Structure Learning and Uncertainty-Guided Exploration in the Human Brain

Structure learning and uncertainty-guided exploration in the human brain

A DISSERTATION PRESENTED
BY
MOMCHIL SLAVCHEV TOMOV
TO
THE DIVISION OF MEDICAL SCIENCES

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
IN THE SUBJECT OF
NEUROBIOLOGY

HARVARD UNIVERSITY
CAMBRIDGE, MASSACHUSETTS
DECEMBER 2019

Dissertation Advisor: Samuel Gershman Momchil Slavchev Tomov

Structure learning and uncertainty-guided exploration in the human brain

Abstract

Over the past several decades, reinforcement learning has emerged as a unifying framework for reward-based learning and decision making in brains, minds and machines. With a long history crisscrossing the fields of psychology, neuroscience, and artificial intelligence, reinforcement learning has made major contributions to explaining many human and animal behaviors, the neural circuits underlying these behaviors, and allowing artificial agents to achieve human-level performance on tasks that were previously beyond the capabilities of computers. Characterizing the reinforcement learning computations performed by the brain can thus simultaneously advance our understanding of neurological disorders affecting decision making and also guide theoretical research towards developing artificial agents capable of dealing with complex real-world domains. Yet despite broad agreement on the prominence of reinforcement learning and its mapping onto brain circuits, many open questions remain regarding the particular kinds of representations and algorithms employed by living organisms. In this body of work, I use a combination of computational modeling, behavioral experiments, and neuroimaging to study two such questions: how the brain tackles the exploration-exploitation dilemma to efficiently learn the values of different options, given that it knows the structure of the environment, and how it might learn and represent this structure in the first place.

I first sought to characterize the neural architecture of uncertainty-guided exploration. Using fMRI, I found that the relative uncertainty of the available options is reflected in rostrolateral prefrontal cortex and drives directed exploration, while the total uncertainty of the available options is reflected in dorsolateral prefrontal cortex

and drives random exploration (Paper 1). Next, I asked which brain regions might be involved in learning the structure of the task. Using fMRI, I found that beliefs about structure are reflected in posterior parietal cortex and anterior insula, while learning signals that update those beliefs are distributed across a network of frontoparietal regions (Paper 2). Finally, I asked how the structure of the world might be represented compactly for efficient goal-directed behavior. Across a series of simulations of past and novel behavioral experiments, I found evidence for a hierarchical representation that is learned incrementally through probabilistic inference (Paper 3).

Contents

Listing of figures

Acknowledgments

First and foremost, I would like to thank my advisor Samuel Gershman for the tremendous support and generosity, and for giving me the intellectual freedom to pursue questions I'm genuinely fascinated by. I would also like to thank the members of my dissertation advisory committee, Jan Drugowitsch, Nao Uchida, and Bence Olveczky for the helpful discussions and feedback throughout my graduate years.

I'm also grateful to all members of the Cognitive Computational Neuroscience Lab, especially to Eric Schulz, Wouter Kool, Nick Franklin, and Ishita Dasgupta for their continuous feedback which significantly shaped my thinking as a scientist. I also thank Finale Doshi-Velez, George Konidaris, Erik Kastman, and Katie Insel for the helpful feedback and ideas.

I would like to extend the most special thanks to the undergraduates Van Truong, Rohan Hundia, Samyu Yagati, Angi Kumar, and Wanqian Wan who allowed me the privilege to mentor them, put faith in my ideas, and devoted many hours studiously working on them. Their inquisitiveness challenged my thinking and helped crystallize the ideas presented here. This thesis would not have been possible without them.

Last but not least, I would like to thank the members of my family for their unconditional support and encouragement throughout my graduate journey.

0

Introduction

All behaving organisms need to make decisions that maximize reward and minimize punishment. Failure to do so can prevent the organism from acquiring resources requisite for life such as food and water, or expose it to mortal threats such as predators. Often such decisions need to be made quickly under conditions of scarce and unreliable information and with little prior experience. Effective decision making in the face of uncertainty is thus a key prerequisite for the survival of humans and non-human animals, and how this process is realized in the nervous system is one of the central open questions in behavioral neuroscience.

The problem of optimal decision making has been studied by multiple disciplines in different guises. In computer science, the problem of optimizing a system's behavior was studied under the banner of optimal control, with the pioneering work of Bellman introducing the foundational theoretical formulation of the optimal control problem[20] and proposing dynamic programming as a way to solve it[19]. In psychology, the question of how animals learn to associate stimuli and actions with rewarding outcomes has been studied within the paradigms of classical (or Pavlovian) and instrumental conditioning[289,220,261,139,291], with the influential model of Rescorla and Wagner ultimately providing a theoretical framework for understanding many of the disperate empirical observations[234].

The marriage of these two parallel traditions in the 80's gave birth to the modern field of reinforcement learning as the study of reward-based learning and decision making in artificial and biological agents[279]. Since then, reinforcement learning has achieved remarkable success both as a normative theory of how agents should behave, as demonstrated by sophisticated reinforcement learning algorithms capable of outperforming engineered systems and humans alike at tasks thought to be hallmarks of human intelligence[285,200], as well as a descriptive theory of Pavlovian and instrumental conditioning, accounting for a slew of empirical phenomena that challenged previous learning theories[223,298,249,102].

The superiority of reinforcement learning as descriptive theory of animal learning and decision making was further cemented in the 90's by the hallmark discovery that the firing of midbrain dopaminergic neurons closely mirrors the properties of reward prediction errors in temporal difference reinforcement learning[251], a particular kind of reinforcement learning algorithm. Understanding the role of dopamine through the lens of reinforce-

ment learning has shed light on many puzzling neural phenomena related to this ubiquitous neuromodulator[212]. It has also provided invaluable insights into the mechanism and symptoms of some of the main neurological disorders implicating dopamine, such as schizophrenia[93], Parkinson's disease[256], attention deficit hyperactivity disorder[322], depression[36], as well as drug abuse and addiction[60]. The so-called dopamine reward prediction error hypothesis has been further validated in multiple species by numerous subsequent studies[92,133,219,277,41,87,88,54,274] that have grounded the reinforcement learning framework in the cortico-basal ganglia circuitry, providing fertile ground for the development of further links between reinforcement learning theory and brain function. The continual success of reinforcement learning in solving ever more challenging artificial intelligence problems[258] and in linking behavioral phenomena to their underlying neural substrates[283] has established reinforcement learning as a unifying framework for studying optimal decision making in both artificial and biological agents.

Despite this success, a fundamental gap remains between the simplicity of reinforcement learning algorithms currently used to model brain function and the plethora of behaviors necessary to cope with complex real-world problems like the ones faced by humans and non-human animals in everyday life. On the artificial intelligence side, such problems are made tractable largely with the use of sophisticated function approximation techniques ("deep" neural networks[120]) in conjunction with massive amounts of training and computing power[200,258]. Yet these approaches fall short of being useful models of brain function, first because they often require orders of magnitude more experience and computational resources than would be available to a biological agent, and second, because they often lack some of the key ways in which the brain is thought to represent and deal with the richness and complexity of the world[172].

In particular, research suggests that both humans and nonhuman animals learn structured representations (or internal models) of the world that can be used to simulate the outcomes of different courses of action[291,122,117]. Equipped with an appropriate model of the environment, an agent can quickly and flexibly adapt its behavior to new situations with little or no training, allowing superior performance in the face of changing task demands[176,171]. Additionally, it is been shown that the uncertainty of these representations is also computed and used in guiding action selection, allowing humans and other animals to calibrate their choices in accordance with

the normative principles of probabilistic inference[65,148,164,108]. In particular, representations of uncertainty can be useful in exploring choices that can significantly improve the agent's internal model or that hold the greatest promise of potential reward[8]. The use of such structured probabilistic representations is also thought to be crucial for improving the sample complexity of state-of-the-art artificial intelligence systems and bringing them closer to human-like learning and performance[172]. Incorporating structure and uncertainty into reinforcement learning theory and its mapping onto neural circuits is therefore essential both for improving our understanding of how the brain can flexibly cope with complex environments and for endowing artificial agents with similar capabilities.

In this work, I approach these questions by evaluating particular representations and algorithms for reinforcement learning using behavioral and neuroimaging experiments. In the first paper, I study how uncertainty is represented in the brain and how it influences exploratory behavior. While standard reinforcement learning algorithms often implement exploration by simply injecting randomness into choices, humans have been shown to explore in a more systematic ways, biasing their choices towards alternatives with greater uncertainty (directed exploration) and also scaling the randomness of their choices with the overall uncertainty in the environment (random exploration). This is consistent with normative theories which show that these strategies are superior to naive exploration strategies under certain conditions[7,288]. Using functional magnetic resonance imaging (fMRI), I show that a hybrid computational model instantiating both strategies can account for the different representations of uncertainty in different brain regions and their relation to behavior. In the second paper, I study how the brain infers the causal structure of the environment, allowing generalization of reward-based associations to novel stimuli. Causal inference is a core aspect of cognition[122] and is thought to be instrumental in allowing inductive inference in ways that are not captured by standard function approximation techniques[171,172]. Using fMRI, I identify brain areas that track signals corresponding to the process of learning the causal structure of the task, as well as regions tracking the subjective belief about the inferred causal structure. Those regions are distinct from but overlapping with the brain regions tracking reward-based associations modulated by the inferred causal structure. In the third paper, I ask how humans represent the transition structure of the world for

purposes of efficient model-based reinforcement learning. The cognitive limitations of humans stand in sharp contrast with the vast domains in which they operate, suggesting that the human brain relies on different forms of abstraction to learn a compressed representation of those domains. Such abstractions are thought to be essential in enabling artificial intelligence systems to handle large-scale problem spaces[26]. I propose a computational model for how the brain might discover such abstractions and show that it is in agreement with a number of previous findings reported in the literature, as well as with a number of new behavioral experiments in which I assess its novel predictions. Overall, this work shows that structured probabilistic representations are an integral part of reinforcement learning in the brain, paving the way for incorporating them into reinforcement learning theory and further interrogating the underlying neural implementation.

1

Dissociable Neural Correlates of Uncertainty Underlie Different Exploration Strategies

1.1 Abstract

Most real-world decisions involve a delicate balance between exploring unfamiliar alternatives and committing to the best known option. Uncertainty lies at the core of this "explore-exploit" dilemma, for if all options were perfectly known, there would be no need to explore. Yet despite the prominent role of uncertainty-guided exploration in decision making, evidence for its neural implementation is still sparse. We investigated this question with model-based fMRI (n = 31) using a two-armed bandit task that independently manipulates two forms of uncertainty underlying different exploration strategies. The relative uncertainty between the two options was correlated with BOLD activity in right rostrolateral prefrontal cortex and drove directed exploration, a strategy that adds an uncertainty bonus to each option. The total uncertainty across the two options was correlated with activity in right dorsolateral prefrontal cortex and drove random exploration, a strategy that increases choice stochasticity in proportion to total uncertainty. The subjective estimates of uncertainty from both regions were predictive of subject choices. Finally, the decision value signal combining relative and total uncertainty to compute choice was reflected in motor cortex activity. The variance of this decision value signal scaled with total uncertainty, consistent with a sampling mechanism for random exploration. Overall, these results are consistent with a hybrid computational architecture in which different uncertainty computations are performed separately and then combined by downstream decision circuits to compute choice.

1.2 Introduction

For every decision that we make, we have to choose between the best option that we know so far (exploitation), or a less familiar option that could be even better (exploration). This "explore-exploit" dilemma appears at all levels of decision making, ranging from the mundane (Do I go to my favorite restaurant, or try a new one?) to the momentous (Do I marry my current partner, or try a new one?). Despite its ubiquity in everyday life, little is known about how the brain handles the exploration-exploitation trade-off. Intuitively, either extreme is undesirable: an agent that solely exploits will never adapt to changes in the environment (or even learn which options are good

in the first place), while an agent that solely explores will never reap the fruits of that exploration. Yet striking the perfect balance is computationally intractable beyond the simplest examples, and hence humans and animals must adopt various heuristics [40,195,313].

Earlier research suggested that people choose options in proportion to their expected values [57,317], a strategy known as softmax exploration that is closely related to other psychological phenomena such as probability matching (for a detailed review, see Schulz & Gershman [253]). Later studies showed that people explore in a more sophisticated manner, using uncertainty to guide their choices towards more promising options [311,105]. These strategies are more adaptive in nonstationary environments and come in two distinct flavors: *directed* and *random* exploration strategies.

Directed exploration strategies *direct* the agent's choices towards uncertain options, which is equivalent to adding an uncertainty bonus to their subjective values. For example, for your next meal, you might forego your favorite restaurant for a new one that just opened down the street, and you might even go there several times until you are certain it is no better than your favorite. Thus while softmax exploration is sensitive to the *relative value* of each option, preferring options with higher payoffs, directed exploration is additionally sensitive to the *relative uncertainty* of each option, preferring options with more uncertainty as they hold greater potential for gain. Directed exploration is closely related to the phenomenon of risk-seeking [136,306] and has strong empirical support [96,267,70].

Previous work [105,106,113] has shown that directed exploration in humans is well captured by the upper confidence bound (UCB) algorithm [7], in which the uncertainty bonus is the one-sided confidence interval of the expected value:

$$a_t = \arg\max_k [Q_t(k) + U_t(k)], \tag{1.1}$$

where a_t is the action chosen at time t, $Q_t(k)$ is the expected reward of action k at time t, and $U_t(k)$ is the up-

per confidence bound of the reward that plays the role of an uncertainty bonus. In a Bayesian variant of UCB[268], $Q_t(k)$ corresponds to the posterior mean and $U_t(k)$ is proportional to the posterior standard deviation $\sigma_t(k)$. Returning to the restaurant example, even if both restaurants have the same expected value ($Q_t(\text{new}) = Q_t(\text{old})$), UCB would initially prefer the new one since it has greater uncertainty ($U_t(\text{new}) > U_t(\text{old})$).

Random exploration strategies introduce randomness into choice behavior, causing the agent to sometimes explore less favorable options. For example, when you move to a new neighborhood, you might initially pick restaurants at random until you learn which ones are good. While earlier studies favored value-based random exploration strategies such as softmax exploration, later work[105,106] has shown that random exploration in people is additionally sensitive to the *total uncertainty* of the available options, increasing choice stochasticity when option values are more uncertain. This can cause choice variability to track payoff variability, a phenomenon sometimes referred to as the payoff variability effect[206,31,86].

One prominent instantiation of random exploration in reinforcement learning is Thompson sampling[288], which samples values randomly from the posterior value distribution of each action and then chooses greedily with respect to the sampled values:

$$\widetilde{Q}_t(k) \sim p(Q_t(k)) \tag{1.2}$$

$$a_t = \arg\max_k \widetilde{Q}_t(k), \tag{1.3}$$

where $p(.)$ is the posterior value distribution and $\widetilde{Q}_t(k)$ is the sampled value for arm k at time t. Returning to the neighborhood example, the familiar restaurants in your old neighborhood have narrow value distributions around their expected values (low total uncertainty). This will cause Thompson sampling to consistently draw samples $\widetilde{Q}_t(k)$ that are close to their expected values, which will often result in choosing the same restaurant, namely the one with the highest expected value. In contrast, the unfamiliar restaurants in the new neighborhood have wide value distributions (high total uncertainty), which will result in significant variation in the Thompson

samples $\widetilde{Q}_t(k)$ and a corresponding variation in the chosen restaurant.

Directed and random exploration strategies confer separate ecological advantages, which has led researchers in reinforcement learning to develop algorithms that use a hybrid of UCB and Thompson sampling [34,189]. Correspondingly, recent evidence suggests that people also employ a combination of directed and random exploration [311,105]. A study by Gershman [106] used a two-armed bandit task to show that human choices are consistent with a particular hybrid of UCB and Thompson sampling. Furthermore, the study showed that different uncertainty computations underlie each exploration strategy, with the relative uncertainty between the two options driving directed exploration, and the total uncertainty of the two options driving random exploration. This led us to hypothesize the existence of dissociable neural implementations of both strategies in the brain. At least three lines of evidence support this claim. First, dopamine genes with anatomically distinct expression profiles are differentially associated with directed and random exploration [113]. Second, transcranial magnetic stimulation of right rostrolateral prefrontal cortex (RLPFC) affects directed, but not random, exploration [321]. Third, directed and random exploration have different developmental trajectories [265].

In the present study, we used functional MRI to probe the neural underpinnings of the uncertainty computations that influence directed and random exploration. Subjects performed a two-armed bandit task in which each arm was either "safe", meaning it delivered the same reward during the whole block, or "risky", meaning it delivered variable rewards. This allowed us to separate the effects of relative and total uncertainty and examine how their neural correlates influence directed and random exploration, in accordance with the theoretical principles outlined above. We found that relative uncertainty is reflected in right RLPFC, and total uncertainty is reflected in right dorsolateral prefrontal cortex (DLPFC), replicating findings reported by Badre et al. [11]. The neural signal in right RLPFC predicted cross-trial variability in directed but not random exploration, whereas the neural signal in right DLPFC predicted cross-trial variability in random but not directed exploration. We also found that the linear combination of relative and total uncertainty with value is reflected in motor cortex, suggesting that these upstream estimates are integrated by motor circuits in order to compute the categorical decision. By linking activity in those regions with human choices via a hybrid UCB/Thompson sampling model, our work

provides new insight into the distinct uncertainty computations performed by the brain and their role in guiding behavior.

1.3 Relative and total uncertainty guide directed and random exploration

We scanned 31 human subjects (17 female, ages 18-35) using functional MRI while they performed a two-armed bandit task in which subjects are informed about the riskiness of each option [106]. On each trial, subjects saw the labels of the two options, each of which could be either "safe" (S) or "risky" (R; Figure 1.1A). A safe option always delivered the same reward for the duration of the block, while a risky option delivered Gaussian-distributed rewards around a mean that remained fixed for the duration of the block (Figure 1.1B). We denote the trial types by the pair of option labels (e.g., on "RS" trials, option 1 is risky and option 2 is safe). Trial types and average rewards remained fixed within blocks and varied randomly across blocks. Subjects were explicitly informed of the statistics of the task and performed four practice blocks before entering the scanner.

Importantly, this task design allowed us to independently measure the effect of different types of uncertainty on subject choices. Directed and random exploration predict different effects across different block conditions, which can be illustrated by considering the probability of choosing option 1, P(choose 1), as a function of the expected value difference for the given block, $\mu(1) - \mu(2)$ (the choice function; Figure 1.1C and D).

RS and SR trials manipulate relative uncertainty (greater for the option 1 on RS trials and greater for option 2 on SR trials) while controlling for total uncertainty (identical across RS and SR trials). A strategy that is insensitive to relative uncertainty such as softmax exploration would be indifferent between the two options (P(choose 1) = 0.5) when they have equal values ($\mu(1) - \mu(2) = 0$). In contrast, UCB predicts a bias towards the risky option, preferring option 1 on RS trials and option 2 on SR trials, even when the expected value difference might dictate otherwise. This would manifest as an opposite intercept shift in the choice probability function of RS and SR trials, such that P(choose 1) > 0.5 when $\mu(1) - \mu(2) = 0$ on RS trials and P(choose 1) < 0.5 when $\mu(1) - \mu(2) = 0$ on SR trials (Figure 1.1C).

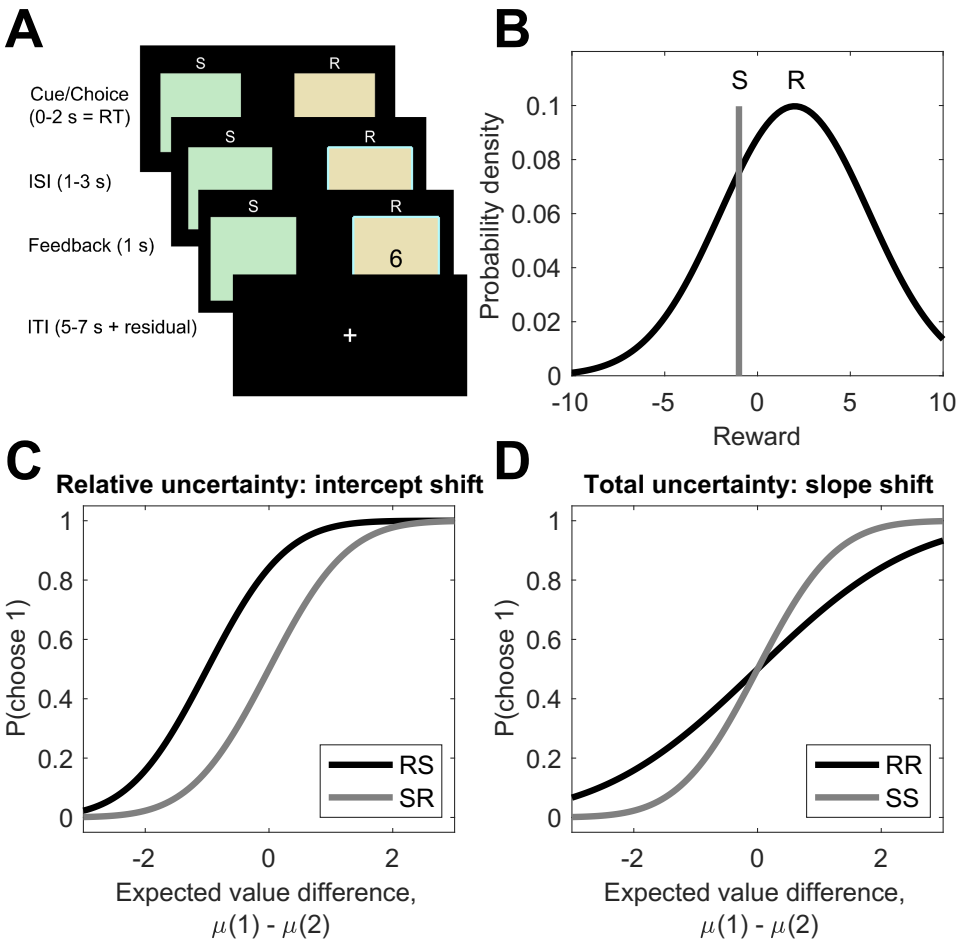


Figure 1.1: Experimental Design and Predictions.
(A) Trial structure. Subjects choose between two options, each labeled as either safe (S) or risky (R). After they make a choice, they receive feedback in the form of points. Option labels remain constant during the block.
(B) Reward structure. Risky options deliver rewards drawn from a Gaussian distribution whose mean remains constant during the block. Safe options deliver the same reward during the block. The means of both options are resampled from the zero-mean Gaussian at the start of each block.
(C) Directed exploration (UCB) predicts a bias towards the uncertain option, which shifts the choice probability function in the opposite directions for RS and SR trials.
(D) Random exploration (Thompson sampling) predicts more randomness when uncertainty is high, which reduces the slope of the choice probability function for RR compared to SS trials.

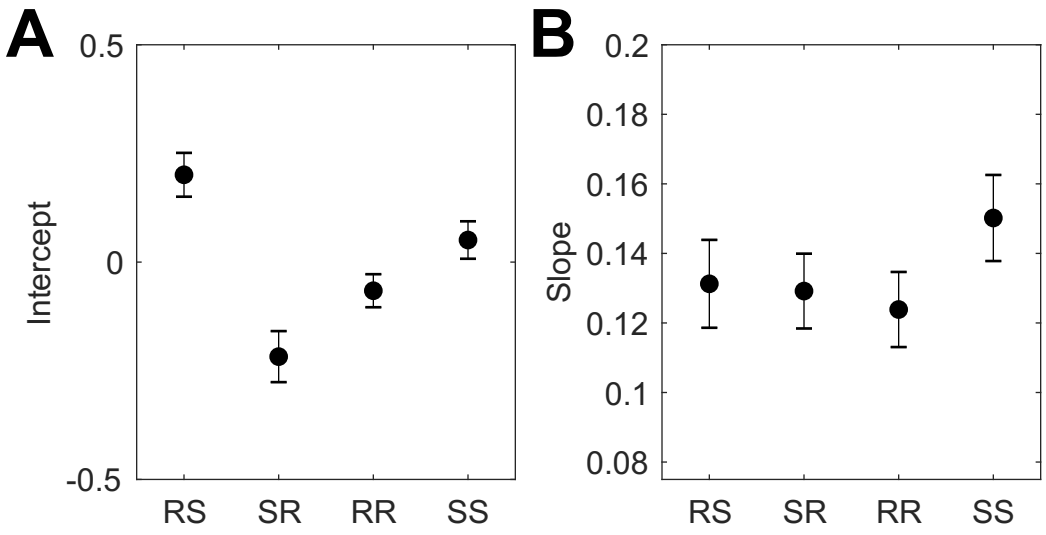


Figure 1.2: Probit regression results.
(A) Intercept and (B) slope of choice probability function fit for each condition using maximum likelihood estimation.

In contrast, RR and SS trials manipulate total uncertainty (high on RR trials and low on SS trials) while controlling for relative uncertainty (identical across RR and SS trials). A strategy that is insensitive to total uncertainty such as UCB would predict the same choice function for both trial types. In contrast, Thompson sampling predicts more stochastic choices when there is more uncertainty, resulting in more random choices (P(choose 1) closer to 0.5) even when the relative expected value strongly favors one option ($\mu(1) - \mu(2)$ far from 0). This would manifest as a shallower slope of the choice probability function of RR compared to SS trials (Figure 1.1D).

Overall, subjects identified the better option in each block (i.e., the option k with the greater expected reward $\mu(k)$) relatively quickly, with average performance plateauing by the middle of each block (Figure A.1). Importantly, in accordance with the theory, we found that manipulating relative uncertainty (RS vs. SR) shifted the intercept of the choice probability function (Figure 1.2A, Figure A.2): the intercept for RS trials was significantly greater than the intercept for SR trials ($F(1, 9711) = 21.0, p = 0.000005$). Moreover, the intercept for RS trials was significantly greater than 0 ($F(1, 9711) = 10.8, p = 0.001$), while the intercept for SR trials was significantly less than 0 ($F(1, 9711) = 17.9, p = 0.00002$). There was only a small effect of total uncertainty on the inter-

cept (RR vs. SS, $F(1, 9711) = 4.1, p = 0.04$). This indicates that, regardless of relative expected value, subjects showed a bias towards the risky option, consistent with UCB.

Conversely, manipulating total uncertainty (RR vs. SS) altered the slope of the choice probability function (Figure 1.2B, Figure A.2): the slope for RR trials is smaller than the slope for SS trials ($F(1, 9711) = 3.4, p = 0.07$). While the effect is small due to the small sample size, it is in the right direction and is consistent with previous replications of this experiment [106,113]. There was no effect of relative uncertainty (RS vs. SR) on the slope ($F(1, 9711) = 0.06, p = 0.8$). This indicates that when both options were risky, subjects were less sensitive to their relative reward advantage, consistent with Thompson sampling.

To examine how relative and total uncertainty influence directed and random exploration on a trial-by-trial basis, we modeled subject choices using a probit regression model [106]:

$$P(a_t = 1|\mathbf{w}) = \Phi(w_1 \mathrm{V}_t + w_2 \mathrm{RU}_t + w_3 \mathrm{V}_t/\mathrm{TU}_t), \tag{1.4}$$

where $\Phi(\cdot)$ is the standard Gaussian cumulative distribution function and the regressors are the following model-derived trial-by-trial posterior estimates:

- Value difference, $\mathrm{V}_t = Q_t(1) - Q_t(2)$.
- Relative uncertainty, $\mathrm{RU}_t = \sigma_t(1) - \sigma_t(2)$.
- Total uncertainty, $\mathrm{TU}_t = \sqrt{\sigma_t^2(1) + \sigma_t^2(2)}$.

Here $Q_t(k)$ corresponds to the posterior expected value of option k (Eq. 1.6) and $\sigma_t(k)$ is the posterior standard deviation around that expectation (Eq. 1.7), proportional to the uncertainty bonus in UCB. Note that these are trial-by-trial estimates based on the posterior quantities computed by the ideal observer model.

Gershman [105] showed that, despite its apparent simplicity, this is not a reduced form model but rather the exact analytical form of the most parsimonious hybrid of UCB and Thompson sampling that reduces to pure UCB

when $w_3 = 0$, to pure Thompson sampling when $w_2 = 0$, and to pure softmax exploration when $w_2 = w_3 = 0$. Thus the hybrid model balances exploitation (governed by w_1) with directed (w_2) and random (w_3) exploration simultaneously for each choice, without the need to dynamically select one strategy over the other (whether and how the brain might perform this meta-decision is beyond the scope of our present work). If subjects use both UCB and Thompson sampling, the model predicts that all three regressors will have a significant effect on choices ($w_1 > 0, w_2 > 0, w_3 > 0$).

Correspondingly, the maximum likelihood estimates of all three fixed effects coefficients were significantly greater than zero: $w_1 = 0.166 \pm 0.016$ ($t(9716) = 10.34, p = p < 10^{-20}$; mean $\pm$ s.e.m., two-tailed t-test), $w_2 = 0.175 \pm 0.021$ ($t(9716) = 8.17, p = p < 10^{-15}$), and $w_3 = 0.005 \pm 0.001$ ($t(9716) = 4.47, p = p < 10^{-5}$). Model comparisons revealed that the UCB/Thompson hybrid model fits subject choices better than UCB or Thompson sampling alone, which in turn fit choices better than softmax alone (Table A.1). Bayesian model comparison strongly favored the hybrid model over alternative models protected exceedance probability = 1; [238].

Furthermore, running these models generatively with the corresponding fitted parameters on the same bandits as the subjects revealed significant differences in model performance (Figure A.3, $F(3, 1236) = 291.58, p < 10^{-20}$, one-way ANOVA). The UCB/Thompson hybrid outperformed UCB and Thompson sampling alone (UCB vs. hybrid, $p < 10^{-8}$; Thompson vs. hybrid, $p < 10^{-8}$, pairwise multiple comparison tests), which in turn outperformed softmax exploration (softmax vs. UCB, $p < 10^{-5}$; softmax vs. Thompson, $p < 10^{-8}$). Similar results replicated across a range of coefficients (Figure A.4), signifying the distinct and complementary ecological advantages of UCB and Thompson sampling. Thus relying on both UCB ($w_2 > 0$) and Thompson sampling ($w_3 > 0$) should yield better overall performance. In line with this prediction, we found better performance among subjects whose choices are more sensitive to RU_t (greater w_2), consistent with greater reliance on UCB (Figure A.5B, $r(29) = 0.47, p = 0.008$, Pearson correlation). Similarly, we found better performance among subjects whose choices are more sensitive to $\mathrm{V}_t/\mathrm{TU}_t$ (greater w_3), consistent with greater reliance on Thompson sampling (Figure A.5C, $r(29) = 0.53, p = 0.002$). Finaly, note that even though optimal exploration is intractable in general, the hybrid model computes choices in constant time by simply computing Eq. 1.4.

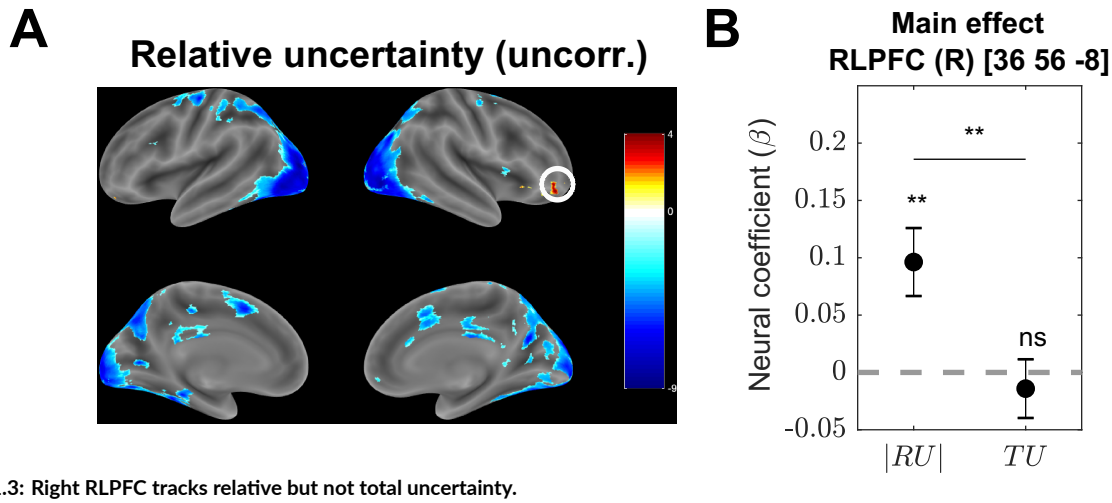


Figure 1.3: Right RLPFC tracks relative but not total uncertainty.
(A) Whole-brain $|\mathrm{RU}_t|$ contrast from GLM 1. Single voxels were thresholded at $p < 0.001$. Multiple comparisons correction was not applied (corrected version is shown in Figure A.7A). The color scale represents t-values across subjects. The circled ROI in right RLPFC (MNI [36 56 -8]) from Badre et al. [11] was used in the subsequent confirmatory analysis (10 mm sphere around the peak voxel).
(B) Neural regression coefficients (betas) from GLM 1 for the parametric modulators $|\mathrm{RU}_t|$ ($\beta_{|\mathrm{RU}|}$) and TU_t (β_{TU}) at trial onset, averaged across voxels in the ROI. Error bars are cross-subject standard errors. ** = p < 0.01, ns = not significant.

Taken together, these results replicate and expand upon previous findings [106], highlighting the superiority of the UCB/Thompson hybrid as a descriptive as well as normative model of uncertainty-guided exploration. Thus humans do and ought to employ both directed and random exploration, driven by relative and total uncertainty, respectively.

1.4 Neural correlates of relative and total uncertainty

Next, we asked whether relative and total uncertainty are represented in distinct anatomical loci. We performed an unbiased whole-brain univariate analysis using a general linear model (GLM 1) with model-derived trial-by-trial posterior estimates of the quantities used in computing the decision (Eq 1.4): absolute relative uncertainty ($|\mathrm{RU}_t|$), total uncertainty (TU_t), absolute value difference ($|\mathrm{V}_t|$), and absolute value difference scaled by total uncertainty ($|\mathrm{V}_t|/\mathrm{TU}_t$) as non-orthogonalized impulse regressors at trial onset (see Methods). We report whole-brain t-maps after thresholding single voxels at $p < 0.001$ (uncorrected) and applying cluster family wise error (FWE) correction at significance level $\alpha = 0.05$.

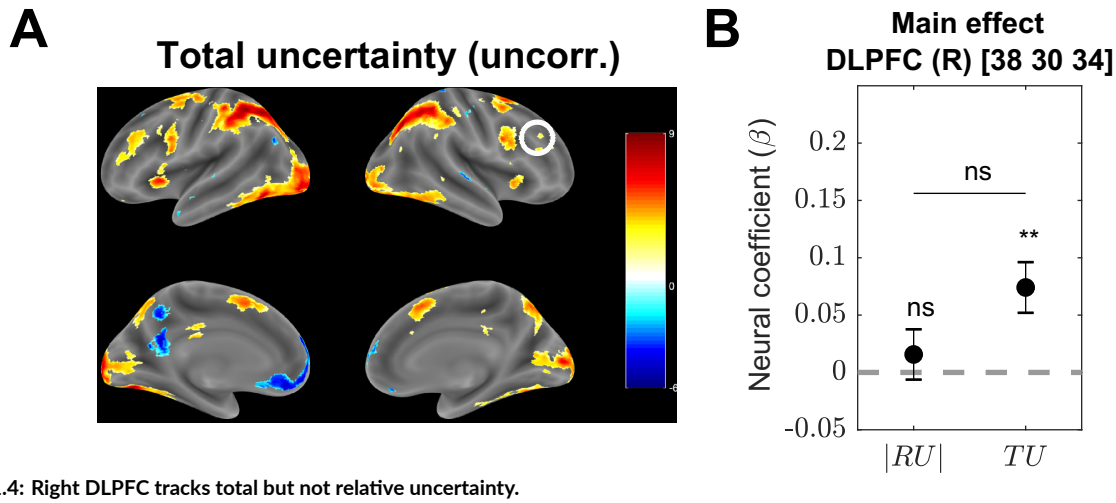


Figure 1.4: Right DLPFC tracks total but not relative uncertainty.
(A) Whole-brain TU_t contrast from GLM 1. Single voxels were thresholded at $p < 0.001$. Multiple comparisons correction was not applied (corrected version is shown in Figure A.7B). The color scale represents *t*-values across subjects. The circled ROI in right DLPFC (MNI [38 30 34]) from Badre et al. [11] was used in the subsequent confirmatory analysis (10 mm sphere around the peak voxel).
(B) Neural regression coefficients (betas) from GLM 1 for the parametric modulators $|RU_t|$ ($\beta_{|RU|}$) and TU_t (β_{TU}) at trial onset. Error bars are cross-subject standard errors. ** = p < 0.01, ns = not significant.

For relative uncertainty, we found a large negative bilateral occipital cluster that extended dorsally into inferior parietal cortex and primary motor cortex, and ventrally into inferior temporal cortex (Figure A.7A, Table A.3). For total uncertainty, we found bilateral positive clusters in the inferior parietal lobe, DLPFC, anterior insula, inferior temporal lobe, midcingulate cortex, superior posterior thalamus, and premotor cortex (Figure A.7B, Table A.4). We did not find any clusters for $|V_t|$ or $|V_t|/TU_t$.

Based on previous studies [11,321], we expected to find a positive cluster for relative uncertainty in right RLPFC. While we did observe such a cluster in the uncorrected contrast (Figure 1.3A), it did not survive FWE correction. We pursued this hypothesis further using *a priori* ROIs from Badre et al. [11], who reported a positive effect of relative uncertainty in right RLPFC (MNI [36 56 -8]) and of total uncertainty in right DLPFC (MNI [38 30 34]). In accordance with their results, in right RLPFC we found a significant effect of relative uncertainty (Figure 1.3B; $t(30) = 3.24, p = 0.003$, two-tailed t-test) but not of total uncertainty ($t(30) = -0.55, p = 0.58$). The significance of this difference was confirmed by the contrast between the two regressors ($t(30) = 2.96, p = 0.006$, paired t-test). Conversely, in right DLPFC there was a significant effect of total uncertainty (Figure 1.4B;

Table 1.1: Model comparison with neurally-decoded regressors. Model fits after augmenting the UCB/Thompson hybrid (Eq. 1.4) with estimates of relative uncertainty, total uncertainty, and decision value, decoded from brain activity. Lower BIC indicates better fit. BIC = Bayesian information criterion.

Model	Regressors	BIC
Baseline		
UCB/Thompson hybrid with intercept (Eq. 1.4)	1 + V + RU + V/TU	6410.00
$\widehat{\text{RU}}$ and $\widehat{\text{TU}}$ from right RLPFC		
Baseline augmented with $\widehat{\text{RU}}$ (Eq. 1.12)	1 + V + RU + V/TU + $\widehat{\text{RU}}$	6406.54
Baseline augmented with $\widehat{\text{TU}}$ (Eq. 1.13)	1 + V + RU + V/TU + V/$\widehat{\text{TU}}$	6421.57
$\widehat{\text{RU}}$ and $\widehat{\text{TU}}$ from right DLPFC		
Baseline augmented with $\widehat{\text{RU}}$ (Eq. 1.12)	1 + V + RU + V/TU + $\widehat{\text{RU}}$	6418.85
Baseline augmented with $\widehat{\text{TU}}$ (Eq. 1.13)	1 + V + RU + V/TU + V/$\widehat{\text{TU}}$	6358.53
$\widehat{\text{RU}}$ from right RLPFC and $\widehat{\text{TU}}$ from right DLPFC		
Baseline augmented with $\widehat{\text{RU}}$ and $\widehat{\text{TU}}$ (Eq. 1.14)	1 + V + RU + V/TU + $\widehat{\text{RU}}$ + V/$\widehat{\text{TU}}$	6359.03
$\widehat{\text{DV}}$ from left M1		
Baseline augmented with $\widehat{\text{DV}}$ (Eq. 1.15)	1 + V + RU + V/TU + $\widehat{\text{DV}}$	6407.84

$t(30) = 3.36, p = 0.002$) but not of relative uncertainty ($t(30) = 0.71, p = 0.48$), although the contrast between the two did not reach significance ($t(30) = 1.74, p = 0.09$). These results replicate Badre et al. [11]'s findings and suggest that relative and total uncertainty are represented in right RLPFC and right DLPFC, respectively.

1.5 Subjective estimates of relative and total uncertainty predict choices

If right RLPFC and right DLPFC encode relative and total uncertainty, respectively, then we should be able to use their activations to decode trial-by-trial subjective estimates of RU_t and TU_t. In particular, on any given trial, a subject's estimate of relative and total uncertainty might differ from the ideal observer estimates RU_t and TU_t stipulated by the hybrid model (Eq. 1.4). This could occur for a number of reasons, such as neural noise, inattention, or a suboptimal learning rate. Importantly, any deviation from the ideal observer estimates would result in a corresponding deviation of the subject's choices from the model predictions. Therefore if we augment the hybrid

model to include the neurally-decoded subjective estimates of relative and total uncertainty (denoted by $\widehat{RU_t}$ and $\widehat{TU_t}$, respectively), then we should arrive at more accurate predictions of subject choices (see Methods).

Indeed, this is what we found. Including the decoded trial-by-trial $\widehat{RU_t}$ (Eq. 1.12) from right RLPFC (MNI [36 56 -8]) significantly improved predictions of subject choices (Table 1.1; BICs: 6407 vs. 6410). Importantly, decoding trial-by-trial $\widehat{TU_t}$ from right RLPFC and augmenting the model with $V_t/\widehat{TU_t}$ (Eq. 1.13) did not improve choice predictions (BICs: 6421 vs. 6410).

Similarly, augmenting the hybrid model with $V_t/\widehat{TU_t}$ (Eq. 1.13) when $\widehat{TU_t}$ was decoded from right DLPFC (MNI [38 30 34]) significantly improved predictions of subject choices (Table 1.1; BICs: 6359 vs. 6410). Conversely, augmenting the model with $\widehat{RU_t}$ (Eq. 1.12) decoded from right DLPFC did not improve choice predictions (BICs: 6419 vs. 6410). Together, these results show that variability in the neural representations of uncertainty in the corresponding regions predicts choices, consistent with the idea that those representations are used in a downstream decision computation.

We additionally augmented the model with both $\widehat{RU_t}$ from right RLPFC and $V_t/\widehat{TU_t}$, with $\widehat{TU_t}$ from right DLPFC (Eq. 1.14, Table 1.1). This improved choice predictions beyond the improvement of including $\widehat{RU_t}$ alone (BICs: 6359 vs. 6406). It also resulted in better choice fits than including $V_t/\widehat{TU_t}$ alone (Eq. 1.13), which is reflected in the lower AIC (6273 vs. 6287) and deviance (6249 vs. 6266), even though the more stringent BIC criterion is comparable (6359 vs. 6359). This suggests that the two uncertainty computations provide complementary yet not entirely independent contributions to choices.

1.6 Neural correlates of downstream decision value computation

We next sought to identify the downstream decision circuits that combine the relative and total uncertainty estimates to compute choice. Following the rationale of the UCB/Thompson hybrid model, we assume that the most parsimonious way to compute decisions is to linearly combine the uncertainty estimates with the value estimate, as in Eq. 1.4. We therefore employed a similar GLM to GLM 1 (GLM 2, Table A.2) with model-derived

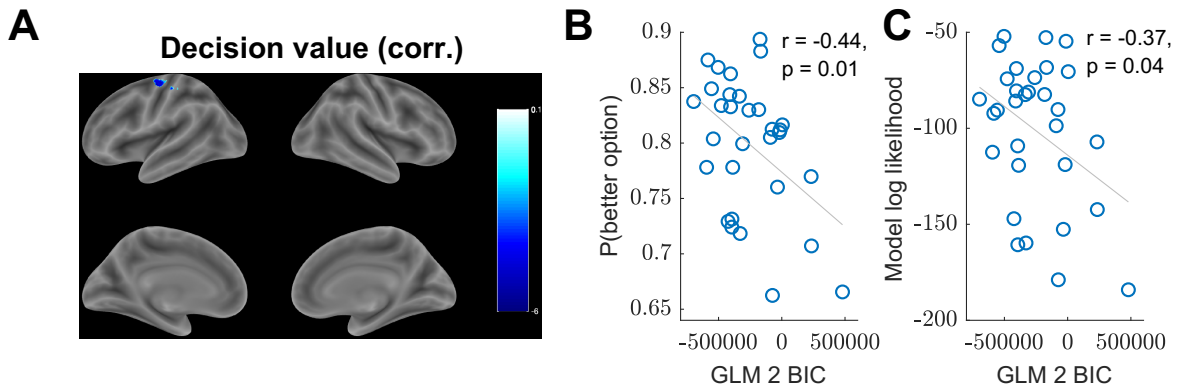


Figure 1.5: Primary motor cortex tracks decision value.
(A) Whole-brain $|\mathrm{DV}_t|$ contrast from GLM 2. Single voxels were thresholded at $p < 0.001$ and cluster FWE correction was applied at significance level $\alpha = 0.05$. The ROI in left primary motor cortex (left M1; peak MNI [-38 -8 62]) was used in the subsequent confirmatory analysis (10 mm sphere around the peak voxel).
(B) Cross-subject correlation between the BIC in left M1, quantifying the extent to which neural activity in that region is captured by GLM 2 (lower BIC indicates better fit), and average performance.
(C) Cross-subject correlation between the BIC in left M1 and model log likelihood, quantifying the extent to which the subject's choices are consistent with the UCB/Thompson hybrid model.

trial-by-trial estimates of the decision value (DV) as the only parametric modulator at trial onset. We quantified decision value as the linear combination of the terms in Eq. 1.4, weighted by the corresponding subject-specific random effects coefficients $\mathbf{w}$ from the probit regression:

$$\mathrm{DV}_t = w_1 \mathrm{V}_t + w_2 \mathrm{RU}_t + w_3 \mathrm{V}_t / \mathrm{TU}_t. \tag{1.5}$$

As before, we took the absolute decision value $|\mathrm{DV}_t|$ for purposes of identifiability. As previously, we thresholded single voxels at $p < 0.001$ (uncorrected) and applied cluster FWE correction at significance level $\alpha = 0.05$. This revealed a single negative cluster in left primary motor cortex (peak MNI [-38 -8 62], Figure 1.5A, Table A.5).

We defined an ROI as a 10 mm sphere around the peak voxel, which we refer to as left M1 in subsequent confirmatory analyses. Note that this activation is not simply reflecting motor responses, since those are captured by a chosen action regressor (Table A.2). Another possible confound is reaction time (RT). When controlling for

RT, motor cortex activations become unrelated to $|DV_t|$ (GLM 2A in Supplemental Information). This could be explained by a sequential sampling implementation of our model (see Discussion), according to which RT would depend strongly on DV. Consistent with this interpretation, model comparison revealed that left M1 activity is best explained by a combination of DV and RT, rather than DV or RT alone (see Supplemental Information).

1.7 Subjective estimate of decision value predicts within-subject and cross-subject choice variability

If left M1 encodes the decision value, then we should be able to use its activation to decode trial-by-trial subjective estimates of DV_t, similarly to how we were able to extract subjective estimates of RU_t and TU_t. In particular, on any given trial, a subject's estimate of the decision value might differ from the linear combination of the ideal observer estimates (Eq. 1.5). Importantly, any such deviations would result in corresponding deviations from the model-predicted choices. Following the same logic as before, we augmented the hybrid model to include a linearly decoded trial-by-trial estimate of the decision value (denoted by $\widehat{DV}$) from left M1 (Eq. 1.15). This improved predictions of subject choices (Table 1.1, BICs: 6408 vs. 6410), consistent with the idea that this region computes the linear combination of relative and total uncertainty with value, which in turn is used to compute choice.

In order to further validate the ROI, we performed a cross-subject correlation between the extent to which GLM 2 captures neural activity in left M1 (quantified by the BIC; see Methods) and subject performance (quantified by the proportion of trials on which the subject chose the better option). We reasoned that some subjects will perform computations that are more similar to our model than other subjects. If our model is a plausible approximation of the underlying computations, then it should better capture neural activity in the decision value ROI for those subjects, resulting in lower BICs. Furthermore, those subjects should also exhibit better performance, in line with the normative principles of our model (Figures A.3, A.4, and A.5; Table A.1). This prediction was substantiated: we found a significant negative correlation between BIC and performance (Fig-

ure 1.5B, $r(29) = -0.44, p = 0.01$, Pearson correlation), indicating that subjects whose brain activity matches the model also tend to perform better. We found a similar correlation between BIC and model log likelihood (Figure 1.5C, $r(29) = -0.37, p = 0.04$), which quantifies how well the subject's behavior is captured by the UCB/Thompson hybrid model (Eq. 1.4). Together, these results build upon our previous findings and suggest that left M1 combines the subjective estimate of relative and total uncertainty from right RLPFC and right DLPFC, respectively, with the subjective value estimate, in order to compute choice.

1.8 Variability in the decision value signal scales with total uncertainty

The lack of any main effect for $|V_t|/TU_t$ in GLM 1 could be explained by a mechanistic account according to which, instead of directly implementing our closed-form probit model (Eq. 1.4), the brain is drawing and comparing samples from the posterior value distributions. This corresponds exactly to Thompson sampling and would produce the exact same behavior as the analytical model. However, it makes different neural predictions, namely that: 1) there would be no explicit coding of V_t/TU_t, and 2) the variance of the decision value would scale with (squared) total uncertainty. The latter is true because the variance of the Thompson sample for arm k on trial t is $\sigma_t^2(k)$, and hence the variance of the sample difference is $\sigma_t^2(1) + \sigma_t^2(2) = TU_t^2$. Thus while we cannot infer the drawn samples on any particular trial, we can check whether the unexplained variance around the mean decision value signal in left M1 is correlated with TU_t^2.

To test this hypothesis, we correlated the residual variance of the GLM 2 fits in the decision value ROI (Figure 1.5A; left M1, peak MNI [-38 -8 62]) with TU_t^2. We found a positive correlation ($t(30) = 2.06, p = 0.05$, two-tailed t-test across subjects of the within-subject Fisher z-transformed Pearson correlation coefficients), consistent with the idea that total uncertainty affects choices via a sampling mechanism that is implemented in motor cortex.

1.9 Discussion

Balancing exploration and exploitation lies at the heart of decision making, and understanding the neural circuitry that underlies different forms of exploration is central to understanding how the brain makes choices in the real world. Here we show that human choices are consistent with a particular hybrid of directed and random exploration strategies, driven respectively by the relative and total uncertainty of the options. This dissociation between the two uncertainty computations predicted by the model was reified as an anatomical dissociation between their neural correlates. Our GLM results confirm the previously identified role of right RLPFC and right DLPFC in encoding relative and total uncertainty, respectively [11]. Crucially, our work further elaborates the functional role of those regions by providing a normative account of how both uncertainty estimates are used by the brain to make choices, with relative uncertainty driving direct exploration and total uncertainty driving random exploration. This account was validated by our decoding analysis and decision value GLM, which suggest that the two uncertainty estimates are combined with the value estimate in downstream motor cortex, which ultimately performs the categorical decision computation.

While our study replicates the results reported by Badre et al. [11], it goes beyond their work in several important ways. First, our task design explicitly manipulates uncertainty – the main quantity of interest – across the different task conditions, whereas the task design in Badre et al. [11] is focused on manipulating expected value. Second, relative and total uncertainty are manipulated independently in our task design: relative uncertainty differs across RS and SR trials, while total uncertainty remains fixed, on average; the converse holds for SS and RR trials. Orthogonalizing relative and total uncertainty in this way allows us to directly assess their differential contribution to choices (Figure A.2). Third, the exploration strategies employed by our model are rooted in normative principles developed in the machine learning literature [7,288], with theoretical performance guarantees which were confirmed by our simulations (Figure A.3, A.4). In particular, the separate contributions of relative and total uncertainty to choices are derived directly from UCB and Thompson sampling, implementing directed and random exploration, respectively. Fourth, this allows us to link relative and total uncertainty and their neural correlates

directly to subject behavior and interpret the results in light of the corresponding exploration strategies.

Previous studies have also found a signature of exploration in RLPFC, also referred to as frontopolar cortex [57,22,11,18], however, with the exception of Badre et al. [11], these studies did not disentangle different exploration strategies or examine their relation to uncertainty. More pertinent to our study, Zajkowski et al. [321] reported that inhibiting right RLPFC reduces directed but not random exploration. This is consistent with our finding that activity in right RLPFC tracks the subjective estimate of relative uncertainty which underlies the directed exploration component of choices in our model. Disrupting activity in right RLPFC can thus be understood as introducing noise into or reducing the subjective estimate of relative uncertainty, resulting in choices that are less consistent with directed exploration.

One important novel contribution of our work is to elucidate the role of the total uncertainty signal from right DLPFC in decision making. Previous studies have shown that DLPFC is sensitive to uncertainty [138] and that perturbing DLPFC can affect decision-making under uncertainty [152,290,89]. Most closely related to our study, Knoch et al. [152] showed that suppressing right DLPFC (but not left DLPFC) with repetitive transcranial magnetic stimulation leads to risk-seeking behavior, resulting in more choices of the suboptimal "risky" option over the better "safe" option. Conversely, Fecteau et al. [89] showed that stimulating right DLPFC with transcranial direct current stimulation reduces risk-seeking behavior. One way to interpret these findings in light of our framework is that, similarly to the Zajkowski et al. [321] study, suppressing right DLPFC reduces random exploration, which diminishes sensitivity to the value difference (Eq. 1.4, third term) while allowing directed exploration to dominate choices (Eq. 1.4, second term), leading to apparently risk-seeking behavior. Stimulating right DLPFC would then have the opposite effect, increasing random exploration and thereby increasing sensitivity to the value difference between the two options, leading to an increased preference for the safe option.

Our definition of uncertainty as the posterior standard deviation of the mean (sometimes referred to as estimation uncertainty, parameter uncertainty, or ambiguity) is different from the expected uncertainty due to the unpredictability of the reward from the risky option on any particular trial sometimes referred to as irreducible uncertainty or risk; [221]. These two forms of uncertainty are generally related, and in particular in our study, es-

timation uncertainty is higher, on average, for risky arms due to the variability of their rewards, which makes it difficult to estimate the mean exactly. However, they are traditionally associated with opposite behaviors: risk-aversion predicts that people would prefer the safe over the risky option [147], while uncertainty-guided exploration predicts a preference for the risky option, all else equal (Figure 1.1C). While we did not seek to explicitly disentangle risk from estimation uncertainty in our study, our behavioral (Figure 1.2A and Figure A.2) and neural (Figure 1.3) results are consistent with the latter interpretation.

Our finding that decision value is reflected in motor cortex might seem somewhat at odds with previous neuroimaging studies of value coding, which is often localized to ventromedial prefrontal cortex vmPFC; [177], orbitofrontal cortex [303], or the intraparietal sulcus [111]. However, most of these studies consider the values of the available options (Q_t) or the difference between them (V_t), without taking into account the uncertainty of those quantities. This suggests that the values encoded in those regions are divorced from any uncertainty-related information, which would render them insufficient to drive uncertainty-guided exploratory behavior on their own. Uncertainty would have to be computed elsewhere and then integrated with these value signals by downstream decision circuits closer to motor output. Our results do not contradict these studies (in fact, we observe traces of value coding in vmPFC in our data as well; see Figure A.9C and Supplemental Information) but instead point to the possibility that the value signal is computed separately and combined with the uncertainty signals from RLPFC and DLPFC downstream by motor cortex, which ultimately computes choice.

One mechanism by which this could occur is suggested by sequential sampling models, which posit that the decision value DV_t drives a noisy accumulator to a decision bound, at which point a decision is made [31]. This is consistent with Gershman [106]'s analysis of reaction time patterns on the same task as ours. It is also consistent with studies reporting neural signatures of evidence accumulation during perceptual as well as value-based judgments in human motor cortex [125,128,132,118,227]. It is worth noting that for our right-handed subjects, left motor cortex is the final cortical area implementing the motor choice. One potential avenue for future studies would be to investigate whether the decision value area will shift if subjects respond using a different modality, such as their left hand, or using eye movements. This would be consistent with previous studies that have identified

effector-specific value coding in human cortex [111].

One prediction following from the sequential sampling interpretation is that motor cortex should be more active for more challenging choices (i.e. when DV_t is close to zero), since the evidence accumulation process would take longer to reach the decision threshold. Indeed, this is consistent with our result, and reconciles the apparently perplexing negative direction of the effect: $|DV_t|$ can be understood as reflecting decision confidence, since a large $|DV_t|$ indicates that one option is significantly preferable to the other, making it highly likely that it would be chosen, whereas a small $|DV_t|$ indicates that the two options are comparable, making choices more stochastic (Eq. 1.4 and 1.5). Since we found that motor cortex is negatively correlated with $|DV_t|$, this means that it is negatively correlated with decision confidence, or equivalently, that it is positively correlated with decision uncertainty. In other words, motor cortex is more active for more challenging choices, as predicted by the sequential sampling framework.

Another puzzling aspect of our results that merits further investigation is the lack of any signal corresponding to V_t/TU_t. This suggests that the division might be performed by circuits downstream from right DLPFC, such as motor cortex. Alternatively, it could be that, true to Thompson sampling, the brain is generating samples from the posterior value distributions and comparing them to make decisions. In that case, what we are seeing in motor cortex could be the average of those samples, consistent with the analytical form of the UCB/Thompson hybrid (Eq. 1.4) which is derived precisely by averaging over all possible samples [105]. A sampling mechanism could thus explain both the negative sign of the $|DV_t|$ effect in motor cortex, as well as the absence of V_t/TU_t in the BOLD signal. Such a sampling mechanism also predicts that the variance of the decision value signal should scale with (squared) total uncertainty, which is precisely what we found. Overall, our data suggest that random exploration might be implemented by a sampling mechanism which directly enters the drawn samples into the decision value computation in motor cortex.

The neural and behavioral dissociation between relative and total uncertainty found in our study points to potential avenues for future research that could establish a double dissociation between the corresponding brain regions. Temporarily disrupting activity in right RLPFC should affect directed exploration (reducing w_2 in

Eq. 1.4), while leaving random exploration intact (not changing w_3 in Eq. 1.4). Conversely, disrupting right DLPFC should affect random but not directed exploration. This would expand upon the RLPFC results of Zajkowski et al. [321] by establishing a causal role for both regions in the corresponding uncertainty computations and exploration strategies.

In summary, we show that humans tackle the exploration-exploitation trade-off using a combination of directed and random exploration strategies driven by neurally dissociable uncertainty computations. Relative uncertainty was correlated with activity in right RLPFC and influenced directed exploration, while total uncertainty was correlated with activity in right DLPFC and influenced random exploration. Subjective trial-by-trial estimates decoded from both regions predicted subject responding, while motor cortex reflected the combined uncertainty and value signals necessary to compute choice. Our results are thus consistent with a hybrid computational architecture in which relative and total uncertainty are computed separately in right RLPFC and right DLPFC, respectively, and then integrated with value in motor cortex to ultimately perform the categorical decision computation.

1.10 Methods

1.10.1 Subjects

We recruited 31 subjects (17 female) from the Cambridge community. All subjects were healthy, ages 18-35, right-handed, with normal or corrected vision, and no neuropsychiatric pre-conditions. Subjects were paid $50.00 for their participation plus a bonus based on their performance. The bonus was the number of points from a random trial paid in dollars (negative points were rounded up to 1). All subjects received written consent and the study was approved by the Harvard Institutional Review Board.

1.10.2 Experimental Design and Statistical Analysis

We used the two-armed bandit task described in Gershman [106]. On each block, subjects played a new pair of bandits for 10 trials. Each subject played 32 blocks, with 4 blocks in each scanner run (for a total of 8 runs per subject). On each trial, subjects chose an arm and received reward feedback (points delivered by the chosen arm). Subjects were told to pick the better arm in each trial. To incentivize good performance, subjects were told that at the end of the experiment, a trial will be drawn randomly and they will receive the number of points in dollars, with negative rewards rounded up to 1. While eliminating the possibility of losses may appear to have altered the incentive structure of the task, subjects nevertheless preferred the better option across all task conditions (Figure A.1A), in accordance with previous replications of the experiment [105,106].

On each block, the mean reward $\mu(k)$ for each arm k was drawn randomly from a Gaussian with mean 0 and variance $\tau_0^2(k) = 100$. Arms on each block were designated as "risky" (R) or "safe" (S), with all four block conditions (RS, SR, RR, and SS) counterbalanced and randomly shuffled within each run (i.e., each run had one of each block condition in a random order). A safe arm delivered the same reward $\mu(S)$ on each trial during the block. A risky arm delivered rewards sampled randomly from a Gaussian with mean $\mu(R)$ and variance $\tau^2(R) = 16$. The type of each arm was indicated to subjects by the letter R or S above the corresponding box. In order to make it easier for subjects to distinguish between the arms within and across blocks, each box was filled with a random color that remained constant throughout the blocks and changed between blocks. The color was not informative of rewards.

Subjects were given the following written instructions.

> *In this task, you have a choice between two slot machines, represented by colored boxes. When you choose one of the slot machines, you will win or lose points. One slot machine is always better than the other, but choosing the same slot machine will not always give you the same points. Your goal is to choose the slot machine that you think will give you the most points. Sometimes the machines are "safe" (always delivering the same points), and sometimes the machines are "risky" (delivering variable*

points). Before you make a choice, you will get information about each machine: "S" indicates SAFE, "R" indicates RISKY. The "safe" or "risky" status does not tell you how rewarding a machine is. A risky machine could deliver more or less points on average than a safe machine. You cannot predict how good a machine is simply based on whether it is considered safe or risky. Some boxes will deliver negative points. In those situations, you should select the one that is least negative. In the MRI scanner, you will play 32 games, each with a different pair of slot machines. Each game will consist of 10 trials. Before we begin the actual experiment, you will play a few practice games. Choose the left slot machine by pressing with your index finger and the right slot machine by pressing with your middle finger. You will have 2 seconds to make a choice. To encourage you to do your best, at the end of the MRI experiment, a random trial will be chosen and you will be paid the number of points you won on that trial in dollars. You want to do as best as possible every time you make a choice!

The event sequence within a trial is shown in Figure 1.1A. At trial onset, subjects saw two boxes representing the two arms and chose one of them by pressing with their index finger or their middle finger on a response button box. The chosen arm was highlighted and, after a random inter-stimulus interval (ISI), they received reward feedback as the number of points they earned from that arm. No feedback was provided for the unchosen arm. Feedback remained on the screen for 1 s, followed by a random inter-trial interval (ITI) and the next trial. A white fixation cross was displayed during the ITI. If subjects failed to respond within 2 s, they were not given feedback and after the ISI, they directly entered the ITI with a red fixation cross. Otherwise, the residual difference between 2 s and their reaction time was added to the following ITI. Each block was preceded by a 6-second inter-block-interval, during which subjects saw a sign, "New game is starting," for 3 s, followed by a 3-second fixation cross. A 10-second fixation cross was added to the beginning and end of each run to allow for scanner stabilization and hemodynamic lag, respectively. ISIs and ITIs were pregenerated by drawing uniformly from the ranges 1-3 s and 5-7 s, respectively. Additionally, ISIs and ITIs in each run were uniformly scaled such that the total run duration is exactly 484 s, which is the expected run length assuming an average ISI (2 s) and an average ITI (6 s). This accounted for small deviations from the expected run duration and allowed us to acquire

242 whole-brain volumes during each run (TR = 2 s). The experiment was implemented using the PsychoPy toolbox[224].

1.10.3 Belief updating model

Following Gershman [106], we assumed subjects approximate an ideal Bayesian observer that tracks the expected value and uncertainty for each arm. Since rewards in our task are Gaussian-distributed, these correspond to the posterior mean $Q_t(k)$ and variance $\sigma_t^2(k)$ of each arm k, which can be updated recursively on each trial t using the Kalman filtering equations:

$$Q_{t+1}(a_t) = Q_t(a_t) + \alpha_t[r_t - Q_t(a_t)] \tag{1.6}$$

$$\sigma_{t+1}^2(a_t) = \sigma_t^2(a_t) - \alpha_t\sigma_t^2(a_t), \tag{1.7}$$

where a_t is the chosen arm, r_t is the received reward, and the learning rate α_t is given by:

$$\alpha_t = \frac{\sigma_t^2(a_t)}{\sigma_t^2(a_t) + \tau^2(a_t)}. \tag{1.8}$$

We initialized the values with the prior means, $Q_t(k) = 0$ for all k, and variances with the prior variances, $\sigma_1^2(k) = \tau_0^2(k)$. Subjects were informed of those priors and performed 4 practice blocks (40 trials) before entering the scanner to familiarize themselves with the task structure. Kalman filtering is the Bayes-optimal algorithm for updating the values and uncertainties given the task structure and has been previously shown to account well for human choices in bandit tasks[57,254,267,105]. In order to prevent degeneracy of the Kalman update for safe arms, we used $\tau^2(S) = 0.00001$ instead of zero, which is equivalent to assuming a negligible amount of noise even for safe arms. Notice that in this case, the learning rate is $\alpha_t \approx 1$ and as soon as the safe arm k is sampled, the posterior mean and variance are updated to $Q_{t+1}(k) \approx \mu(k)$ and $\sigma_t^2(k) \approx 0$, respectively.

1.10.4 Choice probability analysis

We fit the coefficients $\mathbf{w}$ of the hybrid model (Eq. 1.4) using mixed-effects maximum likelihood estimation (fitglme in MATLAB, with FitMethod=Laplace, CovariancePattern=diagonal, and EBMethod=TrustRegion2D) using all non-timeout trials (i.e., all trials on which the subject made a choice within 2 s of trial onset). In Wilkinson notation [308], the model specification was: Choice ~ V + RU + VoverTU + (V + RU + VoverTU | SubjectID).

We confirmed the parameter recovery capabilities of our approach by running the hybrid model genaratively on the same bandits as the subjects [309]. We drew the weights in each simulation according to $\mathbf{w} \sim \mathcal{N}(0, 10 \times \mathbf{I})$ and repeated the process 1000 times. We found a strong correlation between the generated and the recovered weights (Figure A.6A; $r > 0.99, p < 10^{-8}$ for all weights) and no correlation between the recovered weights (Figure A.6B; $r < 0.03, p > 0.3$ for all pairs), thus validating our approach. We fit the lesioned models in Table A.1 in the same way.

To generate Figure A.4, we similarly ran the model generativity, but this time using a grid of 16 evenly spaced values between 0 and 1 for each coefficient. For every setting of the coefficients $\mathbf{w}$, we computed performance as the proportion of times the better option was chosen, averaged across simulated subjects, averaged across 10 separate iterations. We preferred this metric over total reward as it yields more comparable results across different bandit pairs. To generate Figure A.3, we similarly ran the model generativity, but this time using the fitted coefficients.

In addition to the hybrid model, we also fit a model of choices as a function of experimental condition to obtain the slope and intercept of the choice probability function:

$$P(a_t = 1|\mathbf{w}) = \Phi\left(\sum_j w_1^j \pi_{tj} + w_2^j \pi_{tj} V_t\right), \tag{1.9}$$

where j is the experimental condition (RS, SR, RR, or SS), and $\pi_{tj} = 1$ if trial t is assigned to condition j, and

0 otherwise. In Wilkinson notation, the model specification was: Choice ~ condition + condition:V + (condition + condition:V | SubjectID). We plotted the w_1 terms as the intercepts and the w_2 terms as the slopes.

For Bayesian model comparison, we fit **w** separately for each subject using fixed effects maximum likelihood estimation (Choice ~ V + RU + VoverTU) in order to obtain a separate BIC for each subject. We approximated the log model evidence for each subject as -0.5*BIC and used it to compute the protected exceedance probability for each model, which is the probability that the model is most prevalent in the population [238].

1.10.5 fMRI Data Acquisition

We followed the same protocol as described previously [292]. Scanning was carried out on a 3T Siemens Magnetom Prisma MRI scanner with the vendor 32-channel head coil (Siemens Healthcare, Erlangen, Germany) at the Harvard University Center for Brain Science Neuroimaging. A T1-weighted high-resolution multi-echo magnetization-prepared rapid-acquisition gradient echo (ME-MPRAGE) anatomical scan [297] of the whole brain was acquired for each subject prior to any functional scanning (176 sagittal slices, voxel size = 1.0 x 1.0 x 1.0 mm, TR = 2530 ms, TE = 1.69 - 7.27 ms, TI = 1100 ms, flip angle = 7°, FOV = 256 mm). Functional images were acquired using a T2*-weighted echo-planar imaging (EPI) pulse sequence that employed multiband RF pulses and Simultaneous Multi-Slice (SMS) acquisition [201,90,316]. In total, 8 functional runs were collected for each subject, with each run corresponding to 4 task blocks, one in each condition (84 interleaved axial-oblique slices per whole brain volume, voxel size = 1.5 x 1.5 x 1.5 mm, TR = 2000 ms, TE = 30 ms, flip angle = 80°, in-plane acceleration (GRAPPA) factor = 2, multi-band acceleration factor = 3, FOV = 204 mm). The initial 5 TRs (10 s) were discarded as the scanner stabilized. Functional slices were oriented to a 25 degree tilt towards coronal from AC-PC alignment. The SMS-EPI acquisitions used the CMRR-MB pulse sequence from the University of Minnesota.

All 31 scanned subjects were included in the analysis. We excluded runs with excessive motion (greater than 2 mm translational motion or greater than 2 degrees rotational motion). Four subjects had a single excluded run and two additional subjects had two excluded runs.

1.10.6 fMRI Preprocessing

As in our previous work[292], functional images were preprocessed and analyzed using SPM12 (Wellcome Department of Imaging Neuroscience, London, UK). Each functional scan was realigned to correct for small movements between scans, producing an aligned set of images and a mean image for each subject. The high-resolution T1-weighted ME-MPRAGE images were then co-registered to the mean realigned images and the gray matter was segmented out and normalized to the gray matter of a standard Montreal Neurological Institute (MNI) reference brain. The functional images were then normalized to the MNI template (resampled voxel size 2 mm isotropic), spatially smoothed with a 8 mm full-width at half-maximum (FWHM) Gaussian kernel, high-pass filtered at 1/128 Hz, and corrected for temporal autocorrelations using a first-order autoregressive model.

1.10.7 Univariate Analysis

Our hypothesis was that different brain regions perform the two kinds of uncertainty computations (relative and total uncertainty), which in turn drive the two corresponding exploration strategies (directed exploration, operationalized as UCB, and random exploration, operationalized as Thompson sampling). We therefore defined a general linear model (GLM 1, Table A.2) with model-based trial-by-trial posterior estimates of absolute relative uncertainty, $|\mathrm{RU}_t|$, total uncertainty, TU_t, absolute value difference, $|\mathrm{V}_t|$, and absolute value difference scaled by total uncertainty, $|\mathrm{V}_t|/\mathrm{TU}_t$, as parametric modulators for an impulse regressor at trial onset (trial_onset). All quantities were the same model-derived ideal observer trial by trial estimates which we used to model choices (Eq. 1.4). For ease of notation, we sometimes refer to those parametric modulators as RU, TU, V, and V/TU, respectively. Following[11], we used $|\mathrm{RU}_t|$ instead of RU_t to account for our arbitrary choice of arm 1 and arm 2 (note that TU_t is always positive). We used the absolute value difference $|\mathrm{V}_t|$ for the same reason.

The trial_onset regressor was only included on trials on which the subject responded within 2 s of trial onset (i.e., non-timeout trials). We included a separate regressor at trial onset for trials on which the subject timed out (i.e., failed to respond within 2 s of trial onset) that was not parametrically modulated (trial_onset_timeout), since

failure to respond could be indicative of failure to perform the necessary uncertainty computations. In order to control for any motor-related activity that might be captured by those regressors due to the hemodynamic lag, we also included a separate trial onset regressor on trials on which the subject chose arm 1 (trial_onset_chose_1). Finally, to account for response-related activity and feedback-related activity, we included regressors at reaction time (button_press) and feedback onset (feedback_onset), respectively. All regressors were impulse regressors (duration = 0 s) convolved with the canonical hemodynamic response function (HRF). The parametric modulators were not orthogonalized[204]. As is standard in SPM, there was a separate version of each regressor for every scanner run. Additionally, there were six motion regressors and an intercept regressor.

For group-level whole-brain analyses, we performed t-contrasts with single voxels thresholded at $p < 0.001$ and cluster family-wise error (FWE) correction applied at $\alpha = 0.05$, reported in Figure A.7 and Tables A.3 and A.4. Uncorrected contrasts are shown in Figures 1.3 and 1.4. We labeled clusters based on peak voxel labels from the deterministic Automated Anatomical Labeling (AAL2) atlas[295,240]. For clusters whose peak voxels were not labeled successfully by AAL2, we consulted the SPM Anatomy Toolbox[84] and the CMA Harvard-Oxford atlas[67]. We report up to 3 peaks per cluster, with a minimum peak separation of 20 voxels. All voxel coordinates are reported in Montreal Neurological Institute (MNI) space.

Since the positive cluster for $|\mathrm{RU}_t|$ in right RLPFC did not survive FWE correction, we resorted to using *a priori* ROIs from a study by Badre et al. [11] for our subsequent analysis. Even though in their study subjects performed the different task and the authors used a different model, we believe the underlying uncertainty computations are equivalent to those in our study and hence likely to involve the same neural circuits. We defined the ROIs as spheres of radius 10 mm around the peak voxels for the corresponding contrasts reported by Badre et al. [11]: right RLPFC for relative uncertainty (MNI [36 56 -8]) and right DLPFC for total uncertainty (MNI [38 30 34]).

To compute the main effect in a given ROI, we averaged the neural coefficients (betas) within a sphere of radius 10 mm centered at the peak voxel of the ROI for each subject, and then performed a two-tailed t-test against 0 across subjects. To compute a contrast in a given ROI, we performed a paired two-tailed t-test across subjects

between the betas for one regressor (e.g. $|\mathrm{RU}_t|$) and the betas for the other regressor (e.g. $|\mathrm{TU}_t|$), again averaged within a 10 mm-radius sphere.

We used the same methods for the decision value GLM (GLM 2, Table A.2) as with GLM 1.

1.10.8 Decoding

If the brain regions reported by Badre et al. [11] encode subjective trial-by-trial estimates of relative and total uncertainty, as our GLM 1 results suggest, and if those estimates dictate choices, as our UCB/Thompson hybrid model predicts, then we should be able to read out those subjective estimates and use them to improve the model predictions of subject choices. This can be achieved by "inverting" the GLM and solving for $|\mathrm{RU}_t|$ and TU_t based on the neural data y, the beta coefficients β, and the design matrix X. Using ridge regression to prevent overfitting, the subjective estimate for relative uncertainty for a given voxel on trial t can be computed as:

$$\widehat{|\mathrm{RU}_t|} = \left(y_t - \sum_{i : X_{t,i} \neq |\mathrm{RU}|} X_{t,i} \beta_i \right) \beta_{|\mathrm{RU}|} / (\beta_{|\mathrm{RU}|}^2 + \lambda) \tag{1.10}$$

where y_t is the neural signal on trial t, $X_{t,i}$ is the value of regressor i on trial t, β_i is the corresponding beta coefficient computed by SPM, $\beta_{|\mathrm{RU}|}$ is the beta coefficient for $|\mathrm{RU}|$, and λ is the ridge regularization constant (voxel indices were omitted to keep the notation uncluttered). The sum is taken over all regressors i other than $|\mathrm{RU}|$. To account for the hemodynamic lag, we approximated the neural activity at time t as the raw BOLD signal at time $t + 5$ s, which corresponds to the peak of the canonical HRF in SPM (spm_hrf). Since the beta coefficients were already fit by SPM, we could not perform cross-validation to choose λ and so we arbitrarily set $\lambda = 1$.

To obtain a signed estimate for relative uncertainty, we flipped the sign based on the model-based RU_t:

$$\widehat{\mathrm{RU}_t} = \begin{cases} |\widehat{\mathrm{RU}_t}| & \text{if } \mathrm{RU}_t >= 0 \\ -|\widehat{\mathrm{RU}_t}| & \text{if } \mathrm{RU}_t < 0 \end{cases} \tag{1.11}$$

Note that our goal is to test whether we can improve our choice predictions, given that we already know the model-based RU_t, so this does not introduce bias into the analysis. Finally, we averaged $\widehat{\mathrm{RU}_t}$ across all voxels within the given ROI (a 10 mm sphere around the peak voxel) to obtain a single trial-by-trial subjective estimate $\widehat{\mathrm{RU}_t}$ for that ROI. We used the same method to obtain a trial-by-trial subjective estimate of total uncertainty, $\widehat{\mathrm{TU}_t}$.

To check if the neurally-derived $\widehat{\mathrm{RU}_t}$ predicts choices, we augmented the probit regression model of choice in Eq. 1.4 to:

$$P(a_t = 1|\mathbf{w}) = \Phi(w_0 + w_1\mathrm{V}_t + w_2\mathrm{RU}_t + w_3\mathrm{V}_t/\mathrm{TU}_t + w_4\widehat{\mathrm{RU}_t}). \tag{1.12}$$

In Wilkinson notation [308], the model specification was: `Choice ~ 1 + V + RU + VoverTU + decodedRU + (1 + V + RU + VoverTU + decodedRU | SubjectID)`.

Notice that we additionally included an intercept term w_0. While this departs from the proper analytical form of the UCB/Thompson hybrid (Eq. 1.4), we noticed that including an intercept term alone is sufficient to improve choice predictions (data not shown), indicating that some subjects had a bias for one arm over the other. We therefore chose to include an intercept to guard against false positives, which could occur, for example, if we decode low-frequency noise which adopts the role of a *de facto* intercept.

We then fit the model in the same way as the probit regression model in Eq. 1.4, using mixed effects maximum-likelihood estimation (`fitglme`), with the difference that we omitted trials from runs that were excluded from the fMRI analysis. For baseline comparison, we also re-fitted the original model (Eq. 1.4) with an intercept and

without the excluded runs (hence the difference between the UCB/Thompson hybrid fits in Table 1.1 and Table A.1).

Similarly, we defined an augemented model for $\widehat{\mathrm{TU}_t}$:

$$P(a_t = 1|\mathbf{w}) = \Phi(w_0 + w_1 \mathrm{V}_t + w_2 \mathrm{RU}_t + w_3 \mathrm{V}_t/\mathrm{TU}_t + w_4 \mathrm{V}_t/\widehat{\mathrm{TU}_t}), \tag{1.13}$$

Note that this analysis cannot be circular since it evaluates the ROIs based on behavior, which was not used to define GLM 1 [163]. In particular, all regressors and parametric modulators in GLM 1 were defined purely based on model-derived ideal observer quantities; we only take into account subjects' choices when fitting the weights $\mathbf{w}$ (Eq. 1.4), which were not used to define the GLM 1. Furthermore, since all model-derived regressors are also included in all augmented models of choice (Eq. 1.12, 1.13, 1.14), any additional choice information contributed by the neurally decoded regressors is necessarily above and beyond what was already included in GLM 1.

Finally, we entered both subjective estimates from the corresponding ROIs into the same augmented model:

$$P(a_t = 1|\mathbf{w}) = \Phi(w_0 + w_1 \mathrm{V}_t + w_2 \mathrm{RU}_t + w_3 \mathrm{V}_t/\mathrm{TU}_t + w_4 \widehat{\mathrm{RU}_t} + w_5 \mathrm{V}_t/\widehat{\mathrm{TU}_t}), \tag{1.14}$$

We similarly constructed an augmented model with the decoded decision value, $\widehat{\mathrm{DV}_t}$, from GLM 2, after adjusting the sign similarly to Eq. 1.11:

$$P(a_t = 1|\mathbf{w}) = \Phi(w_0 + w_1 \mathrm{V}_t + w_2 \mathrm{RU}_t + w_3 \mathrm{V}_t/\mathrm{TU}_t + w_4 \widehat{\mathrm{DV}_t}), \tag{1.15}$$

1.10.9 Residual Variance Analysis

To show that the variance of the decision value signal scales with total uncertainty, we extracted the residuals of the GLM 2 fits from the $|DV_t|$ ROI (Figure 1.5; left M1, peak MNI [-38 -8 62]), averaged within a 10-mm sphere around the peak voxel. As with the decoding analysis, we accounted for the hemodynamic lag by taking the residuals 5 s after trial onset to correspond to the residual activations on the given trial. We then performed a Pearson correlation between the square of the residuals (the residual variance) and TU_t^2 across trials for each subject. Finally, to aggregate across subjects, we Fisher z-transformed the resulting correlation coefficients and performed a two-tailed one sample t-test against zero.

1.10.10 Code and data availability

All analyses were conducted in MATLAB using SPM 12 and our custom fMRI analysis pipeline built on top of it, which is available at `https://github.com/sjgershm/ccnl-fmri`. All behavioral data and analysis code are available `https://github.com/tomov/Exploration-fMRI-Task`. The raw fMRI data is available upon request.

2

Neural Computations Underlying Causal Structure Learning

2.1 Abstract

Behavioral evidence suggests that beliefs about causal structure constrain associative learning, determining which stimuli can enter into association, as well as the functional form of that association. Bayesian learning theory provides one mechanism by which structural beliefs can be acquired from experience, but the neural basis of this mechanism is poorly understood. We studied this question with a combination of behavioral, computational and neuroimaging techniques. Male and female human subjects learned to predict an outcome based on cue and context stimuli, while being scanned using functional MRI. Using a model-based analysis of the fMRI data, we show that structure learning signals are encoded in posterior parietal cortex, lateral prefrontal cortex, and the frontal pole. These structure learning signals are distinct from associative learning signals. Moreover, representational similarity analysis and information mapping revealed that the multivariate patterns of activity in posterior parietal cortex and anterior insula encode the full posterior distribution over causal structures. Variability in the encoding of the posterior across subjects predicted variability in their subsequent behavioral performance. These results provide evidence for a neural architecture in which structure learning guides the formation of associations.

2.2 Introduction

Classical learning theories posit that animals learn associations between sensory stimuli and rewarding outcomes [234,222]. These theories have achieved remarkable success in explaining a wide range of behaviors using simple mathematical rules. Yet numerous studies have challenged some of their foundational premises [199,109,83]. One particularly longstanding puzzle for these theories is the multifaceted role of contextual stimuli in associative learning. Some studies have shown that the context in which learning takes place is largely irrelevant [29,180,150,30], whereas others have found that context plays the role of an "occasion setter," modulating cue-outcome associations without itself acquiring associative strength [282,124,28,281]. Yet other studies suggest that context acts like another punctate cue, entering into summation and cue competition with other stimuli [14,126]. The multiplicity

of such behavioral patterns defies explanation in terms of a single associative structure, suggesting instead that different structures may come into play depending on the task and training history.

Computational modeling has begun to unravel this puzzle, using the idea that structure is a latent variable inferred from experience [104]. On this account, each structure corresponds to a causal model of the environment, specifying the links between context, cues and outcomes, as well as their functional form. The learner thus faces the joint problem of inferring both the structure and the strength of causal relationships, which can be implemented computationally using Bayesian learning [130,159,194]. This account can explain why different tasks and training histories produce different forms of context-dependence: variations across tasks induce different probabilistic beliefs about causal structure. For example, Gershman [104] showed that manipulations of context informativeness, outcome intensity, and number of training trials have predictable effects on the functional role of context in animal learning experiments [214,230].

If this account is correct, then we should expect to see separate neural signatures of structure learning and associative learning that are systematically related to behavioral performance. However, the direct neural evidence for structure learning is currently sparse [42,284,184]. In this study, we seek to address this gap using human fMRI and an associative learning paradigm adapted from Gershman [104]. On each block, subjects were trained on cue-context-outcome combinations that were consistent with a particular causal interpretation. Subjects were then asked to make predictions about novel cues and contexts without feedback, revealing the degree to which their beliefs conformed to a specific causal structure. We found that a variant of the structure learning framework developed by Gershman [104] accounted for the subjects' predictive judgments, which led us to hypothesize a neural implementation of its computational components. We additionally found that an alternative structure learning model developed by Collins & Frank [43] also accounts for the subjects' behavior, so we used both models to investigate the neural correlates of structure learning.

We found trial-by-trial signals tracking structure learning above and beyond associative learning. A whole-brain analysis revealed a univariate signature of Bayesian updating of the posterior distribution over causal structures in a frontoparietal network of regions, including the inferior part of posterior parietal cortex (inferior PPC),

lateral prefrontal cortex (lateral PFC), and rostrolateral prefrontal cortex (RLPFC). Bayesian updating of structural beliefs according to the Collins & Frank [43] model correlated with a network of regions that largely overlapped with the regions identified by our model, suggesting that both models tap into a generic structure learning mechanism in the brain. A multivariate analysis implicated some of those regions in the representation of the full posterior distribution over causal structures. Activity in two of those regions – left inferior PPC and right anterior insula – also predicted subsequent generalization on the test trials in accordance with the causal structure learning model. Our results provide new insight into the neural mechanisms of structure learning and how they constrain the acquisition of associations.

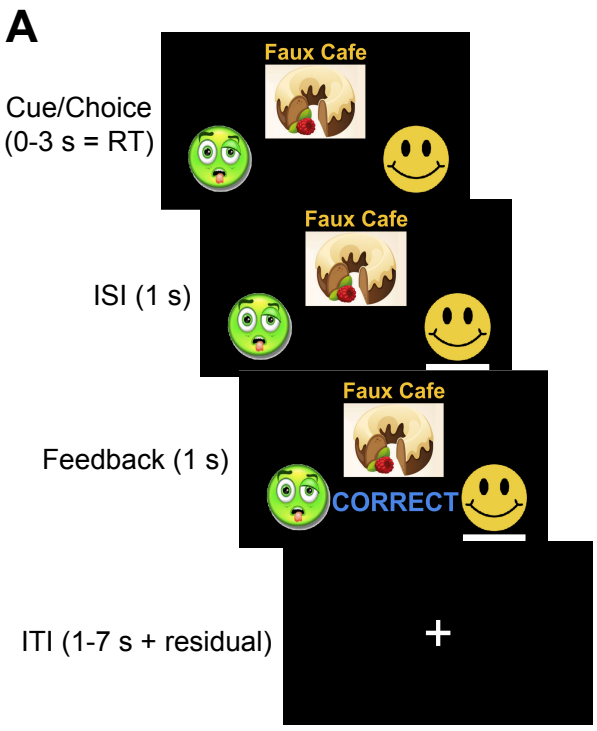
2.3 Methods

2.3.1 Subjects

Twenty-seven healthy subjects were enrolled in the fMRI portion of the study. Although we did not perform power analysis to estimate the sample size, it is consistent with the size of the pilot group of subjects that showed a robust behavioral effect (Figure 2.4, grey circles). Prior to data analysis, seven subjects were excluded due to technical issues, insufficient data, or excessive head motion. The remaining 20 subjects were used in the analysis (10 female, 10 male; 19-27 years of age; mean age 20 ± 2; all right handed with normal or corrected-to-normal vision). Additionally, 10 different subjects were recruited for a behavioral pilot version of the study that was conducted prior to the fMRI portion. All subjects received informed consent and the study was approved by the Harvard University Institutional Review Board. All subjects were paid for their participation.

2.3.2 Experimental Design and Statistical Analysis

We adapted the task used in Gershman [104] to a within-subjects design. Subjects were told that they would play the role of a health inspector trying to determine the cause of illness in different restaurants around the city. The experiment consisted of nine blocks. Each block consisted of 20 training trials followed by four test trials. On



B

Condition	Training phase	Test phase
Irrelevant context	x1c1+ x2c1- x1c2+ x2c2-	x1c1 x3c1 x1c3 x3c3
Modulatory context	x1c1+ x2c1- x1c2- x2c2+	x1c1 x3c1 x1c3 x3c3
Irrelevant cue	x1c1+ x2c1+ x1c2- x2c2-	x1c1 x3c1 x1c3 x3c3

Figure 2.1: Experimental Design

(A) Timeline of events during a training trial. Subjects are shown a cue (food) and context (restaurant) and are asked to predict whether the food will make a customer sick. They then see a line under the chosen option, and feedback indicating a "Correct" or "Incorrect" response. ISI: interstimulus interval; ITI: intertrial interval.

(B) Stimulus-outcome contingencies in each condition. Cues denoted by (x_1, x_2, x_3) and contexts denoted by (c_1, c_2, c_3). Outcome presentation denoted by "$+$" and no outcome denoted by "$-$".

each training trial, subjects were shown a given cue (the food) in a given context (the restaurant) and asked to predict whether that cue-context combination would cause sickness. After making a prediction, they were informed whether their prediction was correct (Figure 2.1A). On a given block, the assignment of stimuli to outcomes was deterministic, such that the same cue-context pair always led to the same outcome. Even though the computational model could support stochastic and dynamically evolving stimulus-outcome contingencies, our goal was to provide a minimalist design that can assess the main predictions of the theory. There were four distinct training cue-context pairs (two foods $\times$ two restaurants) on each block, such that two of the pairs always caused sickness while the other two never caused sickness. Each cue-context pair was shown five times in each block for a total of 20 randomly shuffled training trials.

Crucially, the stimulus-outcome contingencies in each block were designed to promote a particular causal interpretation of the environment (Figure 2.1B, Figure 2.2). On *irrelevant context* blocks, one cue caused sickness in both contexts, while the other cue never caused sickness, thus rendering the contextual stimulus irrelevant for making correct predictions. On *modulatory context* blocks, the cue-outcome contingency was reversed across contexts, such that the same cue caused sickness in one context but not the other, and vice versa for the other cue. On these blocks, context thus acted like an "occasion setter", determining the sign of the cue-outcome association. Finally, on *irrelevant cue* blocks, both cues caused sickness in one context but neither cue caused sickness in the other context, thus favoring an interpretation of context acting as a punctate cue. There were no explicit instructions or other signals that indicated the different block conditions other than the stimulus-outcome contingencies. We based our experimental design on the fact that a previously published model with similar structures could capture a wide array of behavioral phenomena [104] and that the chosen stimuli-outcome contingencies establish a clear behavioral pattern that we can build upon to explore the neural correlates of structure learning.

Behavior was evaluated on four test trials during which subjects were similarly asked to make predictions, however this time without receiving feedback. Subjects were presented with one novel cue and one novel context, resulting in four (old cue vs. new cue) $\times$ (old context vs. new context) randomly shuffled test combinations (Figure 2.1B). The old cue and the old context were always chosen such that their combination caused sickness

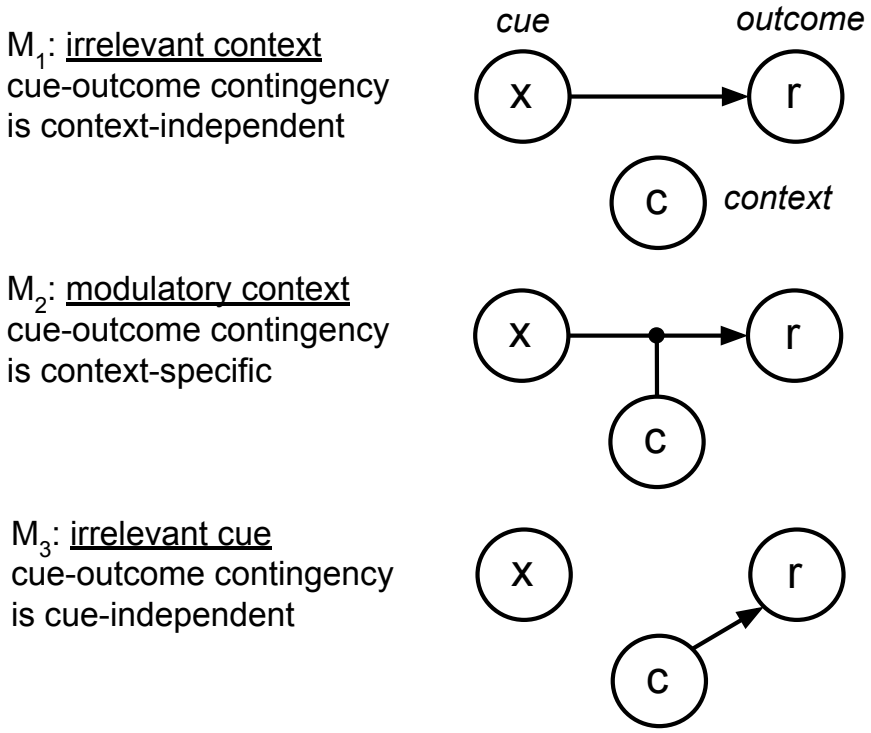


Figure 2.2: Hypothesis Space of Causal Structures

Each causal structure is depicted as a network where the nodes represent variables and the edges represent causal connections. In $\mathcal{M}_2$, the context modulates the causal relationship between the cue and the outcome. Adapted from Gershman [104].

during training. Importantly, different causal structures predict different patterns of generalization on the remaining three trials which contain a new cue and/or a new context. If context is deemed to be irrelevant, the old cue should always predict sickness, even when presented in a new context. If a modulatory role of context is preferred, then no inferences can be made about any of the three pairs that include a new cue or a new context. Finally, if context is interpreted as acting like a cue, then both the old cue and the new cue should predict sickness in the old context but not in the new context.

Each block was assigned to one of the three conditions (irrelevant context, modulatory context, or irrelevant cue) and each condition appeared three times for each subject, for a total of nine blocks. The block order was randomized in groups of three, such that the first three blocks covered all three conditions in a random order, and so did the next three blocks and the final three blocks. We used nine sets of foods and restaurants corresponding to different cuisines (Chinese, Japanese, Indian, Mexican, Greek, French, Italian, fast food and brunch). Each set consisted of three clipart food images (cues) and three restaurant names (contexts). For each subject, blocks were randomly matched with cuisines, such that subjects had to learn and generalize for a new set of stimuli on each block. The assignment of cuisines was independent of the block condition. The valence of the stimuli was also randomized across subjects, such that the same cue-context pair could predict sickness for some subjects but not others.

Prior to the experiment, the investigator read the task instructions aloud and subjects completed a single demonstration block of the task on a laptop outside the scanner. Subjects completed nine blocks of the task in the scanner, with one block per scanner run. Each block had a duration of 200 seconds during which 100 volumes were acquired (TR = 2 s). At the start of each block, a fixation cross was shown for 10 seconds and the corresponding 5 volumes were subsequently discarded. This was followed by the training phase, which lasted 144 seconds. The event sequence within an example training trial is shown in Figure 2.1. At trial onset, subjects were shown a food and restaurant pair and instructed to make a prediction. Subjects reported their responses by pressing the left or the right button on a response box. After trial onset, subjects were given 3 seconds to make a response. A response was immediately followed by a 1-second inter-stimulus interval (ISI) during which their

response was highlighted. The residual difference between 3 seconds and their reaction time was added to the subsequent inter-trial interval (ITI). The ISI was followed by a 1-second feedback period during which they were informed whether their choice was correct. If subjects failed to respond within 3 seconds of trial onset, no response was recorded and at feedback they were informed that they had timed out. During the ITIs, a fixation cross was shown. The trial order and the jittered ITIs for the training phase were generated using the optseq2 program [129] with ITIs between 1 and 12 seconds. The training phase was followed by a 4 second message informing the subjects they are about to enter the test phase. The test phase lasted 36 seconds. Test trials had a similar structure as training trials, with the difference that subjects were given 6 seconds to respond instead of 3 and there was no ISI nor feedback period. The ITIs after the first 3 test trials were 2, 4, and 6 seconds, randomly shuffled. The last training trial was followed by a 6-second fixation cross. The stimulus sequences and ITIs were pre-generated for all subjects. The task was implemented using the PsychoPy2 package [225]. The subjects in the behavioral pilot version of the study performed an identical version of the experiment, except that it was conducted on a laptop.

Behavioral data were analyzed using *t*-tests and computational modeling. Brain imaging data were analyzed using general linear models. The modeling for behavioral and neural data is described in more detail below.

2.3.3 Causal structure learning model

We implemented the causal structure learning model presented in Gershman [104], with the difference that the additive context structure was replaced by an irrelevant cue structure. This replacement was motivated by our observation that the model with an irrelevant cue structure had higher model evidence than the original model for our behavioral data. The key idea is that learners track the joint posterior over associative weights ($\mathbf{w}$) and causal structures (M), computed using Bayes' rule:

$$P(\mathbf{w}, M|\mathbf{h}_{1:n}) = \frac{P(\mathbf{h}_{1:n}|\mathbf{w}, M)P(\mathbf{w}|M)P(M)}{P(\mathbf{h}_{1:n})} \tag{2.1}$$

where $\mathbf{h}_{1:n} = (\mathbf{x}_{1:n}, \mathbf{r}_{1:n}, \mathbf{c}_{1:n})$ denotes the training history for trials 1 to n (cue-context-outcome combinations). The likelihood $P(\mathbf{h}_{1:n}|\mathbf{w}, \mathcal{M})$ encodes how well structure $\mathcal{M}$ predicts the training history, the prior $P(\mathbf{w}|\mathcal{M})$ specifies an inductive bias for the weight vector, and the prior over structures $P(\mathcal{M})$ was taken to be uniform, reflecting the assumption that all structures are equally probable *a priori*.

Generative model

Our model is based on the following assumptions about the dynamics that govern associations between stimuli and outcomes in the world. The training history is represented as $\mathbf{h}_{1:n} = (\mathbf{x}_{1:n}, \mathbf{r}_{1:n}, \mathbf{c}_{1:n})$ for trials 1 to n, consisting of the following variables:

- $\mathbf{x}_n \in \mathbb{R}^D$: the set of D cues observed at time n, where $x_{nd} = 1$ indicates that cue d is present and $x_{nd} = 0$ to indicate that it is absent. Thus each cue can be regarded as a "one-hot" D-dimensional vector and $\mathbf{x}_n$ can be viewed as the sum of all cues present on trial n. In our simulations, we use $D = 3$ and we only have a single cue (the food) present on each trial.

- $c_n \in \{1, \ldots, K\}$: the context, which can take on one of K discrete values. While contexts could in principle be represented as vectors as well, we restrict the model to one context per trial for simplicity. In our simulations, we take $K = 3$.

- $r_n \in \mathbb{R}$: the outcome. In our simulations, we use $r_n = 1$ for "sick" and $r_n = 0$ for "not sick".

We consider three specific structures relating the above variables. All the structures have in common that the outcome is assumed to be drawn from a Gaussian with variance $\sigma_r^2 = 0.01$:

$$r_n \sim \mathcal{N}\left(\bar{r}_n, \sigma_r^2\right), \tag{2.2}$$

where we have left the dependence on $\mathbf{c}_n$ and $\mathbf{x}_n$ implicit. The structures differ in how the mean $\bar{r}_n$ is computed.

- **Irrelevant context** (M_1):

$$\bar{r}_n = \sum_{d=1}^{D} w_d x_{nd} = \mathbf{w}^\top \mathbf{x}_n. \tag{2.3}$$

where d indexes the set of D cues. Under this structure, context c_n plays no role in determining the expected outcome $\bar{r}_n$ on trial n. Instead, a single set of weights $\mathbf{w}$ dictates the associative strength between each cue and the outcome, such that the expected outcome on a given trial is the sum of the associative weights of all present cues. The idea that context is irrelevant for stimulus-outcome associations is consistent with number of behavioral studies [29,180,150,30].

- **Modulatory context** (M_2):

$$\bar{r}_n = \sum_{d=1}^{D} w_{dk} x_{nd} = \mathbf{w}_k^\top \mathbf{x}_n \tag{2.4}$$

when $c_n = k$. Under this structure, each context $c_n = k$ specifies its own weight vector $\mathbf{w}_k$. Thus the same cue can make completely different predictions in different contexts. The view that context modulates stimulus-outcome associations is also supported by previous behavioral findings [282,124,28,281].

- **Irrelevant cue** (M_3):

$$\bar{r}_n = w_{D+k} = \mathbf{w}^\top \tilde{\mathbf{c}}_n, \tag{2.5}$$

where $c_n = k$ and $\tilde{c}_{nk} = 1$ if $c_n = k$, and 0 otherwise. This structure is symmetric with respect to M_1, in that we assume a one-hot context vector $\tilde{\mathbf{c}}_n$ that encodes the context in the same way that $\mathbf{x}_n$ encodes the cue in M_1. The weight vector $\mathbf{w}$ thus contains entries for contexts only. Previous work also suggests that context sometimes acts like another cue [14,126] and that cues are sometimes ignored when they are not predictive of outcomes [183]. Note that this is different from the additive structure used in Gershman [104], in

which the cue and the context summate together to predict the outcome. We chose this simpler structure as it more closely reflects the structure of the task, and preliminary model comparisons revealed that it provides a better account of behavior (data not shown).

We assume each weight is drawn independently from a Gaussian prior with mean w_0 and variance σ_w^2. Each weight can change slowly over time according to a Gaussian random walk with variance τ^2. These free parameters were fit using data from the behavioral pilot version of the study.

In summary, each causal structure corresponds to an internal model of the world in which the relationship between cues, contexts and outcomes can be described by a distinct linear-Gaussian dynamical system (LDS). While the LDS assumptions might seem excessive given the deterministic nature of the task, they have been widely used in the classical conditioning studies[64,149,165,103] to provide a parsimonious account for various learning phenomena. Here we employ them for the purposes of tractability and in order to remain consistent with the causal learning model that Gershman [104] used to explain the seemingly contradictory roles of context reported in the animal learning literature. These causal structures were inspired by different theories that have been advanced in various forms in the literature, none of which has been able to capture the broad range of results on its own.

Probabilistic inference

Assuming this generative model, a rational agent can use Bayesian inference to invert the model and use its training history $\mathbf{h}_{1:n}$ to learn the underlying causal structure M and its associative weights $\mathbf{w}$ (Eq. 2.1). To achieve this, first we can compute the posterior over the weights for a given model M using Bayes' rule:

$$P(\mathbf{w}|\mathbf{h}_{1:n}, M) = \frac{P(\mathbf{h}_{1:n}|\mathbf{w}, M)P(\mathbf{w}|M)}{P(\mathbf{h}_{1:n}|M)}. \tag{2.6}$$

For M_1, the posterior at time n is:

$$P(\mathbf{w}|\mathbf{h}_{1:n}, M = M_1) = \mathcal{N}(\mathbf{w}; \hat{\mathbf{w}}_n, \Sigma_n) \tag{2.7}$$

with parameters updated recursively as follows:

$$\hat{\mathbf{w}}_{n+1} = \hat{\mathbf{w}}_n + \mathbf{g}_n(r_n - \hat{\mathbf{w}}_n^\top \mathbf{x}_n) \tag{2.8}$$

$$\Sigma_{n+1} = \Sigma'_n - \mathbf{g}_n \mathbf{x}^\top \Sigma'_n, \tag{2.9}$$

where $\Sigma'_n = \Sigma_n + \tau^2 \mathbf{I}$. These update equations are known as *Kalman filtering*, an important algorithm in engineering and signal processing that has recently been applied to animal learning[64,165,103]. The initial estimates are given by the parameters of the prior: $\hat{\mathbf{w}}_0 = 0, \Sigma_0 = \sigma_w^2 \mathbf{I}$. The Kalman gain $\mathbf{g}_n$ (a vector of learning rates) is given by:

$$\mathbf{g}_n = \frac{\Sigma'_n \mathbf{x}_n}{\mathbf{x}_n^\top \Sigma'_n \mathbf{x}_n + \sigma_r^2}. \tag{2.10}$$

The same equations apply to M_2, but the mean and covariance are context-specific: $\hat{\mathbf{w}}_n^k$ and Σ_n^k. Accordingly, the Kalman gain is modified as follows:

$$\mathbf{g}_{nk} = \frac{\Sigma'_{nk} \mathbf{x}_n}{\mathbf{x}_n^\top \Sigma'_{nk} \mathbf{x}_n + \sigma_r^2} \tag{2.11}$$

if $c_n = k$, and a vector of zeros otherwise. For M_3, the same equations as M_1 apply, but to the context vector $\tilde{\mathbf{c}}_n$.

To make predictions about future outcomes, we need to compute the posterior predictive expectation, which is also available in closed form:

$$V_n = \mathbb{E}[r_n | \mathbf{x}_n, c_n, \mathbf{h}_{1:n-1}] = \sum_M \mathbb{E}[r_n | \mathbf{x}_n, c_n, \mathbf{h}_{1:n-1}, M] P(M | \mathbf{h}_{1:n-1}). \tag{2.12}$$

The first term in Eq. 2.12 is the posterior predictive expectation conditional on model M:

$$\mathbb{E}[r_n | \mathbf{x}_n, c_n, \mathbf{h}_{1:n-1}, M] = \mathbf{x}_n^\top \hat{\mathbf{w}}_n, \tag{2.13}$$

where again the variables are modified depending on what model is being considered. The second term in Eq. 2.12 is the posterior probability of model $\mathcal{M}$, which can be updated according to Bayes' rule:

$$P(\mathcal{M}|\mathbf{h}_{1:n}) \propto P(r_n|\mathbf{x}_n, c_n, \mathbf{h}_{1:n-1}, \mathcal{M})P(\mathcal{M}|\mathbf{h}_{1:n-1}), \tag{2.14}$$

where the likelihood is given by:

$$P(r_n|\mathbf{x}_n, \mathbf{c}_n, \mathbf{h}_{1:n-1}, \mathcal{M}) = \mathcal{N}(r_n; \mathbf{x}_n^\top \hat{\mathbf{w}}_n, \mathbf{x}_n^\top \Sigma'_n \mathbf{x}_n + \sigma_r^2). \tag{2.15}$$

To make predictions for the predictive learning experiment, we mapped the posterior predictive expectation onto choice probability (outcome vs. no outcome) by a logistic sigmoid transformation:

$$P(a_n = 1) = \frac{1}{1 + \exp[(-2V_n + 1)\beta]}, \tag{2.16}$$

where $a_n = 1$ indicates a prediction that the outcome will occur, and $a_n = 0$ indicates a prediction that the outcome will not occur. The free parameter β corresponds to the inverse softmax temperature and was fit based on data from the behavioral pilot portion of the study.

In summary, we use standard Kalman filtering to infer the parameters of the LDS corresponding to each causal structure. This yields a distribution over associative weights $\mathbf{w}$ for each causal structure $\mathcal{M}$ (Eq. 2.6), which we can use in turn to compute the posterior distribution over all three causal structures (Eq. 2.14). The joint posterior over weights and causal structures is then used to predict the expected outcome V_n (Eq. 2.12) and the corresponding decision a_n (Eq. 2.16). Our model thus makes predictions about computations at two levels of inference: at the level of causal structures (Eq. 2.14) and at the level of associative weights for each structure (Eq. 2.6).

Parameter estimation

The model has four free parameters: the mean w_0 and variance σ_w^2 of the Gaussian prior from which the weights are assumed to be drawn, the variance of the process noise τ^2, and the inverse temperature β used in the logistic transformation from predictive posterior expectation to choice probability. Intuitively, w_0 corresponds to the initial weight given to a cue or context prior to observing any outcome, σ^2 corresponds to the level of uncertainty in this initial estimate, τ^2 reflects how much we expect the weights to change over time, and β reflects choice stochasticity. We estimated a single set of parameters based on choice data from the behavioral pilot version of the study using maximum log-likelihood estimation (Figure 2.4B, grey circles). We preferred this approach over estimating a separate set of parameters for each subject as it tends to avoid overfitting, produces more stable estimates, and has been frequently used in previous studies [58,112,116,117]. Additionally, since none of these pilot subjects participated in the fMRI portion of the study, this procedure ensured that the parameters used in the final analysis were not overfit to the choices of the scanned subjects. For the purposes of fitting, the model was evaluated on the same stimulus sequences as the pilot subjects, including both training and test trials. Each block was simulated independently—i.e., the parameters of the model were reset to their initial values prior to the start of training. The likelihood of the subject's response on a given trial was estimated according to the choice probability given by the model on that trial. Maximum likelihood estimation was computed using MATLAB's fmincon function with 25 random initializations. The bounds on the parameters were $w_0 \in [0,1]$, $\sigma_w^2 \in [0,10]$, $\tau^2 \in [0,1]$, and $\beta \in [0,10]$, all initialized with noninformative uniform priors.

The fitted values of the parameters are shown in Table 2.1. All other parameters were set to the same values as described in Gershman [104]. We used these parameter estimates to construct model-based regressors for the fMRI analysis. For the behavioral analysis, we trained and tested the model on each block separately and reported the choice probabilities on test trials, averaged across conditions (Figure 2.4).

2.3.4 Alternative models

Single causal structure

We evaluated versions of the model that contain only a single causal structure (M_1, M_2, or M_3). Theories corresponding to each of these structures have been advanced as potential explanations of the role of context in associative learning [104], making them plausible candidates for explaining the data. We fit the four free parameters w_0, σ_w^2, τ^2, and β separately for each of the three single-structure models (Table 2.1, M_1, M_2, and M_3).

Simple reinforcement learning

We also evaluated a simple reinforcement learning (RL) model that learns a separate value $V_n(x, c)$ for each cue-context pair (x, c). In particular, after observing the outcome r_n on trial n, the expectation for the observed cue-context pair (x_n, c_n) is updated as:

$$V_{n+1}(x_n, c_n) = V_n(x_n, c_n) + \eta(r_n - V_n(x_n, c_n)) \tag{2.17}$$

where x_n is the cue that was presented on trial n (that is, $x_{nx_n} = 1$), and η is the learning rate. The values of all other cue-context pairs remain unchanged (that is, $V_{n+1}(i,j) = V_n(i,j) \ \forall (i,j) \neq (x_n, c_n)$). Choices were modeled using the same logistic sigmoid transformation as before (Eq. 2.16). All values were initialized to V_0.

This model has three free parameters: the learning rate $\eta \in [0, 1]$, the inverse softmax temperature $\beta \in [0, 10]$, and the initial values $V_0 \in [0, 1]$, which were fit in the same way as the causal structure learning model (Table 2.1, simple RL).

Reinforcement learning with generalization

Since the simple RL model treats each cue-context pair as a unique stimulus, it always predicts V_0 for previously unseen cue-context pairs. In order to allow generalization to new cue-context pairs, we extended the simple RL model in the following way: if either the cue or the context are unknown, then the model takes the mean value over the unknown quantity as experienced in the current block. In particular, if a cue-context pair (x_n, c_n) has never been experienced, but either the cue x_n or the context c_n has been seen in other cue-context pairs, then the predicted value is computed as:

$$V_n(x_n, c_n) = \frac{\sum_{i=1}^{D} V_n(i, c_n) \times count_n(i, c_n) + \sum_{i=1}^{K} V_n(x_n, i) \times count_n(x_n, i)}{\sum_{i=1}^{D} count_n(i, c_n) + \sum_{i=1}^{K} count_n(x_n, i)} \tag{2.18}$$

where $count_n(x, c)$ is the number of times cue-context pair (x, c) has appeared in trials $1..n$. If neither the cue nor the context were seen before, the predicted value is V_0. Note that this extension pertains to predictions only; for learning, the value of new cue-context pairs is still initialized at V_0. The free parameters η, β, and V_0 were fit in the same way as the simple RL model (Table 2.1, RL + generalization).

Reinforcement learning with clustering

We also implemented a structure learning model proposed by Collins & Frank [43] that clusters cues and contexts into latent states, also referred to as "task sets". Reinforcement learning is then performed over this clustered latent state space rather than the original space of cue-context pairs. Structure learning in this case refers to the process of building the latent state space, whereas in our model, we define structure learning as the process of arbitrating among an existing set of candidate causal structures.

Clustering is performed independently for cues and contexts, such that cues are assigned to one set of clusters and contexts are assigned to a different set of clusters. Cluster membership is tracked probabilistically by $P(z_x|x_n)$

and $P(z_c|c_n)$ for cues and contexts, respectively. For a new cue x_n on trial n, the cluster assignment probabilities are initialized as:

$$P(z_x|x_n, \mathbf{h}_{1:n-1}) \propto \begin{cases} \sum_{i=1}^{D} P(z_x|i, \mathbf{h}_{1:n-1}) & \text{for existing clusters } z_x \\ \alpha & \text{for a new cluster } z_x \end{cases} \tag{2.19}$$

where α is a concentration parameter and $P(z_x|i, \mathbf{h}_{1:n-1}) = 0$ for unseen cues i. This is similar to a Chinese restaurant process [101] and implements a "rich-get-richer" dynamic that favors popular clusters which already have many cues assigned to them. Note that a new cluster is created for each new cue. Cluster membership $P(z_c|c_n, \mathbf{h}_{1:n-1})$ for new contexts c_n is initialized in the same way.

A prediction is generated by selecting the maximum *a priori* cue cluster $z'_x = \arg\max_{z_x} P(z_x|x_n, \mathbf{h}_{1:n-1})$ and context cluster $z'_c = \arg\max_{z_c} P(z_c|c_n, \mathbf{h}_{1:n-1})$, and using the value $V_n(z'_x, z'_c)$ in the logistic sigmoid transformation (Eq. 2.16).

Once an outcome r_n is observed, the posterior distributions over clusters are updated according to:

$$P(z_x|x_n, \mathbf{h}_{1:n}) \propto P(z_x|x_n, \mathbf{h}_{1:n-1}) P(r_n|z_x, z'_c, \mathbf{h}_{1:n-1}) \quad \forall z_x \tag{2.20}$$

$$P(z_c|c_n, \mathbf{h}_{1:n}) \propto P(z_c|c_n, \mathbf{h}_{1:n-1}) P(r_n|z'_x, z_c, \mathbf{h}_{1:n-1}) \quad \forall z_c \tag{2.21}$$

where the likelihood $P(r_n|z_x, z_c, \mathbf{h}_{1:n-1})$ is estimated based on the cluster values $V_n(z_x, z_c)$ and Eq. 2.16.

Finally, the maximum *a posteriori* cue cluster $z''_x = \arg\max_{z_x} P(z_x|x_n, \mathbf{h}_{1:n})$ and context cluster $z''_c = \arg\max_{z_c} P(z_c|c_n, \mathbf{h}_{1:n})$ are selected based on the updated posterior distributions, and their value is updated according to:

$$V_{n+1}(z''_x, z''_c) = V_n(z''_x, z''_c) + \eta(r_n - V_n(z''_x, z''_c)) \tag{2.22}$$

This model has four free parameters: the learning rate $\eta \in [0, 1]$, the inverse softmax temperature $\beta \in [0, 10]$, the concentration parameter $\alpha \in [0, 10]$, and the initial values $V_0 \in [0, 1]$ that were fit in the same way as the other models (Table 2.1, RL + clustering).

2.3.5 Model comparison

In order to select models for analyzing the neural data, we performed random effects Bayesian model selection [237] based on the behavioral data from the fMRI session. Since we fit the free parameters using data from the pilot portion of the study, there was no need to penalize for overfitting, so we computed the model evidence as the probability of the subject's choices in the fMRI portion of the study (i.e., the model likelihood). This is equivalent to assuming that the probability density of the parameters is concentrated on the parameter settings obtained from the pilot data. The model evidences were then used to compute the protected exceedance probability (PXP) for each model, which indicates the probability that the given model is the most frequently occurring model in the population.

2.3.6 fMRI data acquisition

Scanning was carried out on a 3T Siemens Magnetom Prisma MRI scanner with the vendor 32-channel head coil (Siemens Healthcare, Erlangen, Germany) at the Harvard University Center for Brain Science Neuroimaging. A T1-weighted

high-resolution multi-echo magnetization-prepared rapid-acquisition gradient echo (ME-MPRAGE) anatomical scan [297] of the whole brain was acquired for each subject prior to any functional scanning (176 sagittal slices, voxel size = 1.0 x 1.0 x 1.0 mm, TR = 2530 ms, TE = 1.69 - 7.27 ms, TI = 1100 ms, flip angle = 7°, FOV = 256

mm). Functional images were acquired using a T2*-weighted echo-planar imaging (EPI) pulse sequence that employed multiband RF pulses and Simultaneous Multi-Slice (SMS) acquisition[201,90,316]. In total, 9 functional runs were collected per subject, with each run corresponding to a single task block (84 interleaved axial-oblique slices per whole brain volume, voxel size = 1.5 x 1.5 x 1.5 mm, TR = 2000 ms, TE = 30 ms, flip angle = 80°, in-plane acceleration (GRAPPA) factor = 2, multi-band acceleration factor = 3, FOV = 204 mm). The initial 5 TRs (10 seconds) were discarded as the scanner stabilized. Functional slices were oriented to a 25 degree tilt towards coronal from AC-PC alignment. The SMS-EPI acquisitions used the CMRR-MB pulse sequence from the University of Minnesota. Four subjects failed to complete all 9 functional runs due to technical reasons and were excluded from the analyses. Three additional subjects were excluded due to excessive motion.

2.3.7 fMRI preprocessing

Functional images were preprocessed and analyzed using SPM12 (Wellcome Department of Imaging Neuroscience, London, UK). Each functional scan was realigned to correct for small movements between scans, producing an aligned set of images and a mean image for each subject. The high-resolution T1-weighted ME-MPRAGE images were then co-registered to the mean realigned images and the gray matter was segmented out and normalized to the gray matter of a standard Montreal Neurological Institute (MNI) reference brain. The functional images were then normalized to the MNI template (resampled voxel size 2 mm isotropic), spatially smoothed with a 8 mm full-width at half-maximum (FWHM) Gaussian kernel, high-pass filtered at 1/128 Hz, and corrected for temporal autocorrelations using a first-order autoregressive model.

2.3.8 Univariate analysis

We defined two general linear models (GLMs) based on the causal structure learning model (GLM 1 and GLM 2) and two GLMs based on the clustering model (GLM 3 and GLM 4). Every GLM had two impulse regressors convolved with the canonical hemodynamic response function (HRF) on all training trials: a stimulus regressor

at trial onset, and a feedback regressor at feedback onset. For every GLM, the feedback regressor included two parametric modulators that differed across the GLMs. The parameric modulators were not orthogonalized. In addition, all GLMs included six motion regressors and a constant regressor for baseline activity.

For all group-level analyses, we report t-contrasts with single voxels thresholded at $p < 0.001$ and whole-brain cluster family-wise error (FWE) correction applied at significance level $\alpha = 0.05$. Anatomical regions of the peak voxels were labeled using the Automated Anatomical Labeling (AAL2) atlas[295,240], the SPM Anatomy Toolbox[84], and the CMA Harvard-Oxford atlas[67]. All voxel coordinates are reported in Montreal Neurological Institute (MNI) space.

GLM 1

The rationale behind GLM 1 was to look for brain regions that might be responsible for inferring the causal structure (i.e. structure learning, Eq. 2.14) versus inferring the associative weights (i.e. associative learning, Eq. 2.6).

At feedback onset on trial n, a region that updates the posterior over causal structures would exhibit a signal that is correlated with the magnitude of the discrepancy between the prior $P(\mathcal{M}|\mathbf{h}_{1:n-1})$ and the posterior $P(\mathcal{M}|\mathbf{h}_{1:n})$. We quantified this discrepancy by the Kullback-Leibler (KL) divergence:

$$KL_{structures} = D_{KL}[P(\mathcal{M}|\mathbf{h}_{1:n})||P(\mathcal{M}|\mathbf{h}_{1:n-1})] = \sum_{\mathcal{M}} P(\mathcal{M}|\mathbf{h}_{1:n}) \log_2 \frac{P(\mathcal{M}|\mathbf{h}_{1:n})}{P(\mathcal{M}|\mathbf{h}_{1:n-1})}. \tag{2.23}$$

Similarly, a region involved in updating the associative weights would show activity correlated with the discrepancy between the weight prior and the weight posterior (i.e. the probability distribution over the weights before and after the update). Since the model keeps track of the weights for all structures, we reasoned that a region involved in associative weight updating would show activity that is correlated with the KL divergence between the joint prior and the joint posterior over the weights for all structures, which can be factored into:

$$KL_{weights} = KL_{weights\_M1} + KL_{weights\_M2} + KL_{weights\_M3} \tag{2.24}$$

where each summand represents the KL divergence between the posterior and the prior over the weights for the respective causal structure. The KL divergence for M_1 is given by:

$$\begin{aligned} KL_{weights\_M1} &= D_{KL}[P(\mathbf{w}|\mathbf{h}_{1:n}, M_1)||P(\mathbf{w}|\mathbf{h}_{1:n-1}, M_1)] \\ &= \int_{\mathbf{w}} P(\mathbf{w}|\mathbf{h}_{1:n}, M_1) \log_2 \frac{P(\mathbf{w}|\mathbf{h}_{1:n}, M_1)}{P(\mathbf{w}|\mathbf{h}_{1:n-1}, M_1)} \mathrm{d}\mathbf{w} \\ &= \frac{1}{2 \ln 2} \left[\mathrm{tr}(\Sigma_{n-1}^{-1}\Sigma_n) + (\hat{\mathbf{w}}_{n-1} - \hat{\mathbf{w}}_n)^{\mathrm{T}} \Sigma_{n-1}^{-1} (\hat{\mathbf{w}}_{n-1} - \hat{\mathbf{w}}_n) - D + \ln \left(\frac{\det \Sigma_{n-1}}{\det \Sigma_n} \right) \right] \end{aligned} \tag{2.25}$$

where D denotes the number of weights, Σ_n denotes the posterior covariance on trial n, and dividing by $\ln 2$ converts the result to bits. Eq. 2.25 follows from the fact that the weights are normally distributed (Eq. 2.7). $KL_{weights\_M2}$ and $KL_{weights\_M3}$ were computed analogously.

We used $KL_{structures}$ and $KL_{weights}$ as parametric modulators for the feedback regressor. We were primarily interested in $KL_{structures}$ as it reflects the structure learning update. Previous work [204] suggests that orthogonalizing it with respect to $KL_{weights}$ would not make a difference for the beta coefficients for $KL_{structures}$, while at the same time it would complicate the analysis of $KL_{weights}$. Therefore we did not orthogonalize the parametric modulators with respect to each other nor with respect to the feedback regressor. In order to look for signals specifically related to structure updating above and beyond associative weight updating, we computed the contrast $KL_{structures} - KL_{weights}$.

GLM 2

Another possibility is that only the weights of the most likely causal structure are updated. This approximation resembles the way in which the clustering model only updates the value of the maximum *a posteriori* clusters. GLM 2 is defined in the same way as GLM 1, except that only the weight update for the maximum *a posteriori* causal structure $M' = \arg\max_M P(M|\mathbf{h}_{1:n})$ is included:

$$KL_{weights} = KL_{weights\_M'} \tag{2.26}$$

GLM 3

GLM 3 was based on the clustering model. Analogously to GLM 1, the purpose was to look for regions responsible for updating the cluster assignments (i.e. structure learning, in the sense used by Collins & Frank [43]; Eq. 2.21) versus updating the cluster values (i.e. associative learning, Eq. 2.22).

Similarly to GLM 1, we quantified cluster updating as the KL divergence between the posterior (Eq. 2.21) and the prior (Eq. 2.19) over cluster assignments, conditioned on the cue-context pair (x_n, c_n). Since clusterings for cues and contexts are independent, this can be factored as a sum of the KL divergences for cues and contexts:

$$KL_{clusters} = KL_{cue\_clusters} + KL_{context\_clusters} \tag{2.27}$$

$$= D_{KL}[P(z_x|x_n, \mathbf{h}_{1:n})||P(z_x|x_n, \mathbf{h}_{1:n-1})] + D_{KL}[P(z_c|c_n, \mathbf{h}_{1:n})||P(z_c|c_n, \mathbf{h}_{1:n-1})] \tag{2.28}$$

$$= \sum_{z_x} P(z_x|x_n, \mathbf{h}_{1:n}) \log_2 \frac{P(z_x|x_n, \mathbf{h}_{1:n})}{P(z_x|x_n, \mathbf{h}_{1:n-1})} + \sum_{z_c} P(z_c|c_n, \mathbf{h}_{1:n}) \log_2 \frac{P(z_c|c_n, \mathbf{h}_{1:n})}{P(z_c|c_n, \mathbf{h}_{1:n-1})} \tag{2.29}$$

Associative updating was quantified by the (cluster) prediction error (Eq. 2.22):

$$CPE = r_n - V_n(z''_x, z''_c) \tag{2.30}$$

We used $KL_{clusters}$ and CPE as parametric modulators for the feedback regressor, not orthogonalized. As in GLM 1, we were primarily interested in $KL_{clusters}$ as a proxy for the structure learning update, and we therefore computed the contrast $KL_{clusters} - CPE$.

GLM 4

As a control, we also included a GLM for the clustering model that was identical to the GLM used to analyze EEG data in Collins & Frank[44]. It had the clustering model prediction error $CPE = r_n - V_n(z''_x, z''_c)$ (Eq. 2.30) and the simple (or "flat") RL prediction error $FPE = r_n - V_n(x_n, c_n)$ (Eq. 2.17) as parametric modulators at feedback onset, not orthogonalized. We then computed the contrast $CPE - FPE$ in order to find brain regions that encode value updating specific to the clustering model.

GLM comparison

We used random effects Bayesian model selection[237] to compare GLMs based on how well they fit whole brain neural activity. While we did not expect our GLMs to account for the activity of all voxels, we did not select *a priori* regions of interest (ROIs), and therefore had no reason to exclude any particular voxels from the analysis. We approximated the log model evidence as -0.5 * BIC, where BIC is the Bayesian information criterion, which we computed using the residual variance of the GLM fits. The BIC quantifies how closely the GLM matches the neural activity of a given subject, while adding a penalty proportional to the number of regressors in the GLM to account for overfitting. Bayesian model selection then produced a PXP for each GLM, which is the probability that this is the most frequently occurring GLM in the population.

2.3.9 Multivariate analysis

Representational similarity analysis

We used representational similarity analysis (RSA) to identify candidate brain regions that might encode the full posterior distribution over causal structures (Eq. 2.14) in their multivariate activity patterns [161]. On a given trial, we expected Bayesian updating to occur when the outcome of the subject's prediction is presented at feedback onset (i.e. whether they were correct or incorrect). We therefore sought to identify brain regions that represent the posterior $P(M|\mathbf{h}_{1:n})$ at feedback onset.

In order to identify regions with a high representational similarity match for the posterior, we used an unbiased whole-brain "searchlight" approach. For each voxel of the entire volume, we defined a spherical ROI (searchlight) of 4-mm radius [160] centered on that voxel, excluding voxels outside the brain (equivalently, radius = 2.6667 voxels, or up to 81 voxels in each searchlight). For each subject and each searchlight, we computed a 180×180 representational dissimilarity matrix $\mathbf{R}$ (the neural RDM) such that the entry in row i and column j is the cosine distance between the neural activity patterns on training trial i and training trial j:

$$\mathbf{R}_{ij} = \mathbf{R}_{ji} = 1 - \cos\theta_{ij} = 1 - \frac{\mathbf{a}_i \cdot \mathbf{a}_j}{|\mathbf{a}_i|\,|\mathbf{a}_j|} \tag{2.31}$$

where θ_{ij} is the angle between the 81-dimensional vectors $\mathbf{a}_i$ and $\mathbf{a}_j$ which represent the instantaneous neural activity patterns at feedback onset on training trials i and j, respectively, in the given searchlight for the given subject. Neural activations entered into the RSA were obtained using a GLM with distinct impulse regressors convolved with the HRF at trial onset and feedback onset on each trial (test trials had regressors at trial onset only). The neural activity of a given voxel was thus simply its beta coefficient of the regressor for the corresponding trial and event. Since the matrix is symmetric and $\mathbf{R}_{ii} = 0$, we only considered entries above the diagonal (i.e. $i < j$). The cosine distance is equal to 1 minus the normalized correlation (i.e. the cosine of the angle between

the two vectors), which has been preferred over other similarity measures as it better conforms to intuitions about similarity both for neural activity and for probability distributions[33].

Similarly, we computed an RDM (the model RDM) such that the entry in row i and column j is the cosine distance between the posterior on training trial i and training trial j, as computed by model simulations using the stimulus sequences experienced by the subject on the corresponding blocks.

If neural activity in the given searchlight encodes the posterior, then the neural RDM should resemble the model RDM: trials on which the posterior is similar should have similar neural representations (i.e. smaller cosine distances), while trials on which the posterior is dissimilar should have dissimilar neural representations (i.e. larger cosine distances). This intuition can be formalized using Spearman's rank correlation coefficient between the model RDM and the neural RDM ($n = 180 \times 179/2 = 16110$ unique pairs of trials in each RDM). A high coefficient implies that pairs of trials with similar posteriors tend show similar neural patterns while pairs of trials with dissimilar posteriors tend to show dissimilar neural patterns. Spearman's rank correlation is a preferred method for comparing RDMs over other correlation measures as it does not assume a linear relationship between the RDMs[161]. Thus for each voxel and each subject, we obtained a single Spearman's ρ that reflects the representational similarity match between the posterior and the searchlight centered on that voxel.

In order to aggregate these results across subjects, for each voxel we Fisher z-transformed the resulting Spearman's ρ from all 20 subjects and performed a t-test against 0. This yielded a group-level t-map, where the t-value of each voxel indicates whether the representational similarity match for that voxel is significant across subjects. We thresholded single voxels at $p < 0.001$ and corrected for multiple comparisons using whole-brain cluster FWE correction at significance level $\alpha = 0.05$. We report the surviving clusters and the t-values of the corresponding voxels (Figure 2.6A).

Since the posterior tends to be similar on trials that are temporally close to each other, as well as on trials from the same block, we computed two control RDMs: a "time RDM" in which the distance between trials i and j is $|t_i - t_j|$, where t_i is the difference between the onset of trial i and the start of its corresponding block; and a "block RDM" in which the distance between trials i and j is 0 if they belong to the same block, and 1 otherwise.

Each Spearman's ρ's was then computed as a partial rank correlation coefficient between the neural RDM and the model RDM, controlling for the time RDM and the block RDM. This rules out the possibility that our RSA results reflect within-block temporal autocorrelations that are unrelated to the posterior.

Temporal autocorrelation is a concern when performing RSA, because it can bias the results[3,32,72]. This concern is partially alleviated by using betas extracted from a GLM with a separate impulse regressor on each trial, in addition to controlling for the time RDM and the block RDM. Furthermore, most of the entries in the RDMs are for pairs of trials across different runs, where temporal autocorrelations are not an issue. While this does not perfectly address the autocorrelation problem, the subsequent classification analysis and its link to behavior (described below) validate the ROIs identified by the RSA in a way that is not confounded by temporal autocorrelations in the BOLD signal.

We performed the same analysis for the clustering model. We looked for brain regions with a high representational similarity match with the joint posterior distribution over stimuli and clusters $P(z_x, x, z_c, c)$, which we computed as:

$$P(z_x, x, z_c, c) = P(z_x|x)P(x)P(z_c|c)P(c) \tag{2.32}$$

The cluster assignments $P(z_x|x)$ and $P(z_c|c)$ were computed as in Eq. 2.21. The priors $P(x)$ and $P(c)$ on a given trial were computed as the average number of times cue x and context c (respectively) have been encountered so far. Since this definition of the priors is somewhat ad hoc, we also performed the analysis assuming uniform $P(x)$ and $P(c)$, which makes the posterior equal to the conditional posterior over cluster assignments:

$$P(z_x, z_c|x, c) = P(z_x|x)P(z_c|c) \tag{2.33}$$

Information mapping

Performing RSA using the spatially smoothed functional images has the advantage of producing spatially continuous activation clusters that are consistent across subjects and easy to interpret. However, smoothing discards the fine-grained spatial structure of the signal [160], which could contain rich information about variables involved in structure learning. Thus, we chose to perform classification on the unsmoothed images, but using ROIs selected from the smoothed images. This allows us to maximize the sensitivity of the classifier while accommodating between-subject variability in anatomical locations of the ROIs. Since the posterior closely tracks the block condition, we expect voxels that encode the posterior to be informative about the block condition. In order to identify such voxels, we employed a whole-brain searchlight classification approach based on the unsmoothed neural data. We used the Searchmight toolbox [226] on betas from a GLM identical to the GLM used for the RSA, except that it was performed on functional images that did not undergo smoothing in the preprocessing step. As in the RSA, for each voxel in the whole-brain volume, we defined a 4-mm searchlight centered on that voxel. For each subject and each searchlight, we trained a separate linear discriminant analysis (LDA) classifier with a shrinkage estimator for the covariance matrix [226] to predict the block condition (irrelevant context, modulatory context, or irrelevant cue) based on neural activity at feedback onset on the training trials. We only considered trials $6 \ldots 20$ since both subject performance and the posterior over causal structures plateaued around trial 6, and hence we did not expect trials $1 \ldots 5$ to be informative. Thus there were $15 \times 9 = 135$ data points, each consisting of up to 81 voxels. We trained and evaluated the classifier using stratified 3-fold cross-validation with whole blocks: there were three data partitions, and each partition contained one block of each condition, chosen at random (for a total of $15 \times 3 = 45$ data points per partition). Including entire blocks in the partitions was necessary due to the temporal autocorrelation of the fMRI signal within each block, which could overfit the classifier to individual blocks rather than block conditions. Since each block was part of one validation set, this allowed us to obtain performance for each data point by a classifier that had not seen that data point, nor any other data points from the same block. Classification accuracy was computed based on the validation sets and was assigned to the

center voxel of the searchlight. Thus for each subject, we obtained an accuracy map for the entire brain volume.

Correlating neural activity with behavior

We sought to leverage the strengths of both the searchlight RSA and the searchlight classifier by combining the group-level ROIs identified by the RSA with the subject-specific accuracy maps identified by the classifier in order to predict subject behavior on the test trials. We conjectured that noise in the neural representation of the posterior might vary systematically across subjects. Subjects with noisier representations would produce test phase choices that are less consistent with the causal structure learning model. Furthermore, this noise would be reflected in the classifier performance, with noisier representations resulting in lower classification accuracy.

In order to test this prediction, we took the peak classification accuracy within each ROI identified by the RSA and correlated it with the log likelihood of the subject's test choices (averaged across blocks). We then applied Bonferroni correction to the resulting set of p-values. If a set of voxels encodes the posterior, then its classification accuracy should predict how well the subject's choices during the test phase conform to the predictions of the causal structure learning model. By restricting the analysis to ROIs identified by the RSA, this approach yields interpretable results on the group level, while simultaneously taking into account idiosyncrasies in the precise locus of the neural representation of the posterior for each subject. Furthermore, since results from both the RSA and the classifier were based on training trials only, circularity in the analysis is avoided.

Results

2.4 Structure learning accounts for behavioral performance

The behavioral results replicated the findings of Gershman [104] using a within-subject design. Subjects from both the pilot and the fMRI portions of the study learned the correct stimulus-outcome associations relatively quickly, with average performance plateauing around the middle of training (Figure 2.3). Average accuracy during the second half of training was $91.2 \pm 2.5\%$ ($t_9 = 16.8, p < 10^{-7}$, one-sample t-test against 50%) for the pilot

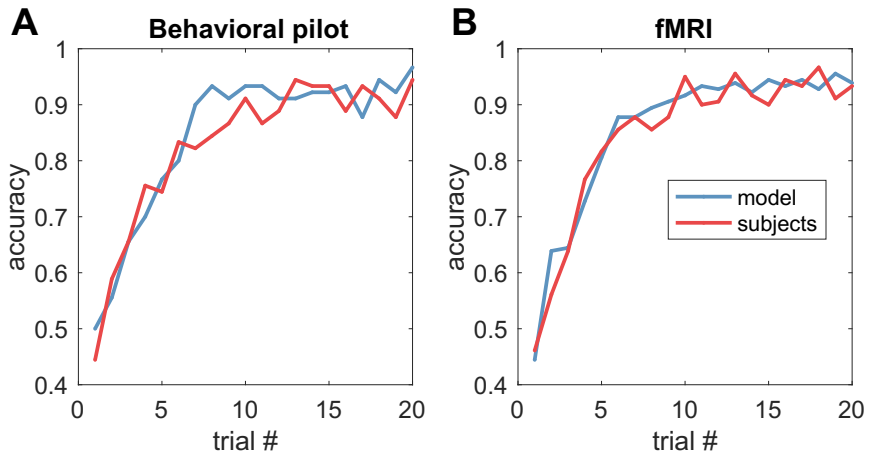


Figure 2.3: Learning curves during training
Performance during training for (A) behavioral pilot subjects ($N = 10$), and (B) fMRI subjects ($N = 20$), averaged across subjects and blocks.

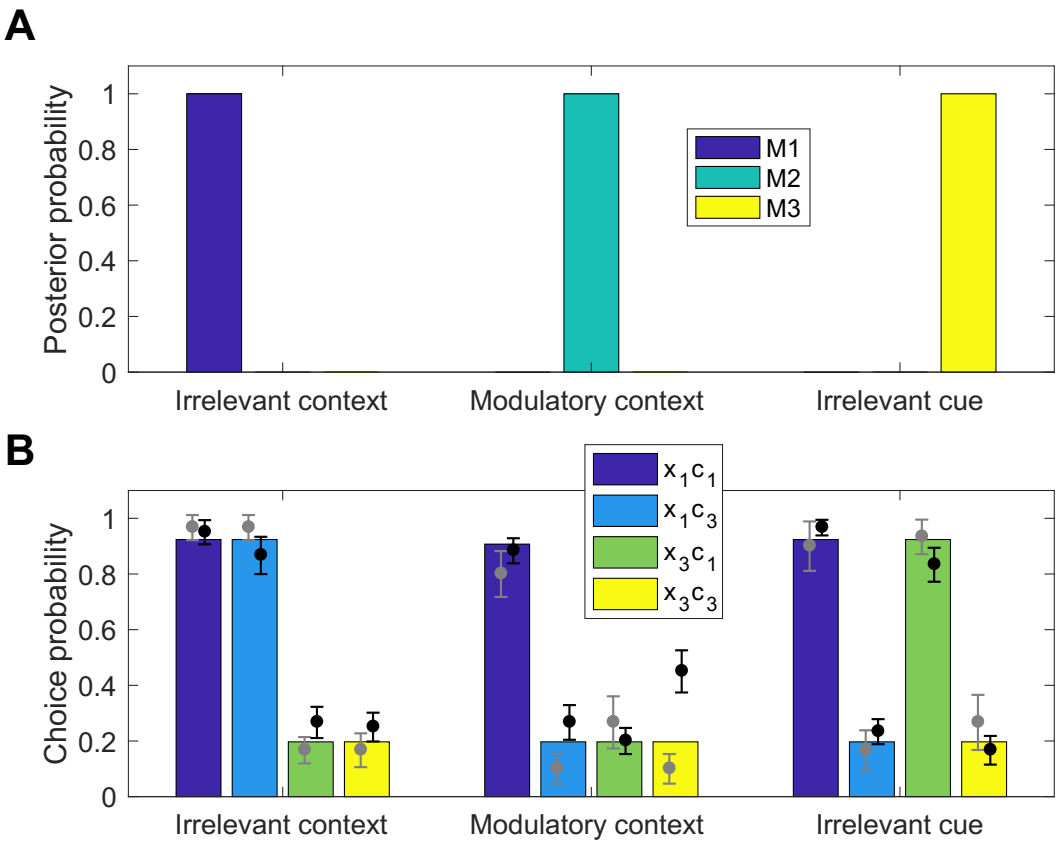


Figure 2.4: Generalization on the test trials

(A) Posterior probability distribution over causal structures in each condition at the end of training. Each block was simulated independently and the posterior probabilities were averaged across blocks of the same condition.

(B) Choice probabilities on the test trials for subjects in the pilot (grey circles) and fMRI (black circles) portions of the study, overlaid over model choice probabilities (colored bars). Each color corresponds to a particular combination of an old (x_1) or new (x_3) cue in an old (c_1) or new (c_3) context. Error bars represent within-subject standard errors of the mean[48]

subjects, and $92.7 \pm 1.7\%$ ($t_{19} = 25.0, p < 10^{-15}$, one-sample t-test against 50%) for the scanned subjects, well above chance.

Importantly, both groups exhibited distinct patterns of generalization on the test trials across the different conditions, consistent with the results of Gershman [104] (Figure 2.4B). Without taking the computational model into account, these generalization patterns already suggest that subjects learned something beyond simple stimulus-response mappings. On blocks during which context was irrelevant (Figure 2.4B, irrelevant context), subjects tended to predict that the old cue x_1, which caused sickness in both c_1 and c_2, would also cause sickness in the new context c_3 (circle for x_1c_3), even though they had never experienced c_3 before. The new cue x_3, on the other hand, was judged to be much less predictive of sickness in either context ($t_{38} = 9.51, p < 10^{-10}$, paired t-test). On modulatory context blocks, subjects appeared to treat each cue-context pair as a unique stimulus independent from the other pairs (Figure 2.4B, modulatory context). On these blocks, subjects judged that the old cue is predictive of sickness in the old context significantly more compared to the remaining cue-context pairs ($t_{38} = 9.01, p < 10^{-10}$, paired t-test). On blocks during which the cue was irrelevant, (Figure 2.4B, irrelevant cue), subjects guessed that the old context c_1, which caused sickness for both cues x_1 and x_2, would also cause sickness for the new cue x_3 (circle for x_3c_1), but that the new context c_3 would not cause sickness ($t_{38} = 11.1, p < 10^{-12}$, paired t-test).

These observations were consistent with the predictions of the causal structure learning model. Using parameters fit with data from the behavioral pilot version of the study, the model quantitatively accounted for the generalization pattern on the test trials choices of subjects in the fMRI portion of the study (Figure 2.4B; $r = 0.97, p < 10^{-7}$). As expected, the stimulus-outcome contingencies induced the model to infer a different causal structure in each of the three conditions (Figure 2.4A), leading to the distinct response patterns on the simulated test trials.

Of the alternative models, only the clustering model provided an equally compelling account of the generalization pattern on the test trials (Table 2.1, RL + clustering; $r = 0.98, p < 10^{-7}$). Bayesian model comparison (Table 2.1) based on all of the subjects' choices favored both the causal structure learning model and the clus-

Table 2.1: Model comparison favors the full causal structure learning model (M1, M2, M3) and the clustering model (RL + clustering). The free parameters were fit based on choice data from the pilot version of the study (Figure 2.4B, grey circles). Protected exceedance probabilities (PXPs) were computed based on the fMRI portion of the study. Pearson's correlations were computed based on test phase choices from the fMRI portion of the study (Figure 2.4B, black circles). RL, reinforcement learning.

Model	free parameters	PXP	Pearson's r
M_1, M_2, M_3	$\sigma_w^2 = 0.0157, \beta = 2.6849,$ $\tau^2 = 7.5724 \times 10^{-5}, w_0 = 0.2382$	0.8840	$r = 0.97, p < 10^{-7}$
M_1	$\sigma_w^2 = 0.0079, \beta = 1.9441,$ $\tau^2 = 1.3340 \times 10^{-5}, w_0 = 0.2641$	0.0013	$r = 0.61, p = 0.0347$
M_2	$\sigma_w^2 = 0.0570, \beta = 2.5302,$ $\tau^2 = 1.3049 \times 10^{-9}, w_0 = 0.3282$	0.0017	$r = 0.73, p = 0.0076$
M_3	$\sigma_w^2 = 0.0111, \beta = 1.5085,$ $\tau^2 = 3.8610 \times 10^{-11}, w_0 = 0.1722$	0.0003	$r = 0.59, p = 0.0447$
simple RL	$\eta = 0.8888, \beta = 2.3983, V_0 = 0.3188$	0.0027	$r = 0.73, p = 0.0076$
RL + generalization	$\eta = 0.5579, \beta = 2.3777, V_0 = 0.2175$	0.0004	$r = 0.88, p = 0.0002$
RL + clustering	$\eta = 0.8397, \beta = 2.4166,$ $\alpha = 1.3963, V_0 = 0.2624$	0.1096	$r = 0.98, p < 10^{-7}$

tering model more strongly than the alternatives. For comparison, generalization was markedly worse when the hypothesis space was restricted to a single causal structure: the correlation coefficients were $r = 0.61$ for the irrelevant context structure ($M_1; p = 0.03$), $r = 0.73$ for the modulatory context structure ($M_2; p = 0.008$), and $r = 0.59$ for the irrelevant cue structure ($M_3; p = 0.04$). As expected, performance of the simple RL model was comparable to M_2, since they both treat each cue-context pair as a unique stimulus. RL with generalization showed an improvement in the generalization pattern ($r = 0.88, p = 0.0002$), however it was not as good as the causal structure learning model nor the clustering model, and its PXP indicated that it is unlikely to be the most prevalent model in the population. Therefore we restricted our subsequent analysis of the neural data to the causal structure learning and clustering models.

Table 2.2: GLM comparison cannot disambiguate between the causal structure learning model (GLM 1: M1, M2, M3) and the clustering model (GLM 3: RL + clustering). Protected exceedance probabilities (PXPs) were based on whole-brain activity from all trials. MAP, maximum *a posteriori*.

GLM	model	parametric modulators	PXP
GLM 1	M1, M2, M3	$KL_{structures}$, $KL_{weights}$ (sum)	0.4009
GLM 2	M1, M2, M3	$KL_{structures}$, $KL_{weights}$ (MAP)	0.0993
GLM 3	RL + clustering	$KL_{clusters}$, CPE	0.4006
GLM 4	RL + clustering, simple RL	CPE, FPE	0.0992

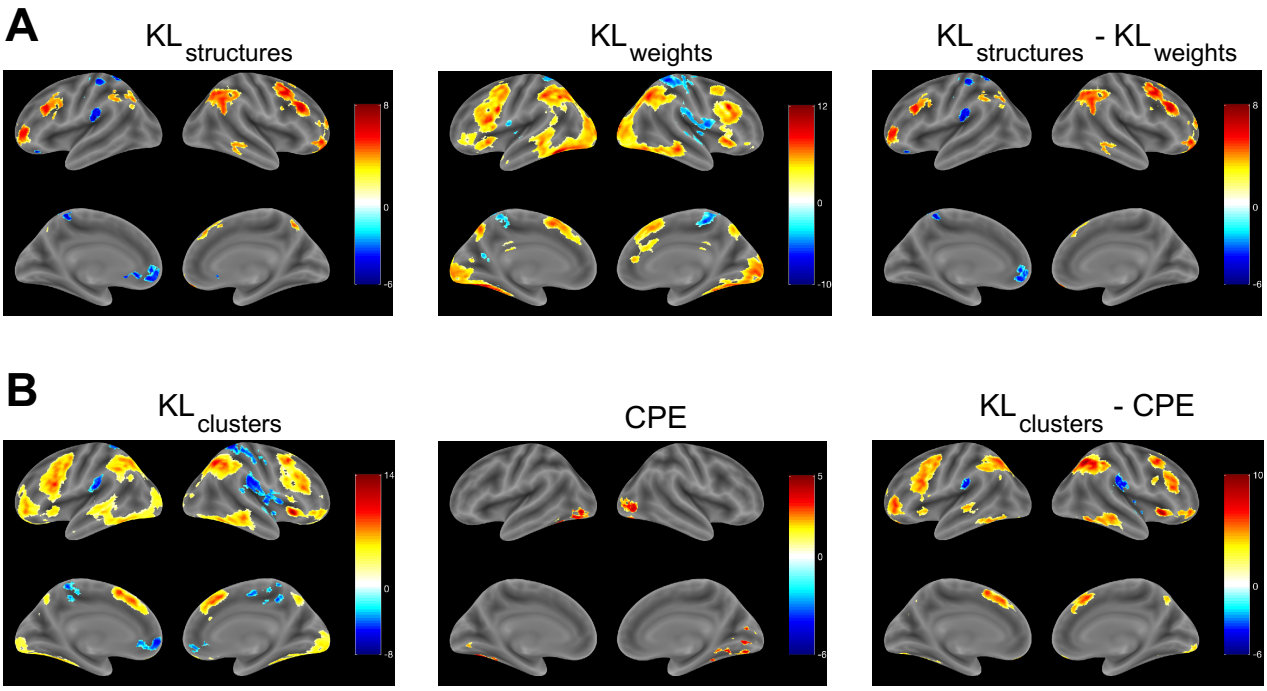


Figure 2.5: Distinct Neural Signatures of Structure Learning and Associative Learning
Statistical maps for GLM 1 (A) and GLM 3 (B), using a threshold of $p < 0.001$, whole-brain cluster FWE corrected at $\alpha = 0.05$. The color scales represent *t*-values.
(A) Regions tracking Bayesian updating of beliefs about causal structures (left), Bayesian updating of beliefs about associative weights for all structures (middle), and the contrast between the two (right).
(B) Regions tracking Bayesian updating of beliefs about cluster assignments (left), the prediction error for the currently active clusters (middle), and the contrast between the two (right).

Table 2.3: GLM 1: $KL_{structures} - KL_{weights}$. Brain regions in which the BOLD signal tracks Bayesian updating of causal structures above and beyond Bayesian updating of associative weights (corresponding to Figure 2.5A, right panel). The anatomical label and the MNI coordinates are based on the voxel with the maximum *t*-statistic from each cluster. Single voxels were thresholded at $p < 0.001$ and whole-brain cluster FWE correction was applied at significance level $\alpha = 0.05$. Regions were labeled using the Automated Anatomical Labeling (AAL2) atlas. IFG, inferior frontal gyrus. MNI, Montreal Neurological Institute. BA, Brodmann area.

Sign	Brain region	BA	Extent	*t*-value	MNI coord.
Positive	Middle frontal gyrus (R)	9	2218	8.377	44 12 50
	Angular gyrus (R)	39	1968	6.351	56 -60 30
	IFG pars triangularis (L)	48	698	6.341	-38 26 26
	Anterior orbital gyrus (R)	11	1430	6.236	24 64 -14
	Middle frontal gyrus (L)	46	809	6.137	-40 56 6
	Cerebellum (L)		423	5.584	-40 -62 -46
	Superior frontal gyrus, dorsolateral (R)	8	332	5.383	2 38 48
	Inferior temporal gyrus (R)	20	278	5.192	62 -44 -12
	Inferior parietal gyrus (L)	7	768	5.044	-34 -66 46
	Cerebellum (L)		211	5.001	-28 -72 -28
Negative	Rolandic operculum (L)	48	352	-6.463	-46 -28 20
	IFG pars orbitalis (L)	11	308	-6.207	-24 32 -10
	Superior parietal gyrus (L)	5	554	-5.831	-18 -48 60

Table 2.4: GLM 3: $KL_{clusters} - CPE$. Brain regions in which the BOLD signal tracks Bayesian updating of cluster assignments above and beyond associative updating (corresponding to Figure 2.5B, right panel). Notation and procedures as in Table 2.3.

Sign	Brain region	BA	Extent	*t*-value	MNI coord.
Positive	IFG pars triangularis (L)	48	2996	10.548	-42 14 30
	Angular gyrus (R)	39	2433	10.541	32 -60 46
	Anterior insula (R)	47	506	10.275	30 22 -4
	Superior frontal gyrus, dorsolateral (R)	8	1625	9.032	2 28 44
	IFG pars opercularis (R)	44	2471	8.438	58 18 34
	Middle frontal gyrus (L)	10	1309	7.943	-38 58 8
	Cerebellum (L)		2200	7.839	-4 -80 -30
	Anterior orbital gyrus (R)	11	1036	7.688	34 48 -16
	Inferior parietal gyrus (L)	7	2220	6.858	-32 -56 48
	Inferior temporal gyrus (R)	37	903	6.792	44 -56 -10
	Medial orbital gyrus (L)	11	274	6.676	-18 44 -18
	Cerebellum (R)	18	1112	6.172	24 -86 -22
	Middle temporal gyrus (L)	21	420	5.886	-58 -28 -4
	Precuneus (R)	7	195	4.797	6 -68 48
Negative	Superior temporal gyrus (R)	48	212	-6.435	44 -12 -4
	Superior temporal gyrus (L)	48	277	-5.693	-48 -28 18
	Superior temporal gyrus (R)	48	520	-5.258	56 -32 22

2.5 Separate brain regions support structure learning and associative learning

We sought to identify brain regions in which the blood oxygenation level dependent (BOLD) signal tracks beliefs about the underlying causal structure. In order to condense these multivariate distributions into scalars, we computed the Kullback-Leibler (KL) divergence between the posterior and the prior distribution over causal structures on each training trial ($KL_{structures}$, Eq. 2.23), which measures the degree to which structural beliefs were revised after observing the outcome. Specifically, we analyzed the fMRI data using a general linear model (GLM) which included $KL_{structures}$ as a parametric modulator at feedback onset. We reasoned that activity in regions involved in learning causal structure would correlate with the degree of belief revision.

Since we were interested in regions that correlate with learning on the level of causal structures rather than their associative weights, we included the KL divergence between the posterior and the prior distribution over associative weights ($KL_{weights}$, Eq. 2.25). These weights encode the strength of causal relationships between cues, contexts and outcomes separately for each causal structure. Including $KL_{weights}$ as an additional parametric modulator at feedback onset would capture any variability in the signal related to weight updating and allow us to isolate it from the signal related to structure updating.

Our Kalman filter implementation of structure learning assumes that the agent performs full Bayesian inference, which necessitates simultaneous updating of the weights for all causal structures, regardless of the agent's beliefs about the causal structures. However, a biologically/cognitively plausible implementation might incorporate certain heuristics, such as devoting less computational resources to updating the weights for causal structures that are less likely[213]. To account for this possibility, we compared two GLMs that included both $KL_{structures}$ and $KL_{weights}$ as parametric modulators and feedback onset, but differed in the way $KL_{weights}$ was computed. In GLM 1, $KL_{weights}$ was computed as the sum of the KL divergences for all causal structures (Eq. 2.24), consistent with our implementation which devotes the same amount of computational resources to updating the weights for all structures. In GLM 2, $KL_{weights}$ was computed only for the maximum *a posteriori* (MAP) structure on the current trial (Eq. 2.26). This is consistent with an implementation that only updates the weights for the most likely

structure, analogously to the clustering model which only updates the value for the MAP cluster assignments.

We used on an analogous GLM (GLM 3) to identify brain regions that correlate with structural updates and associative updates based on the clustering model. The structure learned by the clustering model corresponds to the cluster assignments of the individual cues and contexts, so we reasoned that the structure learning update would elicit a signal proportional to the KL divergence between the posterior and the prior over cluster assignments ($KL_{clusters}$, Eq 2.29). Associative learning in the clustering model corresponds to updating the value of the currently active cue cluster and context cluster, which can be quantified by the (cluster) prediction error (*CPE*, Eq. 2.30). As a control, we included another GLM (GLM 4) for the clustering model, which was based on the GLM used in Collins & Frank [44]. It had the *CPE* and the simple (or "flat") RL prediction error (*FPE*) as parametric modulators at feedback onset.

Bayesian model comparison favored GLM 1 and GLM 3 over the other GLMs (Table 2.2). The high PXP of GLM 1 compared to GLM 2 suggests that the most prevalent causal structure learning model in the population is the one that keeps updating the weights for all structures equally, as predicted by our Kalman filter implementation. The high PXP of GLM 3 compared to GLM 4 favors a model that performs RL over clusters alone, rather than one which performs RL over clusters in addition to RL over individual cues and contexts. We therefore report group-level contrasts for GLM 1 and GLM 3 only.

We were interested in identifying regions that track structure learning above and beyond associative learning. For GLM 1, this corresponds to the contrast $KL_{structures} - KL_{weights}$ (Figure 2.5A, right panel; Table 2.3). We report clusters that show a significant positive effect (i.e., a stronger correlation with $KL_{structures}$ than with $KL_{weights}$) after thresholding single voxels at $p < 0.001$ and applying whole-brain cluster FWE correction at significance level $\alpha = 0.05$ (minimum cluster extent = 211). The contrast highlighted a bilateral network of frontoparietal regions. We observed activations in inferior PPC, with a cluster in right angular gyrus and a smaller one spanning the left angular gyrus and left inferior parietal gyrus (IPG). We also found activations in lateral PFC, with a large cluster in right medial frontal gyrus (MFG), extending ventrally into inferior frontal gyrus (IFG) pars triangularis and dorsally into superior frontal gyrus (SFG), as well as a smaller cluster in IFG pars triangularis in the left hemi-

sphere. We also found bilateral activations in rostrolateral prefrontal cortex (RLPFC), extending into the orbital surface in the right hemisphere. Significant activations were also found on the medial surface of right SFG and in the occipito-temporal part of the right inferior temporal gyrus . Notably, even though the regions that correlated with $KL_{structures}$ (Figure 2.5A, left panel) were highly overlapping with the regions that correlated with $KL_{weights}$ (Figure 2.5A, middle panel), the fact that most of those regions survived in the contrast implies that the signal in these areas cannot be explained by associative learning alone, suggesting a dissociable network of regions that supports causal structure learning.

For GLM 3, the contrast of interest was $KL_{clusters} - CPE$ (Figure 2.5B, right panel; Table 2.4; minimum cluster extent = 195). This revealed a frontoparietal network of regions with a high degree of overlap with the $KL_{structures} - KL_{weights}$ contrast from GLM 1. We found bilateral clusters in inferior PPC (IPG and angular gyrus), lateral PFC (IFG and MFG), and RLPFC. Unlike GLM 1, there were also bilateral clusters in inferior temporal gyrus, middle temporal gyrus, anterior insula, and medial SFG. As in GLM 1, the regions that correlated with $KL_{clusters}$ (Figure 2.5B, left panel) were largely present in the contrast as well, suggesting that their activity tracks a structure update signal that cannot be accounted for by associative updating alone.

2.6 Multivariate representations of the posterior over causal structures

If the brain performs Bayesian inference over causal structures, as our data suggest, then we should be able to identify regions that contain representations of the full posterior distribution over causal structures $P(M|\mathbf{h}_{1:n})$ (Eq. 2.14). We thus performed a whole-brain "searchlight" representational similarity analysis [161] using searchlights of 4-mm radius [160]. For each subject, we centered the spherical ROI on each voxel of the whole-brain volume and computed a representational dissimilarity matrix (RDM) using the cosine distance between neural activity patterns at feedback onset for all pairs of trials (see Materials and Methods). Intuitively, this RDM reflects which pairs of trials look similar and which pairs of trials look different according to the neural representations in the local neighborhood around the given voxel. We then used Spearman's rank correlation to compare this neural

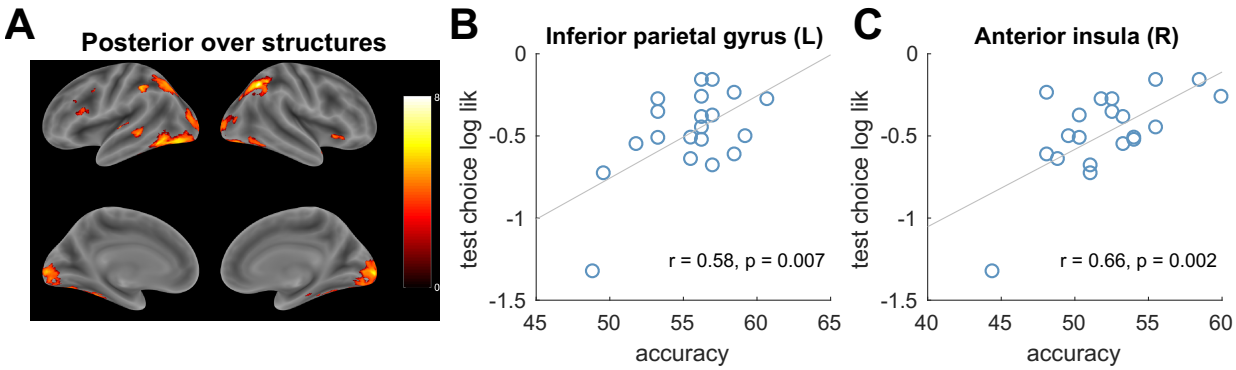


Figure 2.6: Neural Signature of the Posterior over Causal Structures
(A) Statistical map showing regions with a high representational similarity match for the posterior over causal structures at feedback onset ($p < 0.001$, whole-brain cluster FWE corrected at $\alpha = 0.05$). The color scales represent t-values.
(B, C) Between-subject correlation between peak classification accuracy on the training trials and the average log likelihood of the subject's choices on the test trials, according to the causal structure learning model. Significant correlations were found for left inferior parietal gyrus (B) and right anterior insula (C), after Bonferroni correction with adjusted $\alpha = 0.05/6 = 0.0083$. r, Pearson's correlation coefficient.

Table 2.5: Brain regions with a high representational similarity match between neural patterns at feedback onset and the posterior over causal structures (corresponding to Figure 2.6A). The voxel with the maximum t-statistic from each cluster is also reported. Single voxels were thresholded at $p < 0.001$ and whole-brain cluster FWE correction was applied at significance level $\alpha = 0.05$. Notation as in Table 2.3. r is the between-subject Pearson's correlation coefficient between peak classification accuracy within the ROI and test choice log likelihood. * - significant after Bonferroni correction with adjusted $\alpha = 0.05/6 = 0.0083$.

Sign	Brain region	BA	Extent	t-value	MNI coord.	r	Unadjusted p-value	Adjusted p-value
Positive	Angular gyrus (R)	7	1347	8.505	34 -58 42	0.03	0.908	1.000
	Inferior temporal gyrus, calcrine fissure, and surrounding cortex (L)	37	6940	8.112	-50 -58 -18	0.25	0.283	0.864
	Inferior parietal gyrus (L)	7	1279	6.751	-46 -38 44	0.58	0.007*	0.042*
	Superior temporal gyrus (L)	48	270	6.162	-46 -20 6	0.28	0.231	0.793
	Anterior insula (R)	47	253	5.779	44 20 -6	0.66	0.002*	0.010*
	IFG pars triangularis (L)	45	480	4.636	-56 26 24	0.30	0.195	0.729

RDM with a model RDM based on the posterior over causal structures. If a given ROI encodes the posterior, then pairs of trials on which the posterior is similar would also show similar neural representations, while pairs of trials on which the posterior is different would show differing neural representations. This corresponds to a positive rank correlation between the model and the neural RDMs.

For each subject and each voxel, we thus obtained a Spearman's rank correlation coefficient, reflecting the similarity between variability in activity patterns around that voxel and variability in the posterior over causal structures (the representational similarity match). To aggregate these results at the group level for each voxel, we then performed a one-sample t-test against 0 with the Fisher z-transformed Spearman's ρ from all subjects. The resulting t-values from all voxels were used to construct a whole-brain t-map, which was thresholded and corrected for multiple comparisons in the same way as the GLMs (Figure 2.6A and Table 2.5; minimum cluster extent = 253). Each t-value in this map quantifies how likely it is that the given voxel exhibits a positive representational similarity match with the posterior across the population. This revealed some of the same frontoparietal regions identified by the structure learning contrast (Figure 2.5A, right panel), including bilateral inferior PPC (angular gyrus and the neighboring IPG), and left IFG pars triangularis. We also found a large bilateral occipito-temporal cluster spanning the primary visual areas, fusiform gyrus, and inferior temporal gyrus. Additional matches were found in right anterior insula and left superior temporal gyrus.

We then performed the same analysis for the clustering model, using the posterior over clusters and stimuli $P(z_x, x, z_c, c)$ (Eq. 2.32). We did not find any voxels that survive multiple comparisons correction. This was also true when we used the conditional posterior over cluster assignments $P(z_x, z_c|x, c)$ (Eq. 2.33). Taken together, these results favor a causal structure learning account of the data, and point to a network of regions for maintaining beliefs about causal structure, which get updated on a trial-by-trial basis by a distinct but overlapping network of frontoparietal regions.

2.7 Neural representations of the posterior predict subsequent choices

In order to confirm that ROIs identified by the RSA truly contain representations of the posterior over causal structures, we next sought to use the neural activity in those regions to predict subject behavior. We employed a whole-brain searchlight classification approach based on the unsmoothed functional images (see Materials and Methods). For each subject, this produced an accuracy map that quantifies the amount of information about the block condition contained in the local neighborhood of each voxel. To test the hypothesis that a particular ROI identified by the RSA encodes the posterior at the group level, we took the peak classification accuracy within that ROI and correlated it with the average log likelihood of the subject's responses during the test phase. Notice that since the RSA and the classifier results were based on training trials only, there is no circularity in this analysis. The resulting Pearson's correlation coefficients are shown in Table 2.5. After applying Bonferroni correction for all six RSA ROIs, we found a significant positive correlation in right anterior insula (adjusted $p < 0.01$, Figure 2.6B) and left inferior PPC (adjusted $p < 0.05$, Figure 2.6C).

We based this analysis on the assumption that there is some endogenous noise in the neural representation of the posterior[217,131,174]. This noise would disrupt the close correspondence between the block condition and the posterior, resulting in lower classification accuracy. Thus the accuracy assigned to each voxel can be interpreted as the fidelity with which a particular subject represents the posterior in the searchlight around that voxel. The voxel with the highest accuracy within an ROI is also the best candidate for representing the posterior. At the same time, noise in the posterior would give rise to discrepancies between the subject's behavior and the model predictions, which are based on a noise-free representation of the posterior. Since this noise would likely be overshadowed by noise in the BOLD signal on any single trial, we turned to the group level, where any systematic variability in the noise of the posterior across subjects should be manifested as systematic variability in the both the classification accuracy and the likelihood of the subject's test phase choices. While this analysis assumes subjects are using the structure learning model, subjects using a different model could show the same pattern as those having a noisy or posterior: their classification accuracy would be low due to the incorrect representation, and

their test choice log likelihood would be low due to the discrepancy between their model and the structure learning model. That is, subjects should produce test phase choices in accordance with the posterior to the extent that they use the structure learning model and they have a less noisy neural representation of the posterior. Thus the fact that two of the ROIs matching the similarity pattern of the posterior also show this relationship with behavior provides strong evidence that these regions encode the full posterior distribution over causal structures in their multivariate patterns of activity.

2.8 Discussion

Behavioral evidence suggests that humans and animals infer both the structure and the strength of causal relationships [130,159,194,104]. Using functional brain imaging in humans, the current study provides neural evidence that the formation of stimulus-outcomes associations is guided by the inferred structure of the environment. The neural data support the existence of a learning mechanism operating over structural representations that is distinct from the mechanism operating over associative representations, thus reifying the computationally hypothesized division of labor. Our univariate analysis identified areas that were sensitive to belief updates about structure, including inferior PPC, lateral PFC, and RLPFC. In addition, representational similarity analysis revealed an overlapping network of brain areas that appear to represent the full posterior distribution over causal structures, with activity in two of those regions – inferior PPC and anterior insula – showing a significant correlation with subsequent subject responses.

Our behavioral data were equally well explained by an alternative structure learning model put forward by Collins & Frank [43], which implicated some of the same brain areas in relation to belief updates about structure. This is somewhat remarkable, considering that their model offers a different interpretation of structure learning, namely that different stimulus dimensions (cues and contexts) are grouped into latent clusters and that associations are formed based on those latent clusters. In a sense, this offers greater flexibility than our model as it does not assume any pre-existing knowledge of the relationships between different stimulus dimensions, and it allows

for a theoretically unbounded number of latent clusters. Indeed, their model and related latent cause models [109] address the question of how structure might emerge in the first place. In contrast, our model endows the agent with an *a priori* set of relations between stimulus dimensions and outcomes, which are assumed to be innate or acquired through previous experience. This allows for more flexibility in the functional form of the associations, such as the summation of values across different stimulus dimensions, something widely believed to be important for capturing classic animal learning phenomena such as blocking, overshadowing, and overexpectation [234,266]. The fact that a largely overlapping network of regions tracks belief updates about structure for both models, despite their differences, suggests a generic neural mechanism for discovering the latent structure of the world that is agnostic to the particular structure learning interpretation. The limitations of the current study preclude any strong conclusions favoring one model over the other, and thus further work will be required to disentangle the behavioral and neural predictions of the two models.

A notable feature of our data is that inferior PPC appears to encode the full posterior over structures as well as its corresponding Bayesian update. Previous authors [255] have linked this area with the integration of bottom-up multimodal input and top-down predictions from frontal areas. O'Reilly et al. [218] found that angular gyrus encodes the discrepancy between the prior and the posterior distribution over outcomes in a statistical model based on task history. Gläscher et al. [117] found a signature of the state prediction error in intraparietal sulcus and lateral PFC, implicating those regions in computing the discrepancy between the current model and the observed state transitions. Our results resonate with these findings and fit with the idea that inferior PPC acts as a cross-modal hub that integrates prior knowledge with incoming information.

One candidate region where such top-down predictions might originate is lateral PFC, an area with strong functional connectivity with the inferior parietal lobule [301,23]. Previous studies on cognitive control [154,155,10] have proposed the existence of a functional gradient in lateral PFC, with more anterior regions encoding representations of progressively higher levels of abstraction. Donoso et al. [78] found evidence that RLPFC performs inference over multiple counterfactual strategies by tracking their reliability, while IFG pars triangularis is responsible for switching to one of those strategies if the current one is deemed unreliable. Work on hierarchical reinforce-

ment learning [12,95] extends the notion of a functional hierarchy in lateral PFC to the acquisition of abstract latent rules that guide stimulus-outcome associations. If causal structures are likened to alternative strategies or latent rules, then these results may relate to our finding that RLPFC and IFG track structure updating, and that IFG shows a representational similarity match with the posterior (although we were unable to link this representation with behavior, possibly due to the weak signal). Another region where top-down predictions might originate is orbitofrontal cortex (OFC), which has been linked with the representation of a posterior distribution over latent causes [33] and is thought to represent a cognitive map of task space [310,250]. Consistent with this theory, we found a signature of the Bayesian update signal in right OFC, although our multivariate analysis did not implicate this region in the representation of the posterior.

One puzzling aspect of our results is that activity in anterior insula – a region traditionally implicated in affective processing – appears to encode the full posterior over structures, and yet it does not correlate with the update signal. This might relate to previous work by Schapiro et al. [247] who found that stimuli belonging to the same latent state elicit greater representational similarity in IFG and anterior insula, implicating these regions in some form of latent state inference. Further work will be required to investigate the functional role of anterior insula in relation to structure learning.

An important question that remains open is how structure learning might be implemented in biologically realistic neural circuits. Tervo et al. [284] noted the parallels between the hierarchical architecture of cortical circuits and the hierarchical nature of structure learning, with empirical evidence suggesting that different layers of the hierarchy tend to be associated with separate cortical circuits. If the brain indeed performs Bayesian inference over causal structures, this raises the more fundamental question of how ensembles of neurons could represent and perform operations on probability distributions. Different theories have been put forward, ranging from probabilistic population codes to Monte Carlo sampling [228]. Teasing apart the different possible mechanisms would require developing behavioral frameworks that lend themselves to computational modeling and quantitative predictions about the inferred probability distributions [284]. We believe our study is an important step in that direction.

In summary, we used a combination of behavioral, neural and computational techniques to tease apart the neural substrates of structure learning from those of associative learning. Inference over the space of possible structures in the environment recruited frontoparietal regions that have been previously implicated in belief revision and latent state representations, such as inferior PPC, IFG, and RLPFC. Corresponding regions were activated regardless of whether we interpreted structure learning as arbitrating among a set of existing causal structures [104], or as clustering stimuli into latent states [43]. Additionally, our multivariate analysis found a representation of the posterior distribution over structures in inferior PPC and anterior insula that was predictive of subject responding. Together, these results provide strong support for the idea that the brain performs probabilistic inference over latent structures in the environment, enabling inductive leaps that go beyond the given observations.

3

Discovery of Hierarchical Representations for Efficient Planning

3.1 Abstract

We propose that humans spontaneously organize environments into clusters of states that support hierarchical planning, enabling them to tackle challenging problems by breaking them down into sub-problems at various levels of abstraction. People constantly rely on such hierarchical presentations to accomplish tasks big and small – from planning one's day, to organizing a wedding, to getting a PhD – often succeeding on the very first attempt. We formalize a Bayesian model of hierarchy discovery that explains how humans discover such useful abstractions. Building on principles developed in structure learning and robotics, the model predicts that hierarchy discovery should be sensitive to the topological structure, reward distribution, and distribution of tasks in the environment. In five simulations, we show that the model accounts for previously reported effects of environment structure on planning behavior, such as detection of bottleneck states and transitions. We then test the novel predictions of the model in eight behavioral experiments, demonstrating how the distribution of tasks and rewards can influence planning behavior via the discovered hierarchy, sometimes facilitating and sometimes hindering performance. We find evidence that the hierarchy discovery process unfolds incrementally across trials. Finally, we propose how hierarchy discovery and hierarchical planning might be implemented in the brain. Together, these findings present an important advance in our understanding of how the brain might use Bayesian inference to discover and exploit the hidden hierarchical structure of the environment.

3.2 Introduction

Imagine you have a sudden irresistible craving for your favorite ice cream that is only made by a boutique ice cream shop in Lugo, Spain. You must get there as soon as physically possible. What would you do? When faced with this unusual puzzle, most people's first response is that they will look up a flight to Spain. When asked what they would do next, most people say that they would order a taxi to the airport, and when questioned further, that they would walk to the taxi pickup location. Importantly, nobody says or even thinks anything like "I will get up, turn left, walk five steps, etc.", or even worse, "I will contract my left quadricep, then my right one, etc.".

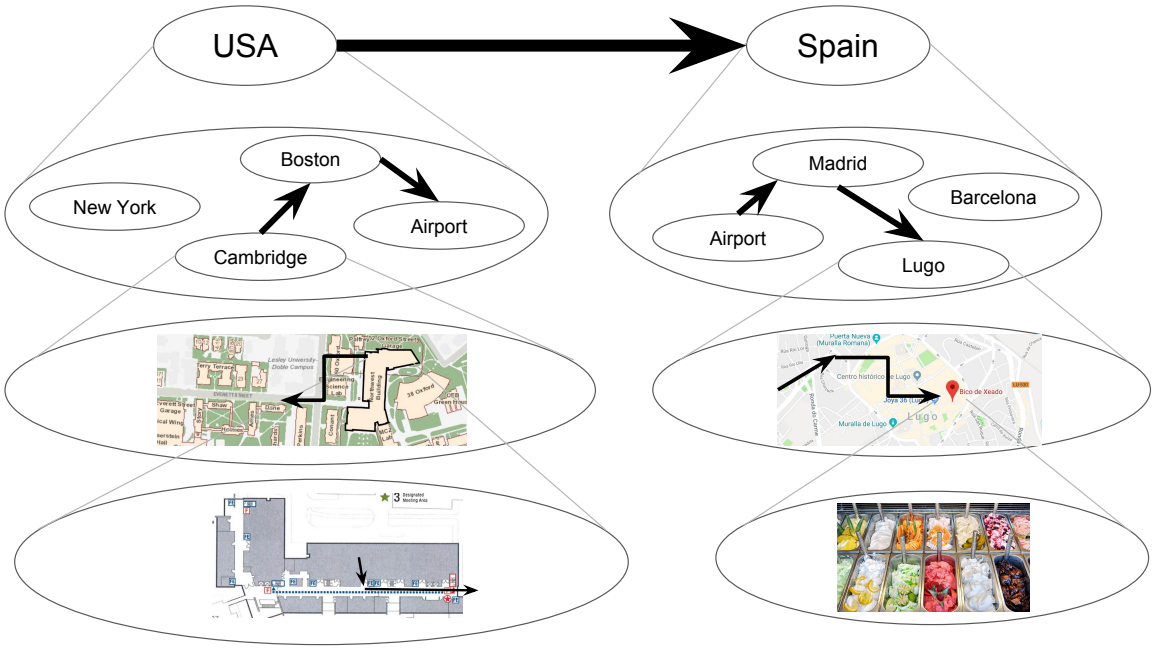


Figure 3.1: Example of hierarchical planning.
How someone might plan to get from their office in Cambridge to their favorite ice cream shop in Lugo, Spain. Circles represent states and arrows represent actions that transition between states. Each state represents a cluster of states in the lower level. Thicker arrows indicate transitions between higher-level states, which often come to mind first.

This example illustrates hierarchical planning (Figure 3.1): people intuitively reason at the appropriate level of abstraction, first sketching out a plan in terms of transitions between high-level *states* (in this case, countries), which is subsequently refined in progressively lower levels and specific steps[307]. This is often done in an online fashion, with details being resolved on-the-fly as the plan is being executed (for example, you would not ponder what snack to buy at the airport before actually getting there). This ability of humans[13] and animals[99] to organize their behavior hierarchically allows them to flexibly pursue distant goals in complex environments, even for novel tasks that they may have never encountered previously.

The study of hierarchical behaviors has deep roots in psychology and neuroscience[173]. Much work has been done to characterize the emergence of such behaviors in humans and animals, often focusing on the acquisition of temporally extended action sequences or *action chunks* that unfold over different time scales. Action chunking occurs after extensive training, involves a specific set of brain regions, and is thought to be essential for pursuing long-term goals and planning[127,262].

Yet in order to leverage action chunks for planning, an agent must also be equipped with a hierarchical representation of the environment, with clusters of states or *state chunks* representing parts of the world at different levels of abstraction. In our ice cream example, the current country, city, neighborhood, building, room, and location within the room are all valid representations of the agent's current state and are all necessary in order to plan effectively. Correspondingly, research shows that people spontaneously discover hierarchical structure[307,246], that the discovered hierarchies are consistent with formal definitions of optimality[263], and that people use these hierarchies to plan[13]. Some of these studies have uncovered distinct neural correlates of *high-level states*, which are sometimes referred to as state clusters, communities, contexts, abstract states, or state chunks[246,13]. Research has also begun to uncover the neural correlates of hierarchical planning and action selection[13,236]. Yet despite these advances, the computational mechanisms underlying the discovery of such hierarchical representations remain poorly understood.

In this study, we propose a Bayesian model of hierarchy discovery for planning. Drawing on the structure learning literature[110] and on concepts developed in robotics[91], the model provides a normative account of how

agents with limited cognitive resources should chunk their environment into clusters of states that support efficient planning. The central novel contributions of this paper are both empirical and theoretical. Our main empirical contribution is to show that the distribution of tasks (experiments one through five) and rewards (experiments six and seven) in the environment can influence the inferred state chunks, whereas past studies have focused exclusively on the effects of environment topology. Our main theoretical contribution is to unify these and previous findings under a single normative model that explains why these phenomena occur, and that encompasses a class of process models that could be leveraged to investigate the implementational details of state chunking in the brain.

In simulations one through five, we demonstrate that the model accounts for previously reported behavioral effects of the environment topology, such as detection of transitions across state clusters[246], identification of topological bottlenecks[263], preference for routes with fewer state clusters[263], and slower reaction times to transitions that violate topological structure[181]. Additionally, the model makes specific predictions about the way in which the distributions of tasks and rewards constrain hierarchy discovery, which in turn constrains planning. We test these novel predictions in a series of eight behavioral experiments. Experiment one shows that the distribution of tasks encountered in the environment can induce different state clusters, even when the topological structure of the environment does not promote any particular clustering. Experiment two shows how this in turn could lead to either improved or hampered performance on novel tasks. Experiment three examines the progression of inferred hierarchies as a function of which tasks participants have seen so far and reveals that participants are sensitive to changes in both the uncertainty and the mode of the posterior distribution over hierarchies. Experiments four and five replicate the results of experiments one and two in a fully visible environment, showing that the effects cannot be explained by incorrect inferences about topological structure. Experiment six shows how rewards generalize within state clusters, while experiment seven shows how rewards can induce clusters that constrain planning even in the absence of state clusters. Finally, experiment eight (Supplemental Results) suggests that people explore in a way that maximally reduces uncertainty about the hierarchy, implying that people consider a probability distribution over hierarchies rather than a single hierarchy. Together, these results

provide strong support for a Bayesian account of hierarchy discovery, according to which the brain implicitly assumes that the environment has a hidden hierarchical structure, which is rationally inferred from observations. No existing theory of hierarchy discovery can account for all of these effects (Table 3.2).

3.3 Theoretical background

The question of how agents should make rewarding decisions is the subject of reinforcement learning (RL;[279]). With a long history crisscrossing the fields of psychology, neuroscience, and artificial intelligence, RL has made major contributions to explaining many human and animal behaviors[235], the neural circuits underlying these behaviors[251], and allowing artificial agents to achieve human-level performance on tasks that were previously beyond the capabilities of computers[200]. Approaches rooted in RL therefore offer promising avenues to understanding decision making and planning.

Most RL algorithms assume that that an agent represents its environment as a Markov decision process (MDP), which consists of states, actions and rewards that obey a particular conditional independence structure. At any point in time, the agent occupies a given state, which could denote, for example, a physical location such as a place in room, a physiological state such as being hungry, and abstract state such as whether a subgoal has been achieved, or a complex state such as a conjunction of such states. The agent can transition between states by taking actions such as moving or eating. Some states deliver rewards, and it is assumed that the agent aims to maximize reward. In an MDP, the transitions and rewards are assumed to depend only on the current state and action.

Following previous authors[263,13], we assume for simplicity that states are discrete and that transitions between states are bidirectional and deterministic (although these restrictions could be relaxed; see General Discussion). In this case, states and actions could be represented by an undirected graph G in which the vertices correspond to states and the edges correspond to actions. We will use the graph G in place of the transition function T that is traditionally used to characterize MDPs.

In graph theoretic notation, $G = (V, E)$, where

- V is the set of vertices (or nodes) such that each node $v \in V$ corresponds to a state that the agent can occupy, and
- $E : \{V \times V\}$ is a set of edges such that each edge $(u, v) \in E$ corresponds to an action that the agent can take to transition from state u to state v or vice versa.

In the following analysis, we use the terms *node* and *state* interchangeably. We also treat edges, actions, and transitions as equivalent. For simplicity, we restrict our analysis to unweighted graphs, which is equivalent to assuming that all actions carry the same cost and/or take the same amount of time (our analysis extends straightforwardly to weighted graphs; see General Discussion). We also assume the agent has learned G, which is equivalent to model-based RL in which the agent learns the transition function.

The task of planning to get from a starting state s to a goal state g efficiently can thus be framed as finding the shortest path between nodes s and g in G. This is a well-studied problem in graph theory and the optimal solution in this setup is the breadth first search (BFS) algorithm[47]. BFS works by first exploring the neighbors of s, then exploring the neighbors of those neighbors, and so on until reaching the goal state g. States whose neighbors haven't been explored yet are maintained in an active queue, with states getting removed from one end of the queue as soon as their neighbors are added to the other end. Intuitively, this corresponds to a forward sweep that begins at s and spreads in all directions across the edges until reaching g, akin to the way in which water might spread in a network of pipes. Its simplicity and performance guarantees have made BFS a standard tool for planning in classical artificial intelligence[243].

However, the time and memory that BFS requires is proportional to the number of states (assuming an action space of constant size) since the size of the active queue and the potential length of the plan grow linearly with the size of the state space. Formally, the time and memory complexity of BFS is $O(N)$, where $N = |V|$ is the total number of states. In environments in which real-world agents operate, this number can be huge; in the realm of navigation alone, there could be billions of locations where a person has been or may want to go. Assuming

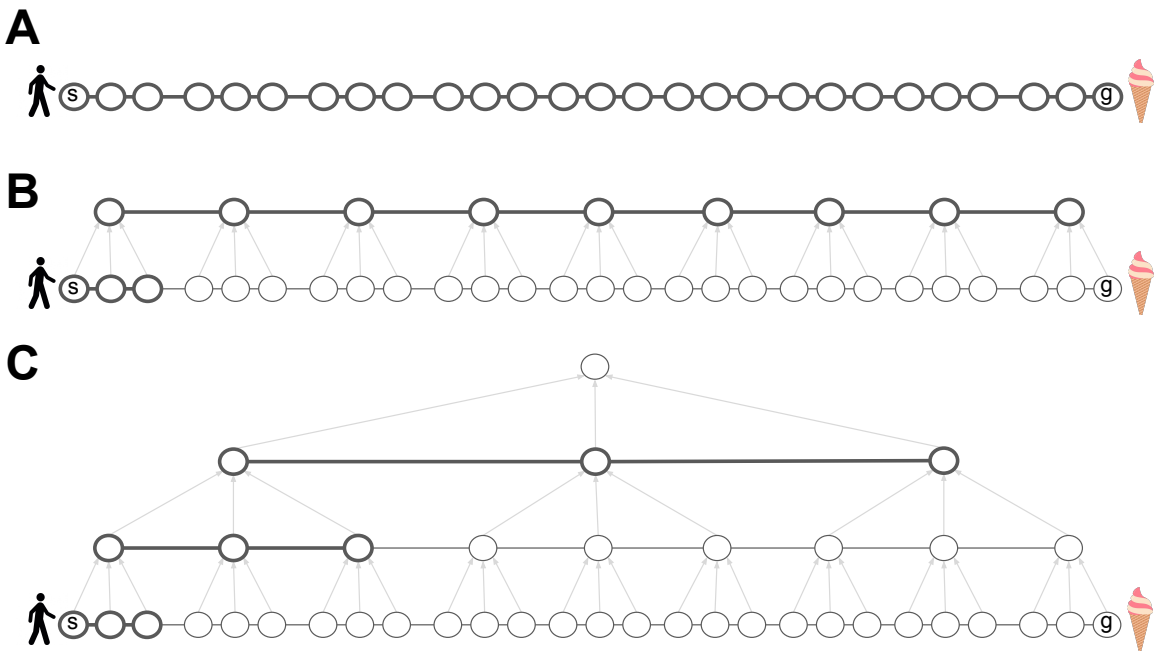


Figure 3.2: Hierarchical representations reduce the computational costs of planning.
A. Planning in the low-level graph G takes at least as many steps as actually executing the plan. All nodes and edges are thick, indicating that they must all be considered and maintained in short-term memory in order to compute the plan.
B. Introducing a high-level graph H alleviates this problem. At any given time during plan execution, the agent only needs to consider the high-level path and the low-level path leading to the next cluster, recomputing the latter on-the-fly. Gray arrows indicate cluster membership.
C. The hierarchy can be extended recursively, further reducing the time and memory requirements of planning.

that online computations such as planning involve systems for short-term storage and symbol manipulation, this far exceeds the working memory capacity of people who can only accommodate a few items (note that we assume the graph is already stored in a different system for long-term storage of relational information, such as the hippocampus)[198]. Furthermore, even without such working memory limitations, artificial agents such as robots would still take a long time to plan the route before they can take the first step. When using a naive or "flat" representation in which the agent plans over low-level actions (for example, individual steps or even joint actuator commands), computing a plan is at least as complicated as actually executing it (Figure 3.2A), and in reality the complexity could be much larger.

To overcome this limitation, work in the field of robotics has led to the development of data structures and algorithms for hierarchical planning[91]. Similar ideas have been put forward in other fields; see General Discussion. The key idea is that an agent can group neighboring states from the flat low-level graph G into state clusters (state chunks), with each cluster represented by a single node in another graph H (the high-level graph), which will be smaller and hence easier to plan in. When tasked to get from state s to state g in G, the agent can first plan in the high-level graph H and then translate this high-level plan into a low-level plan in G. Crucially, after finding the high-level path in H, the agent only needs to plan in the current state cluster in G, that is, it only needs to plan how to get to the next cluster (Figure 3.2B), and then repeat the process in the next cluster, and so on, until reaching the goal state in the final cluster.

This can drastically reduce the working memory requirements of planning, since the agent only needs to keep track of the (much shorter) path in H and the path in the current state cluster in G. Importantly, this also reduces planning time, allowing the agent to begin making progress towards the goal without computing the full path in G – the agent can now follow the high-level plan in H and gradually refine it in G on-the-fly, during execution. This approach can be applied recursively to deeper hierarchies in which high-level states are clustered in turn into even higher-level states, and so on until reaching a single node at the top of the hierarchy that represents the entire environment (Figure 3.2C). Planning using such a hierarchical representation can be orders of magnitude more efficient than "flat" planning, and also accords with our intuitions about how people plan in everyday life.

A particular instantiation of this form of hierarchical planning is the hierarchical breadth first search (HBFS) algorithm, which is a natural extension of BFS (see Methods). It can be shown that with an efficient hierarchy of depth L (that is, consisting of L graphs, with each graph representing clusters in the lower graph), the time and memory complexity of HBFS is $O(\sqrt[L]{N} + L)$ [91]. Thus in a graph with $N = 1,000,000,000$ states and a hierarchy $L = 9$ levels deep, HBFS will only require on the order of 19 memory and time units to compute a plan, compared to 1,000,000,000 time and memory units for BFS, or any other flat planner. Note that while executing the plan would still take $O(N)$ time, HBFS quickly computes the first few actions in the right direction, and can then be applied repeatedly to keep computing the following actions in an online fashion. While it may seem that this hierarchical scheme simply transfers the burden to the long-term storage system, which now needs to remember L graphs instead of one, it can be shown that the storage requirements of an efficient hierarchical representation are $O(N)$ [91], comparable to those of a flat representation of G alone. Following previous authors, [27,263], we restrict our analysis to $L = 2$ levels for simplicity, noting that our approach extends straightforwardly to deeper hierarchies (see General Discussion).

However, in order to reap the benefits of efficient planning, the hierarchical representation necessarily imposes a form of lossy compression. In particular, each successive graph in the hierarchy loses some of the detail present in the lower level graph. This could lead to hierarchical plans that correspond to suboptimal paths in G. For example, in Figure 3.3A, there is a direct edge that can take the agent from starting state s to go state g in a single action. However, since the edge is not represented by the high-level graph H, HBFS will compute a detour through the state cluster w, akin to real-life situations in which people prefer going through a central location rather than taking an unfamiliar shortcut.

Since finding the shortest path will not always be possible, some hierarchies will tend to yield better paths than others. The challenge for the agent is then to learn an efficient hierarchical representation of the environment that facilitates fast planning across many tasks, without placing an overwhelming burden on working memory and long-term memory systems. How agents accomplish this in the real world is the central question of this paper, to which we propose a solution in the following section.

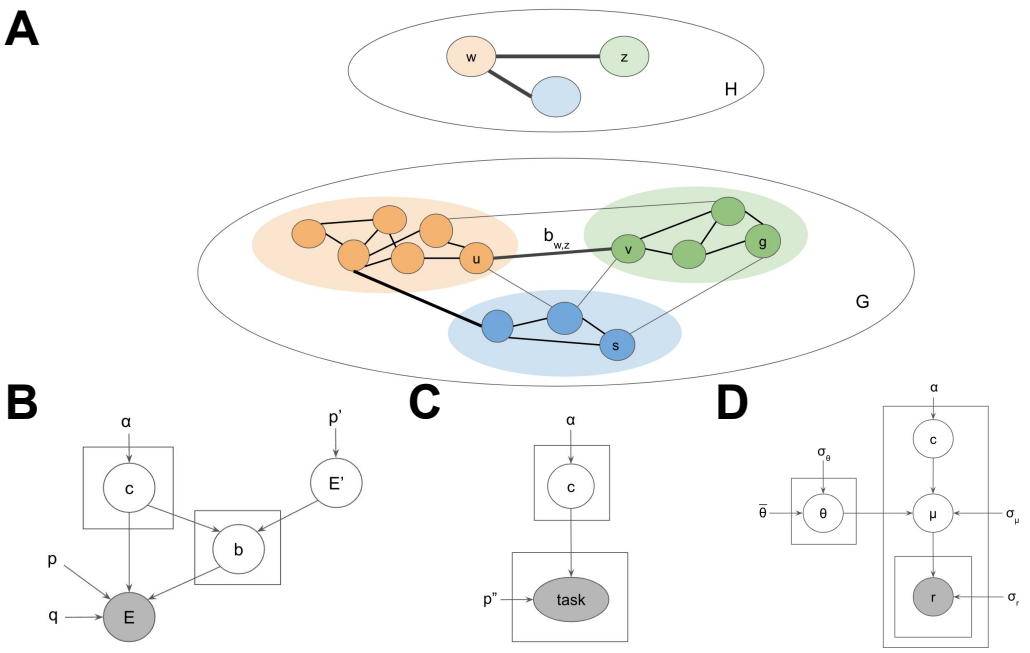


Figure 3.3: Generative model for environments with hierarchical structure.

A. Example low-level graph G and high-level graph H. Colors denote cluster assignments. Black edges are considered during planning. Gray edges are ignored by the planner. Thick edges correspond to transitions across clusters. The transition between clusters w and z is accomplished via the bridge $b_{w,z} = (u, v)$.

B. Generative model defining a probability distribution over hierarchies H and environments G. Circles denote random variables. Rectangles denote repeated draws of a random variable. Arrows denote conditional dependence. Gray variables are directly observed by the agent. Uncircled variables are constant. c, cluster assignments; p', graph density of H; E', edges in H; E, edges in G; b, bridges connecting the clusters; p, within-cluster graph density in G; q, cross-cluster graph density penalty in G. Refer to main text for variable definitions.

C. Incorporating tasks into the generative model. The rest of the generative model is omitted for clarity. p'', cross-cluster task penalty; task = (s, g), task as pair of start-goal states.

D. Incorporating rewards into the generative model. The rest of the generative model is omitted for clarity. $\bar{\theta}$, average reward for G; σ_θ, standard deviation of that average; θ, average cluster rewards; σ_μ, standard deviation around that average; μ, average state rewards; σ_r, standard deviation around that average; r, instantaneous state rewards.

3.4 A Bayesian model of hierarchy discovery

Our proposal assumes two main computational components:

1. An online planner that can flexibly generate plans and select actions on-demand with minimal time and memory requirements.

2. An off-line (and possibly computationally intensive) hierarchy discovery process that, through experience, incrementally builds a representation of the environment that the planner can use.

The focus of this paper is the second component, which must satisfy the constraints imposed by the first component. For the first component, we use HBFS in order to link the hierarchy to behavior, noting that any generic hierarchical planner will make similar predictions. Note that we assume only the online planner is constrained by working memory limitations and time demands – as in the ice cream example, the high-level sketch of a plan is often computed within seconds of the query. In contrast, the hierarchical representation that supports this computation – a rich abstraction of the world, with knowledge of particular locations that belong to cities that belong to countries connected by flights – has been refined through years of experience and is deeply ingrained in long-term memory.

One approach to deriving an optimal hierarchy discovery algorithm would be to define the constraints of the agent, such as memory limitations and computational capacity, its utility function, and the constraints of the environment, such as the expected structure and tasks. Here we adopt an alternative approach, motivated by the literature on structure learning which has been used to successfully account for a wide range of phenomena in the animal learning literatire [110]. The key idea is that the environment is assumed to have a hidden hierarchical structure that is not directly observable, which in turn constrains the observations the agent can experience. The agent can then infer this hidden hierarchical structure based on its observations and use it to plan efficiently. Assuming that some hierarchies are *a priori* more likely than others, this corresponds to a generative model for environments with hierarchical structure, which the agent can invert to uncover the underlying hierarchy based

on its experiences in the environment.

Formally, we represent the observable environment as an unweighted, undirected graph $G = (V, E)$ and the hidden hierarchy as $H = (V', E', c, b, p', p, q)$, where:

- V is the set of low-level nodes or states, corresponding to directly observable states in the environment,
- $E : \{V \times V\}$ is the set of edges, corresponding to possible transitions between states via taking actions,
- V' is the set of high-level nodes or states, corresponding to clusters of low-level states,
- $E' : \{V' \times V'\}$ is the set of high-level edges, corresponding to transitions between high-level states,
- $c : V \to V'$ are the *cluster assignments* linking low-level states to high-level states,
- $b : E' \to E$ are the *bridges* which link high-level transitions back to low-level transitions,
- $p' \in [0, 1]$ is the density of the high-level graph,
- $p \in [0, 1]$ is the within-cluster density of G,
- $q \in [0, 1]$ is the across-cluster density penalty of G.

Here we use the term *graph density* to denote the probability that a pair of nodes are connected by an edge. Together, (V', E') define the high-level graph, which we also refer to as H for ease of notation. Each low-level state u is assigned to a cluster $w = c_u$. Each high-level edge (w, z) has a corresponding low-level edge (the bridge) $(u, v) = b_{w,z}$, such that $c_u = w$ and $c_v = z$ (see Figure 3.3A). Bridges (sometimes referred to as bottlenecks or boundaries) thus specify how different clusters are connected in H. Bridges are the only cross-cluster edges considered by the hierarchical planner; all other edges between nodes in different clusters are ignored, leading to lossy compression that improves planning complexity but could lead to suboptimal paths. In contrast, all edges within clusters are preserved. The purpose of the cluster assignments c is to translate the low-level planning problem in G into an easier high-level planning problem in H. The purpose of the bridges is to translate the solution found

in H back into a low-level path in G. For a detailed description of how HBFS uses the hierarchy to plan, see the Methods section. For simplicity, we only allow a single bridge for each pair of clusters, noting that our approach generalizes straightforwardly to multigraphs (i.e., allowing multiple edges between pairs of nodes in E and E') and maintains its performance guarantees as long as the maximum degree of each node is constant (that is, the size of the action space is $O(1)$).

Informally, an algorithm that discovers useful hierarchies would satisfy the following desiderata:

1. Favor smaller clusters.
2. Favor a small number of clusters.
3. Favor dense connectivity within clusters.
4. Favor sparse connectivity between clusters[49], with the exception of the bridges that connect clusters.

Intuitively, having too few (for example, one) or too many clusters (for example, each state is its own cluster) creates a degenerate hierarchy that reduces the planning problem to the flat scenario, and hence medium-sized clusters are best (desiderata 1 and 2). Additionally, the hierarchy ignores transitions across clusters, which could lead to suboptimal paths generated by the hierarchical planner. It is therefore best to minimize the number of cross-cluster transitions (desiderata 3 and 4). The exception is bridges, which correspond to the links between clusters.

These desiderata can be formalized as a generative model for hierarchies and environments (Figure 3.3B):

$$c \sim CRP(\alpha) \qquad \text{cluster assignments} \tag{3.1}$$

$$p' \sim Beta(1,1) \qquad \text{density in } H \tag{3.2}$$

$$Pr[(w,z) \in E'] = p' \qquad \text{edges in } H \tag{3.3}$$

$$Pr[b_{w,z} = (u,v) \mid (w,z) \in E', c_u = w, c_v = z] = \frac{1}{n_w n_z} \qquad \text{bridges} \tag{3.4}$$

$$p \sim Beta(1,1) \qquad \text{within-cluster density in } G \tag{3.5}$$

$$q \sim Beta(1,1) \qquad \text{cross-cluster density penalty in } G \tag{3.6}$$

$$Pr[(u,v) \in E \mid c_u = c_v] = p \qquad \text{within-cluster edges in } G \tag{3.7}$$

$$Pr[(u,v) \in E \mid c_u \neq c_v, b_{c_u,c_v} \neq (u,v)] = pq \qquad \text{cross-cluster edges in } G \tag{3.8}$$

$$Pr[(u,v) \in E \mid b_{c_u,c_v} = (u,v)] = 1 \qquad \text{bridge edges in } G \tag{3.9}$$

Where $n_w = |\{u : c_u = w\}|$ is the size of cluster w and CRP is the Chinese restaurant process, a nonparametric prior for clusterings [107]. We additionally impose the constraint that no cluster is disconnected, that is, the induced subgraph for each cluster forms a single connected component (see Methods).

Eq. 3.1 fulfills desiderata 1 and 2, with the concentration parameter α determining the trade-off between the two: lower values of α favor few, larger clusters, while higher values of α favor more, smaller clusters. Eq. 3.2 and Eq. 3.3 generate the high-level graph H, with higher values of p' making H more densely connected. Eq. 3.4 generates the bridges by connecting a random pair of nodes (u,v) for each pair of connected clusters (w,z). Eq. 3.5 and Eq. 3.7 fulfil desideratum 3 by generating the low-level edges in G within each cluster, with higher values of p resulting in dense within-cluster connectivity. Eq. 3.6 and Eq. 3.8 fulfill desideratum 4 by generating the low-level edges across clusters, with higher values of q resulting in more cross-cluster edges. Note that $pq < p$ and hence the density of cross-cluster edges will always be lower than the density of within-cluster edges. Finally, Eq. 3.9 ensures that each bridge edge always exists.

This generative model captures the agent's subjective belief about the generative process that gave rise to the environment and the observations. This belief could itself have been acquired from experience or be evolutionarily hardwired. The assumed generative process defines a joint probability distribution $P(G, H) = P(G|H)P(H)$ over the observable graph G and the hidden hierarchical structure H that generated it. Importantly, the generative process is biased to favor graphs G with a particular "clustered" structure. In order to discover the underlying hierarchy H and to use it to plan efficiently, the agent must "invert" the generative model and infer H based on G.

Formally, hierarchy discovery can be framed as performing Bayesian inference using the posterior probability distribution over hierarchies $P(H|G)$:

$$\begin{aligned} P(H|G) &= \frac{P(G|H)P(H)}{P(G)} \\ &\propto P(E|c, b, p, q)P(p)P(q)P(b|E', c)P(E'|p')P(p')P(c) \end{aligned} \tag{3.10}$$

This predicts that neighboring states which are more densely connected will tend to be clustered together. We assess this prediction in simulations one through five.

Note that while our generative model is motivated by normative considerations, it does not comprise a formal analysis of optimality. Such analyses have been reported previously[263] and fail to capture all the effects reported here (Table 3.2). Furthermore, any such analysis would require, at minimum, the assumption of a probability distribution over graphs, tasks, and rewards. Our generative model is equivalent to precisely such a probability distribution, from which we can directly derive a recognition model by framing hierarchy discovery as inference over that probability distribution, thus obviating the need for any additional assumptions.

3.4.1 Task distribution

Previous studies have demonstrated that people discover hierarchies based on topological structure (simulations one through five). However, other factors may also play a role. In particular, the distribution of tasks that an

agent faces in the environment may make some hierarchies less suitable than others[91], independently of the graph topology. For example, if the agent has to frequently navigate from state s to state g in the graph G in Figure 3.3A, then the current hierarchy H would clearly be a poor choice, even if it captures the topological community structure of G well. Since hierarchical planning is always optimal within a cluster, one way to accommodate tasks is to cluster together states that frequently co-occur in the same task.

Casting hierarchy discovery as hidden state inference allows us to formalize this intuition with a straightforward extesion to our model. Following previous authors[264,13], we assume the agent faces a sequence of tasks in G, where each task is to navigate from a starting state $s \in V$ to a goal state $g \in V$. We assume the agent prefers shorter routes. Defining $tasks = \{task_t\}$ and $task_t = (s_t, g_t)$, we can extend the generative model with (Figure 3.3C):

$$p'' \sim Beta(1,1) \qquad \text{cross-cluster task penalty} \qquad (3.11)$$

$$Pr[s_t = u] = \frac{1}{N} \qquad \text{starting states} \qquad (3.12)$$

$$Pr[g_t = v \mid s_t = u] \propto \begin{cases} 1 & \text{if } c_u = c_v \\ p'' & \text{otherwise} \end{cases} \qquad \text{goal states} \qquad (3.13)$$

where c_u and c_v are the cluster assignments of states u and v, and $N = |V|$ is the total number of states. Eq. 3.12 expresses the agent's belief that a task can start randomly in any state. Eq. 3.13 expresses the belief that tasks are less likely to have goal states in a different cluster, with p'' controlling exactly how much less likely that is.

If we denote the observable data as $D = (tasks, G)$, the posterior becomes:

$$
\begin{aligned}
P(H|D) &\propto P(D|H)P(H) \\
&= P(tasks|G,H)P(G|H)P(H) \\
&= \left[\prod_t P(g_t|s_t,p'',G,H)P(s_t|G,H)\right] P(p'')P(G|H)P(H), \qquad (3.14)
\end{aligned}
$$

where the last two terms are the same as in Eq. 3.10.

The model will thus favor hierarchies that cluster together states which frequently co-occur in same tasks. This predicts that, in the absence of community structure, hierarchical planning will occur over clusters delineated by task start and goal states. This is a key novel prediction of our model which we assess in experiments one through five.

3.4.2 Reward distribution

Besides topology and tasks, another factor that may play a role in hierarchy discovery is the distribution of rewards in the environment. In accordance with RL, we assume that each state delivers stochastic rewards and the agent aims to maximize the total reward [279]. Hierarchy discovery might account for rewards by clustering together states that deliver similar rewards. This is consistent with the tendency for humans to cluster based on perceptual features [13] and would be rational in an environment with autocorrelation in the reward structure [269,252,314]. Note that from the perspective of planning alone, rewards or mere perceptual similarity should be irrelevant. This highlights one way in which our model departs from formal notions of optimality. Adhering to the idea of hidden states in structure learning [110], it treats state chunks akin to latent causes which generate similar observations, so it can use those observations to infer the unobservable state chunks.

We can incorporate this intuition into the generative model by positing that states in the same cluster deliver similar rewards (Figure 3.3D):

$$\theta_w \sim \mathcal{N}(\bar{\theta}, \sigma_\theta^2) \qquad \text{average cluster rewards} \tag{3.15}$$

$$\mu_v \sim \mathcal{N}(\theta_{c_v}, \sigma_\mu^2) \qquad \text{average state rewards} \tag{3.16}$$

$$r_{v,t} \sim \mathcal{N}(\mu_v, \sigma_r^2) \qquad \text{rewards} \tag{3.17}$$

where $w \in V'$ and $v \in V$. $\bar{\theta}$ is the average reward of all states, θ_w is the average reward of states in cluster w, μ_v is the average reward of state v, $r_{v,t}$ is the actual reward delivered by state v at time t, and σ_r^2 is the variance of that reward.

The observable data thus becomes $D = (r, tasks, G)$ and the posterior can be computed as:

$$\begin{aligned} P(H|D) &\propto P(D|H)P(H) \\ &= P(r|tasks, G, H)P(tasks|G, H)P(G|H)P(H) \\ &= \left[\prod_v \left[\prod_t P(r_{v,t}|\mu_v)\right] P(\mu_v|\theta_{c_v}, H)\right] \left[\prod_w P(\theta_w|H)\right] P(tasks|G, H)P(G|H)P(H) \end{aligned} \tag{3.18}$$

The model will thus favor hierarchies that cluster together states which deliver similar rewards. This predicts a certain pattern of reward generalization, with states inheriting the rewards of other states in the same cluster. Importantly, this implies that boundaries in the reward landscape will induce clusters that in turn will influence planning. This is another key novel prediction of our model which we assess in experiments six and seven.

3.4.3 Inference

We approximated Bayesian inference using Markov chain Monte Carlo (MCMC) sampling (see Methods) to draw samples approximately distributed according to $P(H|D)$. We simulate each participant by drawing a single hierarchy H (the sampled hierarchy) from the posterior and then using it to make decisions. This is equivalent to

Table 3.1: Model parameter settings. These were held constant across all simulations and experiments.

Parameter	Range	Role	Value
α	$[0, +\infty)$	CRP concentration parameter: larger values favor more clusters	1
ε	$[0, 1]$	choice stochasticity: larger values lead to more deterministic choices	0.6
# samples	$[1, +\infty)$	number of MCMC iterations per simulated participant	10000
$\bar{\theta}$	$(-\infty, +\infty)$	average reward of entire graph	15
σ_θ	$(-\infty, +\infty)$	standard deviation of average cluster rewards around $\bar{\theta}$	10
σ_μ	$(-\infty, +\infty)$	standard deviation of average state rewards	10
σ_r	$(-\infty, +\infty)$	standard deviation of state rewards	5

assuming participants perform probability matching in the space of hierarchies.

In all simulations, we assume perfect (lossless) memory for the observations $D = (r, \mathit{tasks}, G)$. While the process of learning the graph structure and the task and reward distributions is interesting in its own right, our focus is on inferring the hidden hierarchy H. We thus aim to develop a computational-level theory (in the Marrian sense)[188] of hierarchy discovery that remains agnostic of the particular algorithmic and implementational details but rather instantiates an entire class of process models that could approximate the ideal Bayesian observer.

3.4.4 Choices

For simulations one through five, we simulate choices based on H using linking assumptions analogous to those used by the authors of the original studies. For experiments one through five and experiment seven, we simulate hierarchical planning based on H using HBFS. For experiment six, we assume participants prefer the state with the highest expected reward. For experiment eight (Supplemental Results), we assume participants prefer to reduce the entropy of the posterior as much as possible. In order to account for choice stochasticity, for each decision, we simulate the appropriate choice as dictated by the model with probability ε, or choose randomly with probability $1 - \varepsilon$. We picked all model parameters by hand based on simulations one through five and based on the design for experiments six and seven. We used the same parameters throughout all simulations and experiments (Table 3.1).

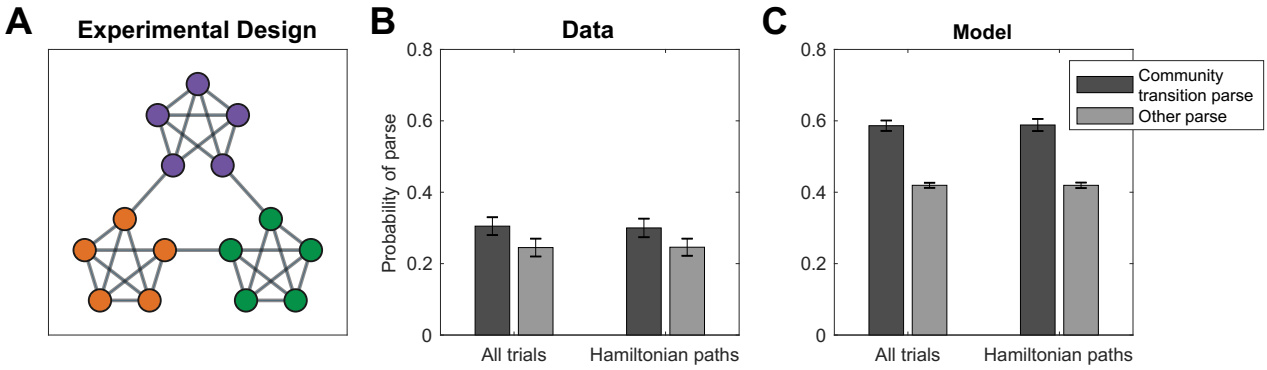


Figure 3.4: Detecting transitions between communities.
A. Graph from Schapiro et al. [246]. Colors visualize the communities of states. Participants never saw the graph or received hints of the community structure.
B. Results from Schapiro et al. [246], experiment 1, showing that participants were more likely to parse the graph along community boundaries. Participants indicated transitions across communities as "natural breaking points" more often than transitions within communities. Error bars are s.e.m. (30 participants).
C. Results from simulations showing that hierarchy inference using our model is also more likely to parse the graph along community boundaries. Error bars are s.e.m. (30 simulations).

Simulations

Our hierarchy discovery framework can account for a wide range of behavioral effects which present difficulty for existing models (Table 3.2). To assess its basic predictions, we simulate a series of behavioral experiments reported in the literature. Our primary focus is on validating the foundational assumptions behind our model without committing to any particular parameterization. We therefore use the same set of generic handpicked parameters across all of our simulations (Table 3.1) and focus primarily on reliable qualitative effects that are independent of the particular choice of parameters.

3.5 Simulation one: bottleneck transitions

Schapiro et al. [246] demonstrated that people can detect transitions between states belonging to different clusters in a graph with a particular topological structure, such that nodes in certain parts of the graph are more densely connected with each other than with other nodes. This type of topology is also referred to as *community struc-*

ture (Figure 3.4A), with the clusters built into the graph topology referred to as *communities*. Thirty participants viewed sequences of stimuli representing random walks or Hamiltonian paths in the graph. Participants were instructed to press a key whenever they perceived a natural breaking point in the sequence. The authors found that participants pressed significantly more for cross-community transitions than for within-community transitions (Figure 3.4B). Participants did this despite never seeing a bird's-eye view of the graph or receiving any hints of the community structure.

Following experiment 1 from Schapiro et al. [246], for each simulated participant, we sampled a hierarchy H based on the graph G and performed 18 random walks of length 15 initiated at random nodes, and 18 random Hamiltonian paths. For simplicity, we simulated key presses deterministically. In particular, we counted a transition from node u to node v as a natural breaking point if and only if the nodes belonged to different clusters in the inferred hierarchy H, that is, if $c_u \neq c_v$, where c are the cluster assignments in H. This recapitulated the empirical results (Figure 3.4C; random walks: $t(58) = 7.35, p < 10^{-9}$; Hamiltonian paths: $t(58) = 7.32, p < 10^{-9}$, two sample, two-tailed t-tests).

The dense connectivity within communities and sparse connectivity across communities drives the posterior to favor hierarchies with cluster assignments similar to the true underlying community structure. This arises due to Eq. 3.7 and Eq. 3.8 in the generative model, corresponding to desiderata 3 and 4, respectively. This posits that, during the generative process, edges across clusters are less likely than edges within clusters, resulting in a posterior distribution that penalizes alternative hierarchies in which many edges end up connecting nodes in different communities.

3.6 Simulation two: bottleneck states

Solway et al. [263] performed several experiments demonstrating that people spontaneously discover graph decompositions that fulfill certain formal criteria of optimality. Similarly to Schapiro et al. [246], in their first experiment, forty participants were trained on a graph with community structure (Figure 3.5A). As before, participants never

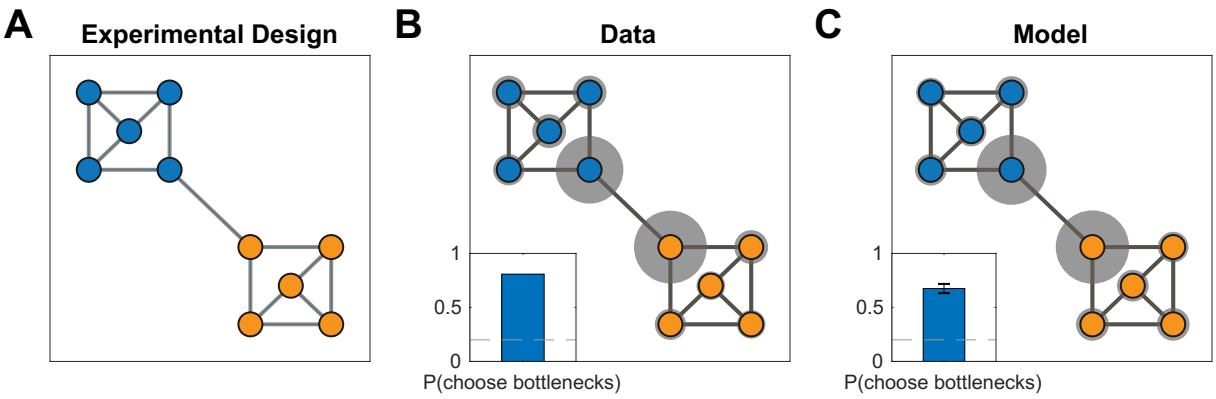


Figure 3.5: Detecting bottlenecks states
A. Graph from Solway et al. [263], experiment 1, with colors indicating the optimal decomposition according to their analysis.
B. Results from Solway et al. [263], experiment 1, showing that people are more likely to select the bottleneck nodes as bus stop locations. Gray circles indicate the relative proportion of times the corresponding node was chosen. Inset, proportion of times either bottleneck node was chosen. Dashed line is chance (40 participants).
C. Results from simulations showing that our model is also more likely to pick the bottleneck nodes since they are more likely to end up as endpoints of a bridge. Notation as in B. Inset error bars are s.e.m (40 simulations).

saw the full graph or were made aware of its community structure but instead had to to rely solely on transitions between nodes. Participants were then asked to designate a single node in the graph as a "bus stop", which they were told would reduce navigation costs in a subsequent part of the experiment. Participants preferentially picked the two bottleneck nodes on the edge that connects the two communities (Figure 3.5B), which are the optimal subgoal locations under these constraints. This suggests that participants were able to infer the graph topology based on adjacency information only, and to decompose it in an optimal way.

As in simulation one, for each participant we sampled a hierarchy H based on the graph G used in the experiment. Since participants were asked to identify three candidate bus stops, we randomly sampled three nodes among all nodes that belonged to bridges in H, i.e. $\{u : b_{w,z} = (u,v), \text{ for some } (w,z) \in E' \text{ and } u \in V\}$, where the b and E' are the bridges and edges in the sampled hierarchy, respectively. This replicated the empirical result (Figure 3.5C; 65% of choices, $p < 10^{-20}$, right-tailed binomial test), with most simulated participants inferring hierarchies respecting the underlying community structure. Similarly to simulation one, this was due to the higher connectivity within communities than across communities.

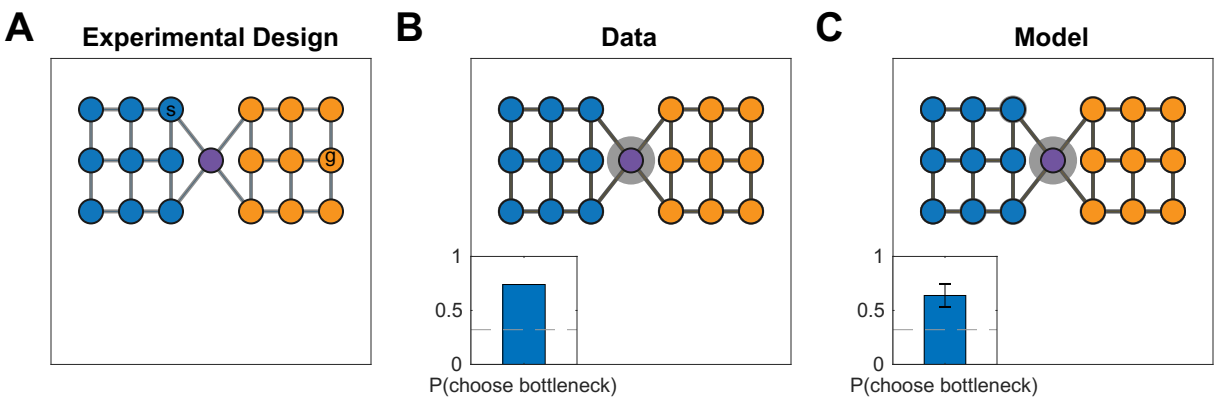


Figure 3.6: Planning transitions across communities first
A. Graph from Solway et al. [263], experiment 2, with colors indicating the optimal decomposition according to their analysis. The nodes labeled *s* and *g* indicate an example start node and goal node, respectively.
B. Results from Solway et al. [263], experiment 2, showing that people are more likely to think of bottlenecks states first when they plan a path between states in different communities. Notation as in Figure 3.5B (10 participants).
C. Results from our simulation demonstrating that our model also shows the same preference. Using the hierarchy identified by our model, the hierarchical planner is more likely to consider the bottleneck state first, since it is more likely to end up as the endpoint of a bridge connecting the two clusters. Error bars are s.e.m (10 simulations).

3.7 Simulation three: hierarchical planning

In their second experiment, Solway et al. [263] trained 10 participants to navigate between pairs of nodes in a different graph (Figure 3.6A). On some trials, participants were asked to indicate a single node that lies on the shortest path between two nodes in different communities. They found that participants overwhelmingly selected the bottleneck node between the two communities (Figure 3.6B), suggesting that they not only discovered the underlying community structure, but also leveraged that structure to plan hierarchically, "thinking first" of the high-level transitions between clusters of states.

As in simulations one and two, we sampled H based on the graph G for each participant. We then used the hierarchical planner to find 50 hierarchical paths between random start locations in the left community and random goal locations in the right community. For each such path, we counted the first node that belongs to a bridge as the response on the corresponding trial, since this is the first node considered by the planner and is therefore the closest approximation to what a participant would think of first (see definition of HBFS in the

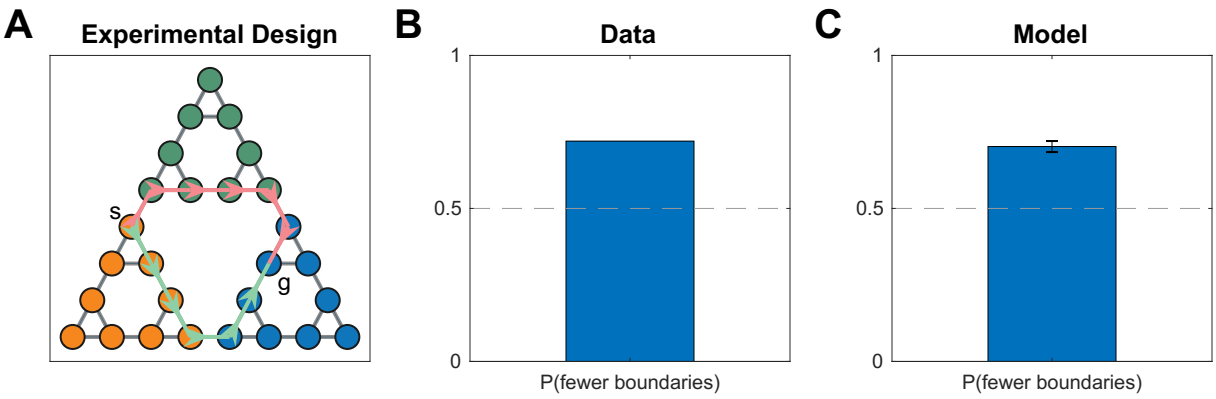


Figure 3.7: Preferring paths with fewer community boundaries
A. Graph representing the Towers of Hanoi task used in Solway et al. [263], experiment 4. Vertices represent game states, edges represent moves that transition between game states. The start and goal states (s, g) show an example of the kinds of tasks used in the experiment. Colored arrows denote the two shortest paths that could accomplish the given task, with the red path passing through two community boundaries and the green path passing through a single community boundary.
B. Results from Solway et al. [263], experiment 4, showing that participants preferred the path with fewer communities, or equivalently, the path that crosses fewer community boundaries. Bar graph shows fraction of participants (35 participants). Dashed line is chance.
C. Results from simulations showing that our model also exhibits the same preference. Bar graph shows the fraction of simulations that chose the path with fewer community boundaries. Error bar is s.e.m. (35 simulations).

Methods). This replicated the empirical results, with a strong tendency for the bottleneck location to be selected (Figure 3.6C; 60% of choices, $p < 0.001$, right-tailed Monte Carlo test). The discovered hierarchies resembled the underlying community structure for the same reason as in simulations one and two, resulting in the bottleneck node frequently becoming part of a bridge that all paths between the two communities would pass through.

3.8 Simulation four: shorter hierarchical paths

In their final experiment, Solway et al. [263] demonstrated hierarchical decomposition and planning in the Towers of Hanoi task, which can be represented by a graph (Figure 3.7A) in which each node is a particular game state and each edge corresponds to move that transitions from one game state to another. They leveraged the fact that there are two different shortest paths between some pairs of states (for example, the start and goal states in Figure 3.7A), but those paths cross a different number of community boundaries as defined by their optimal decomposition analysis. Hierarchical planning predicts that participants will prefer the path which crosses fewer

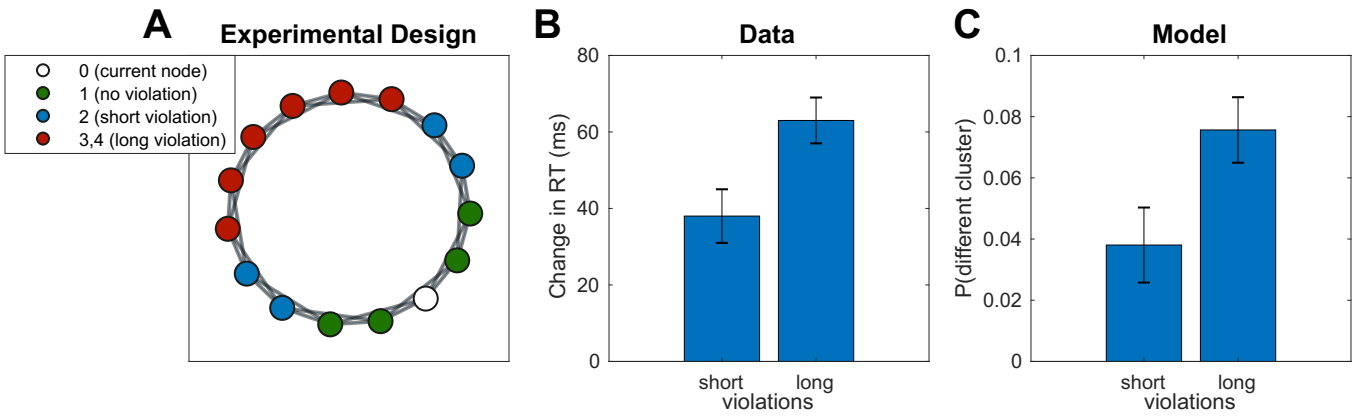


Figure 3.8: Slower reactions to cross-cluster transitions

A. Graph used in Lynn et al. [181]. Each node (white) is connected to its neighboring nodes and their neighbors (green). Blue nodes are 2 transitions away from the white node, while red nodes are 3 or 4 transitions away.

B. Results from Lynn et al. [181] showing that, on the test trial, participants were more slower to respond to long violations than to slow violations. Change in RT is computed with respect to average RT for no-violation transitions. Error bars are s.e.m (78 participants). RT, reaction time.

C. Results from simulations showing that long violations are more likely to end up in a different cluster, which would elicit a greater surprise and hence a slower RT, similar to crossing a cluster boundary.

community boundaries, and this is indeed what the authors found (Figure 3.7B).

After choosing a hierarchy H for each simulated participant as in the previous simulations, we then used the hierarchical planner to find paths between the six pairs of states that satisfy the desired criteria. In accordance with the data, this resulted in the path with fewer community boundaries being selected more frequently (Figure 3.7C; 71.4% of choices, $p < 10^{-10}$, right-tailed binomial test). Similarly to the previous simulations, the model tended to carve up the graph along community boundaries. Since the planner first plans in the high-level graph, it prefers the path with the fewest clusters, and hence with the fewest cluster boundaries.

3.9 Simulation five: cross-cluster jumps

In our final simulation, we considered a study performed by Lynn et al. [181], in which participants experienced a random walk along the graph shown in Figure 3.8A. This was a self-paced serial reaction time task in which participants had to press a different key combination for each node. A small subset of transitions violated the

graph structure and instead "teleported" the participant to a node that is not connected to the current node. Importantly, there were two types of violations: short violations of topological distance 2 and long violations of topological distance 3 or 4. The authors found that participants were slower to respond to longer than to shorter violations, suggesting that participants had inferred the large scale structure of the graph.

While this is not a planning task, we assumed that reaction times (RTs) for cross-cluster transitions would be slower for the same reason as in experiment three: a cross-cluster transition requires planning in the high-level graph, imposing a kind of context switch cost that would slow RTs.

Like in the previous simulations, we sampled H for each participant and then simulated a random walk along the graph G, with occasional violations as described in Lynn et al. [181]. In order to model RTs, we assumed a bimodal distribution of RTs, with fast RTs for transitions within clusters and slow RTs for transitions across clusters, consistent with the notion that cross-cluster transitions are more surprising. Instead of actually simulating RTs, we simply counted the number of cross-cluster transitions during the random walk. This revealed a greater number of cross-cluster transitions for long violations than for short violations, consistent with the data (Figure 3.8B; $t(154) = 4.50, p = 0.00001$, two-sample t-test). This occurs because nearby nodes are more likely to be clustered together, and hence violations of greater topological distance increase the likelihood that the destination node would be in a different cluster than the starting node, resulting in a greater surprise and a slower RT.

Experiments

Framing hierarchy discovery as a particular kind of hidden state inference allows us to straightforwardly extend our framework to account for the distribution of tasks and rewards in the environment. The best of our knowledge, the effect of tasks and rewards on hierarchy discovery has not been systematically investigated in prior empirical work. In particular, our model predicts that people will cluster together adjacent states that regularly co-occur in the same task (Eq. 3.12 and 3.13) as well as adjacent states that deliver similar rewards (Eq. 3.15, 3.16,

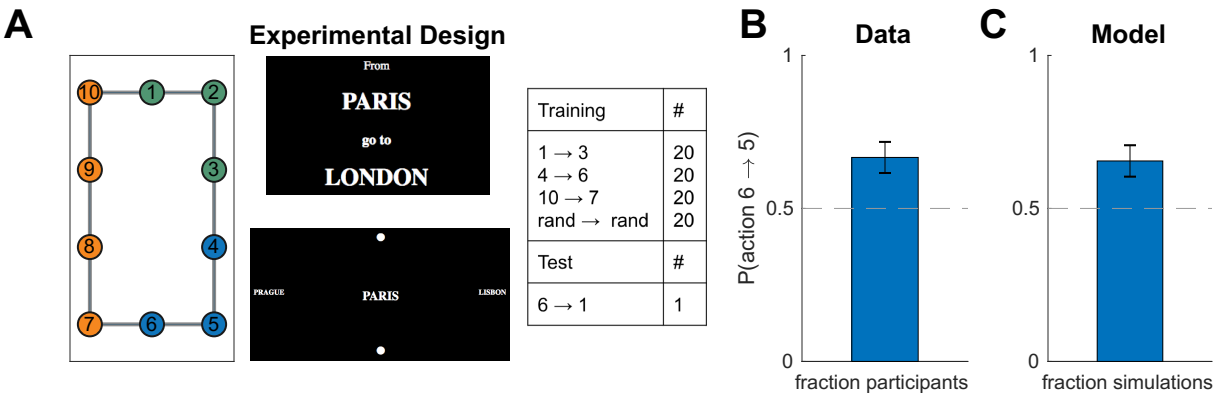


Figure 3.9: Hierarchy discovery is sensitive to the task distribution

A. (Left) graph used in experiment one with no topological community structure. Colors represent clusters favored by the training protocol (right). Numbers serve as node identifiers and were not shown to participants. "Rand" denotes a node that is randomly chosen on each trial. (Middle) trial instruction (top) and screenshot from the starting state (bottom).

B. Results from experiment one showing that, on the test trial, participants were more likely to go to state 5 than to state 7, indicating a preference for the route with fewer cluster boundaries. Dashed line is chance. Error bars are s.e.m. (87 participants)

C. Results from simulations showing that our model also preferred the transition to state 5. Notation as in B.

and 3.17). Note that these novel predictions are not accounted for by any existing models (Table 3.2). Some of these predictions are counterintuitive – for example, is not clear why states associated with the same reward should be relevant from the perspective of planning. Understanding state chunks as hidden states which deliver observations from the same distribution – similar in spirit to latent cause models in structure learning [110,43] – sheds insight into these phenomena, which we assess in a series of eight behavioral experiments. Each experiment aims to verify a unique prediction of our framework, or to provide a nuance or address a potential concern related to a previous experiment. As such, we believe that no experiment stands on its own and that our empirical results should be considered together, providing convergent evidence that hierarchy discovery is a form of structure learning in which the inferred state chunks are shaped by tasks and rewards in addition to environment topology.

3.10 Experiment one: task distributions

In our first experiment, we sought to validate the prediction of our model that clusters could be induced by the distribution of tasks alone, even when the graph topology does not favor any particular clustering. In particular, our model predicts that states which frequently co-occur in the same task should be clustered together, since the hierarchical planner is optimal within clusters.

3.10.1 Participants

We recruited 87 participants (28 female) from Amazon Mechanical Turk (MTurk). All participants received informed consent and were paid $3 for their participation. We reasoned that paying participants a fixed amount would incentivize them to complete the experiment in the least amount of time possible, which entails balancing path length with planning time in a way that is characteristic of many real-world tasks. The experiment took 17 minutes on average. All experiments were approved by the Harvard Institutional Review Board.

3.10.2 Design

We asked participants to navigate between pairs of nodes ("subway stops") in a 10-node graph (the "subway network", Figure 3.9A, left). The training trials (Figure 3.9A, right) were designed to promote a particular hierarchy: the task to navigate from node 1 node 3 would favor clustering nodes 1,2,3 together; the task to navigate from node 4 to node 6 would favor clustering nodes 4,5,6 together; and the task to navigate from node 10 to node 7 would favor clustering nodes 7,8,9,10 together (Figure 3.9A, left). The normative reason for this is that hierarchical planning is always optimal within a cluster. In the generative model, this is taken into account by Eq. 3.13, which leads to a preference to cluster together start and goal states from the same task. The purpose of the tasks with random start and goal states was to encourage participants to learn a state representation for efficient planning and not simply to respond habitually.

In order to test the model prediction, after training, we asked participants to navigate from node 6 node 1. Note that the two possible paths are of the same length and a planner with a flat representation of the graph would show no preference for one path over the other. Furthermore, since there is no community structure and the graph is perfectly symmetric, any clustering strategy based on graph structure alone would not predict a preference. Conversely, our model predicts that participants will tend to choose the path through node 5 since it passes through a single cluster boundary, whereas the path through node 7 passes through two cluster boundaries.

3.10.3 Procedure

The experiment was implemented as a computer-based game similar to Balaguer et al. [13] in which participants had to navigate a virtual subway network. At the start of each trial, participants saw the names of the starting station and the goal station. After 2 s, they transitioned to the navigation phase of the trial, during which they could see the name of the current station in the middle of the screen, surrounded by the names of the four neighboring stations, one in each cardinal direction. If there was no neighboring station in a particular direction, participants saw a filled circle instead of a station name. The name of the goal station was also indicated in the top left corner of the screen as a reminder. The navigation phase began with a 3-s countdown during which participants could see the starting station and its neighbors but could not navigate. Participants were instructed to plan their route during the countdown. After the countdown, participants could navigate the subway network using the arrow keys. Transitions between stations were instantaneous. Once participants reached the goal station, they had to press the space bar to complete the trial. This was followed by a 500-ms "success" message, after which the trial ended and the instruction screen for the next trial appeared. Pressing the space bar on a non-goal station resulted in a "incorrect" message flashing on the screen. Attempting to move in a direction without a neighboring station had no effect. Following Balaguer et al. [13], stations were named after cities, with the names randomly shuffled for each participant.

The subway network corresponded to the graph in Figure 3.9A. In order to assign arrow keys to edges, we first

embedded the graph in the Cartesian plane by assigning coordinates to each vertex, which resulted in the planar graph shown in the figure. Then we assigned the arrow keys to the corresponding cardinal directions. For each participant, we also randomly rotated the graph by 0°, 90°, 180°, or 270°. Participants performed 80 training trials (20 in each condition; Figure 3.9, right) in a random order. After the training trials, we showed a message saying that now the subway system was unreliable, so that some trips may randomly be interrupted midway. This was followed by the test phase, during which participants performed the test trial $6 \rightarrow 1$ and two additional test trials. In order to prevent new learning during the test phase, all test trials were interrupted immediately after the first valid keypress. The two additional test trials were not included in the analysis or any of the following experiments. We used the destination of the first transition on the test trial as our dependent measure in the analysis. We reasoned that the direction in which participants attempt to move first is along what they perceive to be the best route to the goal station. Since participants were paid a fixed fee for the whole experiment, they were incentivized to complete it as fast as possible, which can be best achieved by planning the shortest route during the 3-s countdown and then following it.

3.10.4 Results and discussion

In accordance with our model predictions, more participants moved to state 5 on the test trial, rather than to state 7 (Figure 3.9B; 58 out of 87, $p = 0.003$, two-tailed binomial test). Notice that this could not be explained by habitual responding: while participants may have learned action chunks that solve the corresponding tasks (for example, pressing right and down from state 1 to state 3), the actions from state 6 to its neighboring states were never reinforced as part of a stimulus-response association or as part of a longer action chunk. In particular, state 6 is never a starting state or an intermediary state, except possibly in the random tasks, in which both directions have equal probability of being reinforced. The effect cannot be explained by state familiarity either, since participants experienced states 5 and 7 equally often, on average. Finally, it is worth noting that a model-free RL account would make the opposite prediction, since state 7 would on average have a higher value than state 5, as it gets rewarded directly.

This result was consistent with the model predictions (Figure 3.9C; 57 out of 87, $p = 0.005$, two-tailed binomial test), suggesting that people form hierarchical representations for planning in a way that tends to cluster together nodes that co-occur in the same tasks.

In order to rule out the possibility that the preference for the $6 \rightarrow 5$ transition on the test trial was simply due the frequent $5 \rightarrow 6$ transitions during training, which might have somehow reinforced the reverse action, we performed a logistic regression of test trial choices on the total number of $6 \rightarrow 5$, $5 \rightarrow 6$, $6 \rightarrow 7$, and $7 \rightarrow 6$ transitions during training. This did not show a significant effect for any of the transitions ($F(1, 83) < 2.8, p > 0.09$ for all coefficients), thus disproving such a "flat", non-hierarchical associative account.

3.11 Experiment two: task distributions and suboptimal planning

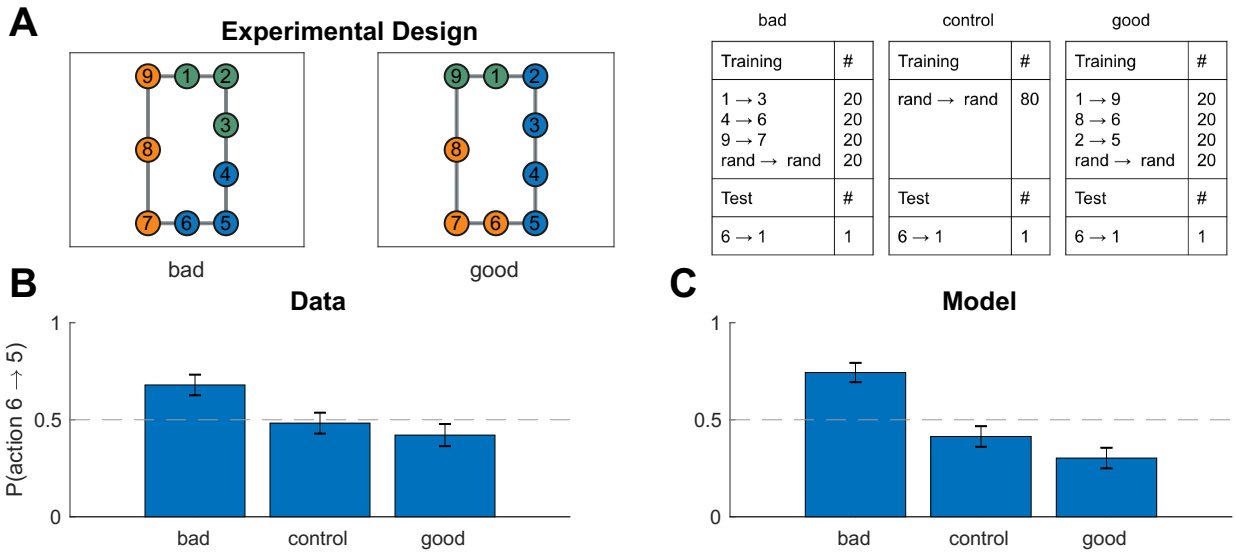
Since both paths on the test trial in the previous experiment were of equal length, the bias that participants developed would make no difference for their performance on that task. Next we asked whether such a bias would occur even if it might lead to suboptimal planning, as the hierarchical planner would predict. For example, most of us have had the experience of navigating between two locations through other, more familiar location, even though a shorter but less familiar route might exist (see also [141]).

3.11.1 Participants

We recruited 241 participants (112 female) from MTurk. Of those, 78 were assigned to the "bad" clusters condition, 87 were assigned to the "control" condition, and 76 were assigned to the "good" clusters condition. Participants were paid $2.50 for their participation. The experiment took 15 minutes on average.

3.11.2 Design and procedure

We used the same paradigm as experiment one, with the only difference that node 10 was removed from the graph (Figure 3.10A, left). Now the analogous training regime (Figure 3.10A, right) would promote "bad" clus-



bad		control		good	
Training	#	Training	#	Training	#
1 → 3 4 → 6 9 → 7 rand → rand	20 20 20 20	rand → rand	80	1 → 9 8 → 6 2 → 5 rand → rand	20 20 20 20
Test	#	Test	#	Test	#
6 → 1	1	6 → 1	1	6 → 1	1

Figure 3.10: Different task distributions can induce different hierarchies in the same graph

A. (Left) graph used in experiment two with colors representing clusters favored by the training protocol in the "bad" (left) and "good" (middle) condition. (Right) training and test protocols for all three conditions.

B. Results from experiment two showing that, on the test trial, participants were more likely to go to state 5 than to state 7 in the bad condition, leading to the suboptimal route. The effect was not present in the control condition or in the good condition. Dashed line is chance. Error bars are s.e.m. (78, 87, and 76 participants, respectively).

C. Results from simulations showing that our model exhibited the same pattern. Notation as in B.

ters that lead to suboptimal planning on the test trial. We also performed a control version of the experiment with different participants using random training tasks only, as well as a "good" condition with a third group of participants that promotes clusters that lead to optimal planning on the test trial (Figure 3.10A).

3.11.3 Results and discussion

On the test trial, participants preferred the suboptimal move from 6 to 5 in the "bad" clusters condition (Figure 3.10B; 53 out of 78, $p = 0.002$, two-tailed binomial test), significantly more than the control condition ($\chi^2(1, 165) = 6.52, p = 0.01$, chi-square test of independence) and the "good" condition ($\chi^2(1, 154) = 10.4, p = 0.001$). This was in accordance with the model predictions (Figure 3.10C; 58 out of 78 simulated participants, $p = 0.00002$, in the bad condition; $\chi^2(1, 165) = 18.2, p = 0.00002$ for bad vs. control condition; $\chi^2(1, 154) = 30.0, p < 10^{-7}$ for bad vs. good condition). This suggests that participants formed clusters based on the distribution of tasks and used these clusters for hierarchical planning, even when that was suboptimal on the particular test task.

Note that the model showed a slight preference for the $6 \rightarrow 7$ transition in the control condition, which did not reach significance. This occurs because there are fewer nodes along that path and, in the presence of random clusters induced by the random tasks, there will be, on average, fewer cluster boundaries along that shorter path. Participants exhibited a similar qualitative pattern, which however was not significant. Somewhat surprisingly, there was also no significant difference between the control condition and the good condition in the data ($\chi^2(1, 163) = 0.6, p = 0.5$) and the model ($\chi^2(1, 163) = 2.2, p = 0.14$), although both showed a slightly greater preference for the $6 \rightarrow 7$ transition compared to the control condition. This suggests that in this particular scenario, the "good" clusters have a relatively weaker effect.

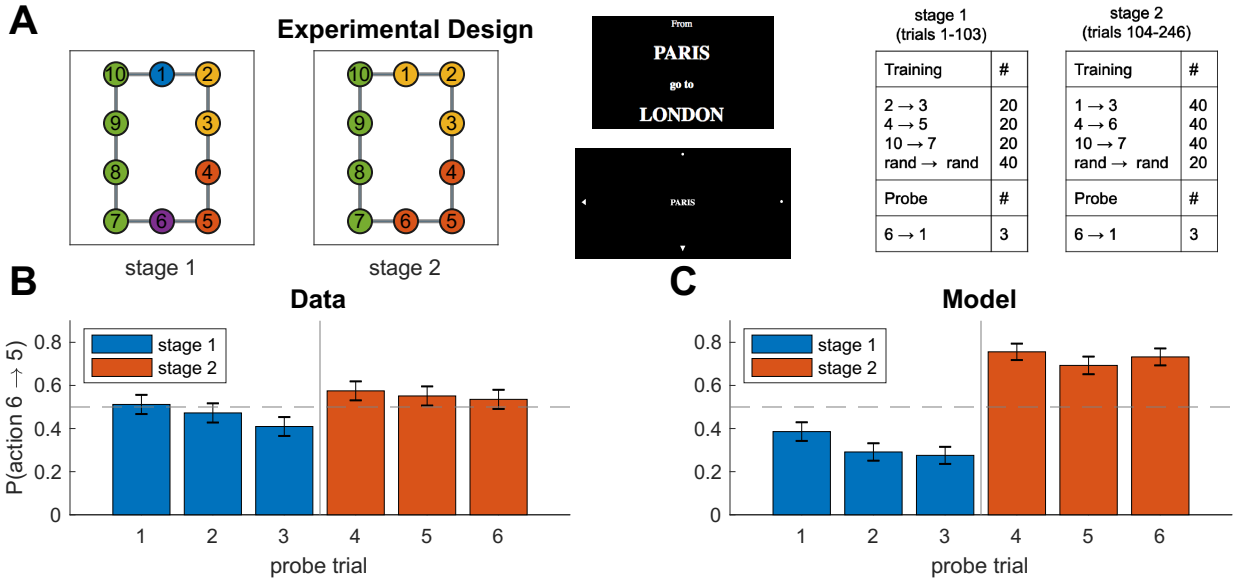


Figure 3.11: Learning dynamics.

A. Experiment three used the same graph as experiment one, with main difference that training (right panel) took part in two stages that promoted different hierarchies (first and second panel), with probe trials interspersed throughout training. Notation as in Figure 3.9A.

B. Results from experiment three showing that (1) the first stage of training makes participants more likely to go to state 7 on the probe trials, which could not be explained by a "flat" associative account, (2) this tendency appears gradually as participants accumulate more evidence, and (3) this preference is reversed during the second stage of training. Error bars are s.e.m. (127 participants).

C. Results from simulations showing that our model exhibited the same learning dynamics. Notation as in B.

3.12 Experiment three: learning effects

While the experiments so far demonstrate the key predictions of our model, they only test asymptotic behavior and as such do not assess how beliefs about hierarchy evolve over the course of learning. Further, these results could conceivably be explained by simpler, non-hierarchical accounts, such as a bidirectional association forming between states 5 and 6 due to the frequent transition from 5 to 6 during training. While we addressed this in our previous analysis, in our next experiment we sought to definitively rule out such "flat" associative explanations and to study the dynamics of learning.

3.12.1 Participants

We recruited 127 participants (54 female) from MTurk. Participants were paid $8.50 for their participation. The experiment took 47 minutes on average.

3.12.2 Design

We trained participants on the same graph experiment one, but using a different training regimen, with two stages of training and multiple probe trials (Figure 3.11A). The first stage of training (trials 1-103) promoted clustering states 2 and 3 together, separately from state 1, and similarly promoted clustering states 4 and 5 together, separately from state 6 (Figure 3.11A, first panel). In order to investigate the dynamics of hierarchy discovery, we interspersed three probe trials that tasked participants to navigate from state 6 to state 1 throughout the first stage, expecting participants' preferences to become stronger over time. Note that predictions on the probe trials are the opposite of predictions on the test trial in experiment one: the path from 6 to 1 through state 5 now crosses three cluster boundaries, whereas the path through state 7 crosses only two cluster boundaries, so participants should prefer the path through state 7. This cannot be explained by a naive associative account since state 6 is no more likely to co-occur with state 7 than with state 5.

To further assess learning, which sought to reverse this effect during the second stage of training which promoted the same clusters as experiment one (Figure 3.11A, second panel). Our prediction was that, in accordance with the results of experiment one, this would eliminate participants' preference for going to state 7 on the probe trials in favor of state 5, since the path through state 5 now crosses a single cluster boundary.

3.12.3 Procedure

We used the same procedure as experiment one, with the following changes. First, there was no information indicating to participants that something had changed between stages (i.e., between trials 103 and 104). Additionally, instead of having a test stage in the end, we interspersed six probe trials $6 \rightarrow 1$, spaced evenly throughout training (trials 34, 68, 103 during the first stage and trials 150, 197, 246 during the second stage). Unlike the test trials in experiment one, probe trials were indistinguishable from training trials and were not interrupted after the first move. Another difference from experiment one was that instead of being rotated, the graph was randomly flipped horizontally and/or vertically for each participant. Finally, unlike experiment one, participants did not see the names of adjacent stations but instead saw arrowheads indicating they could move in the corresponding direction (Figure 3.11A, third panel).

3.12.4 Results and discussion

Consistent with our predictions, participants showed a significant preference for moving to state 7 on the last probe trial of the first stage (Figure 3.11B, probe trial 3; 75 out of 127 participants, $p = 0.05$, two-tailed binomial test). This effect was modulated by the amount of training, with the smallest effect on the first probe trial and the largest effect on the third probe trial (slope = -0.27, $F(1, 379) = 3.96, p = 0.05$, mixed effects logistic regression with probe trials 1-3). The effect was reversed during the second stage (slope = 0.73, $F(1, 252) = 8.10, p = 0.005$, mixed effects logistic regression with probe trials 3-4).

These results were consistent with the model predictions (Figure 3.11C). The model preference for state 7

on the third probe trial (92 out of 127 simulated participants, $p < 10^{-6}$, two-tailed binomial test) is once again due to the preference of the hierarchical planner for paths with fewer state clusters. As in our empirical data, this effect became stronger over time (slope = -0.34, $F(1, 379) = 5.58, p = 0.02$, mixed effects logistic regression with probe trials 1-3). The reason is that, as the hierarchy inference process observes more tasks, it accumulates more evidence (i.e., more terms in the product in Eq. 3.14) which sharpens the posterior distribution $P(H|D)$ around its mode (i.e., the most probable hierarchy, Figure 3.11A, first panel). This corresponds to a decrease in the uncertainty over H, which makes it more likely that the sampler will draw the hierarchy in Figure 3.11A, first panel, and plan according to it. Hence this effect of learning occurs due to the decrease in uncertainty resulting from the additional observations. Finally, like our participants, the model reversed its preference during the second stage of training (slope = 2.13, $F(1, 252) = 53.43, p < 10^{-11}$, mixed effects logistic regression with probe trials 3-4). This effect occurs because tasks during the second stage of training shift the mode of the posterior (Figure 3.11A, second panel). Together, these results rule out simple, nonhierarchical associative accounts and demonstrate that our model can account for learning dynamics due to changes in uncertainty and changes in the mode of the posterior distribution over hierarchies.

It is worth noting that there is a significant discrepancy between the effect size predicted by the model and the effect size observed in the data, particularly with regard to the difference between the first stage and the second stage of training. We believe this is attributable to the lack of any adjacency information in this experiment, which ameliorates the potential associative confound, but by the same token renders the low-level graph G more difficult to learn compared to the other experiments. This results in noisier choices, which could be modeled by adjusting the noise parameter ε. However, since our aim was to assess robust qualitative effects that are independent of particular parameterizations, for consistency we preferred using the same set of parameters as in simulations one through five (Table 3.1).

One prediction of our hierarchical planner is that transitions across cluster boundaries should take longer, since they involve planning both in the high-level graph H as well as the low-level graph G, whereas transitions within clusters only involve planning in G. We therefore analyzed log-transformed RTs during the first stage of

training as a function of transition type using mixed effects linear regression. We classified each transition in one of three types:

- Action chunk: transitions along the shortest path for any of the training tasks ($2 \to 3, 4 \to 5, 10 \to 9, 9 \to 8, 8 \to 7$).
- State chunk: transitions along the shortest path for any of the training tasks, but in the reverse direction ($3 \to 2, 5 \to 4, 9 \to 10, 8 \to 9, 7 \to 8$).
- Boundary: all other transitions ($1 \to 2, 2 \to 1, 1 \to 10, 10 \to 1, 6 \to 7, 7 \to 6, 6 \to 5, 5 \to 6, 3 \to 4, 4 \to 3$).

We separated within-cluster transitions into action chunk transitions and state chunk transitions in order to account for the fact that actions along frequently occurring task solutions are reinforced much more frequently and are thus likely to be chunked together into stereotyped action sequences by the motor system. Consistent with this, we found that boundary transitions and state chunk transitions were both significantly slower than action chunk transitions ($F(1, 46592) = 121.00, p < 10^{-27}$ and $F(1, 46592) = 69.56, p < 10^{16}$, respectively). In contrast, state chunk transitions are reinforced as frequently as boundary transitions, on average. Despite this, boundary transitions were slower than state chunk transitions ($F(1, 46592) = 10.68, p = 0.001$), as predicted by our hierarchy discovery account. Together, these results further support our hypothesis that task boundaries delineate cluster boundaries and are consistent with the notion that state abstraction drives temporal abstraction [157] —humans first decompose the environment into clusters that then constrain the chunking of actions into sequences that operate within those clusters.

3.13 Experiment four: perfect information

One downside of experiments one and two is that effects of memory confound hierarchy inference and planning. In particular, it is unclear that participants are able to learn and represent the full graph G. This is most evident

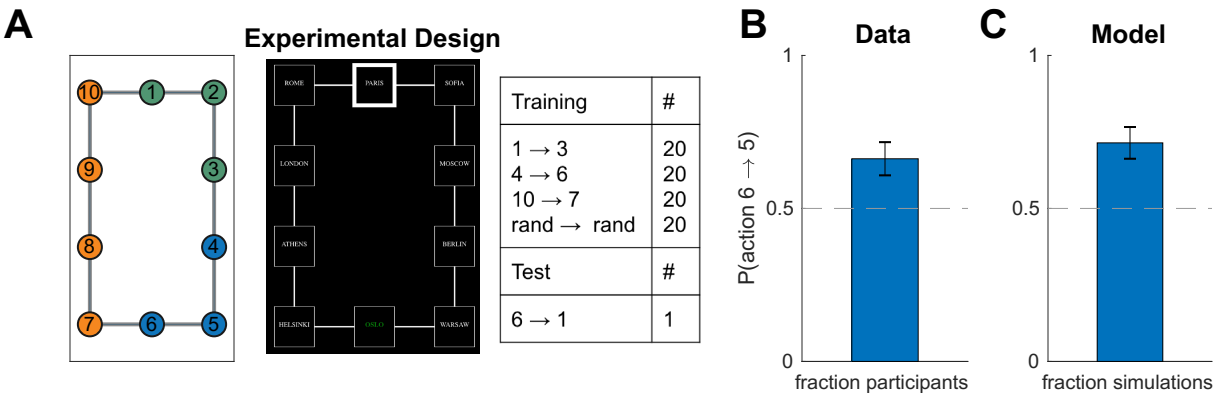


Figure 3.12: Hierarchy discovery based on task distribution in fully visible graphs
A. (Left) experiment four used the same graph as experiment one, however this time the graph was fully visible on each trial (middle). Notation as in Figure 3.9A.
B. Results from experiment four showing that, like in experiment one, participants were more likely to go to state 5 on the test trial. Dashed line is chance. Error bars are s.e.m. (77 participants)
C. Results from simulations showing that our model also preferred the transition to state 5. Notation as in B.

in the control condition of experiment two (Figure 3.10B), in which our model predicts that participants should be better than chance, when in fact they are not, thus questioning whether they are learning the graph or planning efficiently in the first place. Note that this does not pose a significant challenge to our hierarchy discovery account; people's choices could still be accounted for without invoking a hierarchical planner, for example by assuming a preference to remain in the same cluster (moving to state 5) rather than crossing a cluster boundary (moving to state 7). Nevertheless, we sought to overcome this limitation by ensuring participants know the full graph G.

3.13.1 Participants

We recruited 77 participants (33 female) from MTurk. Participants were paid $2.00 for their participation. The experiment took 10 minutes on average.

3.13.2 Design

We used the same graph and training protocol as experiment one, except this time participants could see the whole graph at any given time (Figure 3.12A). This put participants on an equal footing with the hierarchy inference and hierarchy planning algorithms, both of which assume perfect knowledge of the graph G.

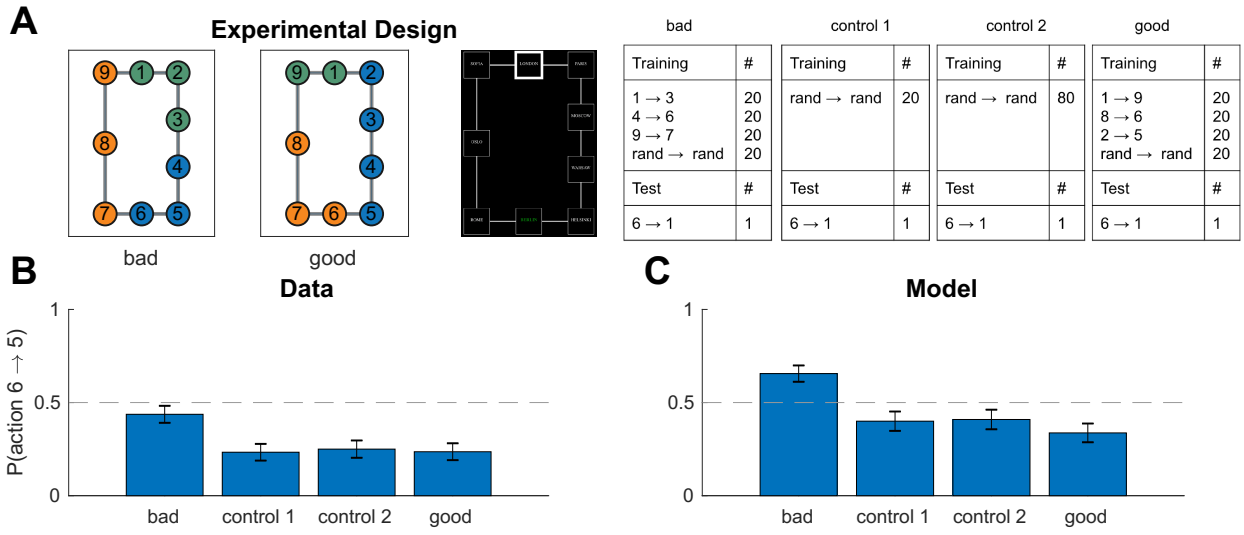
3.13.3 Procedure

The procedure was similar to experiment one, with the main difference that participants had a bird's-eye view of the entire subway network throughout the experiment (Figure 3.12A, middle panel). Subway stations were represented by squares connected by lines which represented the connections between stations. The current station was highlighted with a thick border and the goal station was in green font.

Since planning is significantly easier in the setting, we removed the 3-s countdown, so that participants could start navigating immediately after the 2-s instruction. Additionally, instead of rotating the map, we randomly flipped it horizontally for half of the participants. We also omitted the "unreliable trips" warning before the test phase since participants now only saw a single test trial, which was immediately interrupted after the first move. The experiment took 10 minutes on average and participants were paid $2.00.

3.13.4 Results and discussion

We found that, even with full knowledge of the graph, participants still developed a bias (Figure 3.12B; 51 out of 77 participants, $p = 0.0014$, right-tailed binomial test), consistent with the predictions of our model (Figure 3.12C; 55 out of 77, $p = 0.0002$, two-tailed binomial test). This provides strong support for our hierarchy discovery account, suggesting tasks constrain cluster inferences above and beyond constraints imposed by graph topology, which in turn constrain hierarchical planning on novel tasks.



bad		control 1		control 2		good	
Training	#	Training	#	Training	#	Training	#
1 → 3 4 → 6 9 → 7 rand → rand	20 20 20 20	rand → rand	20	rand → rand	80	1 → 9 8 → 6 2 → 5 rand → rand	20 20 20 20
Test	#	Test	#	Test	#	Test	#
6 → 1	1	6 → 1	1	6 → 1	1	6 → 1	1

Figure 3.13: Task distributions can bias hierarchical planning even in fully visible graphs
A. (Left) experiment five used the same graph as experiment two, however this time the graph was fully visible on each trial. Notation as in Figure 3.10A.
B. Results from experiment five showing that participants were still biased by the training tasks in the bad condition, performing worse on the test trial compared to the other conditions. Dashed line is chance. Error bars are s.e.m. (119, 90, 88, and 89 participants, respectively).
C. Results from simulations showing that our model exhibited the same pattern. Notation as in B.

3.14 Experiment five: perfect information and suboptimal planning

We next asked whether hierarchy discovery could lead to suboptimal planning even when the full graph is visible.

3.14.1 Participants

We recruited 386 participants (175 female) from MTurk. Of those, 119 were assigned to the "bad" clusters condition, 90 were assigned to the "control 1" condition, 88 were assigned to the "control 2" condition, and 89 were assigned to the "good" clusters condition. Participants were paid $2.00 for their participation. The experiment took 9 minutes on average.

3.14.2 Design and procedure

We used the same graph and training protocol as experiment two, except this time participants could see the whole graph at any given time (Figure 3.13A), as in experiment four. Additionally, we included two control conditions, one with 20 training trials ("control 1") and one with 80 training trials ("control 2"). The first control condition ensured participants received the same number of random tasks as the bad and good conditions, while the second control condition ensured that participants received the same total number of tasks as in the bad and good conditions. We used the same experimental procedure as experiment four.

3.14.3 Results and discussion

As in experiment two, inducing "bad" clusters still led to significantly worse performance on the test trial than either control condition (Figure 3.13B; $\chi^2(1, 209) = 9.35, p = 0.002$ for bad vs. control 1; $\chi^2(1, 207) = 7.7, p = 0.006$ for bad vs. control 2, chi-square test of independence). Inducing "good" clusters (89 participants) led to significantly better performance than "bad" clusters ($\chi^2(1, 208) = 9.03, p = 0.003$ for bad vs. good), although not significantly better than the control conditions ($\chi^2(1, 179) = 0.002, p = 0.96$ and $\chi^2(1, 177) = 0.05, p = 0.8$ for good vs. control 1 and good vs. control 2, respectively). This accords with our model predictions (Figure 3.13C; $\chi^2(1, 209) = 10.1, p = 0.002$ for bad vs. control 1; $\chi^2(1, 207) = 4.8, p = 0.03$ for bad vs. control 2; $\chi^2(1, 208) = 17.7, p = 0.00003$ for bad vs. good) and strongly suggests that people default to hierarchical planning over clusters influenced by the task distribution, even in simple, fully observable graphs. Notice that in both control conditions, participant preferred the shorter path (21 out of 90 participants, $p < 10^{-6}$ for control 1; 22 out of 88 participants, $p < 10^{-6}$ for control 2, two-tailed binomial tests), indicating that they were indeed able to plan effectively when given the full graph without tasks to bias them towards particular clusters, thus overcoming the limitation of experiment two.

One notable difference between our model predictions and the empirical data is that our model predicts a preference for state 5 in the "bad" condition (78 out of 119 simulated participants, $p = 0.0009$, two-tailed bi-

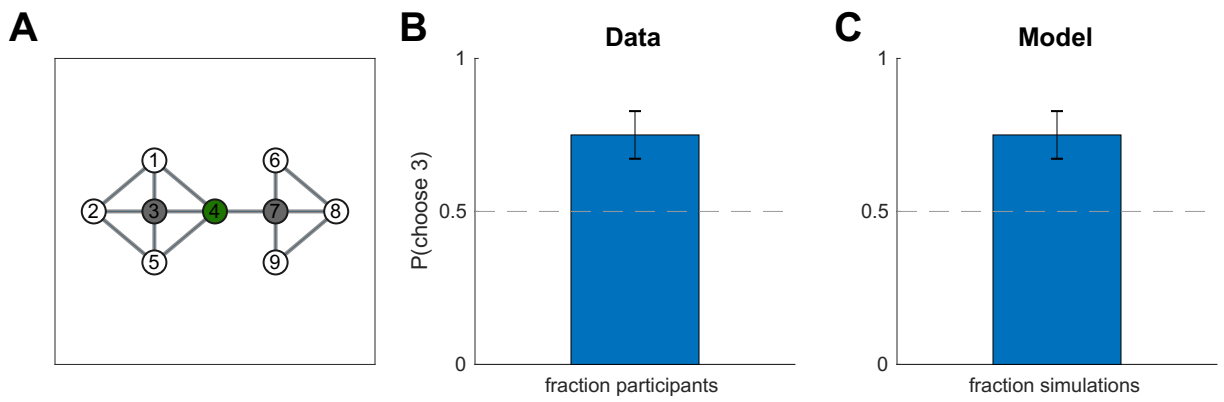


Figure 3.14: Reward generalization within clusters.
A. Graph used in experiment six. Numbers indicate state identifiers and were not shown to participants. Participants were told that states deliver 15 points on average and that, on a given day, state 4 (green) delivered 30 points. They were then asked which of the two gray nodes (states 3 and 7) they would choose.
B. Results from experiment six showing that participants preferred state 3, which is in the same topological cluster as state 4, suggesting they generalized the reward within the cluster. Error bars are s.e.m (32 participants).
C. Results showing that the model exhibited the same pattern. Notation as in B.

nomial test), whereas participants did not show significant preference (52 out of 119 participants, $p = 0.2$, two-tailed binomial test). We believe this occurs because the task is much simpler when the graph is fully visible and participants could easily perform optimal "flat" planning, rather than having to resort to hierarchical planning. This effect could be captured straightforwardly using a mixture of BFS and HBFS for planning, rather than just HBFS.

3.15 Experiment six: reward generalization

In this experiment, we tested the prediction that rewards generalize within clusters. While this prediction is not unique to our model and could be accounted for by Gaussian processes over graphs [156,314] or by the successor representation [270,59], the idea that clusters generate similar rewards is a core assumption of our model that we sought to validate before assessing how clusters inferred based on rewards could influence planning (experiment seven).

3.15.1 Participants

We recruited 32 participants from the MIT undergraduate community. The experiment took around 3 minutes and participants were not paid for their participation.

3.15.2 Design

We showed participants the graph ("network of gold mines") in Figure 3.14A and told them that in the past, states delivered an average reward of 15 ("grams of gold"), but today, state 4 (green) delivered a reward of 30. We then asked participants to choose one between state 3 and state 7 ("mines to explore").

3.15.3 Procedure

Each participant was given a sheet of paper with instructions and the graph in Figure 3.14A, without node identifiers. The instructions were as follows:

You work in a large gold mine that is composed of multiple individual mines and tunnels. The layout of the mines is shown in the diagram below (each circle represents a mine, and each line represents a tunnel). You are paid daily, and are paid $10 per gram of gold you found that day. You dig in exactly one mine per day, and record the amount of gold (in grams) that mine yielded that day. Over the last few months, you have discovered that, on average, each mine yields about 15 grams of gold per day. Yesterday, you dug in the blue mine in the diagram below, and got 30 grams of gold. Which of the two shaded mines will you dig in today? Please circle the mine you choose.

Half of the participants were given a version in which the graph was flipped horizontally, i.e. the topological cluster was on the right side.

3.15.4 Results and discussion

Participants preferred state 3, the state in the same topological community as state 4 (Figure 3.14B; 24 out of 32 participants, $p = 0.007$, two-tailed binomial test), as the model predicts (Figure 3.14C; 24 out of 32 simulated

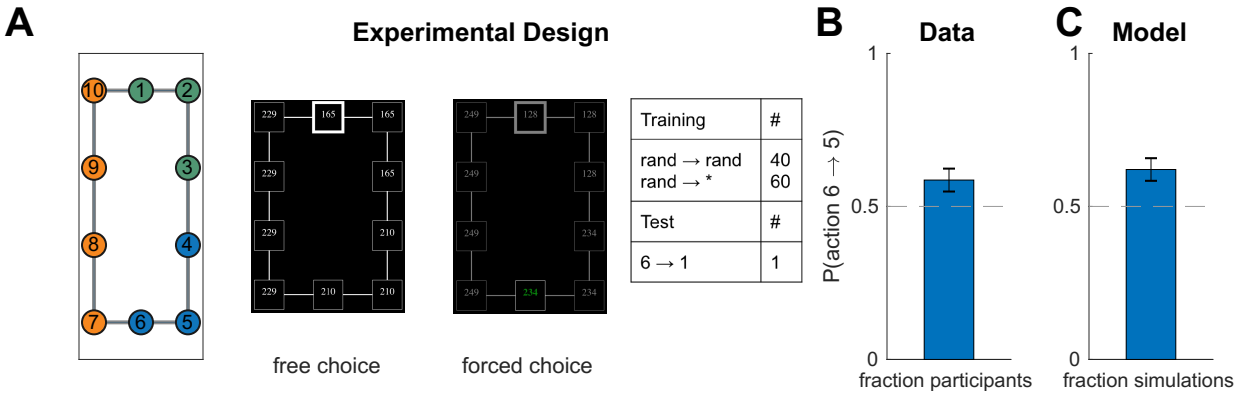


Figure 3.15: Rewards induce clusters that influence planning.
A. (Left) Experiment seven employed the same graph as in experiments one and four, with the difference that clusters were induced via the reward rather than the task distribution. (Middle) screenshots from free choice and forced choice trials. (Right) training and test protocol. "Rand" indicates that a random state was chosen on each trial, while the asterisk indicates a free choice trial (i.e., the participant was free to choose any node).
B. Results from experiment seven showing that participants were more likely to prefer the path with fewer reward cluster boundaries. Error bars are s.e.m. (174 participants).
C. Results from simulations showing that the model exhibited the same preference. Notation as in B.

participants, $p = 0.007$). Topological structure is the only driver of hierarchy discovery in this case, since there is only a single reward and no tasks. The structure of the graph favors clustering state 4 together with state 3 since they belong to the same community. The higher-than-average reward of state 4 then drives up the average reward θ for that cluster, which in turn drives up average reward μ for the states that belong to it, and in particular for state 3. In contrast, state 7 often ends up in a separate cluster which is not influenced by the reward of state 3 and thus has an expected reward of 15, the average $\bar{\theta}$ for the entire graph.

3.16 Experiment seven: rewards and planning

While the results of experiment six validated the reward assumptions of our model, they could also be accounted for by alternative models, such as the successor representation [270,59] or a Gaussian process with a diffusion kernel [156,314]. We therefore sought to test the unique prediction of our model that, in the absence of community or task structure, hierarchical planning will occur over clusters delineated by boundaries in the reward landscape of

the environment.

3.16.1 Participants

We recruited 174 participants (68 female) from MTurk. Participants were paid $0.50 plus a bonus equal to the number of points earned on a randomly chosen trial in cents (up to $3.00). This encouraged participants to do their best on every trial. The experiment took 9 minutes on average.

3.16.2 Design

We asked participants to navigate a graph ("network of gold mines") with the same structure as in experiments one and four (Figure 3.15A). Unlike the previous experiments, participants performed a mix of randomly shuffled free choice and forced choice trials. On free choice trials, participants started in a random node and could navigate to any node they chose. Once they reached their target node, they collected the reward ("grams of gold") from that node. On forced choice trials, participants had a specified random target node and they could only collect reward from that node, similarly to the tasks in experiments one through five. Free choice trials encouraged participants to learn the reward distribution, while forced choice trials encouraged planning and prepared participants for the test trial, which was also a forced choice trial. Crucially, the rewards favored a clustering like the one in experiments one and four: states 1,2,3 always delivered the same reward, as did states 4,5,6 and states 7,8,9,10. Similarly to experiments one and four, participants were tested on a forced choice task from node 6 to node 1.

3.16.3 Procedure

Participants were told they would navigate a virtual network of gold mines. The graph layout was identical to experiment four (Figure 3.15A) and participants could see a bird's-eye view of the entire graph, with each node representing a mine and each edge representing a tunnel between mines. Each mine delivered a certain number of points, which were displayed inside the node of the corresponding mine. Similarly to experiment four, par-

ticipants could navigate between mines using the arrow keys and choose a mine using the space bar, after which they were informed how many points they earned from the chosen mine and immediately began the next trial. Mines always delivered deterministic points as indicated on each mine. Only points earned from the chosen mine counted.

On free choice trials, any mine could be chosen. On forced choice trials, only a single target mine was available and all other minds are grayed out (Figure 3.15A, right). Attempting to select a different mine resulted in a "incorrect" message flashing and did not change the current mine. There were 60 free and 40 forced choice training trials, starting at random states. The target state on the forced choice trials during training was also random. In order to prevent participants from developing a model-free bias towards one side of the graph based on reward magnitude (for example, because states there happened to be more rewarding), we randomly resampled the points of all states with probability 0.2 on every free choice trial. All reward values were sampled uniformly between 0 and 300.

3.16.4 Results and discussion

As in experiments one and four, participants showed a preference for the route with fewer cluster boundaries (Figure 3.15B; 102 out of 174 participants, $p = 0.03$, two-tailed binomial test). The model made same prediction (Figure 3.15C; 108 out of 174 participants, $p = 0.001$, two-tailed binomial test). However, this time the clusters were inferred solely based on rewards, rather than topological structure or task distributions. This shows that the reward distribution in the environment can affect hierarchy discovery and consequently planning over that hierarchy, in accordance with our model predictions.

3.17 General discussion

In this study, we proposed a Bayesian model for discovering state hierarchies based on a generative model of the topological structure, rewards, and tasks in the environment. Building on the Bayesian brain hypothesis[63,35,39]

Table 3.2: Model comparison. Summary of which results could potentially be accounted for by alternative models and which results rule out certain models.

Result \ Model	Efficient state space modularization [193]	Optimal behavioral hierarchy [263]	Temporal community structure [246]
Simulation one: bottleneck transitions	YES	YES	YES
Simulation two: bottleneck states	YES	YES	YES
Simulation three: hierarchical planning	YES	YES	NO (representations not suitable for planning)
Simulation four: shorter hierarchical paths	YES	YES	NO (representations not suitable for planning)
Simulation five: cross-cluster jumps	YES	YES	YES
Experiment one: task distributions	YES	NO (fixed task distribution)	NO (no task distribution)
Experiment two: task distributions and suboptimal planning	YES	NO (fixed task distribution)	NO (no task distribution)
Experiment three: learning effects	NO (no uncertainty)	NO (fixed task distribution)	NO (no task distribution)
Experiment four: perfect information	YES	NO (fixed task distribution)	NO (no task distribution)
Experiment five: perfect information and suboptimal planning	YES	NO (fixed task distribution)	NO (no task distribution)
Experiment six: reward generalization	NO (no reward distribution)	NO (no reward distribution)	NO (no reward distribution)
Experiment seven: rewards and planning	NO (no reward distribution)	NO (no reward distribution)	NO (no reward distribution)
Experiment eight: uncertainty and active learning	NO (no uncertainty)	NO (no uncertainty)	NO (no uncertainty)

and on principles developed in structure learning[110] and robotics[91], it postulates that the brain learns useful hierarchical representations by inverting this generative model to infer the hidden hierarchical structure of the world. These representations can be subsequently used to plan efficiently and flexibly in the face of changing task demands. The model accounts for a number of phenomena previously reported in the literature and makes new predictions which we verified empirically.

The model goes beyond previous accounts of clustering states in the environment both in the scope of findings it can explain and in the predictions it makes. Schapiro et al.[246] proposed a recurrent neural network that learns community structure based on the temporal statistics of the environment. Their model and other event segmentation models[320] can learn to predict stimuli that tend to co-occur, such as states in the same community, and can detect boundaries when those predictions are violated, for example after a transition to a new community. While this could explain the effects of simulations one through five and possibly experiments one, two and three, it will be challenged by experiments four, five and six, in which the environment is fully visible and all stimuli co-occur together. More importantly, these models make no predictions regarding action selection, and it is not clear how the learned representations can be used for planning.

One way to implement action selection using the temporal statistics experienced by the agent is to assume that bidirectional associative chains are built based on training sequences and planning is done over low-level states using association-based distances (that is, less closely associated states are considered to require a larger transition cost). This is similar in spirit to the model proposed by Lynn et al.[181] and could replicate the results of experiments one through five. However, the lack of a hierarchical planning component means that it would fail to account for simulations three and four, and the lack of an explicit hierarchy means that simulation two would also pose a challenge. Simulation one controls for the number of transitions, so an associative account would not detect the boundary transitions. It would also fail to capture the reward-based clustering observed in experiment seven.

Solway et al.[263] offer a formal definition of an optimal hierarchy; however, as the authors themselves point out, their analysis is not a plausible account of how hierarchy is learned, because it assumes the agent already

knows the optimal solution to all possible tasks in the environment, which defeats the purpose of learning a hierarchy in the first place. A detailed comparison between our model and alternative candidate models that make similar claims is shown in Table 3.2. Note that there is no need to simulate these models as they categorically cannot capture some of the effects presented here.

The problem of planning has been studied extensively both for biological as well as artificial agents. Correspondingly, our work resonates with several important strands of research, which we discuss in turn below. Refer also to the Supplemental Material accompanying this manuscript for further discussion on related ideas.

3.17.1 Model-based and model-free reinforcement learning

In psychology, planning and goal-directed behavior, which selects actions based on their outcomes, is often contrasted with habitual behavior, which selects actions automatically[76]. The dichotomy between these two kinds of action selection strategies – fast, reflexive, habitual, unconscious responding (also referred to as System 1) on one hand, and slow, reflective, goal-directed, conscious deliberation (also referred to a System 2) on the other – has permeated the field for over a century[289,291], has been reincarnated in many forms[71,146,271], and is realized in distinct neural circuits[318,319,16,15]. In RL, habitual and goal-directed behaviors have been associated with *model-free* and *model-based* algorithms, respectively[80,56].

In model-free algorithms, the agent learns a value function for each state and/or each action by trial-and-error, and then chooses actions with respect to the learned values. In model-based RL, the agent learns the reward and transition structure of the environment and combines that information to find actions leading to reward. Model-free RL is associated with fast, reflexive responding because it myopically considers the value of the current state/action only, while model-based RL is associated with slow, reflective responding because it considers the transitions and rewards of multiple states/actions ahead. Despite the superficial similarity of these two systems and the two computational components we propose – a fast online planner, and a slow offline representation learner – there is a profound difference. Both model-free and model-based RL (as well as other versions of habitual and goal-directed strategies) perform online action selection, with agents having to rely on one or the other

to solve a particular task. In contrast, our planner makes decisions about which actions to select in a particular task, while the hierarchy discovery process works offline, outside the context of a particular task. In a way, our planner is more like model-based RL in that it performs deliberate, goal-directed computations based on its internal model of the world (the "model" in model-based RL). The purpose of the hierarchy discovery process is to learn such an internal representation that can be used by a planner with limited computational resources. Indeed, the cognitive limitations inherent in this type of reflective decision-making [146] are the foundational assumptions behind our approach.

Notice that we only invoke a model-based planner in order to motivate hierarchy discovery and to link the hierarchy to decision-making, without committing to the particular HBFS algorithm. While hierarchy discovery can be justified in this way without invoking a model-free system, we certainly do not exclude the possibility of such a system existing and operating in parallel with the planner. A model-free component could easily be incorporated into our framework, for example by having a parallel model-free system which learns the values of states and actions. During decision making, the agent can use a meta-cognitive process to arbitrate between the planner and the model-free system [158]. Like the planner, the model-free system could also operate at different levels of abstraction, learning values for the clusters in addition to the low-level states. This could account for effects of reflexive responding that we explicitly controlled for in our experiments.

Another distinguishing feature of our approach is that, by leveraging the deterministic nature of the transition structure, our planner can rely on simple shortest path algorithms to find solutions to tasks. In contrast, traditional approaches to model-based RL involve computationally costly operations such as value iteration or sampling of candidate trajectories, both of which scale poorly with the size of the state space. Even though the combinatorial explosion of sampling could be managed by heuristics such as pruning of decision trees [140,141], such approaches are unnecessary in the deterministic domains often considered in planning problems.

It is worth highlighting that our results cannot be explained by standard model-free or model-based RL algorithms. In experiments one, two, four, and five, model-free RL would have learned high values for states 3, 6, and 7, and hence would prefer the transition $6 \rightarrow 7$ on the test trial. In experiments one and four, model-based RL

would be indifferent between the two possible trajectories on the test trial due to the symmetry of the graph. In experiments two and five, model-based RL should actually favor the shorter path via $6 \rightarrow 7$. All of these predictions go against our participants' tendency to pick the transition $6 \rightarrow 5$.

The two systems in our proposal might also bear superficial resemblance to the Dyna architecture in RL[278] in which a slow, offline model-based simulator trains a fast, online model-free system to respond adaptively to situations it has never experienced before. This process is reminiscent of hippocampal replay[257] and is particularly useful when the environment changes too often for trial-and-error learning to be effective. In contrast, we propose an offline inference process for learning representations (the "model" in model-based RL) that a model-based system can use to plan in previously unseen tasks. If our proposal is extended with a separate model-free component as suggested above, it could be integrated with Dyna, which would in turn use the model-based system to train the model-free system.

Our results also cannot be explained by the successor representation[59,202], an intermediate approach along the model-free/model-based continuum that predicts which states a given policy will visit. This will fail to account for experiment three, in which none of the optimal policies visit state 1, and the policies of the random tasks would not favor the transition $6 \rightarrow 7$ over $6 \rightarrow 5$.

3.17.2 Hierarchical reinforcement learning

A long-standing challenge for traditional RL has been the combinatorial explosion that occurs when planning and learning take place over long time horizons. This challenge been addressed by hierarchical RL (HRL), which breaks down the problem into sub-problems at multiple levels of abstraction. One influential approach to HRL extends the agent's action repertoire to include *options*[280] which consist of sequences of actions (the option policy) that accomplish certain subgoals (for example, exiting a room). When an option is selected, the corresponding action sequence is executed as a single behavioral unit. Options are also referred to as skills, subroutines, partial policies, macro-actions, or policy chunks, while the original actions are sometimes referred to as primitive actions.

As with regular or "flat" RL, HRL also comes in two distinct flavors[27]: model-free HRL and model-based HRL. Like model-free RL, model-free HRL[69,137,232] learns a value function; however, in this case the value function is additionally defined for options. Error-driven learning occurs both on the high level by learning which options lead to rewarding outcomes, as well as on the lower level by learning the best option policy for each option. Importantly, there is no notion of a transition function which could be used for planning, and hence model-free HRL could not exhibit the behaviors predicted by our model. Although a model-free component could be incorporated into our framework as discussed above, it would not account for the results of experiments one through five. The only options that could conceivably have been learned by model-free HRL are the ones corresponding to the training tasks (for example, $1 \rightarrow 3$, $4 \rightarrow 6$, and $10 \rightarrow 7$ for experiment one). This set of options could not explain the results of the test trial which requires going in the opposite direction.

In model-based HRL[24], as in model-based RL, the agent separately learns a transition function and a reward function. Additionally, the agent is furnished with an *option model* that specifies the initiation states (for example, locations within a room), termination states (subgoals; for example, doors), average duration, and average reward of each option. The agent can thus plan over options rather than primitive actions, essentially performing mental "jumps" in the state space, allowing it to first form a high-level plan between subgoals reachable via options and then refining that plan on the lower level by simply following the corresponding option policies. This form of *saltatory* model-based HRL is conceptually identical to our proposal. While our work builds on concepts developed in parallel to HRL in the field of robot navigation and planning[91], our model can be cast in HRL terms by considering each task as equivalent to placing a positive reward in the goal state and a small negative reward in all other states, thus encouraging the agent to find the shortest path to the goal state. Edges in the high-level graph H can be seen as options, with subgoals specified by the endpoints of bridges and option policies specifying how to reach the subgoals within a cluster.

Our work introduces two critical improvements to model-based HRL. First, as in model-based RL, model-based HRL assumes planning occurs by sampling trajectories through the state space, which in our proposal is performed by the deterministic and much more efficient HBFS algorithm. Second, as Botvinick & Weinstein[24]

point out, a critical open question is how useful options are discovered in the first place. The approach they propose is based on the successor representation[59] under a random walk policy, which partitions the state space along topological bottlenecks. However, this would not predict the results of experiments one through five and experiment seven, in which clustering occurs based on tasks and rewards only.

When framed within HRL, our approach can be viewed as a solution to the option discovery problem which has plagued the field of HRL since its inception, as the original formulation never specified how useful options are learned in the first place. Discovering useful subgoals and, correspondingly, useful options is critical, since an inadequate set of options can lead to dramatically worse performance compared to regular RL[27]. This has led to the proliferation of a rich literature on option discovery, to which we turn next.

3.17.3 Option discovery

While earlier work on HRL assumed the options are supplied manually[280], a growing number of HRL studies have focused on the problem of discovering useful options. A detailed review of the option discovery literature is beyond the scope of this discussion, but we highlight some of the main approaches. Most option discovery methods fall in one of two broad categories: *state abstraction* and *temporal abstraction* methods[190].

State abstraction methods first decompose the state space by identifying clusters or subgoals, and subsequently identify options based on that decomposition[62,300]. The state space could be partitioned into clusters based on the value function[73], the state features[134], or the transition function[182,233]. Other approaches designate certain states as subgoals and learn options that lead to those subgoals, which could be defined by salient events[37], object-object interactions[167], frequently visited states[276,191,74], or large changes in the reinforcement gradient[74]. Other subgoal discovery methods rely on graph theoretic notions to identify bottlenecks (such as "doors" between rooms)[50], the boundaries of strongly connected regions of the state space (such as the "walls" of rooms)[196], or clusters of states (such as the rooms themselves)[187,51] as subgoals.

Temporal abstraction (or policy abstraction) methods directly learn the option policies, without resorting to state abstraction as an intermediate step. Some of these approaches identify frequently used action sequences

from successful trajectories[190,115,299]. Other approaches posit a generative model for policies that favors temporal abstraction, and then perform probabilistic inference to find the optimal policy[312,52].

Viewed in HRL terms, our model falls into the state abstraction category, since it first partitions the state space into clusters which in turn define subgoals and constrain behavior. However, our model goes beyond these previous attempts: it unifies multiple ideas from these different approaches under a single Bayesian framework that allows it to account for all the behavioral phenomena in our study. To the best of our knowledge, no single option discovery method based on state abstraction would capture all of them. Option discovery methods based on temporal abstraction would also fail to account for the results of experiments one through five, since participants were never trained to navigate in the directions tested in the test trial.

3.17.4 Future directions

Thus far our model only addresses the question of state chunking (how the environment is represented as discrete states at different levels of abstraction), while leaving open the question of action chunking (how actions are stitched together into larger behavioral units at different levels of temporal abstraction). Fitting the model into the HRL framework described above might seem like one way to incorporate action chunking, by assuming that the options leading to bridge endpoints are like action chunks that, when invoked, delegate behavior to a low-level controller that executes the sequence of actions in the option policy as a single behavioral unit. Yet the metaphor of options as action chunks is not necessarily appropriate; option policies execute in a closed-loop fashion, taking into consideration each state they encounter before choosing the next action. In contrast, action chunks are often operationally defined as open-loop action sequences that disregard intermediate states[69].

An alternative way to accommodate action chunking that more closely adheres to this definition is to allow caching of solutions to repeated calls to the planner with the same arguments[141]. That way, when a certain subpath within a cluster is traversed frequently as part of many tasks (for example, getting from your bed your bedroom door), the corresponding action sequences could be cached and, when the subtask needs to be solved as part of a larger task (for example, getting from your bed to work), the action sequence can be retrieved from

the cache and executed as a single unit, thus removing the need to unnecessarily recompute it every time. Indeed, such a simple scheme could account for phenomena such as action slips[69], or even be used to model task-bracketing activity in striatum (Figure B.1C in the Supplemental Material). Implementing action chunking in this way could also account for the speeding up of responses during training in our experiments.

Another limitation of our approach is the hard restriction that each low-level node is a member of a single cluster. This restriction could be relaxed by explicitly building in "soft" cluster membership into the generative model, or by allowing agents to entertain multiple possible hierarchies when planning. While we do not exclude the former approach as a possibility, the latter approach arises naturally from our generative model as the posterior can represent graded beliefs in multiple hypotheses. This is consistent with previous work showing how mixtures of deterministic representations can account for graded effects[121].

Our treatment of hierarchy discovery is restricted to the computational level of analysis (in the Marrian sense)[188] and assumes the agent can accurately sample from the posterior over hierarchies $P(H|D)$. Scaling up the model beyond the toy experiments considered here would require relaxing this assumption and instead resorting to approximate Bayesian inference. This points to a fruitful avenue for future research, namely probing the algorithmic details of hierarchy discovery. One possibility is to take our Monte Carlo sampler as a process model of hierarchy inference in the brain. This is consistent with previous work casting human probabilistic inference as a form of Monte Carlo sampling[55,66,114,245,302,286]. Of particular relevance are accounts of theory acquisition during development as a form of stochastic search in theory space[296]. In this light, hierarchy discovery can be understood as stochastic search in the space of hierarchies that occurs throughout development and aims to distill the complex structure of the world into a compact representation that can be used for nimble decision-making. This algorithmic account predicts a bias towards the prior and autocorrelation in the sampled hierarchies that could be assessed empirically. A noisy approximate inference process could also potentially explain the quantitative discrepancy between our model and the participants in experiment three.

Even with a sampling approximation, inference in large-scale environments would pose a challenge to computing the (unnormalized) posterior, which requires iterating over all nodes in the low-level graph. This means

that the complexity of a single update of a single cluster assignment is $O(N)$, which is very inefficient even for an offline process. This computational burden can be alleviated by noticing that updating the cluster assignment of a given node mainly affects the posterior through the immediately adjacent nodes. Thus instead of computing the posterior from scratch for each node update, the agent can simply adjust the posterior based on local changes (in fact, a Metropolis-Hastings rule like the one we use here can allow the agent to completely ignore the rest of the posterior and only consider local changes). Such a factorization of the posterior would bring the computational complexity of a single node update to $O(1)$. Furthermore, the Gibbs sampler we use here allows updates to be performed in any order, without sacrificing convergence guarantees. Overall, this suggests an ecologically appealing account of hierarchy discovery, according to which agents organize their immediate surroundings into clusters using only local update rules, without considering the rest of the world, yet in doing so they form global representations that can facilitate efficient planning between states that they never perceive in a single instant. The only aspect of the posterior which cannot be factorized is the connectivity constraint (see Methods), which could be addressed, for example, by assigning nodes to adjacent clusters only. Such a cheap and efficient online approximation might also explain how people were able to perform hierarchy inference during our relatively short experiments.

There are several ways in which our model could be extended to support planning in richer, large-scale environments. One way would be to allow deeper hierarchies ($L > 2$), which would be necessary in order to maintain the computational efficiency of the planner as the size of the graph increases. This could be achieved by recursively clustering the high-level states in H into higher-level states in another graph H', then clustering those in yet another graph H'', and so on up to hierarchy depth L that could be prespecified or also inferred from the data. The hierarchical planner can similarly extend recursively by reusing the same logic at the higher levels.

Yet even with deep hierarchies, the learned representations would pose a challenge to the planner as a number of low-level states in G grows to the scale encountered in real life (indeed, it would also pose a challenge to the long-term storage system). This highlights another limitation of our model, namely the use of tabular representations in which each state is represented by a separate token. This could be overcome by recognizing that

there is redundancy in the environment[25] – that is, the local structure of the graph G can be highly repetitive, with the same theme occurring over and over in different parts of the graph. For example, most cities have streets and buildings, most buildings have rooms and hallways, and most rooms have similar layouts. Representing each and every part of the environment would thus be wasteful, and this redundancy can be exploited by introducing templates or modules – blueprints for clusters that can compress the hierarchical representation by extracting the shared structure across clusters of the same type and only representing differences from some prototypical cluster. Clustering these modules will naturally give rise to the kind of compositionality characteristic of natural environments. Combined with partial observability and deeper hierarchies, this would allow the model to learn representations that support efficient planning in environments that approach the real world in their scale and complexity. Finally, the requirements that G is unweighted and undirected can be lifted if the planner is a hierarchical extension of a more sophisticated shortest path algorithm, such as Dijkstra's algorithm[47].

3.18 Conclusion

In summary, we propose a normatively motivated Bayesian model of hierarchy discovery for planning. The model builds on first principles from the fields of structure learning and robotics, and recapitulates a number of behavioral effects such as detection of boundary transitions, identification of bottleneck states, and surprise to transitions across clusters. The novel predictions of the model were validated in a series of behavioral experiments demonstrating the importance of the task and reward distributions in the environment, which could bias the discovered hierarchy in a way that is either beneficial or detrimental on new tasks. We also showed that the model accounts for reward generalization and uncertainty-based learning effects in a way that is consistent with human behavior. Together, these results provide strong support for a computational architecture in which an incremental offline process infers the hidden hierarchical structure of the environment, which is then used by an efficient online planner to flexibly solve novel tasks. We believe our approach is an important step towards understanding how the brain constructs an internal representation of the world for adaptive decision making.

3.19 Methods

3.19.1 Ethics Statement

All experiments involved human participants and were approved by the Harvard Institutional Review Board, number IRB15-2048. Participants that performed experiments on Amazon MTurk gave written consent (experiments one through five and experiment seven). Verbal consent was obtained for experiments six and eight.

3.19.2 Inference

We frame hierarchy discovery as computation of the posterior probability distribution $P(H|D)$. Since computing the posterior exactly is intractable, we approximate Bayesian inference over H using Metropolis-within-Gibbs sampling[239], a form of Markov chain Monte Carlo (MCMC). We initialized H by sampling from the prior $P(H)$ and on each MCMC iteration, we updated each component of H in turn by sampling from its (unnormalized) posterior conditioned on all other components in a single Metropolis-Hastings step, in the following order:

1. update the cluster assignments c using the conditional CRP prior (algorithm 5 in[208]),
2. update p, q, p', p'' using a truncated Gaussian random walk with standard deviation 0.1 (i.e. the proposal distribution is a Gaussian centered on the old value, excluding values above 1 or below 0),
3. update update the hierarchical edges E' with a proposal distribution that randomly flips the presence or absence of each edge with probability 0.1,
4. update the average cluster rewards θ using a Gaussian random walk with standard deviation 1, and
5. update the average state rewards μ using a Gaussian random walk with standard deviation 1.

The samples generated in this way approximate draws from the posterior, with asymptotic convergence to the true posterior in the limit of infinite samples. This approach can also be interpreted as stochastic hill climbing

with respect to a utility function defined by the posterior, which has been previously used to find useful hierarchies for robot navigation[91].

In all five simulations and experiments one through seven, for each simulated participant, we generated a Markov chain of a given length (see Table 3.1) and used the final sample H to generate predictions. This can be viewed as a form of probability matching over hierarchies, and is consistent with psychologically plausible algorithms for hypothesis generation and updating, although elucidating the algorithmic details of hierarchy discovery is beyond the scope of our present work. For experiment three, we ran the MCMC sampler before each probe trial, while for the other experiments we ran it before the test trial. We did this for purposes of efficiency, since we only evaluated predictions on the probe/test trials, and since our computational-level analysis is agnostic to the algorithmic details of hierarchy discovery.

In order to estimate entropy in experiment eight, we used a separate MCMC sampler for each graph for each participant to generate a set of $M = 50$ samples, using a lag of 100 iterations and burn-in period of 5000 iterations (for a total of 10000 generated samples, as in all other simulations).

Additionally, HBFS requires that the subgraph induced in G by each cluster must form a single connected component; that is, for every pair of states (u, v) in a cluster $w = c_u = c_v$, there must exist a path $(u, x_1, \ldots, x_k, v)$ such that $c_{x_k} = w \; \forall k$. To enforce this constraint, we imposed a penalty by subtracting 100 from the (unnormalized) log posterior for each pair of states in the same cluster that are not connected by a path passing through the cluster. This is equivalent to augmenting the generative model with the following rejection sampling procedure: draw H according to $P(H|D)$ and for each such pair of disconnected states, perform a Bernoulli coin flip with probability of success equal to e^{-100}. If all coin flips are successful, keep H, otherwise repeat the process with a new H. By imposing a "soft" constraint in this way, we ensured that the sampling algorithm can recover from a bad initialization by incrementally adjusting pairs of nodes that violate the constraint. Note that this still results in a valid posterior since normalization is not necessary for approximate inference.

3.19.3 Decision making

We assume choices based on H are either optimal (according to the model) with probably ε, or random with probability $1 - \varepsilon$. This is similar in spirit to the ε-greedy algorithm in RL, with the difference that we assume the meta-choice to choose randomly occurs before computing the optimal answer. This is equivalent to assuming that on $1 - \varepsilon$ of trials, participants simply do not perform the necessary computation. Such lapses could be due to a number of reasons, such as inattention or fatigue. While other factors such as motor variability may also contribute to choice stochasticity, we allow those to be absorbed by the ε parameter and leave them as the potential subject of future work.

Another source of variance in choices is the sampled hierarchy H, which could be different for each simulated participant. While our current experiments prohibit direct comparisons between hierarchies inferred by participants and hierarchies inferred by the model, any systematic variation in the sample hierarchies (for example, due to uncertainty of the posterior, as in experiments three and eight) would manifest when choices are aggregated across participants, as in our analyses.

Hierarchical planning

The optimal algorithm to find the shortest path between a pair of low-level vertices (s, g) in G is breadth-first search (BFS)[47] whose time and memory complexity is $O(N)$ (assuming $O(1)$ vertex degrees, i.e. $|E| = O(N)$). We use a natural extension of BFS to hierarchical graphs (hierarchical BFS or HBFS)[91] that leverages H to find paths more efficiently than BFS (approximately $O(\sqrt{N})$ time and memory). Intuitively, HBFS works by first finding a high-level path between the clusters of s and g, c_s and c_g, and then finding a low-level path within the cluster of s between s and the first bridge on the high-level path.

In particular, HBFS first finds a high-level path $(w_1, \ldots, w_{m'})$ between c_s and c_g in the high-level graph H (note that $w_1 = c_s$ and $w_{m'} = c_g$). Then it finds a low-level path $(y_1, \ldots, y_m)$ between s and u in $G[S]$ (note that $s = y_1$ and $u = y_m$), where $(u, v) = b_{w_1,w_2}$ is the first bridge on the high-level path, $S = \{x : c_x = c_s\}$ is the set of all

low-level vertices in the same cluster as s, and $G[S]$ is the subgraph induced in G by S. HBFS then returns y_2, the next vertex to move to from s, or, alternatively, full the path to the next cluster, $(y_2, \ldots, y_m)$.

In an efficient hierarchy, the number of clusters will be $|V'| = O(\sqrt{N})$ and the size of each cluster w will also be $n_w = O(\sqrt{N})$, resulting in $O(\sqrt{N})$ time and memory complexity for HBFS. Note that actually traversing the full low-level path from s to g in G still takes $O(N)$ time; HBFS simply computes the next step, ensuring the agent can progress towards the goal without computing the full low-level path in advance (in our simulations, we actually computed the full path in order to simulate execution in addition to planning). HBFS can straightforwardly extend to deeper hierarchies with $L > 2$, with the corresponding complexity becoming $O(\sqrt[L]{N} + L)$.

Algorithm 1 HBFS(s, g, H, G)

```
path' ← BFS(c_s, c_g, (V', E'))
path ← []
for all (w, z) in path' do
    (u, v) ← b_{w,z}
    S ← {x : c_x = c_s}
    append(path, BFS(s, u, G[S]))
    append(path, (u, v))
    s ← v
end for
S ← {x : c_x = c_s}
append(path, BFS(s, g, G[S]))
return path
```

The pseudocode for HBFS used in our simulations is shown in Algorithm 1. Our particular implementation of HBFS takes as arguments the starting state $s \in V$, the goal state $g \in V$, the hierarchy H and the low-level graph G. The variables c and b refer to the cluster assignments and bridges in H, respectively. Note that s changes after each iteration. We assume the existence of a function BFS which takes as arguments a starting state, a goal state and a graph and returns the shortest path between those states as a list of edges.

Note that we are not making any specific commitments to the cognitive plausibility of HBFS, and that any other hierarchical planner based on shortest paths would make similar predictions. Also note that HBFS could be

straightforwardly extended to deeper hierarchies by introducing a depth level parameter l and recursively calling HBFS instead of BFS on line 1 if $l > 2$. Finally, note that HBFS as implemented here still requires $O(N)$ time and memory as it finds the full path in G. For returning only the first few actions, the for-loop could be interrupted after the first iteration, which would yield the $O(\sqrt[L]{N} + L)$ complexity.

Rewards

To model reward generalization in experiment six, we set $\bar{\theta} = 15$ and $r_{4,1} = 30$, in accordance with the experimental instructions. The node selected by the simulation was the one with the greater (approximate) expected reward $\mathbb{E}[r_s|D] \approx \theta_{c_s}$, where θ are the cluster rewards and c are the cluster assignments in the sampled hierarchy H.

To model cluster inferences based on rewards in experiment seven, we only simulated a single training trial and set $r_{1,1} = r_{2,1} = r_{3,1}, r_{4,1} = r_{5,1} = r_{6,1}$, and $r_{7,1} = r_{8,1} = r_{9,1} = r_{10,1}$, with the specific values chosen at random between 0 and 30 (we scaled down the rewards experienced by participants by a factor of 10). Since the hierarchy discovery algorithm has no notion of maximizing reward (it merely treats rewards as features), the magnitude of the rewards is irrelevant. We did not model the random changes of reward values throughout training, which were introduced purely for control purposes and are irrelevant for hierarchy discovery as framed here. Note that the model could easily accommodate dynamic rewards by assuming drifting μ's, however this would unnecessarily complicate the model without making substantial contributions to the core theoretical predictions. As in experiment six, we set $\bar{\theta} = 15$, the expected reward based on the instructions.

4

Conclusion

In this work, I study how structured probabilistic representations shape reinforcement learning in the human brain. In Paper 1, I investigated the neural correlates of exploration driven by the uncertainty of the values of the available options. The behavioral and neuroimaging data point to hybrid computational architecture in which the relative uncertainty of the available options is encoded in right rostrolateral prefrontal cortex and drives directed exploration, while the total uncertainty of the available options is encoded in right dorsolateral prefrontal cortex and drives random exploration. The ecological advantages of the hybrid model were confirmed by simulations, showing that the combination of both strategies outperforms either strategy alone. I also found evidence that motor cortex integrates these uncertainty estimates to compute choices via a sampling mechanism.

While Paper 1 examined choices in the face of uncertainty about values, in Paper 2 I studied how the brain resolves uncertainty pertaining to the structure of the environment, and in particular, the structure of the latent causal relationships between stimuli and outcomes. The generalization pattern exhibited by our participants was consistent with structure learning as a form of probabilistic inference in the space of causal structures. Those causal structures in turn constrain the learned stimulus-outcome associations and how they are transferred to novel stimuli. The learning signal corresponding to updates of these associations was encoded in a frontoparietal network of regions that were distinct but partially overlapping with the regions encoding the update of beliefs about the causal structure governing the associations. Additional analyses revealed signatures of multivariate representations of that belief in parietal cortex and interior insula.

Both Papers 1 and 2 implemented Bayesian theories of learning in the model-free setting, in which the agent is concerned with predicting and responding to the values of immediately available stimuli and actions. In contrast, many problems require computing multi-step action plans given knowledge of the transition dynamics of the environment, something which falls into the realm of model-based reinforcement learning. Paper 3 is concerned precisely with how an agent might represent these transition dynamics in a way that supports such efficient model-based computations. Across five simulations of previous studies and eight new behavioral experiments, I found evidence that humans build a hierarchical representation of the environment which clusters together low-level states into abstract state chunks in ways consistent with our normative Bayesian model.

This work expands upon the canonical reinforcement learning framework by showing what kinds of structured probabilistic representations are used by the brain, where they are computed and stored, and how they guide action selection. Although the domains and particular structures I consider are of limited scope, they are sufficient to illustrate their basic principles of operation and conceivably tap into generic neural processes for structure learning that extend beyond the toy examples considered here. This is hinted at by Paper 2 in which I found that an alternative structure learning model relying on a different mechanism recruited the same frontoparietal regions in a different subset of participants, suggesting that the same circuits can support different kinds of structure learning. Further studies can use richer domains such as Atari games to investigate whether those same regions and neural mechanisms are involved in more complex forms of probabilistic inference about the structure and dynamics of the world, such as theory learning[296] or program synthesis[171]. The process of hierarchy discovery for state abstraction discussed in Paper 3 could conceivably also rely on the same circuits.

Another set of open questions pertains to how structure learning interacts with the rest of the reinforcement learning circuitry. In particular, on the model-free side, what is the mechanism by which information about structure reaches and modulates value representations in the striatum and reward prediction errors in the midbrain? On the model-based side, how does information about structure reach orbitofrontal cortex and the hippocampus, both thought to represent task structure[216,250]? Further, it is unclear how this information is integrated by downstream decision circuits. While Paper 1 suggests that uncertainty estimates resulting from the inference process might be directly entered into the choice computation in motor cortex, it remains unclear whether this mechanism is specialized only for the case of uncertainty of the value estimates, or whether it is also involved in active learning processes that seek to reduce structural uncertainty.

Overall, the results presented here point to an entire research program that can broaden the scope of this work and answer number of questions regarding structured probabilistic representations in reinforcement learning, namely: 1) how is the model of the world represented?, 2) how is it learned?, 3) how is it used for simulations, planning, and action selection?. From a neuroscience perspective, answering these questions is an essential step towards elucidating the neural circuits of structure learning. From an engineering perspective, answering these

questions can guide the development of sophisticated machine intelligence that learns and performs in ways similar to humans. Much headway has been made in tackling these questions using cleverly designed behavioral experiments [172]. However, since the hypothesis space for the answers to all of these questions is vast, I believe that neural data can be used in addition to behavioral data in order to significantly constrain the search for representations and algorithms that are most consistent with the way humans represent, learn about, and act in the world. I discuss each of those questions in turn, outlining possible solutions and experiments that could arbitrate between them, and speculating about the neural circuitry.

4.1 How does the brain represent the world?

A prevailing view in cognitive science is that the brain builds an internal model of the world which it can use to simulate the possible outcomes of different courses of action in different situations . For example, in Paper 2, the internal model that the agent learns is the causal structure M and the causal strengths $\mathbf{w}$, while in Paper 3, the internal model is the high-level graph H and the low-level graph G. More generally, it is thought that the brain learns a causal model which describes how different unobservable variables in the world influence each other and how they give rise to the observations that are directly available to the agent. Assuming a prior probability distribution over causal models, the agent can use its observations to uncover the underlying causal model of the world using Bayesian inference. The model could be represented by a causal Bayes net, as in Paper 2. More generally, it could be represented by a probabilistic program, in which case the inference process is a form of program synthesis. The relevant question then becomes, what are the fundamental program primitives that are combined to build richer programs that describe the world, and perhaps even more broadly, what is the best probabilistic programming language that can be used to model the world in a way similar to humans?

A number of ongoing projects in our lab are addressing these questions using Atari-style games. In one of those projects, the agent learns the rules of the game by doing inference over the space of possible rule sets (theories) in the game domain. The rules of any game in the domain can be described in a program-like fashion using

the *video game description language* (VGDL[248]). The theories inferred by the agent are also expressed in VGDL. However, there are other possible languages in which games can be expressed[100,179,287,185,192]. There are even different versions of VGDL, each with different representations focusing on different game features. For example, in VGDL 1, theories are factorized into objects (the sprite set), object-object relations (the interaction set), and goals (the termination set). This is just one of many possible factorizations, each having its own advantages and disadvantages.

One way to disambiguate between different ways of representing the game rules is to compare human game play with predictions generated by theories from different game description languages. However, the behavioral data alone would likely be too sparse to distinguish between the different possibilities, especially if they make similar behavioral predictions. Furthermore, in principle, alternative approaches such as meta-reinforcement learning[305,82] could acquire behaviors that are indistinguishable from those predicted by VGDL, and yet they may rely on completely different internal representations.

To distinguish between these possibilities, we can image brain activity while subjects are playing the same games as the different models, and compare representations predicted by the different models using the techniques from Paper 2, such as decoding and representational similarity analysis. This can be performed using fMRI, which will give us spatial precision at the expense of temporal precision. This would be beneficial in comparing the representations in brain regions which we *a priori* hypothesize to encode the causal model, such as rostrolateral prefrontal cortex, the anterior insula (based on Paper 2), or the hippocampus. Using fMRI could also facilitate the discovery of new brain areas that were previously not implicated in structure learning. In particular, it would allow us to assess the theory factorization predicted by VGDL 1 by checking whether the different components of the theory map onto distinct brain regions. The factorization could also be assessed by looking for scalar signals associated with the separate components. In particular, we could look for separate value signals for the sprite set, interaction set, and termination set, which could correspond, for example, to different subregions of the striatum. Similarly, we could look for factorized theory update signals (e.g., the KL divergence, as in Paper 2), which could be localized to posterior parietal cortex. Alternatively, we could use MEG instead of fMRI,

which will give us temporal precision at the expense of spatial precision, giving us more samples and thus greater statistical power for comparing the different representations.

If the space of possible causal models is small enough for the agent to be able to compute the full posterior distribution over causal models (as it is in Paper 2), then we could simply use that probability distribution as the agent's representation of the world at any given time. Yet as soon as the space of possible causal models is anything beyond trivial, exact inference becomes intractable, and therefore the agent must resort to approximations. However, if the agent is using a small set of causal models (perhaps even just a single causal model) to approximate the entire distribution, then it is not obvious what is being represented at any given time. To compare different representations using the neural data, the experimenter must be able to figure out what causal model the agent is using any given time. I refer to this as the problem of inverse structure learning, which I turn to next.

4.2 How does the brain learn a model of the world?

In order to use neural data to compare the different ways in which the brain might be representing the world (e.g., different probabilistic programming languages), we should be able to figure out what particular causal model (e.g., program) the agent is using at any given time. The question about the kinds of representations humans are using is thus intimately tied to the question of what algorithm they are using to learn that representation. Different approximations to Bayesian inference have been proposed as a way to circumvent the intractability issue, from sampling methods (such as Metropolis-Hastings, Hamiltonian Monte Carlo, and particle filters) which approximate the distribution over causal models using a set of samples, to variational methods which approximate the distribution using a simpler parametric distribution. Since simulations and planning usually require a single causal model, I will focus on sampling methods. Such sampling approximations have been shown to be consistent with human behavior across a number of domains[21,175,114].

Returning to the VGDL example, one possibility is to maintain a single theory that is updated using Markov chain Monte Carlo (MCMC) when new data are observed, and to perform action selection based on that the-

ory. Yet there are many ways to do MCMC in the space of theories. Importantly, for a given method, there are many possible Markov chains (that is, sequences of inferred theories) – since the sampling process is stochastic, a given observation may or may not lead to the update of the theory, and if it does, the new theory is not uniquely determined. To find the Markov chain that most closely corresponds to the sequence of theories inferred by the subject, experimenters can use the subject's behavior as an additional constraint – for example, if the way the subject responds to a certain object changes drastically after a given time point, she likely inferred a new theory involving that object at that time point, even though another rational observer might have inferred the new theory earlier.

More generally, the problem of inferring the subject's sequence of causal models for a given representation and inference algorithm can be framed as an inhomogenous hidden Markov model (from the experimenter's point of view), with the agent's observations, actions, and neural data as the experimenter's observations, and agent's causal model as the hidden variable. The fit of the inferred sequence of causal models to the behavioral and neural data could be compared across different learning algorithms to answer the question of how humans learn the causal model of the world (assuming a sampling approximation). This can be extended to include inference (from the experimenter's point of view) over the space of possible representations and inference algorithms used by the agent, which would allow the experimenter to simultaneously address the question of how humans represent the world and how they learn that representation. As before, this can be done using fMRI in order focus on specific brain regions thought to be involved in structure learning, such as posterior parietal cortex (Paper 2), or using MEG for greater statistical power. While this meta-inference problem might seem hopelessly intractable, it is closely related to similar work on inverse rational control[315,53,168] and could be tackled using particle smoothing.

As another example, consider the hierarchy inference process described in Paper 3, in which the inference process is similarly a single MCMC chain corresponding to a sequence of inferred hierarchies, where each hierarchy is updated one component at a time using the Metropolis-Hastings rule. If we take this algorithm seriously as a process model of how the brain does infers the underlying hierarchy, it would be impossible to figure out what

hierarchy a given subject has inferred at a given time if we only run the process in the forward direction, as we do in our simulations. We can address this within the inverse structure learning framework above by additionally considering the subject's behavior, allowing us to get trial-by-trial estimates of the inferred hierarchy for analyzing brain data, or for comparing different inference algorithms based on behavior.

4.3 How is the model used for simulations, planning, and action selection?

Learning a rich sophisticated representation of the world would be futile unless it can be used for action selection. For example, in Paper 3, we propose that the agent uses hierarchical breadth first search (HBFS) to find the shortest path between the starting state and the goal state, and then takes the actions that take it along that path. However, there many possible alternatives, such as hierarchical depth first search, or a hierarchical random walk (that is, random sampling of trajectories), that would make identical behavioral predictions. More generally, if the causal model corresponds to the transition dynamics $T(s'|s,a)$, and if the agent aims to reach a set of goal states, then planning can be thought of as considering different action sequences and using the causal model to simulate what states they would lead to, while action selection would correspond to following the action sequence that leads to the goal states according to the simulations.

Different simulation algorithms have different trade-offs, but many of them might make identical behavioral predictions. We can disambiguate between the different algorithms using human neuroimaging with high temporal resolution, such as MEG or EEG. In particular, we can look at the time period right before an action is taken and try to decode different state sequences that are predicted by the different planning algorithms. Similar methods have been used to show sequence replay using human MEG [178]. For example, in Paper 3, HBFS predicts a forward sweep of states from the starting state in the high-level graph H, followed by a similar forward sweep in the state chunk in the low-level graph G. In contrast, hierarchical depth first search predicts that all states will be activated sequentially from the starting state, one state at a time, first in H and then in G. A random sampler, in contrast, would predict a sequence of random trajectories, again sampled one state at a time.

If the causal model of the world is represented by a program, then simulations would correspond to execution traces, which we could similarly look for in the brain data as particular sequences of neural activity immediately before action selection. In this way, we could also use this method to disambiguate between different probabilistic programming languages, if they predict sufficiently different execution traces. In fact, if the programs are encoded in synaptic weights only (as is predicted, for example, by meta-reinforcement learning[304]), then they could really only be compared based on their execution traces. These neural execution traces should also be systematically linked to subsequent choices, consistent with their role in action selection.

Another question is how action selection can be used to the advantage of the model learning process, that is, to balance the exploration-exploitation trade-off. One possibility is that a kind of curiosity bonus (or pseudo-reward) would be added to certain subgoal states[166], somewhat akin to the uncertainty bonus in Paper 1. Different strategies for selecting subgoals would lead to different patterns of pseudo-reward prediction errors (pseudo-RPEs), which can be measured by looking at neural signals in the ventral striatum and ventromedial prefrontal cortex, regions associated with value coding and RPEs. Alternatively, it could be that the brain is exploring in ways that are maximally informative for the inference process, in line with active learning. This could be tested by looking for neural signals corresponding to the expected entropy of the distribution over causal models for the chosen action.

Overall, this line of research could provide more neurally plausible accounts of structure learning and also pave the way for working out the implementational details at the neural circuit level[188]. It can also enable the integration of the structure learning principles developed here into state-of-the-art reinforcement learning systems, bringing the field a step closer to simulating human-level cognition. Continuing this line of work offers enticing prospects for advancing our understanding of how the brain copes with the complexity and uncertainty of the world, and for replicating that functionality *in silico*.

A

Supplemental Information for Paper 1

A.1 Variance inflation factors

Since the parametric modulators of the trial_onset regressor are correlated (e.g. V and V/TU), we computed variance inflation factors (VIFs) for all four parametric modulators on all runs for all subjects (Figure A.8). Overall, only 5% of VIFs for RU were above the threshold of 10 (12 out of 240), however we found that around 20% of VIFs for TU and V/TU were above 10 (53 and 47 out of 240, respectively), and more than half of VIFs for V were above 10 (128 out of 240). Note that inferences are valid even for regressors with a high VIF since the estimate of the beta coefficients is still unbiased and thus the type I error rate is preserved[204]. However, the inflated variance of the beta estimates might reduce the power of the analysis. To investigate this possibility, we re-analyzed the data using four new GLMs that were nearly identical to GLM 1, with the only difference that each new GLM had a single parametric modulator at trial onset rather than four. Thus there was one GLM with RU, one with TU, one with V, and one with V/TU. As before, we thresholded single voxels at $p < 0.001$. Since we are not using these results for ROI selection, we report uncorrected whole-brain contrasts.

We found the same network of brain regions for RU and TU (Figure A.9A and B, respectively) and no regions for V/TU. For V, we found clusters in medial PFC (Figure A.9C), which is consistent with previous reports of value coding in this region[2,123]. ROI analysis using an anatomically defined vmPFC region (as a conjunction of Superior frontal gyrus, medial orbital; Superior frontal gyrus, medial; and Gyrus rectus from the AAL2 atlas[295,240]) showed a significant positive effect of V in left vmPFC ($t(30) = 2.14, p = 0.04$, t-test of ROI-averaged betas across subjects). Since the value-coding function of this region has been characterized extensively[57,177] and since our primary interest was in the role of uncertainty in guiding exploration, we chose not to pursue this finding.

A.2 Reaction times and decision value

One potential confound of our decision value result (GLM 2) in motor cortex is reaction time (RT). When including RT's as a parametric modulator in addition to DV (GLM 2A), we found no effect of DV in motor cortex

(no voxels survived cluster FWE correction). Note, however, that the sequential sampling framework predicts a strong relationship between DV and RT's: when DV is close to zero, the two options are similar to each other and hence it takes longer for the evidence accumulator to reach a decision bound. This prediction was manifested in our data (coefficient $= -0.006, F(1, 9717) = 23.8, p = 0.000001$, mixed effects linear regression: `RT ~ 1 + DV + (1 + DV | SubjectID)`), indicating that the negative result could be due to RT's capturing some of the shared variance in the BOLD signal.

To account for this possibility, we performed random effects Bayesian model comparison [238] between the GLM with DV alone (GLM 2), the GLM with both DV and RT (GLM 2A), and a GLM with RT alone (GLM 2B) in the left motor cortex ROI identified by GLM 2 (Figure 1.5A). Specifically, following our previous work [292], we approximated the log model evidence as -0.5 * BIC, where the BIC was computed based on the residual variance of the GLM fits within a 10 mm sphere around the peak voxel in left M1 from GLM 2. To prevent circularity [163], we performed this using leave-one-subject-out cross-validation: for each subject, we computed the BIC in the peak ROI from the group-level DV contrast computed using all other subjects. Since `SPM` fits each subject separately, this means that we used independent data for ROI selection and model comparison, resulting in an unbiased analysis. To ensure the validity of our inference, we confirmed that the resulting ROIs were highly overlapping (Figure A.10), with all but one subject having the same left M1 ROI as the contrast using all subjects (Figure 1.5A, MNI [-38 -8 62]). This analysis strongly favored GLM 2A (PXP = 0.96) over GLM 2 (PXP = 0) and GLM 2B (PXP = 0.04). This indicates that the BOLD signal in left M1 is best explained by combination of DV and RT, rather than RT or DV alone, pointing to decision value coding in motor cortex above and beyond RT's.

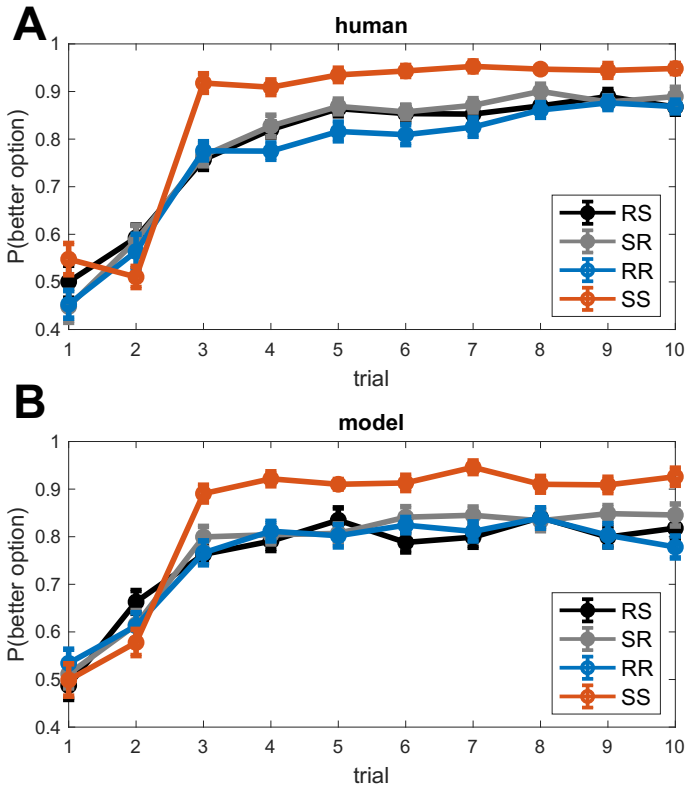


Figure A.1: Learning curves for human (A) and model (B) data. The better option is defined as the option with the greater expected reward $\mu(k)$.

Table A.1: Model comparison between different exploration strategies, which can be thought of as lesioned versions of the UCB/Thompson hybrid model (Eq. 1.4). Lower AIC, BIC, and deviance indicate better fit. AIC = Akaike information criterion; BIC = Bayesian information criterion; LL = maximized log likelihood.

Model	Regressors	AIC	BIC	LL	Deviance
Softmax	V	8414.73	8400.37	-4198.18	8396.37
UCB	V + RU	7982.34	7953.62	-3972.81	7945.62
Thompson sampling	V/TU	8315.20	8300.84	-4148.42	8296.84
UCB/Thompson hybrid	V + RU + V/TU	6655.89	6612.80	-3300.40	6600.80

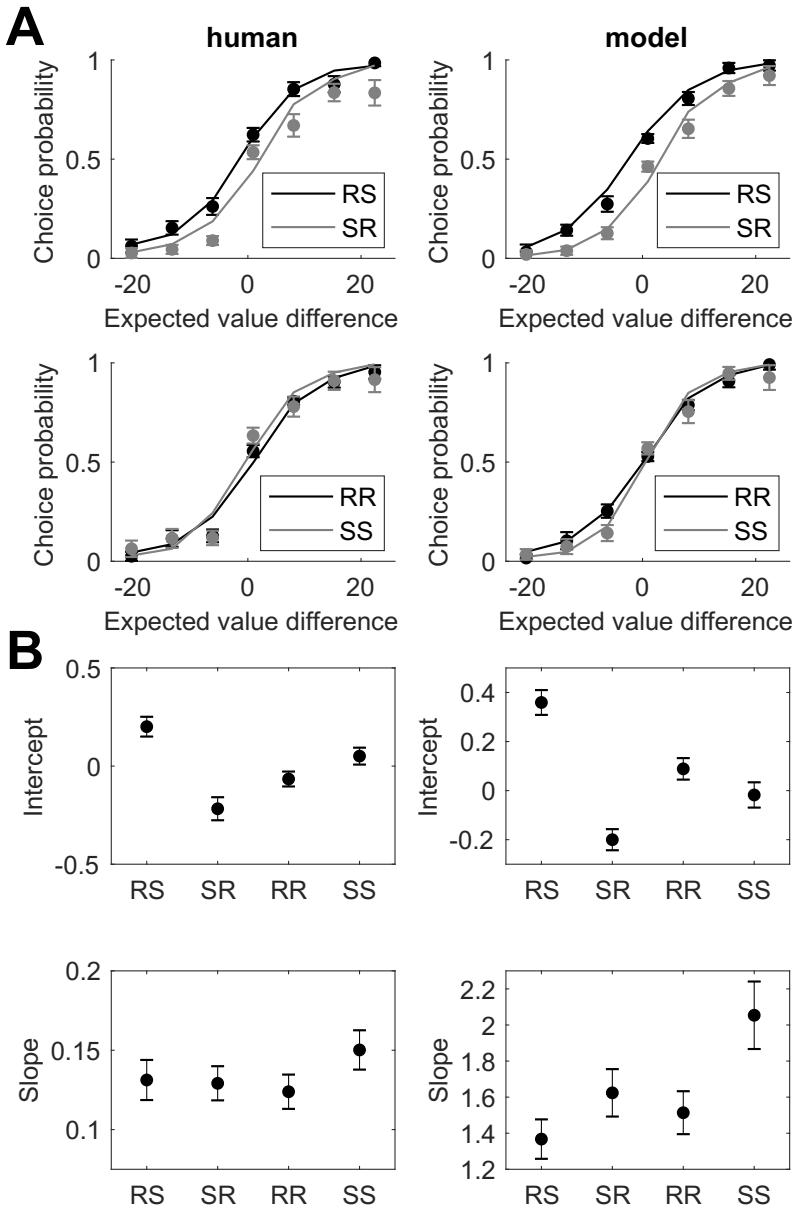


Figure A.2: Choice probability functions (A) and probit regression results (B) for human (left) and model (right) data.

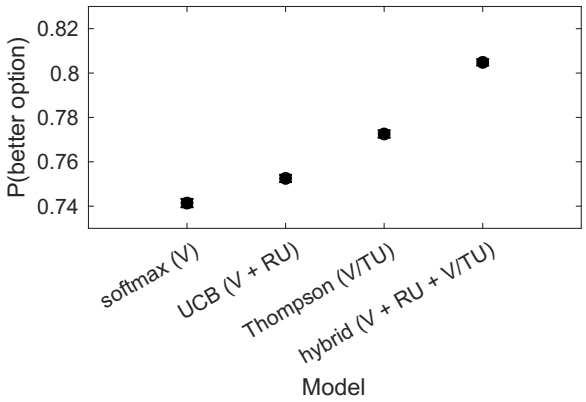


Figure A.3: Performance comparison of different exploration strategies. Simulation results from running different models generatively with subject-specific fitted coefficients. Error bars indicate s.e.m. across simulations.

Table A.2: GLM definitions. GLMs used to analyze the fMRI data in the main text.

Regressor	Event	Duration	Pmods	Which trials
GLM 1: RU, TU, V, V/TU				
trial_onset	trial onset	0 s	$\lvert RU_t \rvert$, TU_t, $\lvert V_t \rvert$, $\lvert V_t \rvert / TU_t$	non-timeout
trial_onset_timeout	trial onset	0 s		timeout
trial_onset_chose_1	trial onset	0 s		chose arm 1
button_press	reaction time	0 s		all
feedback_onset	feedback onset	0 s		all
GLM 2: DV				
trial_onset	trial onset	0 s	$\lvert DV_t \rvert$	non-timeout
trial_onset_timeout	trial onset	0 s		timeout
trial_onset_chose_1	trial onset	0 s		chose arm 1
button_press	reaction time	0 s		all
feedback_onset	feedback onset	0 s		all

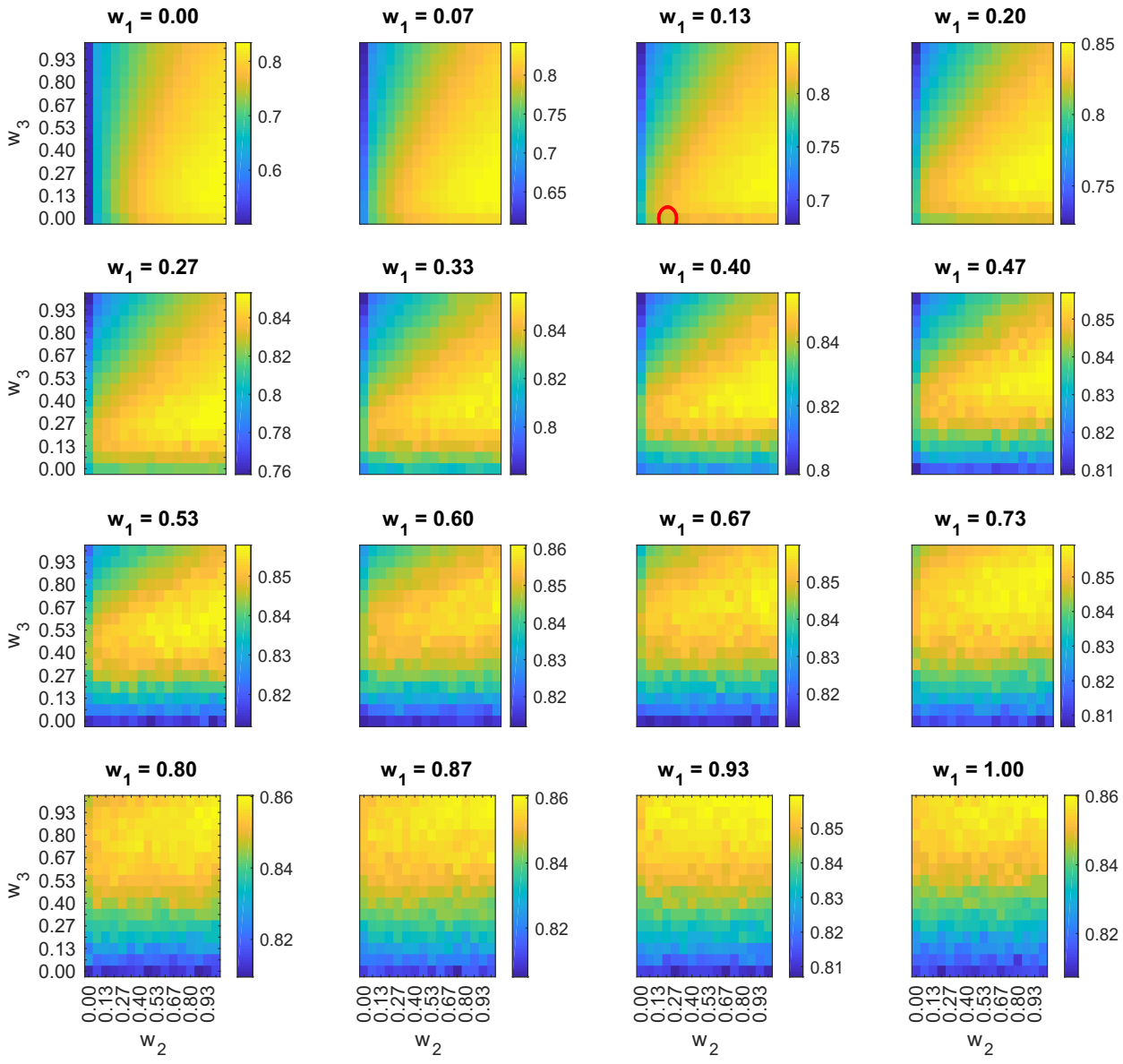


Figure A.4: Performance comparison of simulations of the UCB/Thompson hybrid model (Eq. 1.4) with different parameter settings $\mathbf{w}$. Color scale indicates P(better option), averaged across simulations. Red circle denotes the fitted fixed effects coefficients.

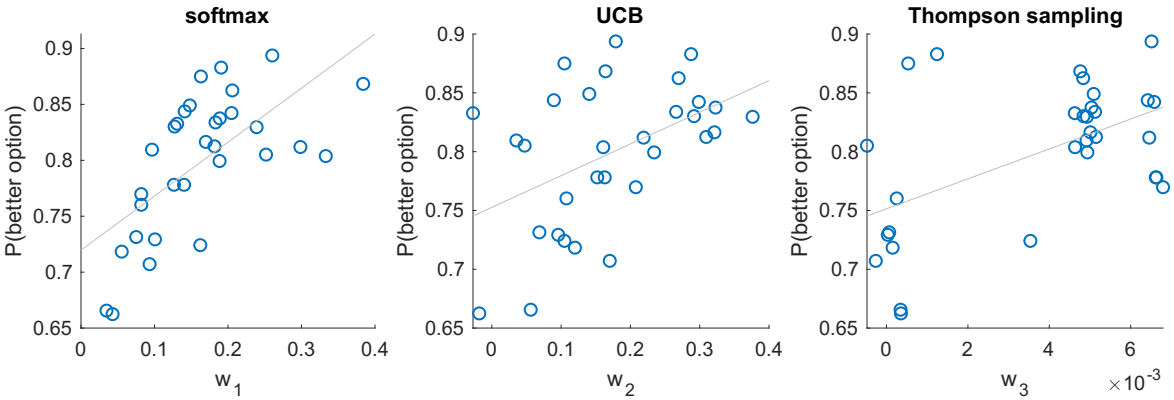


Figure A.5: Subject performance based on exploration strategy. Correlation between subject performance and fitted subject-specific coefficients (Eq. 1.4), indicating greater reliance on the corresponding strategy.

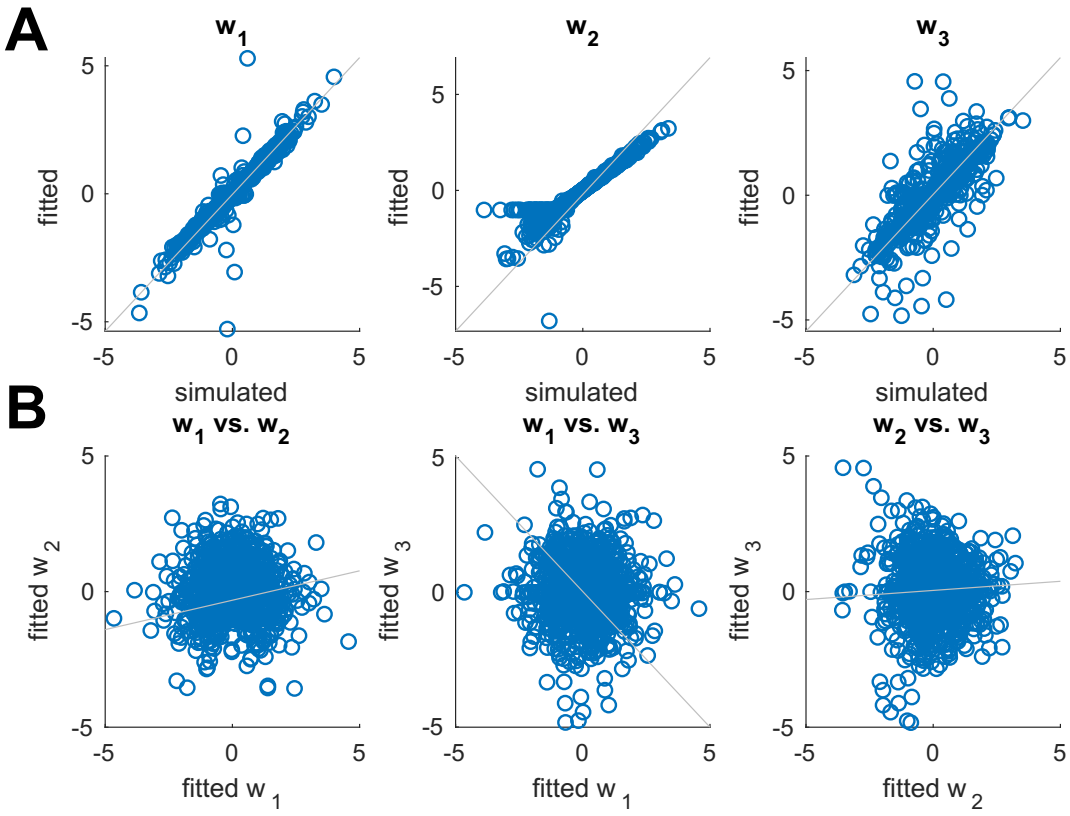


Figure A.6: Parameter recoverability.
(A) Correlation between simulated and fitted parameters.
(B) Correlation between fitted parameters.

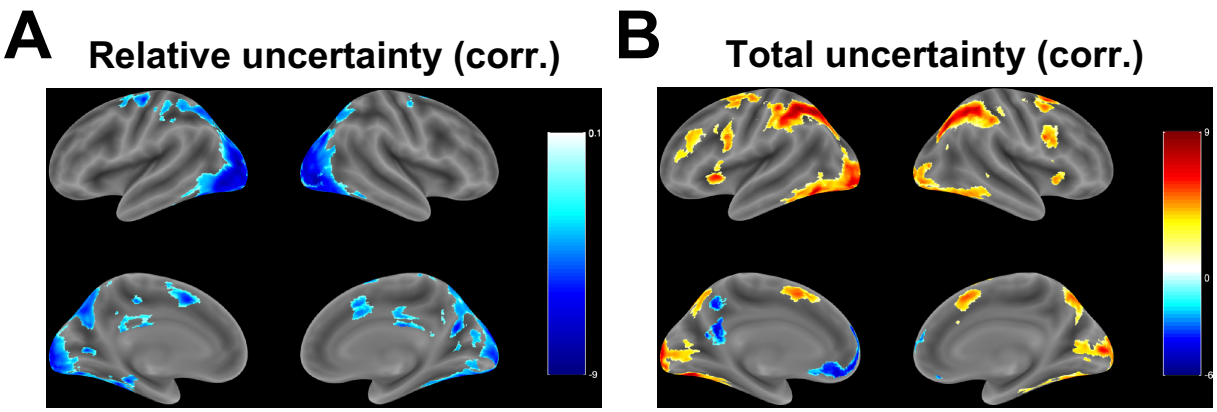


Figure A.7: Corrected GLM 1 contrasts with single voxels thresholded at $p < 0.001$ and cluster FWE correction applied at significance level $\alpha = 0.05$.
(A) Relative uncertainty ($|\mathrm{RU}_t|$) contrast. See Table A.3.
(B) Total uncertainty (TU_t) contrast. See Table A.4.

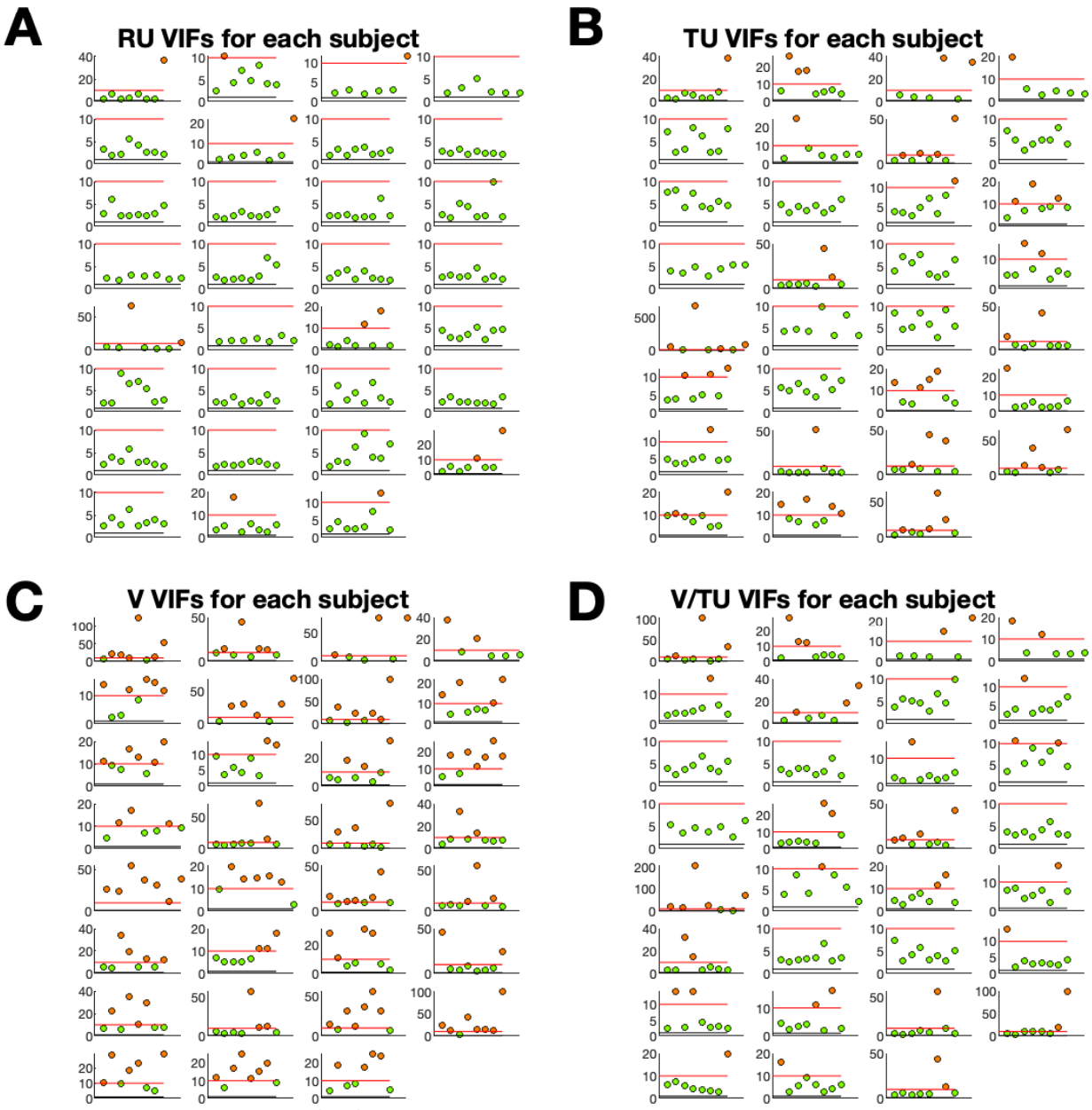


Figure A.8: Variance inflation factors (VIFs) for parametric modulators of the trial onset regressor in GLM 1. Each plot shows the VIFs for all runs of a given subject. Green circles correspond to runs with VIF <= 10, red circles correspond to runs with VIF > 10. A red horizontal line denotes the cutoff at 10.
(A) relative uncertainty (RU),
(B) total uncertainty (TU),
(C) value difference (V),
(D) value difference scaled by total uncertainty (V/TU).

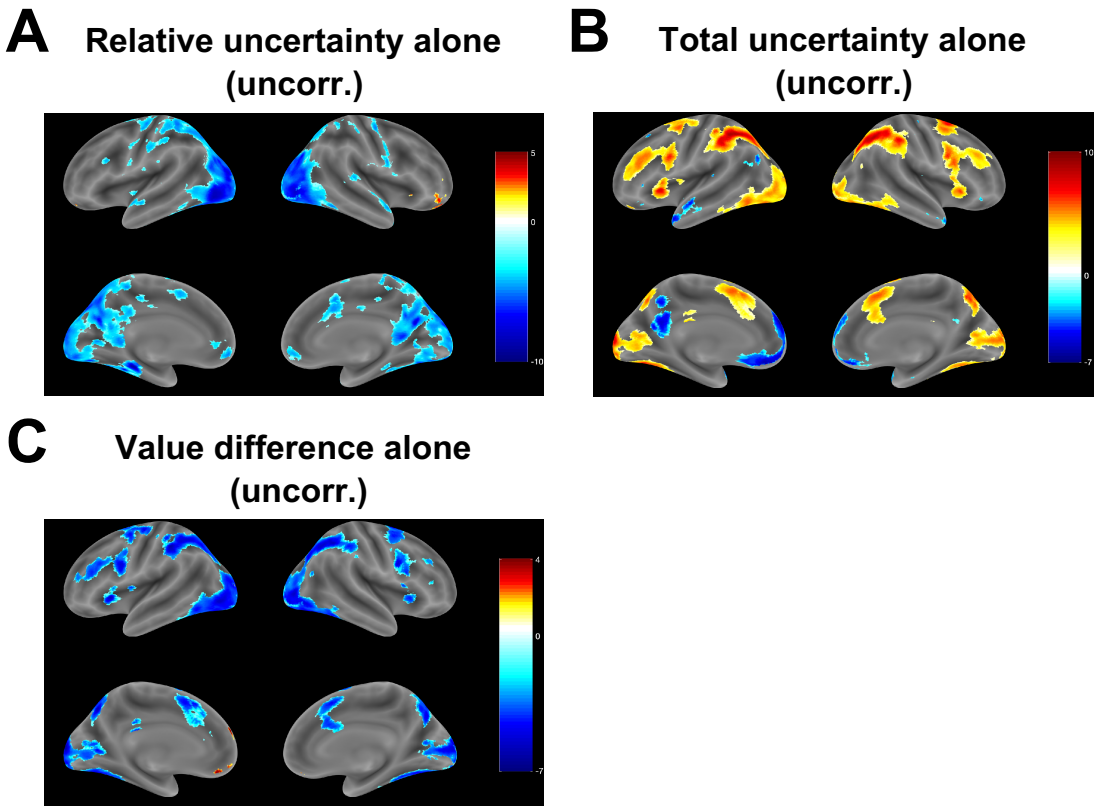


Figure A.9: Contrasts from GLMs with a single parametric modulator.
(A) Uncorrected whole-brain RU contrast when only RU was included as a parametric modulator. Compare with Figure 1.3A.
(B) Uncorrected whole-brain TU contrast when only TU was included as a parametric modulator. Compare with Figure 1.4A.
(C) Uncorrected whole-brain V contrast when only V was included as a parametric modulator.

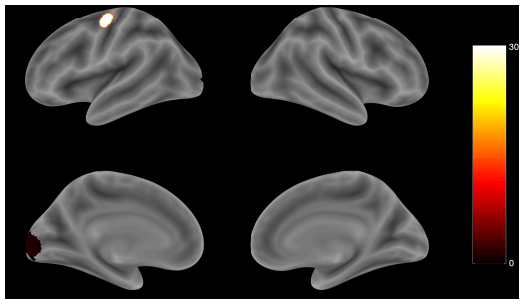


Figure A.10: Heatmap of ROIs from leave-one-subject-out GLM 2 DV contrasts.
Overlay of spherical ROIs around the peak voxel from the group-level DV contrast from GLM 2 using leave-one-subject-out cross-validation. Colorbar indicates how many folds each voxel was part of.

Table A.3: GLM 1 results: relative uncertainty. Brain regions in which the BOLD signal tracks $|\mathrm{RU}_t|$ from GLM 1 (corresponding to Figure A.7A). Anatomical labels and MNI coordinates are based on peak voxels (maximum t-statistic), with up to three peaks extracted per cluster (minimum separation of 20 voxels). Single voxels were thresholded at $p < 0.001$ and whole-brain cluster FWE correction was applied at significance level $\alpha = 0.05$. Regions were labeled using the Automated Anatomical Labeling (AAL2) atlas, the SPM Anatomy Toolbox, and the CMA Harvard-Oxford atlas. MNI = Montreal Neurological Institute; BA = Brodmann area.

Sign	Brain region	BA	Extent	t-value	MNI coord.
Negative	Middle occipital gyrus (L)	18	27153	-9.164	-30 -92 6
	Cerebellum (L)	37	27153	-8.433	-46 -52 -30
	Inferior occipital gyrus (R)	19	27153	-8.398	40 -84 -8
	Precentral gyrus (L)	6	1066	-6.702	-38 -16 68
	Superior frontal gyrus, dorsolateral (L)	6	1066	-4.582	-24 -6 48
	Superior frontal gyrus, dorsolateral (L)	6	1066	-4.075	-20 -10 76
	Supplementary motor area (L)	32	860	-6.231	-10 8 46
	Middle cingulate & paracingulate gyri (R)	32	860	-5.291	10 14 38
	Posterior cingulate gyrus (R)		530	-5.550	4 -22 28
	Left cerebral white matter (L)		530	-5.380	-16 -28 38
	Posterior cingulate gyrus (L)	26	530	-4.700	-6 -42 24
	Middle frontal gyrus (R)		732	-5.197	38 -4 68
	Supplementary motor area (R)	6	732	-5.169	16 2 62

Table A.4: GLM 1 results: total uncertainty. Brain regions in which the BOLD signal tracks TU_t (corresponding to Figure A.7B). Notation and procedures as in Table A.3.

Sign	Brain region	BA	Extent	t-value	MNI coord.
Positive	Inferior parietal gyrus (L)	40	25346	9.793	-42 -40 52
	Inferior parietal gyrus (R)	40	25346	9.362	38 -42 44
	Middle occipital gyrus (R)	19	25346	9.111	32 -66 36
	Precentral gyrus (L)	44	785	7.353	-44 4 30
	Insula (L)	48	614	7.054	-28 20 2
	Precentral gyrus (R)	44	709	7.018	48 8 30
	Thalamus (L)		294	6.099	-4 -16 14
	Insula (R)	47	307	5.369	36 18 0
	Middle frontal gyrus (L)	46	704	5.152	-42 36 36
	Middle frontal gyrus (L)	46	704	4.405	-44 48 14
Negative	Superior frontal gyrus, dorsolateral (L)	10	1650	-6.957	-4 62 30
	Superior frontal gyrus, medial orbital (L)	11	1650	-6.448	-2 58 -8
	Gyrus rectus (L)	11	1650	-5.884	-4 36 -16
	Precuneus (L)	30	436	-5.904	-10 -54 16
	Precuneus (L)	30	436	-5.196	-10 -50 38

Table A.5: GLM 2 results: decision value. Brain regions in which the BOLD signal tracks $|DV_t|$ (corresponding to Figure 1.5A). Notation and procedures as in Table A.3.

Sign	Brain region	BA	Extent	*t*-value	MNI coord.
Negative	Precentral gyrus (L)	6	721	-6.851	-38 -8 62
	Postcentral gyrus (L)	2	721	-4.303	-48 -38 56
	Superior frontal gyrus (L)		721	-4.014	-14 6 74

B

Supplemental Information for Paper 3

B.1 Supplemental Discussion

B.1.1 Cognitive architectures

The earliest attempts to develop a formal theory of planning date back to the work of Newell and Simon [210,211] who laid the foundational concepts of planning and problem-solving both in human and in artificial intelligence research. Simon [259] framed problem-solving as an interaction between the participant and the environment, and planning as a search through a state space that represents the structure of the problem, with operators (or actions) performing transitions between states. In parallel, the seminal work of Miller et al. [197] highlighted the hierarchical organization of human action plans, which they linked to people's highly structured representations of the world. Miller et al. [197] also proposed the existence of a fast-access, limited-capacity working memory that loads information from a large-capacity "dead storage" system, concepts later formalized as the short-term and long-term memory stores in Newell et al. [211]'s production systems. These earlier attempts led to the development of contemporary cognitive architectures such as Soar [209,169] and ACT-R [4,5], which aim to capture all aspects of human cognition. Both of these systems can perform hierarchical problem-solving in complex domains based on subgoals, however the subgoals have to be supplied manually. Chunking (referred to as compilation in ACT-R) is implemented by caching or memoizing the solutions to these subgoals [170].

Our approach builds on concepts developed in this tradition. In our model, hierarchical behavior directly arises from the hierarchical representation of the environment. The two memory systems we assume also mirror the short-term/long-term memory stores in these earlier cognitive accounts. Additionally, the form of action chunking we propose as a future addition to the model (see Future directions) is similar in spirit to the chunking mechanisms in Soar and ACT-R. In principle, our hierarchy discovery method could be integrated with these production systems to allow them to decompose the problem space and identify subgoals automatically.

B.1.2 Information-theoretic approaches

Closely related to our study is work by McNamee et al. [193] proposing an alternative approach to hierarchically decomposing the environment for planning under working memory limitations. Similarly to our proposal, they divide the state space into clusters (or modules) and assume planning first occurs at a high-level (across modules), and is subsequently refined at a lower level (within modules). They define an optimal modularization of the state space as the one which minimizes the expected information-theoretic description length of planning trajectories. Intuitively, this means that the average hierarchical plan in the modularized state space is as simple as possible. Unlike the analysis of Solway et al. [263], this method does not require knowing the optimal behaviors in advance and can also accommodate different task distributions. This implies that it could account for effects based on graph topology and task distribution (see Table 3.2 in the main text). Unlike our model, it does not account for effects of the reward distribution and uncertainty. By framing the process of hierarchy discovery in terms of Bayesian inference, our model can be used to make predictions about how beliefs evolve during learning (see experiment three and Future directions), and how plans and choices will change correspondingly. Performing Bayesian inference incrementally in this way can also be used to investigate the neural correlates of hierarchy discovery and to understand the underlying neural computations (Figure B.1E,F). Indeed, our model does not appeal to a strict definition of optimality (in the sense of producing a hierarchy that is provably optimal), and hence the two approaches can be seen as complementary, with our model explaining how hierarchy is discovered and the analysis of McNamee et al. [193] validating the hierarchies learned by our model and by people.

Another information-theoretic method for clustering state spaces was proposed by Maisto et al. [186]. Their approach relies on an extension of the CRP that allows clustering based on similarity between states, which can be defined via a prespecified kernel function. Using different kernels, they demonstrate clustering based on bottlenecks, goal states, paths, or other aspects of the graph structure. Their approach relies on algorithmic probability theory to define the kernels (see also [77]). This involves precomputing all possible paths between each pair of states, which renders planning unnecessary. Nonetheless, this approach could provide a useful tool to analyze the

optimality of the hierarchies inferred by our model.

B.1.3 Structure learning and other notions of hierarchy

Frank & Badre [94] proposed an alternative notion of hierarchy in terms of action rules at different levels of abstraction [43,45]. In their framework, low-level rules map stimuli to responses, whereas high-level rules dictate which stimulus dimensions are relevant for the low-level rules. For example, a low-level rule might say "if the traffic light is red, don't walk", while a high-level rule might say "when crossing the street, pay attention to the color of the traffic light". This implements a form of *state aggregation*, which in RL refers to the grouping together of different states and then treating them as a single state, for example by assigning the same values and actions to all states in the group [203,260]. Note that this is different from state clustering in our model, which still treats each state within the cluster as distinguishable from the rest. By determining which stimulus dimensions are relevant for responding, the high-level rules implicitly render all states with the same value for the particular stimulus dimension as indistinguishable for the purposes of responding according to low-level rules (for example, "a red light on the sunny day" versus "a red light on a rainy day" both elicit the same response). This drastically reduces the total number of stimulus-response mappings (low-level rules) that need to be learned and allows generalization to previously unseen stimuli.

The connectionist model proposed by Frank & Badre [94] implements error-driven learning at these different levels of abstraction in parallel loops that map onto the corticostriatal hierarchy, with more anterior regions representing rules at increasing levels of abstraction. Despite its ability to account for a range of behavioral and neural data, this approach is fundamentally restricted to learning stimulus-response mappings only, and as such falls into the category of model-free RL since it has no notion of a transition structure over which to plan. As discussed previously, model-free RL – even with state aggregation – cannot account for the results of experiments one through five since the predominant response ($6 \rightarrow 5$) was never reinforced, nor would it support the kind of goal-directed planning presented here. In fact, state aggregation would likely not be possible in these experiments, since there is only a single stimulus dimension (the name of the current station). In their model, the term

hierarchy is used to denote a hierarchy of rules that amounts to a compressed mapping from one stimulus to one action. This is fundamentally different from our notion of a hierarchy, in which a hierarchy of states supports the flexible generation of multistep action plans that achieve distant goals. It also differs from the traditional notion of HRL, which refers to a temporal hierarchy, with options consisting of sequences of primitive actions. While in theory the notion of a high-level rule could be extended to include options, the model they propose can only learn to ignore stimulus dimensions and as such can only compress stimulus-response mappings, whereas to the contrary, options require an expansion of the stimulus-response space.

B.1.4 Partial observability

Closely related to state discovery models is work in RL on partially observable environments[145]. In this scenario, the agent never directly observes its current state but must instead infer it from observations as it interacts with the environment. Formally, the environment is represented by a partially observable Markov decision process (POMDP) in which states, actions and rewards are represented similarly to MDPs, with the key difference that states additionally generate observations. The agent then uses these observations to infer a probability distribution over states – the *belief state* – which it uses for decision making. Building on RL and Bayesian principles, POMDPs provide a normative way to maximize reward under uncertainty. Correspondingly, they have been used to account for a wide range of behavioral and neural results in the animal learning and decision making literature[61,231]. Recently, neurophysiological evidence from rodents[272,273,9] has shown that midbrain dopaminergic firing is consistent with a RL signal computed over such a belief state, thus grounding the POMDP framework in the well-established brain circuits for reward-based learning.

Our model can be seen as an extension of the POMDP framework, with clusters (high-level states in H) acting as hidden states and low-level states in G acting as observations. However, our model differs from the standard POMDP definition in two ways. First, as latent cause models, POMDPs assume observations are independent given the state, whereas our model relies on the relations between states (E) in order to infer the clusters. Second, unlike latent cause models, POMDPs usually assume a prespecified state space, whereas our model allows for a

theoretically unbounded number of clusters, recruiting more clusters as dictated by the data. The second property of our model makes it similar to an infinite POMDP[79], which dynamically expands the state space as more observations are acquired. The first way that our model differs from POMDPs suggests that the analogy between observations and low-level states might be inappropriate, and that our model can be better thought of as a particular kind of infinite hierarchical MDP, in which states are fully observable but there is additional hidden structure which is not observable. Viewed in this way, our model does not support partial observability, a limitation which could be remedied by making the low-level states unobservable and having them generate observations, which would drive inferences about the states, which would in turn drive inferences about the clusters. While this would complicate the inference process, it would bring our model more closely in line with the POMDP framework, making it more applicable in a world in which agents only receive partial information about their state in the environment.

B.1.5 Action chunking and motor sequences

Our work is closely related to the notion of hierarchical control and motor sequencing[242,241,153], which studies the behavioral effects predicted by hierarchical action plans. Our work speaks directly to that literature by proposing one particular way in which the representations that support such hierarchical planning might be learned. Indeed, our hierarchical planner is reminiscent of the tree traversal process described by Rosenbaum et al. [242], and our reaction time analysis suggests that participants indeed executed sequences of actions in accordance with hierarchical motor sequencing, with action plans generated according to the hierarchical representations predicted by our model.

Our work is also intimately related to a broad literature on chunking in sequence learning[244], also referred to as action chunking. Action chunking refers to the "gluing" of consecutive actions that are reinforced repeatedly into a stereotyped action sequence that is executed as a single behavioral unit. One of the most robust findings in the animal learning literature is the emergence of such stereotyped action sequences after extensive training on a particular task. It is thought to occur as control is transferred from a goal-directed system that chooses actions

based on their anticipated consequences to a habitual system that executes entire action sequences in response to perceived stimuli[76].

Action chunking has a distinct neural signature, with bursts of neural activity emerging at key choice points as an animal becomes proficient at a particular task[127,17,142]. This so-called *task-bracketing* activity first appears in prelimbic cortex – often associated with goal-directed behavior – and then gradually shifts to infralimbic cortex and dorsolateral striatum – often associated with habitual behavior[262]. Task-bracketing has also been measured in midbrain dopaminergic neurons that project to striatum[142], with a difference between the fraction of direct and indirect pathway neurons that code for the initiation and termination of action sequences[143]. Neural activity representing action sequence boundaries has also been measured in striatum and prefrontal cortex of macaques[97,68]. Similar task-bracketing activity has also been observed in songbirds[98] and humans[294,135], suggesting a conserved neural mechanism. One interpretation of these results is that task-bracketing activity reflects start/stop signals that gate overtrained action sequences, particularly since it appears to be causally involved in the initiation and termination of action chunks[99].

Executing entire sequences as single behavioral units could be beneficial if the cost of processing the outcome of each action is outweighted by the benefit of acting fast at the risk of making mistakes[69]. In its current form, our model only captures state chunking (the creation of state clusters), however it can straightforwardly accommodate action chunking using some form of caching or memoization (see Future directions). In fact, our model makes a distinct prediction about the structure of action chunks, namely that they will fall within the boundaries defined by state chunks (Figure B.1C in the Supplemental Material). In other words, we predict that state abstraction will drive temporal abstraction: agents will first carve up their environment into clusters of states (state chunks), which in turn will constrain the sequences of actions (action chunks) that are learned, which in turn will operate within the state chunks. This stands in contrast to accounts which assume that the agent first learns useful action sequences and then learns state representations consistent with those sequences post-hoc[157]. This translates into specific predictions about the neural activity of brain regions thought to support action chunking (see Neural Implementation in the Supplemental Material).

B.1.6 Neural implementation

We speculate about how the computational processes proposed here might be implemented in neural circuits, and which brain regions might perform the computations. We also simulate within-trial and across-trial neural signals that could be used to identify the key brain areas involved in hierarchy discovery and hierarchical planning.

First, consider the flat graph G only. A straightforward way to encode G in a neural circuit would be to have a single unit (a neuron, such as a place cell, or an ensemble of neurons) represent each node $u \in V$ and excitatory synapses between pairs of units (u, v) represent the edges E (Figure B.1A, bottom). The graph structure could be learned via local Hebbian plasticity: when two units are activated right after each other (for example, during a transition between the corresponding states), the synapse between them is potentiated. In order to perform (flat) BFS to find the shortest path from node s to node g, an external input can successively probe each neighbor of s by transiently activating the corresponding unit, triggering a "forward sweep" of activation that propagates through the circuit until it reaches and activates the unit of the goal state g. The neighbor of s that activates g in the least amount of time is the next node along the shortest path to g, so the agent can then physically transition to that state and repeat the process again and again, until finally reaching the goal state g. Assuming some form of short-term synaptic depression or ion channel inactivation that prevents the sweep from going backwards[75], this implements precisely BFS. Similar schemes have been proposed to support forward trajectory planning in the hippocampus[85,119].

The hierarchical graph H could then be incorporated into the circuit by designating its own set of units and synapses, corresponding to the nodes V' and edges E', respectively. The cluster assignments c could also be implemented as synapses between the units of G and the units of H. In an elegant way, the inferred hierarchy would be isomorphic to the neural circuit that represents it (Figure B.1A). This could straightforwardly extend to deeper hierarchies and is consistent with the presence of place cells and grid cells with different receptive field sizes in the hippocampus and entorhinal cortex[144,151]. HBFS can be implemented in the exact same way as BFS, with the difference that now the forward sweep can take "shortcuts" through the higher levels of the hierarchy, thus signif-

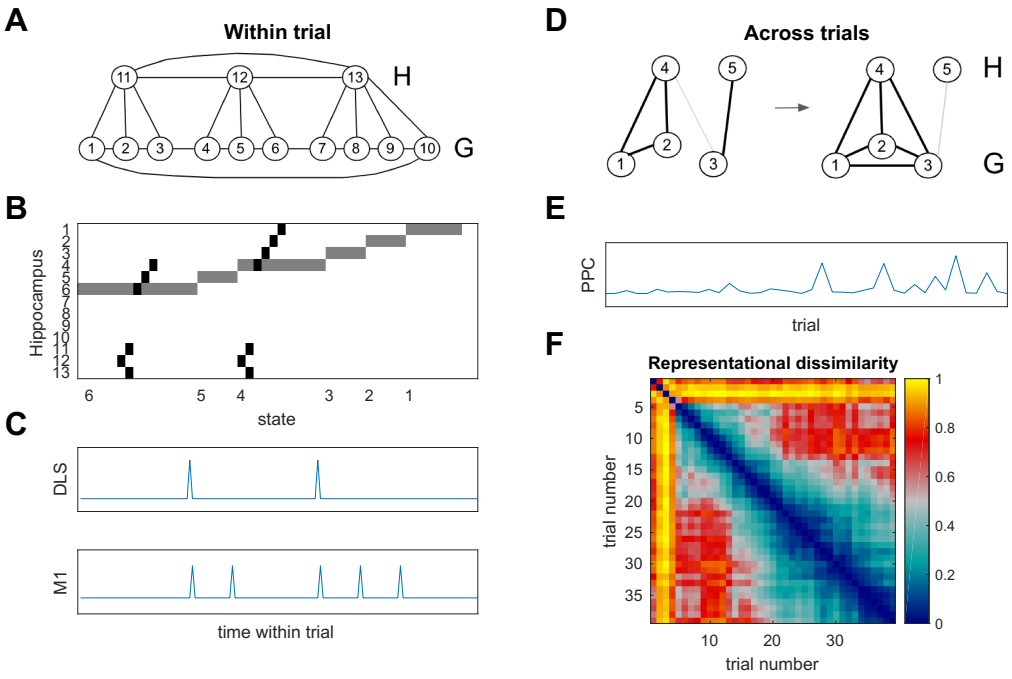


Figure B.1: Hierarchy discovery and hierarchical planning in the brain.

A. Example neural circuit encoding the low-level graph G (bottom), the high-level graph H (top), and the cluster assignments c from experiment one (Figure 3.9A). Circles denote units representing graph nodes. Lines denote bidirectional excitatory synapses representing edges and cluster assignments. Number are unit identifiers.

B. Example (idealized) circuit activity during the test trial $6 \rightarrow 1$. Each row represents the activity of the corresponding unit over the course of the trial. States along the X-axis denote the current state following a transition. Gray denotes intermediate levels of activation representing the current state of the agent, akin to hippocampal place cell activity. Black denotes high levels of activation during planning, akin to hippocampal preplay.

C. (Top) Example (idealized) "start" activity at the initiation of each action chunk in dorsolateral striatum (DLS), with action chunks assumed to fall within the boundaries of the state chunks (clusters) in A. (Bottom) For comparison, primary motor cortex (M1) activity at key presses corresponding to transitions. Time course corresponds to B.

D. Example neural circuit illustrating hierarchy discovery via local Hebbian plasticity. (Left) Low-level graph with a single edge (1,2) has nodes 1 and 2 assigned to cluster 4 and node 3 is assigned to cluster 5. (Right) Observing edges (1,3) and (2,3) causes transient activation of nodes 1,2,3 and cluster 4, strengthening the connection between node 3 and cluster 4 and hence reassigning node 3 to cluster 4.

E. Simulated Bayesian update of the (approximate) posterior $P(H|D)$ over the course of learning the graph from simulation four (Figure 3.7A), which could take place in posterior parietal cortex (PPC).

F. Representational dissimilarity matrix showing the difference in the (approximate) posterior $P(H|D)$ between pairs of trials during the same simulation as in E.

icantly reducing the time it would take to activate the goal unit g. Note that this performs almost the exact same computation as HBFS, with the small difference that transitions "up" the hierarchy (i.e., computing c_u and c_v on line 1 in Algorithm 1) also count as transitions along the path.

The hierarchy could be learned similarly to G, with local Hebbian plasticity strengthening the synapses between units of H (for example, during boundary transitions) as well as the synapses representing the cluster assignments between G and H. To illustrate this, consider the example in Figure B.1D (left), with units 1,2,3 representing nodes in G and units 4,5 representing nodes in H. On the left, there is only one edge between nodes 1 and 2 and hence these two nodes are clustered together ($c_1 = c_2 = 4$), separately from node 3 ($c_3 = 5$). However, once the edges (1,3) and (2,3) are observed (Figure B.1D, right), this would transiently activate units 1,2,3 together, which (because of nodes 1 and 2) would transiently activate unit 4, leading to potentiation of the synapse between 3 and 4. Assuming some form of local homeostatic plasticity that constrains the total synaptic weight of each unit, this would weaken the synapse between 3 and 5, effectively reassigning 3 to the same cluster as 1 and 2 ($c_1 = c_2 = c_3 = 4$).

Note that this implements a kind of "soft" hierarchy, with the same node potentially being strongly associated with one cluster and weakly associated with other clusters. This could be one way to take into account a probability distribution over hierarchies rather than a single point estimate. Indeed, allowing all synaptic weights to have continuous values rather than forcing them to be binary can keep track of probability distributions over the edges E and E' as well. In fact, this could also allow the neural circuit to take into account stochastic transitions: transitions that have low probability will simply have the corresponding synapses potentiated less often, resulting in weaker weights. This addresses the limitation of our graph-theoretic approach to only support deterministic transitions, thus extending the framework to support regular MDPs. The continuous range of the synaptic weights would naturally be taken into account by our "neural" HBFS algorithm: the forward sweep will simply be less likely to propagate through weaker synapses. In effect, this will perform a kind of simultaneous, parallel sampling of an entire set of possible trajectories in a way that is drastically more efficient than sampling trajectories one by one (linear versus exponential time). Investigating the theoretical properties of such a mechanism

could be the subject of future work.

A neural circuit with the above-mentioned properties could naturally be implemented in the hippocampus and the surrounding cortex. Hippocampus has long been known to encode locations in physical space[215] and has been hypothesized to encode a cognitive map that applies across various non-spatial domains[216]. Recent studies have shown that this is indeed the case, with encoding of non-spatial task-relevant variables such as sound frequency in rodents[6] and even abstract conceptual domains in humans[46]. The units we hypothesize could thus be implemented in the hippocampus-entorhinal circuit, with HBFS taking the form of hippocampal preplay (Figure B.1B) which is known to occur at decision points and is predictive of future behavior[81].

While our model only discovers state chunks, action chunking could straightforwardly be incorporated by caching (or memoizing) the output of BFS and/or HBFS for regularly occurring subgoals (see Future directions). The acquisition of action chunks after extensive training in animals is associated with the emergence of characteristic start/stop signals in basal ganglia circuits[127,17,142,143,99]. Our model makes the distinct prediction that action chunks and the corresponding start/stop signals will fall within state chunk boundaries (Figure B.1C), rather than be dictated purely by the temporal statistics of action sequences. This prediction could be validated empirically by subjecting animals to similar training and test protocols as our participants while measuring neural activity in dorsolateral striatum.

If the probability distribution over hierarchies is implicitly encoded in synaptic weights, as proposed above, then it would be difficult to read it out directly from neural activity. Alternatively, the distribution could be encoded in neural activity patterns, for example using probabilistic population codes or neuronal sampling[229] . This would be consistent with our previous work[292] which found a neural signature of the Bayesian update of the posterior over hidden structures in a frontroparietal network of brain regions, as well as representations of the full posterior in several brain areas. Our model falls within the framework of structure learning and is thus likely to recruit the same underlying neural mechanisms. This prediction can be tested by generating regressors that track Bayesian updates of the posterior $P(H|D)$ during learning (Figure B.1E) and using them to identify neurons or brain areas that might implement the actual hierarchy discovery process. In addition, representa-

tional similarity analysis [162] could be used to identify areas that maintain the (approximate) posterior $P(H|D)$ (Figure B.1F).

B.2 Supplemental Methods

B.2.1 Active learning

Drawing on the active learning framework from the causal inference literature [205,293], we assume the agent will chose to learn about edges of G in a way that provides maximal information about H. Maximizing information about H is equivalent to minimizing uncertainty about H, which can be quantified as the entropy of H:

$$\mathbb{H}(H|D) = -\sum_{H_{disc}} \int P(H|D) \log P(H|D) dH_{cont} \tag{B.1}$$

Where $H_{disc} = (V', E', c, b)$ are the discrete components of H, $H_{cont} = (p', p, q)$ are the continuous components of H, and D is the data observed so far. We use $\mathbb{H}$ to denote the entropy of a mixed random variable with discrete and continuous components [207].

Computing the entropy in this way is neither computationally feasible nor psychologically plausible. Following previous authors [275], we assume the agent has a subjective probability distribution over possible hierarchies H which can be represented by a set of samples $[H^{(1)}, \ldots, H^{(M)}]$ with multinomial probabilities $[p^{(1)}, \ldots, p^{(M)}]$ ($\sum_m p^{(m)} = 1$). If the samples are drawn from the posterior $P(H|D)$, the agent can approximate the entropy as:

$$\mathbb{H}(H|D) \approx -\sum_m p^{(m)} \log p^{(m)} \tag{B.2}$$

Note that while this is not a proper estimate of the entropy, it can serve as a basis for rational hypothesis testing. In our simulations, we used MCMC to generate the samples from the (approximate) posterior and set the

subjective probabilities according to $p^{(m)} \propto P(H^{(m)}|D) \propto P(D|H^{(m)})P(H^{(m)})$.

We use $a_{u,v}$ to denote the action of observing edge (u, v), i.e. finding out whether $(u, v) \in E$. Since there is no way to know in advance what the outcome would be, the agent has to minimize the expected entropy over the two possible outcomes:

$$\begin{aligned}\mathbb{H}(H|D, a_{u,v}) = \mathbb{H}(H|D, (u, v) \in E)Pr[(u, v) \in E|D] \\ + \mathbb{H}(H|D, (u, v) \notin E)Pr[(u, v) \notin E|D]\end{aligned} \tag{B.3}$$

We can compute the probability of each outcome by marginalizing over H and using the sampling approximation:

$$Pr[(u, v) \in E|D] = \sum_{H_{disc}} \int Pr[(u, v) \in E|H]P(H|D)dH_{cont} \tag{B.4}$$

$$\approx \frac{1}{M} \sum_{m} Pr[(u, v) \in E|H^{(m)}] \tag{B.5}$$

Where $Pr[(u, v) \in E|H]$ is p or pq, according to the generative model. $Pr[(u, v) \notin E|D]$ is approximated analogously.

The agent then chooses the action that minimizes the expected entropy:

$$a = \operatorname*{argmin}_{a} \mathbb{H}(H|D, a) \tag{B.6}$$

B.2.2 Neural simulations

The example within-trial circuit activations in Figure B.1B (see Supplemental Material) were generated manually, assuming a hierarchy like the one in Figure B.1A (responding to the decomposition in Figure 3.9A). We assumed HBFS is executed at every cluster boundary and that the entire path within the cluster is traversed in a single action sequence that is executed as a single behavioral unit, akin to an action chunk.

To generate the Bayesian update in Figure B.1D, we simulated online inference on the graph from the Towers of Hanoi puzzle (Figure 3.7A), using a particle filter with $M = 100$ particles $[H^{(1)}, \ldots, H^{(M)}]$, each initialized from the prior $P(H)$. We started with an empty graph and added edges one by one, with single edge added on each trial. We approximated the posterior $P(H|D)$ with multinomial probabilities $[p^{(1)}, \ldots, p^{(M)}]$, where $p^{(m)} \propto P(H^{(m)}|D) \propto P(D|H^{(m)})P(H^{(m)})$ and $\sum_m p^{(m)} = 1$.

Following our previous work[292], we quantified the Bayesian update after each observed edge by computing the Kullback-Liebler divergence between the multinomial approximation to the posterior before and after the update. We additionally performed 10 iterations of MCMC (as described in the inference section) for each particle after each trial in order to rejuvenate the particles[38]. Similar approximations to online Bayesian inference have been used in previous studies[1]. To generate the dissimilarity matrix in Figure B.1E, as in our previous work[292], we computed the cosine distance between the approximate posterior for each pair of trials in the simulation.

Data and code for all simulations and experiments are freely available at https://github.com/tomov/chunking.

References

[1] Abbott, J. T. & Griffiths, T. L. (2011). Exploring the influence of particle filter parameters on order effects in causal learning. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 33.

[2] Alexander, W. H. & Brown, J. W. (2011). Medial prefrontal cortex as an action-outcome predictor. *Nature neuroscience*, 14(10), 1338.

[3] Alink, A., Walther, A., Krugliak, A., van den Bosch, J. J., & Kriegeskorte, N. (2015). Mind the drift-improving sensitivity to fmri pattern information by accounting for temporal pattern drift. *bioRxiv*, (pp. 032391).

[4] Anderson, J. (1993). Rules of the mind.

[5] Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological review*, 111(4), 1036.

[6] Aronov, D., Nevers, R., & Tank, D. W. (2017). Mapping of a non-spatial dimension by the hippocampal–entorhinal circuit. *Nature*, 543(7647), 719.

[7] Auer, P. (2002). Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov), 397–422.

[8] Auer, P., Cesa-Bianchi, N., & Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3), 235–256.

[9] Babayan, B. M., Uchida, N., & Gershman, S. J. (2018). Belief state representation in the dopamine system. *Nature communications*, 9(1), 1891.

[10] Badre, D. & D'Esposito, M. (2007). Functional magnetic resonance imaging evidence for a hierarchical organization of the prefrontal cortex. *J Cogn Neurosci*, 19(12), 2082–2099.

[11] Badre, D., Doll, B. B., Long, N. M., & Frank, M. J. (2012). Rostrolateral prefrontal cortex and individual differences in uncertainty-driven exploration. *Neuron*, 73(3), 595–607.

[12] Badre, D., Kayser, A. S., & D'Esposito, M. (2010). Frontal cortex and the discovery of abstract action rules. *Neuron*, 66(2), 315–326.

[13] Balaguer, J., Spiers, H., Hassabis, D., & Summerfield, C. (2016). Neural mechanisms of hierarchical planning in a virtual subway network. *Neuron*, 90(4), 893–903.

[14] Balaz, M. A., Capra, S., Hartl, P., & Miller, R. R. (1981). Contextual potentiation of acquired behavior after devaluing direct context-us associations. *Learning and Motivation*, 12, 383–397.

[15] Balleine, B. W. (2005). Neural bases of food-seeking: affect, arousal and reward in corticostriatolimbic circuits. *Physiology & behavior*, 86(5), 717–730.

[16] Balleine, B. W. & Dickinson, A. (1998). Goal-directed instrumental action: contingency and incentive learning and their cortical substrates. *Neuropharmacology*, 37(4-5), 407–419.

[17] Barnes, T., Kubota, Y., Hu, D., Jin, D., & Graybiel, A. (2005). Activity of striatal neurons reflects dynamic encoding and recoding of procedural memories. *Nature*, 437, 1158–61.

[18] Beharelle, A. R., Polanía, R., Hare, T. A., & Ruff, C. C. (2015). Transcranial stimulation over frontopolar cortex elucidates the choice attributes and neural mechanisms used to resolve exploration–exploitation trade-offs. *Journal of Neuroscience*, 35(43), 14544–14556.

[19] Bellman, R. (1957a). *Dynamic Programming*. Princeton, NJ, USA: Princeton University Press.

[20] Bellman, R. (1957b). A markov decision process. *Journal of Mathematical Mechanics*.

[21] Bonawitz, E., Denison, S., Griffiths, T. L., & Gopnik, A. (2014). Probabilistic models, learning algorithms, and response variability: sampling in cognitive development. *Trends in cognitive sciences*, 18(10), 497–500.

[22] Boorman, E. D., Behrens, T. E., Woolrich, M. W., & Rushworth, M. F. (2009a). How green is the grass on the other side? Frontopolar cortex and the evidence in favor of alternative courses of action. *Neuron*, 62(5), 733–743.

[23] Boorman, E. D., Behrens, T. E., Woolrich, M. W., & Rushworth, M. F. (2009b). How green is the grass on the other side? Frontopolar cortex and the evidence in favor of alternative courses of action. *Neuron*, 62(5), 733–743.

[24] Botvinick, M. & Weinstein, A. (2014). Model-based hierarchical reinforcement learning and human action control. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 369(1655), 20130480.

[25] Botvinick, M., Weinstein, A., Solway, A., & Barto, A. (2015). Reinforcement learning, efficient coding, and the statistics of natural tasks. *Current Opinion in Behavioral Sciences*, 5, 71 – 77. Neuroeconomics.

[26] Botvinick, M. M. (2012). Hierarchical reinforcement learning and decision making. *Current opinion in neurobiology*, 22(6), 956–962.

[27] Botvinick, M. M., Niv, Y., & Barto, A. C. (2009). Hierarchically organized behavior and its neural foundations: a reinforcement learning perspective. *Cognition*, 113(3), 262–280.

[28] Bouton, M. E. & Bolles, R. (1993). Contextual control of the extinction of conditioned fear. *Learning and Motivation*, 10, 445–466.

[29] Bouton, M. E. & King, D. A. (1983). Contextual control of the extinction of conditioned fear: Tests for the associative value of the context. *J Exp Psychol: Anim Behav Process*, 9, 248–265.

[30] Bouton, M. E. & Peck, C. A. (1989). Context effects on conditioning, extinction, and reinstatement in an appetitive conditioning preparation. *Anim Learn Behav*, 17, 188–198.

[31] Busemeyer, J. R. & Townsend, J. T. (1993). Decision field theory: a dynamic-cognitive approach to decision making in an uncertain environment. *Psychological review*, 100(3), 432.

[32] Cai, M. B., Schuck, N. W., Pillow, J. W., & Niv, Y. (2016). A Bayesian method for reducing bias in neural representational similarity analysis. In *Adv Neural Inf Process Syst* (pp. 4951–4959).

[33] Chan, S. C. Y., Niv, Y., & Norman, K. A. (2016). A probability distribution over latent causes, in the orbitofrontal cortex. *J Neurosci*, 36(30), 7817–7828.

[34] Chapelle, O. & Li, L. (2011). An empirical evaluation of Thompson sampling. In *Advances in neural information processing systems* (pp. 2249–2257).

[35] Chater, N., Tenenbaum, J. B., & Yuille, A. (2006). Probabilistic models of cognition: Conceptual foundations.

[36] Chen, C., Takahashi, T., Nakagawa, S., Inoue, T., & Kusumi, I. (2015). Reinforcement learning in depression: a review of computational research. *Neuroscience & Biobehavioral Reviews*, 55, 247–267.

[37] Chentanez, N., Barto, A. G., & Singh, S. P. (2005). Intrinsically motivated reinforcement learning. In L. K. Saul, Y. Weiss, & L. Bottou (Eds.), *Advances in Neural Information Processing Systems 17* (pp. 1281–1288). MIT Press.

[38] Chopin, N. (2002). A sequential particle filter method for static models. *Biometrika*, 89(3), 539–552.

[39] Clark, A. (2013). Whatever next? predictive brains, situated agents, and the future of cognitive science. *Behavioral and brain sciences*, 36(3), 181–204.

[40] Cohen, J. D., McClure, S. M., & Angela, J. Y. (2007). Should i stay or should i go? how the human brain manages the trade-off between exploitation and exploration. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 362(1481), 933–942.

[41] Cohen, J. Y., Haesler, S., Vong, L., Lowell, B. B., & Uchida, N. (2012). Neuron-type-specific signals for reward and punishment in the ventral tegmental area. *nature*, 482(7383), 85.

[42] Collins, A. G., Cavanagh, J. F., & Frank, M. J. (2014). Human EEG uncovers latent generalizable rule structure during learning. *J Neurosci*, 34, 4677–4685.

[43] Collins, A. G. E. & Frank, M. J. (2013). Cognitive control over learning: Creating, clustering, and generalizing task-set structure. *Psychol Rev*, 120, 190–229.

[44] Collins, A. G. E. & Frank, M. J. (2016a). Neural signature of hierarchically structured expectations predicts clustering and transfer of rule sets in reinforcement learning. *Cognition*, 152, 160–169.

[45] Collins, A. G. E. & Frank, M. J. (2016b). Neural signature of hierarchically structured expectations predicts clustering and transfer of rule sets in reinforcement learning. *Cognition*, 152, 160–169.

[46] Constantinescu, A. O., Oâ□□Reilly, J. X., & Behrens, T. E. (2016). Organizing conceptual knowledge in humans with a gridlike code. *Science*, 352(6292), 1464–1468.

[47] Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introduction to algorithms*. MIT press.

[48] Cousineau, D. (2005). Confidence intervals in within-subject designs: A simpler solution to Loftus and Masson's method. *Tutor Quant Methods Psychol*, 1(1), 42 – 45.

[49] Cowan, N. (2001). The magical number 4 in short-term memory: A reconsideration of mental storage capacity. *The Behavioral and brain sciences*, 24, 87–114; discussion 114.

[50] Şimşek, O. & Barto, A. G. (2008). Skill characterization based on betweenness. In *Proceedings of the 21st International Conference on Neural Information Processing Systems*, NIPS'08 (pp. 1497–1504). USA: Curran Associates Inc.

[51] Şimşek, O., Wolfe, A. P., & Barto, A. G. (2005). Identifying useful subgoals in reinforcement learning by local graph partitioning. In *Proceedings of the 22Nd International Conference on Machine Learning*, ICML '05 (pp. 816–823). New York, NY, USA: ACM.

[52] Daniel, C., van Hoof, H., Peters, J., & Neumann, G. (2016). Probabilistic inference for determining options in reinforcement learning. *Machine Learning*, 104(2), 337–357.

[53] Daptardar, S., Schrater, P., & Pitkow, X. (2019). Inverse rational control with partially observable continuous nonlinear dynamics. *arXiv preprint arXiv:1908.04696*.

[54] D'Ardenne, K., McClure, S. M., Nystrom, L. E., & Cohen, J. D. (2008). Bold responses reflecting dopaminergic signals in the human ventral tegmental area. *Science*, 319(5867), 1264–1267.

[55] Dasgupta, I., Schulz, E., & Gershman, S. J. (2017). Where do hypotheses come from? *Cognitive psychology*, 96, 1–25.

[56] Daw, N. D., Niv, Y., & Dayan, P. (2005). Uncertainty-based competition between prefrontal and dorsolateral striatal systems for behavioral control. *Nature neuroscience*, 8(12), 1704.

[57] Daw, N. D., O'doherty, J. P., Dayan, P., Seymour, B., & Dolan, R. J. (2006a). Cortical substrates for exploratory decisions in humans. *Nature*, 441(7095), 876.

[58] Daw, N. D., O'Doherty, J. P., Dayan, P., Seymour, B., & Dolan, R. J. (2006b). Cortical substrates for exploratory decisions in humans. *Nature*, 441(7095), 876–879.

[59] Dayan, P. (1993). Improving generalization for temporal difference learning: The successor representation. *Neural Computation*, 5(4), 613–624.

[60] Dayan, P. (2009). Dopamine, reinforcement learning, and addiction. *Pharmacopsychiatry*, 42(S 01), S56–S65.

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

[62] Dayan, P. & Hinton, G. E. (1993). Feudal reinforcement learning. In S. J. Hanson, J. D. Cowan, & C. L. Giles (Eds.), *Advances in Neural Information Processing Systems 5* (pp. 271–278). Morgan-Kaufmann.

[63] Dayan, P., Hinton, G. E., Neal, R. M., & Zemel, R. S. (1995). The helmholtz machine. *Neural computation*, 7(5), 889–904.

[64] Dayan, P. & Kakade, S. (2000). Explaining away in weight space. In *Proc 13th Int Conf Neural Inf Process Syst*, NIPS'00 (pp. 430–436). Cambridge, MA, USA: MIT Press.

[65] Dayan, P. & Kakade, S. (2001). Explaining away in weight space. In *Advances in neural information processing systems* (pp. 451–457).

[66] Denison, S., Bonawitz, E., Gopnik, A., & Griffiths, T. L. (2013). Rational variability in childrenâ□□s causal inferences: The sampling hypothesis. *Cognition*, 126(2), 285–300.

[67] Desikan, R. S., Ségonne, F., Fischl, B., Quinn, B. T., Dickerson, B. C., Blacker, D., Buckner, R. L., Dale, A. M., Maguire, R. P., Hyman, B. T., Albert, M. S., & Killiany, R. J. (2006). An automated labeling system for subdividing the human cerebral cortex on MRI scans into gyral based regions of interest. *NeuroImage*, 31(3), 968–980.

[68] Desrochers, T., ichi Amemori, K., & Graybiel, A. (2015). Habit learning by naive macaques is marked by response sharpening of striatal neurons representing the cost and outcome of acquired action sequences. *Neuron*, 87(4), 853 – 868.

[69] Dezfouli, A. & Balleine, B. W. (2012). Habits, action sequences and reinforcement learning. *European Journal of Neuroscience*, 35(7), 1036–1051.

[70] Dezza, I. C., Angela, J. Y., Cleeremans, A., & Alexander, W. (2017). Learning the value of information and reward over time when solving exploration-exploitation problems. *Scientific reports*, 7(1), 16919.

[71] Dickinson, A., Nicholas, D., & Adams, C. D. (1983). The effect of the instrumental training contingency on susceptibility to reinforcer devaluation. *The Quarterly Journal of Experimental Psychology*, 35(1), 35–51.

[72] Diedrichsen, J., Ridgway, G. R., Friston, K. J., & Wiestler, T. (2011). Comparing the similarity and spatial structure of neural representations: a pattern-component model. *NeuroImage*, 55, 1665–1678.

[73] Dietterich, T. G. (2000). Hierarchical reinforcement learning with the maxq value function decomposition. *J. Artif. Int. Res.*, 13(1), 227–303.

[74] Digney, B. (1996). Emergent hierarchical control structures: Learning reactive / hierarchical relationships in reinforcement environments. In *Proceedings of the Fourth Conference on the Simulation of Adaptive Behavior: SAB 96*.

[75] Dobrunz, L. E., Huang, E. P., & Stevens, C. F. (1997). Very short-term plasticity in hippocampal synapses. *Proceedings of the National Academy of Sciences*, 94(26), 14843–14847.

[76] Dolan, R. J. & Dayan, P. (2013). Goals and habits in the brain. *Neuron*, 80(2), 312–325.

[77] Donnarumma, F., Maisto, D., & Pezzulo, G. (2016). Problem solving as probabilistic inference with subgoaling: explaining human successes and pitfalls in the tower of hanoi. *PLoS computational biology*, 12(4), e1004864.

[78] Donoso, M., Collins, A. G. E., & Koechlin, E. (2014). Foundations of human reasoning in the prefrontal cortex. *Science*, 344(6191), 1481–1486.

[79] Doshi-Velez, F. (2009). The infinite partially observable markov decision process. In Y. Bengio, D. Schuurmans, J. D. Lafferty, C. K. I. Williams, & A. Culotta (Eds.), *Advances in Neural Information Processing Systems 22* (pp. 477–485). Curran Associates, Inc.

[80] Doya, K., Samejima, K., Katagiri, K.-i., & Kawato, M. (2002). Multiple model-based reinforcement learning. *Neural computation*, 14(6), 1347–1369.

[81] Dragoi, G. & Tonegawa, S. (2011). Preplay of future place cell sequences by hippocampal cellular assemblies. *Nature*, 469(7330), 397.

[82] Duan, Y., Schulman, J., Chen, X., Bartlett, P. L., Sutskever, I., & Abbeel, P. (2016). Rl ˆ2: Fast reinforcement learning via slow reinforcement learning. *arXiv preprint arXiv:1611.02779*.

[83] Dunsmoor, J. E., Niv, Y., Daw, N., & Phelps, E. A. (2015). Rethinking extinction. *Neuron*, 88, 47–63.

[84] Eickhoff, S. B., Stephan, K. E., Mohlberg, H., Grefkes, C., Fink, G. R., Amunts, K., & Zilles, K. (2005). A new SPM toolbox for combining probabilistic cytoarchitectonic maps and functional imaging data. *NeuroImage*, 25(4), 1325–1335.

[85] Erdem, U. M. & Hasselmo, M. (2012). A goal-directed spatial navigation model using forward trajectory planning based on grid cells. *European Journal of Neuroscience*, 35(6), 916–931.

[86] Erev, I. & Barron, G. (2005). On adaptation, maximization, and reinforcement learning among cognitive strategies. *Psychological review*, 112(4), 912.

[87] Eshel, N., Bukwich, M., Rao, V., Hemmelder, V., Tian, J., & Uchida, N. (2015). Arithmetic and local circuitry underlying dopamine prediction errors. *Nature*, 525(7568), 243.

[88] Eshel, N., Tian, J., Bukwich, M., & Uchida, N. (2016). Dopamine neurons share common response function for reward prediction error. *Nature neuroscience*, 19(3), 479.

[89] Fecteau, S., Knoch, D., Fregni, F., Sultani, N., Boggio, P., & Pascual-Leone, A. (2007). Diminishing risk-taking behavior by modulating activity in the prefrontal cortex: a direct current stimulation study. *Journal of Neuroscience*, 27(46), 12500–12505.

[90] Feinberg, D. A., Moeller, S., Smith, S. M., Auerbach, E., Ramanna, S., Gunther, M., Glasser, M. F., Miller, K. L., Uğurbil, K., & Yacoub, E. (2010). Multiplexed echo planar imaging for sub-second whole brain fMRI and fast diffusion imaging. *PLoS ONE*, 5(12), e15710.

[91] Fernández, J. A. & González, J. (2013). *Multi-hierarchical representation of large-scale space: Applications to mobile robots*, volume 24. Springer Science & Business Media.

[92] Flagel, S. B., Clark, J. J., Robinson, T. E., Mayo, L., Czuj, A., Willuhn, I., Akers, C. A., Clinton, S. M., Phillips, P. E., & Akil, H. (2011). A selective role for dopamine in stimulus–reward learning. *Nature*, 469(7328), 53.

[93] Frank, M. J. (2008). Schizophrenia: a computational reinforcement learning perspective. *Schizophrenia bulletin*, 34(6), 1008–1011.

[94] Frank, M. J. & Badre, D. (2011). Mechanisms of hierarchical reinforcement learning in corticostriatal circuits 1: computational analysis. *Cerebral cortex*, 22(3), 509–526.

[95] Frank, M. J. & Badre, D. (2012). Mechanisms of hierarchical reinforcement learning in corticostriatal circuits 1: Computational analysis. *Cereb Cortex*, 22(3), 509–526.

[96] Frank, M. J., Doll, B. B., Oas-Terpstra, J., & Moreno, F. (2009). Prefrontal and striatal dopaminergic genes predict individual differences in exploration and exploitation. *Nature neuroscience*, 12(8), 1062.

[97] Fujii, N. & Graybiel, A. M. (2003). Representation of action sequence boundaries by macaque prefrontal cortical neurons. *Science*, 301(5637), 1246–1249.

[98] Fujimoto, H., Hasegawa, T., & Watanabe, D. (2011). Neural coding of syntactic structure in learned vocalizations in the songbird. *Journal of Neuroscience*, 31(27), 10023–10033.

[99] Geddes, C. E., Li, H., & Jin, X. (2018). Optogenetic editing reveals the hierarchical organization of learned action sequences. *Cell*, 174(1), 32–43.

[100] Genesereth, M., Love, N., & Pell, B. (2005). General game playing: Overview of the aaai competition. *AI magazine*, 26(2), 62–62.

[101] Gershman, S. & Blei, D. (2012a). A tutorial on Bayesian nonparametric models. *J Math Psychol*, 56, 1–12.

[102] Gershman, S. J. (2015a). A unifying probabilistic view of associative learning. *PLoS computational biology*, 11(11), e1004567.

[103] Gershman, S. J. (2015b). A unifying probabilistic view of associative learning. *PLoS Comput Biol*, 11(11), e1004567.

[104] Gershman, S. J. (2017). Context-dependent learning and causal structure. *Psychon Bull Rev*, 24, 557–565.

[105] Gershman, S. J. (2018). Deconstructing the human algorithms for exploration. *Cognition*, 173, 34–42.

[106] Gershman, S. J. (2019). Uncertainty and exploration. *Decision*, (pp. 265504).

[107] Gershman, S. J. & Blei, D. M. (2012b). A tutorial on bayesian nonparametric models. *Journal of Mathematical Psychology*, 56(1), 1–12.

[108] Gershman, S. J., Blei, D. M., & Niv, Y. (2010). Context, learning, and extinction. *Psychological review*, 117(1), 197.

[109] Gershman, S. J., Norman, K. A., & Niv, Y. (2015a). Discovering latent causes in reinforcement learning. *Curr Opin Behav Sci*, 5, 43–50. Neuroeconomics.

[110] Gershman, S. J., Norman, K. A., & Niv, Y. (2015b). Discovering latent causes in reinforcement learning. *Current Opinion in Behavioral Sciences*, 5, 43–50.

[111] Gershman, S. J., Pesaran, B., & Daw, N. D. (2009a). Human reinforcement learning subdivides structured action spaces by learning effector-specific values. *Journal of Neuroscience*, 29(43), 13524–13531.

[112] Gershman, S. J., Pesaran, B., & Daw, N. D. (2009b). Human reinforcement learning subdivides structured action spaces by learning effector-specific values. *J Neurosci*, 29(43), 13524–13531.

[113] Gershman, S. J. & Tzovaras, B. G. (2018). Dopaminergic genes are associated with both directed and random exploration. *Neuropsychologia*, 120, 97–104.

[114] Gershman, S. J., Vul, E., & Tenenbaum, J. B. (2012). Multistability and perceptual inference. *Neural computation*, 24(1), 1–24.

[115] Girgin, S., Polat, F., & Alhajj, R. (2006). Learning by automatic option discovery from conditionally terminating sequences. In *ECAI 2006, 17th European Conference on Artificial Intelligence*, volume 141 (pp. 494–498).

[116] Gläscher, J. (2009). Visualization of group inference data in functional neuroimaging. *Neuroinformatics*, 7(1), 73–82.

[117] Gläscher, J., Daw, N., Dayan, P., & O'Doherty, J. P. (2010). States versus rewards: Dissociable neural prediction error signals underlying model-based and model-free reinforcement learning. *Neuron*, 66(4), 585–595.

[118] Gluth, S., Rieskamp, J., & Büchel, C. (2012). Deciding when to decide: time-variant sequential sampling models explain the emergence of value-based decisions in the human brain. *Journal of Neuroscience*, 32(31), 10686–10698.

[119] Gönner, L., Vitay, J., & Hamker, F. H. (2017). Predictive place-cell sequences for goal-finding emerge from goal memory and the cognitive map: A computational model. *Frontiers in computational neuroscience*, 11, 84.

[120] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

[121] Goodman, N. D., Tenenbaum, J. B., Feldman, J., & Griffiths, T. L. (2008). A rational analysis of rule-based concept learning. *Cognitive science*, 32(1), 108–154.

[122] Gopnik, A., Glymour, C., Sobel, D. M., Schulz, L. E., Kushnir, T., & Danks, D. (2004). A theory of causal learning in children: causal maps and bayes nets. *Psychological review*, 111(1), 3.

[123] Grabenhorst, F. & Rolls, E. T. (2011). Value, pleasure and choice in the ventral prefrontal cortex. *Trends in cognitive sciences*, 15(2), 56–67.

[124] Grahame, N. J., Hallam, S. C., Geier, L., & Miller, R. R. (1990). Context as an occasion setter following either CS acquisition and extinction or CS acquisition alone. *Learning and Motivation*, 21, 237–265.

[125] Gratton, G., Coles, M. G., Sirevaag, E. J., Eriksen, C. W., & Donchin, E. (1988). Pre-and poststimulus activation of response channels: a psychophysiological analysis. *Journal of Experimental Psychology: Human perception and performance*, 14(3), 331.

[126] Grau, J. W. & Rescorla, R. A. (1984). Role of context in autoshaping. *J Exp Psychol: Anim Behav Process*, 10, 324–332.

[127] Graybiel, A. M. (1998). The basal ganglia and chunking of action repertoires. *Neurobiology of learning and memory*, 70(1-2), 119–136.

[128] Graziano, M., Polosecki, P., Shalom, D. E., & Sigman, M. (2011). Parsing a perceptual decision into a sequence of moments of thought. *Frontiers in integrative neuroscience*, 5, 45.

[129] Greve, D. N. (2002). Optseq2 home page. Available online at http://surfer.nmr.mgh.harvard.edu/optseq (Accessed July 3, 2017).

[130] Griffiths, T. L. & Tenenbaum, J. B. (2005). Structure and strength in causal induction. *Cogn Psychol*, 51(4), 334–384.

[131] Haefner, R. M., Berkes, P., & Fiser, J. (2016). Perceptual decision-making as probabilistic inference by neural sampling. *Neuron*, 90, 649–660.

[132] Hare, T. A., Schultz, W., Camerer, C. F., O'Doherty, J. P., & Rangel, A. (2011). Transformation of stimulus value signals into motor commands during simple choice. *Proceedings of the National Academy of Sciences*, 108(44), 18120–18125.

[133] Hart, A. S., Rutledge, R. B., Glimcher, P. W., & Phillips, P. E. (2014). Phasic dopamine release in the rat nucleus accumbens symmetrically encodes a reward prediction error term. *Journal of Neuroscience*, 34(3), 698–704.

[134] Hengst, B. (2002). Discovering hierarchy in reinforcement learning with HEXQ. In *Proceedings of the Nineteenth International Conference on Machine Learning*, ICML '02 (pp. 243–250). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

[135] Herrojo Ruiz, M., Rusconi, M., Brücke, C., Haynes, J.-D., Schönecker, T., & Kühn, A. A. (2014). Encoding of sequence boundaries in the subthalamic nucleus of patients with parkinson's disease. *Brain*, 137(10), 2715–2730.

[136] Hertwig, R., Barron, G., Weber, E. U., & Erev, I. (2004). Decisions from experience and the effect of rare events in risky choice. *Psychological science*, 15(8), 534–539.

[137] Holroyd, C. B. & McClure, S. M. (2015). Hierarchical control over effortful behavior by rodent medial frontal cortex: A computational model. *Psychological Review*, 122(1), 54.

[138] Huettel, S. A., Song, A. W., & McCarthy, G. (2005). Decisions under uncertainty: probabilistic context influences activation of prefrontal and parietal cortices. *Journal of Neuroscience*, 25(13), 3304–3311.

[139] Hull, C. L. (1943). *Principles of behavior*, volume 422. Appleton-century-crofts New York.

[140] Huys, Q. J., Eshel, N., O'Nions, E., Sheridan, L., Dayan, P., & Roiser, J. P. (2012). Bonsai trees in your head: how the pavlovian system sculpts goal-directed choices by pruning decision trees. *PLoS computational biology*, 8(3), e1002410.

[141] Huys, Q. J., Lally, N., Faulkner, P., Eshel, N., Seifritz, E., Gershman, S. J., Dayan, P., & Roiser, J. P. (2015). Interplay of approximate planning strategies. *Proceedings of the National Academy of Sciences*, (pp. 201414219).

[142] Jin, X. & Costa, R. M. (2010). Start/stop signals emerge in nigrostriatal circuits during sequence learning. *Nature*, 466(7305), 457–462.

[143] Jin, X., Tecuapetla, F., & Costa, R. M. (2014). Basal ganglia subcircuits distinctively encode the parsing and concatenation of action sequences. *Nature Neuroscience*, 17, 423.

[144] Jung, M. W., Wiener, S. I., & McNaughton, B. L. (1994). Comparison of spatial firing characteristics of units in dorsal and ventral hippocampus of the rat. *Journal of Neuroscience*, 14(12), 7347–7356.

[145] Kaelbling, L. P., Littman, M. L., & Cassandra, A. R. (1998). Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1-2), 99–134.

[146] Kahneman, D. & Egan, P. (2011). *Thinking, fast and slow*, volume 1. Farrar, Straus and Giroux New York.

[147] Kahneman, D. & Tversky, A. (1979). Prospect theory: An analysis of decision under risk. *Econometrica*, 47(2), 263–292.

[148] Kakade, S. & Dayan, P. (2002a). Acquisition and extinction in autoshaping. *Psychological review*, 109(3), 533.

[149] Kakade, S. & Dayan, P. (2002b). Acquisition and extinction in autoshaping. *Psychol Rev*, 109, 533–44.

[150] Kaye, H., Preston, G. C., Szabo, L., Druiff, H., & Mackintosh, N. J. (1987). Context specificity of conditioning and latent inhibition: Evidence for a dissociation of latent inhibition and associative interference. *Q J Exp Psychol B*, 39(2), 127–145.

[151] Kjelstrup, K. B., Solstad, T., Brun, V. H., Hafting, T., Leutgeb, S., Witter, M. P., Moser, E. I., & Moser, M.-B. (2008). Finite scale of spatial representation in the hippocampus. *Science*, 321(5885), 140–143.

[152] Knoch, D., Gianotti, L. R., Pascual-Leone, A., Treyer, V., Regard, M., Hohmann, M., & Brugger, P. (2006). Disruption of right prefrontal cortex by low-frequency repetitive transcranial magnetic stimulation induces risk-taking behavior. *Journal of Neuroscience*, 26(24), 6469–6472.

[153] Koch, I. & Hoffmann, J. (2000). Patterns, chunks, and hierarchies in serial reaction-time tasks. *Psychological research*, 63(1), 22–35.

[154] Koechlin, E., Ody, C., & Kouneiher, F. (2003). The architecture of cognitive control in the human prefrontal cortex. *Science*, 302(5648), 1181–1185.

[155] Koechlin, E. & Summerfield, C. (2007). An information theoretical approach to prefrontal executive function. *Trends Cogn Sci*, 11(6), 229–235.

[156] Kondor, R. I. & Lafferty, J. (2002). Diffusion kernels on graphs and other discrete structures. In *Proceedings of the 19th international conference on machine learning*, volume 2002 (pp. 315–322).

[157] Konidaris, G. (2016). Constructing abstraction hierarchies using a skill-symbol loop. In *IJCAI: proceedings of the conference*, volume 2016 (pp. 1648).: NIH Public Access.

[158] Kool, W., Gershman, S. J., & Cushman, F. A. (2017). Cost-benefit arbitration between multiple reinforcement-learning systems. *Psychological science*, 28(9), 1321–1333.

[159] Körding, K. P., Beierholm, U., Ma, W. J., Quartz, S., Tenenbaum, J. B., & Shams, L. (2007). Causal inference in multisensory perception. *PLoS One*, 2, e943.

[160] Kriegeskorte, N., Goebel, R., & Bandettini, P. (2006). Information-based functional brain mapping. *Proc Natl Acad Sci*, 103(10), 3863–3868.

[161] Kriegeskorte, N., Mur, M., & Bandettini, P. (2008a). Representational similarity analysis – connecting the branches of systems neuroscience. *Front Syst Neurosci*, 2, 4–10.

[162] Kriegeskorte, N., Mur, M., & Bandettini, P. A. (2008b). Representational similarity analysis-connecting the branches of systems neuroscience. *Frontiers in systems neuroscience*, 2, 4.

[163] Kriegeskorte, N., Simmons, W. K., Bellgowan, P. S., & Baker, C. I. (2009). Circular analysis in systems neuroscience: the dangers of double dipping. *Nature neuroscience*, 12(5), 535.

[164] Kruschke, J. K. (2008a). Bayesian approaches to associative learning: From passive to active learning. *Learning & behavior*, 36(3), 210–226.

[165] Kruschke, J. K. (2008b). Bayesian approaches to associative learning: From passive to active learning. *Learning & Behavior*, 36(3), 210–226.

[166] Kulkarni, T. D., Narasimhan, K., Saeedi, A., & Tenenbaum, J. (2016a). Hierarchical deep reinforcement learning: Integrating temporal abstraction and intrinsic motivation. In *Advances in neural information processing systems* (pp. 3675–3683).

[167] Kulkarni, T. D., Narasimhan, K. R., Saeedi, A., & Tenenbaum, J. B. (2016b). Hierarchical deep reinforcement learning: Integrating temporal abstraction and intrinsic motivation. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16 (pp. 3682–3690). USA: Curran Associates Inc.

[168] Kumar, A., Wu, Z., Pitkow, X., & Schrater, P. (2019). Belief dynamics extraction. *arXiv preprint arXiv:1902.00673*.

[169] Laird, J. E. (2012). *The Soar cognitive architecture*. MIT press.

[170] Laird, J. E., Rosenbloom, P. S., & Newell, A. (1986). Chunking in soar: The anatomy of a general learning mechanism. *Machine learning*, 1(1), 11–46.

[171] Lake, B. M., Salakhutdinov, R., & Tenenbaum, J. B. (2015). Human-level concept learning through probabilistic program induction. *Science*, 350(6266), 1332–1338.

[172] Lake, B. M., Ullman, T. D., Tenenbaum, J. B., & Gershman, S. J. (2017). Building machines that learn and think like people. *Behavioral and brain sciences*, 40.

[173] Lashley, K. S. (1951). *The problem of serial order in behavior*, volume 21. Bobbs-Merrill.

[174] Legenstein, R. & Maass, W. (2014). Ensembles of spiking neurons with noise support optimal probabilistic inference in a dynamically changing environment. *PLoS Comput Biol*, 10, e1003859.

[175] Levy, R. P., Reali, F., & Griffiths, T. L. (2009). Modeling the effects of memory on human online sentence processing with particle filters. In *Advances in neural information processing systems* (pp. 937–944).

[176] Liberman, A. M., Cooper, F. S., Shankweiler, D. P., & Studdert-Kennedy, M. (1967). Perception of the speech code. *Psychological review*, 74(6), 431.

[177] Lim, S.-L., O'Doherty, J. P., & Rangel, A. (2011). The decision value computations in the vmpfc and striatum use a relative value code that is guided by visual attention. *Journal of Neuroscience*, 31(37), 13214–13223.

[178] Liu, Y., Dolan, R. J., Kurth-Nelson, Z., & Behrens, T. E. (2019). Human replay spontaneously reorganizes experience. *Cell*, 178(3), 640–652.

[179] Love, N., Hinrichs, T., Haley, D., Schkufza, E., & Genesereth, M. (2008). General game playing: Game description language specification.

[180] Lovibond, P. F., Preston, G., & Mackintosh, N. (1984). Context specificity of conditioning, extinction, and latent inhibition. *J Exp Psychol: Anim Behav Process*, 10, 360–375.

[181] Lynn, C. W., Kahn, A. E., & Bassett, D. S. (2018). Structure from noise: Mental errors yield abstract representations of events. *arXiv preprint arXiv:1805.12491*.

[182] Machado, M. C., Bellemare, M. G., & Bowling, M. H. (2017). A Laplacian framework for option discovery in reinforcement learning. *Computing Research Repository*, abs/1703.00956.

[183] Mackintosh, N. J. (1975). A theory of attention: variations in the associability of stimuli with reinforcement. *Psychol Rev*, 82, 276–298.

[184] Madarasz, T. J., Diaz-Mataix, L., Akhand, O., Ycu, E. A., LeDoux, J. E., & Johansen, J. P. (2016). Evaluation of ambiguous associations in the amygdala by learning the structure of the environment. *Nat Neurosci*, 19, 965–972.

[185] Maggiore, G., Spanò, A., Orsini, R., Bugliesi, M., Abbadi, M., & Steffinlongo, E. (2012). A formal specification for casanova, a language for computer games. In *EICS* (pp. 287–292).

[186] Maisto, D., Donnarumma, F., & Pezzulo, G. (2016). Nonparametric problem-space clustering: learning efficient codes for cognitive control tasks. *Entropy*, 18(2), 61.

[187] Mannor, S., Menache, I., Hoze, A., & Klein, U. (2004). Dynamic abstraction in reinforcement learning via clustering. In *Proceedings of the Twenty-first International Conference on Machine Learning*, ICML '04 (pp. 71–). New York, NY, USA: ACM.

[188] Marr, D. & Poggio, T. (1976). *From Understanding Computation to Understanding Neural Circuitry*. Technical report, Cambridge, MA, USA.

[189] May, B. C., Korda, N., Lee, A., & Leslie, D. S. (2012). Optimistic Bayesian sampling in contextual-bandit problems. *Journal of Machine Learning Research*, 13(Jun), 2069–2106.

[190] McGovern, A. (2002). Autonomous discovery of abstractions through interaction with an environment. In S. Koenig & R. C. Holte (Eds.), *Abstraction, Reformulation, and Approximation* (pp. 338–339). Berlin, Heidelberg: Springer Berlin Heidelberg.

[191] McGovern, A. & Barto, A. G. (2001). Automatic discovery of subgoals in reinforcement learning using diverse density. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01 (pp. 361–368). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

[192] McGugan, W. (2007). *Beginning game development with Python and Pygame: from novice to professional*. Apress.

[193] McNamee, D., Wolpert, D. M., & Lengyel, M. (2016). Efficient state-space modularization for planning: theory, behavioral and neural signatures. In *Advances in Neural Information Processing Systems* (pp. 4511–4519).

[194] Meder, B., Mayrhofer, R., & Waldmann, M. R. (2014). Structure induction in diagnostic causal reasoning. *Psychol Rev*, 121, 277–301.

[195] Mehlhorn, K., Newell, B. R., Todd, P. M., Lee, M. D., Morgan, K., Braithwaite, V. A., Hausmann, D., Fiedler, K., & Gonzalez, C. (2015). Unpacking the exploration–exploitation tradeoff: A synthesis of human and animal literatures. *Decision*, 2(3), 191.

[196] Menache, I., Mannor, S., & Shimkin, N. (2002). Q-cut—dynamic discovery of sub-goals in reinforcement learning. In T. Elomaa, H. Mannila, & H. Toivonen (Eds.), *Machine Learning: ECML 2002* (pp. 295–306). Berlin, Heidelberg: Springer Berlin Heidelberg.

[197] Miller, G., Galanter, E., & Pribram, K. (1960). Plans and the structure of behavior.

[198] Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological review*, 63(2), 81.

[199] Miller, R. R., Barnet, R. C., & Grahame, N. J. (1995). Assessment of the Rescorla-Wagner model. *Psychological Bulletin*, 117, 363–386.

[200] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. (2015). Human-level control through deep reinforcement learning. *Nature*, 518(7540), 529.

[201] Moeller, S., Yacoub, E., Olman, C. A., Auerbach, E., Strupp, J., Harel, N., & Uğurbil, K. (2010). Multiband multislice GE-EPI at 7 tesla, with 16-fold acceleration using partial parallel imaging with application to high spatial and temporal whole-brain fMRI. *Magn Reson Med*, 63(5), 1144–1153.

[202] Momennejad, I., Russek, E. M., Cheong, J. H., Botvinick, M. M., Daw, N. D., & Gershman, S. J. (2017). The successor representation in human reinforcement learning. *Nature Human Behaviour*, 1(9), 680.

[203] Moore, A. W. (1991). Variable resolution dynamic programming: Efficiently learning action maps in multivariate real-valued state-spaces. In *Machine Learning Proceedings 1991* (pp. 333–337). Elsevier.

[204] Mumford, J., Poline, J.-B., & Poldrack, R. (2015). Orthogonalization of regressors in fMRI models. *PLoS ONE*, 10, e0126255.

[205] Murphy, K. P. (2001). Active learning of causal bayes net structure.

[206] Myers, J. L. & Sadler, E. (1960). Effects of range of payoffs as a variable in risk taking. *Journal of Experimental Psychology*, 60(5), 306.

[207] Nair, C., Prabhakar, B., & Shah, D. (2006). On entropy for mixtures of discrete and continuous variables. *arXiv preprint cs/0607075*.

[208] Neal, R. M. (2000). Markov chain sampling methods for dirichlet process mixture models. *Journal of computational and graphical statistics*, 9(2), 249–265.

[209] Newell, A. (1992). Unified theories of cognition and the role of soar. In *SOAR: A cognitive architecture in perspective* (pp. 25–79). Springer.

[210] Newell, A., Shaw, J. C., & Simon, H. A. (1958). Elements of a theory of human problem solving. *Psychological review*, 65(3), 151.

[211] Newell, A., Simon, H. A., et al. (1972). *Human problem solving*, volume 104. Prentice-Hall Englewood Cliffs, NJ.

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

[213] Niv, Y., Daniel, R., Geana, A., Gershman, S. J., Leong, Y. C., Radulescu, A., & Wilson, R. C. (2015). Reinforcement learning in multidimensional environments relies on attention mechanisms. *J Neurosci*, 35, 8145–8157.

[214] Odling-Smee, F. (1978). The overshadowing of background stimuli: some effects of varying amounts of training and UCS intensity. *Q J Exp Psychol*, 30, 737–746.

[215] O'Keefe, J. (1976). Place units in the hippocampus of the freely moving rat. *Experimental neurology*, 51(1), 78–109.

[216] O'Keefe, J. & Nadel, L. (1978). *The hippocampus as a cognitive map*. Oxford: Clarendon Press.

[217] Orbán, G., Berkes, P., Fiser, J., & Lengyel, M. (2016). Neural variability and sampling-based probabilistic representations in the visual cortex. *Neuron*, 92, 530–543.

[218] O'Reilly, J. X., Jbabdi, S., Rushworth, M. F. S., & Behrens, T. E. J. (2013). Brain systems for probabilistic and dynamic prediction: Computational specificity and integration. *PLoS Biol*, 11(9), e1001662.

[219] Pan, W.-X., Schmidt, R., Wickens, J. R., & Hyland, B. I. (2005). Dopamine cells respond to predicted events during classical conditioning: evidence for eligibility traces in the reward-learning network. *Journal of Neuroscience*, 25(26), 6235–6242.

[220] Pavlov, I. (1927). Conditioned reflexes.,(oxford university press: London).

[221] Payzan-LeNestour, E. & Bossaerts, P. (2011). Risk, unexpected uncertainty, and estimation uncertainty: Bayesian learning in unstable settings. *PLOS Computational Biology*, 7(1), 1–14.

[222] Pearce, J. M. & Bouton, M. E. (2001). Theories of associative learning in animals. *Annu Rev Psychol*, 52, 111–139.

[223] Pearce, J. M. & Hall, G. (1980). A model for pavlovian learning: variations in the effectiveness of conditioned but not of unconditioned stimuli. *Psychological review*, 87(6), 532.

[224] Peirce, J. W. (2007a). PsychoPy - Psychophysics software in Python. *J Neurosci Methods*, 162(1), 8–13.

[225] Peirce, J. W. (2007b). PsychoPy - Psychophysics software in Python. *J Neurosci Methods*, 162(1), 8–13.

[226] Pereira, F. & Botvinick, M. (2011). Information mapping with pattern classifiers: A comparative study. *NeuroImage*, 56(2), 476–496. Multivariate Decoding and Brain Reading.

[227] Polanía, R., Krajbich, I., Grueschow, M., & Ruff, C. C. (2014). Neural oscillations and synchronization differentially support evidence accumulation in perceptual and value-based decision making. *Neuron*, 82(3), 709–720.

[228] Pouget, A., Beck, J. M., Ma, W. J., & Latham, P. E. (2013a). Probabilistic brains: knowns and unknowns. *Nat Neurosci*, 16(9), 1170–1178. Review.

[229] Pouget, A., Beck, J. M., Ma, W. J., & Latham, P. E. (2013b). Probabilistic brains: knowns and unknowns. *Nature neuroscience*, 16(9), 1170.

[230] Preston, G., Dickinson, A., & Mackintosh, N. (1986). Contextual conditional discriminations. *Q J Exp Psychol*, 38, 217–237.

[231] Rao, R. P. (2010). Decision making under uncertainty: a neural model based on partially observable markov decision processes. *Frontiers in computational neuroscience*, 4, 146.

[232] Rasmussen, D., Voelker, A., & Eliasmith, C. (2017). A neural model of hierarchical reinforcement learning. *PloS one*, 12(7), e0180234.

[233] Ravindran, B. & Barto, A. G. (2002). Model minimization in hierarchical reinforcement learning. In S. Koenig & R. C. Holte (Eds.), *Abstraction, Reformulation, and Approximation* (pp. 196–211). Berlin, Heidelberg: Springer Berlin Heidelberg.

[234] Rescorla, R. A. & Wagner, A. R. (1972). A theory of Pavlovian conditioning: Variations on the effectiveness of reinforcement and non-reinforcement. In A. H. Black & W. F. Prokasy (Eds.), *Classical conditioning II: Current research and theory* (pp. 64–99). New York: Appleton-Century-Crofts.

[235] Rescorla, R. A., Wagner, A. R., et al. (1972). A theory of pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement. *Classical conditioning II: Current research and theory*, 2, 64–99.

[236] Ribas-Fernandes, J. J., Solway, A., Diuk, C., McGuire, J. T., Barto, A. G., Niv, Y., & Botvinick, M. M. (2011). A neural signature of hierarchical reinforcement learning. *Neuron*, 71(2), 370–379.

[237] Rigoux, L., Stephan, K., Friston, K., & Daunizeau, J. (2014a). Bayesian model selection for group studies - revisited. *NeuroImage*, 84, 971–985.

[238] Rigoux, L., Stephan, K. E., Friston, K. J., & Daunizeau, J. (2014b). Bayesian model selection for group studiesâ□□revisited. *Neuroimage*, 84, 971–985.

[239] Roberts, G. O. & Rosenthal, J. S. (2009). Examples of adaptive mcmc. *Journal of Computational and Graphical Statistics*, 18(2), 349–367.

[240] Rolls, E. T., Joliot, M., & Tzourio-Mazoyer, N. (2015). Implementation of a new parcellation of the orbitofrontal cortex in the automated anatomical labeling atlas. *NeuroImage*, 122, 1 – 5.

[241] Rosenbaum, D. A., Inhoff, A. W., & Gordon, A. M. (1984). Choosing between movement sequences: A hierarchical editor model. *Journal of Experimental Psychology: General*, 113(3), 372.

[242] Rosenbaum, D. A., Kenny, S. B., & Derr, M. A. (1983). Hierarchical control of rapid movement sequences. *Journal of Experimental Psychology: Human Perception and Performance*, 9(1), 86.

[243] Russell, S. J. & Norvig, P. (2016). *Artificial intelligence: a modern approach*. Malaysia; Pearson Education Limited,.

[244] Sakai, K., Kitaguchi, K., & Hikosaka, O. (2003). Chunking during human visuomotor sequence learning. *Experimental brain research*, 152(2), 229–242.

[245] Sanborn, A. N. & Chater, N. (2016). Bayesian brains without probabilities. *Trends in cognitive sciences*, 20(12), 883–893.

[246] Schapiro, A. C., Rogers, T. T., Cordova, N. I., Turk-Browne, N. B., & Botvinick, M. M. (2013a). Neural representations of events arise from temporal community structure. *Nature neuroscience*, 16(4), 486.

[247] Schapiro, A. C., Rogers, T. T., Cordova, N. I., Turk-Browne, N. B., & Botvinick, M. M. (2013b). Neural representations of events arise from temporal community structure. *Nat Neurosci*, 16(4), 486–492.

[248] Schaul, T. (2013). A video game description language for model-based or interactive learning. In *2013 IEEE Conference on Computational Inteligence in Games (CIG)* (pp. 1–8).: IEEE.

[249] Schmajuk, N. A. & Larrauri, J. A. (2006). Experimental challenges to theories of classical conditioning: application of an attentional model of storage and retrieval. *Journal of Experimental Psychology: Animal Behavior Processes*, 32(1), 1.

[250] Schuck, N. W., Cai, M. B., Wilson, R. C., & Niv, Y. (2016). Human orbitofrontal cortex represents a cognitive map of state space. *Neuron*, 91, 1402–1412.

[251] Schultz, W., Dayan, P., & Montague, P. R. (1997). A neural substrate of prediction and reward. *Science*, 275(5306), 1593–1599.

[252] Schulz, E., Franklin, N. T., & Gershman, S. J. (2018). Finding structure in multi-armed bandits. *bioRxiv*, (pp. 432534).

[253] Schulz, E. & Gershman, S. J. (2019). The algorithmic architecture of exploration in the human brain. *Current opinion in neurobiology*, 55, 7–14.

[254] Schulz, E., Konstantinidis, E., & Speekenbrink, M. (2015). Learning and decisions in contextual multi-armed bandit tasks. In *CogSci*.

[255] Seghier, M. L. (2013). The angular gyrus. *The Neuroscientist*, 19(1), 43–61.

[256] Shiner, T., Seymour, B., Wunderlich, K., Hill, C., Bhatia, K. P., Dayan, P., & Dolan, R. J. (2012). Dopamine and performance in a reinforcement learning task: evidence from parkinson's disease. *Brain*, 135(6), 1871–1883.

[257] Shohamy, D. & Daw, N. D. (2015). Integrating memories to guide decisions. *Current Opinion in Behavioral Sciences*, 5, 85–90.

[258] Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., et al. (2017). Mastering the game of go without human knowledge. *Nature*, 550(7676), 354.

[259] Simon, H. A. (1978). Information-processing theory of human problem solving. *Handbook of learning and cognitive processes*, 5, 271–295.

[260] Singh, S. P., Jaakkola, T., & Jordan, M. I. (1995). Reinforcement learning with soft state aggregation. In *Advances in neural information processing systems* (pp. 361–368).

[261] Skinner, B. (1938). The behavior of organisms: an experimental analysis.

[262] Smith, K. & Graybiel, A. (2013). A dual operator view of habitual behavior reflecting cortical and striatal dynamics. *Neuron*, 79(2), 361–374.

[263] Solway, A., Diuk, C., Córdova, N., Yee, D., Barto, A. G., Niv, Y., & Botvinick, M. M. (2014a). Optimal behavioral hierarchy. *PLoS computational biology*, 10(8), e1003779.

[264] Solway, A., Diuk, C., Córdova, N., Yee, D., Barto, A. G., Niv, Y., & Botvinick, M. M. (2014b). Optimal behavioral hierarchy. *PLOS Computational Biology*, 10(8), 1–10.

[265] Somerville, L. H., Sasse, S. F., Garrad, M. C., Drysdale, A. T., Abi Akar, N., Insel, C., & Wilson, R. C. (2017). Charting the expansion of strategic exploratory behavior during adolescence. *Journal of Experimental Psychology: General*, 146, 155–164.

[266] Soto, F. A., Gershman, S. J., & Niv, Y. (2014). Explaining compound generalization in associative and causal learning through rational principles of dimensional generalization. *Psychol Rev*, 121, 526–558.

[267] Speekenbrink, M. & Konstantinidis, E. (2015). Uncertainty and exploration in a restless bandit problem. *Topics in cognitive science*, 7(2), 351–367.

[268] Srinivas, N., Krause, A., Kakade, S., & Seeger, M. (2010). Gaussian process optimization in the bandit setting: No regret and experimental design. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10 (pp. 1015–1022). USA: Omnipress.

[269] Srivastava, V., Reverdy, P., & Leonard, N. E. (2015). Correlated multiarmed bandit problem: Bayesian algorithms and regret analysis. *arXiv preprint arXiv:1507.01160*.

[270] Stachenfeld, K. L., Botvinick, M. M., & Gershman, S. J. (2017). The hippocampus as a predictive map. *Nature neuroscience*, 20(11), 1643.

[271] Stanovich, K. E. & West, R. F. (2000). Individual differences in reasoning: Implications for the rationality debate? *Behavioral and brain sciences*, 23(5), 645–665.

[272] Starkweather, C. K., Babayan, B. M., Uchida, N., & Gershman, S. J. (2017). Dopamine reward prediction errors reflect hidden-state inference across time. *Nature Neuroscience*, 20(4), 581–589.

[273] Starkweather, C. K., Gershman, S. J., & Uchida, N. (2018). Medial prefrontal cortex shapes dopamine reward prediction errors under state uncertainty. *Submitted for publication*.

[274] Steinberg, E. E., Keiflin, R., Boivin, J. R., Witten, I. B., Deisseroth, K., & Janak, P. H. (2013). A causal link between prediction errors, dopamine neurons and learning. *Nature neuroscience*, 16(7), 966.

[275] Steyvers, M., Tenenbaum, J. B., Wagenmakers, E.-J., & Blum, B. (2003). Inferring causal networks from observations and interventions. *Cognitive science*, 27(3), 453–489.

[276] Stolle, M. & Precup, D. (2002). Learning options in reinforcement learning. In S. Koenig & R. C. Holte (Eds.), *Abstraction, Reformulation, and Approximation* (pp. 212–223). Berlin, Heidelberg: Springer Berlin Heidelberg.

[277] Stuber, G. D., Klanker, M., de Ridder, B., Bowers, M. S., Joosten, R. N., Feenstra, M. G., & Bonci, A. (2008). Reward-predictive cues enhance excitatory synaptic strength onto midbrain dopamine neurons. *Science*, 321(5896), 1690–1692.

[278] Sutton, R. S. (1991). Dyna, an integrated architecture for learning, planning, and reacting. *ACM SIGART Bulletin*, 2(4), 160–163.

[279] Sutton, R. S. & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

[280] Sutton, R. S., Precup, D., & Singh, S. (1999). Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112(1), 181 – 211.

[281] Swartzentruber, D. (1995). Modulatory mechanisms in Pavlovian conditioning. *Anim Learn Behav*, 23, 123–143.

[282] Swartzentruber, D. & Bouton, M. E. (1986). Analysis of the associative and occasion setting properties of contexts participating in a Pavlovian discrimination. *J Exp Psychol: Anim Behav Process*, 12, 333–350.

[283] Takahashi, Y., Schoenbaum, G., & Niv, Y. (2008). Silencing the critics: understanding the effects of cocaine sensitization on dorsolateral and ventral striatum in the context of an actor/critic model. *Frontiers in neuroscience*, 2, 14.

[284] Tervo, D. G. R., Tenenbaum, J. B., & Gershman, S. J. (2016). Toward the neural implementation of structure learning. *Curr Opin Neurobiol*, 37, 99–105. Neurobiology of cognitive behavior.

[285] Tesauro, G. (1994). Td-gammon, a self-teaching backgammon program, achieves master-level play. *Neural computation*, 6(2), 215–219.

[286] Thaker, P., Tenenbaum, J. B., & Gershman, S. J. (2017). Online learning of symbolic concepts. *Journal of Mathematical Psychology*, 77, 10–20.

[287] Thielscher, M. (2010). A general game description language for incomplete information games. In *Twenty-Fourth AAAI Conference on Artificial Intelligence*.

[288] Thompson, W. R. (1933). On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4), 285–294.

[289] Thorndike, E. L. (1911). *Animal intelligence; experimental studies*. New York, The Macmillan Company.

[290] Tobler, P. N., O'Doherty, J. P., Dolan, R. J., & Schultz, W. (2007). Reward value coding distinct from risk attitude-related uncertainty coding in human reward systems. *Journal of neurophysiology*, 97(2), 1621–1632.

[291] Tolman, E. C. (1948). Cognitive maps in rats and men. *Psychological Review*, 55(4), 189–208.

[292] Tomov, M. S., Dorfman, H. M., & Gershman, S. J. (2018). Neural computations underlying causal structure learning. *Journal of Neuroscience*, 38(32), 7143–7157.

[293] Tong, S. & Koller, D. (2001). Active learning for structure in bayesian networks. In *International joint conference on artificial intelligence*, volume 17 (pp. 863–869).: Citeseer.

[294] Tricomi, E., Balleine, B. W., & O'Doherty, J. P. (2009). A specific role for posterior dorsolateral striatum in human habit learning. *European Journal of Neuroscience*, 29(11), 2225–2232.

[295] Tzourio-Mazoyer, N., Landeau, B., Papathanassiou, D., Crivello, F., Etard, O., Delcroix, N., Mazoyer, B., & Joliot, M. (2002). Automated anatomical labeling of activations in SPM using a macroscopic anatomical parcellation of the MNI MRI single-subject brain. *NeuroImage*, 15(1), 273–289.

[296] Ullman, T. D., Goodman, N. D., & Tenenbaum, J. B. (2012). Theory learning as stochastic search in the language of thought. *Cognitive Development*, 27(4), 455–480.

[297] van der Kouwe, A. J., Benner, T., Salat, D. H., & Fischl, B. (2008). Brain morphometry with multiecho MPRAGE. *NeuroImage*, 40(2), 559–569.

[298] Van Hamme, L. J. & Wasserman, E. A. (1994). Cue competition in causality judgments: The role of nonpresentation of compound stimulus elements. *Learning and motivation*, 25(2), 127–151.

[299] Vezhnevets, A., Mnih, V., Agapiou, J., Osindero, S., Graves, A., Vinyals, O., & Kavukcuoglu, K. (2016). Strategic attentive writer for learning macro-actions. *Computing Research Repository*, abs/1606.04695.

[300] Vezhnevets, A. S., Osindero, S., Schaul, T., Heess, N., Jaderberg, M., Silver, D., & Kavukcuoglu, K. (2017). Feudal networks for hierarchical reinforcement learning. *Computing Research Repository*, abs/1703.01161.

[301] Vincent, J. L., Kahn, I., Snyder, A. Z., Raichle, M. E., & Buckner, R. L. (2008). Evidence for a frontoparietal control system revealed by intrinsic functional connectivity. *Journal of Neurophysiology*, 100(6), 3328–3342.

[302] Vul, E., Goodman, N., Griffiths, T. L., & Tenenbaum, J. B. (2014). One and done? optimal decisions from very few samples. *Cognitive science*, 38(4), 599–637.

[303] Wallis, J. D. (2012). Cross-species studies of orbitofrontal cortex and value-based decision-making. *Nature neuroscience*, 15(1), 13.

[304] Wang, J. X., Kurth-Nelson, Z., Kumaran, D., Tirumala, D., Soyer, H., Leibo, J. Z., Hassabis, D., & Botvinick, M. (2018). Prefrontal cortex as a meta-reinforcement learning system. *Nature neuroscience*, 21(6), 860.

[305] Wang, J. X., Kurth-Nelson, Z., Tirumala, D., Soyer, H., Leibo, J. Z., Munos, R., Blundell, C., Kumaran, D., & Botvinick, M. (2016). Learning to reinforcement learn. *CoRR*, abs/1611.05763.

[306] Weber, E. U., Shafir, S., & Blais, A.-R. (2004). Predicting risk sensitivity in humans and lower animals: risk as variance or coefficient of variation. *Psychological review*, 111(2), 430.

[307] Wiener, J. M. & Mallot, H. A. (2003). 'fine-to-coarse'route planning and navigation in regionalized environments. *Spatial cognition and computation*, 3(4), 331–358.

[308] Wilkinson, G. & Rogers, C. (1973). Symbolic description of factorial models for analysis of variance. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 22(3), 392–399.

[309] Wilson, R. & Collins, A. (2019). Ten simple rules for the computational modeling of behavioral data.

[310] Wilson, R., Takahashi, Y., Schoenbaum, G., & Niv, Y. (2013). Orbitofrontal cortex as a cognitive map of task space. *Neuron*, 81(2), 267–279.

[311] Wilson, R. C., Geana, A., White, J. M., Ludvig, E. A., & Cohen, J. D. (2014). Humans use directed and random exploration to solve the explore–exploit dilemma. *Journal of Experimental Psychology: General*, 143(6), 2074.

[312] Wingate, D., Diuk, C., O'Donnell, T., Tenenbaum, J., & Gershman, S. (2013). Compositional policy priors.

[313] Wu, C. M., Schulz, E., Speekenbrink, M., Nelson, J. D., & Meder, B. (2018a). Generalization guides human exploration in vast decision spaces. *Nature Human Behaviour*, 2(12), 915.

[314] Wu, C. M., Schulz, E., Speekenbrink, M., Nelson, J. D., & Meder, B. (2018b). Generalization guides human exploration in vast decision spaces. *Nature Human Behaviour*, 2(12), 915–924.

[315] Wu, Z., Schrater, P., & Pitkow, X. (2018c). Inverse pomdp: Inferring what you think from what you do. *arXiv preprint arXiv:1805.09864*.

[316] Xu, J., Moeller, S., Auerbach, E. J., Strupp, J., Smith, S. M., Feinberg, D. A., Yacoub, E., & Uğurbil, K. (2013). Evaluation of slice accelerations using multiband echo planar imaging at 3 T. *NeuroImage*, 83, 991–1001.

[317] Yechiam, E. & Busemeyer, J. R. (2005). Comparison of basic assumptions embedded in learning models for experience-based decision making. *Psychonomic bulletin & review*, 12(3), 387–402.

[318] Yin, H. H., Knowlton, B. J., & Balleine, B. W. (2004). Lesions of dorsolateral striatum preserve outcome expectancy but disrupt habit formation in instrumental learning. *European journal of neuroscience*, 19(1), 181–189.

[319] Yin, H. H., Ostlund, S. B., Knowlton, B. J., & Balleine, B. W. (2005). The role of the dorsomedial striatum in instrumental conditioning. *European Journal of Neuroscience*, 22(2), 513–523.

[320] Zacks, J. M. & Swallow, K. M. (2007). Event segmentation. *Current directions in psychological science*, 16(2), 80–84.

[321] Zajkowski, W. K., Kossut, M., & Wilson, R. C. (2017). A causal role for right frontopolar cortex in directed, but not random, exploration. *eLife*, 6, e27430.

[322] Ziegler, S., Pedersen, M. L., Mowinckel, A. M., & Biele, G. (2016). Modelling adhd: A review of adhd theories through their predictions for computational models of decision-making and reinforcement learning. *Neuroscience & Biobehavioral Reviews*, 71, 633–656.