new methodological framework that enables data-driven discovery of interpretable computational principles, offering a path beyond the traditional dichotomy between theory-driven and data-driven approaches. Second, it reveals specific computational mechanisms by which humans evaluate information under uncertainty, captured in a compact mathematical form that generalizes across different decision contexts. Third, it identifies the neural substrates of these computations, showing how different brain regions contribute to distinct aspects of information valuation and uncertainty-guided decision-making. Together, these contributions advance our understanding of how intelligent systems—both biological and artificial—can effectively navigate the fundamental challenge of learning in an uncertain world.

1.2.4 Structured World

In the previous section we explored how intelligent agents learn under uncertainty. Now we turn to the second fundamental feature of the world that shapes intelligence: its *structure*. The world presents patterns, regularities, and hierarchical organizations that can be reused. Understanding how biological and artificial systems accomplish this transformation of structured experience into abstract, generalizable knowledge is fundamental to understanding intelligence.

The Challenge of Continuous Experience

Humans and other animals face the challenge of navigating a world that arrives as a continuous, high-dimensional stream of sensory input. Every moment brings visual, auditory, tactile, and other sensory information that arrives without obvious boundaries or labels. For our alien Ann, her first restaurant experience unfolds as an uninterrupted sequence of perceptions.

This continuous nature of experience poses two fundamental computational challenges. First, how can an agent parse this flow into discrete, meaningful units? Second, how can an agent recognize superficially different units that actually belong to the same category? Without solutions to these challenges, every moment would be entirely novel, requiring complete relearning of appropriate responses.

The magnitude of this challenge becomes clear when we consider the dimensionality of the input space. Even a simple visual scene contains millions of pixel intensities, each varying continuously over time. The number of possible configurations is astronomical, yet somehow biological agents routinely extract meaningful patterns from this complexity. They segment the continuous flow into discrete events, abstract away irrelevant details, and discover the underlying structure that enables prediction and generalization.

Event Segmentation: Discovering Boundaries in Experience

The first step in transforming continuous experience into structured knowledge is *event segmentation*: the process of identifying meaningful boundaries in the ongoing stream of experience [41]. Returning to Ann's restaurant experience, although her sensory input arrives continuously, she must learn to parse it into discrete events such as being seated, examining the menu, placing an order, receiving food, eating, and paying the bill. These segmentation boundaries are not arbitrary; they reflect genuine changes in the underlying dynamics of the situation.

One approach to detect these boundaries is to use prediction error. According to this approach, agents use the representation of the current event to predict what will happen next [41]. When these predictions are violated, a prediction error signals the occurrence of an event boundary. This prediction-error-driven segmentation connects naturally to the reinforcement learning framework presented earlier. Events can be conceptualized as sequences of states with coherent transition dynamics. Within an event, state transitions follow relatively predictable patterns governed by the event's internal logic. Event boundaries occur when these dynamics change, that is, when the rules governing state transitions shift to those of a new event type.

The Power of Abstraction: From Episodes to Schemas

Once discrete events have been identified, the next challenge is abstraction. This process transforms concrete experiences into abstract knowledge structures that can generalize across contexts.

The capacity for abstraction is precisely what Ann needs to generalize from her specific restaurant experiences. Experimental evidence in humans shows that abstract schemas are

used to guide the encoding and retrieval of specific details. In an experiment by Anderson et al. [1], participants read stories about either dining at a restaurant or shopping at a supermarket. Crucially, the same food items appeared in both stories, but only in the restaurant context did these items fit into the slots of a pre-existing schema. Participants who read the restaurant story showed superior recall for food items that matched their restaurant schema.

For Ann, abstraction involves recognizing that while each restaurant visit contains unique episodic details, there exists an underlying structure that transcends these specifics. She must learn to preserve the essential relational information while discarding the incidental surface features. This abstraction process can be seen as a form of dimensionality reduction: compressing the high-dimensional space of possible experiences into a lower-dimensional representation that captures the essential structure.

Schema Structure and Hierarchical Organization

The knowledge structures that emerge from this abstraction process—schemas—are not simple linear sequences but hierarchically organized representations with modular components [6]. A restaurant schema contains nested sub-schemas for ordering (examining options, communicating preferences, confirming choices), eating (using utensils, managing courses, social interactions), and payment (requesting the bill, calculating tips, executing transaction). Each sub-schema can be independently activated and can participate in multiple higher-level schemas.

This hierarchical organization provides computational advantages. Sub-schemas can be recombined in novel ways: the "ordering" component applies whether Ann visits an Italian trattoria or a Japanese sushi bar, even though the specific foods and customs differ dramatically. The "payment" component transfers to other commercial transactions. This modularity enables rapid adaptation to new contexts by leveraging previously learned components rather than learning everything from scratch.

From a reinforcement learning perspective, this hierarchical structure can be understood in terms of temporal abstraction and hierarchical policies. Each sub-schema corresponds to a temporally extended action or policy that achieves a particular subgoal. The restaurant schema orchestrates these sub-policies in the appropriate sequence, with each subgoal serving as both the terminal condition for one sub-policy and the initiation condition for the next. This creates a natural hierarchy where high-level goals are decomposed into intermediate subgoals, which are in turn achieved through primitive actions.

The discovery of this hierarchical structure likely involves identifying natural breaking points in the temporal flow of experience, states that serve as subgoals because they represent stable configurations that multiple behavioral sequences converge upon. In the restau-

rant context, "having examined the menu" is a natural subgoal because multiple different sequences of menu-reading behavior all converge on this state, and this state serves as the natural launching point for the ordering sub-schema.

The Adaptive Value of Structured Representations

Why should the brain invest considerable computational resources in building these structured representations? The answer lies in their effect on learning efficiency, generalization capability, and behavioral flexibility.

Structured representations transform the learning problem itself. Rather than learning independent associations between countless specific stimulus-response pairs, agents can learn the underlying generative structure of their environment. This structure captures the causal relationships, temporal dependencies, and conditional probabilities that govern how events unfold. Once learned, this structure can be instantiated with new specific content, dramatically reducing the amount of learning required in novel situations.

Real-world environments present far too much variation for agents to learn specific responses to every possible situation. Instead, successful agents must discover the underlying structure that generates this surface variation, enabling them to predict and respond appropriately even to situations they have never encountered before.

Algorithmic Approaches to Learning the Structure of the World

In the previous section, we explored how event segmentation, abstraction, and hierarchical organization are fundamental capabilities of intelligent agents because they transform the intractable problem of learning in high-dimensional, continuous spaces into manageable subproblems with reusable components.

In this section, I will present three computational approaches that enable machines to achieve state abstraction. First, I will discuss compression-based methods that discover essential features by minimizing representational complexity. Second, I will explore meta-learning approaches that extract shared patterns across multiple related problems. Third, I will consider brain-inspired scaffolding techniques that leverage pre-structured representations to support learning across tasks. Each approach offers a different lens on how structured knowledge can be discovered and exploited to enable generalization.

Abstraction as Compression State abstraction can be understood as a compression problem. From this perspective, intelligent systems discover essential features of their environment by learning to represent experiences using minimal information while preserving the details necessary for effective action. This approach to abstraction rests on the insight that essential features for behavior tend to be those that can be used to reconstruct

or predict important aspects of experience with minimal information. Features that are incidental (e.g. specific colors, textures, or irrelevant spatial arrangements) require additional representational resources without contributing to predictive accuracy on a generalization task. An optimal learning system should therefore evolve representations that capture the maximum amount of task-relevant information using the minimum representational complexity. This principle can be formalized using information theory. Consider the relationship between sensory observations S and their internal representations Z. The mutual information $I(S;Z)$ quantifies how much information the representation Z contains about the sensory input S. A representation that perfectly preserves all details of the input would maximize $I(S;Z)$, but such representations offer no compression and provide no basis for generalization. Conversely, representations that compress the input too aggressively may lose critical information needed for effective behavior. The optimal solution lies in finding representations that balance the two competing objectives. This can be expressed as an optimization problem that seeks to maximize reward while constraining the complexity of the representations used to achieve that reward. Formally, this might take the form of maximizing $\mathbb{E}[R] - \lambda I(S;Z)$, where $\mathbb{E}[R]$ represents expected reward, $I(S;Z)$ represents representational complexity, and λ controls the trade-off between accuracy and compression.

Compressed representations naturally support generalization because they can identify the invariant features that define a category. When Ann learns that the spatial layout of tables and the sequence of ordering, eating, and paying are the essential features of restaurant experiences, she has effectively compressed her representation by discarding irrelevant details like specific décor, particular menu items, or individual waitstaff. This compressed representation then enables her to navigate novel restaurants by focusing on the preserved essential features.

Evidence for compression-based abstraction comes from multiple sources. Recent work on human learning shows that people follow efficient coding principles when learning to generalize, implementing information-theoretic approaches to abstraction [37].

Complementary evidence comes from computational models of memory consolidation, where generative models trained on episodic experience naturally extract schemas that compress common patterns across events [111]. In these models, the consolidation process can be understood as training generative networks to compress episodic memories by separating invariant, schema-based components from unpredictable, episode-specific details. The resulting compressed representations support both efficient storage and flexible reconstruction.

Another demonstration of compression-based abstraction comes from model-based reinforcement learning systems that learn compressed world models. A recent model called

DreamerV3 achieves remarkable generalization across diverse control domains by learning compact representations that capture the structure of the environment while discarding domain-specific details [51]. The system compresses high-dimensional sensory streams into low-dimensional latent representations that preserve the information necessary for prediction and control, enabling the same algorithm with fixed hyperparameters to master tasks as diverse as Atari games, robotics control, and complex open-world environments like Minecraft.

Abstraction through Meta-Learning A complementary approach to abstraction emerges from the meta-learning perspective. Meta-learning systems extract patterns that enable rapid adaptation to new but related problems. Rather than learning how to solve particular problems, the meta-learning approach learns how to learn efficiently within task families. This creates a form of algorithmic abstraction where the learned procedures embody general principles for rapid adaptation rather than specific solutions [129, 4].

Consider how Ann might approach learning about different social institutions after mastering restaurant etiquette. Rather than learning each new institution (offices, hospitals, schools) from scratch, she could extract meta-level patterns about how social hierarchies work, how to identify authority figures, and how to infer appropriate behaviors from environmental cues. These algorithmic insights would then enable her to rapidly adapt to new institutions by deploying the right learning strategies rather than starting with no knowledge.

This approach naturally creates a two-level learning system: a slow outer loop that extracts general learning strategies from experience across multiple tasks, and a fast inner loop that applies these strategies to adapt rapidly to specific new tasks. The abstraction emerges in the outer loop, which discovers invariant algorithmic patterns that prove effective across the task distribution. These patterns encode inductive biases about how to explore, what features to attend to, and how to generalize from limited experience [4].

Evidence for meta-learning approaches comes from systems that learn task-specific learning algorithms through experience with structured task distributions. Meta-RL systems develop adaptive behaviors that can exploit structural regularities such as correlated rewards or hierarchical relationships, automatically discovering exploration strategies and learning rates optimized for specific problem structures [129]. These learned algorithms often differ dramatically from the training procedures used to discover them, demonstrating that meta-learning can discover novel algorithmic approaches tailored to the statistics of particular task families [4].

This algorithmic form of abstraction proves particularly valuable in domains where the surface features may vary dramatically but the underlying learning challenges remain con-

sistent across task instances.

Abstraction through Scaffolding A third approach to abstraction draws inspiration directly from neural circuits in the brain, particularly the hippocampus-entorhinal system. This perspective suggests that well-organized neural coordinate systems, evolved for one purpose, can be repurposed to support learning in seemingly unrelated areas.

One foundation of this approach lies in the discovery of grid cells in the entorhinal cortex, which fire in distinctive hexagonal patterns as animals navigate through space. These cells create a universal coordinate system that represents spatial location independent of environmental specifics. For instance, the same grid pattern emerges whether an animal explores a square box or a circular arena. This spatial coordinate system provides a structured, low-dimensional representation that captures essential relational information while abstracting away irrelevant environmental details.

The key insight for abstraction is that these grid cell representations can be repurposed beyond spatial navigation. Just as Ann learns that spatial concepts like "sequence" and "adjacency" apply not only to restaurant table arrangements but also to the temporal order of dining activities and social hierarchies, grid cell coordinate systems can scaffold learning across diverse cognitive domains. The pre-structured nature of these representations means that new learning doesn't start from scratch but can immediately leverage existing organizational frameworks.

Computational models demonstrate how this scaffolding principle enables remarkable abstraction capabilities. Systems that implement grid cell-like representations can support both spatial navigation and abstract relational reasoning using the same underlying computational principles, showing that spatial and non-spatial memory may share common structural foundations [135]. Other models reveal how grid-like structured representations can scaffold rapid learning through episodic memory binding. These models use grid cell representations to create abstract maps, then bind sensory experiences to outcomes through backward credit assignment, enabling one-shot learning by linking rewards to the states that caused them [20].

Scaffolding approaches reveal a unique perspective on abstraction: rather than discovering structure through experience or learning algorithmic patterns across tasks, intelligent systems can achieve abstraction by exploiting well-structured representations that evolution has already provided. This suggests that spatial intelligence, with its requirement for organized coordinate systems and relational reasoning, may serve as a foundational scaffold that enables the flexible, generalizable cognition observed in biological intelligence.

Neural Basis

While computational approaches demonstrate how abstraction can emerge through compression, meta-learning, and scaffolding, here I will explore what the brain actually does to achieve abstraction.

From Behavior to Brain: The Neural Basis of Learning Sets The search for neural mechanisms of abstraction begins with one of the most compelling demonstrations of learning abstraction in animals. Harlow's classic studies of "learning sets" in monkeys established that abstraction develops progressively through experience with structured environments [55]. In these experiments, monkeys were presented with hundreds of discrimination problems, each involving a choice between two objects. Initially, each new problem required extensive trial-and-error learning. However, as monkeys gained experience with the general structure of discrimination tasks, they developed what Harlow termed "learning sets", which is the ability to learn how to learn efficiently in similar situations. Eventually, monkeys could solve new discrimination problems in just one or two trials, having extracted the abstract principle that one object would consistently be rewarded while the other would not.

Recent recordings from rodent prefrontal cortex and hippocampus reveal the neural substrates of these learning sets, showing how abstract and problem-specific representations work together to enable transfer [102].

This result suggests that the brain does not simply accumulate specific associations but discovers the underlying structure that generates the observed instantiations. The progression observed in Harlow's experiments suggests that neural mechanisms must support both the gradual acquisition of abstract knowledge and its flexible deployment in novel contexts.

Representational Geometry Recent advances in computational neuroscience have suggested that what distinguishes abstract neural representations from non-abstract ones lies in how populations of neurons organize their activity in neural space. The geometry of these population representations determines whether and how effectively information can generalize across contexts [22].

Abstract representations have a distinctive geometric signature, where different task variables are encoded along orthogonal dimensions in neural activity space. This geometric property enables linear decoders trained on one set of conditions to generalize seamlessly to new conditions, providing a neural basis for the transfer of knowledge across contexts. Such representations have been observed across species and brain regions.

Human studies provide compelling evidence for the emergence of abstract representational geometry during learning. Single-unit recordings from the hippocampus of patients learning an inference task reveal that abstract representations develop only when patients can successfully perform inference [26]. Context and stimulus variables become orthogonally represented in neural space when behavioral performance indicates generalization.

This relationship between representational geometry and behavior extends beyond humans. Recordings from monkey prefrontal cortex during rule-based tasks show that different animals can employ distinct representational geometries that reflect their underlying behavioral strategies [38].

In the mouse hippocampus, representations of social encounters demonstrate task-dependent geometry, where novel individuals are encoded in low-dimensional representations that support generalization across contexts, while familiar individuals are represented in high-dimensional format that maximizes memory storage capacity [8]. This adaptive geometry enables the same neural system to flexibly balance between the competing demands of generalization and discrimination.

These findings indicate that the geometry of these representations provides a neural measure of abstraction.

Temporal Abstraction: From Events to Schemas The neural mechanisms of abstraction extend beyond spatial representations to encompass the temporal dimension, where the brain must extract structured patterns from sequential experience. This temporal abstraction operates through hierarchical organization that mirrors the computational principles observed in spatial domains.

During narrative comprehension, the human brain exhibits nested event boundaries that span multiple timescales, from seconds in sensory areas to minutes in higher-order regions [2]. This hierarchical temporal structure demonstrates how abstraction emerges through the progressive integration of local events into global schemas. Just as spatial representations become increasingly abstract at higher levels of cortical hierarchy, temporal processing reveals a similar gradient where immediate sensory changes are integrated into longer-term narrative structures.

This principle of hierarchical temporal abstraction finds cellular expression in recent recordings from the medial frontal cortex of mice, where neurons organize into ”structured memory buffers” that tile progress toward goals [35]. These goal-progress cells create neural maps of behavioral sequences, forming the cellular basis for temporal schema learning. Crucially, when animals encounter tasks with shared temporal structure, these abstract temporal representations enable rapid transfer of knowledge, demonstrating how the brain reuses temporal schemas while adapting to context-specific requirements.

Grid Codes: Universal Computational Scaffolds The convergence of evidence across learning sets, representational geometry, and temporal abstraction raises a fundamental question: what neural architecture might enable such diverse forms of abstraction? One intriguing possibility lies in grid-like activity patterns that emerge across species and cognitive domains, suggesting a potentially universal computational motif for structured representation.

Grid cells in the entorhinal cortex of rodents fire in hexagonal patterns as animals navigate space, creating a coordinate system that scaffolds spatial cognition [52]. Yet this hexagonal organization transcends spatial navigation. In humans, grid-like patterns emerge during abstract reasoning tasks, organizing conceptual knowledge along dimensions of similarity [25]. When participants learn relationships between abstract stimuli varying along two dimensions, fMRI reveals hexagonal firing patterns in entorhinal cortex identical to those observed during spatial navigation.

This principle extends to novel inference in primates. When macaques encounter previously unseen options composed of familiar features, medial frontal cortex displays grid-like activity patterns that enable rapid value integration [7]. Disruption of this region specifically impairs the ability to integrate reward dimensions for novel choices, demonstrating the causal role of grid codes in abstraction. Similarly, during active choice-making, ventromedial prefrontal cortex constructs cognitive maps of value space using grid-like codes that emerge at the moment of decision [126].

Altogether, these findings reveal how neural circuits implement the compression, scaffolding, and meta-learning principles identified computationally. Grid codes compress high-dimensional experience into structured low-dimensional manifolds, provide scaffolding through their metric properties, and enable meta-learning by offering a reusable computational framework. The structured world that humans and animals inhabit is thus not merely perceived but actively constructed through neural mechanisms that discover and exploit the underlying patterns that make learning and generalization possible.

2

Hybrid Models

Combine cognitive models with artificial neural networks to discover new functions, and use symbolic regression to recover compact, interpretable models of cognition

2.1 Introduction

A common approach in cognitive neuroscience involves formulating hypotheses about the computations that drive specific behaviors and formalizing these through mathematical models. These models serve as explicit computational representations of cognitive processes, allowing researchers to generate quantitative predictions about behavioral and neural data. This approach has been tremendously successful in advancing our understanding of cognition [43, 106, 117]. The strength of these mathematical models lies in their transparency: each parameter has a clear psychological interpretation, the computations are explicit, and the predictions are directly traceable to explicit theoretical assumptions.

However, this transparency comes at a cost. Mathematical models with fixed functional forms may oversimplify the complex computations underlying intelligent behavior. When we specify that the value of information follows a UCB function [53, 54], or that evidence accumulation follows a drift-diffusion process [106, 54, 48], we impose strong constraints on the space of possible computations. While these constraints make models tractable and interpretable, they may prevent us from discovering the true computational principles that biological systems employ.

An alternative approach is to use data-driven models, such as artificial neural networks (ANNs) to model behavior. ANNs can learn complex mappings from data, vastly expanding the space of candidate functions that can be considered. The universal function approximation theorem underpins this method, asserting that sufficiently deep neural net-

works can model any continuous function to arbitrary precision, given sufficient training data. This capability allows us to learn directly from data how individuals might compute complex cognitive functions, without needing to specify the exact form of the function in advance. ANNs have proven in the recent years to be incredibly successful at solving prediction problems in a wide range of domains, from predicting the structure of a protein from its DNA sequence [69] to predicting the next word in a sentence [14].

Yet this expressive power comes with its own limitations. While ANNs may yield more accurate predictions, they typically lack interpretability. The learned representations are distributed across many parameters, encoded in high-dimensional spaces that are difficult to understand. When an ANN successfully predicts behavior, we gain predictive power but lose explanatory insight. We know that the network has captured something important about the computation, but we cannot easily extract what that something is. This opacity makes it difficult to generate new hypotheses, to understand individual differences, or to link computational processes to their neural implementations.

In this chapter, we propose a modeling approach that goes beyond this traditional trade-off. Our approach integrates two key steps. First, we develop a hybrid model that combines knowledge-driven and data-driven components [33]. The knowledge-driven component incorporates established cognitive principles, such as the mechanisms underlying attention, memory, and action selection [82, 106], providing structure and constraint based on prior scientific knowledge. The data-driven component utilizes ANNs to model aspects of cognition that are difficult to define a priori, such as how individuals compute the value of information under uncertainty. This hybrid architecture can be more expressive than a fully knowledge-driven approach, while also being more interpretable and data-efficient than a fully data-driven approach.

Second, we apply symbolic regression to the trained ANN component to recover compact, interpretable mathematical expressions [29, 18]. Symbolic regression searches through the space of mathematical functions to find simple equations that approximate the input-output mapping learned by the ANN. This process transforms the neural network into a mathematical function with few interpretable parameters.

We then applied this approach to investigate a fundamental problem in cognitive neuroscience described in the previous chapter: understanding how humans gather and evaluate information when making decisions under uncertainty. Information sampling requires solving a complex optimization problem that balances the value of additional information against its costs. While several studies have shed light on the computational and neural mechanisms underlying value learning and option selection [107, 106], the factors that determine when to stop gathering information and commit to a final decision, as well as which options to sample from, remain an active area of investigation [53, 32, 115, 54].

The remainder of this chapter is organized as follows. Section 2 introduces the information sampling problem and the experimental task we designed to study information sampling behavior. Section 3 presents the hybrid modeling framework, detailing how we integrate cognitive models with artificial neural networks, and explains how we use symbolic regression to extract interpretable functions from trained ANNs. Finally, Section 4 discusses the broader implications of this methodological innovation for cognitive neuroscience and artificial intelligence research.

2.2 The Information Sampling Problem and Experimental Task

Decisions should be based on evidence. Once sufficient evidence has been sampled, the agent can decide which option to select. But in addition to guiding the choice, evidence should also simultaneously be used to evaluate whether there is sufficient information to warrant a decision, or whether further information must be sampled [54, 48, 95, 32]. When sampling information, attentional constraints mean that decision makers typically only focus on one option at a time [84]. There is therefore a further decision, as to whether more evidence should be gathered from the currently attended option or whether attention should be shifted to an alternative. This creates a hierarchy of interconnected decisions that must be made simultaneously: whether to continue sampling or commit to a choice, and if sampling, where to direct attention next.

Gathering more information can improve decision quality, but it also comes at the expense of time, energy, and lost opportunities to engage in other activities. Therefore, knowing when to stop learning from the environment and use the acquired knowledge to make a choice is crucial for effective decision-making. An illustration of this dilemma is found in the 14th-century thought experiment associated with Jean Buridan [144]. In the scenario, an ass stands exactly midway between two identical piles of hay. The piles are so similar to one another that the ass cannot choose between them. The ass contemplates the piles searching for evidence that one is better than the other but, without a mechanism to stop deliberating, the ass fails to make a choice and is left without anything.

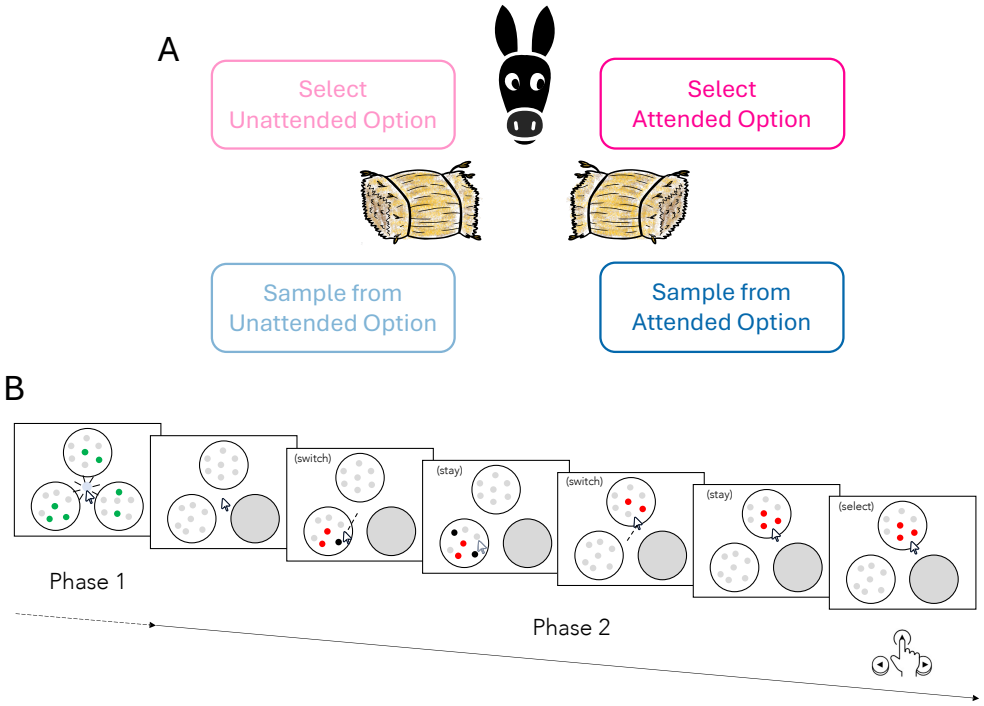

Figure 2.1: Information sampling task. A, The decision-making process involves two key decisions: whether to gather more information or make a selection (top vs. bottom), and if gathering information, whether to sample from the currently attended option or switch to the alternative (top right vs. top left). B, Task structure showing the two phases of the information sampling task. In Phase 1, participants are presented with three patches of dots covered by green or grey covers. After revealing the green-covered dots in each patch, one patch is blocked (grey circle). In Phase 2, participants can freely sample information by hovering over patches, with grey-covered dots revealing their true colors sequentially, before making a final selection.

This thought experiment captures the essence of the explore-exploit dilemma in information sampling. While several studies have shed light on the computational and neural mechanisms underlying value learning and option selection [107, 106], the factors that determine when to stop gathering information and commit to a final decision, as well as which options to sample from, remain an active area of investigation [53, 32, 115, 54].

2.2.1 Task Design: Creating a Controlled Environment for Studying Information Valuation

To study these computational processes, we developed an information-sampling task that allows precise measurement of how people gather and evaluate information under uncertainty (Figure 2.1B). The task was designed with several key principles in mind. First, we separated the time spent sampling from the amount of information gained, allowing us to disentangle temporal costs from informational benefits. Second, we manipulated initial uncertainty independently of the difficulty of the final decision. Third, we created a situation

where participants face time pressure and opportunity costs, making the explore-exploit trade-off meaningful.

Twenty participants (14 females), aged 19 to 32 years, completed the task inside a 7T MRI scanner. On each trial, participants were presented with three patches of moving dots (100 dots per patch). Each dot's true color was either red or black, but initially all dots were hidden under either green or grey covers. The goal was to select the patch with the highest number of red dots. The green covers indicated dots whose colors would be revealed simultaneously upon the participant's first visit to that patch, while grey covers indicated dots that would be revealed sequentially. The number of green-covered dots varied between patches and trials (between 5-30 dots initially visible, with 70-95 dots revealed sequentially), allowing separation of sampling time from patch uncertainty.

2.2.2 Task Structure and Key Manipulations

The sequence of each trial was carefully structured to capture the key decisions in information sampling (Figure 2.1B). First, participants clicked on each patch to reveal the green-covered dots, providing initial information about each option. After viewing all three patches, one patch was randomly blocked and could not be sampled or chosen, leaving two options available for the remainder of the trial. This blocking manipulation allowed us to examine how background uncertainty (e.g., uncertainty about an irrelevant option) affects sampling behavior.

During the main sampling phase, participants could freely sample information by hovering over patches with an MRI-compatible trackball. Upon hovering, they had to wait 2 seconds before any new information appeared, a cost that made switching between patches non-trivial. After the waiting period, the green dot covers were all removed simultaneously to reveal which dots were either red or black. The grey covers were then removed sequentially, one dot every 150ms, gradually revealing the true proportion of red dots in the patch.

This design incorporated several key experimental manipulations:

Initial uncertainty manipulation: By varying the number of green dots (revealed immediately) versus grey dots (revealed sequentially), we could manipulate the initial uncertainty of each option independently of its true value. A patch with fewer green dots would have higher initial uncertainty, requiring more sampling to achieve the same level of confidence.

Background uncertainty: The blocked third option provided a measure of background uncertainty that was task-irrelevant but might still influence sampling behavior. This allowed us to test whether participants could appropriately ignore irrelevant sources of uncertainty.

Time pressure and opportunity costs: Participants performed four sessions, each session had a fixed duration of 25 minutes, and participants were incentivized to maximize correct choices across trials. Spending excessive time on any single trial reduced the number of subsequent trials they could attempt, creating opportunity costs that made the speed-accuracy trade-off meaningful. To incentivize participants to maximize their performance, they were paid, in addition to their base pay of £15 per hour, a performance-dependent bonus that varied between £20 and £40 depending on the number of points collected during the task.

Across-patch correlation in red-dot proportions: Note that the proportions of red dots across patches within a trial were not independent: by design, the pairwise difference between any two patches was constrained to 0.1, 0.2, or 0.3 (each level equally frequent), yielding a Pearson correlation of $r = 0.43$ between patches' red-dot proportions.

2.2.3 The Behavioral Decision Space

At each moment during the sampling phase, participants faced four possible actions:

1. **Stay:** Continue sampling from the currently attended patch, gathering more information about this option.
2. **Switch:** Shift attention to the unattended patch, incurring a switching cost but potentially gaining valuable information about the alternative.
3. **Select attended:** Choose the currently attended patch as the final decision.
4. **Select unattended:** Choose the unattended patch as the final decision.

These four actions allow us to study not just the explore-exploit trade-off (sampling versus selecting) but also the allocation of attention during exploration (staying versus switching). The value of each action depends on multiple factors: the current beliefs about each option's value, the uncertainty associated with those beliefs, the potential for new information to change the decision, and the various costs associated with time and switching.

2.2.4 Gamification and Ecological Validity

To make the task more engaging and ecologically valid, we framed it as a medieval-themed game called "The Last Duel." Participants were told they lived in a hostile world where strangers might challenge them to duels. Their goal was to obtain the best weapon to ensure victory. The three patches represented weapon stores in town, and the proportion of red dots determined which weapon each contained: the patch with the highest proportion contained a sword (100% win rate), the intermediate proportion contained a hammer (50% win rate), and the lowest proportion contained a stone (0% win rate).

This narrative provided an intuitive explanation for why participants should care about finding the option with the most red dots, it made the task more engaging over multiple sessions, and it created a natural context for the time pressure and opportunity costs built into the design.

2.3 The Hybrid Modeling Framework

2.3.1 Overview: A Two-Stage Modeling Approach

To investigate how participants compute the value of information during sampling decisions, we developed a two-stage modeling approach that combines the interpretability of cognitive models with the flexibility of machine learning.

Our framework is structured around two stages. In the first stage, we construct a hybrid model that integrates knowledge-driven components (encoding established cognitive principles) with a data-driven component (an artificial neural network that learns the unknown function of interest).

The hybrid model consists of several interacting modules. Knowledge-driven components handle state representation (transforming objective observations into subjective beliefs), memory dynamics (how unattended information decays over time), and action selection (how values are converted to choice probabilities through a softmax function). The data-driven component, instantiated as a multi-layer perceptron (MLP), takes the current state as input and outputs the value of gathering more information from each patch. These values then flow into the decision architecture to determine which of four actions the model predicts: stay with the current patch, switch to the alternative, or select one of the two options as the final choice.

In the second stage, we apply symbolic regression to the trained ANN to extract interpretable mathematical equations. This transformation is crucial for scientific understanding as it converts thousands of neural network parameters into compact symbolic expressions with just a few interpretable parameters.

2.3.2 Knowledge-Driven Components: Encoding Prior Understanding

The knowledge-driven components of our hybrid model formalize cognitive principles about attention, memory, and decision-making. These components provide the structural scaffolding within which the data-driven ANN operates, ensuring that the model respects known constraints while learning unknown functions Figure 2.2.

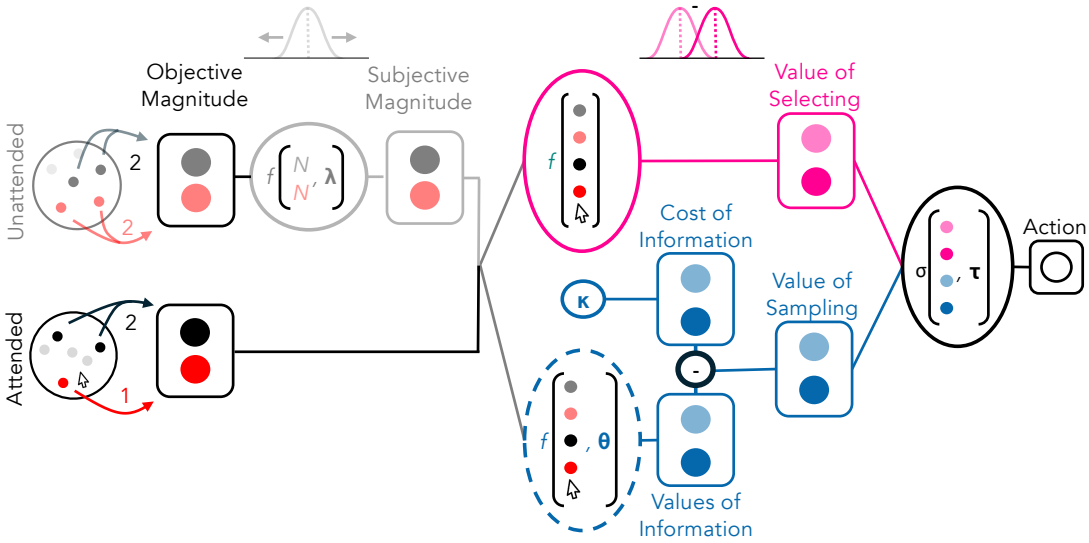

Figure 2.2: Sampling behavior and computational model. Schematic of the computational model. The model transforms objective magnitudes (number and color of dots in each patch) into subjective magnitudes through attentional discounting and memory decay functions. These subjective magnitudes are used to compute the value of selecting each patch (pink) and the value of sampling more information (blue), which together determine the final action. The dotted blue circle highlights the component that computes the value of information, which is implemented using either a linear function, UCB algorithm, or artificial neural network.

State Representation and Transformations

The model transforms objective observations (actual number of revealed dots) into subjective representations that guide decision-making. For the attended patch, subjective magnitudes equal objective ones since participants directly observe the dots. However, for the unattended patch, the transformation depends critically on whether it has been previously visited.

During the first visit to a patch, the model accounts for potential interference from the blocked patch when estimating the unattended patch's contents. The subjective number of dots in the unattended patch is computed as:

$$\hat{n}_u = \omega n_u + (1 - \omega) n_b \tag{2.1}$$

where n_u is the actual number of green dots in the unattended patch, n_b is the number of green dots in the blocked patch, and ω is a parameter quantifying resistance to interference (when $\omega = 1$, there is complete resistance to interference from the blocked patch).

The model then estimates the number of red dots in the unattended patch based on in-

formation from the attended patch:

$$\hat{r}_u = \hat{n}_u \left(0.5 + (\hat{\mu}_a - 0.5)\lambda_0\right) \tag{2.2}$$

$$\hat{\mu}_a = \frac{\hat{r}_a}{\hat{n}_a} \tag{2.3}$$

where λ_0 is a free parameter controlling how strongly the proportion of red dots in the attended patch ($\hat{\mu}_a$) influences expectations about the unattended patch.

Attention and Memory Mechanisms

During subsequent visits to a patch, information about the unattended patch must be maintained in memory rather than being directly sampled. To capture this memory-dependent representation, the model implements an exponential decay of information:

$$\hat{n}_u = n_u e^{-\lambda_2 t} \tag{2.4}$$

$$\hat{r}_u = r_u e^{-\lambda_2 t} \tag{2.5}$$

where t is the time elapsed since the start of the current visit and λ_2 determines the rate of information decay in memory. This decay reflects the deterioration of remembered information when attention is directed elsewhere. Because both the remembered number of red dots ($\hat{r}_u$) and total dots ($\hat{n}_u$) decay at the same rate, this effectively increases uncertainty about the true proportion of red dots in the unattended patch over time. In Bayesian terms, the posterior Beta distribution over the proportion becomes increasingly wide, reflecting growing uncertainty.

The model also incorporates the cost of switching attention between patches. The task imposed a 2-second delay before information could be gathered from a newly attended patch, which we model as a fixed switching cost parameter κ that reduces the value of switching actions. This cost creates a natural tendency to persist with the currently attended option unless the alternative offers substantially more information.

Decision Architecture

The model makes decisions through a softmax action selection mechanism that converts action values into choice probabilities. At each time point, the model computes values for four possible actions.

The value of selecting a patch is determined by comparing the estimated proportion of

red dots between the two patches:

$$Q_{select\_attended} = \hat{\mu}_a - \hat{\mu}_u \tag{2.6}$$

$$Q_{select\_unattended} = \hat{\mu}_u - \hat{\mu}_a \tag{2.7}$$

where $\hat{\mu}_a$ and $\hat{\mu}_u = \hat{r}_u/\hat{n}_u$ represent the estimated proportion of red dots in the attended and unattended patches, respectively.

The value of continuing to sample information depends on whether the participant stays with the currently attended patch or switches to the unattended patch:

$$Q_{sample\_attended} = \text{value\_of\_stay}(\hat{n}_a, \hat{n}_u) \tag{2.8}$$

$$Q_{sample\_unattended} = \text{value\_of\_switch}(\hat{n}_a, \hat{n}_u) - \kappa \tag{2.9}$$

where value_of_stay and value_of_switch are functions that estimate the potential information gain from sampling (learned by the ANN), and κ is the cognitive cost of switching attention between patches.

These four action values are transformed into action probabilities using a softmax function with temperature parameter τ:

$$P(a) = \frac{e^{Q(a)/\tau}}{\sum_{a'} e^{Q(a')/\tau}} \tag{2.10}$$

The temperature parameter controls the randomness in action selection: lower temperatures lead to more deterministic choices (selecting the highest-value action), while higher temperatures increase exploration.

What These Components Capture and What They Cannot

These knowledge-driven components successfully capture several important aspects of information sampling behavior. They model how beliefs are updated through observation, how memory for unattended information degrades, how switching costs influence persistence, and how final decisions emerge from value comparisons. The components are grounded in established psychological principles and use parameters that have clear cognitive interpretations.

However, these components alone cannot explain the full complexity of sampling behavior because they lack a crucial piece: how to compute the value of gathering additional information. The model can track what is known (through state representation), remember what was seen (through memory), and select actions (through softmax), but it cannot determine whether it is worth gathering more information. This is where traditional ap-

proaches would formalize their hypothesis of how to compute the value of gathering additional information, perhaps that information value decreases linearly with samples, or follows an Upper Confidence Bound rule. Instead, we leave this computation unspecified and allow an artificial neural network to learn it from data, as described in the next section.

2.3.3 The Data-Driven Component: Learning the Value of Information

Why ANNs for Value of Information

A critical aspect that makes effective information sampling possible is evaluating the marginal value of acquiring additional information from each option.

We use an artificial neural network (ANN) to learn from the data the mapping between the state variables (such as the number and color of revealed dots) and the value of gathering additional information. This approach, which we refer to as a hybrid model, combines the data-driven nature of the ANN with the theory-driven design of the cognitive model in which the ANN is instantiated. The advantage of this approach is that it gives us the flexibility to capture potentially complex, non-linear relationships between task variables and information value without making strong a priori assumptions about the functional form of this relationship.

Input Features and Their Justification

To identify which task variables were essential for computing the value of information, we performed a systematic feature selection analysis. We first defined a full feature vector containing seven input variables (Figure 2.3):

- Number of dots in the same patch (n-same), i.e. in the patch currently being evaluated
- Number of dots in the other patch (n-other)
- Number of dots in the blocked patch (n-blocked)
- Proportion of red dots in the same patch (ρ-same)
- Proportion of red dots in the other patch (ρ-other)
- Current gaze position (gaze)
- Whether it was the first visit to the patch (visit)

We conducted a leave-one-out analysis where we trained separate models for each participant, comparing the full model against versions with each feature individually removed. This resulted in seven different model variants per participant, all trained using 8-fold

cross-validation. We evaluated the impact of each feature by comparing the validation performance of the reduced models against the full input-feature model.

This analysis revealed that four features were critical: the current gaze position, whether it was the first visit, and the number of pieces of evidence (N) in the attended and unattended patches. Removing any of these features led to significant decreases in model performance across participants Figure 2.3.

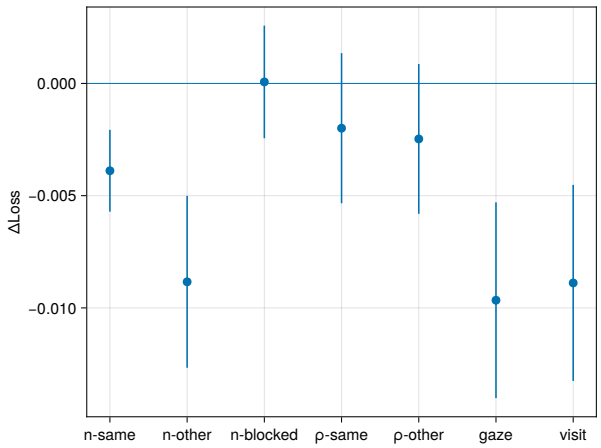

Figure 2.3: Feature importance analysis. Dot plot showing the change in model loss (ΔLoss) when different features are removed from the model. Each blue dot represents the mean loss difference, with vertical blue lines indicating confidence intervals. Features tested include n-same (number of dots in the same patch), n-other (number of dots in the other patch), n-blocked (number of dots in the blocked patch), ρ-same (proportion of red dots in the same patch), ρ-other (proportion of red dots in the other patch), gaze (current gaze position), and visit (whether it's the first visit to a patch).

Network Architecture and Training

The neural network was implemented as a fully connected Lipschitz-Bounded Deep Network (LBDN) [130] with built-in guarantees on the Lipschitz bound. A function f is said to be γ-Lipschitz if $\|f(x_1) - f(x_2)\| \leq \gamma\|x_1 - x_2\|$ for any inputs x_1, x_2, that is, its sensitivity to input perturbations is bounded by γ. We adopted this architecture to ensure that the value of information varies smoothly with evidence, consistent with the expectation that cognitive value signals do not change abruptly between similar states. We used a network with 4 hidden layers of 32 neurons each and tanh activation functions. The network takes as input the normalized counts of revealed dots in each patch ($\hat{n}_{left}/100$, $\hat{n}_{right}/100$), patch gaze indicators (left, right), and a first-visit flag. It outputs a value of information for each patch, representing the expected utility of gathering more evidence from that location.

The training procedure of the hybrid model consists of three main stages. First, we jointly optimized both the neural network and cognitive module parameters using the Adam optimizer, with different learning rates for each component ($\eta_{nn} = 0.01$ for neural network, $\eta_{cm} = 0.02$ for cognitive module). This allowed both components to adapt to each other while learning at appropriate rates for their respective architectures.

Next, we performed a focused optimization of the cognitive parameters using L-BFGS while keeping the neural network fixed, which helped refine the psychological parameters of the model.

Finally, we conducted a fine-tuning phase where both components were again jointly optimized but with smaller learning rates ($\eta_{nn} = 0.001$, $\eta_{cm} = 0.0001$), ensuring stable convergence of the complete model. The model was trained to minimize the cross-entropy loss between predicted and actual choices using backpropagation.

Integration with the Cognitive Model

Once the ANN estimates the value of information for sampling each patch, these values flow directly into the decision architecture described earlier. For the two sampling actions, the value of staying to sample the currently attended patch is its ANN-estimated value of information. The value of switching to sample the unattended patch is its ANN-estimated value of information, reduced by the fitted cost of switching (κ). These ANN outputs are combined with the selection values computed from the subjective proportions, and all four action values are then transformed via the softmax function to yield action probabilities.

This integration ensures that the data-driven component operates within the constraints of established cognitive principles while providing the flexibility to discover novel computational strategies that participants actually employ.

Validation Through Simulation-Recovery

To validate our hybrid modeling approach's ability to recover underlying value-of-information computations, we conducted a series of simulation studies. We generated synthetic data from four different artificial agents, each using a distinct non-linear function to compute their value of information:

$$voi_1(n_A, n_U) = \sigma(0.94^{n_A - n_U}) - 0.2 \tag{2.11}$$

$$voi_2(n_A, n_U) = \sigma(0.94^{n_U - n_A}) - 0.2 \tag{2.12}$$

$$voi_3(n_A, n_U) = \sigma(0.94^{n_A - 37}) - 0.2 \tag{2.13}$$

$$voi_4(n_A, n_U) = \sigma(0.94^{37 - n_U}) - 0.2 \tag{2.14}$$

where σ represents the logistic function, and n_A and n_U represent the number of revealed dots in the attended and unattended patch respectively. These functions were chosen to represent different patterns of information valuation: the first two functions compute relative differences between patches, while the last two compare each patch against a fixed reference point.

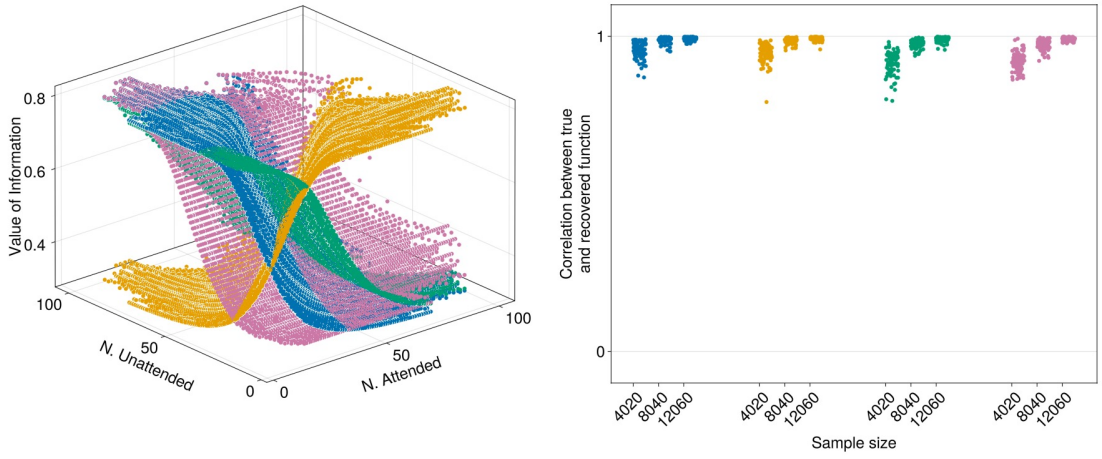

Figure 2.4: Function recovery validation. Left: Three-dimensional scatter plot showing the value of information as a function of the number of dots in the attended (N. Attended) and unattended (N. Unattended) patches. Different colors (blue, green, pink, and orange) represent different simulated functions used to generate the data. Right: Scatter plot showing the correlation between true and recovered functions across different sample sizes (4,000, 8,000, and 12,000 observations). Each color represents a different simulated function, with each dot representing a single recovery attempt.

For each function, we simulated behavioral data using different sample sizes (4,000, 8,000, and 12,000 observations) to assess the robustness of our recovery procedure. We then applied our hybrid modeling approach to these simulated datasets, training the model to recover the underlying value-of-information function from the behavioral data alone. The recovered functions were compared to the true generating functions using Pearson correlation coefficients, showing very high correlations ($r > 0.95$) across all sample sizes Figure 2.4. This validation demonstrates that our hybrid approach can reliably recover complex value-of-information computations from behavioral data, giving us confidence in applying this method to understand human behavior.

2.3.4 Symbolic Regression

The ANN's value-of-information computation is distributed across thousands of parameters in high-dimensional space, making it impossible to extract clear insights about the underlying computational principles. To address this interpretability challenge, we apply

symbolic regression as the second stage of our framework, transforming the neural network into mathematical equations.

The Symbolic Regression Approach

Symbolic regression is a machine learning technique that searches through the space of mathematical expressions to find equations that best fit a given dataset [27]. Unlike traditional regression, which assumes a specific functional form (e.g., linear, polynomial), symbolic regression discovers both the functional form and the parameters simultaneously. This makes it particularly powerful for scientific discovery, as it can reveal mathematical relationships that were not hypothesized a priori.

The symbolic regression process works by evolving mathematical expressions through genetic programming. Starting from a population of randomly generated expressions, the algorithm iteratively applies operations such as mutation (changing individual components of expressions) and crossover (combining parts of different expressions) to create new candidate equations. Each expression is evaluated based on both its accuracy in fitting the data and its complexity, with simpler expressions preferred when accuracy is comparable.

In our framework, symbolic regression operates on the trained ANN's input-output mapping rather than directly on behavioral data. We extract a representative sample of input-output pairs from the trained network, covering the range of states encountered during the task, and then search for mathematical expressions that can approximate this learned function. The key insight is that the ANN has already solved the difficult problem of learning from noisy behavioral data; symbolic regression then distills this learned knowledge into interpretable form.

This approach offers several advantages over traditional curve-fitting approaches. First, it makes no assumptions about the underlying functional form, allowing for the discovery of novel mathematical relationships. Second, it naturally balances accuracy and complexity through multi-objective optimization, preventing overly complicated expressions while maintaining good fit. Third, it can discover expressions using a rich vocabulary of mathematical operations, from simple arithmetic to exponential and logarithmic functions, providing flexibility in the types of relationships that can be discovered.

Why Not Apply Symbolic Regression Directly?

A natural question arises: why not apply symbolic regression directly to behavioral data rather than first fitting an ANN? There are several computational and methodological reasons for our two-stage approach.

First, direct symbolic regression on the complex mapping from task state to behavior

would involve an intractably large search space. The space of possible functions mapping from multi-dimensional state representations (attended evidence, unattended evidence, gaze position, visit history) to behavioral choices is enormous. By first learning this mapping with ANNs and then applying symbolic regression to the learned function, we factorize this complex problem into manageable components, dramatically reducing the search space from multiplicative to additive complexity.

Second, our neural network architecture (Lipschitz-Bounded Deep Network) inherently enforces smoothness constraints on the learned function, providing stability that would not be guaranteed with direct symbolic regression. The LBDN ensures that small changes in input lead to proportionally small changes in output, creating a well-behaved function space for symbolic regression to explore.

Third, and crucially, our symbolic regression operates within a broader cognitive model optimized via gradient descent. Embedding a non-differentiable genetic algorithm (symbolic regression) directly within this gradient-based optimization framework would create significant computational challenges. By replacing the value-of-information computation with a differentiable neural network, we transform this into a standard, tractable optimization problem.

2.4 Discussion

We believe this approach holds significant potential for refining existing theories and discovering novel computational principles across diverse domains, from learning and decision-making to perception and social cognition. The framework is particularly valuable when the underlying computations are expected to be complex but the scientific goal requires interpretable models that can generate testable hypotheses and link to neural mechanisms.

The next chapter will demonstrate the application of this framework to human information sampling behavior, revealing specific mathematical equations that describe how people value information under uncertainty.

3

Information Sampling

Interpretable abstractions of artificial neural networks predict behavior and neural activity during human information gathering[1]

3.1 Introduction

In the previous chapter, I introduced a hybrid modeling framework that combines theory-driven cognitive components with data-driven artificial neural networks to understand how humans compute the value of information during sampling decisions. I performed a validation test to show how symbolic regression can transform these learned neural network functions into interpretable mathematical expressions, providing a bridge between the flexibility of machine learning and the interpretability required for scientific understanding.

Several brain regions are thought to play key roles in information sampling under uncertainty. Notably all the neuromodulatory systems, with their origins in the ventral tegmental area (VTA), substantia nigra (SN), dorsal raphe nucleus (DRN), locus coeruleus (LC), and ventral septal nucleus (VSN) have at one time or another been proposed to reflect uncertainty or the potential for information gain [95, 10, 134, 142, 50, 40, 86, 9]. It is less clear whether each neuromodulatory system has a specific or unique relationship with uncertainty. It has been difficult to record simultaneously from multiple nuclei in animal models, especially in the same individual. At the same time, it has been difficult to identify reliable signals from some of these nuclei in humans using standard neuroimaging approaches which have limited spatial resolution.

[1]This chapter is based on the following paper:
Simone D'Ambrogio, Jan Grohn, Nima Khalighinejad, Marcelo Mattar, Laurence Hunt, and Matthew F. S. Rushworth. Interpretable abstractions of artificial neural networks predict behavior and neural activity during human information gathering, June 2025.

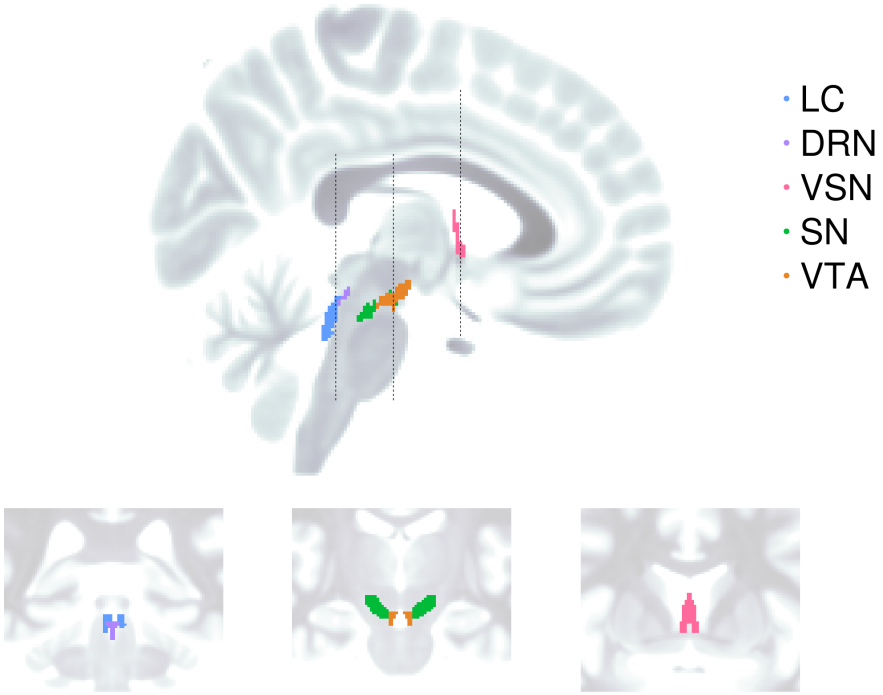

Figure 3.1: Brain regions of interest: locus coeruleus (LC), dorsal raphe nucleus (DRN), ventral septal nucleus (VSN), substantia nigra (SN), and ventral tegmental area (VTA), which have been implicated in uncertainty processing and information sampling.

Here, I exploit high resolution (1mm isotropic), rapid repetition time (1.378 s), accelerated ultra-high field (7T) functional magnetic resonance imaging together with an established suite of careful data collection and pre-processing steps to measure activity in VTA, SN, DRN, LC, and VSN simultaneously [9, 75]. The functional magnetic resonance imaging (fMRI) data collection was positioned to record simultaneously from all five ascending neuromodulatory nuclei and two interconnected cortical areas, the anterior insula (AI), and anterior cingulate cortex (ACC), which are distinguished by projecting to these nuclei and exhibiting related activity in other contexts [95, 134, 76, 21, 68, 67, 73]. Guided by the hybrid model, I identified specific and distinct patterns of activity across these seven brain areas, linked to distinct components of information valuation and evidence accumulation for action selection.

3.2 Results

3.2.1 Sampling Behavior Adaptively Scales with Task Difficulty and Uncertainty

Twenty participants completed the information-sampling task inside a 7T MRI scanner described in the previous chapter. Participants exhibited varying preferences for speed versus accuracy in their sampling behavior (Figure 3.2A). Some participants opted to spend more time gathering information, aiming to increase accuracy, while others prioritized faster decisions, accepting a higher risk of error (correlation between amount of information and accuracy: $r = 0.74$, $t = 4.33$, $P = 4.00 \times 10^{-4}$). As noted, each trial differed in two key aspects: the initial uncertainty of each option, as indicated by the green dots, and the final difference in red dots between the patches, ranging from a difference of 30 (easy trials) to 10 (difficult trials). Participants tended to gather more samples when the initial uncertainty was higher (there were fewer green dots, which were uncovered at the beginning of the sampling period, and more grey dots which were only uncovered one-by-one during sampling: $\beta = -0.069$, $s.e. = 0.013$, $z = -5.18$, $P = 2.27 \times 10^{-7}$) and when the final discrepancy in red dots was smaller ($\beta = -0.131$, $s.e. = 0.019$, $z = -6.87$, $P = 6.48 \times 10^{-12}$; Figure 3.2B). To assess whether this pattern is adaptive we estimated the optimal policy for this task solving the Bellman equation using dynamic programming [117, 17] (see Methods). This optimal policy treats the proportion of red dots in each patch as independent under the prior, whereas in the actual design the patch proportions were correlated (Section 2.2); the policy therefore underestimates the information available to an ideal observer who exploits this correlation, and provides a conservative reference for human-versus-optimal comparisons. We simulated action sequences from this optimal agent and compared them to the actual action sequences exhibited by participants. We found that participants' sampling behavior qualitatively resembled the optimal policy derived from our computational analysis: they adaptively increased their sampling time when initial uncertainty (100 – number of green dots, or in other words initial number of grey dots) was higher or when decisions were more challenging (lower disparity between the number of red dots associated with each option). However, the absolute number of samples collected was substantially higher than that prescribed by the optimal policy (Figure 3.2A). Two factors likely contribute to this gap. First, the optimal policy has perfect access to the running counts of red and black dots in both patches, whereas participants must rely on memory of past evidence whenever they are not attending a patch. Second, the dots within each patch were in continuous motion, so even the count of revealed red and black dots in the attended patch had to be inferred from a noisy perceptual signal rather than read off directly. Both factors inflate participants' effective uncertainty about each option, plausi-

bly motivating additional sampling. As a consequence, while accuracy generally increased with more samples, participants tended ultimately to earn fewer points (Figure 3.2A) due to the opportunity cost of performing fewer trials.

Task difficulty was determined partly by the difference between patches in terms of the proportion of red dots (i.e. total number of red dots divided by 100); when this difference in proportions was smaller, the task was more difficult. However, the task was also more challenging when the mean of the two proportions were closer to 0.5, because the posterior belief about the proportion of red dots (which can be modeled using a Beta distribution) is wider (there is more uncertainty in the estimate) and so there is more overlap between the distributions associated with the two patches, when their mean is near 0.5, given the same difference in proportion of red dots. Conversely, when the mean proportion is closer to the boundaries of 0 or 1, the posterior belief is narrower, making it easier to distinguish the two estimates from one another. For example, if 10 dots are revealed in each patch, discriminating between one patch with 4 red dots and another with 6 red dots (a difference in proportion of 0.2, mean proportion 0.5) is harder than discriminating between a patch with 1 red dot and another with 3 red dots (also a difference of 0.2, but mean proportion 0.2). We tested whether participants adapted their sampling strategy based on the inherent difficulty of discriminating proportions near 0.5 by examining how sampling behavior varied with the absolute distance of the mean proportion from 0.5. Both human participants and the optimal agent gathered more samples when the mean proportions were closer to 0.5 (participants: $\beta = -0.085$, $s.e. = 0.017$, $z = -4.87$, $P = 1.09 \times 10^{-6}$; optimal agent: $\beta = -0.038$, $s.e. = 0.015$, $z = -2.44$, $P = 0.015$), after controlling for the total number of revealed dots and the relative difference between patches. This pattern suggests that participants adapted their sampling strategy based on the inherent difficulty of the task.

Finally, we examined which kinds of uncertainty influenced participants' sampling behavior. In our task design, participants knew the initial uncertainty (number of green dots) of the blocked option, which should be irrelevant for optimal sampling between the two available options, as well as the uncertainty of the decision-relevant options that were available to be chosen. We tested whether this irrelevant uncertainty affected participants' sampling behavior using a mixed-effects Poisson regression model. It did not. The analysis revealed that participants sampled less from the attended option when this attended option had more initial green dots ($\beta = -0.138$, $s.e. = 0.017$, $z = -8.11$, $P = 5.24 \times 10^{-16}$). They also sampled more from the attended option when the other option, the unattended option, had more initial green dots ($\beta = 0.173$, $s.e. = 0.024$, $z = 7.19$, $P = 6.42 \times 10^{-13}$). Crucially, however, the number of green dots in the blocked option did not significantly influence sampling behavior ($\beta = -0.006$, $s.e. = 0.006$, $z = -1.01$, $P = 0.310$), suggesting

that participants were able to appropriately ignore task-irrelevant background uncertainty even when their behavior was influenced by task-relevant uncertainty. This pattern held true even immediately after exposure to the blocked option (no significant interaction between blocked dots and first visit, $\beta = 0.011$, $s.e. = 0.008$, $z = 1.41$, $P = 0.158$) and remained consistent across sessions, indicating that participants maintained this optimal strategy throughout the experiment. Consequently, this background uncertainty will not be the focus of our subsequent modeling and neural analyses. These results suggest that participants were able to distinguish between relevant and irrelevant sources of uncertainty in their sampling behavior.

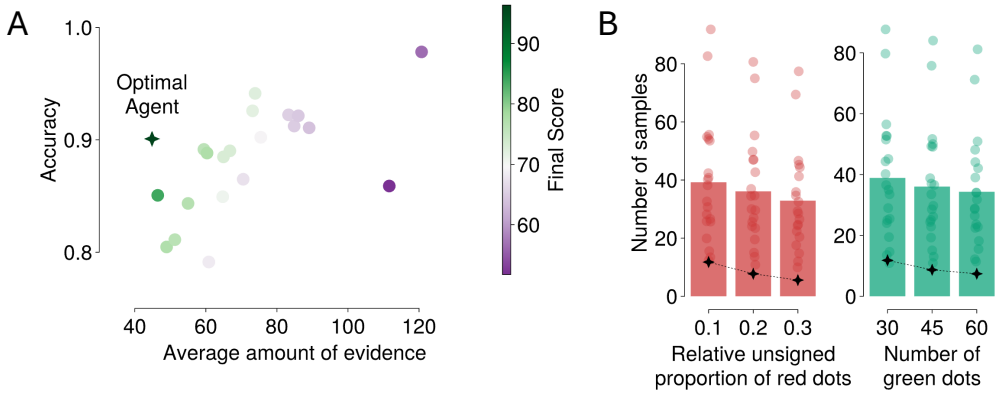

Figure 3.2: Sampling behavior and computational model. A, Relationship between accuracy and average amount of evidence collected. Each colored dot represents an individual participant, with color indicating their final score according to the color scale on the right. The dark green star marks the position of the optimal agent. B, Number of samples collected under different task conditions. Left: Bar graph showing the number of samples collected across three levels of relative unsigned proportion of red dots between patches (0.1, 0.2, 0.3). Right: Bar graph showing the number of samples collected across three levels of initial information (30, 45, 60 green dots). Individual participant data points are shown as colored dots, and black stars indicate the optimal agentś behavior.

3.2.2 The ANN-derived value of information predicts participants' sampling decisions

To investigate participants' strategies, we developed and fitted a computational model that takes as input the current state (i.e. the number of red and black dots in both patches) to compute action-specific values. We considered four actions: staying (sampling from the currently attended patch; Figure 2.1B; Figure 2.1A, bottom right), switching (sampling from the unattended patch; Figure 2.1B; Figure 2.1A, bottom left), and selecting either the attended or unattended patch (Figure 2.1A, top; Figure 2.1B). To make any one of

these four actions, the model maintains and updates its beliefs about the likely number of red dots in both the currently attended patch and the unattended patch. For the attended patch, this belief is updated directly based on the colors of the dots revealed during ongoing sampling.

For the unattended patch, where there is no direct sensory input, the model relies on information presumed to be stored in working memory. The representation of this information is handled differently depending on whether the patch has been previously sampled in the trial: if the unattended patch has not yet been visited in the trial, its initial state (and thus the potential value of sampling it) is estimated using the number of green dots. The number of green dots provides a measure of the information that would be gained upon a first visit, influencing the model's assessment of its current uncertainty. To estimate the proportion of red dots within it, the model draws upon information from the currently attended patch, effectively using the ongoing sampling as a cue. If the unattended patch has been visited previously, the model relies on its memory of the dots observed during those past visits. However, this remembered information is assumed to degrade the longer the patch remains unattended, leading to an increase in uncertainty about its contents over time [3, 85]. The detailed implementation of these transformations is described in the Methods.

A critical aspect that makes effective information sampling possible is evaluating the marginal value of acquiring additional information from each option. To evaluate how participants computed this value of information, we compared three distinct models with cross validation. The first model assumes a linear relationship between the value of information and the number of collected samples (the total number of red and black dots revealed after removal of the green or grey covers). This linear model assigns the same value to new information, regardless of the number of previously collected samples (Figure 3.3A, top). In other words, each new sample reduces the value of learning about the proportion of red dots in the patch by the same amount. The second model implements an Upper Confidence Bound (UCB) function to estimate the value of information. This model assigns decreasing value to information as the number of collected samples increases (Figure 3.3A, middle). Finally, the third model uses an artificial neural network (ANN) to learn from the data the mapping between the state variables (such as the number and color of revealed dots) and the value of gathering additional information (Figure 3.3A, bottom). This approach, which we refer to as a hybrid model, combines the data-driven nature of the ANN with the theory-driven design of the cognitive model in which the ANN is instantiated. The advantage of this approach is that it gives us the flexibility to capture potentially complex, non-linear relationships between task variables and information value without making strong a priori assumptions about the functional form of this relationship.

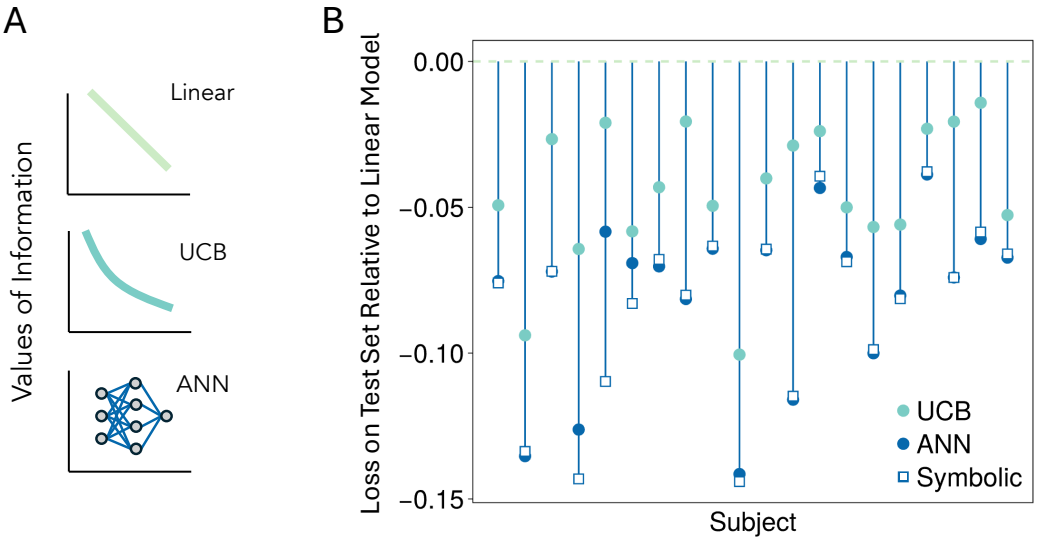

Figure 3.3: Model comparison. A, Three approaches to computing the value of information: linear function (top), Upper Confidence Bound (UCB) algorithm (middle), and artificial neural network (ANN; bottom). B, Model comparison across participants showing loss relative to the linear model. Each vertical line represents a participant, with light blue circles (UCB model), dark blue circles (ANN model), and open squares (symbolic model derived through symbolic regression) showing the relative loss for each model type. Lower values on the y-axis indicate better model fit.

Once the ANN estimates the value of information for sampling each patch, the model computes the overall value for four potential actions (Figure 2.2, right). For the two sampling actions, the value of staying to sample the currently attended patch is its ANN-estimated value of information. The value of switching to sample the unattended patch is its ANN-estimated value of information, reduced by a fitted cost of switching. For the two selection actions, the model compares the current subjective estimates of red dots (ρ) in the attended and unattended patches. The value of selecting the attended patch is based on the difference in these subjective estimates ($\rho_{attended} - \rho_{unattended}$), and correspondingly for selecting the unattended patch ($\rho_{unattended} - \rho_{attended}$). These four action values are then scaled by a temperature parameter and transformed via a softmax function to yield the probability of choosing each action.

We next compared how well each approach to computing the value of information (e.g. linear, UCB, or hybrid-ANN) could account for participants' behavior. Using a cross-validation approach, we found that for each participant the hybrid model achieved a better fit than the other two models for all participants (Figure 3.3B). We also considered a Symbolic Model in this model comparison (Figure 3.3B), which we detail in the next section. This suggests that the ANN was able to find an alternative strategy that the first two competing models did not capture.

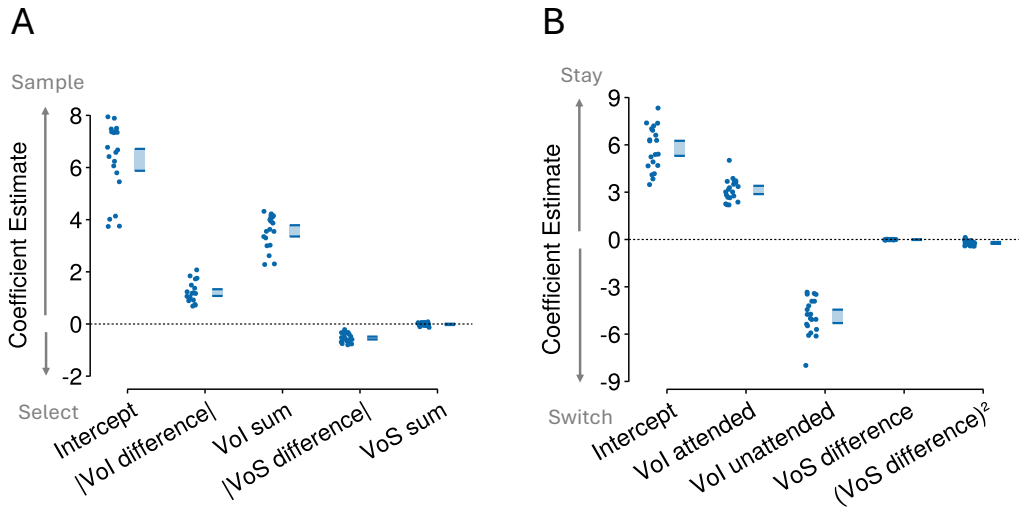

Figure 3.4: Behavioral predictions. A, Regression analysis predicting the probability of sampling versus selecting. Each dot represents a subject-specific random effect estimate from a mixed-effects logistic regression model, with blue rectangles showing the ±95% confidence intervals of the fixed effects. Predictors include the intercept, absolute difference in value of information (|VoI difference|), sum of value of information (VoI sum), absolute difference in value of selection (|VoS difference|), and sum of value of selection (VoS sum). B, Regression analysis predicting the probability of staying versus switching. Subject-specific random effect estimates (dots) and fixed effect ±95% confidence intervals (blue rectangles) for predictors including intercept, value of information for attended patch (VoI attended), value of information for unattended patch (VoI unattended), and value of selection differences.

We then assessed how the ANN-derived value of information affected sampling decisions (Figure 3.4A). Firstly, we used a mixed-effect logistic regression model to predict the probability of sampling information versus selecting a patch. In other words, we estimated the probability that participants would take either of the first two actions (Figure 2.1A, bottom: staying to continue sampling from the current patch or switching to sample from the other patch) as opposed to either of the last two actions (Figure 2.1A, bottom: selecting the currently attended option or the currently unattended option). We found that the probability of sampling was positively associated with the sum of the ANN-derived value of information of both options ($\beta = 2.860$, $s.e. = 0.167$, $z = 17.16$, $P = 5.25 \times 10^{-66}$), and with the unsigned difference in value of information between the two options ($\beta = 1.292$, $s.e. = 0.085$, $z = 15.16$, $P = 6.14 \times 10^{-52}$). This suggests that participants were more likely to sample information when more information was potentially obtainable in the near future, and when the discrepancy between the amount of information obtainable from the two options was large. From here on we refer to the value of information as being higher when these two factors were higher.

Secondly (Figure 3.4B), we used a mixed-effect logistic regression model to predict the

probability that one or other of the two different options would be sampled for information (Figure 2.1A, bottom, right versus left): staying – continuing to attend the currently attended patch – versus switching to attend to the alternative patch. We found that the probability of staying was positively associated with the value of information of the currently attended patch ($\beta = 3.971$, $s.e. = 0.434$, $z = 9.14$, $P = 6.03 \times 10^{-20}$) and negatively associated with the value of information of the unattended patch ($\beta = -5.880$, $s.e. = 0.456$, $z = -12.89$, $P = 5.13 \times 10^{-38}$), indicating that participants were more likely to sample from the currently attended patch when its estimated value of information was high relative to the unattended option.

Finally, we examined how participants made their final selection between the attended and unattended patches (Figure 2.1A, top). Using a mixed-effects logistic regression, we found that participants' choices were strongly influenced by the proportion of red dots in both patches, with a positive effect for the proportion difference between attended and unattended patch ($\beta = 4.572$, $s.e. = 0.336$, $z = 13.60$, $P = 3.77 \times 10^{-42}$) and a positive effect for the proportion sum ($\beta = 0.631$, $s.e. = 0.104$, $z = 6.08$, $P = 1.19 \times 10^{-9}$). The value of information also influenced these selection decisions: participants were less likely to select the attended patch when its value of information was higher than the value of information of the unattended patch ($\beta = -0.782$, $s.e. = 0.159$, $z = -4.91$, $P = 9.16 \times 10^{-7}$). This pattern suggests that participants interpreted high remaining information value as uncertainty about the true proportion of red dots, making them less likely to select patches with high remaining uncertainty. It is also consistent with an account of choice in which participants seek to minimize uncertainty about the option that they will ultimately choose [61].

In summary, cross-validation analyses revealed that, for each participant, the hybrid-ANN model provided a superior account of information sampling behavior compared to both linear and UCB models (Figure 3.3B). Subsequent regression analyses demonstrated how this ANN-derived value of information guided participants' decisions (Figure 3.4): it significantly predicted when participants chose to sample further information versus make a final selection, and where they directed their sampling (i.e., whether to stay with the current patch or switch to the alternative). While final selection decisions were primarily driven by the perceived proportion of red dots, the value of information also played a secondary role, with participants tending to avoid options with a high remaining value of information (implying uncertainty about the number of red dots associated with the option).

3.2.3 The ANN integrates evidence from both patches to compute value of information

Our behavioral analyses demonstrate that the hybrid-ANN approach has a better predictive performance than the linear and UCB models. To gain insight into the nature of the computation learned by the ANN, we first performed a qualitative analysis of the ANN-learned function (Figure 3.5).

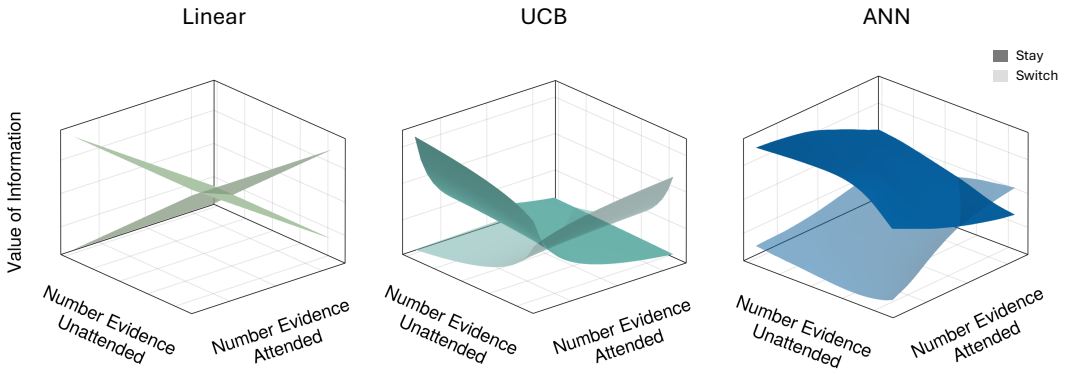

Figure 3.5: Three-dimensional surface plots showing how the value of information varies with the number of revealed dots in both the attended and unattended patches. Each model (Linear, UCB, and ANN) is represented by two surfaces: darker color for the value of staying with the attended patch and lighter color for the value of switching to the unattended patch.

We started by simply visualizing the input-output relationship of the three functions (linear, UCB, and ANN) as a surface in 3D space relating the number of pieces of evidence (N) available in the attended option and in the unattended option to the value of information associated with staying and continuing to sample the attended option, and to the value of information associated with switching to the alternative unattended option (Figure 3.5). It should be noted that, because the participants might take another course of action, actually selecting an option, the value of staying to gather more information about the current option and the value of switching to gather information about the alternative option, are not simply inverses of one another.

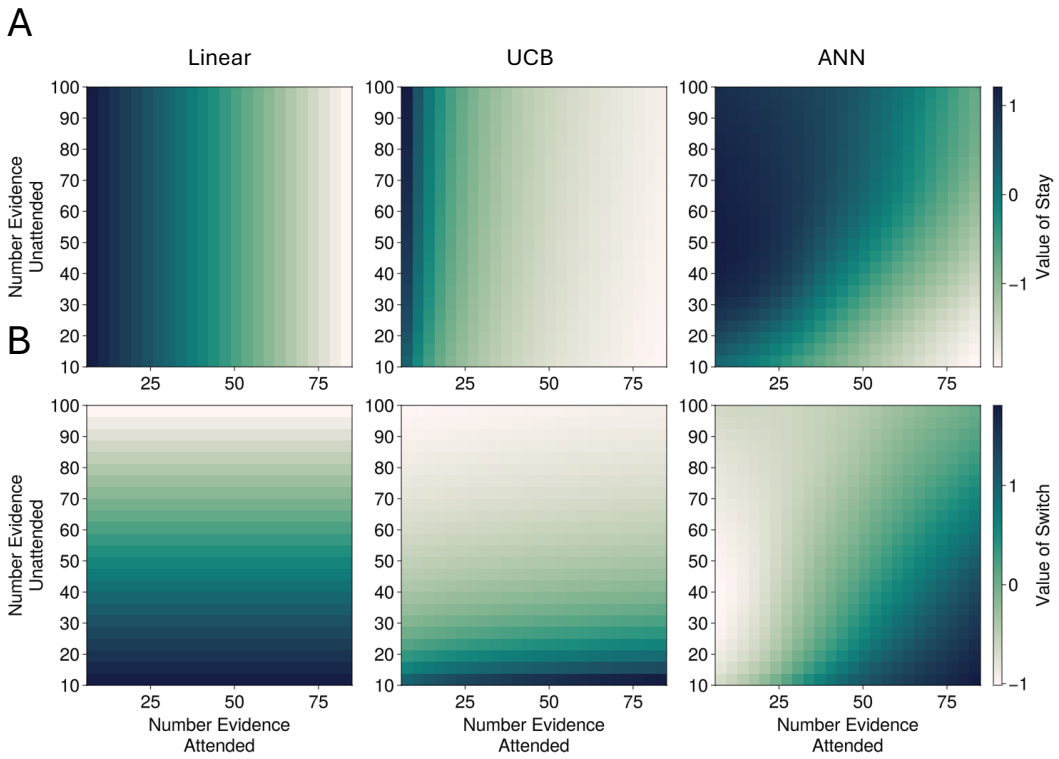

Figure 3.6: A, Heatmaps showing the value of staying as a function of evidence collected from both patches. Color intensity represents the magnitude of the value, with the scale shown on the right. B, Heatmaps showing the value of switching as a function of evidence collected from both patches.

We then created heatmaps for each model showing the value of gathering information from the attended (value of staying; Figure 3.6A) and unattended patches (Figure 3.6B; value of switching) as a function of the evidence available to both the attended and unattended patches. Figure 3.6A shows distinct patterns in how the three models integrate information from both patches. The linear model's gradient varies exclusively along one dimension, the attended evidence, when computing the value of staying, and the unattended evidence when computing the value of switching. The UCB model shows a similar predominant dependence on one dimension, though with some modulation by the second dimension. This behavior is an expected consequence of the models' core assumptions. The linear model calculates the value of information for each patch in isolation, based solely on the evidence accumulated for that specific patch. Therefore, the decision to stay or switch is influenced only by the state of the patch being considered for the next sample. In contrast, UCB evaluates the informational value of an option in relation to the overall information gathered from all options. This means that the evidence from both patches contributes, albeit to different degrees, when assessing the value of sampling either the attended or the unattended option. The ANN-derived value of stay, shows clear depen-

dencies on both dimensions: when computing the value of staying, it increases with attended evidence but simultaneously decreases with unattended evidence, resulting in a diagonal gradient pattern. This pattern indicates that the ANN, unlike the Linear model, learns to integrate evidence from both patches. While the UCB model also exhibits a two-dimensional influence, the ANN assigns a more substantial role to unattended evidence when computing the value of staying (contributing to the observed diagonal gradient).

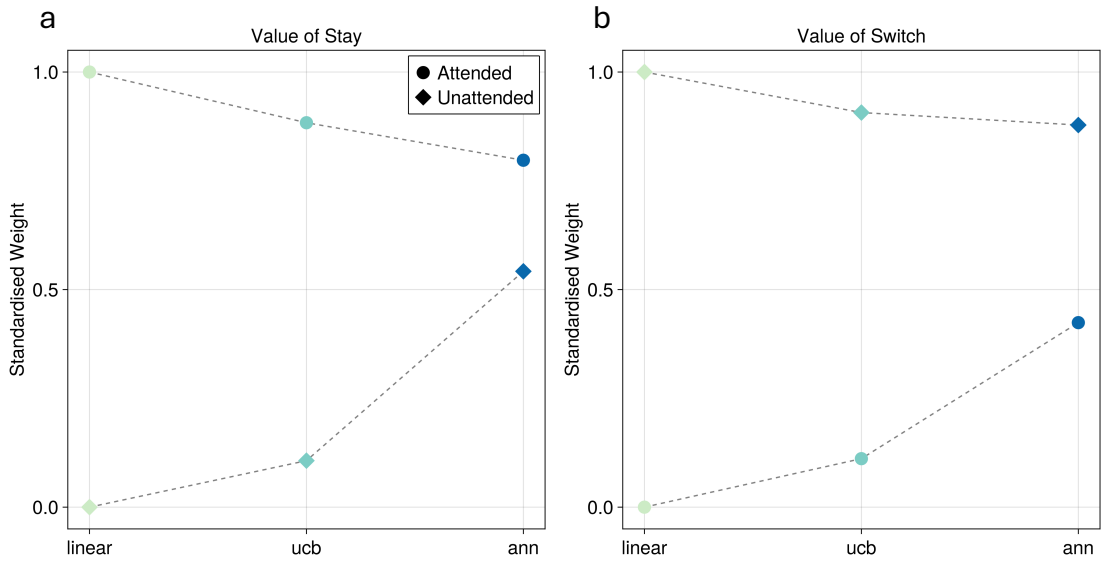

Figure 3.7: Model comparison for evidence weighting. a, Standardized weights for the value of stay computation across three models (linear, UCB, and ANN). Circles represent weights for the attended evidence, while diamonds represent weights for the unattended evidence. Colors transition from light green to dark blue across models. Dotted lines connect the weights across models. b, Standardized weights for the value of switch computation across the same three models, using the same visual conventions as in panel a.

To quantify this pattern, we performed a linear regression analysis to estimate how strongly each model's output depends on attended versus unattended evidence (Figure 3.7A). For the value of stay computation, the Linear model shows an exclusive dependence on attended evidence ($\beta_{attended} = 1.0$) with no influence of unattended evidence ($\beta_{unattended} = 0.0$). The UCB model shows a similar but less extreme pattern, with a strong weight on attended evidence ($\beta_{attended} = 0.8$) and a modest one on the unattended evidence ($\beta_{unattended} = 0.2$). The ANN, in contrast, shows more balanced weights for both attended ($\beta_{attended} = 0.7$) and unattended ($\beta_{unattended} = 0.6$), confirming our visual observation that it integrates information from both patches more evenly when computing the value of staying. A similar pattern emerges for the value of switch computation, where the ANN again shows more balanced weighting of evidence from both patches compared to the Linear and UCB models (Figure 3.7B).

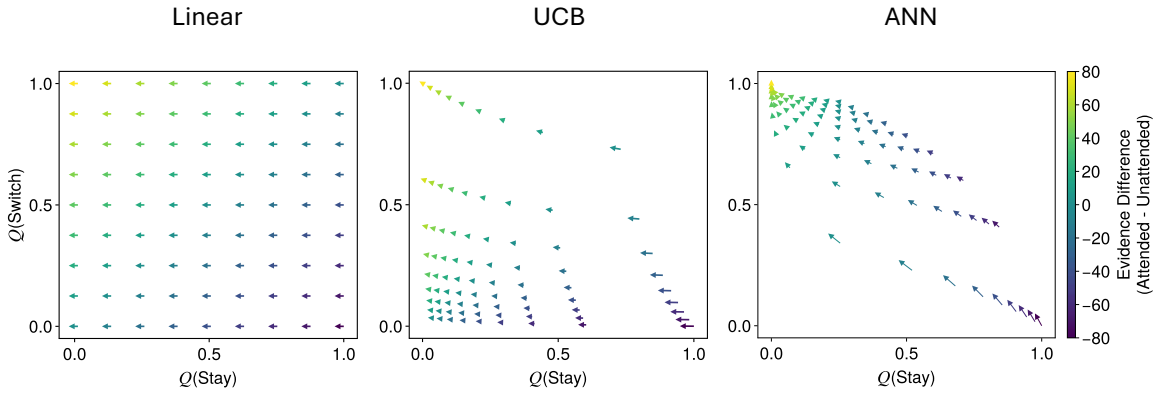

Figure 3.8: Vector field plots showing how collecting an additional sample from the attended option affects both the value of staying (x-axis) and switching (y-axis). Each arrow represents the transition from current values (arrow origin) to updated values (arrow tip) after collecting one new sample. The color of the arrows indicates the difference in evidence between attended and unattended patches according to the color scale on the right, while the length of the arrow indicates the size of the update.

To further analyze the difference between the ANN and the other two models, we visualized in Figure 3.8How collecting an additional sample from the attended option affects both the value of staying and switching. Each arrow in the plot represents the transition from current values (arrow origin) to updated values (arrow tip) after collecting one new sample. Note that the value of staying and the value of switching are computed separately rather than being inverses of each other, because both are weighed against the value of selecting either patch in the final softmax comparison across all four possible actions. In the Linear and UCB models, these arrows are predominantly horizontal, indicating that new samples from the attended option strongly affect the value of staying but have minimal impact on the value of switching. In contrast, the ANN model has more diagonal arrows, revealing that new information from the attended option influences both the value of staying and switching. Specifically, the diagonal pattern suggests that as evidence accumulates from the attended patch, the value of staying decreases while the value of switching increases by a similar amount. This coordinated change means that while there may be relatively little change in the overall probability of continuing to sample (versus making a final selection), there is an increase in the probability of switching attention to sample the unattended patch. Overall, these results suggest that the ANN has learned a function that uses information from both the attended and unattended patches to guide sampling decisions.

3.2.4 The ANN can be transformed into an interpretable symbolic function

While the qualitative analyses reported so far provide valuable insights into how the ANN operates, they do not provide a precise mathematical description of the computation it performs. To obtain a quantitative interpretation of the ANN's learned function, we turned to symbolic regression, a method that can discover interpretable mathematical expressions that approximate complex nonlinear functions [28]. This approach is particularly powerful for neural network interpretation for two reasons. First, it can find human-readable expressions that capture the essential computations performed by the network while eliminating the complex transformations that typically obscure these computations in the network's internal representations [80]. Second, it can dramatically reduce the number of parameters (in this case, 7592 trainable units in the ANN) to a few, interpretable features.

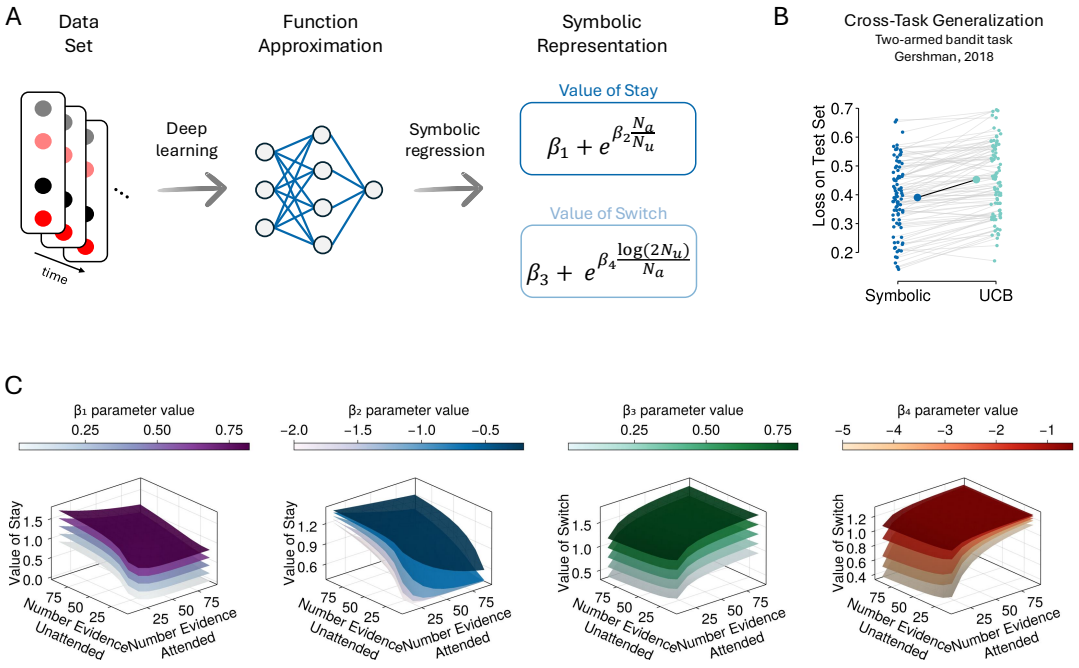

Figure 3.9: Symbolic representation of the ANN-derived value of information function. a, Process of deriving interpretable mathematical expressions from behavioral data. Left: Schematic of the dataset containing patterns of dots across time. Middle: Function approximation using a deep neural network. Right: Symbolic representation showing the mathematical equations derived through symbolic regression for the value of staying and the value of switching, where N_a and N_u represent the number of dots in the attended and unattended patches, respectively. b, Cross-task generalization comparison. Scatter plot showing loss on a test set from a two-armed bandit task (Gershman, 2018) for both the symbolic model (left) and UCB model (right). Each gray line connects performance of both models for the same participant. c, Parameter sensitivity analysis. Three-dimensional surface plots showing how the value functions change with different parameter values. Each surface represents the function with a specific parameter value according to the color scales above each plot.

We extracted a mathematical expression for the value of stay and one for the value of switch. For the value of stay, we extracted the following expression $\beta_1 + e^{\beta_2 N_a/N_u}$ (Figure 3.9A), where N_a and N_u represent the number of dots revealed in the attended and unattended patches respectively. For the value of switch, we found $\beta_3 + e^{\beta_4 log(2N_a)/N_u}$ (Figure 3.9A). As can be seen from these equations, β_1 acts as a global offset for the value of stay, effectively controlling the balance between undirected exploration and maintaining attentional focus on the currently attended option (see Figure 3.9C, left). β_2 modulates how strongly the value of stay is influenced by the interaction between attended and unattended evidence (Figure 3.9C, second from left). Similarly, β_3 acts as an intercept for value of switch and β_4 determines the sensitivity of the value of switch computation to the relative amounts of evidence between patches (Figure 3.9C, right). Notably, while these

equations contain four core parameters, our feature selection analysis revealed that the visit number (first visit of the patch vs subsequent visits) was critical for optimal ANN performance. Consequently, each parameter is fitted independently for first visits versus subsequent visits, yielding eight total parameters that capture distinct sampling strategies across different phases of exploration. These interpretable parameters provide insight into individual differences in sampling strategies (Figure 3.10).

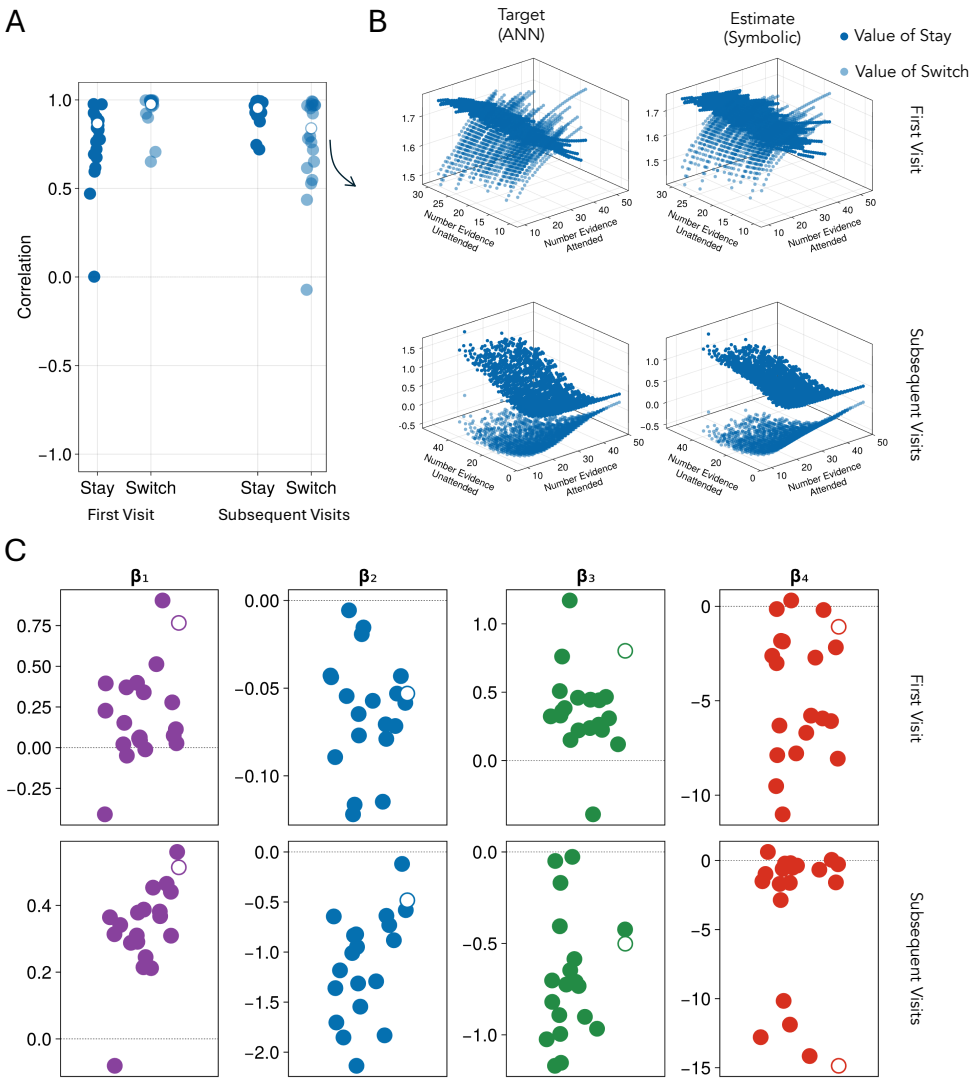

Figure 3.10: Symbolic regression approximation of ANN-derived value of information functions. A, Correlation between ANN-generated value of information and symbolic function approximations across all participants (n=20). Four correlations are shown for value of staying and switching during first visits and subsequent visits to patches. Each point represents one participant, with the highlighted participant (white circle) shown in detail in panel B. B, Example participant showing 3D visualization of value of information functions. Left column shows target functions from the trained ANN model; right column shows estimates from the optimized symbolic functions. Top row displays first visit functions, bottom row shows subsequent visit functions. Dark points represent value of staying, light points represent value of switching. C, Individual parameter estimates ($\beta_1 - \beta_8$) controlling the symbolic functions across all participants. Each parameter corresponds to specific components of the value of information computation: β_1,β_5 (stay VoI offsets), β_2,β_6 (stay VoI scaling), β_3,β_7 (switch VoI offsets), β_4,β_8 (switch VoI scaling) for first and subsequent visits respectively. The highlighted participant (white circle) corresponds to the example shown in panel B.

To examine whether these symbolic functions accurately capture the function learned by the ANN, we replaced the ANN component in our hybrid model (Figure 3.3A, bottom) with these newly discovered functions and assessed how well this new variant of the model explained participant behavior using the approach. We found the symbolic-hybrid model and ANN-hybrid had comparable power to explain participant behavior (Figure 3.3B), suggesting we had successfully discovered a transparent and interpretable mathematical relationship between evidence and the value of information.

Finally (Figure 3.9B), we tested whether these symbolic functions capture general principles of information sampling rather than task-specific features. We evaluated performance of the symbolic model on an independent dataset where participants completed a two-armed bandit task [42]. Participants in this study performed a very different task from ours (e.g. they repeatedly chose between two options and received point rewards), yet in both tasks participants had to strike a balance between exploration and exploitation to maximize their rewards. Remarkably, our symbolic functions outperformed the UCB model in predicting participants' choices also in this different context (Wilcoxon signed-rank test: $W = 15.0$, $n = 89$, median difference $= -0.07$, $P = 4.31 \times 10^{-16}$). We noticed that one simple difference between the standard UCB formulation and our symbolic model is that the UCB lacks an offset parameter while our symbolic model includes one. To ensure that this superior performance was not simply due to the model's ability to adjust the baseline offset, but rather reflected a fundamental difference in the shape of the value-of-information function, we tested an UCB model that included an offset parameter in its exploration bonus computation. Even when compared to this more flexible UCB variant, the symbolic model still demonstrated significantly better predictive performance (Wilcoxon signed-rank test: $W = 286.0$, $n = 89$, median difference $= -0.03$, $P = 2.21 \times 10^{-12}$). These results suggest that the equations discovered with symbolic regression capture fundamental aspects of human exploration behavior that are not specific to the information sampling task we used and cannot be accounted for by simple parametric extensions to existing models.

3.2.5 The ANN-derived value of information can predict neural activity

Our behavioral analyses demonstrate that the ANN-derived value of information better predicts participants' behavior (Figure 3.3B). Next, we sought to understand whether this value of information could also predict neural activity. Participants performed the behavioral task while undergoing ultra-high field fMRI recordings of the blood oxygen level dependent (BOLD) signal. To achieve high spatial resolution (1mm isotropic voxels), our functional imaging protocol used a limited field of view that captured key regions of interest in the midbrain, brainstem, and interconnected cortical areas (Figure 3.11), rather

than whole-brain coverage. We used these recordings to investigate brain activity that represented the main task variables. We employed a general linear model (GLM) analysis across the whole brain volume scanned. The GLM used the value of information as a predictor of neural activity, while controlling for other task-relevant variables such as outcome, and value of selection.

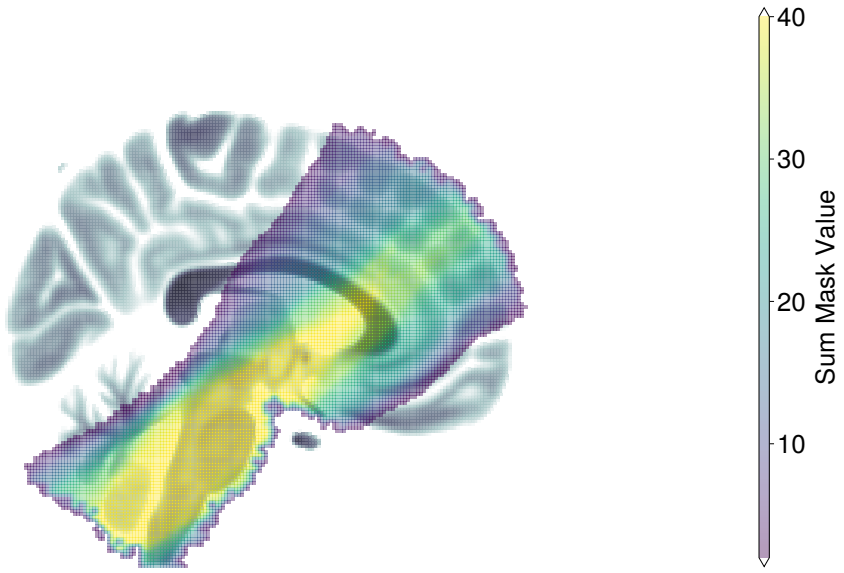

Figure 3.11: Field of view coverage across sessions. Sagittal view (x = 87) showing the spatial coverage of the limited field of view functional imaging across all 40 sessions (2 sessions × 20 participants). The grayscale background shows the MNI152 T1 1mm brain template. The colored overlay represents the sum of individual session masks, with the color scale indicating the number of sessions with coverage at each voxel location. Warmer colors indicate regions consistently captured across more sessions, while cooler colors represent areas with coverage in fewer sessions.

We compared the goodness of fit when using three different value-of-information computations: our ANN-derived estimates, the linear model estimates, and the UCB model estimates. For each voxel within our imaging field of view, this GLM analysis yielded a predicted BOLD timeseries for each of the three VoI models. We then calculated the discrepancy (Mean Squared Error, MSE) between the observed BOLD timeseries and the model-predicted timeseries on a per-voxel basis. This provided a quantitative measure of each model's ability to fit the neural data in each voxel. Note that (unlike for behavior in Figure 3.3B), this model comparison fits the same number of parameters per model when predicting the BOLD signal: it takes the model output value of information (hold-

ing the number of internal parameters in the model fixed) and uses them as parametric modulators in a conventional GLM analysis of fMRI data. Consistent with our behavioral findings, the ANN-derived value of information demonstrated a better overall fit to the neural data across the brain. Specifically, when comparing the distribution of these per-voxel MSEs, the ANN-derived VoI resulted in significantly lower MSEs than both the linear and UCB models (Wilcoxon signed-rank test: median $MSE_{ann} - MSE_{linear} = -82.42$, $P = 2.15 \times 10^{-12}$; median $MSE_{ann} - MSE_{ucb} = -90.77$, $P = 5.74 \times 10^{-13}$; Figure 3.12). This suggests that the computational principles captured by our hybrid ANN model not only better describe participants' behavior but also more accurately reflect the underlying neural computations driving information sampling decisions. We then investigated activity in specific brain regions.

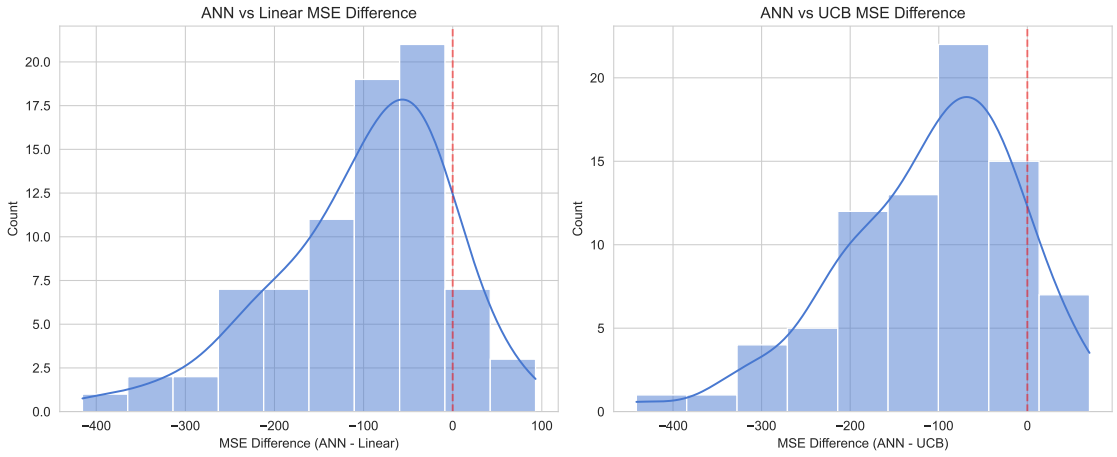

Figure 3.12: Neural model comparison. Histograms showing the distribution of Mean Squared Error (MSE) differences between models in predicting neural activity. Left: Histogram of MSE differences between the ANN and Linear models (ANN - Linear), with negative values indicating better performance by the ANN model. Right: Histogram of MSE differences between the ANN and UCB models (ANN - UCB). Both histograms include a blue curve showing the estimated probability density and a vertical red dashed line at zero.

3.2.6 Anterior Insula and Anterior Cingulate Cortex covary with the ANN-derived Value of Information

To identify the neural correlates of the ANN-derived value of information, we conducted two complementary analyses. First, we performed a whole-brain GLM analysis to explore cortical regions most strongly associated with the ANN-derived value of information. Second, we conducted targeted analyses on pre-defined regions of interest (ROIs) at the origins of the neuromodulatory systems, DRN, LC, VSN, VTA, and SN (Figure 3.1). The neuromodulators associated with each of these nuclei have been proposed previously as mediators of the impact of uncertainty on decision-making [95, 10, 134, 142, 50, 40, 86, 9]. In

general, unlike the cortical regions, they are too small to survive cluster correction analysis strategies that incorporate a thresholding criterion based on the spatial extent of the activity [74, 75, 122].

The whole-brain analysis revealed that activity in the anterior insula (AI) and anterior cingulate cortex (ACC) was associated with the ANN-derived value of information (Figure 3.13A and Figure 3.13B). AI activity exhibited a negative association with the difference between value of stay and value of switch (Figure 3.13A top) and a negative association with the sum of value of stay and value of switch (Figure 3.13B top); in other words, AI activity increased as the value of switching attention increased and decreased as the value of maintaining attention at the current location increased. The ACC had the same pattern (Figure 3.13B). These computational signatures appear crucial for guiding participants' decisions about whether to stay with the current patch or switch to the alternative (Figure 2.1A, bottom), and whether to keep sampling information or stop the learning process and initiate an option selection (Figure 2.1A, bottom). To test whether the superior model fit extended to these two specific regions, we compared the MSE between models within AI and ACC ROIs. Consistent with the whole-brain results, the ANN-derived VoI showed significantly lower MSE than both linear and UCB models in both regions (AI: ANN vs Linear MSE difference $= -175.183$, $P = 1.94 \times 10^{-12}$; ANN vs UCB MSE difference $= -200.76$, $P = 1.01 \times 10^{-12}$; ACC: ANN vs Linear MSE difference $= -57.088$, $P = 7.02 \times 10^{-5}$; ANN vs UCB MSE difference $= -50.246$, $P = 1.42 \times 10^{-5}$).

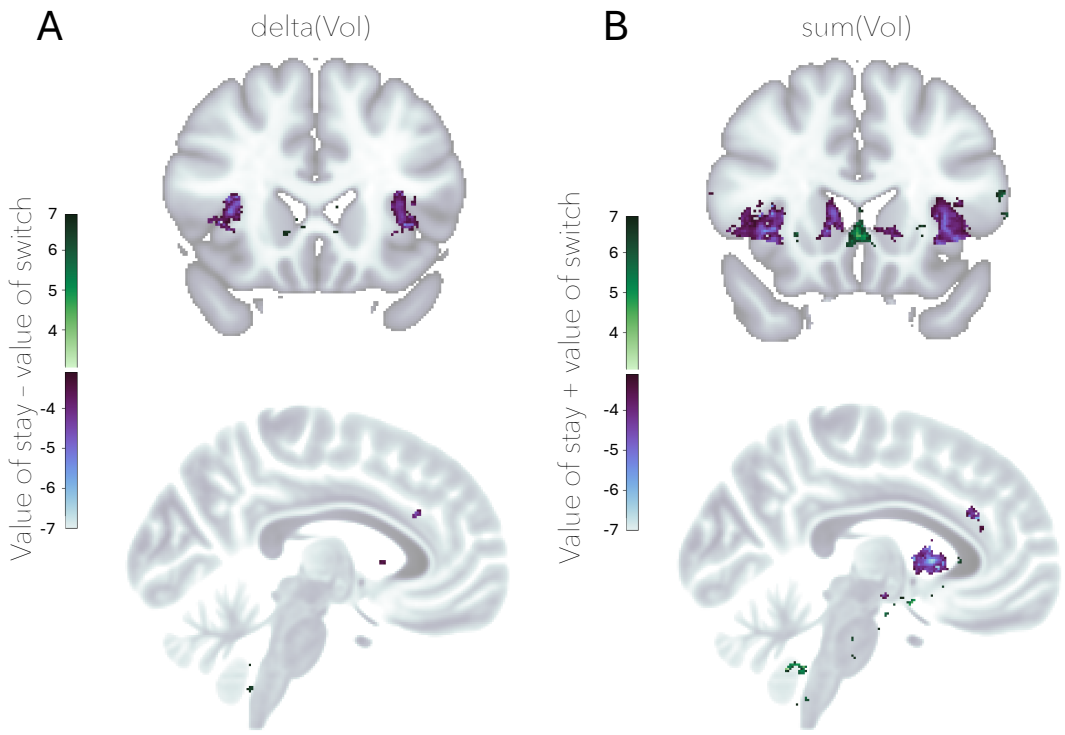

Figure 3.13: Neural correlates of ANN-derived value signals. A, Whole-brain analysis showing regions where BOLD activity correlates with the difference between the value of staying and the value of switching (VoI Stay - VoI Switch). B, Whole-brain analysis showing regions correlating with the sum of the value of staying and switching (VoI Stay + VoI Switch). Color bars indicate Z-statistic values; thresholded at $Z > 3.1$, cluster-corrected $P < 0.001$.

Analysis of the subcortical ROIs revealed a distinct pattern in the VTA (Figure 3.14), which showed positive coding of the sum of value of selecting the attended and unattended patch ($\beta = 11.295$, $s.e. = 3.929$, $z = 2.87$, $P = 0.020$) but negative coding for the sum of value of information ($\beta = -46.111$, $s.e. = 9.978$, $z = -4.620$, $P = 1.93 \times 10^{-5}$). This activity pattern would be sufficient to guide participants' decisions about whether to continue sampling information from the patch they are attending or to make a final selection (Figure 2.1A, bottom right versus top right).

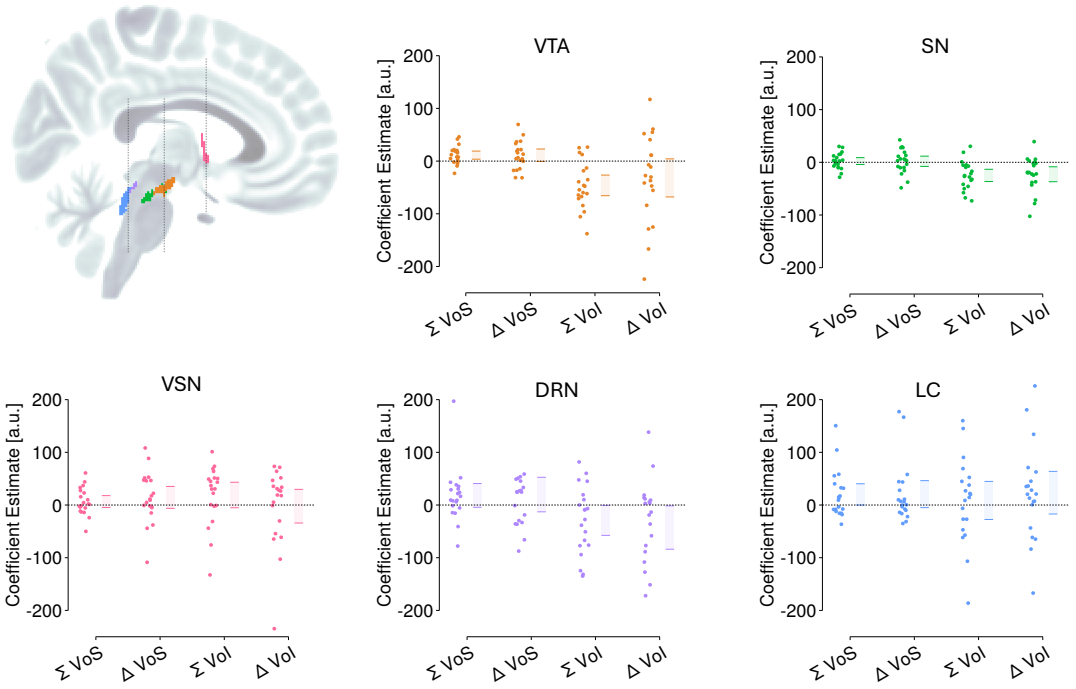

Figure 3.14: ROI analysis results for neuromodulatory nuclei. Left: Sagittal view showing locations of VTA (orange), SN (green), VSN (pink), DRN (purple), and LC (blue). Right: Coefficient estimates (effect sizes from weighted mixed-effects models on beta values) for regressors representing the sum (Σ) and difference (Δ) of the value of selection (VoS) and value of information (VoI) within each ROI. Each small dot represents an individual participant's random effect estimate plus the fixed effect; shaded bars indicate the 95% confidence interval of the fixed effect (group mean).

The SN exhibited a pattern similar to that observed in the cortical regions AI and ACC (Figure 3.14). Activity in the SN was negatively associated with both the sum of the value of information ($\beta = -24.863$, $s.e. = 5.924$, $z = -4.19$, $P = 6.38 \times 10^{-5}$) and the difference in the value of information between staying and switching ($\beta = -22.665$, $s.e. = 7.243$, $z = -3.129$, $P = 0.008$). This suggests the SN, like AI and ACC, tracks aspects related to the overall potential for information gain and the relative informational value of the current versus alternative option.

There has been particular interest in the possibility that the LC, and the noradrenergic system with which it is linked, the DRN, and its serotonergic system, or the VSN, associated with the cholinergic system, encode uncertainty. Our high-field fMRI recordings gave us a unique opportunity to test these hypotheses in humans. We first examined whether LC, DRN, or VSN activity can predict the ANN-derived value of information, an index closely, but inversely, related to uncertainty.

We further examined activity related to the sum and difference of the value of information and the sum and difference of the value of selection within the anatomically defined

VSN, DRN, and LC ROIs (Figure 3.14). In contrast to VTA and SN, we found no significant association between BOLD activity in VSN, DRN, or LC and any of these four regressors. All p-values in the ROI analyses have been Bonferroni corrected for multiple comparisons across ROIs. These results suggest that, within the sensitivity limits of our measurement and analysis approach, these specific neuromodulatory nuclei do not strongly encode these particular decision variables related to the value of information or selection in this task.

To comprehensively assess model performance across subcortical regions, we compared the MSE between models within all five subcortical ROIs (VTA, SN, DRN, LC, VSN). First-level GLMs fitted using ANN-derived VoI demonstrated significantly lower MSE than those fitted using linear-derived VoI across all subcortical regions, and significantly lower MSE than those fitted using UCB-derived VoI in four of the five regions (all except LC after Bonferroni correction; Table 3.1). This consistent superiority of the ANN-derived approach across diverse neuromodulatory nuclei further supports the conclusion that the computational principles captured by our hybrid model better reflect the underlying neural mechanisms of information valuation, even in regions where the VoI signal itself may not reach statistical significance.

Table 3.1: Comparison of ANN vs LINEAR and ANN vs UCB models across brain regions

ROI	ANN vs LINEAR		ANN vs UCB	
	Median Diff.	p-value*	Median Diff.	p-value*
VTA	−31.12	0.009	−37.63	0.003
SN	−18.626	6.05×10^{-4}	−18.556	3.33×10^{-4}
LC	−67.461	0.041	−87.452	0.083
DRN	−84.7	4.50×10^{-4}	−90.77	0.002
VSN	−110.112	9.00×10^{-6}	−117.639	1.34×10^{-6}

*Bonferroni corrected p-values. All comparisons based on n=80. Negative values indicate ANN model performed better (lower MSE).

3.2.7 AI and ACC Activity Reflects Information Value and Predicts Sampling Behavior

Our preceding analyses established that the ANN-derived value of information (VoI) predicts both sampling behavior (Figure 3.3B) and BOLD activity (Figure 3.12). This implies a crucial, though as yet untested, link: the trial-by-trial variations in neural activity within VoI-encoding regions should directly correspond to trial-by-trial variations in sampling behavior. To explicitly test this implication and to investigate whether the representational structure of activity in these regions—beyond average activation levels—relates to behav-

ior, we employed Representational Similarity Analysis (RSA). We reasoned that if a brain region computes decision variables guiding information sampling, then the similarity of its neural activity patterns across pairs of trials should mirror the similarity in the extent of sampling behavior observed on those same trials (Figure 3.15). Specifically, based on our univariate results (Figure 3.13), we hypothesized that the neural patterns in AI and ACC would exhibit this correspondence with sampling duration.

To test this, we constructed Representational Dissimilarity Matrices (RDMs) for each participant and session (Figure 3.15B). A behavioral RDM captured the dissimilarity in sampling duration (number of samples taken) between pairs of trials using Euclidean distance. Correspondingly, a neural RDM captured the dissimilarity in BOLD activity patterns between the same pairs of trials, also using Euclidean distance. We focused on the four regions implicated in our univariate analyses (VTA, SN, AI, ACC), and to control for potential confounds related to the size and shape of regions of interest, we used masks matched to the VTA's shape, positioned at the center of each respective target region.

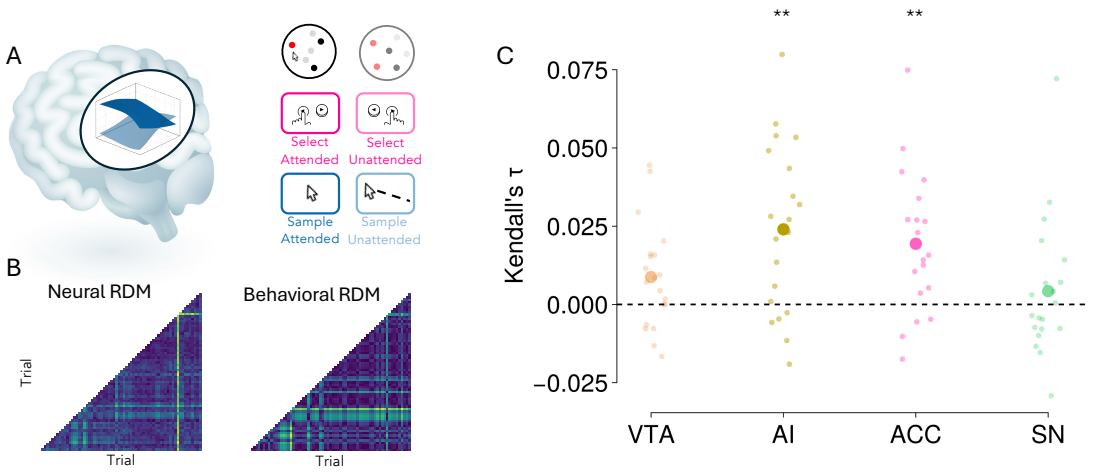

Figure 3.15: Representational Similarity Analysis links neural patterns to sampling behavior. A, Schematic illustrating the relationship between brain activity and behavior. B, Example Neural RDM (left, based on multi-voxel BOLD activity patterns) and Behavioral RDM (right, based on the number of samples taken per trial) for a participant session. Each cell represents the dissimilarity between a pair of trials; higher values (yellow) indicate greater dissimilarity. C, Kendall's τ correlation coefficients quantifying the relationship between neural and behavioral RDMs for VTA, AI, ACC, and SN. Neural RDMs for AI, ACC, and SN were derived using masks matched in shape and size to the VTA ROI to control for region size. Each small dot represents the correlation calculated for one participant session; larger dots indicate the mean correlation across participants. Asterisks denote significant positive correlations across participants (Wilcoxon Signed Rank Test, $^{**}P < 0.01$, Bonferroni corrected).

We then calculated the Kendall's τ correlation between the flattened neural and behavioral RDMs for each session. Statistical testing across participants revealed significant positive correlations between neural pattern similarity and behavioral similarity in both the

AI (Signed Rank Test, $\tau = 0.024$, $P = 0.0006$, Bonferroni corrected $P = 0.0024$) and in the ACC ($\tau = 0.019$, $P = 0.0004$, Bonferroni corrected $P = 0.0017$), but not in the VTA ($\tau = 0.008$, $P = 0.072$) or SN ($\tau = 0.004$, $P = 0.46$, Bonferroni corrected $P = 1.0$) (Figure 3.15C). This suggests that the patterns of activity within AI and ACC, beyond their average activation levels, contain information related to the amount of information participants choose to sample on a given trial. The lack of significant correlation in VTA and SN, despite their univariate associations with decision variables, indicates their trial-by-trial activity patterns may relate less directly to the specific duration of sampling compared to AI and ACC.

3.3 Discussion

Information seeking has been proposed to be a fundamental behavior in humans and other animals [95, 10, 77, 108]. Understanding how and why information is valued and sought can not only help us understand how people interact with information sources as variable as other individuals or the internet [125, 44] but how information seeking can become pathological, maladaptive, or how it can affect mental wellbeing [124, 72]. At the same time, understanding the value of information can help us understand how people and animals resolve decisions; inherent within decision making is a trade-off between, on the one-hand information seeking in order to ensure that the best option is identified and selected and, on the other hand, quick and effective decision making that allows the individual to move on to the next behavior without forgoing other opportunities. If the decision maker deliberates for too long over the first decision, she may forgo the possibility of making a decision about subsequent opportunities. That the utility of information diminishes with time during decision making is consistent with observations of neural activity in the lateral intraparietal (LIP) cortex and superior colliculus that selects the target for an eye movement and brings about the end of deliberation and initiation of the movement [53, 32, 115, 54]. In the current study, however, we examined how the value of information is itself determined.

Four key parameters (Figure 3.9A and Figure 3.9C) determine value of information obtained from the current focus of attention as opposed to value of information at an alternative location. The first and third, β_1 and β_3, set the balance between maintaining attention at the current focus as opposed to exploring anywhere else. The second and fourth, β_2 and β_4, determine how the relative amount of information at the currently attended location and at a specific alternative influence maintenance of attention at the current location and of switching attention to the alternative location respectively. The characterization of value of information in the current study captures features of potential choices that are akin to the uncertainty that a decision maker might have about their value. However, the

precise characterization of value of information was a result of the ANN approach that was adopted; the flexible ANN approach was, first, able to capture non-linear relationships between choice features and value of information and, second, it was able to discover the relationship in the absence of a prior hypothesis about the precise nature of the relationship. Critically, however, the nature of the relationship between the choice features and value of information did not remain implicit and uninterpretable within the ANN but, instead it was rendered interpretable by symbolic regression. While the limit to the range of situations in which the current model of value of information applies is yet to be determined, it is clear that the model has some generalizability; it was able to predict behavior in a very different task but which was also characterized by an explore-exploit dilemma [42] (Figure 3.9B).

There is a growing recognition that purely theory-driven models may sometimes oversimplify complex cognitive processes, while purely data-driven approaches like deep learning often suffer from a lack of interpretability. Recent work explores various ways to bridge this gap, often falling into two main streams. One stream focuses on leveraging neural networks while enhancing interpretability: this includes explicitly integrating ANNs within classic cognitive frameworks [34] or utilizing deep learning architectures specifically designed for interpretability or cognitive plausibility, such as tiny [66] or disentangled [91] recurrent neural networks. A second, less explored, stream aims for interpretability by discovering mathematical equations directly from behavioral data using equation discovery algorithms [46]. The workflow we propose, integrates aspects of both these streams. First, we use a flexible ANN to learn complex input-output mappings without strong a priori constraints. Then, we use symbolic regression to distill these learned mappings into human-readable equations. A natural question that arises is why not apply symbolic regression directly to behavioral data rather than first fitting an ANN. There are several computational and methodological reasons for our two-stage approach. First, direct symbolic regression on the complex mapping from task state to behavior would involve an intractably large search space. By first learning this mapping with ANNs and then applying symbolic regression to the learned function, we factorize this complex problem into manageable components, dramatically reducing the search space from multiplicative to additive complexity. Second, our neural network architecture (Lipschitz-Bounded Deep Network [130]) inherently enforces smoothness constraints on the learned function, providing stability that would not be guaranteed with direct symbolic regression. Third, and crucially, our symbolic regression operates within a broader cognitive model optimized via gradient descent. Embedding a non-differentiable genetic algorithm (symbolic regression) directly within this gradient-based optimization framework would create significant computational challenges. By replacing the value-of-information computation with a differen-

tiable neural network, we transform this into a standard, tractable optimization problem. This approach thus combines the computational efficiency and stability of neural network training with the interpretability benefits of symbolic regression. This approach was central to the current study but it is likely to have wider applicability in cognitive science and neuroscience. We believe it holds significant potential for refining existing theories and discovering novel computational principles across diverse domains, from learning and decision-making to perception and social cognition.

As noted, value of information is related to uncertainty about a decision option and uncertainty has, in turn, been related, at one time or another, to all the major neuromodulatory systems [95, 10, 134, 142, 50, 40, 86, 9]. However, identifying where one system makes a specific or preeminent contribution remains difficult. By using ultra-high field 7T fMRI we recorded from the nuclei – VTA, SN, DRN, LC, and VSN – from which the neuromodulatory systems innervating the forebrain originate. By simultaneously recording from them all we hoped to ascertain if any had an especially strong relationship with value of information. This was the case for VTA (Figure 3.14); the sum of the values of information of the options was associated with significant activity change. At the same time, the sum of the values of the options – the number of red dots, which determined participants' scores and final payouts – was also associated with significant activity modulation but in the opposite direction. Such a pattern of opposed activity change is consistent with VTA balancing the value of making a choice selection against the value of information to be gained from exploring the options further (arbitrating between the top and bottom rows of Figure 2.1A). In light of the classical reward-prediction-error theory of phasic dopamine [104], the positive coding of the value of selection by VTA is expected. The negative coding of the value of information, however, is not directly predicted by RPE theory; its sign also differs from primate single-unit work in which dopamine neurons *positively* signal preference for information about upcoming rewards [9, 10]. We interpret the present pattern instead as encoding the relative advantage of committing over sampling: VTA activity is maximal when expected reward is high and remaining information value is low, exactly the conditions under which a final choice should be made. The SN exhibited a related pattern of activity (Figure 3.14); it also encoded the sum of the values of information of the options, but its activity was also determined by the difference in the values of information of the options. This suggests that SN encodes both the potential for overall information gain but also the relative value of information from the current focus of attention and an alternative (the bottom row of Figure 2.1A). Similar patterns were also found in two cortical regions, ACC and AI (Figure 3.13A and Figure 3.13B). These are the two cortical regions known to project to, or adjacent to, VTA and SN as well as other midbrain and brainstem neuromodulatory nuclei and which have been reported to have activity that is related to

that found in the neuromodulatory nuclei [95, 134, 76, 21, 68, 67, 73, 78, 79]. There are also multisynaptic routes running between these cortical regions and VTA/SN via other brain structures such as the habenula [140, 58]. Importantly, multivariate analysis demonstrated that ACC and AI are especially intimately concerned with information evaluation; the multi-voxel pattern of activity in both ACC and AI covaried with the number of samples of information taken on each trial (Figure 3.15).

Previous studies have identified ACC and VTA and anatomical structures that interconnect them, such as habenula, in evaluation of information. Activity in individual neurons in both habenula and VTA reflects both the value of a stimulus in terms of the reward that it predicts [104, 121, 87] but also value of information [10, 134, 9, 12] in line with the suggestion that VTA may compare the relative advantage to be gained from making a selection and seeking more information about a choice. Activity in ACC has also been linked to information seeking and initiation of behavioral change [123, 82, 134, 94, 96, 62, 70, 119] consistent with the proposal that ACC might arbitrate when it is advantageous to seek more information about a current opportunity and when it is better to pursue an alternative. AI is relatively less investigated but, in line with the current findings, AI and activity in dopaminergic midbrain areas such as SN have been reported when people evaluate whether and when to initiate an action [74, 75].

3.4 Methods

3.4.1 Subjects

Twenty participants (14 females), aged 19 to 32 years, completed the study. All participants were paid £15 per hour and an additional performance-dependent bonus of between £20 and £40 for rewards collected during the task. Each participant provided written informed consent at the beginning of each testing session. Ethical approval was given by the Oxford University Central University Research Ethics Committee (Ethics Approval Reference: R82877/RE001). Behavioral data from all participants were used for the analysis.

3.4.2 Behavioral Analysis

To analyze participants' decision-making behavior, we employed mixed-effects logistic regression analyses to examine three key aspects of choice behavior. First, we investigated how participants decided between staying and sampling information at the current patch versus switching to sample information from the other patch (Figure 2.1A, bottom). This analysis modelled the probability of staying versus switching as a function of the value of information for both the attended and unattended patches, while also accounting for the difference in means between patches and its quadratic term.

$$
\begin{aligned}
logit(stay_{i,j}) =& (\beta_0 + u_{0,j}) + (\beta_1 + u_{1,j}) \cdot voi_a + (\beta_2 + u_{2,j}) \cdot voi_u + \\
& (\beta_3 + u_{3,j}) \cdot first\_visit + (\beta_4 + u_{4,j}) \cdot voi_a \cdot first\_visit + \\
& (\beta_5 + u_{5,j}) \cdot voi_u \cdot first\_visit + (\beta_6 + u_{6,j}) \cdot (\mu_a - \mu_u) + \\
& (\beta_7 + u_{7,j}) \cdot (\mu_a - \mu_u)^2
\end{aligned} \tag{3.1}
$$

where i indexes observations, j indexes subjects, β_k are the fixed effects, and $u_{k,j}$ are the subject-specific random effects. The random effects are assumed to follow a multivariate normal distribution: $\mathbf{u}_j \sim \mathcal{N}(0, \Sigma)$, where Σ is the variance-covariance matrix of the random effects. voi_a and voi_u are the value of information for the attended and unattended patches, respectively. μ_a and μ_u are the observed proportions of red dots in the attended and unattended patches, respectively.

Second, we examined the factors influencing participants' decisions between continuing to sample information versus making a final selection. This model included predictors capturing both the absolute difference and sum of the value of information between patches, including interactions with first visit, as well as the absolute difference and sum of the observed means.

$$
\begin{aligned}
logit(sample_{i,j}) =& (\beta_0 + u_{0,j}) + (\beta_1 + u_{1,j}) \cdot |voi_a - voi_u| + (\beta_2 + u_{2,j}) \cdot (voi_a + voi_u) + \\
& (\beta_3 + u_{3,j}) \cdot first\_visit + (\beta_4 + u_{4,j}) \cdot |voi_a - voi_u| \cdot first\_visit + \\
& (\beta_5 + u_{5,j}) \cdot (voi_a + voi_u) \cdot first\_visit + (\beta_6 + u_{6,j}) \cdot |mean_a - mean_u| + \\
& (\beta_7 + u_{7,j}) \cdot (mean_a + mean_u)
\end{aligned} \tag{3.2}
$$

Third, we analysed participants' final patch selections to understand what drove the choice between attended and unattended patches, using both the observed means and value of information of both patches as predictors. For each of these analyses, we compared three different approaches to computing the value of information: linear, upper confidence bound (UCB), and hybrid approaches. All models incorporated random effects for all predictors grouped by subject and used a logistic link function to account for the binary nature of the choices. This comprehensive modelling approach allowed us to dissect different aspects of the decision-making process while accounting for individual differences through the random effects structure.

$$\begin{aligned} logit(select\ attended) =& (\beta_0 + u_{0,j}) + (\beta_1 + u_{1,j}) \cdot mean_a + (\beta_2 + u_{2,j}) \cdot mean_u + \\ & (\beta_3 + u_{3,j}) \cdot voi_a + (\beta_4 + u_{4,j}) \cdot voi_u \end{aligned} \tag{3.3}$$

We fit each model for the three different approaches to computing the value of information. We used the library MixedModels.jl in Julia to fit the mixed-effects logistic regressions.

3.4.3 Optimal Model

We formulated the information sampling task as a Markov Decision Process (MDP) with states representing the agent's knowledge about dots in each patch, actions corresponding to stay, switch, and selection decisions, and rewards capturing both sampling costs and final choice outcomes. Specifically, each state s contains: the current time step, the number of revealed red dots in each patch, the total number of revealed dots (red + black) in each patch, the current gaze position (left or right), and whether each patch has been visited. The action space consists of four possible actions: stay (continue sampling current patch), switch (move to the other patch), select attended patch, or select unattended patch.

The transition function $T(s'|s, a)$ defines the probability of transitioning to the next state s' when taking action a in the current state s. For sampling actions (stay/switch), the transition function follows a Beta-Binomial distribution where the probability of observing a new red dot is based on the current counts plus a uniform prior ($\alpha = 1, \beta = 1$). Specifically, given n previously revealed dots (r red and $n - r$ black) in the attended patch, the probability of observing a new red dot follows $\frac{r+1}{n+2}$, while the probability of observing a black dot follows $\frac{n-r+1}{n+2}$. For selection actions, the episode terminates. Note that this formulation treats the proportion of red dots in each patch as *a priori* independent across patches, which differs from the actual experimental design in which the pairwise difference between patch proportions was constrained to 0.1, 0.2, or 0.3 (Section 2.2). An ideal observer with knowledge of this constraint could exploit it through a joint prior over the two patch proportions, so that observations from one patch would inform beliefs about the other; the optimal policy considered here therefore provides a conservative reference that ignores this source of cross-patch information.

The reward function $R(s, a)$ assigns different rewards based on the type of action. For sampling actions, the agent gets an immediate negative reward when she chooses to sample an additional piece of information from the option she was currently attending (costs for staying) and a bigger immediate negative reward when the agent chooses to switch, and sample from the other patch (cost of switch). Participants in the real task experi-

enced a switch delay of 13.3 times longer than the stay delay (e.g. 2000ms versus 150ms). To reflect this difference in the reward function, we set the cost of switch to be 13 times bigger than the cost of stay. For selection actions (choosing either the attended or unattended patch), the reward is computed based on the probability that the selected patch has a higher true proportion of red dots. Specifically, given the current evidence for the attended patch (α_1, β_1) and unattended patch (α_2, β_2), we compute:

$$p_A = P(Beta(\alpha_1 + 1, \beta_1 + 1) > Beta(\alpha_2 + 1, \beta_2 + 1)) \tag{3.4}$$

where the probability is calculated using a normal approximation based on the difference in means normalized by the square root of the sum of variances. The final reward for selecting the attended patch is then $p_A - (1 - p_A)$, and conversely $(1 - p_A) - p_A$ for selecting the unattended patch.

We computed the value function $V(s)$ using backwards induction over a finite horizon for each possible initial state configuration. Since trials could start with different proportions of green dots in each patch (ranging from 0.05 to 0.3 in increments of 0.01), we computed separate optimal policies for a subset of 36 combinations of initial green dot proportions in the left and right patches (6 possible proportions per patch: 0.1, 0.14, 0.18, 0.22, 0.26, 0.3). For each initial condition, we performed backwards induction over the complete state space with the following parameters: maximum number of steps = 100, maximum number of dots per patch = 100, step size (dots revealed per sampling) = 2, cost of staying = 0.005, cost of switching = 0.065, and a uniform prior ($\alpha = \beta = 1$) for the Beta-Binomial transition function. The resulting lookup tables mapped each state to its optimal action, allowing us to simulate optimal behaviour given the reward function we defined.

The optimal value function $V^*(s)$ was calculated recursively by iterating over all possible states and actions.

$$V^*(s) = \max_{a \in \mathcal{A}} \left[R(s, a) + \sum_{s' \in \mathcal{S}'} T(s'|s, a) \cdot V^*(s') \right] \tag{3.5}$$

The optimal policy $\pi^*(s)$ selects actions that maximize the expected sum of rewards from each state.

For each state s, the optimal policy $\pi^*(s)$ was computed using the Bellman equation:

$$\pi^*(s) = argmax_{a \in \mathcal{A}} \left[R(s,a) + \sum_{s' \in \mathcal{S}'} T(s'|s,a) \cdot V^*(s') \right] \tag{3.6}$$

where $\mathcal{A}$ is the set of possible actions, $R(s,a)$ is the immediate reward, $T(s'|s,a)$ is the transition probability to state s', and $V(s')$ is the value function at the next state. $\mathcal{S}'$ represents the set of possible next states given the current state and action. This yielded a lookup table mapping states to optimal actions that we used to simulate ideal observer behaviour in the task.

3.4.4 Fitting Symbolic and UCB Models to independent Two-Armed Bandit Task Datasets

To assess the generalizability of the insights derived from our modeling approaches, and to specifically compare the UCB model with the Symbolic model (derived from the symbolic regression analysis), we evaluated their performance on two additional, independent datasets from a previously published study by Gershman et al. (2018; referred to as "data1" and "data2" in our analyses, N=44 and N=44 respectively after excluding one subject from data1 that led to an extremely large value of loss when trying to fit the UCB model to their choices). These datasets involved similar information sampling tasks, providing a strong test for cross-task generalization.

For each participant in these external datasets, we fitted three models: the standard UCB model, an augmented UCB (AUCB) model, and the Symbolic model. The standard UCB model used the formulation $U = c\sqrt{\frac{2\log(t)}{N}}$, where c is a scaling parameter, t is the time step, and N is the number of times the option has been sampled. To control for the possibility that the Symbolic model's superior performance was due to its additional offset parameter rather than its functional form, we also tested an augmented UCB model with the formulation $U = c_0 + c_1\sqrt{\frac{2\log(t)}{N}}$, where c_0 provides a baseline offset and c_1 scales the exploration bonus. This AUCB model has the same parametric flexibility as the core components of our Symbolic model, allowing for a more controlled comparison of functional forms.

To evaluate how well each model could predict each participant's choices, we employed a Leave-One-Out (LOO) cross-validation procedure on a per-subject basis. This involved iteratively training each model on all but one trial for a given subject and then testing its predictive accuracy on the held-out trial. This process was repeated such that every trial served as part of a test set.

The primary metric for evaluating model performance was the average prediction loss

(e.g., cross-entropy loss) on these held-out test trials, calculated for each subject for both models. This provides a measure of how well each model generalized to unseen data within each participant from the external datasets.

The per-subject average LOO losses for the Symbolic model and the UCB model were then statistically compared to determine if one model offered consistently better predictions of choice behavior on these novel datasets. This comparison involved paired t-tests on the differences in mean losses per subject, as visualized in the comparative loss plot.

3.4.5 Imaging data acquisition

Structural and functional MRI data was collected with a Siemens 7 Tesla MRI scanner. High-resolution functional data were acquired with a multiband gradient echo T2* echo planar imaging sequence with 1mm isotropic voxels, multiband acceleration factor 2, repetition time (TR) = 1.378s, echo time (TE) = 27ms, flip angle = 90°, and GRAPPA acceleration factor 2. The parameters were selected to maximise signal-to-noise ratio in subcortical areas. To accommodate the high temporal and spatial resolution of the protocol, functional scans had a limited field of view (FOV) oriented at 45 degrees with respect to the AC-PC line (36 slices). The FOV captured all regions of interest in the midbrain, brainstem and cortex. Before acquiring the task-related functional scan, we acquired a presaturation single-measurement, whole-brain functional scan with the same orientation. The pre-saturation scan was used to facilitate registration of the limited-FOV task-related functional scan to the whole brain. Structural data were acquired using a T1-weighted MP-RAGE sequence with 0.7mm isotropic voxels, GRAPPA acceleration factor 2, TR = 2200ms, TE = 3.02ms, and; inversion time (TI) = 1050ms. To correct distortions arising from inhomogeneities in the magnetic field, a fieldmap sequence was acquired with 2mm isotropic voxels, TR = 620ms, TE1 = 4.08ms, and TE2 = 5.1ms. To account for the effects of physiological noise on functional MRI data, participants were fitted with a pulse oximeter and respiratory bellows that acquired cardiac and respiratory timeseries at 50Hz using a BioPac MP160 device (BIOPAC Systems Inc., USA).

3.4.6 fMRI data preprocessing

Preprocessing of fMRI data was performed with the FMRIB Software Library [65, 110]. The Brain Extraction Tool [109] was used to separate brain from non-brain matter in structural and functional images. Functional images were normalised, spatially smoothed (Gaussian kernel with a 3mm full-width half-maximum) and temporally high-pass filtered (3 dB cut-off = 100s), and artefacts arising from head motion were removed using MCFLIRT [64]. Registration of task-related functional images to Montreal Neurological In-

stitute (MNI)-space was performed in three stages: 1. The task-related limited-FOV EPI was registered to the pre-saturation whole-brain EPI using FMRIB's Linear Image Registration Tool with 6 degrees of freedom transformation. 2. The whole-brain EPI was registered to the subject-specific structural images using Boundary-Based Registration (BBR) incorporating fieldmap correction [49]. 3. Subject-specific structural images were registered to a 1mm resolution Standard MNI template with FMRIB's Non-linear Registration Tool (FNIRT; [65]).

3.4.7 fMRI data analysis

Statistical analysis of whole-brain functional data was performed at three levels using FMRIB's Expert Analysis Tool (FEAT; [65, 110]). In the first level, a univariate general linear model was used to compute parameter estimates for each regressor in each session [136]. Contrast and variance estimate for each parameter in each participant were subsequently combined in a fixed-effects analysis conducted at the second level. Finally, a random-effects analysis was conducted at the third level, where subject-identity was a random effect [137]. Significance testing was performed with cluster-correction, a cluster significance threshold of $P = .001$, and a voxel inclusion threshold of $z = 3.1$. Data were pre-whitened before analysis to account for temporal autocorrelations in BOLD signal. We performed one whole-brain analysis for each of the three different approaches to computing the value of information. The GLM identified voxels where BOLD signal represented the value of staying, switching, or selecting the attended or unattended patch.

$$\begin{aligned} BOLD = &\beta 1 \cdot value\_of\_stay + \beta 2 \cdot value\_of\_switch + \\ &\beta 3 \cdot \mu_{attended} + \beta 4 \cdot \mu_{unattended} + \beta 5 \cdot outcome + \epsilon \end{aligned} \tag{3.7}$$

All regressors were convolved with a double-gamma hemodynamic response function. Further non-task confound regressors were added to reduce noise in BOLD signal, including: (1) head motion parameters estimated using MCFLIRT during pre-processing [64]; (2) regressors for voxel-wise estimates of physiological noise arising from cardiac and respiratory activity, estimated using FSL's Physiological Noise Monitoring (PNM) tool [13], and; (3) regressors for motion outliers, indicating volumes with head motion that could not be corrected with linear methods.

To compare the three different approaches to computing the value of information (linear, UCB, and ANN), we ran identical GLM analyses for each approach, varying only in how the value of staying and switching were computed. For each of the 80 sessions (4 sessions $\times$ 20 subjects), we fitted three separate GLMs using the value computations from

each model. We then computed the Mean Squared Error (MSE) between the GLM predictions and the actual BOLD signal for each session, resulting in 80 MSE values per model. To statistically compare the performance of the three approaches, we conducted non-parametric Wilcoxon signed-rank tests on these paired MSE values, allowing us to assess whether one approach consistently provided better predictions of neural activity than the others.

3.4.8 ROI analysis

To analyze activity in specific regions of interest (ROIs: VTA, SN, DRN, LC, VSN), we extracted voxel-wise beta estimates (effect sizes) and their corresponding standard errors from the first-level GLM analysis for each relevant contrast (contrast of parameter estimates, or "cope" in FSL). For each ROI and contrast, we fitted a linear mixed-effects model using the MixedModels.jl package in Julia to estimate the group-level effect. The model formula was $effsize \sim 1 + (1|subject) + (1|voxel)$, predicting the voxel-wise beta estimate (effsize) with a fixed intercept (representing the group effect) and random intercepts for subject and voxel to account for inter-subject and inter-voxel variability. Crucially, these models were weighted by the inverse variance of the beta estimates ($1/standard\_error^2$) to give more influence to more precise measurements at the voxel level. This approach yields a robust estimate of the average activation (the fixed intercept) for each contrast within each ROI across the group, while appropriately accounting for different sources of variance. P-values for the fixed intercept were extracted and corrected for multiple comparisons across the tested ROIs using the Bonferroni method.

3.4.9 RSA analysis

Representational Similarity Analysis (RSA) was employed to test whether the similarity structure of neural activity patterns related to the similarity structure of behavioral sampling duration on a trial-by-trial basis within each session. For each participant and session, we constructed two types of Representational Dissimilarity Matrices (RDMs). First, a behavioral RDM was computed based on the number of samples taken in each trial; the dissimilarity between any pair of trials (i, j) was defined as the Euclidean distance between their respective sample counts. Second, neural RDMs were constructed for specific regions of interest (VTA, SN, AI, ACC). To control for ROI size and shape differences, we used masks matched to the VTA's shape, positioned within each target region. Trial-level BOLD activation patterns (z-statistics from the first-level GLM) were extracted for each trial within these masks. The dissimilarity between neural patterns for any pair of trials (i, j) was calculated as the Euclidean distance between their multi-voxel activation vectors.

Both behavioral and neural RDMs were then vectorized by taking their upper triangular elements (excluding the diagonal). The correspondence between neural patterns and behavior was quantified by computing the Kendall's τ correlation between the vectorized neural RDM and the vectorized behavioral RDM for each session. These session-level correlation coefficients were averaged within each participant for each ROI. Group-level significance was assessed using a one-sample Wilcoxon Signed Rank Test against zero across participants for each ROI, testing for positive correlations. P-values were Bonferroni corrected for the number of ROIs tested.

4

Generalization

4.1 Introduction

When entering a new airport, we do not need to learn everything from scratch. Despite the specific layout, people can navigate effectively by leveraging their knowledge of other airports. We recognize the general structure, and apply this schema with new specific content.

Animals across species demonstrate this capacity for rapid generalization from limited experience. Classic studies of "learning sets" in monkeys showed that after solving hundreds of discrimination problems, animals could solve new problems in just one or two trials [55]. This phenomenon extends beyond primates: humans organize conceptual knowledge using schema-like representations [25], rodents develop complementary task representations that support flexible behavior [102], and spatial mapping systems are repurposed for abstract reasoning [35, 7]. The central question is how biological systems accomplish this rapid adaptation when the abstract structure remains constant but specific content changes.

In contrast, artificial neural networks often require vast amounts of data to achieve comparable performance. While deep learning has produced impressive results in many domains, advances frequently come from scale (e.g. larger datasets, bigger models, more compute) rather than from architectural principles that enable learning with less data, as biological systems do. One explanation for the better performance of biological agents is that they can rely on episodic memory to recall specific past experiences to guide decisions in novel situations.

4.1.1 An Experimental System for Studying Generalization

To investigate the mechanisms underlying rapid generalization, we examine behavior in a controlled experimental setting. Two macaque monkeys learned a series of six related decision-making tasks. Each task required learning associations between visual stimuli and reward magnitudes through a multi-stage trial structure.

When animals achieved criterion performance on one sub-task, they transitioned to the next, which maintained the identical abstract structure (number of options, number of reward levels, trial sequence) but introduced entirely new visual stimuli. Animals thus faced the challenge of relearning which specific sensory features mapped to which reward magnitudes, while the underlying task structure remained constant. This design provides a controlled setting where abstract structure is preserved while specific sensory-reward associations change, allowing us to study mechanisms that support generalization. Detailed task specifications are provided in the Methods.

4.1.2 The Computational Challenge

Modeling this behavior presents two computational challenges. First, deep RL has historically relied on minibatch training and replay buffers to maintain learning stability. However, this creates a fundamental mismatch with biological learning: animals learn from a continuous stream of experience, updating their representations after each interaction with the environment. Recently, streaming deep RL methods have been developed that enable weight updates after every simple $s \to a \to s' \to r$ interaction, just as an animal would [36]. We adopt this streaming approach to maintain biological plausibility of the learning process.

Second, standard RL with incremental value updates appears insufficient for the rapid generalization observed in animals. Biological systems leverage episodic memory. Recent evidence suggests that grid cell scaffolds, spatial coordinate systems in the entorhinal cortex originally evolved for navigation, can be repurposed for abstract cognitive maps that support both spatial and non-spatial memory (Chapter 1). The grid cell system generates a structured, low-dimensional representation space that provides a substrate for organizing experiences and supporting generalization. However, the precise mechanisms by which episodic memory interacts with value-based decision-making to enable rapid adaptation remain unclear.

4.1.3 Model Architecture

We developed a computational model that integrates an action selection system with an episodic memory component. The model combines a recurrent neural network (RNN) ac-

tor for selecting actions with a Vector Hippocampal Scaffolded Heteroassociative Memory (VectorHaSH) system for episodic memory [20]. The VectorHaSH architecture factorizes the creation of stable dynamical states (via a prestructured grid cell network) from the storage of content (via heteroassociative weights). Grid cells generate exponentially many stable states that can scaffold episodic memories, and this spatial coordinate system can be repurposed for abstract cognitive domains beyond navigation.

In our implementation, the grid state is updated via path integration using two velocity dimensions: goal progress (advancement through the trial sequence) and state value (expected reward magnitude). Critically, the agent must learn to estimate these velocity signals from the current observation and action, rather than receiving them directly from the environment. The grid state is then projected to a hippocampal state representation $\mathbf{h}$ through a fixed transformation $\mathbf{W}_{hg}$, and this hippocampal state serves as input to the RNN actor Figure 4.4.

This architecture enables generalization through abstraction. The grid-hippocampal projection creates representations of abstract concepts (states corresponding to "low," "medium," and "high" expected reward) that are independent of the specific sensory features of the images. When the task transitions to a new sub-task with novel images, the model only needs to associate the new sensory information with one of these pre-existing abstract value concepts. The RNN can then leverage its previously learned policy over abstract reward states, requiring minimal additional learning. In contrast, a simple RNN without VectorHaSH must relearn the entire sensory-action mapping for each sub-task, effectively ignoring the preserved task structure.

4.1.4 Two Mechanisms for Memory-Guided Generalization

We propose and compare two mechanisms by which the episodic memory system might support generalization. Both mechanisms use the same core architecture (velocity-based grid state updating with projection to hippocampal states) but differ in whether and how episodic memory encoding and retrieval occur.

The first mechanism, which we term *feedforward state update*, operates similarly to recent multi-region brain models of spatial decision-making [139]. During the decision phase, the model computes velocity signals (goal progress and state value) from the current observation and action, updates the grid and hippocampal states via path integration, and provides the hippocampal state as input to the RNN actor. There is no explicit memory encoding or retrieval; instead, abstract value information flows continuously through the grid-hippocampal-RNN pathway as representations are updated online.

The second mechanism, which we term *backward memory binding and retrieval*, augments the feedforward architecture with a second VectorHaSH system dedicated to sen-

sory information, enabling explicit episodic memory binding. The key idea is to link reward outcomes to the states that caused them. During each trial, the model accumulates a causally-weighted memory trace by computing causal attribution weights for each observation. At reward delivery, this accumulated causally-weighted trace serves as a retrieval cue for the sensory VectorHaSH, which performs pattern completion to retrieve the stored stimulus representation most similar to the trace. The retrieved representation is then associated with the observed reward magnitude through supervised learning of a reward velocity network. During subsequent decision phases, this network provides value estimates for each option based on retrieved stimulus-reward associations. This mechanism explicitly separates memory formation (backward binding at outcome) from memory use (retrieval during choice).

Both mechanisms support rapid generalization by providing abstract representations to the action selection system, but they differ in their memory dynamics. The feedforward mechanism relies on continuous updating of abstract value states, while the binding and retrieval mechanism explicitly encodes and recalls specific stimulus-reward associations. These different dynamics lead to distinct predictions about neural activity patterns, which we discuss in the final section of this chapter.

4.1.5 Contributions and Chapter Organization

This chapter makes four contributions to understanding how episodic memory supports generalization. First, we present a computational framework that integrates VectorHaSH episodic memory with streaming deep RL, where the agent learns to estimate abstract velocity signals from sensory observations. Second, we demonstrate how grid cell scaffolds enable rapid adaptation by abstracting specific sensory-reward mappings into reusable conceptual representations of reward magnitude. Third, we compare two biologically plausible mechanisms that achieve equivalent behavioral generalization but differ in their memory encoding and retrieval dynamics. Fourth, we show that both mechanisms with VectorHaSH substantially outperform RNN-only models in generalizing across sub-tasks, suggesting that episodic memory scaffolds are necessary for efficient transfer learning in this setting.

The remainder of this chapter is organized as follows. We first present behavioral results comparing the a baseline RNN-only model versus the two VectorHaSH mechanisms (Results). We then discuss neural predictions that could distinguish between the two memory mechanisms (Discussion). Finally, we describe the task structure and model architecture in detail (Methods).

4.2 Results

4.2.1 Experimental Design

To investigate how episodic memory supports rapid generalization, we examined behavior in a decision-making task where animals learned to associate visual stimuli with reward magnitudes. The task structure remained constant while specific stimulus-reward mappings changed across sub-tasks, creating a controlled setting for studying generalization mechanisms.

Figure 4.1 illustrates both the experimental task performed by animals (top panel) and the simplified version used for computational modeling (bottom panel). Each trial proceeded through multiple stages. Animals first were presented with a black screen for an inter-trial interval (Stage 0) where they were required to keep their hands away from the response sensors. Then, they entered a two-alternative-choice phase (Stage 1) where they selected between two visual stimuli. The chosen stimulus was displayed during a transition period (Stage 2), followed by a motion discrimination task (Stage 3), and finally outcome delivery (Stage 4). During stage 1, two visual stimuli were randomly selected from a set of three, and each stimulus was associated with different reward magnitudes: low (0 drops), medium (1-3 drops), or high (4-9 drops), with arbitrary mappings that remained fixed within each sub-task but changed across the six sequential sub-tasks.

The models were trained on a simplified four-step version that omitted the motion discrimination component to focus on the generalization mechanisms. At each time step, the model selected from three possible actions: choose left option (action 1), choose right option (action 2), or wait (action 3). At stage 0 (inter-trial interval), stage 2 (transition display), and stage 3 (outcome delivery), the optimal action was to wait (action 3). At stage 1 (first-stage choice), the correct action depended on the relative values of the two presented options: select action 2 (right) when the higher-value stimulus appeared on the right (e.g., low vs. medium, low vs. high, or medium vs. high), or select action 1 (left) when the higher-value stimulus appeared on the left (e.g., medium vs. low, high vs. low, or high vs. medium).

Although both stimuli appeared simultaneously on the screen (one on the left, one on the right), the model received them as a sequential stream, first one stimulus (e.g., s_L), then the other (e.g., s_R), with presentation order randomized across trials. The model thus had to decide whether to select the first or second fixated stimulus, simulating the gaze patterns animals exhibit when evaluating options before committing to a choice. This task maintained the core challenge of learning which visual stimuli corresponded to which reward magnitudes, and leveraging the preserved structure (three stimuli, three reward levels) when transitioning to new sub-tasks with novel visual features.

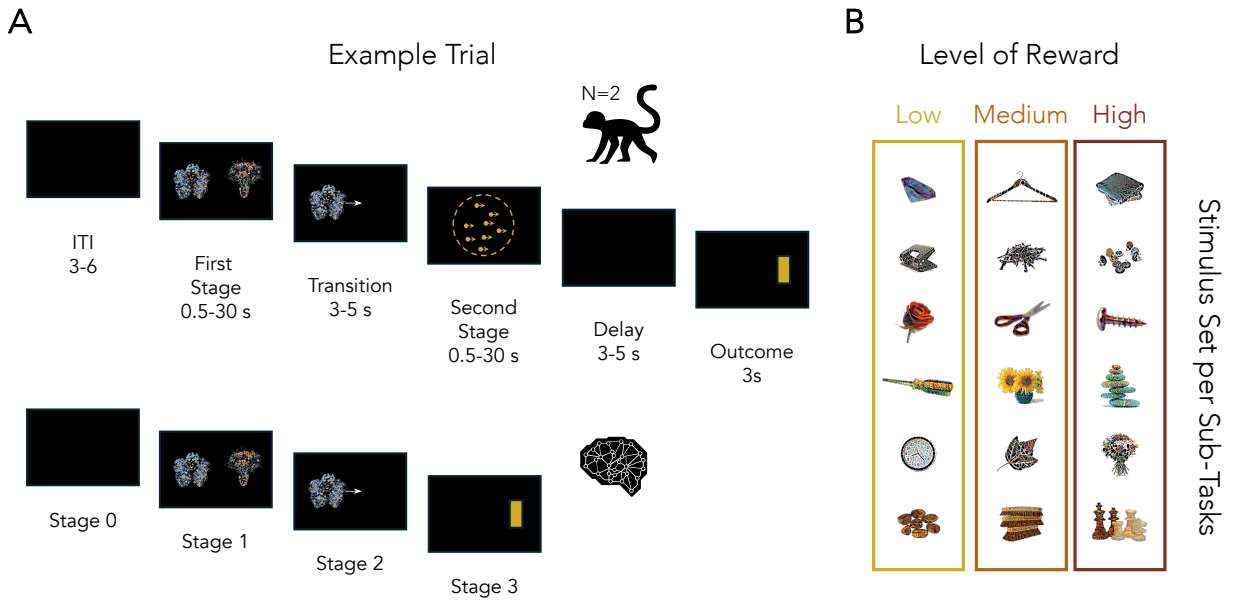

Figure 4.1: Task structure for animal experiments and computational modeling. Top: Animals performed a five-stage task including fixation, first-stage choice between visual stimuli, transition display, motion discrimination, and outcome delivery. Three stimuli (shown as colored shapes) were associated with low, medium, or high reward magnitudes. Bottom: Computational models performed a simplified four-stage version omitting motion discrimination, focusing on learning stimulus-reward associations and generalizing this structure across sub-tasks with novel visual features.

4.2.2 Behavioral Evidence for Generalization

Both animals demonstrated clear evidence of meta-learning across the six sequential sub-tasks. When initially presented with novel visual stimuli, animals required hundreds of trials to reach criterion performance (here defined as each animal's own asymptotic accuracy θ, computed as the mean correct rate across the last two sessions of each sub-task; rolling accuracy was computed using a centered window equal to the average number of trials per session). However, as animals progressed through subsequent sub-tasks, learning became progressively more efficient.

Figure 4.2 shows, for each animal, the rolling accuracy across sub-tasks (top row) and the number of trials needed to first reach the asymptotic threshold θ in each sub-task (bottom row). Animal A required approximately 1,300 trials to reach θ in the first sub-task and 250–500 trials in later sub-tasks; Animal B showed a comparable acceleration, from approximately 2,600 trials in the first sub-task to roughly 500–1,000 trials in later sub-tasks. A power-law fit to the trials-to-threshold values (dashed line, shaded prediction band) summarises this rapid initial drop followed by progressive refinement. This progressive reduction in trials-to-criterion demonstrates that animals were not simply relearning arbitrary stimulus-reward associations from scratch in each sub-task, but were instead ap-

plying learned structural knowledge to new stimulus sets.

The animals' behavior is coherent with a schema-like representation of the task structure: they might have developed abstract representations of reward magnitude categories (low, medium, high) to rapidly associate with new sensory features. This result provides the behavioral foundation that our episodic memory models aim to explain mechanistically.

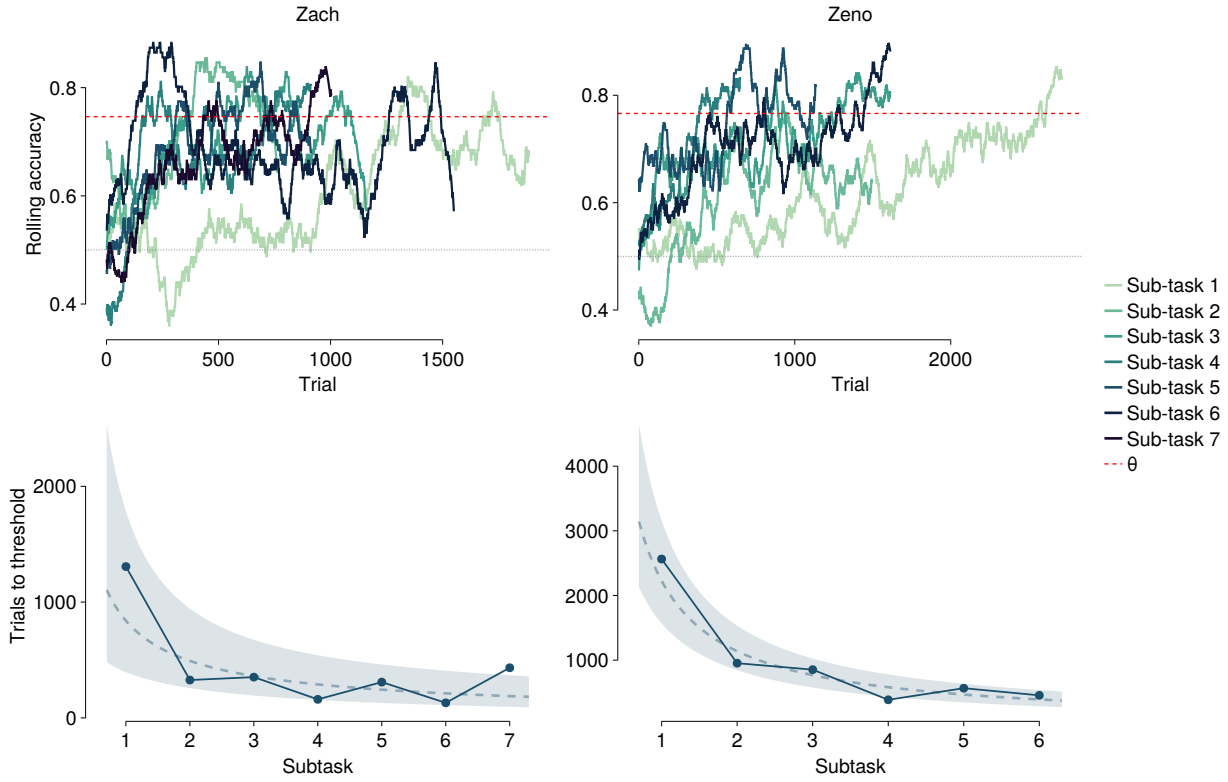

Figure 4.2: Meta-learning across sub-tasks in monkey behaviour. Top row: rolling accuracy (centered window equal to the average number of trials per session) for Animal A (Zach, left) and Animal B (Zeno, right) across sequential sub-tasks with novel visual stimuli but identical task structure (light $\rightarrow$ dark indicates sub-task index). The dashed red line marks each animal's asymptotic accuracy threshold θ, defined as the mean correct rate across the last two sessions of each sub-task. Bottom row: number of trials needed to first reach θ in each sub-task; the dashed grey line shows a power-law fit $y = ax^b$ with shaded ± 1 standard error band of the fit.

4.2.3 Computational Models of Generalization

We developed and compared three computational models to investigate the mechanisms underlying generalization: a baseline vanilla recurrent neural network (RNN) model without episodic memory (M0), and two variants that integrated Vector Hippocampal Scaf-

folded Heteroassociative Memory (VectorHaSH) [20] systems with distinct memory dynamics (M1f and M1b).

Baseline RNN-Only Model (M0)

The baseline M0 model learned to map current sensory input and previous action to action probabilities using streaming deep RL (see 4.4.3). M0 had no access to an explicit episodic memory module, and relied on the RNN's recurrent dynamics to integrate sensory information and guide decisions.

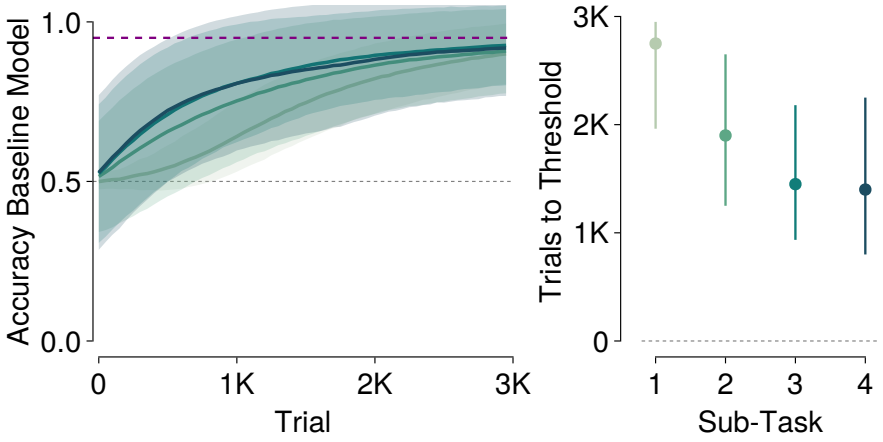

Figure 4.3: Baseline RNN model demonstrates modest meta-learning. Left: Learning curves showing mean accuracy across four sub-tasks (light to dark curves represent Tasks 1-4). Lines show mean accuracy across training runs, with shaded bands indicating ±1 standard deviation. Right: Median trials-to-criterion (reaching 95% accuracy) for each sub-task. Error bars represent the 15th to 85th percentile range across training runs.

Figure 4.3 shows the M0 model's performance across four sub-tasks. As the sub-tasks progressed, M0 learned on average faster to make the correct action in stage 1 of the task. As shown in the left panel of Figure 4.3, darker curves (later sub-tasks) reach asymptotic performance faster than lighter curves (earlier sub-tasks), and the right panel shows this improvement as trials-to-criterion, indicating a reduction from approximately 2,700 trials in Task 1 to 1,400 trials by Task 4. This nearly two-fold reduction demonstrates that pure RNN architectures can extract transferable task structure through their recurrent dynamics.

This meta-learning behavior in vanilla RNNs is expected [47]. Despite changing visual features, there is an inherent structure in the network's input that is preserved across sub-tasks (e.g. the spatial arrangement of options).

However, the M0 model's improvement is modest compared to the dramatic acceleration observed in animals. While animals also begin with approximately 1,300 trials in their first

sub-task (comparable to M0's 2,700 trials), they achieve a much more dramatic reduction, dropping to under 200 trials by the sixth sub-task, a more than 6-fold improvement. This indicates a qualitative difference in the learning trajectory: M0 exhibits a smooth, gradual reduction in trials-to-criterion across all four sub-tasks (2,700 $\rightarrow$ 1,900 $\rightarrow$ 1,500 $\rightarrow$ 1,400), reflecting gradual acquisition of the invariant structure of the input and gradual learning of the policy in response to new stimuli. In contrast, animals display a markedly different pattern, where a prolonged learning period for the first sub-task followed a sharp acceleration in subsequent sub-tasks. One interpretation is that animals are using mechanisms beyond gradual learning through recurrent dynamics alone.

The next two models aim to bridge this gap by augmenting the RNN with an explicit medial entorhinal-hippocampal system (M1f and M1b). Rather than relying only on implicit abstraction through recurrent states, these models implement structured episodic memory that can rapidly encode and retrieve abstract task structure, potentially providing a mechanistic account of the sharp acceleration characteristic of animal meta-learning.

VectorHaSH Models: Memory-Guided Generalization

To enable rapid generalization, we augmented the RNN policy with a VectorHaSH episodic memory system [20]. The core architectural innovation involves representing the task in an abstract two-dimensional abstract map with axes corresponding to goal progress (position within the trial sequence) and state value (expected reward magnitude). This structured representation space provides a substrate for organizing experiences and supporting generalization through two complementary mechanisms.

Reusable Concepts Rather than learning direct stimulus-action mappings, the two M1 models learn to estimate velocity signals in an abstract space and use these velocities to update a grid cell state via path integration. The grid state represents the agent's current position along two non-spatial dimensions: how far through the trial sequence (goal progress) and what reward magnitude to expect (state value). This grid state is then projected through a fixed random transformation to create hippocampal representations that serve as input to the policy RNN.

The critical insight is that this architecture separates the problem of learning abstract task structure from the problem of associating specific sensory features with abstract concepts. The grid-hippocampal projection creates stable representations corresponding to "low reward state," "medium reward state," and "high reward state" that remain invariant across changes in visual features. When transitioning to a new sub-task, the model only needs to learn which new stimulus corresponds to which pre-existing abstract category, rather than relearning the entire decision policy.

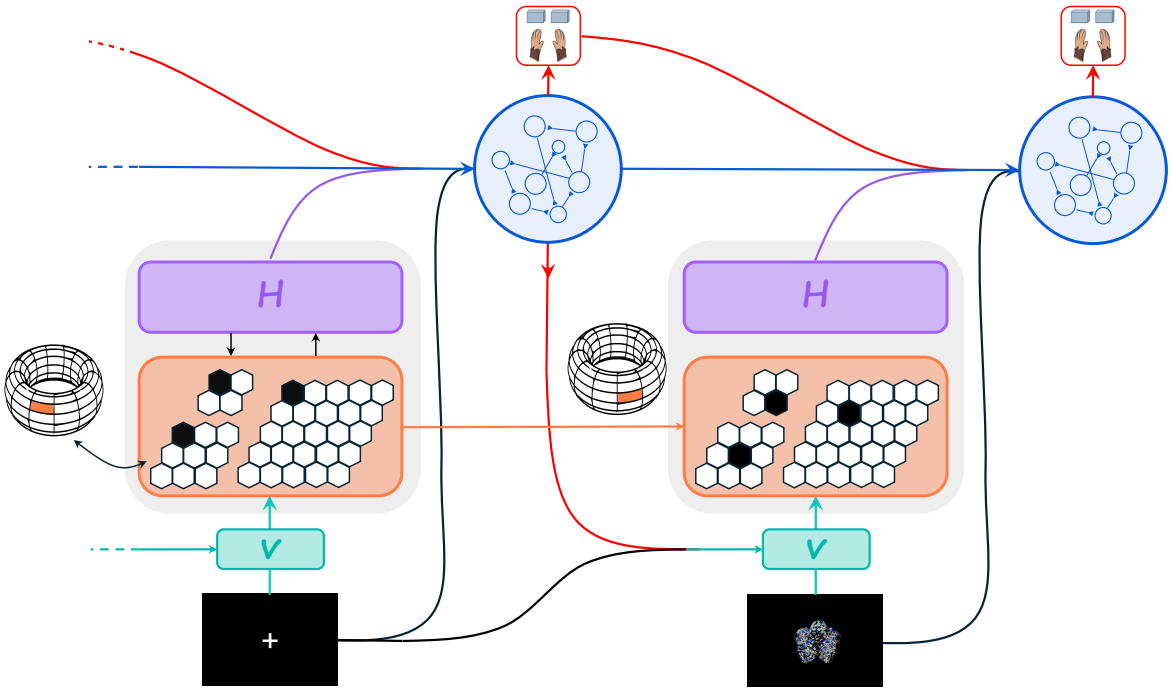

Figure 4.4: Architecture of M1 models. The diagram illustrates the basic structure shared by both M1f and M1b models. Sensory inputs (bottom black squares) are processed through velocity networks (V, teal boxes) that compute abstract velocity signals along two dimensions: goal progress and state value. These velocities drive path integration updates to grid cell representations (orange boxes with hexagonal patterns). Grid states are projected through fixed random transformations to create hippocampal representations (H, purple boxes), which provide abstract value information to the RNN policy network (blue circle). The RNN outputs action probabilities (top red boxes). The left and right pathways represent sequential processing of the two available options during choice phases.

M1f: Feedforward State Updates.

The M1f model implements continuous feedforward updating of abstract representations. During the choice phase, the model simulates sequential gazes at each available option (analogously to the baseline model). For each option viewed, the critic network estimates its expected value, this value estimate is discretized and used as velocity along the value axis to update the grid state, and the updated grid state is projected to a hippocampal representation. The policy RNN receives both the raw sensory tokens (preserving stimulus-specific information) and the hippocampal representations (providing abstract value information) for each option.

Between trial stages, goal progress velocity is computed from a learned distance network that estimates how many steps remain until reward delivery. The distance is defined recursively: $D(o_t, a_t) = \min_{a'} D(o_{t+1}, a') + 1$, with base case $D(o_t, a_t) = 0$ when the current

state-action pair leads to reward. The goal progress velocity is then:

$$v_{\text{progress}} = \min_{a'} D(o_{t-1}, a') - D(o_t, a_t) \tag{4.1}$$

representing how much closer the agent moved toward the goal.

This feedforward mechanism provides abstract value representations to the policy continuously throughout the trial without explicit memory encoding or retrieval operations. The abstraction emerges naturally from the grid-hippocampal projection structure combined with critic-based value estimation.

M1f Learning Performance and Internal Representations Figure 4.5 shows the M1f model's performance across four sub-tasks. In striking contrast to M0's modest improvement, M1f demonstrated dramatic acceleration in learning. The first sub-task required approximately 2,825 trials to reach criterion (95% accuracy), comparable to M0's 2,700 trials. However, by the second sub-task, learning accelerated substantially, with trials-to-criterion dropping to 500. For the subsequent sub-tasks, this further decreased to around 200 trials. This represents a 14-fold improvement from Task 1 (2,825 $\rightarrow$ 200 trials) and a 7-fold advantage over M0's final performance (200 vs 1,400 trials). This rapid transfer approaches the dramatic acceleration observed in animal behavior.

This acceleration arose from the model's ability to leverage pre-existing abstract value representations. Rather than relearning which stimuli to select in each sub-task, M1f only needed to learn the value estimates for new visual features (via the critic network), and could immediately apply its previously learned policy over abstract hippocampal states that represented low, medium, and high reward options.

To understand how M1f achieved this generalization, we examined the internal representations learned by the critic and distance networks that provide velocity signals for updating the abstract grid state.

Value Representations via the Critic Network. I trained a critic network using temporal-difference learning with the TD error:

$$\delta_t = R_t + \gamma V(o_{t+1}) - V(o_t) \tag{4.2}$$

where R_t is the immediate reward, $\gamma = 0.99$ is the discount factor, and $V(o_t)$ is the value estimate for observation o_t.

Within each sub-task, the critic rapidly learns to discriminate between the three reward levels (Figure 4.6), displaying higher efficiency as the sub-tasks progressed. These value estimates provide the velocity signals along the state value axis of the abstract map.

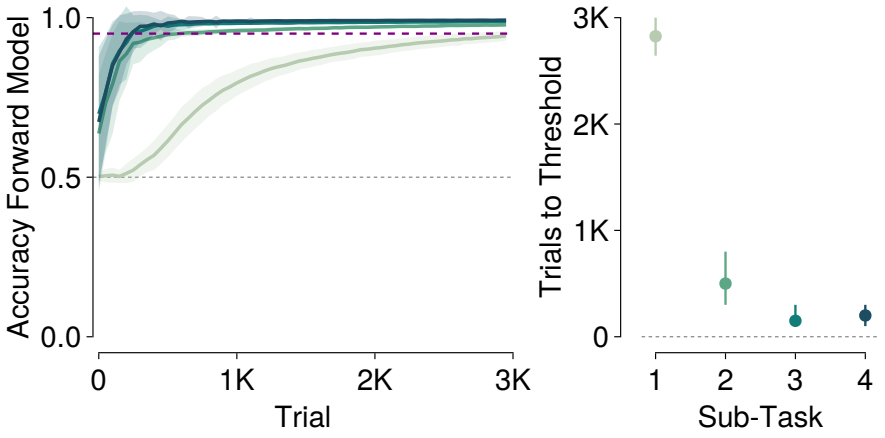

Figure 4.5: VectorHaSH model M1f demonstrates rapid generalization across sub-tasks. Left: Learning curves showing mean accuracy across four sub-tasks (light to dark curves represent Tasks 1-4). Lines show mean accuracy across training runs, with shaded bands indicating ±1 standard deviation. Right: Median trials-to-criterion (reaching 95% accuracy) for each sub-task. Error bars represent the 15th to 85th percentile range across training runs.

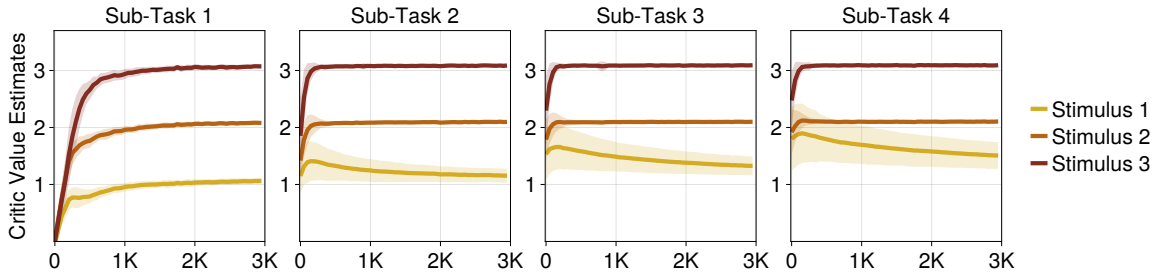

Figure 4.6: Critic network learns consistent value representations across sub-tasks. Value estimates from the M1f critic network for three stimuli (s1, s2, s3) corresponding to low, medium, and high reward magnitudes. Each panel shows a different sub-task with novel visual features. Lines show mean accuracy across training runs, with shaded bands indicating ±1 standard deviation.

Temporal Structure via the Distance Network. Complementary to value estimation, the distance network learns to estimate goal progress by predicting how many steps remain until reward delivery. Figure 4.7 shows distance estimates (represented as probability distributions over distances 0-4) as a function of task stage (columns) and action taken (rows), comparing early (Task 1, solid lines) versus late (Task 4, dashed lines) sub-tasks. The network learned the correct distance given observation and action. For instance, at Stage 0 (fixation), the correct behavior is to wait (action 3), which yields a distance estimate of 2 steps to reward. Taking incorrect actions (action 1 = left or action 2 = right) at this stage results in a distance estimate of 3, reflecting the additional step required: these actions cause the trial to loop back to Stage 0, requiring an extra action before the agent can proceed optimally through the remaining stages. Conversely, at Stage

1 (choice), the correct actions are left or right (actions 1 or 2, depending on option values), which yield distance estimates of 1. Taking action 3 (wait) at this stage prevents progression and maintains higher distance estimates, as the agent remains in Stage 1 until an appropriate choice action is taken.

More generally, distance decreases as trials progress through stages, with highest probability on distance = 2 at Stage 0 (fixation), distance = 1 at Stage 1 (choice), distance = 0 at Stage 2 (transition), and distance = 3 at Stage 3 (outcome). Distance estimates transfer rapidly from Task 1 to Task 4, with nearly identical distributions appearing by early training in the later task.

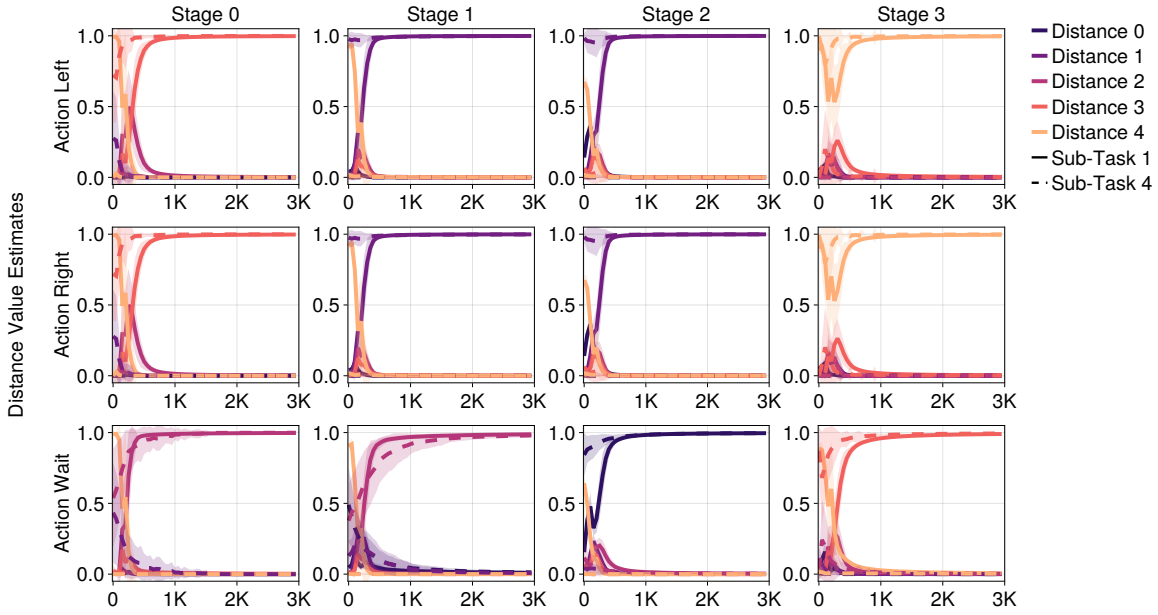

Figure 4.7: Distance network captures temporal task structure. Goal distance estimates from the M1f distance network across sub-tasks. Each panel shows distance predictions as a function of task step (x-axis) and action taken (different lines). Lines show mean accuracy across training runs, with shaded bands indicating ±1 standard deviation.

Goal Progress Computation. The distance estimates are then converted into goal progress velocity signals using the computation $v_{\text{progress}} = \min_{a'} D(o_{t-1}, a') - D(o_t, a_t)$. This velocity directly reflects whether actions advance the trial toward reward. At Stage 0, taking the correct action (wait, action 3) produces $v_{\text{progress}} = 1$ because distance decreases from 2 to 1, whereas incorrect actions (left or right, actions 1-2) produce $v_{\text{progress}} = 0$ because the loop back to Stage 0 prevents distance reduction. Conversely, at Stage 1, taking action left or right, (actions 1-2) produces $v_{\text{progress}} = 1$ as distance decreases from 1 to 0, while taking the wait action (action 3) yields $v_{\text{progress}} = 0$ as the agent remains in Stage 1 with unchanged distance.

Figure 4.8 shows the computed goal progress values across all four sub-tasks and four trial stages (columns), separately for each action (Left, Right, Wait). The cumulative path integration of these velocity signals over a trial produces systematic increases in goal progress, with correct actions leading to goal progress and incorrect actions leading to zero progress.

This goal progress signal, combined with the state value signal from the critic, defines the two-dimensional trajectory through the abstract map that the grid state follows during each trial. These maps remain invariant to changes in specific visual features, explaining why M1f can rapidly generalize to new sub-tasks: the abstract map structure persists, and only the sensory-to-value mapping (learned by the critic) and the sensory-to-distance mapping (learned by the distance network) need to be updated for new stimuli.

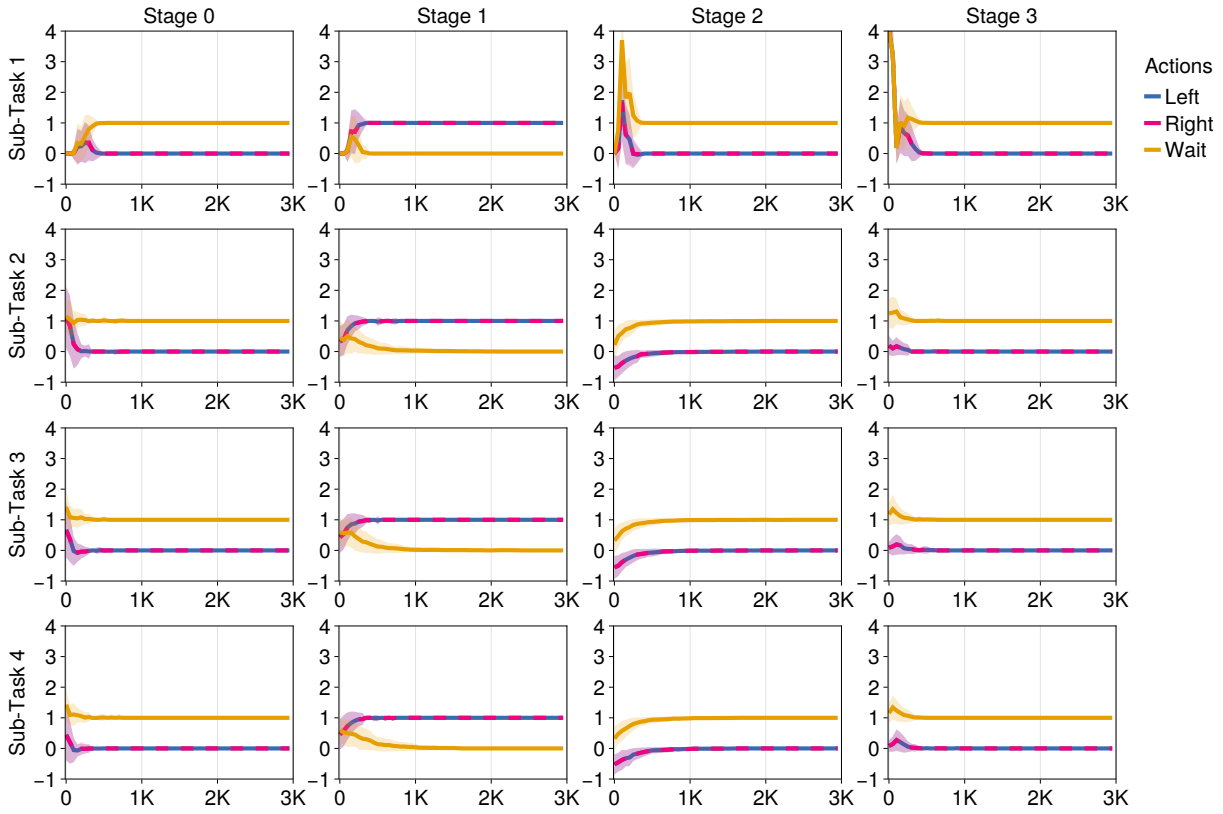

Figure 4.8: Goal progress increases systematically throughout trials. Goal progress values computed from distance network estimates in the final sub-task of M1f training. Lines show mean across 100 training runs, with shaded bands indicating ±1 standard deviation.

In summary, M1f achieved generalization through continuous feedforward updating, estimating values and distances during choice to update abstract representations in real-time. This strong performance likely arises from two architectural features: (1) the grid-to-hippocampal projection creates invariant representations that remain stable across sub-

tasks (e.g., "high reward option" produces the same hippocampal pattern regardless of visual features), enabling policy reuse, and (2) path integration through structured abstract space provides geometric organization that supports compositional generalization. Critically, however, M1f does not utilize explicit episodic memory encoding or retrieval operations, mechanisms that the VectorHaSH architecture inherently supports and that animals likely employ during learning. This raises the question of whether such explicit memory mechanisms could further enhance generalization. I next consider the M1b model, which addresses this question through backward temporal credit assignment and episodic binding.

M1b: Backward Memory Binding and Retrieval.

The M1b model extends M1f with explicit episodic memory operations. Rather than relying solely on critic estimates during choice, M1b performs credit assignment at reward delivery to identify which chosen stimulus was causally responsible for the observed outcome, and binds stimulus-reward associations in episodic memory through a second VectorHaSH system dedicated to sensory information.

At reward delivery, M1b uses a *hindsight* network to compute causal attribution weights for each observation experienced during the trial. The hindsight network learns to predict which action should have been taken given an observation and the final outcome: $P(\text{action} \mid \text{observation}, \text{outcome})$. Observations for which this hindsight prediction varies strongly with the outcome receive high causal weights, indicating they were decision-relevant states. The causally-weighted observations are used as a retrieval cue for the VectorHaSH, and the retrieved representation is associated with the observed reward magnitude through supervised training of a simple MLP (reward velocity network).

During subsequent choice phases, the reward velocity network (rather than the critic) estimates option values by predicting which of the three reward levels (low, medium, high) each viewed option corresponds to. This explicit binding and retrieval mechanism separates memory formation (which occurs at reward time when outcomes are known) from memory use (which occurs at choice time when options are evaluated), implementing a form of backward temporal credit assignment through episodic memory.

M1b Learning Performance and Internal Representations The M1b model implemented an alternative mechanism for memory-guided generalization through explicit backward credit assignment and episodic binding. Rather than relying on critic estimates during choice phases, M1b performed causal attribution at reward delivery to determine which observations were responsible for the outcome, retrieved the causally-relevant stimulus representation from a sensory VectorHaSH memory system, and bound this retrieved

representation to the observed reward magnitude.

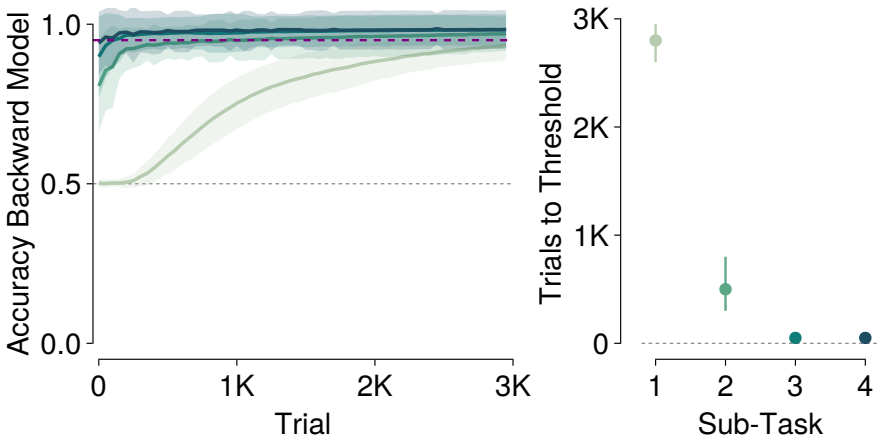

Figure 4.9: Backward binding model M1b achieves near-instantaneous transfer. Left: Learning curves showing mean accuracy across four sub-tasks (light to dark curves represent Tasks 1-4). Lines show mean accuracy across training runs, with shaded bands indicating ±1 standard deviation. Right: Median trials-to-criterion (reaching 95% accuracy) for each sub-task. Error bars represent the 15th to 85th percentile range across training runs.

Figure 4.9 shows that M1b achieved learning acceleration exceeding M1f, with progressively faster adaptation across sub-tasks. The first sub-task required approximately 2,800 trials to reach criterion, comparable to both M0 (2,700 trials) and M1f (2,825 trials). By the second sub-task, trials-to-criterion dropped to 500, identical to M1f. However, by the third and fourth sub-tasks, M1b reached criterion in just 50 trials, substantially outperforming M1f's 150-200 trials.

Causal Attribution via the Hindsight Network. The main idea behind M1b is that it links the reward value with state that leads to that reward. This mechanism poses a temporal credit assignment problem: which state should be assigned credit for the reward? To solve this, I implemented a hindsight network to compute causal attribution weights for each observation experienced during the trial. The Hindsight network learns to predict which action should have been taken at each time step, conditioned on both the observation and the final trial outcome: $P(\text{action} \mid \text{observation}, \text{outcome})$ [56, 88, 89].

Figure 4.10 shows the hindsight network's predictions across all four trial stages and three possible outcomes. At Stage 1 (the choice phase), the network's action predictions strongly depend on the outcome: when conditioned on high-reward outcomes, it predicts selecting the left or right options, with probabilities diverging based on which option yielded high reward. In contrast, at Stages 0, 2, and 3 (fixation, transition, and outcome phases), the hindsight predictions remain invariant across outcomes, correctly identifying these stages as non-causal for the reward.

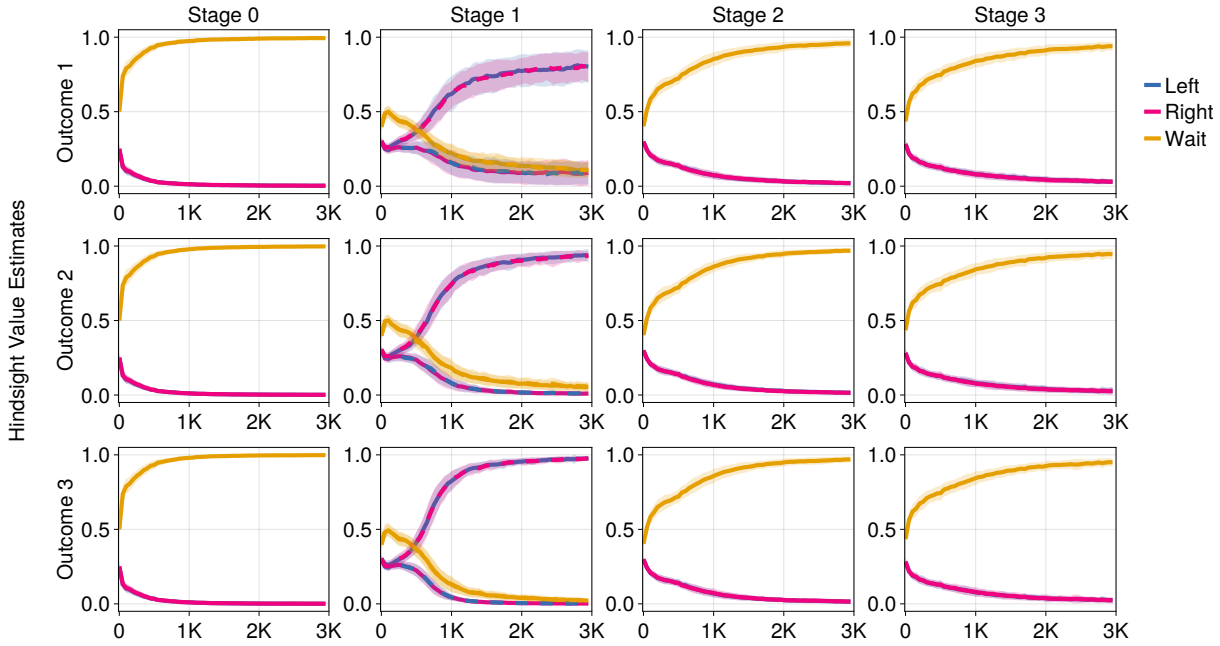

Figure 4.10: Hindsight network identifies decision-relevant observations. Hindsight value estimates showing predicted action probabilities $P(\text{action} \mid \text{observation}, \text{outcome})$ across trial stages (columns) and outcome conditions (rows). Lines show mean probability for each action (Left, Right, Wait) across training trials, with shaded bands indicating ± 1 standard deviation. Stage 1 (choice phase) shows strong outcome-dependent predictions, while other stages remain outcome-invariant. Solid lines denote conditions where the left stimulus was associated with rewards 1, 2, or 3 (first, second, and third rows, respectively), while dashed lines denote conditions where the right stimulus was associated with those same reward levels.

The degree of outcome-dependence in the hindsight network's predictions provides a quantitative measure of causal relevance. Specifically, the causal weight for each observation is computed as the variance in action probabilities across different outcome conditions, normalized by a constant (0.15):

$$w_{\text{causal}}(o_t) = \frac{1}{0.15}\mathbb{E}_a[\text{Var}_{\text{outcome}}[P(a \mid o_t, \text{outcome})]] \tag{4.3}$$

where the variance is computed across the three possible outcomes for each action, then averaged across actions. Observations with high causal weights are those for which the hindsight network's action recommendation changes substantially depending on the outcome, precisely the decision-relevant states where the action taken by the agent causes the reward.

During each trial, M1b accumulates a causally-weighted memory trace by summing ob-

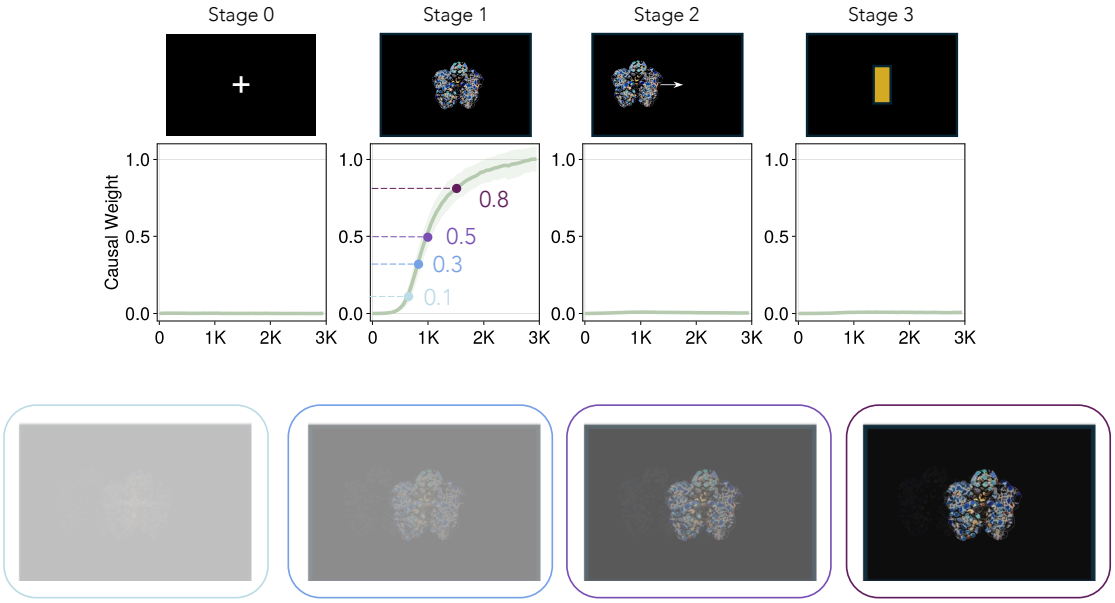

Figure 4.11: Causal weights selectively accumulate decision-relevant observations. Top: Causal attribution weights across the four trial stages over training during sub-task 1. Lines show mean across 100 training runs, with shaded bands indicating ±1 standard deviation. Bottom: Visualization of causally-weighted memory trace at different stages of learning (different colored boxes correspond to different levels of causal weights, including 0.1, 0.3, 0.5, 0.8).

servations weighted by their causal attribution:

$$s_{\text{causal}} = \sum_t o_t \cdot w_{\text{causal}}(o_t) \tag{4.4}$$

This mechanism can be thought of as a way to make the model allocate more attention to observations that are causally relevant to the reward, thereby strengthening the memory trace: the higher the causal weight, the more attention is directed to a specific state, resulting in a stronger memory trace.

Figure 4.11 shows this process across the four trial stages during the first sub-task. The top panel displays causal weights over training: Stage 1 (choice phase) rapidly develops high weights, while Stages 0, 2, and 3 maintain near-zero weights throughout training. The bottom panel visualizes the causally-weighted memory trace, showing how it becomes progressively cleaner and more similar to the choice-phase stimulus stored in memory, as the causal weights increase.

The causal weight also shows some meta-learning behavior, it converges to 1 more quickly in the later sub-tasks Figure 4.12.

At reward delivery, this causal trace serves as a retrieval cue for the VectorHaSH memory system. The VectorHaSH performs pattern completion using its error correction mech-

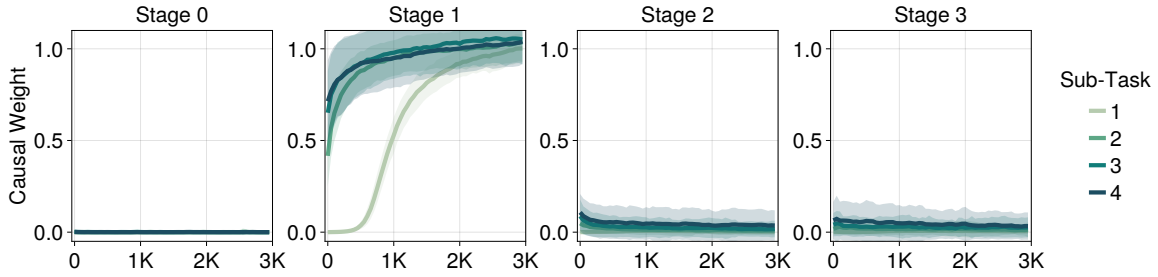

Figure 4.12: Hindsight network exhibits meta-learning across sub-tasks. Causal attribution weights for each trial stage across the four sub-tasks (light to dark curves represent Sub-Tasks 1-4). Stage 1 (choice phase) shows progressive acceleration in learning: later sub-tasks converge to high causal weights substantially faster than earlier sub-tasks, indicating that the hindsight network itself benefits from meta-learning. Stages 0, 2, and 3 maintain near-zero weights across all sub-tasks. Lines show mean across training runs, with shaded bands indicating ±1 standard deviation.

anism, retrieving the stored sensory representation most similar to the causally-weighted trace. This retrieved pattern $\hat{o}$ is then used as input to the reward velocity network, a simple feedforward MLP trained via supervised learning to predict which reward level (low, medium, or high) was associated with the retrieved stimulus. Figure 4.13 shows the reward velocity network's probability estimates across the three stimuli over training. Each stimulus rapidly develops strong, stable predictions for its associated reward value (Value 1, 2, or 3), with learning curves showing the characteristic rapid acceleration across sub-tasks.

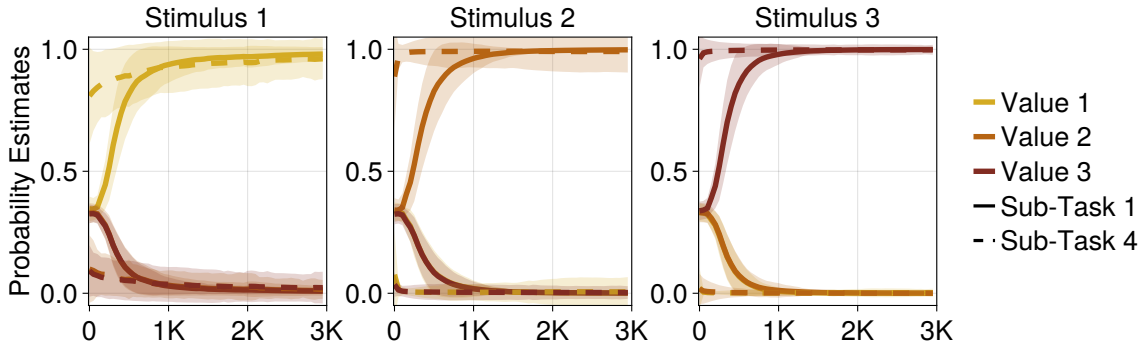

Figure 4.13: Reward velocity network learns stimulus-reward associations. Probability estimates for each of the three reward values across the three stimuli in Sub-Task 1 (solid lines) and Sub-Task 4 (dashed lines). Lines show mean across training runs, with shaded bands indicating ±1 standard deviation. Each stimulus develops selective association with its corresponding reward value.

During subsequent choice phases, when the agent sequentially gazes at each available option, the trained reward velocity network estimates the expected reward for each viewed stimulus. These value estimates replace the critic network's role in M1f, providing the

state value velocity signal for grid state updates via path integration. This architecture separates memory formation (which occurs at reward time through causal attribution and binding) from memory use (which occurs at choice time through retrieval and value estimation), implementing a form of backward temporal credit assignment that enables rapid one-shot learning of new stimulus-reward associations.

4.2.4 Summary of Model Comparisons

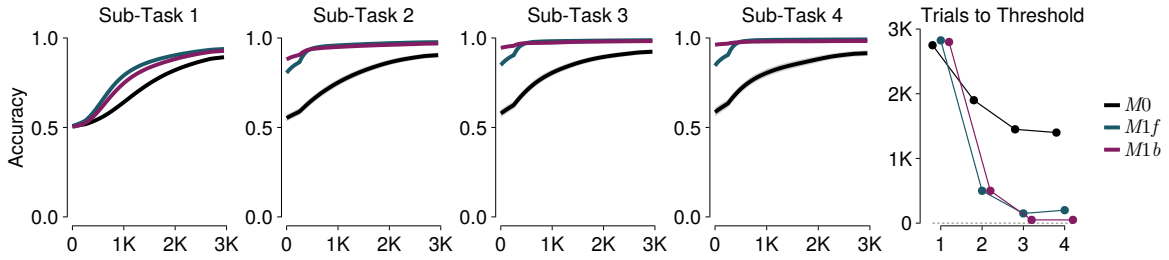

Figure 4.14: Direct comparison of meta-learning across model architectures. Left four panels: Learning curves showing accuracy over trials for Sub-Tasks 1-4. Each panel displays performance for M0 (black), M1f (teal), and M1b (magenta). Right panel: Trials-to-criterion (reaching 95% accuracy) across the four sub-tasks for all three models.

Figure 4.14 directly compares the learning trajectories of all three models across the four sequential sub-tasks. The left panels show learning curves (accuracy over trials) for each sub-task separately, while the right panel displays trials-to-criterion (95% accuracy threshold) as a function of sub-task, providing a quantitative measure of meta-learning efficiency.

4.2.5 Control Experiments

One reason why M1f and M1b outperform M0 is that they have access to abstract value and distance signals, which are unavailable to the baseline model. To test whether incorporating the VectorHaSH architecture into the baseline RNN is necessary for rapid generalization, I trained two additional control models that received these abstract signals directly as inputs (in addition to current observation, and previous action). M0d received goal distance estimates as additional input, while M0dv received both distance and critic value estimates. Figure 4.15 shows that both control models achieved only marginal improvements over M0, with M0d and M0dv learning curves remaining clustered with the baseline and far below M1f and M1b performance across all sub-tasks.

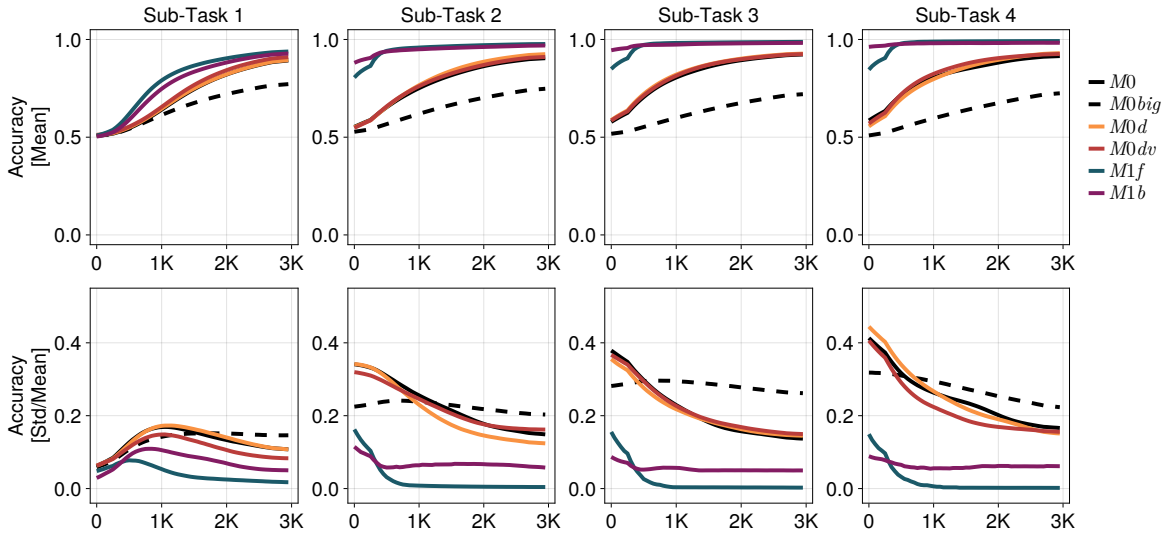

Figure 4.15: Control models with abstract inputs fail to match VectorHaSH performance. Top row: Mean accuracy across Sub-Tasks 1-4 for M0 (black), M0d with distance input (orange), M0dv with distance and value inputs (dark red), M0big with the same network size as M1b (black, dashed line), M1f (teal), and M1b (magenta). Bottom row: Coefficient of variation (standard deviation / mean) across training runs.

Another reason why M1f and M1b might outperform M0 is that they have a lot more neurons than M0. To rule out that the superior performance of M1f and M1b is due to the additional neurons, I trained an additional control model (M0big) that matched the network size of M1b (the biggest model in the study). Figure 4.15 shows that M0big performed comparably to or worse than M0, ruling out network size as an explanatory factor.

4.3 Discussion

This study investigated the computational mechanisms underlying generalization in sequential decision-making tasks. We developed three computational models, a baseline recurrent neural network (M0) and two models augmented with structured episodic memory systems (M1f and M1b), and compared their learning trajectories against animal behavior across multiple sub-tasks with novel stimuli but preserved task structure. The results demonstrate that structured episodic memory, implemented through grid-hippocampal abstract maps, can achieve the learning acceleration characteristic of animal meta-learning. Critically, these computational findings generate specific neural predictions that can be directly tested using fMRI data already collected during the animal experiments, and can be causally validated through targeted interventions using Transcranial Ultrasound Stimulation (TUS).

4.3.1 Summary of Key Findings

The baseline M0 model demonstrated modest meta-learning capabilities, reducing trials-to-criterion from approximately 2,700 trials in Sub-Task 1 to 1,400 trials by Sub-Task 4—a 2-fold improvement. This gradual acceleration reflects the ability of recurrent neural networks to extract transferable structure through implicit representations in their hidden states. However, this improvement was qualitatively different from animal behavior: M0 exhibited smooth, incremental learning across all sub-tasks, whereas animals displayed prolonged initial learning followed by sharp acceleration in later sub-tasks.

Both VectorHaSH models achieved substantially more dramatic generalization. M1f, which implemented continuous feedforward updates via critic-based value estimation and distance-based goal progress signals, reduced trials-to-criterion from 2,825 to 200 trials. M1b, which augmented this architecture with explicit backward temporal credit assignment through hindsight-based causal attribution and episodic binding, achieved even stronger performance, reaching criterion in just 50 trials by Sub-Tasks 3-4.

The critical architectural innovation enabling this rapid generalization was the separation of abstract task structure from sensory-specific mappings through grid-hippocampal abstract maps. By representing the task in a two-dimensional abstract space (goal progress $\times$ state value) and projecting grid states through fixed random transformations to create hippocampal representations, both M1 models created invariant "low reward," "medium reward," and "high reward" representations that remained stable across changes in visual features. When transitioning to new sub-tasks, these models needed only to learn which new stimuli corresponded to which pre-existing abstract categories, rather than relearning entire decision policies.

Control experiments confirmed that this architectural separation was necessary. Models that received abstract distance and value signals directly as RNN inputs (M0d, M0dv) showed only marginal improvements over the baseline, and increasing network size to match M1b (M0big) yielded comparable or worse performance than M0. These results demonstrate that providing abstract signals to an RNN is insufficient, suggesting that the structured grid-hippocampal architecture that creates invariant representations is necessary for rapid generalization.

4.3.2 Neural Predictions from Computational Models

The computational mechanisms implemented in M1f and M1b generate specific, testable predictions about neural activity patterns in medial entorhinal cortex (MEC) and hippocampus during task performance and learning.

Grid Cell Activity Patterns

Both M1 models maintain grid states that evolve via path integration along two abstract dimensions: goal progress (temporal position within trials) and state value (expected reward magnitude). This predicts that grid cells in MEC should exhibit structured firing patterns that tile the two-dimensional abstract task space, with individual grid cells showing periodic firing as the agent progresses through trials and encounters options of different values.

Critically, these grid representations should remain stable across sub-tasks despite changes in visual stimuli. If grid cells encode abstract task structure rather than sensory-specific features, then grid firing fields should align to the same abstract positions (e.g., "high reward choice phase") across all sub-tasks, even when the visual stimuli defining "high reward" change completely.

These predictions can be tested using fMRI pattern similarity analysis. Within each sub-task, we can identify voxel patterns in MEC that discriminate between different abstract task positions (e.g., fixation vs. choice vs. transition stages) and different value levels (low vs. medium vs. high reward options). If these representations encode abstract structure, the same voxel patterns should recur across sub-tasks when animals encounter analogous abstract task positions, even with novel stimuli. Representational similarity analysis (RSA) comparing pattern similarity within versus across sub-tasks can quantify the degree of invariance in MEC representations.

Hippocampal Representations

The models project grid states through fixed random transformations to create hippocampal representations. This architecture predicts that hippocampal activity patterns should exhibit stability of abstract value representations across sub-tasks.

For this prediction, hippocampal voxel patterns corresponding to "high reward option" should remain similar across sub-tasks, even as the visual features defining high reward change. Multi-voxel pattern analysis (MVPA) can decode option values from hippocampal activity patterns within each sub-task, then test whether decoders trained on one sub-task generalize to others. Cross-task generalization would support the hypothesis that hippocampus maintains abstract value representations.

Distinguishing M1f from M1b: Temporal Dynamics of Memory Operations

While both M1 models achieve rapid generalization through grid-hippocampal abstract maps, they differ fundamentally in their temporal dynamics of memory formation and use.

These differences generate distinct neural predictions that can discriminate between feedforward (M1f) and backward binding (M1b) mechanisms.

Choice-Phase Activity. During the choice phase, both M1f and M1b predict value-guided decisions based on hippocampal representations. M1f estimates values via the critic network, while M1b retrieves previously bound stimulus-reward associations via the reward velocity network. However, both mechanisms ultimately result in hippocampal activity patterns that correlate with option values and predict subsequent choices. Thus, distinguishing these mechanisms based on choice-phase hippocampal activity is challenging.

Outcome-Phase Activity. The models make opposite predictions about hippocampal activity at reward delivery. M1f performs no special operations at outcome; the critic simply receives a TD error to update value estimates. M1b, in contrast, performs causal attribution via the hindsight network and uses the accumulated causally-weighted trace (maintained in prefrontal working memory) to retrieve stimulus representations from hippocampal sensory VectorHaSH, then binds these retrieved patterns to reward outcomes. This predicts strong, sustained hippocampal activity at outcome in M1b (reflecting retrieval and binding operations) but minimal outcome-related activity in M1f (beyond standard reward processing).

Prefrontal-Hippocampal-Entorhinal Interactions. The models also predict distinct patterns of functional connectivity across brain regions. M1f implements feedforward information flow: MEC grid states drive hippocampal activity during choice. M1b implements a more complex pattern involving prefrontal cortex (PFC): during the trial, PFC accumulates causally-weighted observations through recurrent dynamics (trace = trace + observation $\times$ causal weight); at outcome, this accumulated PFC trace drives hippocampal retrieval (PFC$\rightarrow$hippocampus), and the retrieved hippocampal pattern then drives MEC sensory VectorHaSH binding (hippocampus$\rightarrow$MEC).

These predictions can be tested using time-resolved functional connectivity analysis. During the outcome phase, M1f predicts weak or absent connectivity between these regions, whereas M1b predicts a sequential pattern: PFC$\rightarrow$hippocampus (retrieval cue) followed by hippocampus$\rightarrow$MEC (episodic binding). Granger causality analysis or dynamic causal modeling applied to fMRI data can test these directional connectivity predictions across task phases, with particular attention to PFC involvement during outcome processing in M1b.

4.3.3 Causal Validation Through Transcranial Ultrasound Stimulation

While the fMRI predictions outlined above can provide correlational evidence for grid-hippocampal involvement in rapid generalization, they cannot establish causality. The animals in this study provide a unique opportunity for causal intervention through Transcranial Ultrasound Stimulation (TUS), a minimally invasive technique that can modulate neural activity in deep brain structures with millimeter-scale precision. Unlike pharmacological or lesion approaches, TUS allows for reversible and targeted region-specific interventions, making it ideal for testing the causal necessity of specific brain regions.

TUS effects persist for several hours following stimulation, precluding within-trial manipulations of specific task phases (e.g., choice versus outcome). Instead, TUS experiments must operate at the session or multi-session timescale, comparing performance across behavioral sessions with and without MEC or hippocampal disruption, or disrupting these regions during specific learning phases (e.g., early versus late sub-tasks).

Targeting Medial Entorhinal Cortex

Both M1 models critically depend on MEC grid representations to create abstract maps. Disrupting MEC activity should therefore impair rapid generalization by preventing the formation or maintenance of invariant abstract representations. However, the specific predictions differ based on mechanism.

Applying TUS to MEC throughout task should selectively impair performance of M1f and M1b while leaving M0 intact, since only the VectorHaSH models depend on grid-based representations. This would manifest as increased trials-to-criterion within each sub-task, reflecting impaired use of abstract value representations to guide choices.

Critically, the impairment should be specific to generalization rather than basic task performance. Once animals learn stimulus-reward associations within a sub-task (even if this takes longer under MEC disruption), they should still be able to perform the task accurately, since the RNN policy can represent stimulus-action mappings without requiring abstract representations. This dissociation—impaired learning speed but preserved asymptotic performance—would provide strong evidence that MEC contributes specifically to the rapid acquisition of new associations through abstract representations rather than to basic stimulus-response learning or motor execution.

Targeting Hippocampus

The hippocampus receives grid state projections and provides abstract value representations to the policy network. Session-level hippocampal disruption should impair both M1

models, but the pattern and magnitude of impairment may provide clues about underlying mechanisms.

Applying TUS to hippocampus during behavioral sessions should impair value-guided decision-making in both models. The key prediction is that hippocampal disruption should produce qualitatively similar impairments to MEC disruption if these regions operate as a tightly coupled system implementing grid-hippocampal cognitive maps.

Targeting Prefrontal Cortex to Distinguish M1f from M1b

The two VectorHaSH models differ critically in their reliance on causal attribution. M1b uses a hindsight network to compute causal weights that selectively bind decision-relevant observations to outcomes. M1f, in contrast, performs no causal attribution, instead it learns the values of observations during choice via continuous critic-based updates, without evaluating which observations were causally responsible for outcomes. This architectural difference creates a clear dissociation: disrupting causal attribution should impair M1b but leave M1f intact, providing a direct test of which mechanism animals employ.

fMRI data could identify regions computing causal weights by searching for areas where representations of observations vary depending on trial outcomes. Based on M1b's architecture and prior work on causal inference, mPFC and dACC are strong candidates. Disrupting these regions would test whether animals rely on M1b-like backward binding or M1f-like feedforward updating.

If animals use M1b-like mechanisms, PFC disruption should produce intermediate performance by impairing selective memory binding while preserving abstract structure. Without accurate causal weights, all observations would receive similar weights during binding, producing a noisy retrieval cue. This predicts degraded but not abolished learning and increased susceptibility to interference when similar features appear across trial phases. Conversely, if animals use M1f-like mechanisms, PFC disruption targeting causal attribution should have minimal effect on learning, since M1f does not depend on distinguishing causally-relevant from irrelevant observations. Comparing animal performance under PFC disruption to predictions from both models would reveal which computational strategy the brain implements.

4.3.4 Broader Implications for Episodic Memory and Rapid Learning

The finding that structured episodic memory enables rapid generalization has implications beyond this specific task. The VectorHaSH architecture implements a general principle: abstract structured representations can serve as scaffolds for organizing sensory experiences, enabling rapid learning by reducing the problem of associating new stimuli with re-

wards to the simpler problem of categorizing stimuli into pre-existing abstract categories.

This principle may extend to other domains requiring rapid learning from limited data. In social learning, for instance, abstract representations of social hierarchies or reputations could enable rapid learning of appropriate behaviors toward novel individuals.

The distinction between M1f (feedforward) and M1b (backward binding) mechanisms is particularly relevant for understanding the diverse memory phenomena observed in hippocampal-dependent learning. M1f's continuous value estimation resembles online learning mechanisms proposed for spatial navigation, where grid-hippocampal representations are continuously updated during exploration. M1b's backward credit assignment resembles episodic memory consolidation, where experiences are "replayed" and bound to their outcomes during rest or sleep.

These mechanisms are not mutually exclusive. Animals may employ both feedforward estimation during choice and backward binding during outcome, combining their complementary strengths: feedforward mechanisms provide immediate value estimates to guide decisions, while backward mechanisms enable one-shot learning by directly associating outcomes with their causes. The superior performance of M1b over M1f suggests that backward binding may be particularly important for rapid generalization, but hybrid models combining both mechanisms could provide even better fits to animal behavior.

4.3.5 Prefrontal Cortex: Velocity Signals and Working Memory

While the discussion above has focused on MEC and hippocampus as implementing grid-hippocampal cognitive maps, it is important to recognize that these regions likely receive velocity signals from other brain areas. Both M1 models require two critical velocity signals: goal progress (temporal position within trials) and state value (expected reward magnitude). Additionally, M1b requires a working memory mechanism to accumulate causally-weighted observations across the trial. The computational models implement these via learned auxiliary networks (distance network, critic network, reward velocity network, and hindsight network), but the neural substrates computing these signals and maintaining working memory traces in biological brains remain to be determined.

We propose that prefrontal cortex regions are strong candidates for both velocity computation and working memory maintenance. Specifically, medial prefrontal cortex (mPFC) and dorsal anterior cingulate cortex (dACC) have been repeatedly implicated in monitoring task progress, maintaining task structure representations, and signaling transitions between task phases. These regions could compute goal progress velocity by tracking the animal's position within the trial sequence and signaling forward movement through stages. Conversely, ventromedial prefrontal cortex (vmPFC) and orbitofrontal cortex (OFC) are well-established value-processing regions that represent expected rewards and encode out-

come predictions. These regions could compute state value velocity by estimating the reward magnitudes associated with current options.

Critically, in M1b, PFC also maintains the causally-weighted trace through recurrent dynamics. The recurrent computation (memory_trace = memory_trace + observation × causal_weight) is naturally implemented by persistent neural activity in PFC working memory circuits. This trace accumulates throughout the trial and, at outcome, serves as a retrieval cue for hippocampal sensory VectorHaSH. This dual role of PFC (computing velocity signals and maintaining working memory traces) aligns with extensive evidence that PFC supports both value-based decision-making and working memory maintenance through persistent activity.

This proposed division of labor suggests that the EC-hippocampal system receives abstract velocity signals from PFC and uses them to update grid states via path integration, rather than computing these velocities itself. The VectorHaSH models would then be better conceptualized as implementing the path integration and memory storage functions, with velocity computation outsourced to prefrontal circuits. This perspective has several implications for interpreting the TUS experiments proposed above. Disrupting MEC or hippocampus should impair the use of velocity signals to update abstract representations and guide decisions, but should leave the computation of velocity signals intact. If velocity computation depends on PFC, then animals under MEC/hippocampal disruption might still show normal value learning curves (as measured by choice probabilities toward high-reward options) despite impaired meta-learning, because they can still estimate values but cannot organize them into abstract structured representations.

I can test this hypothesis by examining PFC activity during task performance and testing whether PFC disruption produces qualitatively different impairments than MEC/hippocampal disruption. If PFC disruption impairs value estimation or progress monitoring (producing random choices or perseveration) while MEC/hippocampal disruption impairs generalization (producing slow learning across sub-tasks), this would support a division of labor between velocity computation (PFC) and abstract representation (MEC/hippocampus).

4.3.6 Limitations and Future Directions

A few limitations of this study suggest directions for future research. First, the models were trained on a simplified four-stage task that omitted the motion discrimination component performed by animals. While this simplification allowed us to focus on mechanisms of value-based generalization, it limits direct comparison to animal behavior.

Second, we compared only a small number of architectural variants. Other implementations of structured memory, such as successor representations, factored world models, or

transformer-based architectures with explicit memory modules, might achieve comparable or superior performance. Systematic comparison across a broader range of architectures could identify the minimal sufficient components for rapid generalization and reveal which aspects of the VectorHaSH architecture are essential versus incidental.

4.3.7 Conclusion

This study demonstrates that structured episodic memory, implemented through grid-hippocampal cognitive maps, is necessary to achieve the rapid generalization characteristic of animal meta-learning. By separating abstract task structure from sensory-specific mappings, VectorHaSH models enable dramatic acceleration in learning across sequential sub-tasks with novel stimuli but preserved structure. Control experiments confirm that this architectural separation is essential: providing abstract signals directly to an RNN is insufficient, and increasing network size does not compensate for the lack of structured memory.

The computational mechanisms implemented in these models generate specific, testable predictions about neural activity patterns in medial entorhinal cortex and hippocampus that can be validated using fMRI data collected during the animal experiments. Furthermore, the distinction between feedforward (M1f) and backward binding (M1b) mechanisms generates distinct neural and behavioral predictions that can be causally tested through temporally-precise TUS interventions. These experiments provide a path from computational modeling to neural validation, bridging the gap between abstract algorithms and biological implementation.

More broadly, this work illustrates how structured representations in episodic memory can scaffold rapid learning by transforming the problem of acquiring new associations into the simpler problem of categorization. This principle may extend beyond value-based decision-making to other domains requiring rapid learning from limited data, suggesting a general computational role for grid-hippocampal systems in enabling flexible, data-efficient cognition.

4.4 Methods

4.4.1 Experimental Design

Two male rhesus macaques (referred to as animal A and B, 4-7 years old, 11-13 kg) performed a two-stage decision-making task across six sequential sub-tasks. Each sub-task shared an identical structure but used different visual stimuli, requiring the animals to relearn stimulus-reward associations while the abstract task structure remained constant.

Task Structure

Each trial consisted of up to five stages:

Stage 0: Inter-trial interval (ITI). A uniform gray screen was displayed for 2-5 seconds (animal A) or 4-6 seconds (animal B), sampled from a uniform distribution. During this period, the animal was required to keep hands away from the response sensors.

Stage 1: First-stage choice. Two visual stimuli appeared on the left and right sides of the screen. After a minimum delay of 0.5 seconds, the animal could respond by touching infrared sensors aligned with the stimuli. The three possible stimuli (presented two at a time) were associated with different reward magnitudes: low (0 drops), medium (3 drops for animal A, 1 drop for animal B), or high (9 drops for animal A, 4 drops for animal B). These associations were arbitrary and fixed within each sub-task but changed across sub-tasks. Visual stimuli were defined by color (RGB values in stimulus space) and motion direction for random dot kinematograms (RDK), with color and angle parameters rotated across sub-tasks to ensure novel perceptual features.

Stage 2: Transition display. The chosen stimulus was displayed alone for 3-5 seconds (uniform distribution), providing feedback about the animal's choice. During this period, response sensors were inactive.

Stage 3: Second-stage motion discrimination. A random dot kinematogram (RDK) was presented centrally. After 0.5 seconds, sensors were activated and the animal indicated the perceived direction of motion (left or right) using the same infrared sensors. The RDK consisted of 100 dots moving at 2 pixels per frame within a 300-pixel circular aperture with a coherence of 100%, a dot size of 10 pixels. Motion direction left or right with a probability of 50%.

Stage 4: Outcome delivery. Following an action-outcome delay of 0.5 seconds, a visual cue indicated the trial outcome. If both the first-stage choice and second-stage motion discrimination were correct, juice reward was delivered immediately through a spout (0.5 ml drops, 40 ms delivery time). The outcome display lasted for the duration of reward delivery. Incorrect responses on either stage resulted in no reward delivery. An ITI then began for the next trial.

Adaptive Task Progression

Task difficulty was held constant within each sub-task, but progression across sub-tasks followed an adaptive schedule based on learning criterion. Performance was assessed using a sliding window of 300 trials. When accuracy within this window exceeded 75% correct for the first-stage performance and the animal performed at least 450 trials, the task automatically progressed to the next sub-task. This criterion ensured a minimum of 450 trials per animal and sub-task and that the animal had learned the stimulus-reward associations before being presented with new stimuli in the subsequent sub-task.

Each new sub-task maintained the same structure and reward magnitudes but introduced three new visual stimuli. The animal was required to relearn which stimulus corresponded to low, medium, or high reward, while the rest of the task remained the same.

Stimulus Presentation and Response Recording

Stimuli were displayed on a screen at 30 cm viewing distance using the python library psychopy with a black background to maintain constant luminosity throughout the task. Visual stimuli were presented at 60 Hz refresh rate. Animals sat in a sphinx position in an MRI-safe chair (Rogue Research) with the head fixed using an MRI-compatible cranial implant. Responses were recorded using custom-built infrared sensors positioned to align with the left and right stimulus locations.

Juice reward was 25% blackcurrant Ribena diluted in 75% water.

Experimental Sessions

Both animals completed six sub-tasks following initial behavioral training. Sessions continued until the animal stopped working, until 80-120 trials were completed, or after 90 minutes. Session duration varied depending on the animal's motivation and performance, with typical sessions lasting 30-45 minutes.

4.4.2 Behavioral Training

Behavioral Shaping

Both animals had extensive prior experience with a different two-stage decision-making task described in Miyamoto and colleagues (2021) [92]. In that prior work, animals learned to perform multi-step decisions where an initial choice determined a subsequent state transition, followed by a second choice that determined reward outcome. Animals were not proficient at learning arbitrary stimulus-reward associations but they were familiar with

waiting during inter-trial intervals, and performing accurate motion discrimination judgments.

During training and experimental sessions, we implemented strategies to prevent the development of side choice biases. Animal A received trials with random left-right assignment of stimulus values, with all three reward magnitudes (low, medium, high) presented together in various two-option combinations. Animal B received a more adaptive procedure designed to actively discourage side preferences. For Animal B, we tracked choice history over a sliding window of the most recent 10 trials. After the initial 10 trials, we calculated the proportion of left versus right choices. When a spatial bias emerged, the low-value stimulus was preferentially placed on the less-chosen side. For example, if the animal selected the left option in 7 of the last 10 trials, the probability of placing the low-value option on the left side increased proportionally (0.7 in this example).

The familiarization phase lasted approximately 5-10 sessions until both animals demonstrated stable performance with the initial stimulus set. Following this brief training period, animals began the six experimental sub-tasks.

Performance Monitoring

During both training and experimental sessions, performance was monitored in real time. Accuracy was computed separately for first-stage choices and second-stage motion discrimination. For first-stage choices, accuracy reflected whether the animal selected the higher-value option (in trials presenting two different values). For second-stage discrimination, accuracy reflected correct motion direction judgments.

Performance metrics were saved for each trial, including choice (left/right), reaction time, trial type, condition (stimulus values presented), correctness for each stage, and temporal information for each trial phase.

Animals lived on a 12-hour light/dark cycle, were fed once per day after testing, and had ad libitum water access for an average of 15 hours per day (minimum 3 hours). All procedures were approved by the institutional animal care and use committee and followed guidelines for the ethical treatment of non-human primates.

4.4.3 Computational Modeling

We compared four baseline model (M0) variants with two models that used episodic memory (M1).

Baseline Model Architectures (M0s)

We trained four baseline model variants. All models were implemented as actor-critic architectures using recurrent neural networks with streaming reinforcement learning [36].

M0: The M0 model consisted of a recurrent neural network that received only current sensory observations and previous action as input. Visual stimuli were encoded as 64-dimensional token vectors through a learned projection layer. At each time step, the current observation token and previous action token (also 64-dimensional) were concatenated and fed to a vanilla RNN with 256 hidden units. The RNN output was passed through a linear layer to produce action logits over three possible actions (choose left option, choose right option, or maintain fixation).

The policy network parameters included the observation encoder (64×64), action encoder (3×64), RNN weights (input: 128×256, recurrent: 256×256), and output layer (256×3), totaling approximately 103,360 trainable parameters.

M0d: RNN with Goal Distance. The M0d model augmented the baseline architecture with an auxiliary network that estimated goal distance, namely the number of steps remaining until reward delivery. The distance network learned to predict $D(o_t, a_t)$, defined recursively as:

$$D(o_t, a_t) = \min_{a'} D(o_{t+1}, a') + 1 \tag{4.5}$$

with the base case:

$$D(o_t, a_t) = 0 \quad \text{when observation and action at time } t \text{ leads to reward delivery} \tag{4.6}$$

This recursive formulation captures the intuition that the distance from the current state-action pair (o_t, a_t) is one more than the minimum distance achievable from any action in the next state. The base case anchors the recursion at trial completion: when an action leads to reward delivery, the distance from the resulting observation to the goal is zero by definition.

The distance network consisted of a two-layer feedforward network (128→64→5) that took the concatenated observation and action tokens as input and produced a probability distribution over five discrete distance values $\{0, 1, 2, 3, 4\}$.

During training, supervised targets were computed by unrolling the recursive definition backwards from reward delivery. The target distance at time t was computed as:

$$D_{\text{target}}(o_t, a_t) = \begin{cases} 0 & \text{if reward delivered at } t \\ \min_{a'} D(o_{t+1}, a') + 1 & \text{otherwise} \end{cases} \tag{4.7}$$

The distance network was trained by minimizing cross-entropy loss between its predicted distribution and a one-hot encoding of the target distance. The distance estimate was then encoded as a one-hot vector and concatenated with the observation and action tokens before being fed to the RNN policy. This provided the policy with an explicit representation of temporal structure within each trial, enabling it to condition behavior on the estimated progress toward the goal.

Each model with the distance network (M0d, M0dv, M1f, M1b) employed experience replay specifically for training the distance network. Every 50 trials, the distance network received an additional 25 gradient updates using state-action-distance triplets from an imagined experience using action probabilities inversely proportional to the ones under the current policy. This allowed the distance network to learn using counterfactual reasoning. The policy network continued to use streaming updates without replay. This design allowed us to improve distance estimation accuracy.

M0dv: RNN with Goal Distance and State Value. The M0dv model extended M0d by providing as an additional input signal the estimated state value from the critic network. At each step, the critic's value estimate for the current observation was concatenated with the distance signal and provided to the policy RNN. This gave the policy direct access to both temporal (distance to goal) and reward-related (expected value) information. The policy network input dimensionality increased accordingly to accommodate the additional scalar value input (total auxiliary input dimension: 6, comprising 5 for one-hot distance and 1 for value).

VectorHaSH Memory Models (M1)

Both M1 models augmented the RNN policy with a Vector Hippocampal Scaffolded Heteroassociative Memory (VectorHaSH) system [20]. VectorHaSH provides structured episodic memory through grid cell scaffolds that create exponentially many stable attractor states for binding and retrieving sensory information. The key architectural component is a grid state that evolves via path integration using learned velocity signals, which is then projected to a hippocampal representation that serves as input to the policy RNN.

Grid State Path Integration. Both M1 models maintained a grid state $\mathbf{g}_t$ that was updated via path integration. The grid state represented the agent's position in an abstract two-dimensional space. At each time step, the grid state was updated according to:

$$\mathbf{g}_{t+1} = \text{PathIntegrate}(\mathbf{g}_t, \mathbf{v}_t) \tag{4.8}$$

where $\mathbf{v}_t = [v_{\text{progress}}, v_{\text{value}}]$ is a two-dimensional velocity vector that must be estimated from observations and actions. The two velocity dimensions represent abstract task variables: goal progress (how many steps toward reward delivery) and state value (expected reward magnitude). The path integration operation updated the grid state by shifting it along the hexagonal grid manifold according to the velocity signal, preserving the periodic structure of grid cell firing patterns.

The grid state was represented using three grid modules with periods $\lambda \in \{7, 8, 11\}$, producing a total of $7^2 + 8^2 + 11^2 = 234$ grid cells.

The grid state was projected to a hippocampal representation $\mathbf{h}_t$ through a fixed linear transformation:

$$\mathbf{h}_t = \text{ReLU}(\mathbf{W}_{hg}\mathbf{g}_t) \tag{4.9}$$

where $\mathbf{W}_{hg} \in \mathbb{R}^{256\times234}$ was a random sparse projection matrix (10% density) initialized at the beginning of training and held fixed thereafter. This projection created high-dimensional hippocampal representations (256 dimensions) from the structured grid state, with the ReLU nonlinearity ensuring biological plausibility through non-negative firing rates.

M1f: Feedforward VectorHaSH. The M1f model implemented feedforward state updates where grid and hippocampal states were continuously updated based on estimated velocity signals, without explicit memory binding or retrieval operations.

During the choice phase (stage 1), the model simulated sequential "gazes" at each available option. For each option viewed:

1. The observation token $\mathbf{o}_i$ corresponding to option i was passed to the critic network to obtain a value estimate: $v_i = V(\mathbf{o}_i)$
2. This value estimate was discretized to the nearest integer: $\hat{v}_i = \text{round}(v_i) \in \{1, 2, 3\}$
3. The grid state was updated using velocity $\mathbf{v} = [0, \hat{v}_i]$, where goal progress velocity was zero (no advancement toward reward during option viewing) and value velocity reflected the estimated value
4. The updated grid state was projected to hippocampal representation: $\mathbf{h}_i = \text{ReLU}(\mathbf{W}_{hg}\mathbf{g}_i)$

The policy RNN then received as input the concatenated observation tokens and hippocampal representations from both options: $[\mathbf{o}_1, \mathbf{o}_2, \mathbf{h}_1, \mathbf{h}_2]$, along with the previous action token. This provided the policy with abstract value information (through $\mathbf{h}_1, \mathbf{h}_2$) grounded in the specific sensory features (through $\mathbf{o}_1, \mathbf{o}_2$).

Between stages, the goal progress velocity was estimated by the distance network. The distance network computed how much closer the agent was to reward delivery based on the transition from observation $\mathbf{o}_{t-1}$ to $\mathbf{o}_t$ after taking action $\mathbf{a}_t$:

$$v_{\text{progress}} = D(\mathbf{o}_{t-1}, \mathbf{a}) - D(\mathbf{o}_t, \mathbf{a}_t) \tag{4.10}$$

where $D(\cdot, \cdot)$ is the distance network's prediction of steps remaining to reward.

The M1f architecture contained the same core policy parameters as M0 models (approximately 135,000), plus the fixed grid-to-hippocampus projection weights. No additional trainable parameters were introduced beyond those in the baseline models.

M1b: Backward Memory Binding. The M1b model augmented M1f with explicit episodic memory binding and retrieval through a second VectorHaSH system dedicated to sensory information.

M1b maintained two parallel grid-hippocampal systems:

1. **Structure VectorHaSH**: Identical to M1f, tracking position × reward in a 2D grid space
2. **Sensory VectorHaSH**: Enabling content-addressable memory

The sensory VectorHaSH grid was updated according to task-specific velocity mappings:

$$\mathbf{v}_{\text{sensory}} = [v_x, v_y] \tag{4.11}$$

where v_x and v_y were arbitrary velocity signals constrained to lead to grid-states with a unique correspondence of sensory input. At each time step, the current observation $\mathbf{o}_t$ was bound to the current sensory grid state using an incremental pseudoinverse learning algorithm [118]. This online learning method, based on Greville's algorithm, preserves previous sensory-hippocampal associations while incrementally learning new ones. The bidirectional weight matrices $\mathbf{W}_{\text{sensory}\to\text{hippo}}$ and $\mathbf{W}_{\text{hippo}\to\text{sensory}}$ were updated according to:

$$\mathbf{b}_k = \frac{\boldsymbol{\theta}\mathbf{a}_k}{1 + \mathbf{a}_k^\top \boldsymbol{\theta}\mathbf{a}_k} \tag{4.12}$$

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \boldsymbol{\theta}\mathbf{a}_k\mathbf{b}_k^\top \tag{4.13}$$

$$\mathbf{W} \leftarrow \mathbf{W} + (\mathbf{y}_k - \mathbf{W}\mathbf{a}_k)\mathbf{b}_k^\top \tag{4.14}$$

where $\mathbf{a}_k$ is the hidden layer activation, $\mathbf{y}_k$ is the target output (either sensory or hippocampal representation), $\boldsymbol{\theta}$ is an auxiliary matrix that accumulates the pseudoinverse,

and $\mathbf{b}_k$ is the update direction. This algorithm converges to the Moore-Penrose pseudoinverse solution in a single forward pass per observation, enabling efficient online heteroassociative memory formation without catastrophic interference.

M1b introduced three additional auxiliary networks:

Reward Velocity Network. A two-layer feedforward network (64→128→3) that predicted which option (1, 2, or 3) an observation belonged to. During the choice phase, the reward velocity network replaced the critic for value estimation:

$$\mathbf{p}_i = \text{softmax}(\text{RewardVelocity}(\mathbf{o}_i)) \tag{4.15}$$

Hindsight Network. A two-layer feedforward network (64+3→128→3) that predicted which action should have been taken given an observation and final outcome:

$$P(\text{action} \mid \text{observation}, \text{outcome}) = \text{softmax}(\text{Hindsight}(\mathbf{o}_t, \mathbf{a}_{\text{outcome}})) \tag{4.16}$$

The causal weight for an observation was computed as the variance of the hindsight distribution across different outcomes:

$$w_{\text{causal}}(\mathbf{o}_t) = \text{Var}_{\text{outcomes}}[P(\text{action} \mid \mathbf{o}_t, \text{outcome})] \tag{4.17}$$

High variance indicated that actions at this state causally influenced the final outcome, while low variance indicated the state was not decision-relevant.

Credit Assignment via Memory Retrieval. At reward delivery, M1b performed credit assignment:

1. Accumulated causally-weighted observations throughout the trial: $\mathbf{o}_{\text{causal}} = \sum_t w_{\text{causal}}(\mathbf{o}_t) \cdot \mathbf{o}_t$
2. Used this weighted sum as a retrieval cue for the sensory VectorHaSH. The recall operation proceeds through five sequential transformations:

$$\hat{\mathbf{h}}_1 = \mathbf{W}_{hs}\mathbf{o}_{\text{causal}} \quad \text{(project sensory to hippocampus)} \tag{4.18}$$

$$\hat{\mathbf{g}}_{\text{noisy}} = \text{ReLU}(\mathbf{W}_{\text{gh}}\hat{\mathbf{h}}_1) \quad \text{(project hippocampus to grid)} \tag{4.19}$$

$$\hat{\mathbf{g}} = \text{CAN}(\hat{\mathbf{g}}_{\text{noisy}}) \quad \text{(error correction via continuous attractor)} \tag{4.20}$$

$$\hat{\mathbf{h}}_2 = \text{ReLU}(\mathbf{W}_{\text{hg}}\hat{\mathbf{g}}) \quad \text{(project grid to hippocampus)} \tag{4.21}$$

$$\hat{\mathbf{o}} = \mathbf{W}_{sh}\hat{\mathbf{h}}_2 \quad \text{(project hippocampus to sensory)} \tag{4.22}$$

where $\mathbf{W}_{hs}$ and $\mathbf{W}_{sh}$ are the bidirectional sensory-hippocampal projection weights learned via incremental pseudoinverse, $\mathbf{W}_{\text{gh}}$ and $\mathbf{W}_{\text{hg}}$ are the fixed random scaffold projections between grid and hippocampus, and CAN() denotes continuous attractor

network dynamics [15] that snap the noisy grid state to the nearest stored attractor state in the grid book, performing pattern completion.

3. Trained the reward velocity network using supervised learning: RewardVelocity($\hat{\mathbf{o}}$) $\rightarrow$ $\mathbf{a}_{\text{outcome}}$

This mechanism created an explicit binding between retrieved option representations and their associated rewards, enabling rapid value estimation during subsequent choices without requiring the critic network.

Critic Network

All models included a separate critic network for computing temporal-difference errors. The critic learned to approximate the value function $V(o_t)$, which represents the expected discounted return from observation o_t under the current policy:

$$V(o_t) = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k r_{t+k} \mid o_t \right] \tag{4.23}$$

where $\gamma \in [0, 1]$ is the discount factor and r_t is the reward at time t.

The value function satisfies the Bellman equation, which expresses the recursive relationship between the value of the current state and the value of successor states:

$$V(o_t) = \mathbb{E}_\pi[r_t + \gamma V(o_{t+1})] \tag{4.24}$$

This recursive structure implies that the value of the current observation equals the immediate reward plus the discounted value of the next observation. The critic network was trained to satisfy this consistency constraint using temporal-difference (TD) learning.

The critic was a one-layer feedforward network (64$\rightarrow$64$\rightarrow$1) that took observation tokens as input and produced scalar value estimates. At each time step, the network computed its current value estimate $V(o_t)$ and compared it to the bootstrapped target $r_t + \gamma V(o_{t+1})$. The temporal-difference error quantified the discrepancy:

$$\delta_t = r_t + \gamma V(o_{t+1}) - V(o_t) \tag{4.25}$$

with the terminal condition $V(o_{t+1}) = 0$ when the trial ended at time $t + 1$.

This TD error served two functions. First, it provided the learning signal for updating the critic itself: the critic's loss was the TD error, and gradients were computed with respect to $V(o_t)$ to minimize this discrepancy. Second, the TD error served as the learning signal for the policy network, indicating whether the selected action led to better-than-expected ($\delta_t > 0$) or worse-than-expected ($\delta_t < 0$) outcomes.

The critic was trained using the same streaming ObGD optimizer as the policy, with separate eligibility traces that accumulated gradients over time. At each step, the critic received a gradient update proportional to δ_t times its eligibility trace, allowing it to learn from multi-step consequences of actions through eligibility trace mechanisms. Like the policy, the critic's eligibility traces were reset to zero at the end of each trial.

Training Procedure

Models were trained using streaming deep reinforcement learning, where network weights were updated after every environmental interaction without storing past experiences in a replay buffer. This approach mirrors biological learning, where animals update the strengths of their synaptic connections following each interaction with the environment.

The training protocol proceeded as follows. For each sub-task, models were trained for 3,000 trials. At trial onset, the environment initialized with a fixation cross displayed centrally. The model sequentially fixated on the two available options according to its gaze pattern, encoding each as a 64-dimensional observation token. After viewing both options, the model selected an action using categorical sampling from its policy distribution. Following the choice, the environment provided the outcome (reward or no reward based on the chosen option's value and motion discrimination accuracy), and the model received the resulting TD error signal.

Weight updates were computed using the ObGD (Overshooting-bounded Gradient Descent) optimizer [36], which prevents the instabilities that typically arise in streaming deep RL. ObGD dynamically adjusts the effective learning rate based on the magnitude of eligibility traces and TD errors, preventing destructive weight updates while maintaining the benefits of immediate learning. The optimizer maintains eligibility traces z that accumulate gradients over time according to $z_t = \gamma\lambda z_{t-1} + \nabla_\theta \log \pi(a_t|s_t)$, where γ is the discount factor and λ is the trace decay parameter.

The policy loss combined the standard actor-critic objective with an entropy regularization term:

$$L_\pi = -\log \pi(a_t|s_t)\delta_t + \tau \cdot \text{sign}(\delta_t) \cdot H[\pi(\cdot|s_t)] \tag{4.26}$$

where $H[\pi]$ is the policy entropy and τ is the entropy regularization coefficient. The entropy term encouraged exploration by penalizing overly deterministic policies, with the penalty sign matched to the TD error to provide a consistent learning signal.

Eligibility traces were reset to zero at the end of each trial, ensuring that credit assignment did not propagate across trial boundaries. This reflected the episodic structure of the task, where each trial represented an independent decision problem.

Hyperparameters

Training hyperparameters were held constant across all models to ensure fair comparison. Table 4.1 summarizes the key hyperparameters used.

Table 4.1: Neural network training hyperparameters

Parameter	Value	Description
Architecture		
Token size	64	Dimension of observation/action embeddings
RNN hidden size	256	Number of recurrent units
Critic hidden size	64	Hidden layer dimension
Distance hidden size	64	Hidden layer dimension
Hindsight hidden size	256	Hidden layer dimension
Reward velocity hidden size	128	Hidden layer dimension
Training		
Trials per task	3,000	Number of trials per sub-task
Discount factor (γ)	0.99	Reward discounting
Trace decay (λ)	0.9	Eligibility trace decay
Base learning rate (α)	1.0	Initial step size
Entropy coefficient (τ)	0.01	Entropy regularization weight
ObGD Optimizer		
Policy κ	3.0	Step size scaling for policy
Critic κ	2.0	Step size scaling for critic
Distance κ	2.0	Step size scaling for distance network
Adam Optimizer		
Hindsight α	0.0002	Learning rate
Reward velocity α	0.001	Learning rate
Evaluation		
Evaluation frequency	Every 50 trials	Performance assessment interval
Evaluation episodes	100	Sample size per evaluation

Evaluation Protocol

Model performance was assessed every 100 training trials by running 100 evaluation episodes with the current policy.

For each evaluation episode, we recorded: (1) total reward obtained, (2) accuracy at each trial stage (fixation, first-stage choice, transition viewing, motion discrimination), (3) TD error at each time step, (4) estimated state values, and (5) estimated option values (the critic's value estimate for each of the three possible options in the current sub-task) (6) estimated distance to goal for each model except for M0 that did not use the distance

network. These metrics were averaged across the 100 evaluation episodes to produce a single performance summary at each training checkpoint.

5 Conclusion

This thesis began with Jane, the chimpanzee fishing for termites at the edge of a mound in Gombe. That single act captured the essence of intelligence: reducing uncertainty through strategic information sampling and exploiting environmental structure through abstraction and generalization. The investigations presented here have explored how these two fundamental capacities emerge through the interplay between learning mechanisms and the environments that shape them.

Three interconnected investigations provide converging evidence for this thesis. The first contribution is methodological: a hybrid modeling framework that combines theory-driven cognitive models with data-driven neural networks, using symbolic regression to recover interpretable equations from learned functions. This approach bridges the interpretability required for scientific understanding with the flexibility needed to discover computational principles that may not conform to our initial hypotheses. The framework provides a general method for cognitive neuroscience, demonstrating how machine learning can augment rather than replace traditional modeling approaches.

The second contribution applies this framework to understand how humans compute the value of information during decision-making under uncertainty. Using ultra-high field 7T fMRI, I identified distinct computational roles across neuromodulatory nuclei and cortical regions. The ventral tegmental area encodes opposing signals for information value and selection value, balancing exploration and exploitation. The anterior insula and anterior cingulate cortex guide information sampling strategy. Symbolic regression revealed that information value follows exponential functions integrating evidence from both attended and unattended options, with individual differences in exploration captured through interpretable parameters. These equations generalized beyond the training task, predicting behavior in independent exploration-exploitation paradigms.

The third contribution investigates how episodic memory enables rapid generalization across learning tasks. Computational models implementing abstract two-dimensional maps through grid cell path integration achieved dramatic learning acceleration comparable to animal behavior. The critical mechanism involves backward temporal credit assignment through causal attribution and episodic binding, enabling near-instantaneous transfer after initial learning. These findings generate specific neural predictions testable through pattern analysis and targeted stimulation.

Yet fundamental questions remain. How do these mechanisms develop over individual lifetimes? What evolutionary pressures shaped these particular computational solutions? How can artificial systems acquire similar structured representations without extensive pre-specification? The methods and insights developed here provide tools for addressing these questions, but the answers will require integrating computational modeling, neural measurement, and causal intervention across species and timescales.

The chimpanzee Jane ultimately mastered termite fishing not through exhaustive trial and error but through structured learning: observing others to reduce uncertainty, abstracting the relationship between tool properties and foraging success, and generalizing this knowledge across contexts. This thesis suggests that such intelligence (whether in chimpanzees, humans, or machines) emerges not from raw computational power but from the discovery and exploitation of structure in both the world and the representations that model it.

References

[1] Richard C Anderson, Rand J Spiro, and Mark C Anderson. Schemata as scaffolding for the representation of information in connected discourse. 15(3):433–440. ISSN 0002-8312. doi: 10.3102/00028312015003433. URL https://doi.org/10.3102/00028312015003433.

[2] Christopher Baldassano, Janice Chen, Asieh Zadbood, Jonathan W. Pillow, Uri Hasson, and Kenneth A. Norman. Discovering event structure in continuous narrative perception and memory. 95(3):709–721.e5. ISSN 0896-6273. doi: 10.1016/j.neuron.2017.06.041. URL https://www.sciencedirect.com/science/article/pii/S0896627317305937.

[3] Paul M. Bays, Sebastian Schneegans, Wei Ji Ma, and Timothy F. Brady. Representation and computation in visual working memory. *Nature Human Behaviour*, 8(6): 1016–1034, June 2024. ISSN 2397-3374. doi: 10.1038/s41562-024-01871-2. URL https://www.nature.com/articles/s41562-024-01871-2. Publisher: Nature Publishing Group.

[4] Jacob Beck, Risto Vuorio, Evan Zheran Liu, Zheng Xiong, Luisa Zintgraf, Chelsea Finn, and Shimon Whiteson. A tutorial on meta-reinforcement learning. 18(2–3): 224–384. ISSN 1935-8237, 1935-8245. doi: 10.1561/2200000080. URL http://arxiv.org/abs/2301.08028.

[5] Timothy E. J. Behrens, Mark W. Woolrich, Mark E. Walton, and Matthew F. S. Rushworth. Learning the value of information in an uncertain world. 10(9):1214–1221. ISSN 1097-6256. doi: 10.1038/nn1954.

[6] Oded Bein and Yael Niv. Schemas, reinforcement learning and the medial prefrontal cortex. 26(3):141–157. ISSN 1471-0048. doi: 10.1038/s41583-024-00893-z. URL https://www.nature.com/articles/s41583-024-00893-z.

[7] Alessandro Bongioanni, Davide Folloni, Lennart Verhagen, Jérôme Sallet, Miriam C. Klein-Flügge, and Matthew F. S. Rushworth. Activation and disruption of a neural mechanism for novel choice in monkeys. 591(7849):270–274. ISSN 1476-4687. doi: 10.1038/s41586-020-03115-5. URL https://www.nature.com/articles/s41586-020-03115-5.

[8] Lara M. Boyle, Lorenzo Posani, Sarah Irfan, Steven A. Siegelbaum, and Stefano Fusi. Tuned geometries of hippocampal representations meet the computational demands

of social memory. 112(8):1358–1371.e9. ISSN 0896-6273. doi: 10.1016/j.neuron.2024.01.021. URL https://www.cell.com/neuron/abstract/S0896-6273(24)00047-3.

[9] Ethan S. Bromberg-Martin and Okihide Hikosaka. Midbrain dopamine neurons signal preference for advance information about upcoming rewards. *Neuron*, 63(1): 119–126, July 2009. ISSN 0896-6273. doi: 10.1016/j.neuron.2009.06.009. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2723053/.

[10] Ethan S. Bromberg-Martin and Ilya E. Monosov. Neural circuitry of information seeking. *Current Opinion in Behavioral Sciences*, 35:62–70, October 2020. ISSN 2352-1546. doi: 10.1016/j.cobeha.2020.07.006.

[11] Ethan S. Bromberg-Martin, Masayuki Matsumoto, and Okihide Hikosaka. Dopamine in Motivational Control: Rewarding, Aversive, and Alerting. 68(5):815–834. ISSN 0896-6273. doi: 10.1016/j.neuron.2010.11.022. URL https://www.sciencedirect.com/science/article/pii/S0896627310009384.

[12] Ethan S. Bromberg-Martin, Yang-Yang Feng, Takaya Ogasawara, J. Kael White, Kaining Zhang, and Ilya E. Monosov. A neural mechanism for conserved value computations integrating information and rewards. *Nature Neuroscience*, 27(1): 159–175, January 2024. ISSN 1546-1726. doi: 10.1038/s41593-023-01511-4. URL https://www.nature.com/articles/s41593-023-01511-4. Publisher: Nature Publishing Group.

[13] Jonathan C. W. Brooks, Christian F. Beckmann, Karla L. Miller, Richard G. Wise, Carlo A. Porro, Irene Tracey, and Mark Jenkinson. Physiological noise modelling for spinal functional magnetic resonance imaging studies. *NeuroImage*, 39(2):680–692, January 2008. ISSN 1053-8119. doi: 10.1016/j.neuroimage.2007.09.018.

[14] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901, 2020.

[15] Yoram Burak and Ila R. Fiete. Accurate path integration in continuous attractor network models of grid cells. 5(2):e1000291. ISSN 1553-7358. doi: 10.1371/journal.

pcbi.1000291. URL https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1000291.

[16] Frederick Callaway. Cognition as a sequential decision problem. URL https://search.proquest.com/openview/91e0b9f7a561a80bbc46ecaa04fa03f7/1?pq-origsite=gscholar&cbl=18750&diss=y.

[17] Frederick Callaway, Sayan Gul, Paul M. Krueger, Thomas L. Griffiths, and Falk Lieder. Learning to select computations, August 2018. URL http://arxiv.org/abs/1711.06892. arXiv:1711.06892 [cs].

[18] Pablo Samuel Castro, Nenad Tomasev, Ankit Anand, Navodita Sharma, Rishika Mohanta, Aparna Dev, Kuba Perlin, Siddhant Jain, Kyle Levin, Noémi Éltető, Will Dabney, Alexander Novikov, Glenn C. Turner, Maria K. Eckstein, Nathaniel D. Daw, Kevin J. Miller, and Kimberly L. Stachenfeld. Discovering symbolic cognitive models from human and animal behavior, February 2025. URL https://www.biorxiv.org/content/10.1101/2025.02.05.636732v1. Pages: 2025.02.05.636732 Section: New Results.

[19] Franco Cauda, Federico D'Agata, Katiuscia Sacco, Sergio Duca, Giuliano Geminiani, and Alessandro Vercelli. Functional connectivity of the insula in the resting brain. 55 (1):8–23. ISSN 1095-9572. doi: 10.1016/j.neuroimage.2010.11.049.

[20] Sarthak Chandra, Sugandha Sharma, Rishidev Chaudhuri, and Ila Fiete. Episodic and associative memory from spatial scaffolds in the hippocampus. 638(8051):739–751. ISSN 1476-4687. doi: 10.1038/s41586-024-08392-y. URL https://www.nature.com/articles/s41586-024-08392-y.

[21] T. Chiba, T. Kayahara, and K. Nakano. Efferent projections of infralimbic and prelimbic areas of the medial prefrontal cortex in the Japanese monkey, Macaca fuscata. *Brain Research*, 888(1):83–101, January 2001. ISSN 0006-8993. doi: 10.1016/s0006-8993(00)03013-4.

[22] SueYeon Chung and L. F. Abbott. Neural population geometry: An approach for understanding biological and artificial neural networks. 70:137–144. ISSN 0959-4388. doi: 10.1016/j.conb.2021.10.010. URL https://www.sciencedirect.com/science/article/pii/S0959438821001227.

[23] Luke Clark, Bettina Studer, Joel Bruss, Daniel Tranel, and Antoine Bechara. Damage to insula abolishes cognitive distortions during simulated gambling. 111(16): 6098–6103. ISSN 1091-6490. doi: 10.1073/pnas.1322295111.

[24] Jeremiah Y. Cohen, Sebastian Haesler, Linh Vong, Bradford B. Lowell, and Naoshige Uchida. Neuron-type-specific signals for reward and punishment in the ventral tegmental area. 482(7383):85–88. ISSN 1476-4687. doi: 10.1038/nature10754. URL https://www.nature.com/articles/nature10754.

[25] Alexandra O. Constantinescu, Jill X. O'Reilly, and Timothy E. J. Behrens. Organizing conceptual knowledge in humans with a gridlike code. 352(6292):1464–1468. doi: 10.1126/science.aaf0941. URL https://www.science.org/doi/10.1126/science.aaf0941.

[26] Hristos S. Courellis, Juri Minxha, Araceli R. Cardenas, Daniel L. Kimmel, Chrystal M. Reed, Taufik A. Valiante, C. Daniel Salzman, Adam N. Mamelak, Stefano Fusi, and Ueli Rutishauser. Abstract representations emerge in human hippocampal neurons during inference. 632(8026):841–849. ISSN 1476-4687. doi: 10.1038/s41586-024-07799-x. URL https://www.nature.com/articles/s41586-024-07799-x.

[27] Miles Cranmer. Interpretable Machine Learning for Science with PySR and SymbolicRegression.jl. URL http://arxiv.org/abs/2305.01582.

[28] Miles Cranmer. Interpretable Machine Learning for Science with PySR and SymbolicRegression.jl, May 2023. URL http://arxiv.org/abs/2305.01582. arXiv:2305.01582 [astro-ph].

[29] Miles Cranmer, Alvaro Sanchez-Gonzalez, Peter Battaglia, Rui Xu, Kyle Cranmer, David Spergel, and Shirley Ho. Discovering symbolic models from deep learning with inductive biases, November 2020. URL http://arxiv.org/abs/2006.11287. arXiv:2006.11287 [cs].

[30] Simone D'Ambrogio, Jan Grohn, Nima Khalighinejad, Marcelo Mattar, Laurence Hunt, and Matthew F. S. Rushworth. Interpretable abstractions of artificial neural networks predict behavior and neural activity during human information gathering, June 2025.

[31] M. Leann Dodd, Kevin J. Klos, James H. Bower, Yonas E. Geda, Keith A. Josephs, and J. Eric Ahlskog. Pathological gambling caused by drugs used to treat parkinson disease. 62(9):1377–1381. ISSN 0003-9942. doi: 10.1001/archneur.62.9.noc50009.

[32] Jan Drugowitsch, Rubén Moreno-Bote, Anne K. Churchland, Michael N. Shadlen, and Alexandre Pouget. The cost of accumulating evidence in perceptual decision making. *Journal of Neuroscience*, 32(11):3612–3628, March 2012. ISSN 0270-6474,

1529-2401. doi: 10.1523/JNEUROSCI.4010-11.2012. URL https://www.jneurosci.org/content/32/11/3612. Publisher: Society for Neuroscience Section: Articles.

[33] 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, May 2023. URL https://www.biorxiv.org/content/10.1101/2023.05.17.541226v1. Pages: 2023.05.17.541226 Section: New Results.

[34] Maria K. Eckstein, Christopher Summerfield, Nathaniel Daw, and Kevin J. Miller. Hybrid Neural-Cognitive Models Reveal How Memory Shapes Human Reward Learning, 2024. URL https://osf.io/u9ks4/download.

[35] Mohamady El-Gaby, Adam Loyd Harris, James C. R. Whittington, William Dorrell, Arya Bhomick, Mark E. Walton, Thomas Akam, and Timothy E. J. Behrens. A cellular basis for mapping behavioural structure. 636(8043):671–680. ISSN 1476-4687. doi: 10.1038/s41586-024-08145-x. URL https://www.nature.com/articles/s41586-024-08145-x.

[36] Mohamed Elsayed, Gautham Vasan, and A. Rupam Mahmood. Streaming deep reinforcement learning finally works. URL http://arxiv.org/abs/2410.14606.

[37] Zeming Fang and Chris R. Sims. Humans learn generalizable representations through efficient coding. 16(1):3989. ISSN 2041-1723. doi: 10.1038/s41467-025-58848-6. URL https://www.nature.com/articles/s41467-025-58848-6.

[38] Valeria Fascianelli, Aldo Battista, Fabio Stefanini, Satoshi Tsujimoto, Aldo Genovesio, and Stefano Fusi. Neural representational geometries reflect behavioral differences in monkeys and recurrent neural networks. 15(1):6479. ISSN 2041-1723. doi: 10.1038/s41467-024-50503-w. URL https://www.nature.com/articles/s41467-024-50503-w.

[39] Lief Fenno, Ofer Yizhar, and Karl Deisseroth. The Development and Application of Optogenetics. 34:389–412. ISSN 0147-006X, 1545-4126. doi: 10.1146/annurev-neuro-061010-113817. URL https://www.annualreviews.org/content/journals/10.1146/annurev-neuro-061010-113817/.

[40] Christopher D. Fiorillo, Philippe N. Tobler, and Wolfram Schultz. Discrete coding of reward probability and uncertainty by dopamine neurons. *Science (New York, N.Y.)*, 299(5614):1898–1902, March 2003. ISSN 1095-9203. doi: 10.1126/science.1077349.

[41] Nicholas T. Franklin, Kenneth A. Norman, Charan Ranganath, Jeffrey M. Zacks, and Samuel J. Gershman. Structured event memory: A neuro-symbolic model of event cognition. 127(3):327–361. ISSN 1939-1471. doi: 10.1037/rev0000177.

[42] Samuel J. Gershman. Deconstructing the human algorithms for exploration. *Cognition*, 173:34–42, April 2018. ISSN 0010-0277. doi: 10.1016/j.cognition.2017.12.014. URL https://www.sciencedirect.com/science/article/pii/S0010027717303359.

[43] Samuel J. Gershman, Eric J. Horvitz, and Joshua B. Tenenbaum. Computational rationality: A converging paradigm for intelligence in brains, minds, and machines. 349(6245):273–278. doi: 10.1126/science.aac6076. URL https://www.science.org/doi/10.1126/science.aac6076.

[44] Moshe Glickman and Tali Sharot. How human–ai feedback loops alter human perceptual, emotional and social judgements. *Nature Human Behaviour*, 9(2):345–359, February 2025. ISSN 2397-3374. doi: 10.1038/s41562-024-02077-2. URL https://www.nature.com/articles/s41562-024-02077-2. Publisher: Nature Publishing Group.

[45] Paul W. Glimcher, Colin F. Camerer, Ernst Fehr, and Russell A. Poldrack, editors. *Neuroeconomics: Decision Making and the Brain*. Academic Press, London, 2009.

[46] Amit Goldenberg, Kyle LaFollette, Janni Yuval, Roey Schurr, and David Melnikoff. Data driven equation discovery reveals non-linear reinforcement learning in humans. OSF Preprints, 2025. URL https://osf.io/65jqh_v3/. Available at OSF.

[47] Vishwa Goudar, Barbara Peysakhovich, David J. Freedman, Elizabeth A. Buffalo, and Xiao-Jing Wang. Schema formation in a neural population subspace underlies learning-to-learn in flexible sensorimotor problem-solving. 26(5):879–890. ISSN 1546-1726. doi: 10.1038/s41593-023-01293-9. URL https://www.nature.com/articles/s41593-023-01293-9.

[48] David M. Green and John A. Swets. *Signal detection theory and psychophysics*. Signal detection theory and psychophysics. John Wiley, Oxford, England, 1966. Pages: xi, 455.

[49] Douglas N. Greve and Bruce Fischl. Accurate and robust brain image alignment using boundary-based registration. *NeuroImage*, 48(1):63–72, October 2009. ISSN 1095-9572. doi: 10.1016/j.neuroimage.2009.06.060.

[50] Cooper D. Grossman, Bilal A. Bari, and Jeremiah Y. Cohen. Serotonin neurons modulate learning rate through uncertainty. *Current biology: CB*, 32(3):586–599.e7, February 2022. ISSN 1879-0445. doi: 10.1016/j.cub.2021.12.006.

[51] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. 640(8059):647–653. ISSN 1476-4687. doi: 10.1038/s41586-025-08744-2. URL https://www.nature.com/articles/s41586-025-08744-2.

[52] Torkel Hafting, Marianne Fyhn, Sturla Molden, May-Britt Moser, and Edvard I. Moser. Microstructure of a spatial map in the entorhinal cortex. 436(7052):801–806. ISSN 1476-4687. doi: 10.1038/nature03721. URL https://www.nature.com/articles/nature03721.

[53] Timothy Hanks, Roozbeh Kiani, and Michael N Shadlen. A neural mechanism of speed-accuracy tradeoff in macaque area LIP. *eLife*, 3:e02260, May 2014. ISSN 2050-084X. doi: 10.7554/eLife.02260. URL https://doi.org/10.7554/eLife.02260. Publisher: eLife Sciences Publications, Ltd.

[54] Timothy D. Hanks, Mark E. Mazurek, Roozbeh Kiani, Elisabeth Hopp, and Michael N. Shadlen. Elapsed decision time affects the weighting of prior probability in a perceptual decision task. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience*, 31(17):6339–6352, April 2011. ISSN 1529-2401. doi: 10.1523/JNEUROSCI.5613-10.2011.

[55] Harry F. Harlow. The formation of learning sets. 56(1):51–65. ISSN 1939-1471. doi: 10.1037/h0062474.

[56] Anna Harutyunyan, Will Dabney, Thomas Mesnard, Mohammad Azar, Bilal Piot, Nicolas Heess, prefix=van useprefix=false family=Hasselt, given=Hado, Greg Wayne, Satinder Singh, Doina Precup, and Remi Munos. Hindsight credit assignment. URL http://arxiv.org/abs/1912.02503.

[57] Eduardo Hernández-Ortiz, Jorge Luis-Islas, Fatuel Tecuapetla, Ranier Gutierrez, and Federico Bermúdez-Rattoni. Top-down circuitry from the anterior insular cortex to VTA dopamine neurons modulates reward-related memory. 42(11). ISSN 2211-1247. doi: 10.1016/j.celrep.2023.113365. URL https://www.cell.com/cell-reports/abstract/S2211-1247(23)01377-3.

[58] Okihide Hikosaka. The habenula: from stress evasion to value-based decision-making. *Nature Reviews Neuroscience*, 11(7):503–513, July 2010. ISSN 1471-0048. doi:

10.1038/nrn2866. URL https://www.nature.com/articles/nrn2866. Publisher: Nature Publishing Group.

[59] Jeffrey R. Hollerman and Wolfram Schultz. Dopamine neurons report an error in the temporal prediction of reward during learning. 1(4):304–309. ISSN 1546-1726. doi: 10.1038/1124. URL https://www.nature.com/articles/nn0898_304.

[60] Laurence T. Hunt, W. M. Nishantha Malalasekera, prefix=de useprefix=true family=Berker, given=Archy O., Bruno Miranda, Simon F. Farmer, Timothy E. J. Behrens, and Steven W. Kennerley. Triple dissociation of attention and decision computations across prefrontal cortex. 21(10):1471–1481. ISSN 1546-1726. doi: 10.1038/s41593-018-0239-5. URL https://www.nature.com/articles/s41593-018-0239-5.

[61] Laurence T. Hunt, Robb B. Rutledge, W. M. Nishantha Malalasekera, Steven W. Kennerley, and Raymond J. Dolan. Approach-Induced Biases in Human Information Sampling. *PLOS Biology*, 14(11):e2000638, November 2016. ISSN 1545-7885. doi: 10.1371/journal.pbio.2000638. URL https://journals.plos.org/plosbiology/article?id=10.1371/journal.pbio.2000638. Publisher: Public Library of Science.

[62] Laurence T. Hunt, W. M. Nishantha Malalasekera, Archy O. de Berker, Bruno Miranda, Simon F. Farmer, Timothy E. J. Behrens, and Steven W. Kennerley. Triple dissociation of attention and decision computations across prefrontal cortex. *Nature Neuroscience*, 21(10):1471–1481, October 2018. ISSN 1546-1726. doi: 10.1038/s41593-018-0239-5. URL https://www.nature.com/articles/s41593-018-0239-5. Publisher: Nature Publishing Group.

[63] Hironori Ishii, Shinya Ohara, Philippe N. Tobler, Ken-Ichiro Tsutsui, and Toshio Iijima. Inactivating anterior insular cortex reduces risk taking. 32(45):16031–16039. ISSN 1529-2401. doi: 10.1523/JNEUROSCI.2278-12.2012.

[64] Mark Jenkinson, Peter Bannister, Michael Brady, and Stephen Smith. Improved Optimization for the Robust and Accurate Linear Registration and Motion Correction of Brain Images. *NeuroImage*, 17(2):825–841, October 2002. ISSN 1053-8119. doi: 10.1006/nimg.2002.1132. URL https://www.sciencedirect.com/science/article/pii/S1053811902911328.

[65] Mark Jenkinson, Christian F. Beckmann, Timothy E. J. Behrens, Mark W. Woolrich, and Stephen M. Smith. FSL. *NeuroImage*, 62(2):782–790, August 2012. ISSN 1053-8119. doi: 10.1016/j.neuroimage.2011.09.015. URL https://www.sciencedirect.com/science/article/pii/S1053811911010603.

[66] Li Ji-An, Marcus K. Benna, and Marcelo G. Mattar. Discovering Cognitive Strategies with Tiny Recurrent Neural Networks, October 2024. URL https://www.biorxiv.org/content/10.1101/2023.04.12.536629v3. Pages: 2023.04.12.536629 Section: New Results.

[67] Siddhartha Joshi and Joshua I. Gold. Context-dependent relationships between locus coeruleus firing patterns and coordinated neural activity in the anterior cingulate cortex. *eLife*, 11:e63490, January 2022. ISSN 2050-084X. doi: 10.7554/eLife.63490.

[68] Siddhartha Joshi, Yin Li, Rishi M. Kalwani, and Joshua I. Gold. Relationships between Pupil Diameter and Neuronal Activity in the Locus Coeruleus, Colliculi, and Cingulate Cortex. *Neuron*, 89(1):221–234, January 2016. ISSN 1097-4199. doi: 10.1016/j.neuron.2015.11.028.

[69] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with AlphaFold. 596(7873):583–589, 2021. doi: 10.1038/s41586-021-03819-2.

[70] Paula Kaanders, Hamed Nili, Jill X. O'Reilly, and Laurence Hunt. Medial Frontal Cortex Activity Predicts Information Sampling in Economic Choice. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience*, 41(40):8403–8413, October 2021. ISSN 1529-2401. doi: 10.1523/JNEUROSCI.0392-21.2021.

[71] Nina N. Karpova, Anouchka Pickenhagen, Jesse Lindholm, Ettore Tiraboschi, Natalia Kulesskaya, Arna Agústsdóttir, Hanna Antila, Dina Popova, Yumiko Akamine, Amine Bahi, Regina Sullivan, René Hen, Liam J. Drew, and Eero Castrén. Fear erasure in mice requires synergy between antidepressant drugs and extinction training. 334(6063):1731–1734. ISSN 1095-9203. doi: 10.1126/science.1214592.

[72] Christopher A. Kelly and Tali Sharot. Web-browsing patterns reflect and shape mood and mental health. *Nature Human Behaviour*, 9(1):133–146, January 2025. ISSN 2397-3374. doi: 10.1038/s41562-024-02065-6. URL https://www.nature.com/articles/s41562-024-02065-6. Publisher: Nature Publishing Group.

[73] Nima Khalighinejad, Alessandro Bongioanni, Lennart Verhagen, Davide Folloni, David Attali, Jean-Francois Aubry, Jerome Sallet, and Matthew F. S. Rushworth. A Basal Forebrain-Cingulate Circuit in Macaques Decides It Is Time to Act. *Neuron*, 105(2):370–384.e8, January 2020. ISSN 1097-4199. doi: 10.1016/j.neuron.2019.10.030.

[74] Nima Khalighinejad, Luke Priestley, Saad Jbabdi, and Matthew F. S. Rushworth. Human decisions about when to act originate within a basal forebrain-nigral circuit. *Proceedings of the National Academy of Sciences of the United States of America*, 117(21):11799–11810, May 2020. ISSN 1091-6490. doi: 10.1073/pnas.1921211117.

[75] Nima Khalighinejad, Neil Garrett, Luke Priestley, Patricia Lockwood, and Matthew F. S. Rushworth. A habenula-insular circuit encodes the willingness to act. *Nature Communications*, 12(1):6329, November 2021. ISSN 2041-1723. doi: 10.1038/s41467-021-26569-1. URL https://www.nature.com/articles/s41467-021-26569-1. Publisher: Nature Publishing Group.

[76] Nima Khalighinejad, Sanjay Manohar, Masud Husain, and Matthew F. S. Rushworth. Complementary roles of serotonergic and cholinergic systems in decisions about when to act. *Current biology: CB*, 32(5):1150–1162.e7, March 2022. ISSN 1879-0445. doi: 10.1016/j.cub.2022.01.042.

[77] Celeste Kidd and Benjamin Y. Hayden. The psychology and neuroscience of curiosity. *Neuron*, 88(3):449–460, November 2015. ISSN 0896-6273. doi: 10.1016/j.neuron.2015.09.010. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4635443/.

[78] Jae-Chang Kim, Lydia Hellrung, Marcus Grueschow, Stephan Nebe, Zoltan Nagy, and Philippe N. Tobler. Neural Representation of Valenced and Generic Probability and Uncertainty. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience*, 44(30):e0195242024, July 2024. ISSN 1529-2401. doi: 10.1523/JNEUROSCI.0195-24.2024.

[79] Jae-Chang Kim, Lydia Hellrung, Stephan Nebe, and Philippe N. Tobler. The Anterior Insula Processes a Time-Resolved Subjective Risk Prediction Error. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience*, 45(23):e2302242025, June 2025. ISSN 1529-2401. doi: 10.1523/JNEUROSCI.2302-24.2025.

[80] Samuel Kim, Peter Y. Lu, Srijon Mukherjee, Michael Gilbert, Li Jing, Vladimir Čeperić, and Marin Soljačić. Integration of Neural Network-Based Symbolic Regression in Deep Learning for Scientific Discovery. *IEEE Transactions on Neural Networks and Learning Systems*, 32(9):4166–4177, September 2021. ISSN 2162-2388. doi:

10.1109/TNNLS.2020.3017010. URL https://ieeexplore.ieee.org/abstract/document/9180100.

[81] Miriam C. Klein-Flügge, Marco K. Wittmann, Anna Shpektor, Daria E. A. Jensen, and Matthew F. S. Rushworth. Multiple associative structures created by reinforcement and incidental statistical learning mechanisms. 10(1):4835. ISSN 2041-1723. doi: 10.1038/s41467-019-12557-z. URL https://www.nature.com/articles/s41467-019-12557-z.

[82] Miriam C. Klein-Flügge, Alessandro Bongioanni, and Matthew F. S. Rushworth. Medial and orbital frontal cortex in decision-making and flexible behavior. *Neuron*, 110(17):2743–2770, September 2022. ISSN 0896-6273. doi: 10.1016/j.neuron.2022.05.022. URL https://www.sciencedirect.com/science/article/pii/S0896627322004639.

[83] Nils Kolling, Timothy E. J. Behrens, Rogier B. Mars, and Matthew F. S. Rushworth. Neural mechanisms of foraging. 336(6077):95–98. doi: 10.1126/science.1216930. URL https://www.science.org/doi/full/10.1126/science.1216930.

[84] Ian Krajbich. Accounting for attention in sequential sampling models of decision making. *Current Opinion in Psychology*, 29:6–11, October 2019. ISSN 2352-250X. doi: 10.1016/j.copsyc.2018.10.008. URL https://www.sciencedirect.com/science/article/pii/S2352250X18301866.

[85] Zhi-Wei Li and Wei Ji Ma. An uncertainty-based model of the effects of fixation on choice. *PLOS Computational Biology*, 17(8):e1009190, August 2021. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1009190. URL https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1009190. Publisher: Public Library of Science.

[86] Louise Marshall, Christoph Mathys, Diane Ruge, Archy O. de Berker, Peter Dayan, Klaas E. Stephan, and Sven Bestmann. Pharmacological Fingerprints of Contextual Uncertainty. *PLOS Biology*, 14(11):e1002575, November 2016. ISSN 1545-7885. doi: 10.1371/journal.pbio.1002575. URL https://journals.plos.org/plosbiology/article?id=10.1371/journal.pbio.1002575. Publisher: Public Library of Science.

[87] Masayuki Matsumoto and Okihide Hikosaka. Lateral habenula as a source of negative reward signals in dopamine neurons. *Nature*, 447(7148):1111–1115, June 2007. ISSN 1476-4687. doi: 10.1038/nature05860. URL https://www.nature.com/articles/nature05860. Publisher: Nature Publishing Group.

[88] Thomas Mesnard, Théophane Weber, Fabio Viola, Shantanu Thakoor, Alaa Saade, Anna Harutyunyan, Will Dabney, Tom Stepleton, Nicolas Heess, Arthur Guez, Éric Moulines, Marcus Hutter, Lars Buesing, and Rémi Munos. Counterfactual credit assignment in model-free reinforcement learning, December 2021.

[89] Alexander Meulemans, Simon Schug, Seijin Kobayashi, Nathaniel Daw, and Gregory Wayne. Would i have gotten that reward? long-term credit assignment by counterfactual contribution analysis. URL http://arxiv.org/abs/2306.16803.

[90] John G. Mikhael and Rafal Bogacz. Learning reward uncertainty in the basal ganglia. 12(9):e1005062. ISSN 1553-734X. doi: 10.1371/journal.pcbi.1005062. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5010205/.

[91] 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. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/c194ced51c857ec2c1928b02250e0ac8-Abstract-Conference.html.

[92] Kentaro Miyamoto, Nadescha Trudel, Kevin Kamermans, Michele C. Lim, Alberto Lazari, Lennart Verhagen, Marco K. Wittmann, and Matthew F. S. Rushworth. Identification and disruption of a neural mechanism for accumulating prospective metacognitive information prior to decision-making. 109(8):1396–1408.e7. ISSN 0896-6273. doi: 10.1016/j.neuron.2021.02.024. URL https://www.cell.com/neuron/abstract/S0896-6273(21)00124-0.

[93] Hiroyuki Mizoguchi, Kentaro Katahira, Ayumu Inutsuka, Kazuya Fukumoto, Akihiro Nakamura, Tian Wang, Taku Nagai, Jun Sato, Makoto Sawada, Hideki Ohira, Akihiro Yamanaka, and Kiyofumi Yamada. Insular neural system controls decision-making in healthy and methamphetamine-treated rats. 112(29):E3930–3939. ISSN 1091-6490. doi: 10.1073/pnas.1418014112.

[94] Ilya E. Monosov. Anterior cingulate is a source of valence-specific information about value and uncertainty. *Nature Communications*, 8(1):134, July 2017. ISSN 2041-1723. doi: 10.1038/s41467-017-00072-y. URL https://www.nature.com/articles/s41467-017-00072-y. Publisher: Nature Publishing Group.

[95] Ilya E. Monosov. Curiosity: primate neural circuits for novelty and information seeking. *Nature Reviews Neuroscience*, 25(3):195–208, March 2024. ISSN 1471-0048. doi: 10.1038/s41583-023-00784-9. URL https://www.nature.com/articles/s41583-023-00784-9. Publisher: Nature Publishing Group.

[96] Ilya E. Monosov, Suzanne N. Haber, Eric C. Leuthardt, and Ahmad Jezzini. Anterior Cingulate Cortex and the Control of Dynamic Behavior in Primates. *Current biology: CB*, 30(23):R1442–R1454, December 2020. ISSN 1879-0445. doi: 10.1016/j.cub.2020.10.009.

[97] Moritz Möller and Rafal Bogacz. Learning the payoffs and costs of actions. 15(2): e1006285. ISSN 1553-734X. doi: 10.1371/journal.pcbi.1006285. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6413954/.

[98] Tomaso Poggio. Plumbing the depths of neural nets, 2018. URL https://betterworld.mit.edu/spectrum/issues/winter-2018/plumbing-the-depths-of-neural-nets.

[99] Kerstin Preuschoff, Peter Bossaerts, and Steven R. Quartz. Neural differentiation of expected reward and risk in human subcortical structures. 51(3):381–390, . ISSN 0896-6273. doi: 10.1016/j.neuron.2006.06.024.

[100] Kerstin Preuschoff, Steven R. Quartz, and Peter Bossaerts. Human Insula Activation Reflects Risk Prediction Errors As Well As Risk. 28(11):2745–2752, . ISSN 0270-6474, 1529-2401. doi: 10.1523/JNEUROSCI.4286-07.2008. URL https://www.jneurosci.org/content/28/11/2745.

[101] Matthew R. Roesch, Donna J. Calu, and Geoffrey Schoenbaum. Dopamine neurons encode the better option in rats deciding between differently delayed or sized rewards. 10(12):1615–1624. ISSN 1546-1726. doi: 10.1038/nn2013. URL https://www.nature.com/articles/nn2013.

[102] Veronika Samborska, James L. Butler, Mark E. Walton, Timothy E. J. Behrens, and Thomas Akam. Complementary task representations in hippocampus and prefrontal cortex for generalizing the structure of problems. 25(10):1314–1326. ISSN 1546-1726. doi: 10.1038/s41593-022-01149-8. URL https://www.nature.com/articles/s41593-022-01149-8.

[103] W. Schultz, P. Dayan, and P. R. Montague. A neural substrate of prediction and reward. 275(5306):1593–1599. ISSN 0036-8075. doi: 10.1126/science.275.5306.1593.

[104] Wolfram Schultz. Behavioral theories and the neurophysiology of reward. *Annual Review of Psychology*, 57:87–115, 2006. ISSN 0066-4308. doi: 10.1146/annurev.psych.56.091103.070229.

[105] Eric Schulz and Samuel J. Gershman. The algorithmic architecture of exploration in the human brain. 55:7–14. ISSN 0959-4388. doi: 10.1016/j.conb.2018.11.003. URL https://www.sciencedirect.com/science/article/pii/S0959438818300904.

[106] Michael N. Shadlen and Roozbeh Kiani. Decision making as a window on cognition. *Neuron*, 80(3):791–806, October 2013. ISSN 1097-4199. doi: 10.1016/j.neuron.2013.10.047.

[107] Michael N. Shadlen and Daphna Shohamy. Decision Making and Sequential Sampling from Memory. *Neuron*, 90(5):927–939, June 2016. ISSN 1097-4199. doi: 10.1016/j.neuron.2016.04.036.

[108] Tali Sharot and Cass R. Sunstein. How people decide what they want to know. *Nature Human Behaviour*, 4(1):14–19, January 2020. ISSN 2397-3374. doi: 10.1038/s41562-019-0793-1. URL https://www.nature.com/articles/s41562-019-0793-1. Publisher: Nature Publishing Group.

[109] Stephen M. Smith. Fast robust automated brain extraction. *Human Brain Mapping*, 17(3):143–155, November 2002. ISSN 1065-9471. doi: 10.1002/hbm.10062.

[110] Stephen M. Smith, Mark Jenkinson, Mark W. Woolrich, Christian F. Beckmann, Timothy E. J. Behrens, Heidi Johansen-Berg, Peter R. Bannister, Marilena De Luca, Ivana Drobnjak, David E. Flitney, Rami K. Niazy, James Saunders, John Vickers, Yongyue Zhang, Nicola De Stefano, J. Michael Brady, and Paul M. Matthews. Advances in functional and structural MR image analysis and implementation as FSL. *NeuroImage*, 23 Suppl 1:S208–219, 2004. ISSN 1053-8119. doi: 10.1016/j.neuroimage.2004.07.051.

[111] Eleanor Spens and Neil Burgess. A generative model of memory construction and consolidation. 8(3):526–543. ISSN 2397-3374. doi: 10.1038/s41562-023-01799-z. URL https://www.nature.com/articles/s41562-023-01799-z.

[112] Elizabeth E. Steinberg, Ronald Keiflin, Josiah R. Boivin, Ilana B. Witten, Karl Deisseroth, and Patricia H. Janak. A causal link between prediction errors, dopamine neurons and learning. 16(7):966–973. ISSN 1546-1726. doi: 10.1038/nn.3413. URL https://www.nature.com/articles/nn.3413.

[113] David W. Stephens, Joel S. Brown, and Ronald C. Ydenberg, editors. *Foraging: Behavior and Ecology*. University of Chicago Press, Chicago, IL, 2007.

[114] Emily R. Stern, Richard Gonzalez, Robert C. Welsh, and Stephan F. Taylor. Updating beliefs for a decision: Neural correlates of uncertainty and underconfidence. 30 (23):8032–8041. ISSN 1529-2401. doi: 10.1523/JNEUROSCI.4729-09.2010.

[115] Gabriel M. Stine, Eric M. Trautmann, Danique Jeurissen, and Michael N. Shadlen. A neural mechanism for terminating decisions. *Neuron*, 111(16):2601–2613.e5, August 2023. ISSN 1097-4199. doi: 10.1016/j.neuron.2023.05.028.

[116] Richard S. Sutton and Andrew G. Barto. Reinforcement learning: An introduction. URL http://incompleteideas.net/book/the-book-2nd.html.

[117] Richard S. Sutton and Andrew G. Barto. *Reinforcement learning: An introduction*. MIT Press, Cambridge, MA, 1998. URL https://www.cambridge.org/core/journals/robotica/article/robot-learning-edited-by-jonathan-h-connell-and-sridhar-mahadevan-kluwer-boston-19737FD21CA908246DF17779E9C20B6DF6.

[118] Jonathan Tapson and prefix=van useprefix=false family=Schaik, given=Andre. Learning the pseudoinverse solution to network weights. 45:94–100. ISSN 08936080. doi: 10.1016/j.neunet.2013.02.008. URL http://arxiv.org/abs/1207.3368.

[119] D. Gowanlock R. Tervo, Elena Kuleshova, Maxim Manakov, Mikhail Proskurin, Mattias Karlsson, Andy Lustig, Reza Behnam, and Alla Y. Karpova. The anterior cingulate cortex directs exploration of alternative strategies. *Neuron*, 109(11):1876–1887.e6, June 2021. ISSN 1097-4199. doi: 10.1016/j.neuron.2021.03.028.

[120] Dougal G. R. Tervo, Mikhail Proskurin, Maxim Manakov, Mayank Kabra, Alison Vollmer, Kristin Branson, and Alla Y. Karpova. Behavioral variability through stochastic choice and its gating by anterior cingulate cortex. 159(1):21–32. ISSN 0092-8674. doi: 10.1016/j.cell.2014.08.037. URL https://www.sciencedirect.com/science/article/pii/S0092867414011076.

[121] Philippe N. Tobler, Christopher D. Fiorillo, and Wolfram Schultz. Adaptive coding of reward value by dopamine neurons. *Science (New York, N.Y.)*, 307(5715):1642–1645, March 2005. ISSN 1095-9203. doi: 10.1126/science.1105370.

[122] Hailey A. Trier, Nima Khalighinejad, Sorcha Hamilton, Caroline Harbison, Luke Priestley, Mark Laubach, Miriam Klein-Flügge, Jacqueline Scholl, and Matthew F. S. Rushworth. A distributed subcortical circuit linked to instrumental information-seeking about threat. *Proceedings of the National Academy of Sciences of the*

United States of America, 122(3):e2410955121, January 2025. ISSN 1091-6490. doi: 10.1073/pnas.2410955121.

[123] Nadescha Trudel, Jacqueline Scholl, Miriam C. Klein-Flügge, Elsa Fouragnan, Lev Tankelevitch, Marco K. Wittmann, and Matthew F. S. Rushworth. Polarity of uncertainty representation during exploration and exploitation in ventromedial prefrontal cortex. *Nature Human Behaviour*, 5(1):83–98, January 2021. ISSN 2397-3374. doi: 10.1038/s41562-020-0929-3. URL https://www.nature.com/articles/s41562-020-0929-3. Publisher: Nature Publishing Group.

[124] Sander van der Linden. Misinformation: susceptibility, spread, and interventions to immunize the public. *Nature Medicine*, 28(3):460–467, March 2022. ISSN 1546-170X. doi: 10.1038/s41591-022-01713-6. URL https://www.nature.com/articles/s41591-022-01713-6. Publisher: Nature Publishing Group.

[125] Valentina Vellani, Moshe Glickman, and Tali Sharot. Three diverse motives for information sharing. *Communications Psychology*, 2(1):107, November 2024. ISSN 2731-9121. doi: 10.1038/s44271-024-00144-y. URL https://www.nature.com/articles/s44271-024-00144-y. Publisher: Nature Publishing Group.

[126] Sebastijan Veselic, Timothy H. Muller, Elena Gutierrez, Timothy E. J. Behrens, Laurence T. Hunt, James L. Butler, and Steven W. Kennerley. A cognitive map for value-guided choice in the ventromedial prefrontal cortex. 188(12):3259–3273.e22. ISSN 0092-8674, 1097-4172. doi: 10.1016/j.cell.2025.03.038. URL https://www.cell.com/cell/abstract/S0092-8674(25)00388-5.

[127] John von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, Princeton, NJ, 2nd edition, 1947.

[128] Pascale Waelti, Anthony Dickinson, and Wolfram Schultz. Dopamine responses comply with basic assumptions of formal learning theory. 412(6842):43–48. ISSN 1476-4687. doi: 10.1038/35083500. URL https://www.nature.com/articles/35083500.

[129] Jane X. Wang, Zeb Kurth-Nelson, Dhruva Tirumala, Hubert Soyer, Joel Z. Leibo, Remi Munos, Charles Blundell, Dharshan Kumaran, and Matt Botvinick. Learning to reinforcement learn, . URL http://arxiv.org/abs/1611.05763.

[130] Ruigang Wang and Ian Manchester. Direct Parameterization of Lipschitz-Bounded Deep Networks. In *Proceedings of the 40th International Conference on Machine Learning*, pages 36093–36110. PMLR, July 2023. URL https://proceedings.mlr.press/v202/wang23v.html. ISSN: 2640-3498.

[131] Yuhao Wang, Armin Lak, Sanjay G. Manohar, and Rafal Bogacz. Dopamine encoding of novelty facilitates efficient uncertainty-driven exploration. 20(4):e1011516, . ISSN 1553-7358. doi: 10.1371/journal.pcbi.1011516.

[132] Mitsuko Watabe-Uchida, Neir Eshel, and Naoshige Uchida. Neural circuitry of reward prediction error. 40:373–394. ISSN 1545-4126. doi: 10.1146/annurev-neuro-072116-031109.

[133] J. Kael White, Ethan S. Bromberg-Martin, Sarah R. Heilbronner, Kaining Zhang, Julia Pai, Suzanne N. Haber, and Ilya E. Monosov. A neural network for information seeking. 10(1):5168. ISSN 2041-1723. doi: 10.1038/s41467-019-13135-z.

[134] J. Kael White, Ethan S. Bromberg-Martin, Sarah R. Heilbronner, Kaining Zhang, Julia Pai, Suzanne N. Haber, and Ilya E. Monosov. A neural network for information seeking. *Nature Communications*, 10(1):5168, November 2019. ISSN 2041-1723. doi: 10.1038/s41467-019-13135-z. URL https://www.nature.com/articles/s41467-019-13135-z. Publisher: Nature Publishing Group.

[135] James C. R. Whittington, Timothy H. Muller, Shirley Mark, Guifen Chen, Caswell Barry, Neil Burgess, and Timothy E. J. Behrens. The tolman-eichenbaum machine: Unifying space and relational memory through generalization in the hippocampal formation. 183(5):1249–1263.e23. ISSN 0092-8674, 1097-4172. doi: 10.1016/j.cell.2020.10.024. URL https://www.cell.com/cell/abstract/S0092-8674(20)31388-X.

[136] Mark W. Woolrich, Brian D. Ripley, Michael Brady, and Stephen M. Smith. Temporal Autocorrelation in Univariate Linear Modeling of FMRI Data. *NeuroImage*, 14 (6):1370–1386, December 2001. ISSN 1053-8119. doi: 10.1006/nimg.2001.0931. URL https://www.sciencedirect.com/science/article/pii/S1053811901909310.

[137] Mark W. Woolrich, Timothy E. J. Behrens, Christian F. Beckmann, Mark Jenkinson, and Stephen M. Smith. Multilevel linear modelling for FMRI group analysis using Bayesian inference. *NeuroImage*, 21(4):1732–1747, April 2004. ISSN 1053-8119. doi: 10.1016/j.neuroimage.2003.12.023.

[138] Lindsay E. Wyatt, Patrick A. Hewan, Jeremy Hogeveen, R. Nathan Spreng, and Gary R. Turner. Exploration versus exploitation decisions in the human brain: A systematic review of functional neuroimaging and neuropsychological studies. 192: 108740. ISSN 0028-3932. doi: 10.1016/j.neuropsychologia.2023.108740. URL https://www.sciencedirect.com/science/article/pii/S0028393223002749.

[139] Yi Xie, Jaedong Hwang, Carlos Brody, David Tank, and Ila Fiete. A multi-region brain model to elucidate the role of hippocampus in spatially embedded decision-making. URL https://www.biorxiv.org/content/10.1101/2025.05.29.656671v1.

[140] Leora Yetnikoff, Anita Y. Cheng, Heather N. Lavezzi, Kenneth P. Parsley, and Daniel S. Zahm. Sources of input to the rostromedial tegmental nucleus, ventral tegmental area, and lateral habenula compared: A study in rat. *The Journal of Comparative Neurology*, 523(16):2426–2456, November 2015. ISSN 1096-9861. doi: 10.1002/cne.23797.

[141] Angela J. Yu and Peter Dayan. Uncertainty, Neuromodulation, and Attention. 46 (4):681–692. ISSN 0896-6273. doi: 10.1016/j.neuron.2005.04.026. URL https://www.cell.com/neuron/abstract/S0896-6273(05)00362-4.

[142] Angela J. Yu and Peter Dayan. Uncertainty, neuromodulation, and attention. *Neuron*, 46(4):681–692, May 2005. ISSN 0896-6273. doi: 10.1016/j.neuron.2005.04.026.

[143] Kelly A. Zalocusky, Charu Ramakrishnan, Talia N. Lerner, Thomas J. Davidson, Brian Knutson, and Karl Deisseroth. Nucleus accumbens d2r cells signal prior outcomes and control risky decision-making. 531(7596):642–646. ISSN 1476-4687. doi: 10.1038/nature17400. URL https://www.nature.com/articles/nature17400.

[144] Jack Zupko. *John Buridan: Portrait of a Fourteenth-century Arts Master*. University of Notre Dame Press, Notre Dame, IN, 2003.