Compositionality in Human Structure Learning

by
Nicholas T. Franklin
B. S., The University of Texas at Austin, 2009
B. A., The University of Texas at Austin, 2009

A dissertation submitted in partial fulfillment of the
requirements for the Degree of Doctor of Philosophy
in the Department of Cognitive, Linguistic & Psychological Sciences at Brown University

Providence, Rhode Island
May 2018

This dissertation by Nicholas T. Franklin is accepted in its present form by the Department of Cognitive, Linguistic & Psychological Sciences as satisfying the dissertation requirement
for the degree of Doctor of Philosophy.

Date _____ _____
Dr. Micheal Frank, Director

Recommended to the Graduate Council

Date _____ _____
Dr. David Badre, Reader

Date _____ _____
Dr. Michael Littman, Reader
(Computer Science)

Approved by the Graduate Council

Date _____ _____
Dr. Andrew G. Campbell
Dean of the Graduate School

Curriculum Vitae

Nicholas T. Franklin

nicholas_franklin@brown.edu
nicholastfranklin.com
512-415-0837

Education

B.S.	The University of Texas at Austin, Neurobiology	2009
B.A.	The University of Texas at Austin, Spanish	2009

Research Experience

PhD Student, Brown University *Advisor: Michael J. Frank*	2011 -
Research Assistant, Weill Cornell Medical College, *Advisor: B.J. Casey*	2009 - 2011
Undergraduate Research Assistant, UT Austin, *Advisor: Alison Preston*	2008 - 2009

Published Articles

Franklin NT and Frank MJ (2015). A cholinergic feedback circuit to regulate striatal population uncertainty and optimize reinforcement learning. *eLife*

Teslovich T, Mulder M, **Franklin NT**, Ruberry EJ, Millner A, Somerville LH, Simen P, Durston S, Casey BJ (2014). Adolescents let sufficient evidence accumulate before making a decision when large incentives are at stake. *Developmental Science*

Casey BJ, Somerville LH, Gotlib IH, Ayduk O, Franklin NT, Askren MK, Jonides J, Berman MG, Wilson NL, Teslovich T, Glover G, Zayas V, Mischel W, & Shoda Y (2011) Behavioral and neural correlates of delay of gratification 40 years later. *PNAS*

Conference Proceedings

Talks

Franklin NT and Frank MJ (October, 2015). Independent clustering and generalization of action-outcome and outcome-values in goal-directed learning. *45th Annual Meeting of the Society for Neuroscience*

Posters

Franklin NT and Frank MJ (June, 2017) "Compositional Task Clusters in Human Transfer Learning", *The Multi-disciplinary Conference on Reinforcement Learning and Decision Making*

Franklin NT and Frank MJ (January, 2017) A Cholinergic Feedback Mechanism to Modulate Dopaminergic Learning within the Striatum in Response to Striatal Population Uncertainty *50th Meeting of the Winter Conference on Brain Research*

Scott D, **Franklin NT** and Frank MJ (January, 2017) Disentangling Impulsivity Mechanisms in the Imagen Dataset Using Machine Learning *50th Meeting of the Winter Conference on Brain Research*

Franklin NT and Frank MJ (May, 2016) Independent generalization of action-effects and outcome-values in multistep and goal-directed learning *38e Symposium International du GRSNC, The Neuroscience of Decision Making*

Franklin NT and Frank MJ (April, 2016) Generalization in goal-directed learning: benefits of independent clustering of world-model and goals *23rd Annual Meeting of the Cognitive Neuroscience Society*

Franklin NT and Frank MJ (February, 2016) Generalization in goal-directed learning: independent clustering of action- effect and outcome-values *Computational and Systems Neuroscience (Cosyne)*

Franklin NT and Frank MJ (February, 2014). A Bayesian perspective on flexibly responding to stochastic and non-stationary task: a role for striatal acetylcholine. *Computational and Systems Neuroscience (Cosyne)*

Franklin NT and Frank MJ (November, 2013) Contributions of tonically active neurons and uncertainty to striatal learning. *43rd Annual Meeting of the Society for Neuroscience*

Franklin NT and Frank MJ (February 2013). Uncertainty and the striatum: How tonically active neurons may aid learning in dynamic environments. *Computational and Systems Neuroscience (Cosyne)*

Casey BJ, **Franklin NT**, Somerville LH, Gotlib IH, Ayduk O, Askren M.K., Jonides J, Berman MG, Wilson NL, Teslovcih T, Glover G, Mischel W, Shoda T (2011). Behavioral and Neural Correlates of Delay of Gratification 40 years later. *41st Annual Meeting of the Society for Neuroscience*

Franklin NT and Dominick A. (2009). The Role of Attention in Reward Motivated Learning. *2009 University of Texas at Austin College of Natural Sciences Undergraduate Forum*

Awards

Kenneth R. and Pamela L. Galner Fund Dissertation Fellowship	2016
Excellence in Human Development, Family, & Social Science Research, UT Austin	2009
Undergraduate Research Fellowship, University of Texas at Austin	2009
Phi Beta Kappa	2008

Technical Skills

Artificial Intelligence: reinforcement learning, Bayesian cognitive modeling, attractor neural networks

Statistical Analysis: statistical modeling, Bayesian analysis, non-parametric analysis, time-series analysis

Machine Learning: classification, regression, clustering

Programming: Python (numpy, scipy, scikit-learn, cython, pymc3), Matlab, Javascript, HTML

Ad-Hoc Reviewing

Behavioral and Brain Sciences, Biological Psychiatry, Cognition, Cognition & Emotion, Connection Science, Journal of Neuroscience, Nature, Nature Neuroscience, Neuropsychopharmacology, PLOS Computational Biology, Psych Review

Teaching Experience

Computational Cognitive Neuroscience *(Graduate Teaching Assistant)*	2013-14, 16
Introduction to Cognitive Neuroscience *(Graduate Teaching Assistant)*	2014
Computational Cognitive Science *(Graduate Teaching Assistant)*	2013

Making Decisions *(Graduate Teaching Assistant)* 2012

University Service

Coordinated department Cognition Seminar Series 2014-2015

Professional Memberships

Society for Neuroscience, Cognitive Neuroscience Society

Acknowledgements

I am truly grateful to have been able to work these past six years in a such a fantastic environment with so many interesting and wonderful people. I have had the privilege to follow my interests and grow a scientist in the process. First of all, I would like to thank my advisor and mentor Michael Frank for all of the patient support, even as I took on projects that were well above my skill level when I started. I came to Brown with a better idea of the type of science I wanted to pursue than the specific questions I wanted to work on. The work presented here owes much to his insight, support and willingness to take scientific risks and it has been an absolute pleasure to work with him.

I would also like to thank my thesis committee members, David Badre and Michael Littman for their helpful and insightful feedback. Additionally, I would like to thank Shelia Blumstein for her encouragement at the start of my grad school career when I was excited about all of the different potential avenues of research. I am grateful of the generosity of the financial support provided the Kenneth and Pamela Galner Fund, provided by Mrs. Judy Banker and her family. I am also thankful to the members of the Frank lab and the community at Brown, who in addition to being a generous and valuable source of feedback and conversation, have become a community of friends.

I want to thank my family and my friends for their continuing support, especially my friends Danne and Jerry with whom I lived for one very fun year. Lastly, I am indebted to my wife, Erin, for navigating living in 4 different states and all of the other the twists and turns with me.

Abstract of "Compositionality in Human Structure Learning" by Nicholas T. Franklin, Ph.D., Brown University, May 2018.

Humans are remarkably adept at generalizing knowledge between experiences in a way that can be difficult for computers. Often, this entails generalizing constituent pieces of experiences that do not fully overlap, but nonetheless share useful similarities with previously acquired knowledge. A young musician may learn to play several instruments in different contexts, for example, learning the flute to play classical and the saxophone to play jazz. Because the fingerings of the saxophone and the flute are nearly the same, a musician can re-use previously learned hand motions for different effect across the two instruments. Conversely, a musician may learn to play a single song across many instruments that require completely distinct physical motions, but yet still transfer knowledge between them. This degree of compositionality is critical for flexible goal-directed behavior but can be difficult for computational frameworks because they often assume an underlying structure of the world that is incompatible with generalization. Here, I propose a novel computational framework that leverages the compositional structure in real-world environments by assuming people generalize goals, or what to do, independently from a word model, or how to do it. I examine the computational framework normatively and ask when it makes sense to generalize goals and world models independently or together. In a series of experiments, I compare human subject behavior to model predictions, showing that people adapt their generalization strategy depending on the environment. Together, these results suggest that no one strategy is best across all environments, and that while it is adaptive to represent a complex environment in learnable components, people pay attention to the relationship between components in their environment.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

A central question in cognitive science is how do people think and act in their environment, making decisions intelligently? While this broad question has inspired entire fields of research and generations of scientists, philosophers and artists, we are far from understanding all of the creativity and nuance with which people interact in their world. Nonetheless, asking this broad question is useful as a loadstone that frames our research as we parse or field into ever smaller, but tractable, problems.

One possible parsing is learning; we can substitute the question of human intelligence for an easier question of how do people learn about regularities in their environment and act accordingly? This problem, while complex, is well motivated theoretically. In order to act in a purposeful way, a person needs knowledge of the structure of their environment, the effect their actions have in that environment and an internal objective. From a theoretical perspective, we can cast two of these problems, learning the structure of an environment and learning action effects, as statistical inference. At its most broad, we can interpret learning to take an action given some internal objective as a classification problem: raw perceptual input is classified to output some subset of possible actions. This insight provides the theoretical foundation needed to formally link human learning to that of artificial agents.

A promising line of research in this regard has linked theoretical accounts of learning, often originating in computer science, to human and animal behavior and neurobiological functions (Dayan and Balleine, 2002). Empirically, there is strong support that simple types of feedback based learning are instantiated by the basal ganglia and

connected systems via a mechanism substantially similar to reinforcement-learning (RL) learning algorithms. RL agents learn through maximizing the reward they receive through the interaction with their environment (Sutton and Barto, 1998). A simple RL agent will update the value of the actions with feedback, such that actions that lead to higher reward will become more likely and actions that lead to lower reward will become less likely. Both humans and animals respond to feed back in a similar way, suggesting they learn using a similar strategy.

A key finding from this line of research has been the implementation of a simple RL algorithm within the basal ganglia. Mid-brain dopamine provides the basal ganglia with a feedback signal in the form of a reward-prediction error (Bayer et al., 2007; Schultz, 1998; Zaghloul et al., 2009). This feedback signal aids learning by biasing action selection towards actions that have been previously associated with reward. In the basal ganglia, a similar form of learning is thought to occur through plasticity in striatal medium spiny neurons (MSNs). There are two classes of MSNs within the striatum exerting opposing influences on action selection. Activity in the D1 dopamine receptor expressing MSNs (D1-MSNs) leads to the selection of an action while activity in D2-MSNs leads to the suppression of action selection (Frank et al., 2004; Kravitz et al., 2010). Dopamine affects plasticity but has separate effects on excitability of the two classes of MSNs as D1 activity is excitatory and D2 activity is inhibitory (Surmeier et al., 2011). As a result, a phasic dopamine increases in response to an unexpectedly rewarding event increase the propensity of an actions that led to that event to be taken in the future. Likewise, an outcome with an unexpectedly low reward will result in a decrease of phasic dopamine and lower the likelihood of selecting the actions that led to that outcome. In this way, the basal ganglia is able to function in the same way as a simple reinforcement learner.

Despite compelling evidence to support reinforcement-learning within the basal ganglia, simple RL algorithms are limited theoretically in the types of problems they can solve. While modern Q-learning systems can learn to play Atari games at a human level of performance (Mnih et al., 2015), they require several orders of magnitude more training time than a human learner requires (Lake et al., 2016). In addition, the policies these systems learn are often brittle to small perturbations in the input space. Kansky et al. (2017) trained a Q-learning system on "Breakout," an Atari

game in which the player controls a paddle at the bottom of the screen that can be used to hit a ball into blocks at the top of the screen to win points, and found that displacing the paddle a few pixels higher on the screen was sufficient to disrupt the learned policy. Humans, in contrast, do not show these types of sensitivities to small perturbation. Furthermore, people are capable of transferring policies learned in one game to another and can reuse to world knowledge to solve a similar problem in a way that can be difficult for artificial agents (Lake et al., 2016).

This insight is not new; the limitations of associative learning systems and the challenges they pose as a model of human intelligence have been debated for decades (Newell and Simon, 1959; Fodor and Pylyshyn, 1988). Nonetheless, there is sufficient empirical support for the involvement in a prediction error based reinforcement learner in higher cognitive function to motivate to how such a system could underlie higher cognitive functions (Badre et al., 2010; Daw et al., 2011; Collins and Frank, 2013, 2016b; Donoso et al., 2014). Two central problems are the lack of generalization and the treatment of non-stationary environments. In the RL models of human and animal learning, it is typically assumed that subjects learn action-values conditional on the current state, which are then used to guide action selection (Daw and Doya, 2006; Sutton and Barto, 1998). A state is some representation of the current environment that is sufficient to determine a policy. In a navigation task, this might be the agent's location as well as the location of all other relevant objects. In a bandit task used to study human learning, the state might represent a trial type in a given task. Lack of generalization arises because states, as a formalism, are Markovian. Thus action-values learned in one state aren't transferable to another state regardless of how similar the states may be.

In ecological environments, the size of the state space is unbounded and large scale problems require substantial experience for simple RL methods to solve (Botvinick et al., 2009). This need to learn a large number of independent state-action values partially drives the long convergence time of the reinforcement learning systems trained on Atari games (Lake et al., 2016; Kansky et al., 2017). It is unclear, given the potential size of an ecological state-space, whether a human learner could ever have sufficient experience to learn enough stimulus-action associations to perform even trial tasks. A far more likely hypothesis is that people pool experiences into

an abstraction on which learning can occur (Wilson et al., 2014b), in much the same way that using a convolutional neural nets as the basis of a reinforcement learner has been fruitful (Mnih et al., 2015). Abstracting experiences into a state may reduce the dimensionality of the problem substantially and thus, learning time, but it leaves open the problems of generalization and non-stationarity.

To illustrate the problem of generalization, consider the following example. If a person goes into the kitchen, regardless of whether they are making lunch or doing the dishes, the set of perceptual stimuli will be similar. Given the same set of stimuli, multiple possible tasks are possible each requiring a separate policy. Likewise, if we go to a friend's house and they ask us to help with the dishes, we won't need to relearn how to do the dishes even if we've never seen the specific dishes or kitchen. We should be able to reuse the previously learned dish washing behavior in this new context, even though the specific percept is different. This problem is complex and poses two separate challenges: separating similar perceptual experiences into multiple policies and reusing the same policy in perceptually different contexts.

A highly similar problem to generalization is non-stationarity. Learning volatile, or non-stationarity, environmental contingencies poses a problem that often be simple for humans to solve but difficult for computational agents as a consequence of the Markov assumption. Non-stationarity, reflecting changes in the observation distribution across time, is common in both experimental and ecological settings. Broadly, these tasks reflect either random, unpredictable changes to the outcome statistics that have to be incorporated into updating the previous model, or they can reflect a change in an (often unobservable) statistical context. In the former case, adaptive strategies allow agents to dynamically estimate stimulus-action-outcome associations by inferring the presence of change points probabilistically and weighing observations following a change point more strongly while discounting prior ones (Behrens et al., 2007; Wilson et al., 2010; Nassar et al., 2010). More simple, heuristic strategies that modulate their learning rate (the degree to which unexpected outcomes serve to update stimulus-action-outcome associations) as a function of a measure of uncertainty are also adaptive (Yu and Dayan, 2002, 2005; Wilson and Finkel, 2009; Payzan-LeNestour and Bossaerts, 2011; Franklin and Frank, 2015). However, because these models assume that previously acquired knowledge is no longer relevant, they

suffer in environments in which this knowledge is again relevant in new contexts.

A promising expansion of RL models to account for both generalization and non-stationarity it to frame fronto-striatal learning in terms of a partially observable Markov decision process (POMDPs) (Todd et al., 2008; Collins and Frank, 2013). A POMDP represents a structure in which determining the current state is accomplished through inference. Each observation is assumed to be generated from an observation distribution and the current state can be inferred by inverting the generative process via Bayesian inference. This poses a potential solution to both generalization non-stationarity, if two distinct percepts have similar task statistics, they can potentially be inferred as belonging to the same latent state, and thus share the same action-values. Similarly, if two similar percepts have different task statistics, they may be inferred to belonging to different states and different action-values.

Nonetheless, POMDPs present a considerable computational challenge. Learning to act optimally in a fully observable Markov Decision Process has a well-defined solution that is computationally cheap and can be computed online (Littman, 1996). Solutions to POMDPs are considerably more complex to the point that learning to act optimally is computationally intractable in the worst case (Kaelbling et al., 1998). In practice, it is useful to estimate the current state and learn the reward contingencies conditional on this estimate. In a Bayesian framework, this estimate is a belief distribution over the state space. An online approximation algorithm, such as a particle filter, can be used to update the belief state as the learner moves through the task. However, the process of updating the belief distribution can be highly computationally demanding. When the state-space is large, it can lead to a high degree of model uncertainty when the agent does not know the structure well and substantial waste of computational resources on unlikely solutions.

One solution is to assume a generative process that enforces parsimony (Chrisman, 1992; Doshi-Velez, 2009; Gershman et al., 2010). In this framework, the number of states in the state-space grows as the evidence requires but is biased towards a lower number of states. Having a generative process over the number of states allows the learner to learn the structure of the task in addition to its solution. Thus, the learning problem faced is the inference of the correct latent state, followed by learning a policy dependent on this state. The extant empirical evidence tends to support

this hypothesis. Recent studies within the human and animal literature suggest that inferring a structure occurs whether or not it is valuable to do so (Acuña and Schrater, 2010; Badre et al., 2010; Gershman et al., 2010; Collins and Frank, 2013, 2016b). This appears to involve the inference of a latent state, potentially represented in the Orbital Frontal Cortex (Wilson et al., 2014b), that influences both dopaminergic (Starkweather et al., 2017) and cholinergic (Stalnaker et al., 2016) signaling.

Mathematically, models that explain this type of structure learning are clustering algorithms that partition experiences into clusters that share a policy. These models are highly similar to models used to explain category learning (Gershman and Niv, 2013; Sanborn et al., 2010; Shafto et al., 2011) and feature learning in humans (Austerweil et al., 2009). In the model proposed by Collins and Frank (2013), a "context" (typically, an observable feature of the data) signals a "Task Set" or rule structure that is operative. Each stimulus within a given Task Set has the same action-values. An alternate interpretation of this model is that each context cues the same policy. In this view, policies are reused based on their appropriateness given the data and a defined prior, an approach that is common in machine learning (Mahmud et al., 2013; Rosman et al., 2016; Wilson et al., 2012). These models assume that two or more contexts (or tasks, environments, etc.) can share a policy if the policy leads the same objective function value to an acceptable degree of error (Mahmud et al., 2013).

While this line of research is promising, it is limited in its capacity as a model of human intelligence. A key feature of cognition is thought to be productivity, or the ability to devise a novel solution to a novel problem using component pieces (James, 1890; Chomsky, 1968; Fodor and Pylyshyn, 1988; Sloman, 1996). A model that learns a policy in context and re-uses that same policy without changes in a new context is not productive; in James' terminology it is reproductive. As a practical matter, this distinction is important as an agent that has simply transferred a policy from one context to another may by highly sensitive to small changes in environment (Lake et al., 2016; Kansky et al., 2017). This leaves open a substantial theoretical challenge: can reinforcement learning, as an associative model of human learning, produce productive behavior? Or is a separate mechanism needed to explain the complex patterns of behavior assumed to be associated with productive thought?

There are two classes of hypotheses in which a reinforcement learning system would be able to account for complex behavior. One, it may be the case that a cognitive system that appears productive is not actually productive (Rumelhart and McClelland, 1986; Johnson, 2004). A second, intriguing possibility is that the fronto-striatal learning system is capable of learning the structured representations necessary for productivity. This raises the question of what would we need to call an RL agent productive? At minimum, we would expect the agent is able to learn component pieces of a task and recombine them in a novel situation. In other words, we would expect a productive reinforcement learning agent to be, at minimum, compositional in its representations.

The central question asked within this work is this precursor question: how can a reinforcement learning agent generalize compositionally? The balance of this chapter will present the mathematical background for the Markov Decision Processes and clustering algorithms used as the basis of the models presented in later chapters. Chapter 2 presents a theoretical model of generalization accompanied with a normative analysis of alternative models across different task domains. Chapter 3 provides a series of behavior experiments to test the predictions of the models presented in chapter 2 and provides evidence that human subjects show compositionality in their generalization.

1.1 Markov Decision Problems

Markov decision problems (MDPs) from the basis of RL agents by providing a definition of the statistical problem the agent has to solve. It is informative to discuss the assumptions underlying the MDP and how those assumptions shape the algorithms used to solve them. An RL agent typically is tasked with solving a completely observable MDP. An MDP describes a process of making sequential decisions and differs from a POMDP in that the agent always has full knowledge of its position or condition in the environment. This knowledge is abstracted into a state, which serves as a summary of all of the potentially relevant information. Many experimental tasks, such as classical conditioning and bandit tasks, can be framed as MDPs, making RL well suited to serve as a model.

Formally, an MDP can be defined by the tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma, \rangle$ (Kaelbling et al., 1996), where

- $\mathcal{S}$ is a set of states s
- $\mathcal{A}$ is a set of actions a available to the agent
- $\mathcal{T} : \mathcal{S} \times \mathcal{A} \rightarrow \Pi(\mathcal{S})$ is a mapping between states and actions that returns a probability distribution over successor states.
- $\mathcal{R} : \mathcal{S} \times \mathcal{A} \times \mathcal{S}' \rightarrow \mathbb{R}$ is a mapping between states, actions and successor states that returns a real valued reward
- γ is a discount rate

The sets $\mathcal{S}$ and $\mathcal{A}$ can be defined as continuous or discrete. We can define $\mathcal{S}$ most broadly as a set of vectors in $\mathbb{R}^d$, but for the purpose of psychological tasks, it is convenient to define $\mathcal{S}$ as small set of integers, with each integer representing some stimulus class (for example, a blue triangle on a screen). Similarly, in most psychological paradigms, there is a small number of discrete actions that make up $\mathcal{A}$ (for example, a button press). The mapping $\mathcal{T}$ defines a transition function, typically represented via a conditional probability distribution $T(s, a, s') = \Pr(s'|a, s)$ that define the probability of moving from state s to state s' given action a is taken. $\mathcal{R}$ is the reward functions $R(s, a, s') = \int_{-\infty}^{\infty} r \Pr(r|s, a, s')\mathrm{d}r$ (often simplified to $R(s, a)$) that define the rewards received by taking action a in state s and moving to the successor state s'. It is important to note that the MDP tuple defines the environment, not the behavior of the agent in the environment. The agent's behavior is defined by a policy $\pi : \mathcal{S} \rightarrow \mathcal{A}$, a function that maps states to actions. Finally, the discount rate γ defines the degree to which an agent values reward in the future less than current rewards. γ is a multiplicative factor with the constrain that $\gamma = [0, 1]$ and while is traditionally defined as a property of the MDP, it is often more natural to think of it as a property of the agent.

An MDP gets its name because it observes the Markov property, meaning that the process is memoryless and both state-transitions and rewards only depend on the current state. Consequently, the transition functions T and reward functions R are independent for each state. If an agent uses some stationary policy π, the expected

value of an action is dependent only on the current state. As a result, we can compute a value function for a given policy, defined by the system of equations known as the Bellman equation (Bellman, 1957):

$$V^{\pi}(s) = R\left(s, \pi(s)\right) + \gamma \sum_{s' \in \mathcal{S}} T\left(s, \pi(s), s'\right) V^{\pi}(s') \quad \forall\ s \in \mathcal{S} \tag{1.1}$$

We can define the goal of an agent in an MDP as maximizing its total discounted reward. Therefore, an optimal policy π^* is any policy that maximizes the expected discounted reward in all states.

$$\pi^*(s) = \text{argmax}_a \left[R\left(s, a\right) + \gamma \sum_{s' \in \mathcal{S}} T\left(s, a, s'\right) V^*(s') \right] \tag{1.2}$$

Because the MDP observes the Markov property, knowing the value of the states is sufficient to determine an optimal policy. If all of the values are known, the optimal policy is to move to the state with the highest expected discounted reward. As a result, there are two general approaches that can be used estimate to state values (Sutton and Barto, 1998). One way would be to estimate the transition and reward functions and use those to determine an optimal policy. Dynamic programming can then be used to estimate the value function. This approach can be referred to as *model-based* RL because its utilizes a model of the MDP to generate a policy.

A second approach would be to estimate the values of each state-action pair directly without learning the full MDP. This is typically referred to as *model-free* RL. In behavioral neuroscience, this distinction is important as the two serve as models for distinct types of behavior (Dayan and Niv, 2008). Model-free RL associates a state-action pair with future rewards and is used as a model of habitual learning. In contrast, because model-based RL determines a behavioral policy from knowledge of the structure of the environment, it is used as a model of goal-directed behavior (Balleine et al., 2008; Daw, 2012). A key distinction between the two types of learning is planning, learning the MDP allows the agent to plan multiple moves in advance while learning state-action values is myopic. Both model-based and model-free RL are thought to be involved in basal-ganglia dependent learning (Balleine et al., 2008; Daw et al., 2011). Model-free RL models of the basal ganglia can be quite detailed, drawing on a wealth of biological data (Frank et al., 2004).

1.1.1 Model-Free RL

Perhaps the simplest form of model-free RL is Q-learning. In Q-learning the agent explores an MDP learning state-action values through feedback. As the agent explores the environment, the observed outcomes are used to update estimated state-action, or "Q", values for each experienced state-action pair. These Q-values are subsequently used in an action selection function to determine the next action. As the agent does not know the veridical expected value of each action and must learn them through exploration, learning and exploration will interact resulting in a trade-off between the exploring the environment and maximizing current reward (Gittins, 1979; Wilson et al., 2014a). As an estimate of the state-action value function, Q-values can be defined recursively for a policy using the Bellman equation.

$$Q^{\pi}(s,a) = R(s,a) + \gamma \sum_{s' \in \mathcal{S}} T(s,a,s')\, Q(s', \pi(s')) \tag{1.3}$$

If the agent learns the optimal policy, then $Q(s, \pi(s))$ will converge to $V(s)$. The optimal Q function is

$$Q^{*}(s,a) = R(s,a) + \gamma \sum_{s' \in \mathcal{S}} T(s,a,s') \max_{a'} Q^{*}(s', a') \tag{1.4}$$

While Q-values are a function of function of T and R, they can be defined as an expectation (specifically, the expected value of an action given a state) and can be estimated independently online. If we initialize Q-values with a prior value, we can update them with the difference between the current Q-value and the observed reward following an action using the update equation

$$Q(s,a) \leftarrow Q(s,a) + \alpha \left(r_t + \gamma \max_{a'} Q(s', a') - Q(s,a) \right) \tag{1.5}$$

where s' is the observed next state and α is the rate at which Q-values are updated, often referred to as the learning rate. If $\alpha = 1/t$, where t is the number of times the state-action pair has been visited, then the equation 1.5 is equivalent to an online maximum likelihood estimator of the expectation of total discounted reward for taking action a in state s. As such the Q-learner's state-action value function will converge to the optimal action-values with probability 1 if all states-action pairs are sampled sufficiently (Watkins and Dayan, 1992).

It is often useful to think of the difference between the predicted and observed values as a separate quantity. This value, known as a temporal difference (TD) error or prediction error is often denoted with δ and can be defined

$$\delta = \left(r_t + \gamma \max_{a'} Q\left(s', a'\right)) - Q(s, a)\right) \tag{1.6}$$

$$Q\left(s, a\right) \leftarrow Q\left(s, a\right) + \alpha\delta \tag{1.7}$$

Updating Q-values with the prediction error propagates received rewards later in time to earlier states that are predictive of future reward. More broadly, Q-learning is a temporal difference algorithm because it uses a δ term to update its values iteratively. In other temporal difference algorithms δ may be defined differently but the basic framework is the same.

This simplified form of reinforcement learning is useful in that there is substantial neurobiological evidence for a dopamine reward-prediction error term similar to δ. Specifically, the magnitude of phasic dopamine bursts in response to unexpectedly positive outcomes closely correlates with δ when δ is positive while the duration of the phasic dopamine pauses in response to unexpectedly negative outcomes correlates with δ when δ is negative (Montague et al., 1996; Schultz, 1998; Bayer et al., 2007). Likewise, human fMRI studies have demonstrated that BOLD signals in the ventral striatum, an area innervated by mid-brain dopamine, correlate with fitted estimates of δ (Pagnoni et al., 2002; McClure et al., 2003a; Abler et al., 2006). Because dopamine is sufficient to drive learning within the basal ganglia (Tsai et al., 2009) and neuronal activity in its targets can control motion (Kravitz et al., 2010), there is strong reason to believe that dopamine acts as a training signal much in the way a prediction error signal in a model-free RL learning would. While the role of mid-brain dopamine in the striatum is likely more complex (Schultz, 2010; Hamid et al., 2016) and there is evidence to suggest model-free RL is not sufficient to explain all extant behavioral data (Balleine et al., 2008), model-free RL models of the basal ganglia have generated numerous empirically substantiated predictions (Maia and Frank, 2011).

It is worth noting that model-free learning systems are quite powerful, if given sufficient training data. Deep reinforcement learning systems have shown human level or superhuman level performance on a range of Atari task (Mnih et al., 2013; Schölkopf, 2015). These models work by training a convolutional neural net to process the raw

pixel input of Atari games while simultaneously training a Q-learner on the output of the neural net. In an analogous way to the visual system, the convolutional neural net reduce the raw visual input into a much more compact, and linearly discriminable, representation (Yamins et al., 2013; DiCarlo and Cox, 2007). Deep reinforcement can often learn an effective policy over this compact representation, as long as the algorithms are provided with a sufficiently large training set.

Exploration

As previously stated, a major challenge for an RL agent is balancing the tradeoff between exploring the environment and leveraging current information to garner as much immediate reward. To illustrate the problem, imagine a Q-learner is playing a game in which it has to select one of two cards. Card A pays +10 reward with 80% probability and card B pays +1 reward with 10% probability. Suppose the Q-learner selects randomly the first trial, selects card B and receives +1 reward. If the agent follows a greedy policy and always chooses the action with the highest Q-value, it will always choose card B. Alternatively, if the Q-learner has some policy to encourage exploration, it will quickly learn a higher Q-value for card A and learn to select it more frequently. However, the question remains to what degree the learner should explore. If the learner continues to explore after it is clear what the optimal policy is, then the learner will not be able to maximize its reward. As such, there is a tradeoff between exploration and exploitation. Any exploration policy that is based on Q-values will have the additional problem that the learned Q-values will be heavily influenced by the exploration strategy (and vis-versa) before the environment is well-learned.

For some small problems in stationary environments, like two arm bandit tasks (an MDP with a single state), there are optimal ways to balance the exploration and exploitation (Cohen et al., 2007). The optimal solution to exploration/exploitation tradeoff in such environments is to calculate the Gittins index for each action, which is the value of the stochastic process associated with each action, and select the option with the highest value (Cohen et al., 2007; Gittins, 1979). For more complex environments, the calculation of the Gittins index can be intractable. As such, a Q-learner will not always be able to explore optimally. Instead it will need a policy that explicitly addresses exploration, typically assumed to be stochastic. There are

multiple ways of accomplishing this. One could initialize the Q-values at a high level, encouraging exploration until the agent learns more accurate lower values (R-max). Alternatively, an agent could select the action with the highest value most often but choose randomly with some probability (ϵ-greedy). In the human behavioral literature, the Q-values are often used in a Boltzmann distribution (softmax function) that adds some degree of noise.

$$\Pr(a|s) = \frac{e^{Q(s,a)\beta}}{\sum_{a_i \in \mathcal{A}} e^{Q(s,a_i)\beta}}$$

The Boltzmann distribution has the attractive characteristic of the having a gain parameter β that controls the degree to which the learner stochastically explores. When used to fit behavioral data, this allows individual differences in exploration to be fit.

However, a limitation of this approach is that the degree to which an agent explores its environment is unrelated to learning. Because the gain parameter in the Boltzmann distribution is fixed, updating the Q-values does not affect exploration. In this sense, stochasticity in action selection due to the Boltzmann equation is a form of *undirected* exploration. An optimal learner will use *directed* exploration that scales their exploration with the uncertainty of the environment and their knowledge of it. (Yu and Dayan, 2005; Cohen et al., 2007). At a behavioral level, humans show evidence of both directed and undirected exploration, exploring more when uncertainty is high and the task horizon is long (Wilson et al., 2014a; Cockburn, 2015). Directed exploration can be implemented in neural network models of the basal ganglia through the dynamics of competitive representations of action values. When knowledge of a task is uncertain, activity for competing responses is roughly similar and action selection is generally stochastic . However, as the network accumulates evidence for one response over another, it will select the correct response more often and explore less (Franklin and Frank, 2015). At a broader level, directed exploration follows the pattern that an agent should explore when it has little knowledge of its environment and exploit when it knows it well. However, in order to govern this trade-off, the agent needs to track its uncertainty of its own knowledge.

Uncertainty and Learning

In an RL setting, uncertainty can be broken down into several types that interact over the course of learning but are asymptotically separate. For a model-free learner, one way to view learning is estimation of the state-action values. In a model-based setting, learning would be the estimation of $\mathcal{T}$ and $\mathcal{R}$. In either setting, the agent can be more or less confident in these estimations and we can refer to the agent's lack of knowledge as its *estimation uncertainty*. As the agent learns the MDP, estimation uncertainty will decline. The precise measure of estimation uncertainty will vary depending on the exact parameterization, but it can be conceptualized as the width of a posterior distribution of expected values for a Bayesian learner.

A separate form of uncertainty that might not affect exploration on its own is *expected uncertainty* or volatility. In any stochastic process, there is a certain amount of irreducible uncertainty about future outcomes that is a product of the inherent stochasticity. Even when the task is well-learned and the agent has very low estimation uncertainty, the outcome of any action will be unpredictable. If an optimal policy is known, then this uncertainty is ignorable as the agent should still choose the action with the highest expected discounted reward.[1] However, the stochasticity of the task will affect the speed with which an ideal observer learns an optimal policy. For example, if an agent is playing a two-armed bandit task in which the rewards are normally distributed around means +10 and +15, the agent will learn the optimal policy much more quickly if the standard deviation is 1 than it will if the standard deviation is 10. With less variance, the difference between the means is easier to detect.

As an agent may experience both forms of uncertainty while learning a task, if it behaves optimally it will change both its exploration and its learning dynamically. As estimation uncertainty can be used as a measure of the agent's the knowledge of the task, it is the relevant measure. When estimation uncertainty is high, any new information is relatively informative and the agent should learn more quickly

[1] There is substantial evidence that people are sensitive to risk and do not value changes in monetary rewards linearly (Kahneman and Tversky, 1984). As a result, the stochasticity of outcomes could change the optimal policy as actions with the same expected monetary value could have different subjective utility. While this may be a problem empirically, if the rewards of an MDP are delivered in units of subjective utility then expected uncertainty does not change the optimal policy beyond the expected discounted reward.

and explore more (Yu and Dayan, 2005; Cohen et al., 2007). However, as estimation uncertainty drops, the agent will want to ignore the irreducible expected uncertainty and pursue a greedy policy in order to maximize reward.

As an empirical matter, it does appear that people consider uncertainty both in terms of learning and exploration in a process that involves the prefrontal cortex (PFC) and anterior cingulate cortex (ACC) (Badre et al., 2012; Cavanagh et al., 2012; Frank et al., 2009; Behrens et al., 2007). Additionally, there is some fMRI evidence to suggest that the basal ganglia may also consider uncertainty directly (Summerfield et al., 2011). The question how people track uncertainty and how that is integrated in striatal learning is still an open question. From a modeling perspective, in order to take advantage of uncertainty an agent must track it. The Q-learning model defined earlier and many RL models of the basal ganglia use point estimates and do not explicitly track uncertainty. Bayesian models implicitly track uncertainty and adjust appropriately but it is unclear how mechanistically a direct Bayesian implementation is consistent with the empirical support for a dopamine based prediction error (but see Nassar et al. 2010).

Non-stationarity

Exploration and uncertainty become more difficult to solve and more important in non-stationary environments. In a non-stationary MDP, there is some probability that $\mathcal{T}$ and $\mathcal{R}$ will change from time-point to time-point. This lack of stationarity could be drift or a change point and may or may not be random. In a stationary MDP, the Q-values will eventually converge on the expected value if a declining learning rate is used and each state-action pair is visited enough times. The assumption of stationarity guarantees convergence but it may not reflect an ecological setting.

A common heuristic for dealing with non-stationarity is using a fixed learning rate when updating Q-values. Using a fixed learning rate has the effect of preferring more recent observations and discounting observations further in the past. This allows the Q-values to approximate a locally weighted average of the recent rewards and reduces the influence of previous experiences over time. If there is a change point and a declining learning rate is used, it will take substantially more observations to learn the new expected values of state action pairs than was required initially. By

using a fixed learning rate, the models are able to adapt more quickly. However, even a fixed learning rate can be slow to respond and may not be appropriate for different levels of predictable stochasticity. To illustrate the point, in a deterministic setting a fast learning rate will allow an agent to adapt quickly to any change. In contrast, a slower learning rate may be preferable in a noisy environment as it allows the learner to integrate information over a longer time frame, reducing its sensitivity to noise. Ideally, a learner would be able to scale its learning rate with uncertainty of its current estimate, learning quickly when something has changed and ignoring predictable negative feedback when the task is well-learned (Dayan and Yu, 2003).

The proper treatment of non-stationarity depends on the generative process that underlies it. For example, in a change point paradigm, the distribution of possible expected values can be approximated with a mixture distribution (Nassar et al., 2010; Wilson et al., 2010). If we assume that the value of a state-action pair of a bandit task has some probability ϕ of changing from trial to trial, then we can estimate its expected value with a mixture distribution. If we assume some prior distribution of expected values $\Pr(V_{s,a})$ then an approximation of our belief distribution for the value of a state-action pair at the next time point would be

$$\Pr\left(V_{s,a}^{t+1}|\mathcal{D}\right) = (1-\phi)\Pr\left(V_{s,a}^{t}|\mathcal{D}\right) + \phi\Pr(V_{s,a})$$

where $\Pr\left(V_{s,a}^{t}|\mathcal{D}\right)$ is the value of the posterior found through Bayesian updating. A fully Bayes treatment will allow for the belief distribution over previous trials to change as new observations are made, but this approximation can be useful as it allows the learner to treat the world as Markovian.

Estimating ϕ can be difficult, however, as it depends on the generative model. While it may not be reasonable to expect people to use the generative model in the general case, in limited cases people do follow strategies close to optimal and appear to integrate information about uncertainty (Nassar et al., 2010). The question of how people can do this is unresolved, but one possibility is that instead of directly estimating ϕ people track *unexpected uncertainty*. While the exact definition of unexpected uncertainty will also depend on the generative process, a heuristic can be derived from the derivative of the change in uncertainty (Payzan-LeNestour and Bossaerts, 2011; Yu and Dayan, 2005). The idea is that if a change has occurred in $\mathcal{T}$ or $\mathcal{R}$, then any general measure of uncertainty (for example, the entropy of distribution over

which action has the highest value) will increase as the agent receives an unexpectedly large degree of negative feedback. As such, this change in uncertainty can be a useful cue for signaling a change point. In a probabilistic process, randomness will cause transitory increases in uncertainty so it is important that an actor be able to ignore expected uncertainty. In an influential model by Yu and Dayan (2005), expected and unexpected uncertainty, signaled by cortical acetylcholine and norepinephrine respectively, interact to facilitate optimal inference.

1.1.2 Limitations of MDPs

While questions of how exploration, uncertainty and non-stationarity are unresolved, they are tractable problems that do not pose a general problem for the MDP as a model for the environment in which people learn (Dayan and Niv, 2008). However, in order to scale RL models and incorporate models of the basal ganglia in richer descriptions of learning and cognitive control, there are a number of challenges that may necessitate an expansion of the model. People can generalize their knowledge to new situations, resolve ambiguity and learn quite quickly, all of which can be difficult for model-free RL algorithms. While simple RL algorithms can solve MDPs, fully observable Markovian states pose a problem. If each state is treated as independent and fully observed, then there is no way to generalize between them. To put this another way, learning an MDP in a model-free way prevents learning a more appropriate structure. This is a poor model of human learning as people impose a structure in order to learn more quickly, even when the strategy is detrimental to performance (Acuña and Schrater, 2010; Yu and Cohen, 2008; Tenenbaum et al., 2011; Collins and Frank, 2013). A different model is needed to explain how people can learn and use these structures. However, it seems likely that the basal ganglia is involved in this process given the broad connectivity with the cortex and the rest of the brain as well as fMRI evidence linking the basal ganglia to abstract rule learning (Alexander et al., 1986; Badre et al., 2010). If this is the case, then a reasonable place to start is expanding limited models that have strong empirical support. If we loosen some of the assumptions of the MDP we may be able to adapt our algorithms to the new environments in a way that more closely matches human learning.

Structure in the MDP

Structure can address many of the problems that arise in learning and the MDP assumes some structure that serves as a simplifying assumption. If reinforcement-learning happens not in an MDP but over all of perception and action, the learner would have to find a policy mapping the entire perceptual space to the entire effector space. Because percepts will vary over time and are unlikely to be repeated exactly multiple times, a policy would likely need to map features of the perceptual space to the effector space. This runs into the problem that both the feature space and the effector space have very high dimensionality. Learning in high dimensional spaces is incredibly slow and any direct mapping between features and effectors is unrealistic because of the large number of training examples needed (Bellman, 1957). In order for any learning system to be effective at all, the dimensionality of both spaces needs to be substantially reduced. Arguably, the visual system dramatically reduces the dimensionality of the feature space in the form of object recognition (DiCarlo and Cox, 2007), much like a deep reinforcement learning system uses a convolutional neural net to reduces the input space (Mnih et al., 2013), but it's still an important concern as a the feature space may be relatively high dimensional with only a few relevant dimensions for any given situation.

One way to reduce dimensionality is adding structure. By using an MDP, we are assuming a particular structure in which all of the available perceptual information and history of perceptual information are summarized by the current state. Likewise, actions in the MDP can be viewed as structured in the sense that they may represent more complex behavior (e.g. "push the left button") than the low-level control of individual muscle fibers. The Markov assumption further simplifies the learning problem because it allows us to treat each state as independent. This is probably a reasonable assumption if we assume the states are reflective of some causal relationship between actions and the outcomes (Gershman and Niv, 2010). What is probably not a reasonable assumption is to equate states to percepts. For example, a rat could learn to associate pressing a lever with receiving a food reward. After a while, this relationship could change and pressing a lever no longer leads to the reward. How the animal should treat this situation is ambiguous. The rat could treat this as a non-stationarity, i.e. a permanent change in $\mathcal{R}$, perhaps caused by the food

dispenser running out of pellets. Alternatively, the rat could treat the change as a state transition, perhaps reflective of a task condition set by an experimenter that is still relevant. The latter possibility requires *perceptual aliasing*, or treating two similar percepts as separate. Both a non-stationarity or an unobserved state transition are plausible interpretations that may result in the animal taking different courses of action. *A priori*, there is no reason to expect that the occurrence of a change or its nature must be signaled. Assuming that percepts can be treated as states forces the interpretation of non-stationarity and this interpretation is sometimes inconsistent with empirical findings (Bouton and Bolles, 1979).

Ambiguity & Generalization

If we treat states as reflective of a causal structure and assume people and animals solve a perceptual aliasing problem to learn that structure, then it is possible to use MDPs as a model for more general cognitive processes. One phenomenon that could be explained this way is contextual learning, something that has been demonstrated in rats using the ABA renewal paradigm (Bouton and Bolles, 1979; Gershman et al., 2010). In an ABA renewal paradigm, an animal is trained on a conditioned stimulus in Context A, undergoes extinction in Context B and is then retrained in Context A. The animals re-learn the conditioned stimulus in less time than during the initial conditioning phase, suggesting the animals do not fully unlearn the relationship during extinction. One interpretation of these results is the rats learn a new state for Context B and learn that in the new state, the stimulus outcome relationship is different (Gershman and Niv, 2010; Gershman et al., 2010). When placed back in context A and reconditioned, the animals quickly identify the new state and behave appropriately. Because the animal has access to context, it can treat the relationship between the stimulus and outcome differently, resolving the ambiguity.

Interestingly, simulations suggest that renewal can occur as a result of drug interventions by preventing the basal ganglia from fully unlearning the association. This suggests these effects may be explainable without the animals learning a new structure (Wiecki et al., 2009). Because the MSNs of the striatum store evidence both for and against taking an action over separate populations of cells, it is possible to learn a new stimulus-action association without fully unlearning the previous association. This is

not consistent with a simpler model-free RL algorithm like Q-learning, which would have no way of retaining the learned representations. In the case of ABA renewal, however, learning different states for the two different stimulus outcome relationships is "correct" in that it mirrors the design of the experimenter (Gershman and Niv, 2010). This is supported empirically as a change of contexts is needed as the animals don't show renewal when they are retrained in the extinction context (Bouton and King, 1983).

In addition to resolving ambiguity by treating a stimulus differently in separate contexts, inferring a causal structure can be useful in generalization as well. ABC renewal is very similar to ABA renewal except that after extinction in context B, the animal is reconditioned in a new context C. In context C, the animal learns the contingency relationship quickly, suggesting they generalize their knowledge of the structure of context A to context C (Bouton and King, 1983; Gershman et al., 2010). Importantly, if the stimulus-outcome relationship is treated as the same in both contexts A and C, then this suggests some form of abstraction because the conjunction of stimulus and context does not uniquely predict the relationship. The states can be interpreted as abstract structure that governs the relationship between actions and outcomes. In contexts A and C, a particular structure is inferred and the animal learns one behavior, while in context B a different structure is inferred and the animal learns another behavior. Once an agent has learned a new state and associated behavior, this has the potential to be generalized to any new situation independent of any similarity in the perceptual space.

1.2 Partially Observable Markov Decision Processes

While treating states as an abstract but discoverable structure is potentially very powerful, it has the drawback of substantially increasing the complexity of the learning problem. By treating the states as discoverable and not fully-observed, we are casting the learning problem as solving a partially-observable MDP (POMDP). In a POMDP, instead of full knowledge of the state space, an agent will make an observation as it takes an action. From this observation, the agent will need to infer the current state. The POMDP can be formally defined by the tuple $< \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \Omega, \mathcal{Z} >$. The

elements $\mathcal{S}$,$\mathcal{A}$,$\mathcal{T}$, and $\mathcal{R}$ define the MDP, $\mathcal{Z}$ is the set of observations and Ω defines the observation functions $O(s', a, z) = \Pr(z|s', a)$. When an agent takes action a to move to state s it makes some observation $z \in Z$ and can use that to infer its position in the state-space.

Because the agent never knows its current state, its actions are conditional on its history of actions and observations. In practice, it can be useful to break down the policy selection into state estimation and planning. State estimation involves summarizing the agent's knowledge of the current state given its history of actions and observations while planning determines a policy conditional on that estimate. Conveniently, if we represent the state estimate as a probability distribution over the state space and update that estimate with Bayesian inference, than the estimate is a sufficient statistic of the agent's history (Kaelbling et al., 1998). This distribution is known as a belief distribution and denoted with b. Given that the agent has some initial belief state b, it takes action a and makes observation z. The resulting belief state b' is defined as:

$$\begin{aligned} b'(s') &= \Pr(s'|, z, a, b) \\ &= \frac{O(a, s', z) \sum_{s \in \mathcal{S}} b(s) T(s, a, s')}{\Pr(z|b, a)} \end{aligned} \tag{1.8}$$

We can also define the updating as $b' = \tau(b, a, z)$. Because b is a sufficient statistic for the agent's history in the POMDP, planning is Markovian, conditional on b (Kaelbling et al., 1998; Ross et al., 2008). We can use the Bellman equation to define the optimal policy as:

$$\pi^*(b) = \text{argmax}_a \left[\sum_{s \in \mathcal{S}} R(s, a) b(s) + \gamma \sum_{z \in \mathcal{Z}} V(\tau(b, a, z)) \Pr(z|b, a) \right]$$

This reduces to a problem that is piecewise linear and convex (Kaelbling et al., 1998). Unfortunately, while there are a number of algorithms that can generate exact solutions, the problem grows exponentially with observations and is computationally intractable. This points to the considerable computational complexity of solving a novel POMDP. When given a POMDP, an agent will need to learn the structure of the MDP in either a model-based or model-free way, learn the relationship between observations and states, generate a state estimate, and plan. Given the general intractability, this is probably too hard to expect a person to be able to solve optimally

in the general case.

The benefit of this complexity is that the learner would have access to a structure that can help it generalize its knowledge and adapt quickly to new circumstances. In principle, these abstractions are quite powerful and can grow in complexity with policies that transverse multiple states. However, the complexity needed to explain ABA or ABC renewal is very low, requiring only 2 states. In this case, the task would be to identify the current state and then take an action that is appropriate given the current state.

Additionally, if the basal ganglia are involved we might expect action selection to be dependent on action values represented in MSNs and updated through a dopamine prediction error. How the basal ganglia could incorporate the uncertainty of the current state during action selection is an open question but there are two heuristic algorithms for model-free action selection in POMDPs that could serve as a boundary to what is possible. One possibility is that state uncertainty is ignored and actions are only considered conditional on the most likely state. This is similar to the Most Likely State algorithm, which assumes the most likely state is the current state and selects the action with the highest Q-value for that state.

$$\pi_{MLS}(b) = \text{argmax}_a Q\left[\text{argmax}_s\left[b(s)\right], a\right]$$

As an alternative, one way to consider the state uncertainty is to weigh Q-values by the belief state.

$$\pi_{Q\text{MDP}}(b) = \text{argmax}_a \left[\sum_s b(s)Q(s,a)\right]$$

Q MDP is optimal if the agent will have full knowledge of the state following the next action (Cassandra, 1998). If the uncertainty of the current state is high, both heuristic methods will perform poorly. One strategy to overcome this is to use a dual-control mechanism that will take actions with respect to the Q-values when uncertainty is below some threshold and take action to actively reduce the uncertainty of the state space otherwise (Cassandra, 1998).

A further complication is both of the algorithms assume the underlying MDP is known. It may not be reasonable to assume that a person knows an arbitrary hidden structure and is attempting to plan given his uncertainty. For either of these algorithms to be a model of model-free learning in the basal ganglia, the Q-values

would have to be updated with feedback. If the uncertainty of the current state were low enough, then presumably this would not be problematic.

1.2.1 Structure Learning Models

Perhaps a more elegant method of solving a POMDP would be to attempt to infer the structure of the POMDP. This would allow the learner to assign a state conditional on an observation and its history. States in the POMDP can be thought of as similar to rules in that they govern the appropriate action to maximize reward. Learning the structure of the POMDP is similar to learning a set of appropriate rules to complete the task at hand. Following the metaphor, inferring the current belief state would be analogous to applying the appropriate rule.

A principled way of learning this structure is to propose a simple state space and progressively increase the complexity of the model. One possibility is to estimate the transition and observation functions while exploring the environment and to "split" the states as the evidence suggests (see Chrisman, 1992). The proposed POMDP can then be used to generate a belief-state estimate and update a Q function.

$$Q(\bar{b}, a) \approx \sum_{s \in S} b(s) Q(s, a)$$

$$Q(s, a) \leftarrow Q(s, a) + \alpha b(s) \left(r + \gamma \max_{a'} Q(\bar{b}', a') - Q(s, a) \right)$$

Like Q_{MDP}, the belief state probability is used to weigh the Q-values for action selection. In addition to action selection, the belief-state probability function can be used to weigh Q-value updating.

As a general framework, progressively increasing the complexity of the model is attractive because it reduces the computational complexity by limiting the model uncertainty considered (Doshi-Velez, 2009). Likewise, using Q-values is attractive because they have a meaningful interpretation in neural network models of the basal ganglia. Unfortunately, balancing exploration and exploitation is much more difficult in a POMDP and in this formulation, Q-values are not useful to balance this trade-off (Chrisman, 1992; Cassandra, 1998). They key problem is that if a state is not identified, then even if it has been visited often, its properties won't be learned. A separate problem is learning when to split the state. A good solution to both of these problems probably requires tracking state uncertainty, as does updating Q-values.

From a computational perspective, one alternative is to frame the problem in a fully Bayesian setting and propose a generative model for the learner. iPOMDP is a Bayesian non-parametric model that formalizes the learning problem as inference over a distribution of possible models (Doshi-Velez, 2009). Each model m is defined by its transition, reward and observation function such that $m = \{\mathcal{T}, \mathcal{R}, \Omega\}$. A non-parametric prior over the model space is set to allow an unbounded number of models of arbitrarily large state spaces. To reduce complexity the prior over the model space is biased to prefer small models. As the agent moves in the environment, it maintains each model as a hypothesis and tracks the posterior of the hypothesis space with Bayesian inference.

A particle filter is used to estimate the posterior over the models. In a particle filter, each proposed model m is maintained in a weighted distribution. For each model, its belief state can be estimated with equation (1.8), giving a hypothesis trajectory through its state space. Consequently, the likelihood equation over observations given a model $\Pr(o|m, a)$ is straightforward to calculate. The weights for each model $w(m)$ are updated after each time-step with

$$w_{t+1}(m) \propto \Pr(o|m, a) w_t(m)$$

Afterwards, the particles are resampled to generate a new posterior distribution. Because each hypothesis model tracks its trajectory, the agent's belief of its trajectory through the state space is updated with resampling. The updated posterior can be used to generate a new state estimate and policy choice.

1.3 The Basal Ganglia as a POMDP-solver

Linking basal ganglia circuitry to solving a POMDP is the gating model. Developed to model working memory tasks and the prefrontal cortex (PFC), the gating model is a neural network model of executive function that can learn to perform several tasks independently in a single network (Braver and Cohen, 2000; Rougier et al., 2005). Historically, executive function had been represented at an abstract level as rule-based symbol manipulation and the question of whether a neural network was capable of producing rule-governed behavior has been a matter of significant debate (Newell et al., 1989; Fodor and Pylyshyn, 1988; Rumelhart and McClelland,

1985; Johnson, 2004). In many ways the gating model addresses those concerns by proposing a method through which the prefrontal cortex can learn to implement rules without the use of symbolic representations or implementing a "homunculus" as control mechanism (Rougier et al., 2005; Hazy et al., 2007; Kriete et al., 2013). As argued before, rules can be thought of as similar to states and the process of learning can be represented by inferring a POMDP.

In the original gating model, working memory representations are maintained in the PFC and a DA signal directly innervating the PFC acts as a gate for working memory updating (Braver and Cohen, 2000). These maintained representations are used as an input for subsequent trials, allowing the network to solve tasks that require sequential information and keeping track of stimuli across time. An alternative version of the gating model proposes that learning to update working memory is dependent on the basal ganglia (O'Reilly and Frank, 2006). A dopamine prediction error signal is proposed to train gating in much the same way that it can train action selection. As a more general model of executive function, the gating model has been shown to be able to learn multiple rules and apply those rules appropriately to solve different behavioral tasks (Rougier et al., 2005).

While the gating model has been developed as a model of working memory and executive function, the model can be formally interpreted as a model-free POMDP solver (Todd et al., 2008). When an agent does not know the structure of an underlying MDP, one solution to finding a good policy is to augment the system with external memory (Peshkin et al., 1999). To augment model-free RL, an eligibility trace can be used. The SARSA(λ) model is a temporal difference algorithm like Q-learning that differs in that the prediction error of SARSA is evaluated on the next chosen Q-value and not the maximum (Peshkin et al., 1999; Sutton and Barto, 1998).

$$\delta = r + \gamma Q(s', a) - Q(s, a)$$

The eligibility trace acts as memory that allows the Q-values not visited to be updated after each time point.

$$Q(s, a) \leftarrow Q(s, a) + \alpha \delta e(s, a) \;\; \forall s, a$$

$$e(s, a) = \begin{cases} \gamma \lambda e(s, a) + 1 & \text{if } s = s_t \text{ and } a = a_t \\ \gamma \lambda e(s, a) & \text{otherwise} \end{cases}$$

SARSA is an "on-policy" algorithm, meaning it will estimate the Q-values for the learned policy (Q^{π}) and not the Q-values assuming all subsequent steps are optimal (Sutton and Barto, 1998). As a result, SARSA(λ) considers the exploration policy in its computation of Q-values. In an on-policy algorithm, the value of the current state depends on the current observation, the agent's history of observations, as well as the policy the agent has used. While SARSA(λ) cannot solve POMDPs in the general case, it can handle a POMDP that requires a controlling a single bit of memory (Peshkin et al., 1999). For POMDPs that require multiple bits of memory to solve, SARSA(λ) has credit assignment problems, but similar algorithms with more sophisticated credit assignment can handle more complex POMDPs.

Todd and colleagues (2008) simplified the gating model from a neural network model to a four-parameter actor-critic model that shares much in common with SARSA(λ). Their model contained two actors that individually controlled gating of working memory and action selection as well as a critic that learned the values of the observed stimuli. In an actor-critic framework, the critic learns the state-values and is used to calculate δ. Actor preferences are updated separately, using δ from the critic as the prediction error. In the Todd et al. model, eligibility traces were used when updating actor and critic values. Because the gating model learns to control it's working memory bit, a gating model effectively turns a non-Markovian problem into a Markovian one, much in the same way that augmenting the state-space in the problem presented in Peshkin et al. (1999). In doing so, Todd and colleagues were able to show that the model was capable of solving a 12-AX working memory task and generated similar errors as human subjects did in an artificial grammar task.

A substantial limitation of these model-free POMDP solvers is that they do not explicitly learn structure that is relevant and generalizable and instead, simply augment the state-space with a working memory bit. This is a useful strategy for tasks like the 12-AX task, as the presence of a working memory bit can act in combination with an ambiguous stimulus to determine a state. This is a generalization of the computational strategy employed by a simple recurrent network, a three layer recurrent network, in which the network's activity in the previous trial acts as an input in the next trial (Elman, 1990; O'Reilly and Frank, 2006). As a result, working memory does not need to contain generalizable structure and can be defined operationally as

whatever information is needed to solve the current non-Markovian problem (Todd et al., 2008).

A POMDP where the temporal order is not as informative could be harder to solve. Indeed, gating models can solve tasks under these conditions, suggesting gating models are doing something more complicated than "memorizing" policies (Bakker, 2001; O'Reilly and Frank, 2006). Of interest is what gating models are learning and particularly, what working memory represents. One possibility is rule structure of the task. Mathematically, this structure can be thought of as state-action contingencies where the states are partially defined by a memory object.

Experimentally, tasks that investigate rule use are typically designed to evoke very simple rules, such as learning different stimulus-response associations in different contexts. Often these associations can be learned without leveraging the rule structure, but human and primates both appear to use these rules in processes involving the PFC and BG (Wallis et al., 2001; Bunge et al., 2003; Badre et al., 2010; Collins and Frank, 2013). This suggests that if gating is a good model for this type of learning, then it needs to be able to represent a more generalizable structure than individual, learned policies.

1.3.1 Causal Inference and Rule Learning

Using a non-parametric Bayesian model similar to iPOMDP, Gershman and colleagues were able to model ABA and ABC renewal (2010). They modeled the effects of ABA and ABC renewal as a process of inferring a latent causal structure that is responsible for the observation. In this interpretation, the problem reduces to a clustering problem where each observation needs to be assigned to a cluster (i.e. cause) and the statistics of that cluster need to be learned.

Like iPOMDP, they used a non-parametric prior over the number of latent causes that allows the number of causes to grow with the complexity needed to solve the task. In their model, they used a Chinese Restaurant Process (CRP) to define the prior over causes. We can define a CRP as:

$$\Pr(c_{t+1} = k | c_{1:t}) = \begin{cases} \frac{N_k}{t+\alpha} & \text{if } k \leq K_t \\ \frac{\alpha}{t+\alpha} & \text{if } k = K_t + 1 \end{cases}$$

where K_t is the number of causes at time t, N_k is the number of observations originating from cause k and α is a free parameter that governs the formation of new causes (Aldous, 1985). In the CRP, the prior probability of a cluster is the number of points assigned to that cluster. While the potential number of clusters is unbounded, this property biases a smaller number of clusters and has a maximum number of clusters equal to the number of observations. Following each observation, the posterior distribution over the structure $\Pr(c_{1:t}|z_{1:t}, r_{1:t})$ is estimated using a particle filter.

The hypotheses for the particle filter in the Gershman et al. model are proposed cluster assignments. After each observation, the weights for the cluster assignment are updated and resampled in a method similar to iPOMDP. These weights are determined by the prior over the clusters, which biases smaller numbers of clusters, and the likelihood function determining the probability of observing a particular reward history given a cluster assignment. The statistics of the reward contingencies are updated analytically for each hypothesis. This differs from iPOMDP, where each hypothesis model was a complete POMDP and learning the statistics of the environment was accomplished by identifying a model. Thus, the model considers multiple reward schedule hypotheses simultaneously, with the uncertainty of a particular reward schedule as well as the cluster assignment contained in the distribution of hypotheses.

The model by Gershman and colleagues can explain ABA and ABC renewal and poses an interesting interpretation of the phenomenon as causal induction. A limitation of the model is that as a strictly computational model it does not directly propose a biological implementation. While there are theories as to how sampling methods like those used in a particle filter could be implemented biologically (Fiser et al., 2010), an alternative possibility is that the learning mechanism produces behavior that approximates Bayesian inference but uses an altogether different approach than mathematical integration.

Collins and Frank (2013) proposed an approximate implementation of a non-parametric structure learning model, explicitly comparing a Bayesian model with a biologically-plausible, multiple loop PFC-BG gating model. Their Bayesian model, C-TS, differed from iPOMDP and the Gershman et al. model in that states were not explicitly represented. Instead, the internal state of the model was represented

by both a 'task set' (rule) and a stimulus. This division is important as it allows the direct comparison with the gating framework in which working memory and the current stimulus each function as separate inputs. The unobserved task set TS is conditionally dependent on an observable context and a CRP was used as a prior over TS. Formally, the C-TS model can be defined:

$$\Pr(r|z,a,c) = \sum_{i=1}^{n} \Pr(r|z,a,TS_i)\Pr(TS_i|c)$$

The task of the learner is to learn the two conditional distributions. This breaks the task into learning to identify the correct rule $\Pr(TS|c)$ and the properties of that rule $\Pr(r|z,a,TS)$. In the task, the trials are interleaved and formally the model is equivalent to a POMDP in which the transitions are random. The agent learns the distributions and identifies the current task set with Bayesian inference. Instead of a particle filter, the task set with the maximum posterior probability is used for action selection and updating.[2] This is done to more closely relate C-TS to the neural network model. In a particle filter, resampling particles will change the belief over past trajectories, something that may be difficult to implement in a neural network. A major drawback of this approach is that it completely ignores any relevant uncertainty. While this approximation did not qualitatively change the pattern of behavior in the task examined by Collins and Frank, it may be more detrimental in more complex environments.

A key feature of the C-TS model is generalization. The model was trained in an environment in which two contexts uniquely determined two task sets. By learning the task sets, the C-TS model was able to learn more quickly than a similar model that treated the conjunction of context and stimulus as a state. When presented with new contexts, the C-TS model tended to attribute task sets to the new contexts instead of generating new task sets. Behaviorally, this was apparent through a performance benefit when previously learned task sets can be applied to the new context and a performance penalty when it cannot. When compared, human subjects display a similar pattern of behavior. Likewise, they are better fit by the C-TS model than a similar model that does not utilize structure.

[2]Using maximum *a posteriori* estimator for the task set also introduces order effects into the C-TS model. The effects are also found in the neural network model.

1.3.2 Multiple Loop BG-PFC Neural Network model

Linking the C-TS model to BG-PFC circuitry is a biologically plausible multiple loop BG-PFC gating model. The neural network model is an extension of a single loop BG model that has been used to make detailed and subsequently substantiated predictions of basal ganglia dependent learning in humans and animals in studies of disease as well as pharmacological and genetic manipulations (Frank, 2005; Frank et al., 2009; Beeler et al., 2012). Recall that action selection within the basal ganglia is thought to occur through the balance of excitation of D1 and D2-MSNs. Activity in D1-MSNs leads to the selective gating of an action through disinhibition of the thalamus, while D2-MSN activity inhibits disinhibition (Frank, 2005; Kravitz et al., 2010). Dopamine provides a training error to the system, biasing action selection in the same way a prediction error term in a TD model updates state-action values (Schultz, 1998; Bayer et al., 2007; Tsai et al., 2009). The training signal acts through D1 and D2 dopamine receptors, which effect the ability of cortical glutamatergic signaling to drive MSN activity (Surmeier et al., 2011). Activation of D1 receptors facilitates a sensitization of D1-MSNs to glutamatergic signaling while activation of D2 receptors have the opposite effect in the D2-MSNs. Activity within the MSN is sufficient to drive action selection (Kravitz et al., 2010). Likewise, stimulation of midbrain dopaminergic neurons that synapse in the striatum is sufficient to drive behavioral conditioning (Tsai et al., 2009). While there are important differences between the two, the similarities between TD learning and basal ganglia learning are strong.

The BG model was extended to explain working memory updating with the PBWM model (O'Reilly and Frank, 2006). In the original gating model, direct dopaminergic projections to the prefrontal cortex provided a gating signal that affect the stability of PFC representations (Braver and Cohen, 2000). As such, the gating signal was global and did not permit the gating of specific (or multiple) items into and out of working memory. In the PBWM model, in contrast, working memory updating acts through the basal ganglia. While working memory is dependent on the PFC (D'Ardenne et al., 2012; D'Esposito et al., 1995), the midbrain and striatum also appear to be involved in working memory updating (Yu et al., 2013). On an anatomical level, the basal ganglia are organized in parallel, largely segregated

functional "loops" that involve motor, prefrontal and other cortical areas (Alexander et al., 1986). In the PBWM model, BG "loops" may correspond to different functions but behave similarly. Working memory representation is gated by a prefrontal loop in the same way action selection is gated by a separate loop. Within the PFC, working memory is maintained across time with recurrence, simulating stable PFC representations. These representations feed into a second basal ganglia loop along with a stimulus to influence action selection. Much like in a non-working memory version of a BG model, learning to gate is driven by a dopamine training signal. For the PBWM to take advantage of working memory, it is forced to learn what to gate through the dopamine training signal. This specificity provides the advantage that multiple items can be gated into and out of memory independently.

In the PBWM, the dynamics of action selection are abstracted away and the training signal, while similar, is not a TD prediction error. The model by Collins and Frank was similar to the PBWM but used a more detailed representation of basal ganglia circuitry and used a more traditional prediction error signal. A second difference between the model used by Collins and Frank and the PBWM is the use of input and output gating. In a working memory task, the PBWM learns to gate a representation into memory if it is predictive of future reward (O'Reilly and Frank, 2006). It is also advantageous for the network to learn when to use the representation separately, as it prevents the memory from influencing a trial when it is not relevant (Hochreiter and Schmidhuber, 1997). The PBWM maintains multiple "stripes" that each contain a separate working memory representation. Over time, the network learns to appropriately gate in and out of memory. This allows the network to encode memories when they are useful and learn to apply them as needed. While input and output gating are possibly quite useful in rule-driven behavior, the mechanism was simplified in the Collins and Frank model. In this model, the network was provided the context as an input to the anterior striatum and had to learn to gate a single relevant task set to the PFC. The PFC representation along with the relevant stimulus acted as an input to the parietal cortex, which fed into the posterior striatum. The PFC representation also had direct connections to the posterior striatum. As a result, both the task set and a conjunction of the task set and stimulus were used by the posterior striatum to choose an action.

The multiple loop BG was trained on the same task as the C-TS model. Like the C-TS model, the multiple loop BG model learned more efficiently and was able to generalize old task sets to new contexts. When presented with a new context, randomly gating an appropriate task set leads the network to select the appropriate action. This leads to positive prediction errors, increasing the propensity of gating the task set and leading to generalization. Although this poses a credit assignment problem in that the neural network does properly apportion credit between task set gating and action gating, the network is still able to learn the task and generalize in a way consistent with the C-TS model.

Like the C-TS model, the behavior of the multiple loop BG model is consistent with human subjects. Both the model and human subjects show a performance benefit when generalizing and old context to a new set of stimuli is beneficial. Likewise, both can show a performance impairment when there is a new task set that is similar, but not identical, to an old task set. Broadly, this pattern of behavior is consistent with previous empirical work linking abstract rule discovery to the anterior prefrontal cortex (Badre et al., 2010; Badre and Frank, 2012).

The link between the levels of modeling and between human behavior suggest a novel framing of rule learning. Empirically, the PFC and basal ganglia appear to be involved in human rule-learning (Badre et al., 2010; Badre and Frank, 2012) and the pattern of behavior observed in subjects is consistent with Bayesian models and neural network models in which the BG gates a task set or rule into the PFC (Collins and Frank, 2013; Frank and Badre, 2012). Formally, the multiple loop BG model can be seen as an approximation of the C-TS model,[3] and the C-TS model, like the Gershman model, is a clustering algorithm. This suggests an interpretation in which we can view human rule learning as clustering observations into some structure (i.e. task set, rule) in which certain stimulus-response-outcome contingencies apply. This structure has the advantage of being Markovian, but has the limitation of being unobservable. Determining which structure to apply can be a difficult question as the complexity of the task grows.

[3]Parametric manipulations further link the two models as parameters in the neural network were found to correlate with parameters in the C-TS model. For example, the connectivity between the Context layer and the PFC in the BG model was correlated with the propensity of the C-TS model prior to generate a new TS.

1.4 State Uncertainty

Linking dopamine driven basal ganglia to structure learning is a substantial expansion of the basal ganglia that poses many new challenges and open questions as to how the system can cope with uncertainty. Through the lens of a POMDP, uncertainty over the relevant structure in a structure learning model can be thought of as *state uncertainty*, or the uncertainty of the state estimate. State uncertainty has peripheral effects that change the behavior of an optimal leaner beyond the challenges of identifying the current state. One notable challenge that is exacerbated by state uncertainty is the exploration/exploitation tradeoff. Recall that in a POMDP exploration is a more challenging problem than in an MDP because the states aren't observable. When exploring in an MDP, an agent uses the outcome of each each action to update its estimated state-action values or MDP model. This exploration can be directed efficiently by taking into consideration the uncertainty of these estimates (Gittins, 1979; Cohen et al., 2007). In a stationary MDP, this uncertainty will drop overtime as the agent gains more experience (Dayan and Yu, 2003). If the agent scales its exploration with its uncertainly, its behavior will lean toward more exploration as it gains knowledge of the MDP.

In a POMDP *state uncertainty*, will complicate the problem. Because an agent will not always have a good estimate of its state, it may have visited an area of the state space repeatedly and not been aware. This area can be unexplored (from the point of view of the agent) after repeated visits as the agent won't have learned any of properties about states it doesn't know it has visited (Chrisman, 1992). Second, state uncertainty could slow learning of the parameters of the underlying MDP (either Q-values or $\mathcal{T}$ and$\mathcal{R}$). For an agent that scales it learning rate with uncertainty in an MDP, the learning rate will be higher when the environment is unknown and will drop as the agent learns (Dayan and Yu, 2003). However, for a POMDP-solver like the state splitting model, Q-values are updated proportional to the belief probability of the state (Chrisman, 1992). This would slow learning Q-values when state uncertainty is high, and as a result, could prolong exploration.

A second complication is the interaction between state uncertainty and estimation uncertainty. For an agent in an MDP, the expected uncertainty will interact with the estimation uncertainty before the environment is well-learned. This happens because

the agent's uncertainty about the parameter estimates is high enough that stochastic noise can move the estimate. Likewise, we would expect estimation uncertainty (and expected uncertainty) to effect state uncertainty if the agent does not already know the POMDP. This would happen because correctly identifying state with unknown properties should be more difficult than identifying states with uniquely identifiable properties. This is a similar problem to perceptual aliasing, in that states with unknown properties are not necessarily distinguishable by observation.

Optimally planning with high state uncertainty can be very computationally costly for little appreciable gain. These effects partially motivate the non-parametric Bayesian POMDP solvers that grow with complexity (Doshi-Velez, 2009). By biasing the world model toward model with lower complexity, the need to consider uncertainty is substantially reduced. However, this generative process introduces a new role that state uncertainty could play: identifying when a new structure needs to be generated. In the non-parametric Bayesian models, this computation falls out of Bayesian inference with out a separate estimate of state uncertainty. In Bayesian models, all of the necessary uncertainties are contained in the probability distributions.

If BG-PFC interactions are reasonable approximation of this generative process, then it is an open question if and how state uncertainty plays a role. In the BG-PFC model by Collins and Frank, state uncertainty was ignored. At the beginning of training, the model randomly gated both actions and PFC stripes. Gating of both was trained by the dopamine signal, and eventually the network would learn to associate a task set with a context. Interestingly, this was similar to the behavior of subjects, who appeared to respond randomly early in training. It is entirely possible that people do not use state uncertainty to mediate ambiguity between state assignments and the state with the preponderance of evidence is used. This would solve several problems computationally at a cost of learning speed. Particle filters track multiple hypotheses and as a result, will track multiple historical paths through the state space. Updating the particle weights following an observation could change the identities and values of unobserved states, something that may be difficult to implement without sampling methods. Likewise, ignoring state uncertainty reduces the number of computations required.

There is evidence in other areas of learning that people may use similar heuristics instead of a fully Bayesian implementation. In a task where subjects are given multi-dimensional stimuli, modeling work suggests that subjects serially attend to the dimension of the stimuli in order to infer the predictive relationship (Wilson and Niv, 2011). An ideal observer would consider all relevant dimensions and update their predictive relationships simultaneously, but this is more computationally complex and requires tracking uncertainty. Similarly, in a sequential decision-making task, subjects appear to use a tree-search method with pruning strategy that reduces the number of considered paths (Huys et al., 2012). For sequential decisions, the number of possible paths can grow exponentially with the sequence length. Pruning algorithms reduce the size of the tree search problem, but the way people prune appears to be suboptimal. People avoids paths that incur a large loss early on, even if that pruning leads to a worse outcome. In both situations, people appear to use approximate instead of optimal strategies. Ignoring state uncertainty and choosing the state with the highest evidence (analogous to the most likely state algorithm) would be similar.

As of yet, there is limited evidence that state uncertainty is tracked and used to influence human behavior. State uncertainty is unobservable, task-dependent and may be correlated with other forms of uncertainty. As a result, task must be carefully designed in order to measure the effects of state uncertainty. This limits our ability to find evidence of state uncertainty consideration in task not specifically designed to investigate it.

A second complication is whether state uncertainty in one task is relevant in another. POMDPs are a formalism that may be applied in many domains. For example, the direct analog of the grid worlds used to study POMDPs may be spatial navigation tasks. While it is possible that state uncertainty in spatial navigation is treated in the same way as state uncertainty in rule learning, this is not guaranteed to be the case. Although the computational problem is the same, we might expect the specific implementation when the underlying neural architecture is different. As such, we might expect the treatment (or ignorance of) state uncertainty to be modality dependent.

This may result in separate representations of state uncertainty as well. Because we expect rule learning to involve working memory (Rougier et al., 2005; Badre et al.,

2010), we might expect to find state-uncertainty in areas involved in working memory processing like the PFC (D'Esposito et al., 1995; D'Ardenne et al., 2012). Likewise, we might expect states in a spatial navigation task to be encoded by the hippocampus, thought to be heavily involved in identifying location (Ekstrom et al., 2003; Cornwell et al., 2008). While both rule learning and spatial navigation both rely on the basal ganglia (Badre et al., 2010; Devan et al., 1996; Retailleau et al., 2012) and can be framed as POMDPs, there is no guarantee the treatment of uncertainty is the same.

However, this does not preclude a shared representation of state uncertainty across modalities. In spatial navigation, state uncertainty does appear to be considered in an PFC dependent process. In an fMRI study by Yoshida and Ishii (2006) in which subjects navigated a POMDP grid world, correlates of state uncertainty were found in the anterior and medial PFC. Yoshida and Ishii interpreted their findings as anterior PFC tracks state re-estimation (when observations contradict the agent's estimate) and medial PFC tracks belief state entropy. However, other interpretations are possible. In model proposed, belief state was tracked but planning behavior was not explicitly accounted for. One possible interpretation of their data is that state uncertainty affects state estimation. An alternate possibility is that state uncertainty is involved in the planning processes. Prefrontal cortex is thought to be involved in planning sequential decisions (Mushiake et al., 2006) as well as model-based reinforcement-learning (Balleine and Dickinson, 1998; Gläscher et al., 2010) and it is possible this uncertainty reflects these processes. Although it is unclear how state uncertainty affected the learning process, at minimum it appears state uncertainty is tracked during spatial navigation and we would expect it to affect some aspect of the planning process. It is equally unclear whether this state uncertainty would affect rule-governed behavior. Given that state uncertainty was found to be tracked by the PFC, it is plausible that rule-governed behavior might have access to uncertainty over the current rule.

Behaviorally, state uncertainty appears to be detrimental to performance in learning tasks. In a series of experiments by Gureckis and Love (2009a; 2009b), subjects played a task in which actions either improved their short-term or long-term rewards at a cost of the other. In their task, each action the subject made resulted in a state transition. High immediate reward actions led to transitions to lower value states

while low immediate reward value actions led to transitions to higher value states. They found that the absence of state cues impaired subjects' ability to follow a long term optimal strategy. When given state cues, subjects were much more likely to follow an optimal strategy. They also found that noise in the state cues decreased performance overall without an identifiable benefit. Interestingly, noise in the reward schedule (expected uncertainty) encouraged exploration and increased the likelihood a subject would find an optimal strategy. One interpretation of this data may be that people are unable to learn any structure equally and require explicit cueing to learn some structures. Overall, subjects found Gureckis and Love's task difficult and maybe be able to learn structure more easily in other situations, for example in sequential dependencies (Acuña and Schrater, 2010; Yu and Cohen, 2008; Green et al., 2010).

One area where the consideration of state uncertainty appears to have a benefit on performance is exploration. In an experiment by Knox and colleagues (2011), subjects were given a variant of a two-armed bandit task called the leapfrog task. In the leapfrog task, one of the two arms pays a higher reward than the other. Which arm pays the highest value switches; periodically, the lowered valued arm will jump in value to become the more valuable option. If the subject has been consistently choosing the arm with the highest value, the change is unobservable, making the process a POMDP. The optimal policy is then to explore as a function of belief state uncertainty, which will increase as a function of time. Knox and colleagues found subjects' exploratory choices fit a pattern consistent with belief drive exploration and inconsistent with a stochastic search policy. They further found that exploration behavior was not completely optimal in that subjects did not appear to take in to account the prospective value of information when choosing to explore. Instead, their behavior was best explained by a model that tracked belief uncertainty but myopically chose actions based on immediate expected value.

1.5 Compositionality in Neural Networks

So far, we have briefly covered the problems of generalization, state inference and non-stationarity and made the case these are all problems a reinforcement learner has to solve. We have not directly dealt with the topic of compositionally or productivity

with respect to reinforcement learning agents, a topic that will be the main focus of the balance of the presented work. Compositionally is rarely discussed with regard to reinforcement learning but exists as a point of controversy in a long-standing debate over the use of neural networks as a model of cognition. In this section, we will introduce the issue of compositionally in neural networks in order to motivate the rest of the work.

Following the development of the back-propagation algorithm in the 1980s, there was a substantial amount of excitement and debate of over the use of neural networks as a model for cognition. In contrast to the symbolic processing models that had previously dominated, like those proposed by Newell and Simon in the 1950s, connectionist models allowed for softer representations that facilitated problem solving through measures of similarity instead of a rigid rule-structure (Rogers and McClelland, 2004). The rise in interest in these models re-sparked an old debate over whether and how connectionism could be considered a model of cognition. In many ways, this was a debate over what phenomena and models are the appropriate level of analysis to be called "cognition". At its core, though, the debate addressed a substantial theoretical question over representation: is cognition better explained as a rule-based, symbolic system or by an associative system that learns through statistical regularities? Both types of systems are advantageous; rule-based systems are clearly advantageous in tasks like mathematics while associative systems can leverage predictability in the world in a powerful way to implement pattern recognition (Sloman, 1996). We might expect a complete description of the cognitive architecture to include both such systems, and when framed this way, the debate over connectionism was a continuation of a much older debate discussed at length by earlier psychologists.

At a superficial level, the debate is trivial. Both symbolic processing models and recurrent neural networks are capable of solving any arbitrary problem (Turing, 1950; Siegelmann and Sontag, 1991, 1995), suggesting the difference between the models may be the description of formally equivalent mathematical systems. Problems thought to require one method – either symbolic processing model or neural networks – can be proved to be solvable by the other. Indeed, problems that were originally thought to require productive thought and unlearnable by an associative system, such as chess (Newell and Simon, 1959; Campbell et al., 2002), Go (Silver et al., 2016) and

the learning past tense of verbs (Rumelhart and McClelland 1985; but see Pinker 2001) have been solved by algorithms that cannot be described as productive. Furthermore, the structured representations needed for productivity can be imbedded in a neural network (Smolensky, 1990; Plate, 1995) and theoretical models suggest fronto-striatal networks can solve simple binding problems(Kriete et al., 2013).

However, a proof that a model can exist to solve any problem trivializes the difficulty with which it can be to find an algorithm to solve a specific problem. While a fully connected neural network can solve any arbitrary problem with the correct weights, training one without heavy constraints on its structure requires an impractically large number of training examples (Geman and Doursat, 1992). Thus, as an empirical and theoretical matter, it is far from clear if symbols and causal reasoning are obviated as a useful description of human thought by neural networks (Fodor and Pylyshyn, 1988; Crick, 1989; Pinker, 2001). Furthermore, the existence of a neural network that can solve a problem thought to require the productive use of symbols doesn't solve the problem, it may merely mask the computational and algorithmic strategy in a neural network implementation (Griffiths et al., 2010).

While addressing this debate is beyond the aim of the current work, it is relevant to consider productivity and its relationship to goal-directed behavior. A key implication of a productive system is a system of component pieces that can be reused in a novel way (Fodor and Pylyshyn, 1988; Sloman, 1996). In the language of reinforcement learning, this is similar to planning over a world model (transition function) and an objective (reward function). In short, a strong interpretation of goal-directed behavior is a productive system of planning. This requires a compositional structure. Previous research has interpreted the differentiation between model-based and model-free reinforcement learning as a division between a habitual and a more controlled process. From theoretical perspective, this isn't necessarily the case. Model-based algorithms are not strictly productive, for example the DYNA-Q algorithm uses a model to solve for state-action values and can flexibly adapt to changes in the reward function, but does not pose a novel solution to a novel problem (Sutton and Barto, 1998; Kulkarni et al., 2016)

A stricter criteria of a productive reinforcement learning system would be one that takes components learned across experiences and recombines them to create a novel

solution to a novel problem. This requires the generalization of component pieces as well as model-based control. In the following chapter, we present such a model, and while we do not make the claim the model represents a productive system, we claim that the imbedded compositional representations provide previous models of generalization a natural extension towards the more complex learning behaviors.

Chapter 2

Theoretical and normative account of compositionally in clustering models of generalization

2.1 Introduction

Compared to artificial agents, humans exhibit remarkable flexibility in our ability to rapidly, spontaneously and appropriately learn to behave in unfamiliar situations, by generalizing past experience and performing symbolic-like operations on constituent components of knowledge (Marcus et al., 2014). Formal models of human learning have cast generalization as an inference problem in which people learn a (latent) shared task structure across multiple contexts and then infer which causal structure best suits the current scenario (Gershman et al., 2010; Collins and Frank, 2013). In these models, a context, typically an observable (or partially observable) feature of the environment, is linked to a learnable set of task statistics or rules. Based on statistics and the opportunity for generalization, the learner has to infer which environmental features (stimulus dimensions, episodes, etc.) should constitute the context that signals the overall task structure, and, simultaneously, which features are indicative of the specific appropriate behaviors for the inferred task structure. This learning strategy is well captured by Bayesian nonparametric models, and neural network approximations thereof, that impose a hierarchical clustering process onto learning

task structures (Collins and Frank, 2013, 2016b). A learner infers the probability that two contexts are members of the same task cluster via Bayesian inference, and in novel situations, has a prior to reapply the task structures that have been more popular across disparate contexts, while also allowing for the potential to create a new structure as needed. Empirical studies have provided evidence that humans spontaneously impute such hierarchical structure, which facilitates future transfer, whether or not it is immediately beneficial – and, indeed, even if it is costly – to initial learning (Collins and Frank, 2013, 2014, 2016b).

A similar approach has been adopted for artificial agents in the domain of lifelong learning, in which previously learned policies are reused in novel tasks when their statistics are similar enough to a previously solved task (Rosman et al., 2016; Mahmud et al., 2013; Wilson et al., 2012). However, a key limitation to these models of both human and artificial learning is that the task statistics are typically generalized as a whole. That is, in a new context, a previously learned policy can either be reused or a new policy must be learned from scratch.

These clustering models can account for aspects of human generalization that are not well explained by standard models of learning. Nevertheless, because task structures as a whole are either reused or not, they are still limited in their ability to reuse and share component parts of previous knowledge; that is, they are not *compositional*. Importantly, in ecological settings, experiences across contexts often share a partial similarity.

To provide a naturalistic example, an adept musician can transfer a learned song between a piano and a guitar, even as the two instruments require completely different physical movements, implying that their goals can be generalized independently of the actions needed to achieve them. A clustering model that generalizes entire task structures cannot account for this behavior and worse, predicts an unlikely interference effect where the similar outcome of playing the same song on two instruments results in the model incorrectly pooling motor policies across instruments. This inability to share partial information is not unique to clustering models, as algorithms agnostic to human behavior often generalize learned policies (Goodman et al., 2011; Rosman et al., 2016; Borsa et al., 2016) and are unable to generalize across two contexts that share the same goal outcome but require different behaviors to achieve

it. Thus, we propose a general strategy to address this problem by independently clustering and generalizing component properties of the task structure, which can be flexibly combined in novel contexts.

This degree of compositionality is critical for flexible goal-directed behavior as its absence makes surprising and often undesirable predictions. To continue with the music example, if the actions needed to produce a note are represented as the same task structure as the notes needed to play (and in what order), an agent that generalizes both together would need to relearn a song from scratch to play it on a new instrument. Likewise, a robot that has learned to transfer a package from one room to another would need to relearn to do so if an obstacle is put in its way. Nonetheless, this can be difficult for computational frameworks as the assumed latent structure needed to generalize is incompatible with basic compositionality. To effectively generalize, an agent needs to structure their experiences in a useful way, clustering related experiences and segregating unrelated experiences.

Here, were propose a framework for compositionality in models of context-based generalization by decomposing the learned structure of tasks into a reward function and a transition function. These two independent functions of a Markov decision process are suitable units of generalization: if we assume that an agent has knowledge of a state-space and the set of available actions, then the reward and transition functions are sufficient to determine the optimal policy. We propose two alternative models, an independent clustering agent that generalizes rewards and transitions compositionally and a joint clustering agent that generalizes them together. We show that these two models lead to different predictions depending on the task environment. In environments where there is a clear, discoverable relationship between transitions and rewards, joint clustering can lead to better generalization. Nonetheless, independent clustering may be a better strategy overall as even in cases where there is a discoverable relationship, independent clustering can provide better generalization if that relationship is weak, difficult to discover, or costly to do so.

2.1.1 Model Description

To provide a test-bed for characterizing the benefits of compositional vs. joint structure, we consider a series of navigation tasks, whereby an agent needs to learn transition functions (the consequences of its actions in terms of subsequent states) and reward functions (the reward values of locations, or goals), as it navigates through space. At each point in time, the agent is given a state tuple $s =< x, c >$ where $x \in \mathbb{R}^d$ is a vector of state variables (e.g., a location vector in coordinate space) and $c \in \mathbb{R}^n$ is a context vector. Here, we define "context" as a vector denoting some mutable property of the world (e.g., the presence or absence of rain, an episodic period of time, etc.) that constrain the statistics of the task domain, whereby these task statistics are consistent for each context that cues the relevant task. Formally, for each context we can define a Markov decision process (MDP) with state variables $x \in X$, actions $a \in A$, a reward function mapping state variables and actions to a real valued number $R : X \times \mathcal{A} \rightarrow \mathbb{R}$, and a transition function mapping state variables and actions to a probability distribution over successors $T : X \times \mathcal{A} \rightarrow \Pi(X)$.

For the purpose of simplicity, we assume that the agent knows the spatial relationship between states and learns how its actions take it from one space to another. Specifically, we assume the agent knows a set of *cardinal movements* $A \in \mathcal{A}_{card}$, where each cardinal movement is a vector that defines a change in the state variables with regard to the known spatial structure per unit time. (For example, in a two dimensional Gridworld we can define North $= \langle dx/dt, dy/dt \rangle = \langle 0, 1 \rangle$ as a cardinal movement). We can thus define a transition function in terms of cardinal movements $T(x, A, x'|c) = \Pr(x'|x, A, c)$ that we assume the agent knows and cast the problem of learning to navigate as the problem of learning some function that maps actions to cardinal movements $\phi_c : \mathcal{A} \rightarrow \mathcal{A}_{card}$, which we assume to be independent of location. This simplifying assumption has the benefit of providing a model of human navigation, whom we assume understand spatial structure. In general, we can think of the relationship between actions and cardinal movements as dependent on environmental conditions, and thus, context (for example, wind condition can change the relationship between actions and movements in space for an aerial drone). Similarly, we can express the reward function in terms of cardinal movements, $R_c(x, A) = \Pr(r|x, A, c)$.

This allows us to consider how the agent receives reward as it moves through coordinate space (as opposed to how it receives reward as it takes actions in space).

The task of the agent is to learn a policy (a function mapping state variables to primitive actions, for each context; $\pi^*|c : X \rightarrow \mathcal{A}$) that maximizes its expected future discounted reward (Kaelbling et al., 1998). Given a known transition function and reward function, the optimal policy given this task can be defined as:

$$\pi_c^*(x) = \arg\max_a \left[\sum_{A \in \Psi} \phi_c(a, A) \left[R_c(x, A) + \gamma \sum_{x' \in X} T_c(x, A, x') V_c(x') \right] \right] \tag{2.1}$$

where $V_c(x)$ is the optimal value function is defined by the Bellman equation:

$$V_c(x) = \max_a \left[\sum_{x' \in X} T_c(x, A, x') \left[R_c(x') + \gamma V_c(x') \right] \right] \; \forall \; x \in X \tag{2.2}$$

As the relationship between locations in space, T_c, is known to the agent, it is sufficient to learn the cardinal mapping function $\phi_c(a, A)$ and reward function $R_c(x, A)$ to determine an optimal policy.

While the optimal policy is dependent on both the mapping function and reward function, crucially, the optimal value function is not: it is dependent only on the reward function (and T_c). Consequently, an agent can determine an optimal policy as a function of movements through space:

$$\pi_c^*(x) = \arg\max_A \left[R_c(x, A) + \gamma \sum_{x' \in X} T_c(x, A, x') V_c(x') \right] \tag{2.3}$$

This allows the agent to learn how it can take an action to move through space – the mapping function $\phi_c(a, A)$ – independently from the desirability of these moves, $R_c(x, A)$. This distinction allows for compositionality during generalization, as we will discuss in the following section.

2.1.2 Context clustering as generalization

A common strategy to support task generalization is to cluster contexts together, assuming they share the same task statistics, if doing so leads to an acceptable degree of error (Mahmud et al., 2013). This logic underlies both models of human generalization (Gershman et al., 2010; Collins and Frank, 2013) and category learning

(Sanborn et al., 2010). Clustering models of human generalization typically rely on a non-parametric Dirichlet process, commonly known as the Chinese restaurant process (CRP) as a clustering prior in a Bayesian inference process. The CRP enforces popularity-based clustering to partition observations, so that in this case the agent will be most likely to reuse those tasks that have been most popular across disparate contexts (as opposed to across experiences; (Collins and Frank, 2016b)), and has the attractive property of being a non-parametric model that grows with the data (Aldous, 1985). Consequently, it is not necessary to know the number of partitions *a priori* and the CRP will tend to parsimoniously favor a smaller number of partitions.

As in prior work, we model generalization as the process of inferring the assignment of contexts $k = \{c_{1:n}\}$ into clusters that share common task statistics. But here, we decompose these task statistics to consider the possibility that that all contexts $c \in k$ share either the same reward function and/or mapping function, such that $R_k(x, A) = R_c(x, A) \; \forall \; c \in k$ and/or $\phi_k(a, A) = \phi_c(a, A) \; \forall \; c \in k$. (We return to the "and/or" distinction, which affects whether clustering is independent or joint across reward and mapping functions, in the following section). Formally, we define generalization as the inference

$$\Pr(c \in k|\mathcal{D}) \propto \Pr(\mathcal{D}|k) \Pr(c \in k) \tag{2.4}$$

where $\Pr(\mathcal{D}|k)$ is the likelihood of the observed data $\mathcal{D}$ given cluster k, and $\Pr(c \in k)$ is a prior over the clustering assignment. As in previous models of generalization, we use the CRP as the cluster prior. If contexts $\{c_{1:n}\}$ are clustered into $N \leq n$ clusters, then the prior probability for any new context $c_{n+1} \notin \{c_{1:n}\}$ is:

$$\Pr(c_{t+1} \in k|c_{1:n}) = \begin{cases} \frac{N_k}{N+\alpha} & \text{if } k \leq K_n \\ \frac{\alpha}{N+\alpha} & \text{if } k = K_n + 1 \end{cases} \tag{2.5}$$

where N_k is the number of contexts associated with cluster k and K_n is the number of unique clusters associated with the n observed contexts. If $k \leq K_n$, then k is a previously encountered cluster whereas if $k = K_n + 1$, then k is a new cluster. The parameter α governs the propensity to assign a new context to a new cluster, that is to create a new task. Higher values of α lead to a greater prior probability that a new cluster is created and will favor a more expanded task space overall, leading to

reduced likelihood of reusing old tasks. Thus, the prior probability that a new context is assigned to an old cluster is proportional to the number of contexts in that cluster (popularity), and the probability that it is assigned to a new cluster is proportional to α. As a non-parametric generative process, the prior allows the number of clusters to grow as new contexts are observed. Importantly, this process is exchangeable, and as such, the order of observation does not alter the inference of the agent (Aldous, 1985).

2.1.3 Independent and joint clustering

As we noted above, there are two key functions the agent learns when navigating in a context: $\phi_c(a, A)$ and $R_c(x, A)$. These functions imply that the agent could cluster $\phi_c(a, A)$ and $R_c(x, A)$ jointly, or it could cluster them independently, such that it learns the popularity of each marginalizing over the other. Formally, the *independent clustering* agent assigns each context c into two clusters via Bayesian inference as in [Eq. 2.4] and using the CRP prior for each cluster [Eq. 2.5]. The likelihood function for the two assignments are the mapping and reward functions

$$L(\mathcal{D}|k_\phi) = \phi_k(a, A) \tag{2.6}$$

and

$$L(\mathcal{D}|k_R) = R_k(x, A), \tag{2.7}$$

respectively. Conversely, the *joint clustering* agent assigns each context c into a single cluster k via Bayesian inference [Eq. 2.4] and, like the independent clustering agent, uses the CRP as the prior over assignments [Eq. 2.5]. The likelihood function for the context-cluster assignment is the product of the mapping and reward functions

$$L(\mathcal{D}|k) = \phi_k(a, A)R_k(x, A) \tag{2.8}$$

The joint clustering agent is highly similar to previous non-compositional models of task structure learning and generalization, which have previously been shown to account for human behavior (but without specifically assessing the compositionality issue) (Collins and Frank, 2013, 2016b).

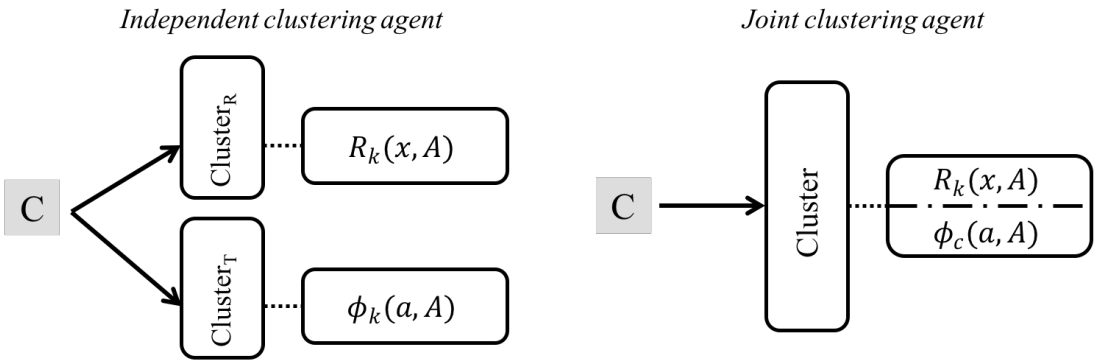

Figure 2.1: Schematic of Independent and Joint clustering agents. Both use the mapping function (a reduced transition function) and reward functions as the likelihood for cluster. The independent clustering agent considers the mapping transition function and reward functions separately for clustering, and is therefore compositional, whereas the joint clustering agent only considers the product of the two functions as a single task unit.

2.2 Gridworld simulations

We first consider two sets of simulations to expose the complementary advantages afforded by the two sorts of clustering agents depending on the statistics of the task domain, using a common set of parameters. A third simulation explores how the benefits of clustering can compound in a modified *Rooms* problem previously used to motivate the need for hierarchical RL approaches (Sutton et al., 1999; Botvinick et al., 2009), and we show how our independent clustering model with hierarchical decomposition of the state space can facilitate more rapid transfer than that afforded by standard approaches. Finally, we conduct an information theoretic analysis to formalize the more general conditions under which each scheme is more beneficial. Code for all of the simulations presented here have been made available in our GitHub repository: `https://github.com/nicktfranklin/IndependentClusters`

2.2.1 Simulation 1: Independent Reward and Mapping Task Statistics

In the first set of simulations, we simulated the agents learning a task domain with four contexts. In every trial, a "goal" location was hidden in a 6x6 grid-world. Agents

were randomly placed in the grid-world and explored action selection until the goal was reached, at which point the trial ended and the agent was placed in the subsequent trial. The task of the agent is to find the goal location as quickly as possible (encoded as +1 reward for finding the goal and a 0 reward otherwise). The agents had a set of eight actions available to them $\mathcal{A} = \{a_1, ...a_8\}$, which could be mapped onto one of four cardinal movements $\mathcal{A}_{card} = \{\text{North}, \text{South}, \text{East}, \text{West}\}$. Both the goal locations and mappings were stationary across all trials within a context and the agents were trained on four trials per context. Each of the four contexts had a unique combination of one of two goal locations and one of two mappings [Fig 2.2.1, left], and hence knowledge about the reward function or mapping function for any context was not informative about the other function. However, because the reward and mapping functions were each common across two contexts, the independent clustering agent should be able to leverage such structure to improve generalization without being bound to the joint distribution of mappings and rewards.

Three agents were simulated in the task environment: both the independent and joint clustering agents as well as a "flat" agent that does not cluster contexts at all and hence has to learn anew in each context. (The flat agent is a special case of both the independent and joint clustering agents such that $k_i = \{c_i\} \; \forall \; i$). Models were simulated using hypothesis-based inference. Hypotheses consist of a proposal assignment of contexts in to clusters, $h : c \in k$, defined generatively, such that when a new context is encountered the hypothesis space is augmented. For each hypothesis, maximum likelihood estimation was used to generate estimates $\hat{\phi}_k(a, A) = \hat{\Pr}(A|a, k)$ and $\hat{R}_k(x) = \Pr(r|x)$. To encourage optimistic exploration, $\hat{R}_k(x)$ was initialized to the maximum observable reward ($\Pr(r|x) = 1$) with a low confidence prior.

The belief distribution over the hypothesis space is defined by the posterior probability of the clustering assignments [Eq. 2.4]. Calculating the posterior distribution over the full hypothesis space is computationally intractable as the size of the hypothesis space grows combinatorially with the number of contexts. As an approximation, we pruned hypotheses with small probability (less than 1/10x posterior probability of the maximum *a posteriori* (MAP) hypothesis) from the hypothesis space. We further approximated the inference problem by using the MAP hypothesis during action selection (Collins and Frank, 2013; Sanborn et al., 2006). Value iteration was used

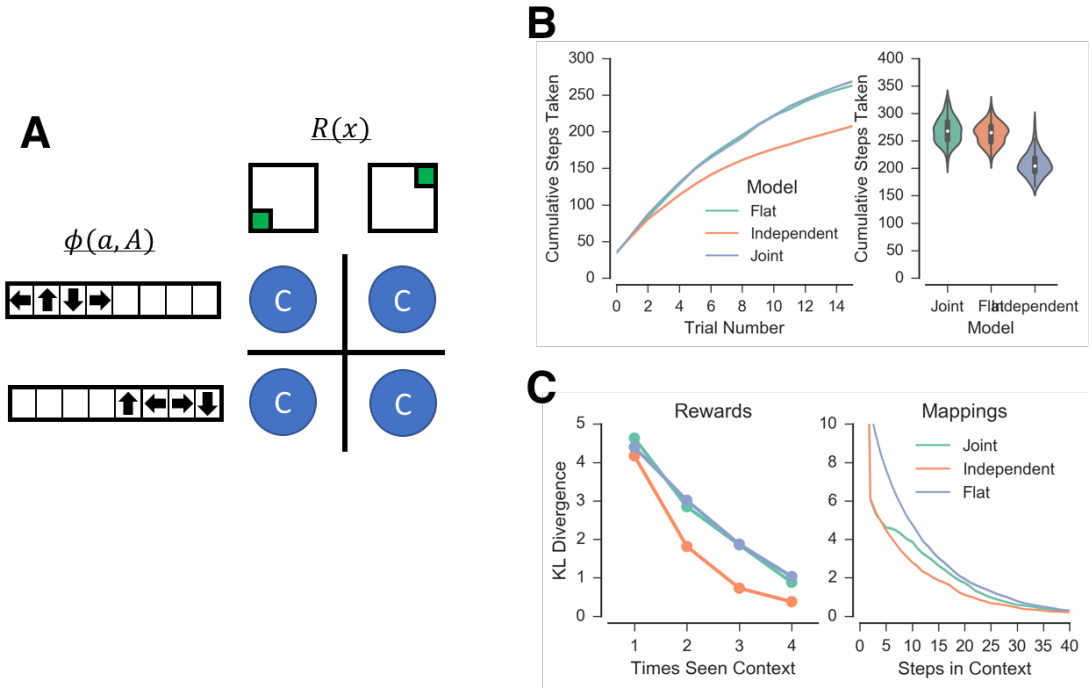

Figure 2.2: *A*: Schematic representation of the first task. Four contexts (blue circles) were simulated, each paired with a unique combination of one of two goal locations (reward functions) and one of two mappings. *B*: Cumulative number of steps taken by each agent shown across trials (left) and at completion (right). Fewer steps taken represent faster learning. *C*: KL-divergence (information gain) of the models estimates of the reward (left) and mapping (right) functions as a function of experience. Lower KL-divergence represents better function estimates. Time shown as the number of trials in a context (left) and the number of steps in a context collapsed across trials (right) for clarity.

to solve the system of equations defined by [Eq. 2.3] using the values of $\hat{\phi}_k(a, A)$ and $\hat{R}_k(x)$ associated with the MAP hypothesis(es). A state-action value function, defined here in terms of cardinal movements

$$\hat{Q}_k(x, A) = \hat{R}_k(x, A) + \gamma \sum_{x' \in X} T(x, A, x') \hat{V}_k(x') \tag{2.9}$$

was used with a softmax action selection policy to select cardinal movements:

$$\Pr(A|x, k) = \frac{e^{\beta \hat{Q}_k(a,A)}}{\sum_{A \in \mathcal{A}_{card}} e^{\beta \hat{Q}_k(a,A)}} \tag{2.10}$$

where β is an inverse temperature parameter that determines the tendency of the agents to exploit the highest estimated valued actions or to explore. Lower level actions (needed to obtain the desired cardinal movement) were sampled using the mapping function:

$$\Pr(a|x,k) = \sum_{A \in \mathcal{A}_{card}} \hat{\phi}_k(a,A) \Pr(A|x,k) \tag{2.11}$$

We first simulated the independent and joint clustering agents as well as the flat agent on 150 random task domains using the parameter values $\gamma = 0.75$, $\beta = 5.0$ and $\alpha = 1.0$ (below we consider a more general parameter-independent analysis). Each of the four contexts was repeated 4 times for a total of 20 trials. The independent clustering agent completed the task more quickly than either other agent, completing all trials in an average of 205.2 (s=17.2) steps in comparison to 267.72 (s=19.1) and 263.2 (s=23.7) steps for the joint clustering and flat agents, respectively [Figure 2.2, B]. This is due to better estimates of reward and mapping functions across time. The independent clustering agent has lower KL-divergence for its estimate than either the joint clustering or flat agents, which show a similar degree of information gain [Figure 2.2, C, left]. Similarly, the independent clustering agent's estimate has lower KL-divergence than either of the two other agents, though both clustering agents show faster learning than the flat agent [Figure 2.2, C, right].

2.2.2 Simulation 2: Dependence in Task Statistics

We next simulated all three agents on separate task domain in which there was a discoverable relationship between the reward and mapping functions across contexts, such that knowledge about one function is informative about the other. There were with four orthogonal reward functions and four orthogonal mappings across eight contexts, with each pairing of a reward and mapping function repeated across two contexts, permitting generalization [Figure 3, A]. As before, 150 random task domains were simulated for each model using the parameter values $\gamma = 0.75$, $\beta = 5.0$ and $\alpha = 1.0$. Each of the eight contexts was repeated 4 times for a total of 20 trials. In these simulations, both clustering agents show a generalization benefit, completing the task more quickly than the flat agent [Fig 2.3, B]. The joint clustering agent showed the largest generalization benefit, completing all trials in average of 5167

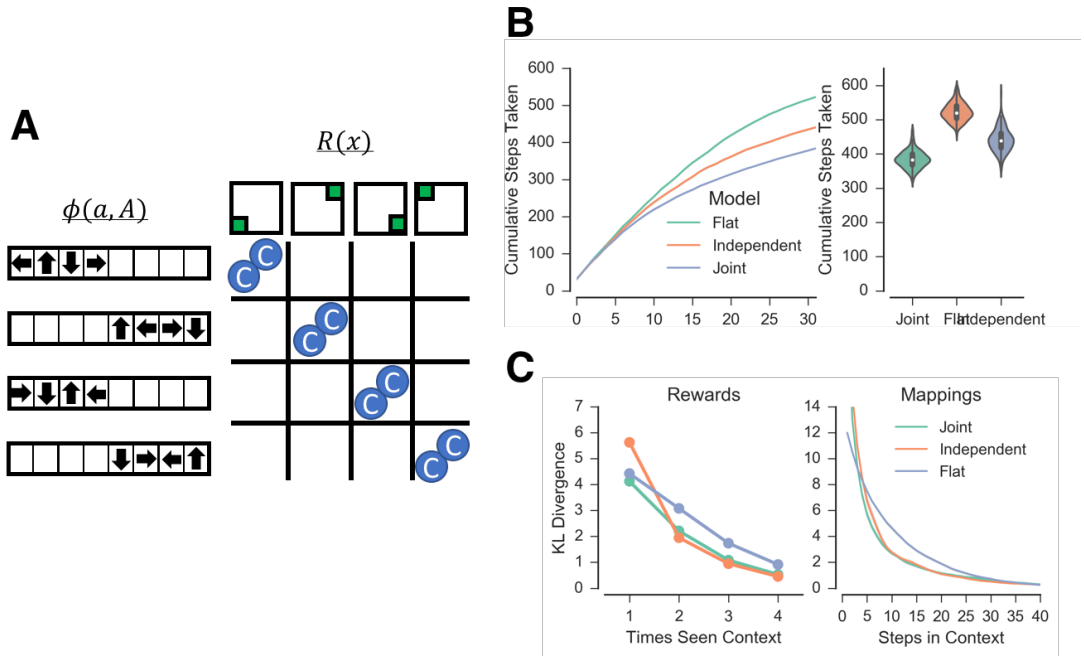

Figure 2.3: A: Schematic representation of the second task domain. Eight contexts (blue circles) were simulated, each paired with a combination of one of four orthogonal reward functions and one of four mappings, such that each pairing was repeated across two contexts providing a discoverable relationship. B: Cumulative number of steps taken by each agent shown across trials (left) and at completion (right). C: KL-divergence of the models estimates of the reward (left) and mapping (right) functions as a function of time.

(s=1051) steps in comparison to 6742 (s=1674) for the independent clustering agent and 8360 (s=841) steps for the flat agents.

Both the independent and joint clustering agents had similar estimates of mapping functions across time [Fig 2.3, C, right] whereas the independent clustering agent uniquely shows an initial generalization deficit in the reward function, as measured by KL divergence in the first trial in a context [Fig 2.3, C, left].

2.2.3 Simulation 3: Exploration Costs in the Diabolical Rooms Problem

A primary benefit of generalization is a reduction exploration costs. If an agent generalizes effectively, it can determine a suitable policy without fully exploring a

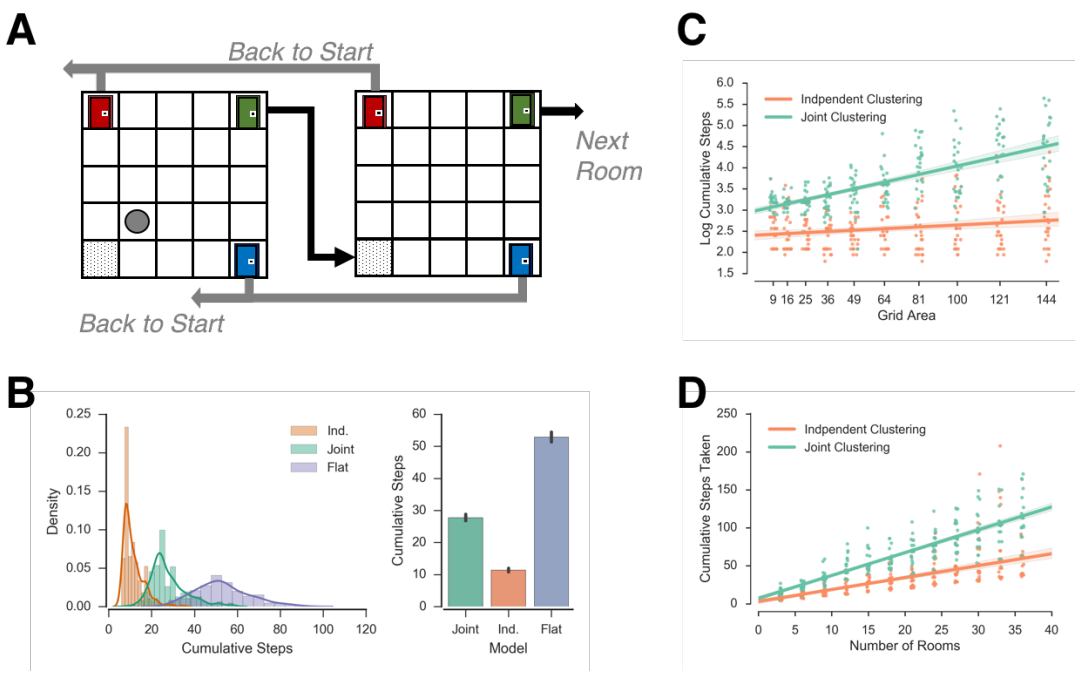

Figure 2.4: *A*: Schamatic diagram of rooms problem. Agents enter a room and choose a door to navigate to the next room. Choosing the correct door (green) leads to the next room while choosing the other two doors leads to the start of the task. *B*: Distribution of steps taken to solve the task by the three agents (left) and median of the distributions (right). *C,D*: Regression of the number of steps to complete the task as a function of grid area (C) and the number of rooms in the task (D) for the joint and independent clustering agents.

new context. Consequently, a good measure of generalization is the reduction in exploration cost. In the above simulations, exploration costs were uniform across all contexts. Here, we consider a set of task domains in which each context has a different exploration cost and that cost of exploration increases across time. In real world situations, the cost of exploring in a domain with multiple subgoals can compound as a person progresses through a task. For example, if a person is baking a cake and uses vinegar instead of vanilla, they may have to start over from scratch. Thus, the benefits and costs of generalization can compound in task with a sequential structure in ways that are not often apparent in a more restricted task domain.

We define a modified 'rooms' problem as a task domain in which an agent has to navigate a series of rooms to reach a goal location in the last room [Fig 2.4, A]. In

each room, the agent must choose one of three doors, one of which will advance the agent to the next room, whereas the other two doors will return the agent to the start of the task. However, there are multiple mappings that could apply and the agent has to discover which actions to select to make the cardinal movements separately from the location of the reward (goal door). All of the rooms are visited in order such that if an agent chooses a door that returns it to the start of the task, it will need to visit each room before it can explore a new door. Consequently, the cost of exploring a door in a new room increases for each newly encountered room.

Botvinick et al (Botvinick et al., 2009) have previously the 'rooms' problem introduced by Sutton et al. (1999) to motivate the of options (hierarchical RL) as a method of reducing computational steps. However, in the traditional options framework, there is no method of reusing options across different parts of the state-space (for example, from room to room). Each option needs to be independently defined for the portion of the state-space it covers. In this variant of the rooms problem, we have afforded the opportunity for reuse of subgoals across rooms, but have modified the task such that there is a large cost to when a learned subgoal (choosing the correct door) is not reused. In addition, the rooms problem here is qualitatively similar to the "RAM combination-lock environments" used by Leffler et al. 2007 to show that organizing states into classes with reusable properties (analogous to clusters presented in the present work) drastically can drastically reduce exploration costs. In the RAM combination-lock environments, agents navigated through a linear series of states, in which one action would take agents to the next state, another to a goal state and all others back to the start. The rooms task environment presented here is highly similar but allows us to vary the cost of exploring each room parametrically by varying its size.

We simulated an independent clustering agent, a joint clustering agent and a flat agent on series of rooms problems with the parameters $\alpha = 1.0$, $\gamma = 0.80$ and $\beta = 5.0$. Each room was represented as a new context (e.g. to simulate differences in surface features). There were three doors in the corners of the room, and the same door advanced the agent to the next room in every room (for simplicity, without loss of generality). Agents received a binary reward for selecting the door that advanced to

the next room.[1]

In the first set of simulations, we simulated the agents in 6 rooms, with each room comprising a 6x6 Gridworld, with three mappings $\phi_c \in \{\phi_1, \phi_2, \phi_3\}$, each repeated once.[2] In this task domain, both clustering agents show a generalization benefit as compared to the flat agent [Fig 2.4B], with the flat agent completing the task in approximately 2x and 4x total steps than joint and independent clustering, respectively.

We further explored how these exploration costs change as the task domain changes. In the simulations presented in figure 2.4C, we varied the dimensions of each room from 3x3 to 12x12. While both clustering models show increased exploration costs as the area of the Gridworld increases, the exploration costs for the joint clustering model grow at a faster exponential rate compared to the independent clustering model.

Similarly, we can increase the exploration costs by increasing the number of rooms in the task domain. In figure 2.4D, we varied the number of rooms in the rooms in the task domain from 3 to 27 in increments of three. As before, the same door in all rooms advanced the agent and three mappings were repeated across the rooms.[3] As before, both clustering agents experience an increased cost of exploration as the number of rooms increases, but the cost of exploration increases at a faster linear rate for the joint clustering agent than the independent clustering agent.

2.3 Normative Analysis

Thus far, we have examined the performance of agents based on variants of a non-parametric clustering algorithm using specific assumptions about how their choices are generated (e.g., planning and exploration, etc.). However, these agents were not constructed by inverting the generative process of the tasks that they confronted. As such, neither of the clustering agents is strictly optimal for any of the above of the

[1] i.e. $R_i(x) = R_j(x) \; \forall \; i, j$

[2] Because the cost of exploration grows as an agent progresses through a task, the ordering of the rooms affects the exploration costs. For simplicity, order of the mappings encountered by rooms can be defined by the sequence $X^{\phi} = \phi_1\phi_1\phi_2\phi_2\phi_3\phi_3$

[3] The order of the mappings encountered is defined by the sequence $X^{\phi} = \phi_1^{(k)}\phi_2^{(k)}\phi_3^{(k)}$ for $k \in \{1, 2, ..., 9\}$, where k is the number of times a mapping is repeated

tasks considered in isolation (and certainly not across the set of tasks, given that the ranked model performances reversed across them). Over a larger set of tasks, optimality can be a nebulous construct as it depends on the assumptions embedded in the model about the generative process, and any agent's performance will depend in part on whether those assumptions are congruent with reality. Indeed, we can expect that there exists an algorithm for each task that outperforms the models presented here. However, we are more concerned with the suitability of generalization across ecological environments, rather than the specific task domains we have simulated. To properly assess this question, one requires a definition and characterization of ecological environments, a problem well outside the scope of the current work.

Nevertheless, we can more formally and generally assess when, and under what conditions, each of the clustering models might be more suitable than the other. In the simulations thus far, we have imposed strong assumptions about the task domain, the agents' knowledge of its structure, exploration policies, and planning. While these assumptions are useful for the purpose of demonstration, they are less likely to be valid in real-world applications (such as a robot navigation) or as a model of human or animal learning. To make a more general normative claim, it is desirable to abstract away the implementation and strictly address the normative basis of context-popularity based clustering as a generalization algorithm by itself.

To do so, we can frame generalization as a classification problem and quantify how well an agent correctly identifies the cluster in which a context belongs without regard to learning the associated tasks statistics. This simplifying assumption allows us to examine the CRP as a generalization agent and abstracts away the effect of the likelihood function on generalization. Let $k \in \mathcal{K}$ be a cluster associated with a Markov decision problem and let context $c \in \mathcal{C}$ be a context experienced by the agent. Given a history of experienced contexts and associated clusters $\{c_{1:n}, k_{1:m}\}$, we cast the problem of generalization as learning the classification function $k = f(c)$ that minimizes the risk of misclassification. Misclassification risk here is loosely defined: some misclassification errors are worse than other and misidentifying a cluster (and consequently, its MDP) may or may not result in a poor policy if the underlying MDPs are sufficiently similar. As such, the loss function is ecological settings is a highly non-linear, domain specific function (as we demonstrate in section 2.2.3).

More formally, we define risk as the expectation $R(p, f) = \mathbb{E}[L(p, f)]$, where $L(p, f)$ is the loss function for misclassification and p is the generative distribution over new contexts $p = \Pr(k|c_{t+1})$. For our purposes here, we wish to abstract away domain specificity in the loss function $L(p, f)$. Because the CRP is a probability distribution, a reasonable domain-independent loss function is the information gain between the CRP's estimate of the probability of k and the realized outcome, or

$$L(p, f) = -\log_2 q_K \tag{2.12}$$

where q_M is the CRP's estimate of the probability of the observed cluster k in context c. The misclassification risk is thus the cross entropy between the CRP and the generative distribution:

$$R(p, f) = \mathbb{E}_p[-\log_2 q] = H(p, q) \tag{2.13}$$

Thus, by casting generalization as classification and assuming information gain as a domain-general loss function, we are in effect evaluating the degree to which the CRP estimates the generative distribution. Risk is minimized when $q = p$, that is, when the CRP perfectly estimates the generative process. There is no upper bound to poor performance, but in many task domains it is possible to make a naïve guess over the space of clusters (for example, a uniform distribution over a known set of clusters). Because any useful generalization model will be better than a naïve guess, we can evaluate whether the CRP will lead to lower information gain than a naïve estimate in different task domains. Thus, we can ask what properties of the task domain effect the quality of the estimate.

2.3.1 CRP performance as a function of task structure

One question we can ask is how well the CRP prior generalizes as a function of the underlying structure of the task domain. Intuitively, we might expect that a generalization agent should show a generalization benefit in highly structured task domains and show a less of a generalization benefit in unstructured domains. We can thus define a set of restricted task domains that vary in their structure. Specifically, we can vary the predictability of the generative process p and evaluate the agent as a function of this predictability.

Let $\mathcal{K} = \{A, B, C, D\}$ be the set of possible clusters in a task domain with probabilities $\mathcal{P} = \{p_A, p_B, p_C, p_D\}$. Let $X = ABCD$ represent a sequence of contexts and cluster identities experienced by an agent, such that cluster A is experienced in context c_1, B is experienced in context c_2, etc. We assume that the cluster identity is observable from the statistics of the associated MDP and that the agent knows the members of the set of $\mathcal{K}$.

We want to evaluate the ability of the CRP prior to predict each cluster in the sequence, conditional on its own history. We do so by calculating the expected loss experienced by the CRP over the sequence X. We define our loss function over sequence X as

$$L(p, f(X)) = -\frac{1}{||X||} \sum_{X} \log_2 q_k^{(t)} \tag{2.14}$$

where $q_k^{(t)}$ is the CRP's probability estimate for the value of X_t given $X_{1:t-1}$. As noted above, this loss function is equivalent to the cross entropy between the CRP and the generative process $H(q, p)$. That is, $H(q, p)$ is the average degree of unpredictability (in bits of information content) of experiencing each MDP given the estimate q. We can assess $H(q, p)$ for the CRP by updating the CRP in each context and probing its estimate for the subsequent context. As the CRP is exchangeable (Aldous, 1985), $H(q, p)$ is invariant to the order of the sequence.

We can quantify the degree of predictability of X similarly by evaluating the entropy of the sequence, defined $H(X) = -\sum_{\mathcal{P}} p_x \log_2 p_x$. Here, we define a sequence $X^{(n)}$ to allow us to monotonically decrease $H(X)$ with n. Let $X^{(n)}$ be the sequence $A^{(n)}BCD$, where n denotes the number of times A appears in the sequence. For example, $X^{(1)}$ would be the sequence $ABCD$ and $X^{(3)}$ would be the sequence $AAABCD$. For simplicity, we assume the probability distribution $\mathcal{P}$ over the ensemble $\mathcal{K}$ is exchangeable and that the probabilities over the members of its ensemble are proportional to their frequency in the sequence such that $p_k = N_k/||X||$ where N_k is the number of times k appears in sequence X. Consequently, the entropy of the sequence $X^{(1)}$ is $H(X^{(1)}) = 2$ bits and the entropy of the sequence $X^{(3}$ is $H(X^{(3)}) \approx 1.79$bits. As n approaches infinity, the entropy $H(X^{(n)})$ asymptotically approaches 0 bits. Intuitively, as A is repeated more often in the sequence, the sequence is more predictable (lower entropy)[4].

[4]It is important to note that the sequence predictability does not depend on order. Because the

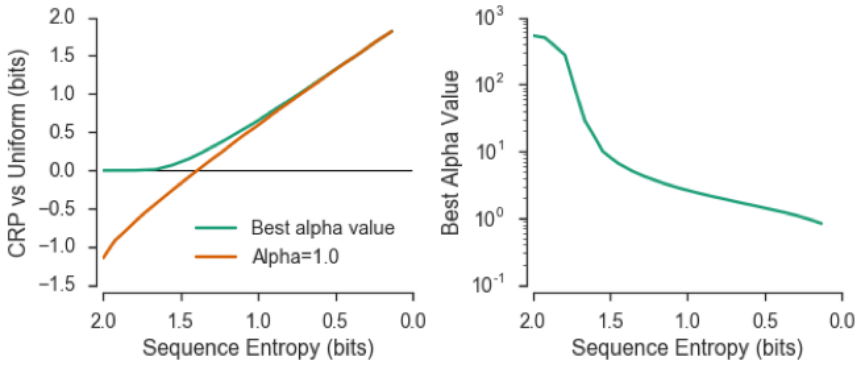

Figure 2.5: Performance of the CRP as a function of task domain structure. *Left*: Relative information gain of a naïve guess over the CRP as function of sequence entropy for a CRP with an optimized alpha parameter (green) or fixed at $\alpha = 1.0$. *Right*: Optimized alpha value (log scale) as a function of sequence entropy.

We evaluated the CRP on $X^{(n)}$ for $n = [1, 100]$ and compared it to a naïve guess (uniform distribution over $\mathcal{M}$). Because the CRP is parameterized by its tendency to generate a new cluster, the value of its α parameter alters the predictive distribution. To establish an upper limit on the performance of the CRP, we used numerical optimization to determine α for each value of n. In addition, we also evaluated the performance of an agent with a fixed $\alpha = 1$, which we believe is a more accurate reflection of a generalizer in an unknown environment. As expected, as we increase the structure of the sequences (lower entropy), the CRP advantage over a naïve guess increases [Fig 2.5, left, green line]. Similarly, the optimal value of α declines with the sequence structure, such that it is more advantageous to cluster as the sequence becomes more predictable [Fig 2.5, right]. However, this benefit is minimal for very unstructured sequences, and for fixed values of α, clustering for highly unstructured sequences ($H(X) \lesssim 1.45$bits) yields worse CRP performance compared to a naïve guess.

CRP is exchangeable, it will have the same predictive error for the sequences ABCDABCD and ADBCDBAC. Order-dependent predictability is beyond of the scope of the current work.

2.3.2 Independent vs Joint clustering

Having confirmed these intuitive properties of the CRP, we can now ask under what conditions is it better to cluster aspects of the task structure (such as reward functions and transition or mapping functions) independently or jointly. To do so, we rely on the assumption that a reward function is typically substantially sparser than a transition function: as an agent interacts with a task it will gain more information about the transition function early in the task than it will about the reward function (unless the environment is very rich with rewards at most locations, in which case any random agent would perform well).

Consequently, for an agent that clusters rewards and transitions together, the information gained about transitions will dominate the likelihood function, such that the inference of rewards can be thought of as approximately conditional on knowledge of the transition structure. Conversely, an agent that clusters rewards and transition independently will not consider the mapping information in their generalization of rewards. Thus, we can compare the two agents by evaluating the consequences of clustering rewards conditional on transitions as compared to clustering rewards independent of transitions. Formally, we can consider the comparison of independent and joint clustering as the comparison between two different classifiers, $R = f(c)$ and $R = f(c, T)$, one of which classifies reward functions solely as a function of contexts and the other as a function of contexts and observed transition statistics.

Interestingly, this approximation leads to the conclusion that joint clustering produces better generalization when the true structure of the task domain is known. Intuitively, this claim is based on the notion that more information is always better (or at least, no worse). More formally, given random variables R and T, $H(R|T) \leq H(R)$. This statement implies that given a known joint distribution between two random variables, knowledge of one of the random variables cannot increase the uncertainty of the other (and an agent can simply learn when there is no relation between the two, in which case the joint distribution doesn't hurt). Note that this relationship is only guaranteed to be true if the generative distribution is known, and experience in the task domain plays an important role (i.e. there needs to be sufficient experience to determine whether the joint distribution is useful or not; see next section). If we consider the CRP to be an estimator, we can consider its bias

$\text{Bias}_p[q] = \mathbb{E}[q-p] = -\alpha/(\alpha+N_c)$, where N_c is the total number of contexts observed. Thus, $\lim_{N_c\to\infty}\text{Bias}_p[q] = 0$ and the CRP will converge to the generative distribution asymptotically. As a consequence, joint clustering will have lower information gain than independent clustering asymptotically as the CRP for a joint clustering agent will converge to the conditional distribution $\Pr(R|T)$ whereas the CRP for an independent clustering agent will converge to the marginal distribution $\Pr(R)$.

Mutual Information

As alluded to above, joint clustering is guaranteed to produce a better estimate only true as $n \to \infty$ whereas here we are concerned with task domains in which an agent has little experience. Intuitively, we might independent clustering to be favorable in task domains where there is no relationship between transitions and rewards. Conversely, we might expect Joint clustering to be more favorable when there is relationship between transition and rewards. Formally, we can consider the relationship between transitions and rewards with mutual information. Let each context c be associated with a reward function $R \in \mathcal{R}$ and a transition function $T \in \mathcal{T}$ and let $\mathcal{P} = \Pr(R,T)$ be the joint distribution of R and T in unobserved contexts. The mutual information between R and T is defined

$$I(R;T) = \sum_{R\in\mathcal{R},T\in\mathcal{T}} \Pr(R,T)\log_2\frac{\Pr(R,T)}{\Pr(R)\Pr(T)} \tag{2.15}$$

Mutual information represents the degree to which knowledge of either variable reduces the uncertainty (entropy) of the other, and can be used as a metric of the degree to which a task environment should be considered independent or not. As such, it satisfies $0 \le I(R;T) \le \min(H(R), H(T))$

To evaluate how mutual information affects the relative performance of independent and joint clustering, we constructed a series of task domains that vary in $I(R;T)$ by manipulating parametrically a single parameter m while holding all else constant. We define $\mathcal{R} = \{A, B\}$ and $\mathcal{T} = \{1, 2\}$ and we define two sequences X^R and X^T such that X^R_i and X^T_i are the reward and transition function for context c_i. We define the sequence $X^R = A^{(2n)}B^{(2n)}$, where $A^{(k)}$ refers to a k repeats of A, and the sequence $X^T = 1^{(n+m)}2^{(2n)}1^{(n-m)}$, where $1^{(k)}$ refers to k repeats of 1. To provide a

concrete example, if $n = 2$ and $m = 0$, the sequence $X^R = AAAABBBB$ and the sequence $X^T = AABBBBAA$. Similarity, if $n = 2$ and $m = 2$, than the sequence X^R is unchanged while X^T is now $AAAABBBB$. Critically, these sequences have the property that $I(R;T) = 0$ if and only if $m = 0$ and that $I(R:T)$ monotonically increases with m. In other words, the residual uncertainty of R given T, $H(R|T)$, declines as a function of m. This allows us to vary $I(R;T)$ independently of all other factors by changing the value of m.

We evaluated the relative performance of the independent and joint clustering agents by using the CRP to predict the sequence X^R, either independent of X^T (modeling independent clustering), or conditionally dependent on X^T (modeling joint clustering) for values of $n = 5$ and $m = [0, 5]$. For these simulations, we first assume both sequences X^R and X^T are noiseless, but below we show that noise parametrically affects these conclusions. For low values of m ($m \leq 2$), independent clustering provides a larger generalization benefit [Fig 2.6, left]. This has the intuitive explanation that as the task domain becomes more compositional, independent clustering provides better generalization. It is important to note that in the considered task domains, independent clustering is better for cases where there is non-zero mutual information $[0 \leq I(R;T) \lesssim 0.2\text{bits}]$.

Noise in observation of transition functions

Above, we assumed that transition functions were fully observable with no uncertainty. But in many real-world scenarios, the relevant state variables are only partially observable and the transition functions may be stochastic. In this section, we therefore relax this noiseless assumption to characterize the effect noisy observations have on inference and generalization. As before, we model generalization as the degree of predictability of X^R_{i+1} given $X^R_{1:i}$ either independent of X^T (independent clustering) or conditionally on X^T (joint clustering).

We first construct reward and transition sequences in which knowledge of the transition function completely reduces the uncertainty about the reward function ($I(R;T) = H(R)$, and hence $H(R|T) = 0$). Consider $X^R = A^{(20)}BCD$ and $X^T = 1^{(20)}234$, where $A^{(20)}$ and $1^{(20)}$ refer to 20 repeats of A and 1, respectively. As such, we would expect joint clustering to produce better generalization than independent

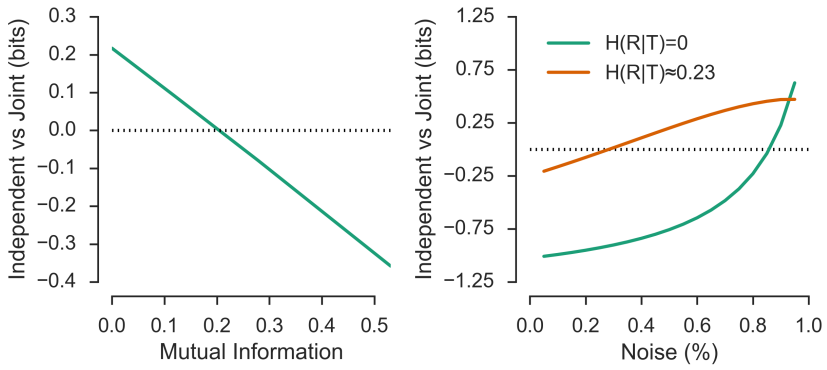

Figure 2.6: Performance of independent vs. joint clustering in predicting a sequence X^R, measured in bits of information gained by observation of each item. *Left*: Relative performance of independent clustering over joint clustering as a function of mutual information in reward and transition functions. *Right*: Noise in observation of X^T sequences parametrically increases advantage for independent clustering. Green line shows relative performance in sequences with no residual uncertainty in R given T (perfect mutual information), orange line shows relative performance for a sequence with residual uncertainty $H(R|T) > 0$bits.

clustering if observations are noiseless. To simulate noise / partial observability, we assume each observation of X_i^T is mis-identified as some new function $T^* \notin \{1, 2, 3, 4\}$ with some probability. Importantly, this simulation has the desiderata that noise does not affect $I(R;T)$ or $H(R)$ themselves (the true generative functions). We compare the inference of X^R independent of X^T (modeling independent clustering) or conditionally dependent on X^T (modeling joint clustering) for various noise levels $\sigma = [0,\ 1.0]$. When noise is sufficiently high ($\sigma > 0.75$), independent clustering produces a better estimate of X^R than joint clustering [Fig 2.6, right, green line] even for this extreme case where the two functions have perfect mutual information.

Next, we assessed how mutual information and noise interact by decreasing the correspondence of the sequences. As noted above, in the sequences used above ($X^R = A^{(20)}BCD$ and $X^T = 1^{(20)}234$), there is no residual uncertainty of R given T, $H(R|T) = 0$. By appending the sequences such that $X^R = A^{(20)}BCDBCD$ and $X^T = 1^{(20)}234342$, we can increase the residual uncertainty of R given T above zero (specifically, to $H(R|T) = 0.23$). As we have previously noted, this increase in $H(R|T)$ reduces the benefit of joint clustering, all else equal. Simulating independent

and joint clustering on these new sequences as a function of $\sigma = [0,\ 1.0]$ reveals a lower level of noise needed to see a benefit of independent clustering (i.e. here independent clustering produces a better estimate of R for $\sigma > 0.3$).

2.4 Discussion

In this paper, we provide two alternative models of context-based clustering for the purpose of generalization, a *joint clustering* agent generalizes reward and transition functions together and an *independent clustering* agent that separately generalizes reward and transition functions. These models are motivated by human learning, which we argue is compositional to a degree that previous models of generalization are unable to capture.

One view of generalization is a solution to a dimensionality problem. In real-world problems, perceptual space is typically high-dimensional. In order to learn a policy, agents need to learn a mapping between the high-dimensional perceptual space and the effector space. Learning this mapping can require a large set of training data, perhaps much larger than a human would have access to (Botvinick et al., 2009; Lake et al., 2016). Generalization can be seen as a solution to this dimensionality problem by projecting the perceptual space onto a lower dimensional latent space in which multiple precepts share the same latent representation. Thus, an agent does not need to learn a policy over the full state space but over the lower dimensional latent space. This is an explicit assumption of the models presented here as well as other clustering models of human generalization (Gershman et al., 2010; Collins and Frank, 2013) and lifelong learning (Wilson et al., 2012; Mahmud et al., 2013; Rosman et al., 2016). A similar logic can be applied to both hierarchical approaches as well as object and symbol based approaches (Diuk et al., 2000; Konidaris and Barto, 2007; Solway et al., 2014; Kansky et al., 2017; Konidaris, 2016).

Compositionality takes this argument further. Instead of learning at the policy level, learning at the component level decomposes a high-dimensional problem into multiple lower-dimensional problems. For example, if areas of the state space share features such as action-outcome relationships, then this feature can be shared across the space (Leffler et al., 2007). If component features are reused in combinations,

then estimating components independently and binding them to a representation in latent space further reduces the learning problem. When an agent enters a new environment, components may be generalized as observations support. Furthermore, while the choice of an appropriate set of task components is open-ended, here we argue that Markov decision problems provide a natural decomposition. Reward functions and transition functions provide separate information about a task and thus, are good candidates for components of generalization.

Nonetheless, learning separate components does not take into consideration the correlation structure between components. In the present work, we ask what are the effects of assuming a strong vs a weak relationship between components as a function of the strength of the relationship? Both joint and independent clustering show meaningful generalization benefits for mappings (Figures 2.1.3 & 2.2.1) but nonetheless realize very different benefits or costs overall as a consequence of goal generalization. With sufficient experience, we typically expect joint clustering to perform better in new contexts as assuming a potential relationship between goals and mappings can be no worse asymptotically than assuming independence. Nonetheless, this is not always what we observe. In the rooms problem above, joint clustering performs much worse than independent clustering, even as it may similarly benefit from the task structure.

While we have shown that noise in the observations, low mutual information and a non-linear cost function can result in strong benefits for independent over joint clustering, this can be puzzling given the asymptotic assurances that joint clustering will be no worse. Here, the comparison between classification and generalization is instructive. We can think of joint clustering as similar to estimating a joint distribution of the generative process and independent clustering as estimating the marginal distribution for each component independently (similar to a naïve Bayes classifier). In this interpretation, independent clustering trades an asymptoticly worse estimate of the generative process for lower variance, with a bias a equal to the mutual information between mappings and goals.[5] In problems with limited experience, such as the type presented here, a biased classifier will often perform better than an asymptotically more accurate estimator because misclassification risk is more sensitive to

[5]As noted previously, the bias of the CRP asymptotically approaches zero as $n \to \infty$ and is ignored here

variance than bias (Friedman, 1997). Thus, by ignoring the correlation structure and increasing the bias to generalize goals that are most popular overall, independent clustering may minimize its overall loss.

We can think of joint clustering as being potentially overly sensitive to noise in the correlation structure and it may be better to ignore real correlations in the environment in order to be robust to this source of noise. To provide a concrete example, we can imagine that if a person is getting food from a buffet, how hungry they are may inform the actions needed to carry the food back to the table. If they are hungry, they may carry more food on a plate and need a subtly different set actions. Nonetheless, the generalization benefit by assuming this relationship may be trivial in comparison to the cost needed to learn the association, especially in other environments where the risk of poor generalization is severe.

An altogether different possibility is that a mix of strategies is appropriate. It is not known what a human learner would consider a context, contexts are typically cued to subjects artificially, but the number of contexts a human could potentially encounter ecologically is quite high. In which case, they would be able to form a more accurate estimate of the correlation structure between components over time. If this speculation is true, then one potential adaptive strategy would be to assume a weak relationship between components early in learning and increasingly relying on the correlation structure as the evidence supports it. Furthermore, high dimensional representations can be useful for decision making (Fusi et al., 2016) as well as binding operations (Smolensky, 1990). This suggests a tension between low-dimensional representations, which are easier to learn and generalize, and high-dimensional representations, which may be more flexible for complex thoughts and behavior.

Finally, while we have presented independent clustering as motivated by human generalization, the question of whether human learning is better accounted for by independent or joint clustering remains an open question. Both models are a generalization of previous models used to account for human behavior (Gershman et al., 2010; Collins and Frank, 2013, 2016b). In the following chapter, we examine the predictions of these two models in a series of behavioral tasks and compare these predictions to human subject behavior.

Chapter 3

Empirical study of compositionally in human structure learning

3.1 Introduction

It has long been proposed that humans are able to combine component pieces of knowledge together beyond simply re-using past associations (James, 1890). A young musician who has learned to play the guitar, for example, can more easily learn the banjo, as the skills learned in one instrument are transferable to the other. In principal, the skills needed to manipulate a fretted instrument are independent of the song being played; they are reusable components that can be combined compositionally to create an arbitrary number of songs (policies). In contrast, computational agents typically lack this degree of compositionality. Instead, they are restricted to re-using previously learned policies exactly as learned. When a computational agent generalizes at all, it typically does so rigidly, either re-using a past policy as a whole or learning a novel policy from scratch (Rosman et al., 2016; Mahmud et al., 2013; Wilson et al., 2012). A young musician following a similar strategy would be able to generalize the same pattern of hand motions to play the same song from an electric guitar to an acoustic guitar but wouldn't be able to generalize any information at all to a banjo, as a banjo has a different number of strings. Consequently, models of generalization without compositionality lack an important component of flexible human behavior.

The distinction posed by whether or not a behavior is best described as compositional has long been the source of debate in cognitive science. Early models of cognition were often symbolic processing models that made the strong assumption of compositional representation (Newell and Simon, 1959). In contrast, later connectionist models and their descendants, modern neural nets, largely depend a form of associative processing the deliberately eschews symbol manipulation (Rumelhart and McClelland, 1985; Elman, 1990; LeCun et al., 2015). While powerful, neural nets often struggle on problems with sparse rewards and a high degree of latent structure – problems that seem to favor the use of a compositional structure (Lake et al., 2016; Kansky et al., 2017).

Reinforcement learning (RL) models of human and animal learning are associative models of learning and fall into the later tradition, assuming representations inconsistent with the degree of compositionality we expect to find in human behavior. In simple RL models, action-values are updated through a process of feedback-based learning (Sutton and Barto, 1998; Montague et al., 1996). These models are typically used to describe very simple tasks, such as learning to take a single action in return for reward, but nonetheless have shown substantial predictive validity describing behavior and neurobiological functioning in humans and animals (Tsai et al., 2009; Kravitz et al., 2012; Shen et al., 2008; McClure et al., 2003b; O'Doherty et al., 2003; Frank et al., 2004).

Although simple, there is reason to believe that the neurobiological mechanisms that underlie this type of feedback based learning may support a more complex and goal-directed learning system involved in volitional action (Balleine et al., 2008; Daw et al., 2011; Daw, 2012; Kool et al., 2016). Theoretical RL models can leverage control over working memory to solve multiple task (Rougier et al., 2005; Todd et al., 2008), as a mechanism for variable binding (Kriete et al., 2013) and to generalize from one task to another (Collins and Frank, 2013). The generalization an RL agent can perform can be described at a computational level with clustering algorithms that partition contexts based on their statistics, in effect treating contexts clustered together as the same to facilitate the transfer of a previously learned policy from one context to the next (Gershman et al., 2010; Collins and Frank, 2016b).

Empirically, humans tent to build latent structures consistent with these clustering

models of generalization in a process that depends on fronto-striatal circuitry (Collins and Frank, 2013; Werchan et al., 2016; Collins and Frank, 2016b; Badre et al., 2010). Although promising, clustering models of generalization nonetheless lack the compositional structure needed to generate novel policies. Like other artificial agents, they are limited to either the reuse an old policy or learning a new one from scratch (Rosman et al., 2016; Mahmud et al., 2013; Wilson et al., 2012). In James' terms, these models are more "reproductive" than "productive" (1890). This poses a potential limitation of clustering models as a description of fronto-striatal rule-like behavior.

Here, we address this limitation by comparing model predictions of a clustering agent extended compositionally to human subject behavior across three navigation tasks. In the navigation tasks, subjects are able to learn goals ("where do you want to go?") independently of a world model ("how can you get there?") but need both pieces of information to solve the task. We examine the predictions of a model that clusters these two task components independently, as well as the predictions of a control model that jointly clusters goals and world models.

In Chapter 2 we described these two computational agents, both of which generalize by popularity based-context clustering. The first model clusters the statistics of the goals and world models independently, while the second model jointly clusters the two. While both of these agents are a generalization of the model proposed by Collins and Frank (2013), only the independent clustering agent is compositional. Independently clustering provides the agent with the ability to combine two functions learned in separate contexts in a novel context to create a novel plan. We show that subjects display the degree of compositionality predicted by the independent clustering model. More over, we show that subjects are sensitive to the task environment and are able to modify their strategy when it is adaptive to do so.

3.2 Separable Predictions of Independent and Joint clustering

In the generalization models we consider here, contexts are clustered based on their popularity and by the degree to which the statistics of a cluster predict the observations in any associated context. The intuition is that contexts that share statistics

will produces similar observations, and can thus be clustered together. We rely on the *Chinese Restaurant Process* (CRP) (Aldous, 1985) as a prior over the assignment of contexts into clusters (see methods). This prior is an unbounded generative process that allows for as many clusters as there are contexts, but nonetheless biases the number of clusters towards parsimonious clusters.

Both independent and joint clustering used popularity to generalize but differ in how they link information about goals and world models. Here, we operationalize goals as a rewarded function over labeled locations in a navigation task and we operationalize a world model with a mapping function that defines the relationship between button responses and movement in the task. Independent clustering assumes the inference of mappings and goals are not mutually informative. In contrast, joint clustering will use information of one to infer the other. During exploration of a new context, an agent will generally observe mapping information prior to observing goal information. As a consequence, the information flow largely goes in one direction: observations about the mappings can inform the inference of a goal, but an agent will have typically already observed the correct mapping by the time it reaches a goal.

Therefore, we can distinguish the predictions of the two models on a qualitative basis by looking at goal generalization to see whether it varies as a function of mapping (See section 2.3.2 for a normative analysis). The three experimental paradigms presented here exploit this logic. In each of the three experiments, subjects learning the statistics of a set of "training" contexts. Each of these training contexts are associated with one of two mappings (a "high popularity" and "low popularity" mapping). Each context is also associated with a single rewarded goal, and the popularity of goals across contexts differs depending on whether mapping is considered, such that the popularity of goals across all contexts with associated with the high popularity mapping is different than the popularity of goals across contexts associate with the low popularity contexts.

These training contexts provided an expectation we can test in novel contexts. Both models predict goals with high popularity across contexts are more likely to be generalized (i.e. explored more quickly) than goals with low popularity. By choosing novel "test" contexts with different mappings, we probe these expectations, looking for evidence of generalization benefits and costs consistent that differentiate the

predictions of the two models.

3.3 Experiment 1

Experimental Design

Participants performed an interactive task in which they navigated a 6x6 Gridworld and selected one of three labeled goal locations to receive reward. Each trial was a new instance of a Gridworld, in which an agent the subject controlled was randomly placed on the board [Figure 3.1, left]. The agent, marked by a colored circle, could be moved in four possible cardinal directions (North, South, East and West). Also on the board were three colored-squares, "goal" locations. labeled A, B, and C. These locations shared the same color as the agent and the subjects were instructed that navigating the agent into any goal location would end the trial. Subjects were also instructed that choosing the correct goal location for a given trial would win them points whereas choosing either of the other two goal locations would win them no points. To incentivize subjects to perform well, subjects were given a bonus payment at the rate of $0.01 per correct goal choice.

Subjects navigated the Gridworld using eight keys on a standard keyboard ('a', 's', 'd', 'f', 'j', 'k' , 'l', and ';'), four of which deterministically corresponded to a cardinal movement. Subjects were required to learn the relationship between responses and cardinal movements (a "mapping") from experience [Figure 3.1, right]. There were two mappings used for the duration of the experiment and all of the valid responses for a single mapping were grouped on either the left or right hand. Invalid responses and errant keypresses led to no movement and were cued to the subject with an animation. Goals locations were distributed randomly on the map in each trial to control for low-level response biases and required the subjects to generate a novel navigation for each trial. In addition, a randomly generated pattern of walls was placed on the map that prevented the agent's movement to further encourage planning.

On each trial, a context was cued to the subject by the color of the agent and goal location. Every trial in a context shared the same color. Subjects were instructed that the keys they used to move the agent as well as the identity of the rewarded goal were contingent on the color of the agent. Subjects were required to learn the association

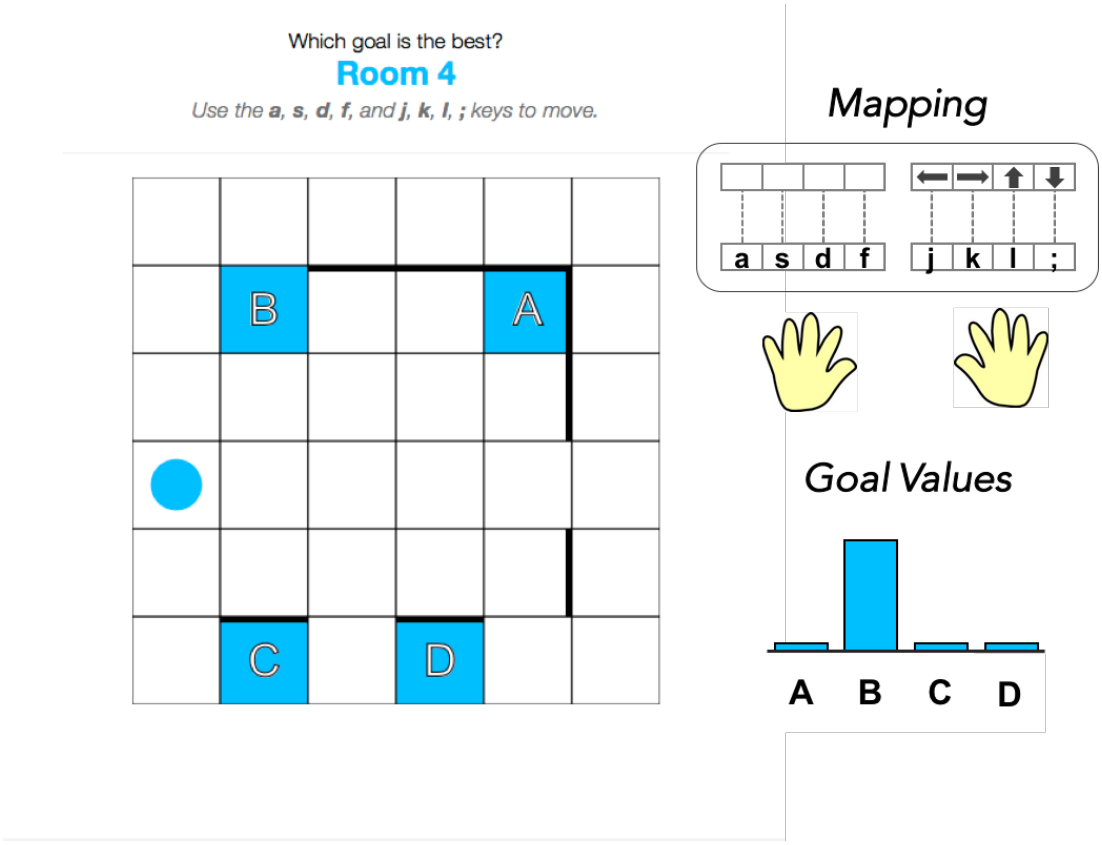

Figure 3.1: Interactive, online task. Subjects controlled a circle agent in a Gridworld (left) and navigate to one of four goals, each a colored square labeled "A", "B", "C" or "D". Context was signaled to the subject with a shared color for the agent and goals. In each context, subjects learned a mapping between the responses of one hand and the cardinal movement within the Gridworld (top right) as well as the identity of the rewarded goal within the trial.

between the contexts, mappings and goals from experience. Prior to the start of the experiment, subjects completely a short tutorial and were required to answer questions about the rule structure correctly before they were allowed to begin.

The five training contexts were chosen such that the popularity of goals differed across contexts with the high and low popularity mappings [Figure 3.2; Table 3.1]. The high popularity mapping was associated with goals A and B in remaining contexts three contexts, with goal A as the correct goal in two of these contexts. The low popularity mapping was paired with goals A and C in two contexts. Thus, goal A had the highest popularity across of all of the contexts, but its popularity was equivalent

to goal C when only contexts with the low-popularity mapping are considered. Each training context was repeated in at least 10 trials and the trials were balanced such that each mapping and each goal was seen in the same number of trials [Table 3.1]. Subjects then saw four novel test contexts (discussed below) immediately following the training contexts. The identity of the goal was randomized for each subject (e.g. subjects may see goal A, B or C as the high popularity goal), as was the relationship between keys-presses and cardinal movements that defined both mappings.

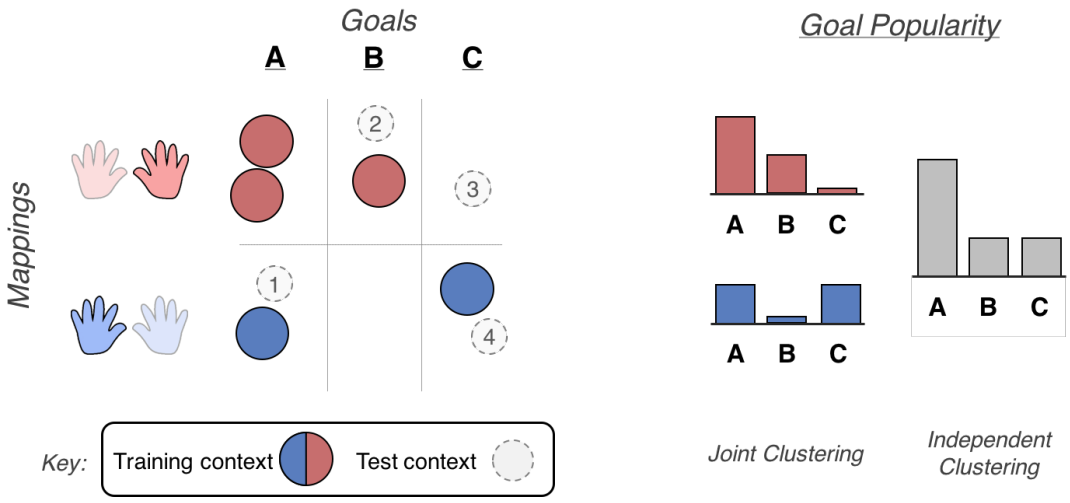

Figure 3.2: Experiment 1: Task design. *Left*: Subjects saw five contexts during training (red/blue circles), each paired with one of two mappings (*red*: high popularity, *blue*: low popularity) and one of three goals (A, B, or C). Subjects saw an additional four contexts following training (grey numbered circles). *Right*: Goal popularity both as a function of mapping (left) and collapsed across all contexts (right). Joint clustering makes its inference of goals as function of mapping while independent clustering is sensitive to goal popularity across all contexts.

Model Predictions

The four test contexts were selected to distinguish independent and joint clustering through the speed of learning correct goal. The null hypothesis is that there would be no differences between the rate at which each test context is learned, reflecting an absence of popularity-based context clustering.

250 iterations of a joint and independent clustering agent were trained on random

Context	Goal	Mapping Popularity	n Trials
Train 1	A	High	10
Train 2	A	High	10
Train 3	A	Low	20
Train 4	B	High	40
Train 5	C	Low	40
Test 1	A	Low	10
Test 2	B	High	10
Test 3	C	High	5
Test 4	C	Low	5

Table 3.1: Experiment 1 Task design. The number of trials within each context is balanced such that each goal and each mapping is presented the same number of trials across both training and test.

initializations of the behavioral task. There are two free parameters in the model, an inverse-temperature parameter β that controls stochastic exploration over cardinal movements and the concentration parameter α that controls the propensity of the model to generate a new cluster. Of the two parameters, only α effects generalization and was drawn from a standard log-normal distribution while β was set to 2.0.

The main predictions of the clustering model are a difference in accuracy in the test contexts as a function of time [Figure 3.3A]. The independent clustering will generalize the highest popularity goal regardless of mapping and predicts the goal A is more likely to be chosen in a new context than goals B or C. As a consequence, independent clustering makes the unique prediction test context 1 to be learned to be learned more quickly than test contexts 2 and 4, [Figure 3.3A,C], as test context 1 is the only test context associated with the high popularity goal. Additionally, independent clustering predicts above chance performance (positive transfer) in the first trial in test context 1 and a below chance performance (negative transfer) in the first trial on in contexts 2, 3 and 4. Independent clustering agent predicts no difference in learning contexts 2, 3 and 4.

In contrast, the joint clustering will generalize goals taking into consideration the mapping observations it makes. Test contexts 2 and 3 are associated with the high popularity mapping while test context 1 and 4 are associated with the low

popularity mapping. Because goal C was paired with the low popularity mapping in the training contexts, the joint model predicts that test context 4 (goal C, low popularity mapping) will be learned the most quickly[1] and test context 3 (goal C, high popularity mapping) will be learned the most slowly. The unique predictions of joint clustering can be summarized by the predictions that the accuracy test context 3 will be less than the accuracy in test contexts 2 and 4 and, to a lesser degree, that accuracy in test context 2 will be less than in test context 4 [Figure 3.3D]. These predictions are accompanied by a further prediction of positive transfer in the first trial in test context 4 [Figure 3.3C].

In addition, we simulated a flat agent that does not cluster contexts for the purposes of a normative comparison. This model is a special case of both the independent and joint agents that assigns each context to its own cluster. The flat model does not predict any differences in the learning speed between contexts. For generalization to be normatively advantageous on a specific task, it needs to outperform an agent that does not generalize. In this task, there is no generalization benefit in either the training or test conditions. A flat agent that does not generalize between contexts receives greater rewards on average than either clustering agents [Figure 3.4] across both training ($r \sim \text{Binom}(\theta)$; $\theta_{flat} = 0.944$, $\theta_{ind} = 0.920$, $\theta_{joint} = 0.934$, $\theta_{flat} - \theta_{ind}$ 95% HDI=[0.017, 0.025], $\theta_{flat} - \theta_{joint}$ 95% HDI=[0.004, 0.012]) and test contexts ($\theta_{flat} = 0.829$, $\theta_{ind} = 0.802$, $\theta_{joint} = 0.811$, $\theta_{flat} - \theta_{ind}$ 95% HDI=[0.057, 0.109], $\theta_{flat} - \theta_{joint}$) 95% HDI=[0.035, 0.086]). As subject pay is tied to rewards in the task, we would expect subjects that generalize to earn less on average for doing so.

Participants

156 online participants were recruited as subjects through Amazon's Mechanical Turk platform. Seven participants reported taking notes and were excluded from analysis. An additional 30 subjects were identified with cluster analysis as having chance performance (see Appendix A for details), leaving 119 subjects carried through to

[1] This is partially a frequency effect predicted by the model. Although the agents do not explicitly generalize based on frequency, the likelihood function is estimated with experience and clusters visited more often will have a better likelihood estimate. As such, the joint model will have a better estimate of the mapping function for frequent clusters, and will consequently be more likely to guess the associated goal.

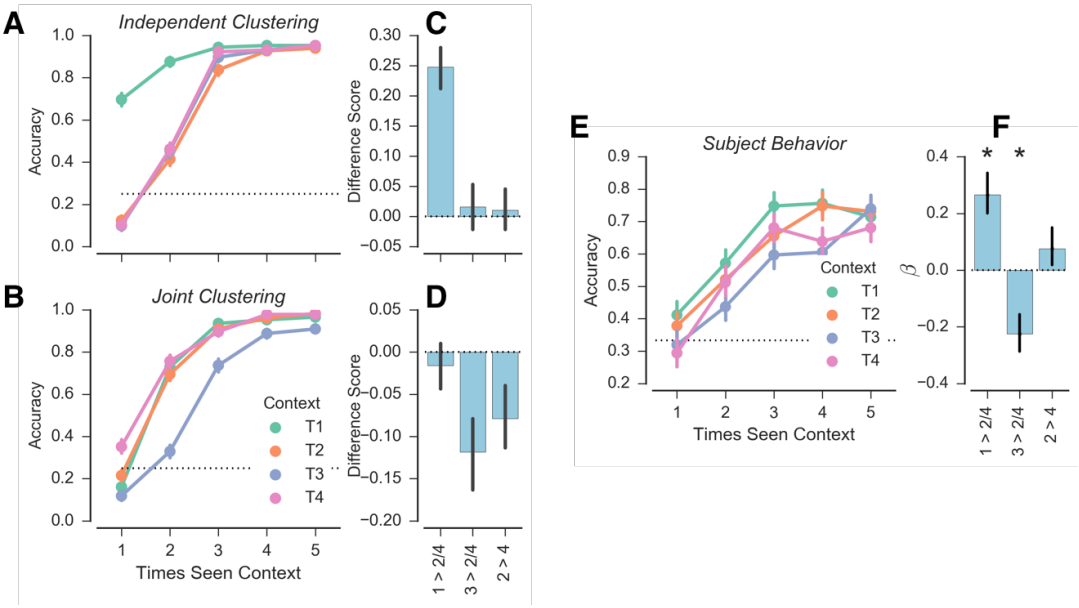

Figure 3.3: Experiment 1 Model predictions and behavioral results in test contexts. *A,B*: Accuracy of independent (A) and joint (B) clustering models of the four test contexts as a function of trials in the context. Chance accuracy is denoted by dotted line. *C,D*: Comparisons of accuracy in test contexts highlight differences between the model. Independent clustering predicts test context 1 will be learned more quickly than contexts 2 and 4 (C, left), while joint clustering predicts a difference in test accuracy for test context 3 as compared to 2 and 4 (D, middle) and a smaller difference between contexts 2 and 4 (D, right). *E*: Subject accuracy in test contexts across time. *F*: Regression coefficients comparing test contexts 1 vs 2/4, 3 vs 2/4 and 2 vs 4.

analysis.

Behavior

We first explore subject performance during training to verify subjects learned the task [Figure 3.5]. We assess accuracy, reaction time and navigation efficiency (the number of steps over and above the minimum path length needed to reach a goal) as performance measures using Bayesian linear modeling (see methods). The number of trials within a context was found to predict accuracy (β_t: 95% posterior highest density interval (HDI) = [0.080, 0.094]), suggesting subject learned the correct goals over time. Likewise, the number of times within a trial also predicted reaction time

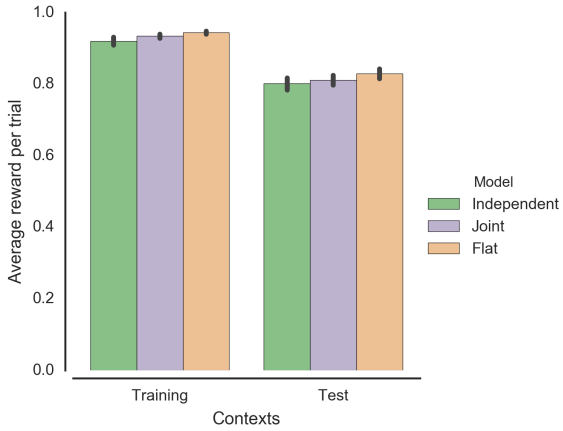

Figure 3.4: Experiment 1: Model performance on the task. A flat agent that does not cluster receives higher reward in both the training and test contexts than either the independent or joint clustering agents.

(β_t: 95% HDI = [0.38, 0.53]) and the number of steps greater than the minimum path length (β_t: 95% HDI = [-0.0095, -0.0008], suggesting that subjects responded more quickly and took more efficient paths over time.

The main predictions of the clustering agents were that subject accuracy would differ as a function of test contexts. We used Bayesian logistic regression analysis to assess these predictions. Three *a priori* comparisons of interest were defined in the regression, test context 1 vs test context 2 and 4, test context 3 vs 2 and 4, and test context 2 vs 4. Both models make directional predictions. Independent clustering predicts that test context 1 will be greater than test contexts 2 and 4 [Figure 3.3C]. Joint clustering predicts that the other two comparisons will be negative [Figure 3.3D]. Nuisance predictors for the number of trials in a context and whether a context was repeated sequentially were also included in the regression. A subject specific bias was included to account for individual differences, with each subject specific bias assumed to be normally distributed around a group average (Kruschke, 2015). A flat prior was assumed for each regression coefficient and a wide, half-Cauchy distribution was used as a prior over both regression noise as well as for the normal distribution over the subject specific offset (Gelman, 2006).

We found evidence for both joint and independent clustering. We first compared our regression model to a null model that did not include context contrasts using

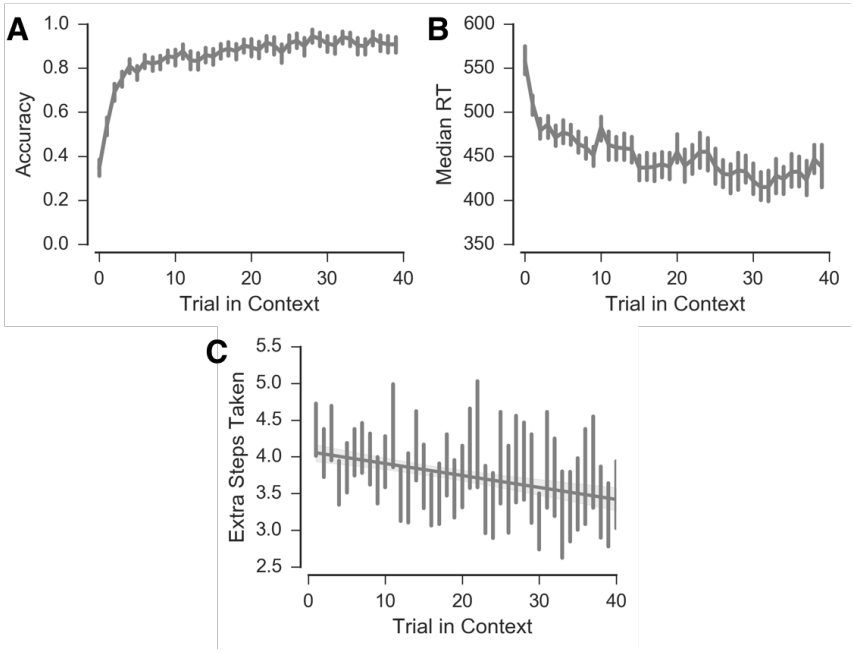

Figure 3.5: Experiment 1: Performance in training contexts. *A*: Subject accuracy as a function of trials within contexts. *B*: Median RT as a function of trials within context. *C*: Navigation efficiency, as measured by the number of responses over the minimum path length needed to reach the selected goal.

WAIC as a model selection measure (Watanabe, 2010) and found the presence of context contrast improved model fit (Model WAIC = 3036.3; null Model WAIC = 3046.1). Consistent with the prediction of the independent clustering agent, accuracy was higher in test context 1 than test contexts 2 and 4 ($\beta_{1>2/4}$: $\Pr(\beta_{1>2/4} < 0) = 0.99987$ 95% HDI $= [0.12, 0.41]$) [Figure 3.3F]. Consistent with the prediction of the joint clustering agent, accuracy was lower in test context 3 than test contexts 2 and 4 ($\beta_{1>2/4}$: $\Pr(\beta_{1>2/4} < 0) = 0.9992$, 95% HDI $= [-0.35, -0.08]$). We did not find evidence that accuracy in test contexts 2 and 4 differed ($\beta_{2>4}$: 95% HDI $= [-0.06, 0.21]$).

We further examined positive and negative transfer in the first trial of a test context. Independent clustering predicts positive transfer in test context 1 and negative transfer in the other test contexts. Joint clustering predicts positive transfer in test context 4. Consistent with independent clustering we found positive transfer in text context 1 ($y \sim \text{Binom}(\theta)$; $\Pr(\theta_1 > \frac{1}{3}) = 0.963$, 95% HDI=[0.325, 0.500]) but did not

find evidence of transfer in the other test context ($y \sim \text{Binom}(\theta)$; θ_2: 95% HDI=[0.29, 0.46]; θ_3: 95% HDI=[0.23, 0.40]; θ_4: 95% HDI=[0.21, 0.38]).

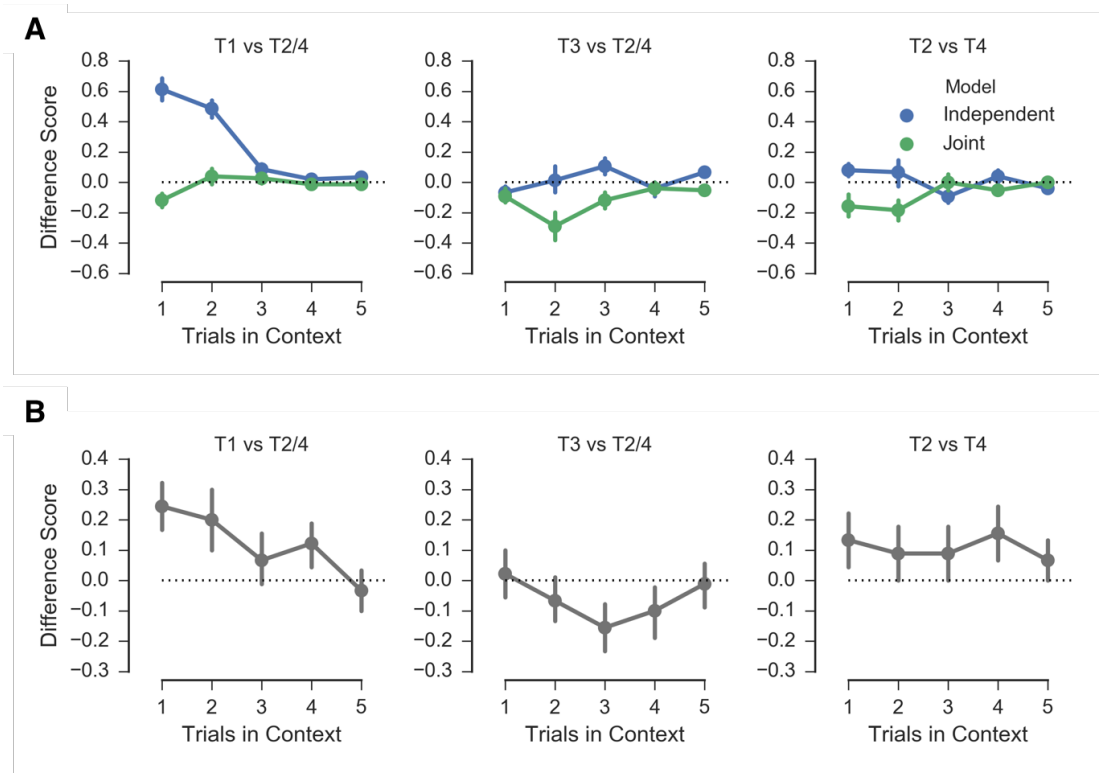

Figure 3.6: Experiment 1: Behavioral Results. Differences scores across time for the comparisons test context 1 vs 2 & 4 (left), 3 vs 2 & 4 (right), and 2 vs 4. *A*: Model predictions for both joint (green) and independent (blue) clustering. *B*: Subject behavior.

As a follow up analysis, we performed a similar Bayesian logistic regression on the accuracy of the training contexts as a function of goal. We included as predictors the identity of the correct goal (β_A and β_B for goals A and B, respectively with goal C used as a baseline) and included the same set of subject specific bias and nuisance parameters. Independent clustering predicts that accuracy in training contexts will be predicted strictly by popularity, such that contexts with goal A will be learned faster than contexts with goals B and C. Joint clustering predicts an interference due to the mappings, with training context 5 (goal C) to show the least degree of negative transfer. Surprisingly, we found improved learning in contexts with goal B than in contexts with goal A or C (β_B: 95% HDI = $[0.12, 0.51]$, $\beta_B - \beta A$: 95% HDI =

$[0.069, 0.406]$) and only week evidence contexts with goal A were learned faster than contexts with goal C ($\Pr(\beta_A > 0) < 0.84$; β_A: 95% HDI $= [-0.07, 0.24]$).

Discussion

In this experiment, we found evidence to suggest subject was consistent with both independent and joint clustering. A key prediction of both models is that accuracy would vary across test contexts, which is supported by model selection. Context contrasts support both independent and joint clustering, test context 1 was learned faster than contexts 2 and 5, the unique prediction of independent clustering, while context 3 was learned more slowly than 2 and 4, a unique prediction of independent clustering. In our data, test context 4 was not learned more quickly than test context 2, a unique prediction of the joint clustering agent.

A potential confound is the frequency of the test contexts. In order to balance the trials such that each goal and each mapping is shown in the same number of trials, test contexts 1 and 2 are shown twice as often as 3 and 4, and thus, twice as frequently due to the random order of presentation. Overall, accuracy was higher for test contexts 1 and 2 than test context 3 and 4 ($\Pr\left(\beta_{1>2/4} - \beta_{3>2/4} + \beta_{2/4} > 0\right) > 0.999$, 95% HDI=[0.28, 0.85]). One possibility is that the more frequent contexts have greater accuracy because they have lower working memory demands. These are partially, but not fully, accounted for by the nuisance parameter for whether a context was repeated sequentially ($\beta_{ctx\ repeat}$: 95% HDI $= [0.44, 0.96]$). More important is the finding of initial positive transfer in test context 1, which is difficult to explain as a working memory effect. Nonetheless, we cannot fully rule out a working memory account in this design.

An alternate hypothesis would be that increased accuracy in the training conditions accounted for some of the context differences during test. For this hypothesis to account for the data, we would have expected to find greater accuracy in training contexts with goal A than in contexts paired with B or C, which we failed to find.

In the following section, we attempt frequency as a potential confound by controlling for the frequency in the test contexts. To do so, we introduce an additional goal to allow us to balance goals, mappings and frequency. In addition, we bias the design to favor independent clustering by further dissociating mappings and goals in

the training phase by having multiple goals associated with both mappings.

3.4 Experiment 2

Experimental Task

Participants performed a similar navigation task as the task presented in section 3.3. Subjects navigated a 6x6 gridworld and selected one of four labeled goal locations (labeled A, B, C, and D) to receive reward. Subjects controlled an agent that could be moved in four cardinal directions (North, South, East and West) using eight keys on a standard keyboard ('a', 's', 'd', 'f', 'j', 'k' , 'l', and ';'), four of which deterministically corresponded to movement in a cardinal direction in the grid.

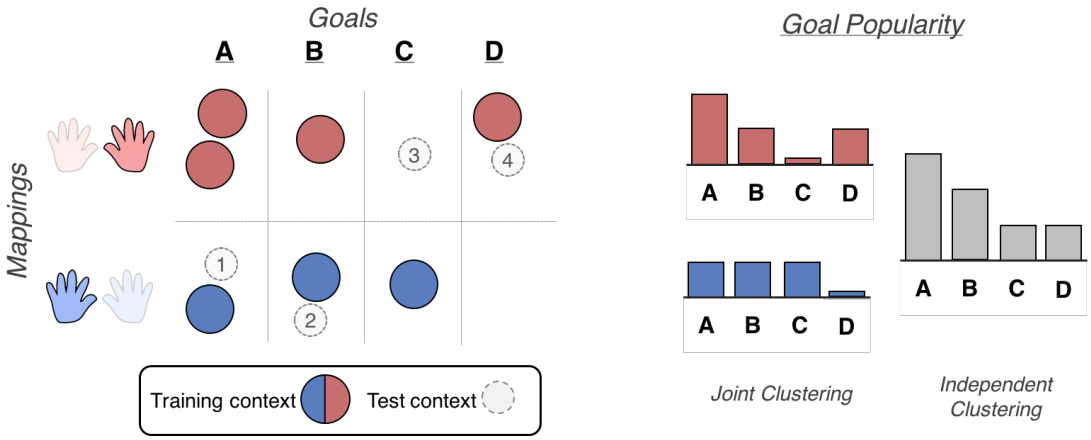

Figure 3.7: Experiment 2: Task design. *Left*: Subjects saw seven contexts during training (red/blue circles), each paired with one of two mappings (*red*: high popularity, *blue*: low popularity) and one of four goals (A, B, C or D). Subjects saw an additional four contexts following training (numbered grey circles). *Right*: Goal popularity both as a function of mapping (left) and collapsed across all contexts (right). Both the joint and independent clustering predict goal generalization based on context-popularity, but differ in their consideration of mappings. Joint clustering (left) is sensitive to the mapping when determining goal popularity while independent clustering (right) predicts goal generalization based on the popularity of all contexts.

Seven training contexts were cued to the subject [Figure 3.7, Table 3.2]. To aid with the additional memory load, in addition to cuing context with the color of the

Context	Goal	Mapping Popularity	n Trials
Train 1	A	Low	20
Train 2	A	High	10
Train 3	A	High	10
Train 4	B	Low	20
Train 5	B	High	20
Train 6	C	Low	40
Train 7	D	High	40
Test 1	A	Low	6
Test 2	B	High	6
Test 3	C	High	6
Test 4	D	Low	6

Table 3.2: Experiment 2 task design. The number of trials within each context is balanced such that each goal and each mapping is presented the same number of trials across both training and test.

agent and goals, a fixed pattern of walls on every trial with a context and a "Room" number at the top of the screen also cued the context. Across the seven training contexts, subjects needed to learn two mappings, each with all of the valid responses for a mapping were grouped on either the left or right hand. Subjects were required to learn the association between the contexts, mappings and goals from experience.

Goal popularity was modulated parametrically across training contexts, with goal A repeated three times, goal B repeated twice and goals C and D paired with one context each. Mappings were chosen to dissociate goal popularity as a function of context and so that both goals A and B were associated with both mappings. Following a total of 160 training trails across the seven contexts, subjects saw 6 trials each of 4 novel contexts. Each test context was uniquely paired with a single goal and each mapping was paired with two goals, such that in the test contexts, all goals and mappings were matched for both frequency and popularity. The goal labels (A, B, C and D) as well as the keys that define each mapping were randomized between subjects.

Model Predictions

The four test contexts were selected to distinguish independent and joint clustering through the speed of learning correct goal. An independent and joint clustering agent were trained on 250 random initializations of the behavioral task. As a control, a flat agent that does not cluster was also simulated. The clustering models each have two free parameters, a CRP concentration parameter that controls the models' tendency to generate a new cluster, α, which was drawn from a standard lognormal distribution, and a softmax inverse-temperature parameter β that controls stochastic exploration and was fixed $\beta = 2.0$.

Both clustering agents collected fewer rewards per trial than the flat agent in both the training and test contexts, suggesting the absence of a normative benefit of generalization in the task [Figure 3.8]. The average reward collected per trial by the flat agent was higher than both the reward collected by the independent clustering and joint agents during both training and test ($r \sim \text{Binom}(\theta)$; Training: $\theta_{flat} = 0.89$, $\theta_{ind} = 0.76$, $\theta_{joint} = 0.85$, $\theta_{flat} - \theta_{ind}$ 95% HDI=[0.116, 0.127], $\theta_{flat} - \theta_{joint}$ 95% HDI=[0.03, 0.04]; Test: $\theta_{flat} = 0.65$, $\theta_{ind} = 0.53$, $\theta_{joint} = 0.58$, $\theta_{flat} - \theta_{ind}$ 95% HDI=[0.104, 0.141]; $\theta_{flat} - \theta_{joint}$ 95% HDI=[0.049, 0.087]).

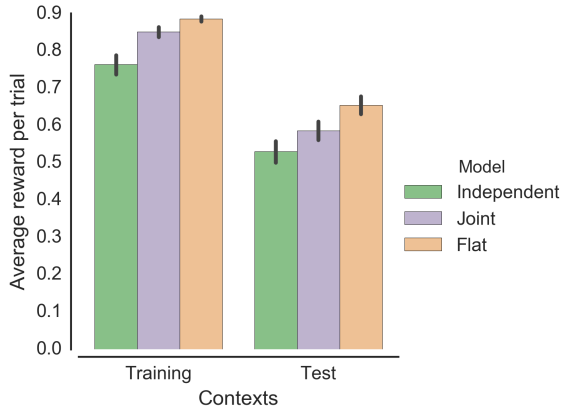

Figure 3.8: Experiment 2: Model performance on the task. A flat agent that does not cluster receives higher reward in both the training and test contexts than either the independent or joint clustering agents.

Differences in test context accuracy are the main empirical predictions of the models. Independent clustering predicts contexts will be learned more quickly if they

have a more popular goal [Figure 3.9A]. Test context 1 is paired with the highest popularity goal (goal A) and is learned faster than test context 2, which is paired with an intermediate popularity goal (goal B). Test contexts 3 and 4 have goals C and D, respectively, which were each low popularity goals and are consequently predicted to be learned to slowest. Importantly, only test context 1 is predicted to show positive transfer in the first trial. The joint clustering model, in contrast predicts negative transfer in the first trial of all four contexts due to the influence of mappings [Figure r3.9B]. This effect is predicted to be weaker in test contexts 1 and 2.

As in section 3.3, we can define specific contrast to highlight the predictions of the models. Both models predict contests 1 and 2 will be learned faster than 3 and 4 and both models predict, to a lesser degree, that context 3 will be learned more slowly than context 4 [Figure 3.9C, D]. Independent clustering makes the unique prediction that test context 1 will be learned faster than test context 2 [Figure 3.9C].

In addition, the two models make a more detailed prediction as to which goals will be guessed in the first trial of a test context as a function of which mapping the context was paired with [Figure 3.10]. Independent clustering predicts the same qualitative pattern across the two mappings, tending to guess A followed by B followed equally by C and D. In contrast, joint clustering predicts a qualitatively different pattern of results dependent on the mapping used in the context. When the high popularity mapping is observed in a test context, joint clustering predicts approximately the same goal preference as independent clustering, with goal A as the most likely selection. In the low popularity mapping, however, joint clustering predicts no preference between that A, B and C and is systematically less likely to choose goal D.

Participants

An additional 156 online participants were recruited as subjects through Amazon's Mechanical Turk platform. Six participants reported taking notes and were excluded from analysis. An additional 28 subjects were identified with cluster analysis as having chance performance (see Appendix A for details), leaving 120 subjects carried through to analysis.

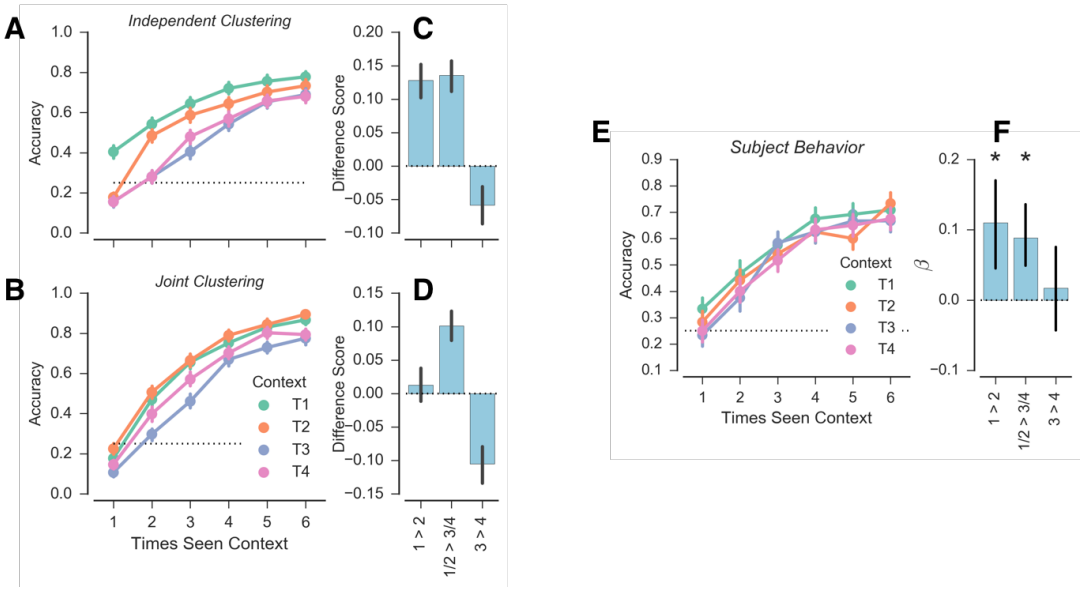

Figure 3.9: Experiment 1 Model predictions and behavioral results in test contexts. *A,B*: Accuracy of independent (A) and joint (B) clustering models of the four test contexts as a function of trials in the context. Chance accuracy is denoted by dotted line. *C,D*: Comparisons of accuracy in test contexts highlight differences between the model. Independent clustering predicts test context 1 will be learned more quickly than contexts 2 and 4 (C, left), while joint clustering predicts a difference in test accuracy for test context 3 as compared to 2 and 4 (D, middle) and a smaller difference between contexts 2 and 4 (D, right). *E*: Subject accuracy in test contexts across time. *F*: Regression coefficients comparing test contexts 1 vs 2/4, 3 vs 2/4 and 2 vs 4.

Behavior

First, we again explore subject performance during training to verify subjects learned the task [Figure 3.11]. We again assess accuracy, reaction time and navigation efficiency as performance measures with linear models (see methods). Accuracy was found to increase across time (β_t: 95% posterior highest density interval (HDI) = [0.095, 0.107]) as was response time in hz (β_t: 95% HDI = [0.048, 0.061]) and the number of steps greater than the minimum path length (β_t: 95% HDI = [-0.0065, -0.0052], all of which are consistent with subjects learning the task.

A main prediction distinguishing the models is the difference between in accuracy between test contexts. Again, we evaluated differences in accuracy between test contexts with a Bayesian logistic regression, including the number of trials in a context

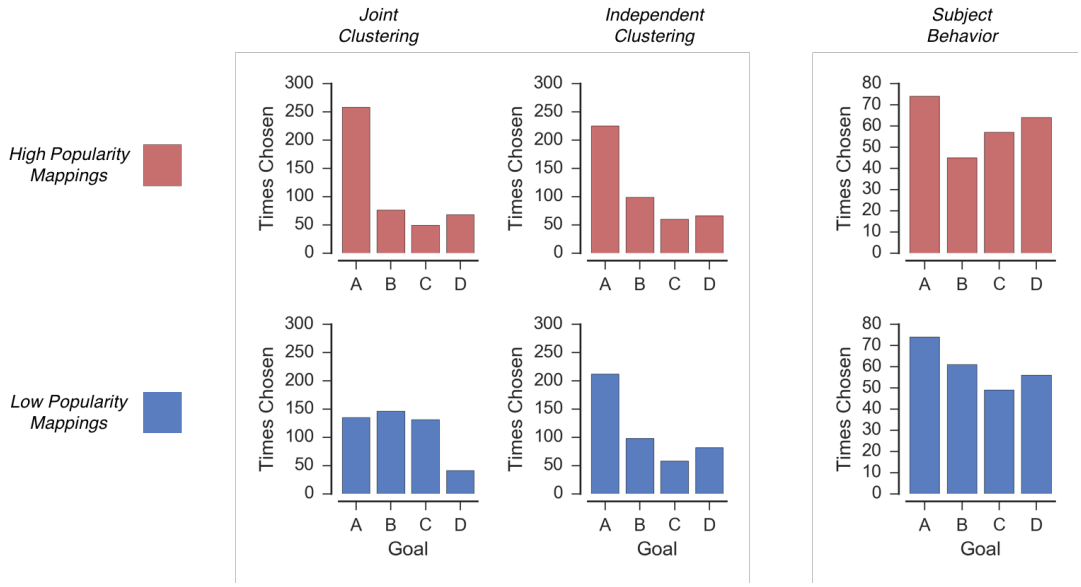

Figure 3.10: Experiment 2: Frequency of goal choices in first trial of a test context, split by mapping. Test context 1 and 2 (*left*) share a mapping while test contexts 3 and 4 (*right*) share the other mapping.

and whether a context was repeated sequentially as nuisance predictors, as well as a subject specific bias term (see methods). The key contrasts of interest were test context 1 vs test context 2 (a unique prediction of independent clustering), test contexts 1 and 2 vs 3 and 4, and test context 3 vs 4. Each was included as a predictor in the linear model.

A logistic model including our contrasts of interest provided a better fit to the data than a null model that did not include these contrasts but was otherwise identical (WAIC = 3322.9; null model WAIC = 3324.04). Consistent with the *a priori* prediction of independent clustering, subjects had greater accuracy in test context 1 than in test context 2 (Figures 3.9E, 3.12B; $\Pr(\beta_{T1-T2} > 0) > 0.959$, 95% HDI = [-0.01, 0.24]). Consistent with the predictions of both clustering agents, accuracy in test contexts 1 and 2 greater than in context 3 and 4 ($\Pr(\beta_{T1/2-T3/4} < 0) > 0.979$, 95% HDI = [0.004, 0.174]). We failed to find sufficient evidence that test context 3 was learned more slowly ($\beta_{T1/2-T3/4}$ 95% HDI = [-0.10, 0.13]).

Further supporting the regression analysis is positive transfer in the first trial of

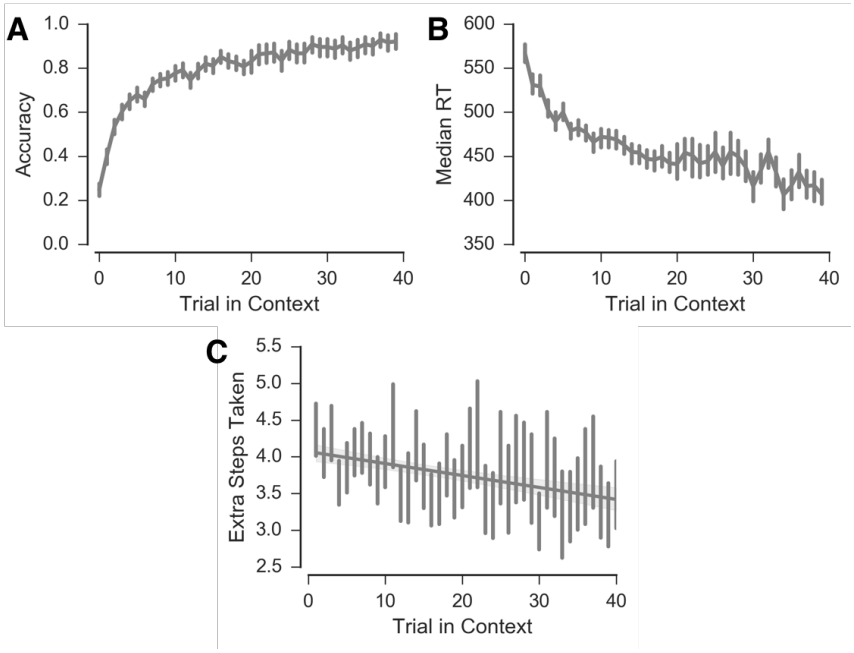

Figure 3.11: Experiment 2: Performance in training contexts. *A*: Subject accuracy as a function of trials within contexts. *B*: Median RT as a function of trials within context. *C*: Navigation efficiency, as measured by the number of responses over the minimum path length needed to reach the selected goal.

test context 1, a key prediction unique to independent clustering [Figure 3.9A]. Accuracy in the first trial of test context one was greater than chance ($y \sim \text{Binom}(\theta)$; $\Pr(\theta_1 > 0.25) > 0.98$, 95% HDI $= [0.252, 0.416]$). We found no evidence of positive or negative transfer in the first trial of the other three test contexts, a null result (θ_2 95% HDI=[0.203, 0.363]; θ_3 95% HDI = [0.159, 0.310]; θ_4 95% HDI=[0.172, 0.326]).

Next, we looked at subjects' selection of individual goals in the first trial of a test context, collapsed across all subjects [Figure 3.10]. Independent clustering predicts goal A will be chosen more frequently and predicts no difference as a function of mapping. Joint clustering predicts a difference as a function of mapping, with goals A, B and C equally favored in contexts with the low popularity mapping while strongly favoring goal A with the high popularity mapping.

We first tested whether there were any differences in which goals were selected through model selection (see methods 3.7). Allowing the choice probabilities for goals A, B, C and D produced a better model fit than a null model (WAIC = 2154.5, null

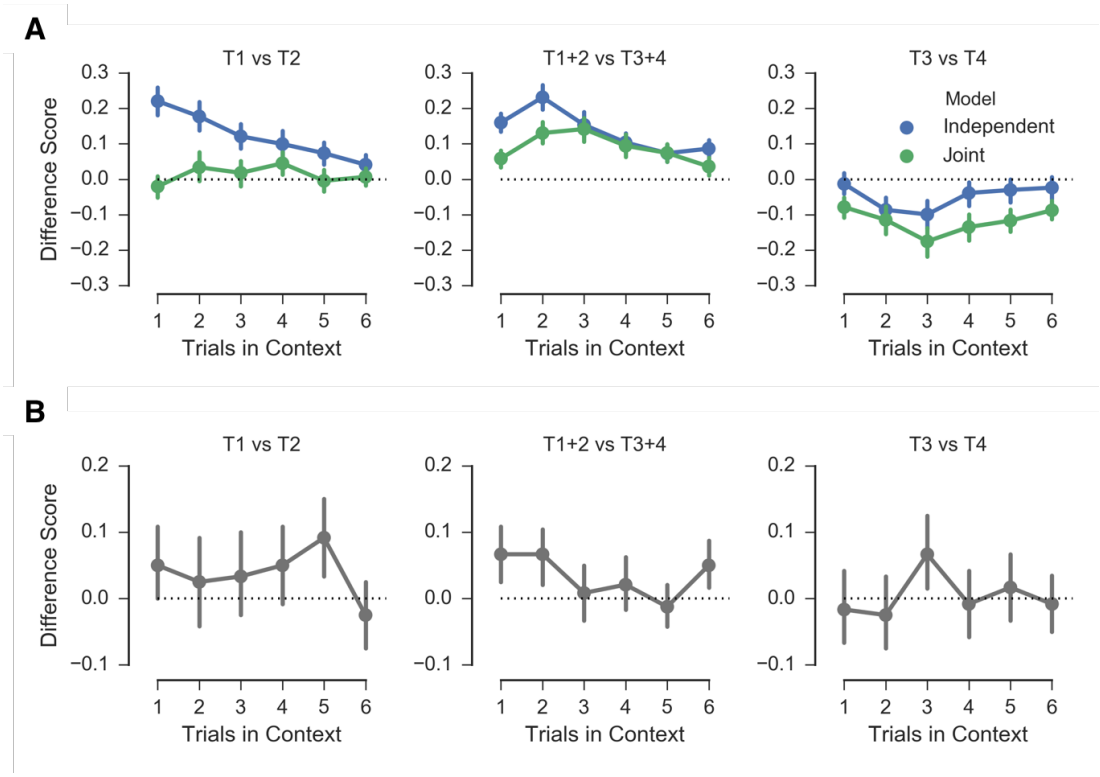

Figure 3.12: Experiment 2, difference scores between test contexts across time. Differences shown between test contexts 1 and 2 (left), 1 and 2 vs 3 and 4 (middle), and 3 vs 4 (right). *A*: Model predictions for both joint (green) and independent (blue) clustering. *B*: Subject behavior.

model WAIC = 2161.4), consistent with both clustering models. Across all test contexts, goal A was chosen the most often [Figure 3.10]. Goal A was chosen above chance rate ($\Pr(\theta_A > 0.25) > 0.998$, 95% HDI=[0.267,0.349]) and was chosen more often than the other three goals ($\Pr(\theta_A > \theta_B) > 0.999$, 95% HDI=[0.032,0.141]; $\Pr(\theta_A > \theta_C) > 0.998$, 95% HDI=[0.032,0.142]; $\Pr(\theta_A > \theta_D) > 0.977$, 95% HDI=[0.002,0.116]). Both clustering models further predicted a parametric effect in which goal B was chosen more often than goals C and D, and where goal C is chosen more often than goal D, both of which we failed to find ($\theta_B - \frac{1}{2}(\theta_C + \theta_D)$ 95% HDI=[-0.059,0.031]; $\theta_C - \theta_D$, 95% HDI=[-0.084,0.024]).

Joint clustering makes the further prediction that goal selection in the first trial of a test context will differ by mapping. In contrast to the predictions of joint clustering, model fit was not improved by allowing goal choice probability to vary as a function of

mapping and instead resulted in a higher value of WAIC (ΔWAIC= 3.6). Follow up analysis examined the *a priori* predictions of the independent clustering model that goal A would have higher choice probability than goals B and C in low popularity mapping contexts [Figure 3.10, bottom]. Choice probability was higher for goal A than goal C ($\Pr(\theta_A > \theta_C) > 0.995$ 95% HDI=[0.029, 0.185]), but there was only week evidence to suggest that choice probability was higher for goal A than goal B ($\Pr(\theta_A > \theta_B) > 0.906$ 95% HDI=[-0.024, 0.127]). These results are inconsistent with joint clustering and consistent with independent clustering.

In order to investigate whether a learned value of goals across all contexts is driving the generalization results, we ran a follow up analysis on the accuracy of goals during training. As in the previous experiment, we expect that subjects are learning a context-independent value for each goal and searching new contexts based on this, then we would expect goals that showed positive transfer to have a higher learned value (i.e. more rewards received). To address this possibility, we analyzed training goal accuracy with the identity of the goal as predictors of interest. Goals A, B and C each were given a parameter β_A, β_B, and β_C, respectively, while goal D was used as a baseline. We also included a subject specific bias term as well as whether context was repeated and the number of trials within a context as nuisance parameters. We found little evidence that goal A was learned the best during training. While training contexts with goal A had higher accuracy ($\Pr(\beta_A > 0)$ 95% HDI = [0.028, 0.22]) than contexts with goal D, the accuracy of contexts with goal A was not higher than the accuracy in those with goals B or C ($\Pr(\beta_A > \beta_B)$ 95% HDI = [-0.12, 0.04]; $\Pr(\beta_A > \beta_C)$ 95% HDI = [-.11, 0.08]), suggesting accuracy in the training contexts (independent of clustering) was insufficient to explain the pattern of responses in the test contexts.

Discussion

These results replicate our findings in the previous experiment, again showing popularity based clustering and evidence consistent with independent clustering. In this experiment, we did not find evidence to support joint clustering. Goal A, the highest popularity goal, was guess more often regardless of mapping and test context 1, paired with goal A, was learned more quickly. This effect does not appear to be driven by

an increase learning of training contexts with goal A, nor can it be accounted for by a frequency effect between test contexts.

Interestingly, while both clustering models predicted negative transfer (including the prediction by both models that test contexts 3 and 4 would show negative transfer), we found no evidence for it in the current experiment. Similarly, we found no evidence for the parametric effect predicted by the independent clustering model.

These may be related effects if the generalization effect we observe in our data are caused by the cumulative effects of a directed search strategy and not be more efficient learning. One possibility is that subjects search the goal-space as a function of popularity based clustering and learn the goals equally. This is in contrast to an alternative possible generalization mechanism that we did not observe in our data, in which all goals are searched equally but higher-popularity goals are learned more quickly. If subjects largely realize a generalization benefit as a benefit of directed search, than a lack of negative transfer would reduce any compounding benefits. In current task, the independent clustering model did not predict an initial positive transfer for goal B, the intermediate popularity goal [Figure 3.9A]. As such the absence of negative transfer may be related to the lack of a parametric finding, though an alternative possibility is that the experiment is simply under-powered to detect this difference.

However, while the current results suggest subjects behave consistently with independent clustering, normative behavior depends on the environment. As discussed in Chapter 2, in task domains where there is a relationship between goals and mappings, it is more adaptive in to cluster jointly. In the experiments presented thus far, there isn't an obvious association between goals and mappings. Nonetheless, the experiment in section 3.3 showed evidence for joint clustering in addition to joint clustering. One possibility is the that task design is influencing the subjects' strategy. It is possible, given the loose relationship between goals and mappings in the training contexts in the experiments thus far, subjects did not expect mappings and goals to be linked and clustered independently (even though clustering there is a cost to goal generalization in this task). This leave open the question, what happens when there is an association between mappings can goals? Do people respond to the association and cluster jointly? We ask this question in the following section.

Context	Goal	Mapping	n Trials
Train 1	A	High Pop	8
Train 2	A	High Pop	8
Train 3	B	Low Pop	16
Joint Test 1	A	High Pop	4
Joint Test 2	A	High Pop	4
Joint Test 3	B	Low Pop	8
Indep Test 1	A	Low Pop	4
Indep Test 2	A	Low Pop	4
Indep Test 3	B	High Pop	8

Table 3.3: Experiment 2: task design

3.5 Experiment 3: Joint clustering

Task Design

Participants performed a similar navigation task as the task presented in sections 3.3 and 3.4. Subjects navigated a 6x6 Gridworld and selected one of two labeled goal locations (labeled A and B) to receive reward. Subjects controlled an agent that could be moved in four cardinal directions (North, South, East and West) using eight keys on a standard keyboard ('a', 's', 'd', 'f', 'j', 'k' , 'l', and ';'), four of which deterministically corresponded to movement in a cardinal direction in the grid. Subjects completed a short tutorial and were required to respond correctly to questions about the rules the game and how they were rewarded prior to starting the experiment.

Subjects saw six contexts over the course of the experiment, split into three test contexts preceded by three training contexts [Figure 3.13, Table 3.3]. Each context was cued to the subject by the color of the agent and goal locations. All subjects saw the same training contexts. Two of the training contexts were paired with a high-popularity goal (goal A) and a high popularity mapping while the other training context was paired with the low-popularity goal (goal B) and the low-popularity mapping. This was done to introduce a relationship between the goals and mapping such that there is no uncertainty of the correct goal given the mapping.

The subjects were randomly assigned into two groups for the test contexts. The first group received the "Joint" design in which the statistics of the training contexts

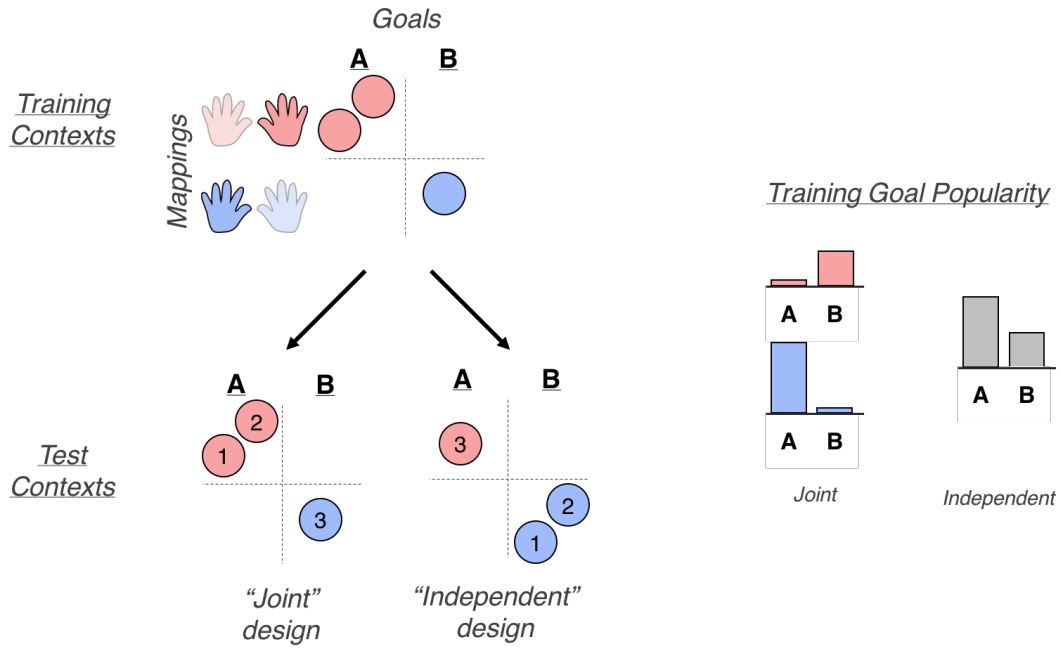

Figure 3.13: Experiment 3: Task design. *Left*: Subjects saw three contexts during training, in which each goals and mappings were related. For the test contexts, subjects with saw a repeat of the 3 new contexts with the goal popularity and goal/mapping relationship ("Joint design") or a set of three new contexts in which the goal/mapping relationship had been reverse ("Independent design") *Right*: Goal popularity both split by mapping (left) and collapsed across mapping (right)

were repeated. In the joint test contexts, goal A was again paired with the high-popularity mapping in two contexts (test contexts 1 and 2) while goal B was paired with the low-popularity mapping in a single text context (test context 3). The second group received the "Independent" design in which the relationship between mappings and goals was switched. In the independent test contexts, goal A was paired with the low-popularity mapping in two contexts (test contexts 1 and 2) whereas the goal B was paired with the high-popularity mapping in a single context (test context 3).

Model Predictions

Critically, all of the training contexts have the same goal statistics independent of mapping. Thus, if subjects are clustering goals independently of mappings, there will be no between-subject differences the two test conditions [Figure 3.14A]. In contrast,

joint clustering predicts that the two test conditions will be meaningfully different, with higher accuracy in the Joint test context than in the Independent test contexts. In addition to the differences predicted in the test conditions, both models predict an accuracy difference during training between contexts paired with goal A and goal B.

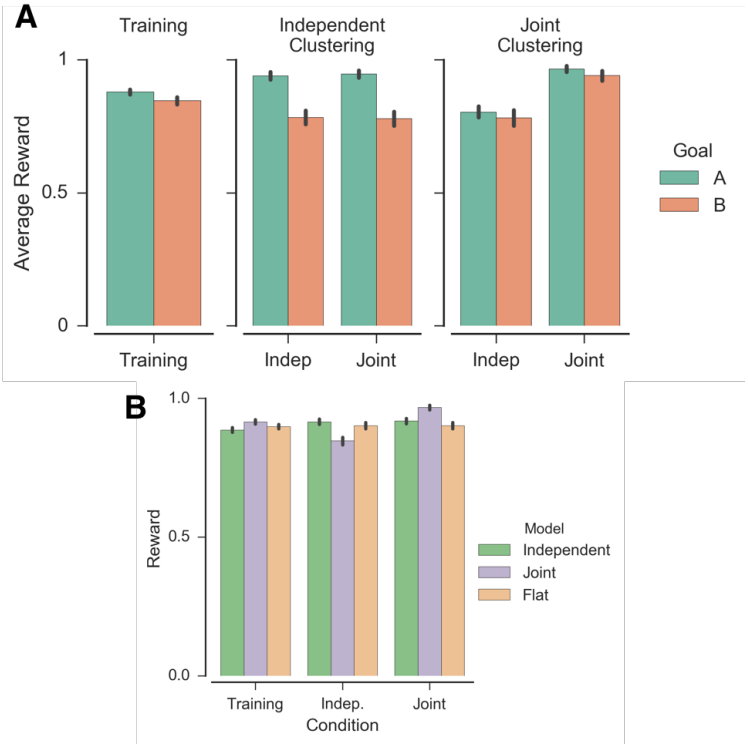

Figure 3.14: Experiment 3 Model predictions. *A*: Average reward collected in the first four trials of a context, collapsed across goal, for training (left, collapsed across both clustering agents), the independent clustering condition (center) and the joint clustering conditions (right). *B*: Average reward per trial in training contexts (left) and the test contexts for the independent (center) and joint conditions (right).

In a separate prediction, the independent model predicts a difference in accuracy of test contexts, with higher accuracy in test contexts 1 and 2 than in the equivalent trials of test contexts 3 across both groups ($y \sim \text{Bernoulli}(\theta)$; $\theta_A - \theta_B$ 95% HDI=[0.128, 0.184]). The joint model predicts this difference as well, but to a much smaller degree that may or may not be detectable experimentally ($\Pr(\theta_A > \theta_B) > 0.917$ 95% HDI=[-0.01, 0.053]) This happens as a consequence of the goal popularity. In both groups, the test contexts 1 and 2 share the high popularity goal (goal A). As such, the independent

clustering model will learn them faster than test context 3, which is paired with the low popularity goal (goal B). In contrast, the joint model makes its inference on the goal popularity taking into account the observed mapping. In the joint test contexts, goal popularity is high for all contexts conditional on the mapping (and the training statistics), and the joint model see positive transfer in all test contexts. In the independent test contexts, all of contexts have low-popularity goals conditional on the observed mapping, such that all contexts will be show negative transfer. Because there are only two goals to explore, the effects of exploration beyond positive/negative transfer are minimal.

It is important to note that the training contexts, the normative strategy is to jointly cluster ($r \sim$ Bernoulli(θ); $\theta_{joint} = 0.915,\ \theta_{ind} = 0.886,\ \theta_{flat} = 0.898$; $\theta_{joint} - \theta_{flat}$ 95%HDI = [0.008, 0.026]). Subjects can't know this *a priori*, but it is possible that this biases their expectation for the test contexts, and they adopt a strategy in the test contexts in which they rely on the mappings to inform their guess of the goals. The normative strategy in the test context depends on the task conditions. In the Joint condition, it is normative to jointly cluster while in the Independent condition, it is normative to independently cluster ("Joint" condition: $\theta_{joint} = 0.968,\ \theta_{ind} = 0.918,\ \theta_{flat} = 0.901$; $\theta_{joint} - \theta_{flat}$ 95%HDI = [0.058, 0.078]; "Independent" condition: $\theta_{joint} = 0.847,\ \theta_{ind} = 0.916,\ \theta_{flat} = 0.901$; $\theta_{ind} - \theta_{flat}$ 95%HDI = [0.002, 0.028]). Assuming subjects can observe the normative strategy at all, they would be unlikely to rely on the normative value of the test contexts as they would not be able to know the correct strategy until after observing the correct response. Hence, if subjects are sensitive to how the strategies interact with the task domain, they are most likely to be sensitive to the conditions of the training contexts.

Hence, part of the logic of the study is to ask if subjects respond to the normative context of the task domain and use that as an expectation for future generalization. This would be consistent with the theoretical observation that neither independent nor joint clustering is normative across all task domains (See Chapter 2).

Participants

198 subjects were recruited online using Amazon's Mechanical Turk platform. 2 subjects were excluded for reporting they took written notes during the experiment

and 65 subjects were excluded for poor performance in the training contexts, as determined by cluster analysis (see Appendix A for details). Of the subjects included in the analysis, 50 subjects were randomly assigned the Joint test contexts and 81 subjects were assigned the Independent test contexts.

Behavior

Accuracy in the training contexts was analyzed with a logistic regression in which the number of trials within a context and a whether the context was sequentially repeated were included as predictors as β_t and β_r, respectively. Individual subjects were accounted for by a subject specific bias, with each subject specific bias assumed to be normally distributed around a group average. Subjects accuracy, response time, and navigation efficiency improved accuracy across time, indicating subjects learned the training contexts (Accuracy: $\beta_t = [0.122, 0.176]$; reaction time $\beta_t = [0.027, 0.052]$; navigation efficiency: β_t was [-0.029, -0.017]).

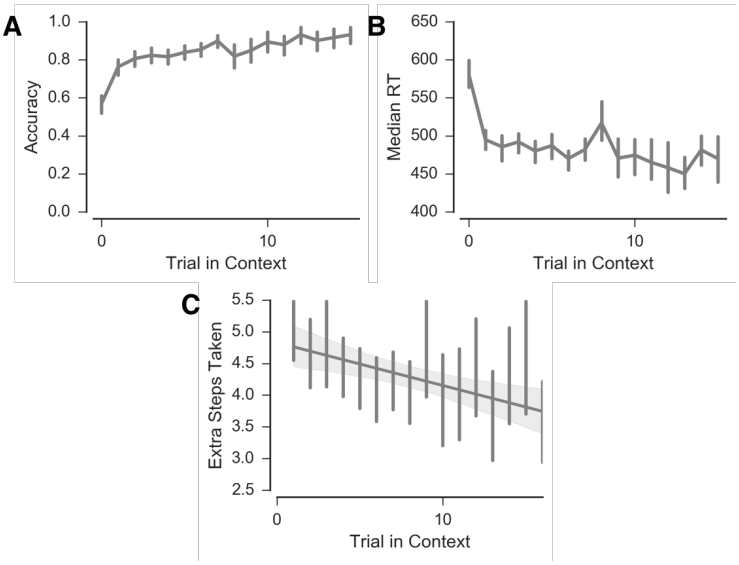

Figure 3.15: Experiment 3, performance in training *A*: Subject accuracy across time. *B*: Reaction Time across time. *C*: Navigation efficiency, as measured by the number of responses over the minimum path length needed to reach the selected goal.

The main predictions of the task were assessed with logistic regression (see Methods). As this was a between subjects design, we analyzed both the training in test

data in one model. This allows us to better estimate individual subject bias terms to account for individual differences without confound, as all subjects saw the same statistics in the training contexts. As *a priori* predictors of interest, we included goal A vs goal B during training ($\beta_{A_{\text{train}}}$) and test ($\beta_{A_{\text{test}}}$) as well as a predictor for the between subjects Joint vs Independent test conditions (β_{task}). We also included whether the context was repeated as well as the number of times a subject had seen a context as nuisance predictors and a comparison of training vs test overall. The regression model provided a better fit to the observed data than a null model that did not include the predictors of interest (WAIC = 3303.5, null model WAIC = 3326.87).

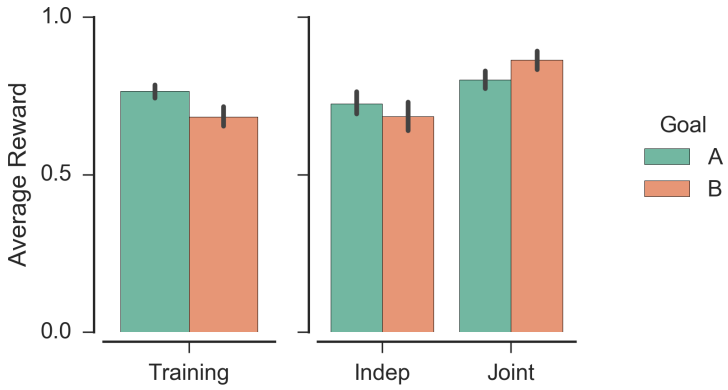

Figure 3.16: Experiment 3 Subject behavior in training contexts (left) the "Independent" test design (middle) and the "Joint" test design (right) as a function of the correct goal in the context.

Consistent with the joint clustering, we found subjects in the "Joint" test design had greater accuracy than subjects in the "Independent" test design (β_{task}, 95% HDI =[0.128, 0.421]). Furthermore, across both test conditions there was no difference between accuracy in contexts with goal A and contexts with goal B, representing a failure to find the key prediction of independent clustering ($\beta_{A_{\text{test}}}$, 95% HDI =[-0.138, 0.128]). Both models predicted contexts with goal A would be learned more quickly than contexts with goal B during training, a prediction that was borne out by the data ($\beta_{A_{\text{train}}}$, 95% HDI =[0.148, 0.387]).

3.6 General Discussion

In the current work, we demonstrate how clustering models of generalization can be expanded to account for simple compositional behaviors. We show how clustering goal and world models independently as a model of generalization can capture aspects of human behavior that previous models of generalization, which generalized them together, fails to predict. Qualitatively, independent and joint clustering are dissociable models of how people can exploit regularities in their environment. While the normative value of either strategy depends on the task domain (Chapter 2), independent clustering represents a basic form of compositionality long to be a critical part of human intelligence.

In this set of experiments, we used both models to generate qualitatively different patterns of behavior in a multistep navigation task that facilitates the separate learning of goal values and world models. In all three experiments, subjects learned a set of training contexts and were given novel test contexts to probe how they used the learned statistics of the training contexts as an estimate of the statistics of the novel contexts. Taken together, the results of the three studies provide a replication of context-popularity based clustering in human subjects as a model of generalization (Collins and Frank, 2013; Collins et al., 2014; Collins and Frank, 2016a). Subject behavior showed evidence of popularity-based clustering in all three experiments. In each experiment, goals were balanced such that each goal was correct in the same number of trials. Even though popular contexts were experienced less often, subjects tended to generalize the goals associated with popular contexts, suggesting popularity-based generalization. Furthermore, in a replication of previous work, subjects tended to generalize even as they paid a cost to do so (Collins et al., 2014).

We further found evidence that subjects are capable of both joint and independent clustering. In the above experiments, we were able to dissociate generalization of goals from mappings by dissociating learned goal-values from navigation with a multistep navigation task that enforced planning (See Appendix B for an analysis of mapping generalization). In experiment 1, goal generalization was consistent with both independent and joint clustering. Experiment 2 highlighted independent clustering while experiment 3 encouraged joint clustering by creating a clear association

between mappings and goals. While we have presented independent and joint clustering as two separate strategies here, it is perhaps better to present them as bound on a spectrum of strategies. Theoretical analysis suggests that neither independent nor normative clustering is always normative (Chapter 2), suggesting that a mix of strategies may be adaptive. Indeed, we show that people exhibit behavior that is neither purely independent nor purely joint and instead respond to the task domain.

It is important to note that not all of the predictions of the model were borne out. Clustering models of human and animal generalization typically rely on the CRP prior (Gershman et al., 2010; Collins and Frank, 2013). In a prediction shared with the current work, these models suggest a parametric effect such that contexts with intermediate popularity will be preferred less than contexts with high popularity but more than contexts with low popularity. We failed to find any evidence of parametric effects. Assuming this null effect does not simply reflect an absence of statistical power, there are multiple potential explanations. One is that subjects do not cluster parametrically but instead favor a single popular cluster in a simpler mechanism. An alternate possibility is that subjects do cluster parametrically via the inference process but use a different exploration strategy. Conceivably, this may also interact with the difficulty of the task. In experiments 2 and 3, there were only 3 or 4 possible goals, respectively. Normatively, there was no benefit of generalization in these restricted domains, and the absence of a normative benefit may have reduced the subjects' motivation to generalize to the point where we were not able to detect parametric effects. It is possible in a more difficult task where there is a clear benefit of generalization that subjects may show a parametric tendency to generalize.

A further limitation in the current study is working memory demands. In order to complete the task, a subject has to correctly remember each association between contexts and their associated functions. While increasing the number of contexts in the task can increase the ability to make detailed predictions distinguishing the models, the increased working memory load can act as a countervailing force. Subjects accuracy was lower in the training contexts in experiment 2 than in experiment 1 (β_{exp} 95%HDI = [-0.617, -0.172]) and similarly lower in experiment 1 than in experiment 3 (β_{exp} 95%HDI = [-0.821, -0.404]). This reflects the complexity of the design, as experiment 2 was the most complex and experiment 3 was the least, and may be

a consequence of increased working memory load. This may cause a problem for interpreting the test data if in a more complex design, subjects have a worse mastery of the training statistics overall, and thus, weaker generalization.

Our findings suggest that subjects can generalize aspects of the task structure independently of each or jointly. Importantly, this suggests that subjects have separate representations of rewards and transitions that are in some degree compositional. Even in cases where it is beneficial to generalize rewards and transitions together, it may be useful to represent them as separate functions (as opposed to learning a single policy function). Separate representations may allow for further flexibility in unseen environments. For example, if we changed the structure of the task from navigating an agent to a goal location to using the agent to pushing a block in the grid-world, we might expect subjects to reuse previously learned mappings. This flexibility is of an altogether different form than the question of whether it is useful to exploit the correlation structure of the task domain when generalizing

Importantly, context-popularity based clustering is thought to involve the same fronto-striatal circuitry that underlies simple feedback based learning (Collins and Frank, 2013; Collins et al., 2014). This leaves open many questions, the first of which is to what degree is the generalization behavior exhibited in these experiments dependent on fronto-striatal circuitry? One simple mechanism to enforce compositional representations of goals and values would be to embed in independent systems. It is possible learning mappings is a motor learning task dependent on the basal ganglia, cerebellum and/or motor areas of the cortex (Doya, 1999) while learning goal values are more represented in areas of the frontal cortex that represent value and or state (Batra et al., 2013). While the orbital frontal cortex has been proposed as representing a latent state (Wilson et al., 2014b; Schuck et al., 2016), it is possible that generalization relies on more complex representations that allow recombinations across between states. A further question is how the relationship between mappings and rewards is learned. One possibilities is that this is also learned hierarchically, as the frontal cortex is thought to embed multiple levels of abstraction Badre et al. (2010), potentially corresponding to a front such that a higher level in the hierarchy controls whether environment is expected to have an exploitable correlation structure. While the question of how goal values and mappings are separately represented

in the brain in such a way that allows a compositional recombination is beyond the scope of the current work, the findings here suggest that people are nonetheless able to learn simple correlations between these representations.

3.7 Methods

Modeling Details

Our models estimate two functions, a reward function over goals ($R_c(g) = \Pr(r|g, c)$) and a mapping function between eight available primitive actions and four available cardinal movements ($\phi_c(a, A|c) = \Pr(a, A)$). The agent is provided a transition function that includes the locations of the goals as well as the locations of any walls as an assumption. Consequently, all the agent needs to learn to solve a trial is the reward function over goals and mapping function. This simplifying assumption has the benefit of providing a model of human navigation, whom we assume understand spatial structure. We assume the planning problem (navigating a 6x6 Gridworld) to be trivial for a human subject and as such, solve action values using value iteration.

At each point in time, the agent is given a state tuple $s =< x, c >$ where x is the agent's location in 6x6 coordinate space and $c \in \{1, .., n_{ctx}\}$ is a context variable. Here, we model generalization as generating clusters of contexts $k = \{c_{1:n}\}$ that share common task statistics. We consider two decompositions of these task statistics to consider two alternate hypotheses. In one, the task statistics are decomposed into reward functions and mapping functions separately. This allows the agent to cluster context twice, into two independent sets of context clusters $K_r = \{k^r_{1:n}\}$ and $K_\phi = \{k^\phi_{1:m}\}$. The second decomposition treats the reward and mapping functions jointly and generates a single set of context clusters $K = \{k_{1:n}\}$. We refer to the fist decomposition as *Independent clustering* as the agent creates two independent sets of cluster and we will refer to the second decomposition as *Joint clustering*.

We define generalization as the inference of contexts into clusters. Stated formally, this is

$$\Pr(c \in k|\mathcal{D}) \propto \Pr(\mathcal{D}|c \in k) \Pr(c \in k) \tag{3.1}$$

where $\Pr(\mathcal{D}|c \in k)$ is the likelihood of the observed data $\mathcal{D}$ given context c is a member of cluster k and $\Pr(c \in k)$ is a prior over the clustering assignment. As in

previous models of generalization, we use the CRP as the cluster prior. If contexts $\{c_{1:n}\}$ are clustered into $N \leq n$ clusters, then the prior probability for any new context $c_{n+1} \notin \{c_{1:n}\}$ is:

$$p(c_{t+1} \in k | c_{1:n}) = \begin{cases} \frac{N_k}{N+\alpha} & \text{if } k \leq K_n \\ \frac{\alpha}{N+\alpha} & \text{if } k = K_n + 1 \end{cases} \tag{3.2}$$

where N_k is the number of contexts associated with cluster k and K_n is the number of unique clusters associated with the n observed contexts. If $k \leq K_n$, then k is a previously encountered cluster whereas if $k = K_n + 1$, then k is a new cluster. The parameter α governs the propensity to assign a new context to a new cluster. Higher values of α lead to a greater prior probability that a new cluster is created for a new context whereas lower values of α will favor the re-use of previous clusters and thus, favor generalization. As the choice of α is ultimately arbitrary, for each simulation the value of α was sampled from the log-normal distribution $\ln \alpha \sim \mathcal{N}(0, 1)$.

Independent and joint clustering

As we noted above, there are two key functions the agent learns when navigating in a context: $\phi_c(a, A)$ and $R_c(g)$. These functions imply that the agent could cluster $\phi_c(a, A)$ and $R_c(g)$ jointly, or it could cluster them independently, such that it learns the popularity of each marginalizing over the other. Formally, the *independent clustering* agent assigns each context c into two clusters via Bayesian inference as in [Eq. 3.1] and using the CRP prior for each cluster [Eq. 3.2]. The likelihood function for the two assignments are the mapping and reward functions

$$L(\mathcal{D} | c \in k_\phi) = \phi_k(a, A) \tag{3.3}$$

and

$$L(\mathcal{D} | c \in k_r) = R_k(g), \tag{3.4}$$

respectively. Conversely, the *joint clustering* agent assigns each context c into a single cluster k via Bayesian inference [Eq. 2.4] and, like the independent clustering agent, uses the CRP as the prior over assignments [Eq. 2.5]. The likelihood function for the context-cluster assignment is the product of the mapping and reward functions

$$L(\mathcal{D} | c \in k) = \phi_k(a, A) R_k(g) \tag{3.5}$$

The joint clustering agent is highly similar to previous non-compositional models of task structure learning and generalization, which have previously been shown to account for human behavior (but without specifically assessing the compositionality issue) (Collins and Frank, 2013, 2016b)).

Planning was accomplished via value iteration of the state-action value function, defined here as

$$Q_k(x, A) = R_k(x) + \gamma \sum_{x' \in X} T(x, A, x')V_k(x') \tag{3.6}$$

where $V_k(x)$ is the value of locations, $T(x, A, x')$ is the known transition function and $R_k(g)$ is the location-based reward function. A location-based reward function $R_k(x)$ is used for the purposes of planning and is created by projecting the goal values of $R_g(x)$ into the locations of the goals in each trial. This is done for convenience and assumes the agents know the locations of the goals, information explicitly provided to the subjects on each trial.

Each function is estimated via maximum likelihood estimation. On each time-step, the agent was given supervised feedback of the selected cardinal movement (as this is visible to human subjects via an animation) and binary reward feedback when it reaches a goal. As a simplifying assumption, the mapping function for each cluster was estimated twice as

$$\Pr(A_i|a_j, k) = \frac{\sum \mathbb{1}_{A=A_i} \mathbb{1}_{a=a_j}}{\sum \mathbb{1}_{a=a_j}} \tag{3.7}$$

and

$$\Pr(a_j|A_i, k) = \frac{\sum \mathbb{1}_{A=A_i} \mathbb{1}_{a=a_j}}{\sum \mathbb{1}_{A=A_i}} \tag{3.8}$$

$\Pr(A_i|a_j, k)$ was used as the estimator for inference [eqns 3.3 and 3.5] while $\Pr(a_j|A_i, k)$ was used for the purpose of sampling actions.

A softmax function was used to create a policy over cardinal movements from the state action value function:

$$\Pr(A = A_i|x, k) = \frac{\exp(\beta Q_k(x, A_i))}{\sum_j \exp(\beta Q_k(x, A_j))} \tag{3.9}$$

Actions were selected by sampling from the conditional mapping distribution $\Pr(a|A = A_i, k)$.

Statistical Analyses

Analysis of training data for experiments 1, 2 and 3 (Sections 3.3, 3.4 and 3.5)

Training performance was assessed using a combination of Bayesian general linear models that each accuracy, reaction time, or navigation efficiency, which we define as the number of responses in excess of the minimum path length to the goal chosen in a given trial. Accuracy was assessed with a logistic regression, defined:

$$\text{logit}(\theta) = \beta_{subj} + \beta_t x_t + \beta_r x_r \tag{3.10}$$

$$y \sim \text{Bernoulli}(\theta) \tag{3.11}$$

where $\text{logit}(\theta) = \log(\theta/(1\text{-}\theta))$, β_{subj} is hierarchically determined, subject specific bias term distributed normally with mean μ_b and standard deviation σ_b (Kruschke, 2015), x_t is the number of trials within a context, and where x_r is a Boolean vector corresponding to whether the previous trial shares the same context as the current trials. Uninformative priors were used for the regression parameters and group mean β_{subj} and a wide, half-cauchy prior was used for model error and the variance over the β_{subj} (Gelman, 2006).

Reaction time, expressed in frequency space was assumed to be normally distributed (Noorani and Carpenter, 2016) and was fit using parameters defined equivalently for eqn. 3.10:

$$\frac{1}{\text{RT}} = \beta_{subj} + \beta_t x_t + \beta_r x_r \tag{3.12}$$

Navigation efficiency, defined as the number of steps over and above the minimum path length from the subjects starting location on the grid and the eventual goal chosen on each trial was fit using a similar linear model that also shares all parameters with eqn. 3.10 but using a Poisson link function:

$$\log \lambda = \beta_{subj} + \beta_t x_t + \beta_r x_r \tag{3.13}$$

$$y \sim \text{Poisson}(\lambda) \tag{3.14}$$

In all three analyses, the parameter β_t was of interest. A positive value of β_t for accuracy and reaction time indicate greater accuracy and faster reaction times, respectively, as a function of time. A negative value for the value of β_t for equation 3.13 reflects more efficient paths as a function of time.

Bayesian Analysis of test context accuracy for experiment 1 (Section 3.3)

Test context accuracy was analyzed using a logistic linear model similar to the model defined in equation 3.10, with terms equivalently defined for a subject specific bias term, trials in a context and sequential repeats of a context. Comparisons of interest were defined as contrast vectors. $x_{1>2/4}$ defined the comparison test context 1 - (test context 2 + test context 4), $x_{3>2/4}$ defined the comparison test context 3 - (test context 2 + test context 4), $x_{2>4}$ defined the comparison test context 2 - test context 4. These contrasts span the space of potential comparisons between tests contexts, facilitating arbitrary comparisons through linear combinations. The full model is defined here:

$$\text{logit}(\theta) = \beta_{subj} + \beta_t x_t + \beta_r x_r + \beta_{1>2/4} x_{1>2/4} + \beta_{3>2/4} x_{3>2/4} + \beta_{2>4} x_{2>4} \tag{3.15}$$

The full model was compared to a reduced null model lacking the contrast predictors to assess whether the addition of context contrasts improves model fit. A failure of contrasts predictors to increase model fit can be interpreted as a lack of evidence for a difference between test contrast accuracy, and a null result for popularity based clustering overall. We used WAIC as the model selection criterion, with lower values of WAIC indicative of better model fit (Watanabe, 2010).

Bayesian Analysis of test context accuracy for experiment 2 (Section 3.4)

As in section 3.3, test context accuracy was using a logistic linear model. Equivalent terms for a subject specific bias, trials in a context and sequential repeats of a context were included as nuisance parameters. Comparisons of interest were defined as contrast vectors. $x_{1>2}$ defined the comparison test context 1 - test context 2, $x_{1/2>3/4}$ defined the comparison (test context 1 + test context 2) - (test context 3 + test context 4), $x_{3>4}$ defined the comparison test context 3 - test context 4. These contrastsspan the space of potential comparisons between tests contexts, facilitating arbitrary comparisons through linear combinations. The full model is defined here:

$$\text{logit}(\theta) = \beta_{subj} + \beta_t x_t + \beta_r x_r + \beta_{1>2} x_{1>2} + \beta_{1/2>3/4} x_{1/2>3/4} + \beta_{3>4} x_{3>4} \tag{3.16}$$

WAIC was used to compare the model to a reduced null model lacking the contrast predictors to assess whether the addition of context contrasts improves model fit.

Bayesian Analysis of goal choice probability for Experiment 2 (Section 3.4)

The probability of choosing goal $g \in \{A, B, C, D\}$ in the first trial of a new context was modeled independently for each goal as a Bernoulli random variable with probability θ_g. To assess whether goal choice probability varied between goals, this model was compared to a reduced, null model in which goal choice probability was fixed at $\theta_i = \theta_j \ \ \forall\ i, j \in \{A, B, C, D\}$, which is equivalent to a chance model.

In order to assess whether goal choice probability varied as a function of mapping, the previous model was compared to a model in which $\theta_{g,m}$ were allowed to vary independently, where $\theta_{g,m} = \Pr(g|m),\ \forall\ g \in \{A, B, C, D\},\ m \in \{\text{low popularity, high popularity}\}$. Estimates of $\theta_{g,m}$ were carried through for follow up analysis to examine *a priori* predictions.

Bayesian Analysis of accuracy for Experiment 3 (Section 3.5)

In experiment 3, Bayesian logistic regression was used to assess model predictions. As before, a hierarchically determined subject specific bias term β_{subj}, a term for the number of trials within a context x_t and a Boolean variable for whether the trail featured a sequential repeat of the same context x_r were included as nuisance variables.

Predictors of interest were the between subjects variable corresponding to the test design x_{test} and whether the correct goal in a test context was the high popularity goal, goal A, $x_{A_{test}}$. Because the between subjects design causes subject specific bias terms to be correlated with the manipulation of interest, we included the subjects' performance in the training contexts in the analysis as well. This de-correlates the bias term with the manipulation of interest. We further included training contexts as a predictor x_{train} and the identity of the goal during training $x_{A_{train}}$. We further compared our regression model to a null model that did not include any of our predictors of interest using model selection.

Comparison of Task difficulty

We performed pairwise comparisons the training data for experiments 1 vs 2 and 1 vs 3 using Bayesian logistic regression. The training data for each context was equated for the number of trials across both experiments within a single comparison. As

before, a term for the number of trials within a context x_t and a Boolean variable for whether the trail featured a sequential repeat of the same context x_r were included as nuisance variables. A hierarchically determined subject specific bias term β_{subj} was assumed to be normally distributed with mean fixed at 0 and a standard deviation estimated from the data. The comparison of interest was modeled with a Boolean predictor variable and associated coefficient $\beta_e xp$ that coded for experiment 1, with a non-zero coefficient reflecting a difference in training accuracies between the tasks.

Chapter 4

Conclusions

In this dissertation, I have expanded extant theory of context-based clustering to account for a basic form of compositionality and shown that predictions of this model are able to capture aspects of human subject behavior unaccounted for by previous models of generalization. This work is motivated by a desire to build more general theories of cognition. While the fronto-striatal learning systems thought to underlie context-popularity based generalization (Collins and Frank, 2013; Collins et al., 2014) are thought to be capable of complex behavior (Kriete et al., 2013; Donoso et al., 2014), it is an open question if these learning systems are sufficiently flexible to serve as the basis as a model of cognition. One of the limitations of the system, which I have attempted to address here at a computational level, is compositionality. This characteristic is important for the ability of recombining knowledge in novel ways to solve novel problems, a critical feature of human behavior (Fodor and Pylyshyn 1988; but see Johnson 2004). Nonetheless, while the studies presented in Chapter 3 suggest that humans are capable of generalizing compositionally, they do not provide evidence that this occurs using fronto-striatal circuitry. Furthermore, as the models presented in chapters 2 and 3 are at a computational level, it remains unclear how fronto-striatal circuitry could be implement compositional representations.

Theoretically, this is a non-trivial problem. Learned values can be represented easily with vectors as can the linear mapping functions, but the question of how to combine a representation into a multi-step policy is non-trivial. The models presented here rely on dynamic programming to solve a value function; it is not clear how

dynamic programming could be plausibly implemented in a neural system. More generally, the question of combining compositional representations into a policy is a planning problem. Planning problems are computationally demanding and it is not known how people can solve planning problems, though several possibilities have been raised, including tree-search and pruning algorithms (Huys et al., 2012), the use of hierarchical policies (Solway et al., 2014) or via a successor representation (Russek et al., 2017; Momennejad et al., 2017; Stachenfeld et al., 2017). In the present work, I have avoided difficult planning problems in experiment designs in an attempt to minimize the difference between the model's planning algorithm and a human's.

A further limitation of the studies presented here is how contexts are inferred. For the purpose of modeling, we assumed the model has veridical access to the current contexts and in the human studies, subjects were cued unambiguously. In real-world settings, this is likely to be a much more difficult problem. One possibility is that humans infer context through the internal consistency of their structure (Zacks and Tversky, 2001). Alternatively, contexts may be constructed more simply through perceptual similarity. It is important to note that here we have assumed that context is a component of a state variable that is distinct from the rest of the state-space in that it defines some set of rules to follow. It is possible that humans do not make this assumption and are treating the provided context as an equivalent part of the state space and are nonetheless able to use it as a unit of generalization.

A further non-trivial limitation is the choice of components in compositional learning. Here, I have introduced a reduced version of the reward function (goals) and transition function (mappings) of an MDP as a natural choice of units of information. However, humans may be better described by different subunits. It is unclear if a 'goal' as I have represented it is different than an 'option' in a hierarchical framework and it may easily be the case that a policy operates at a higher level (for example, collect as many points as possible). Substituting that policy could change the behavior (for example, get exactly 100 points) and the 'goals' learned by the subject could be reused as a different type of information. Moreover, it is unlikely that a mapping as described here is the atom of movement. We might expect to see repeated motor plans or skills as compositional units.

Future work will focus on addressing these theoretical challenges and probe what

are the neurobiological foundations of this form of compositional behavior in humans. Nonetheless, I would argue that I have provided a normative basis for compositionality. Even as parts of policy space are correlated, there is often a benefit to ignoring this correlation structure in generalization. Ignoring this correlation structure requires a compositional representation whereas strictly assuming a correlation structure can be represented compositionally or non-compositionally. This feature provides the basis for additional flexibility in human and animal behavior.

Appendix A

Cluster analysis exclusion criteria

In our online experiments, we were concerned some subjects would not engage with the task and instead rush through it as quickly as possible. In order to control for such a possibility, we used subjects relative performance on the training contexts as a control. Specifically, we looked at the accuracy on trials in which the preceding trial shared the same context and the subject was correct. All versions of our online task used a deterministic reward schedule. As a consequence, if our subjects understand the rule structure, they can perform close to 100% in these trials (on a small percentage of trials, it is not possible to reach the correct goal due to the random placement of objects in the grid).

As a second measure of accuracy, we conducted a binomial test on all other training trials against a chance. These two measures of accuracy were used as the basis of a cluster analysis. A Gaussian mixture model were fit to the data, we allowed the number of clusters to range between one and three and allowed for the and allowed for constraints over the covariance matrix to vary from spherical, diagonal, tied or full. Bayesian information criterion (BIC) was used for model selection, and the model with the lowest BIC was chosen. For the experiments presented in sections 3.3 and 3.4, a model with 2 components and a diagonal constraint on the covariance matrix had the lowest BIC (Figure A.1)

The members of the largest cluster were carried forward for analysis and all other subjects were used excluded. Follow up analyses found the included subjects spent significantly more time viewing the instructions, (experiment 1: $t(54.1) = 4.58, p <$

3×10^{-5}, cohen's d = 4.65; experiment 2: $t(67.5) = 4.03, p < 0.0002$, cohen's d = 4.09 and were significantly more likely to choose the goal closet to the initial location of the agent at the start of a trial (experiment 1: $t(34.3) = 6.61, p < 2 \times 10^{-7}$, cohen's d = 6.71; experiment 3: $t(47.3) = 5.51, p < 2 \times 10^{-6}$, cohen's d = 5.59) (Figure A.2), suggesting the excluded subjects were less engaged in the task.

A similar clustering procedure was used to determine the subjects in experiment 3 that were carried through for further analysis but with loosened criteria as a large number of subjects were found to be excluded using the criteria above. This was likely due task differences as experiment three was substantially easier that the other two tasks. Accuracy in trials in which the context was sequentially repeated and in which the previous trial was correct was again used as a dimension of the clustering analysis. In lieu of the binomial probability that all other trials were at chance accuracy, the raw accuracy in these trials was used. Chance accuracy in experiment 3 was 50%, and this was found to be too strict of a criterion. Subjects excluded in were less likely to choose the closest goal closest to the initial location of the agent ($t(110.4) = -2.16, p < 0.033$, cohen's d = 2.18) but we saw no difference in the time spent viewing instructions ($t(156.7) = 0.86, p < 0.39$, cohen's d = 0.87).

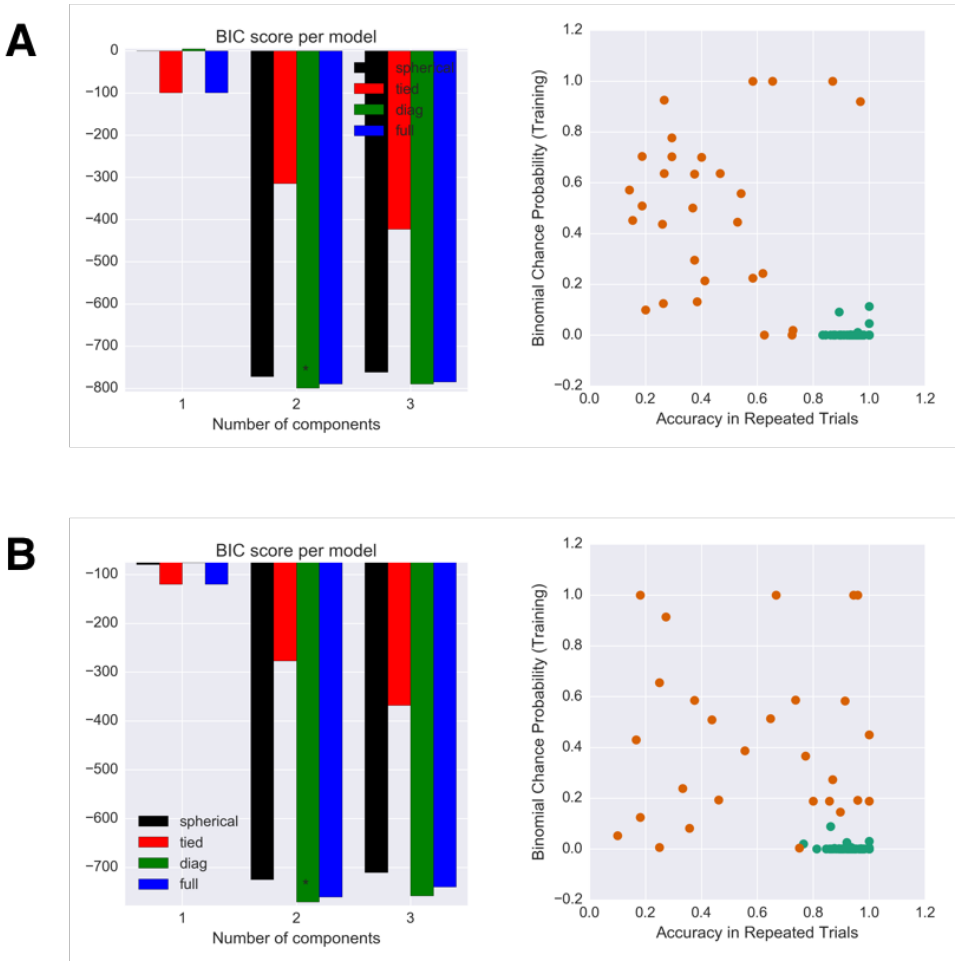

Figure A.1: Cluster analysis of subjects training performance. Panels A & B were performed on subjects collected for the experiments presented in sections 3.3 and 3.4 *A & B, Right*: Included subjects (green) were less likely to respond at chance levels and were more likely to respond correctly on trials immediately following a correct trial with the same context

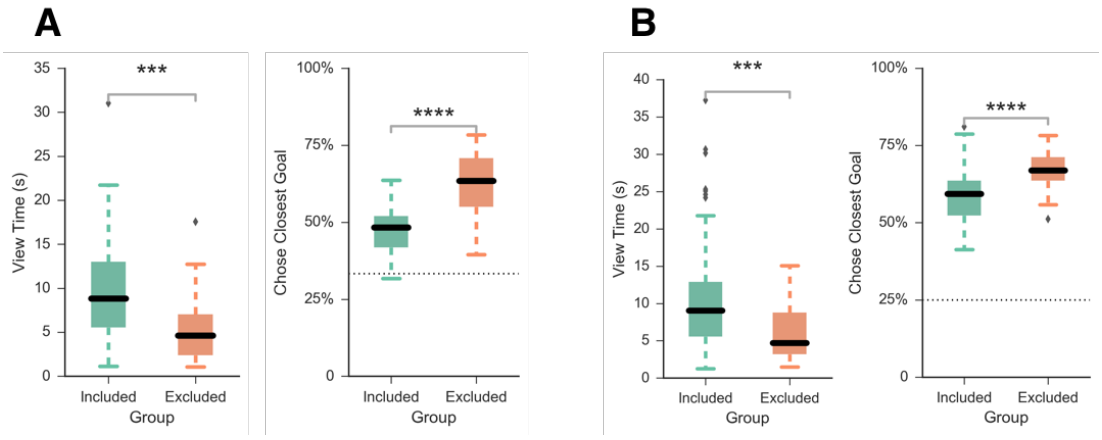

Figure A.2: Group difference in time spent viewing instructions (left, both panels) and the proportion of times subjects chose the goal closet to the initial location of the agents (right, both panels) Panels A & B were performed on subjects collected for the experiments presented in sections 3.3 and 3.4.

Appendix B

Analysis of mapping generalization

In experiments 1 and 2 (Sections 3.3 and 3.4, respectively). We have argued the finding that goal generalization appears not to necessarily depend conditionally on the observed mapping is equivalent to independent clustering. However, an alternative possibility is that subjects do not make an observation of the mapping and learn the mappings through trial and error in each trial or for each context independently. Training data appear to rule out the former: subjects respond more quickly and take more efficient paths to the goal as the experience more trials within a context, suggesting subjects learn the mappings. However, this does not rule out the possibility subject re-learn the mappings from scratch in the new test contexts.

Both joint and independent cluster predict that high popularity mappings will show a generalization benefit over low popularity mappings. In addition, both predict a difference between training and test conditions as the models are able to generalize for both learned mappings. An agent that does not generalize mappings predicts all contexts, regardless of mapping or train/test difference, will show the same performance.

In order to assess mapping generalization, a Bayesian linear model with a Poisson link function was fit to the number of excess steps taken over the minimum path length to reach the goal, defined here as:

$$\log \lambda = \beta_{subj} + \beta_t x_t + \beta_r x_r + \beta_{HP} x_{HP} + \beta_{train} x_{train} \tag{B.1}$$

$$y \sim \text{Poisson}(\lambda) \tag{B.2}$$

where β_{subj} is a hierarchically determined subject specific bias term distributed $\beta_{subj} \sim \mathcal{N}(\mu, \sigma)$, x_t is the number of trials a subject has seen a context, x_r is whether the trial was a sequential repeat of the previous context, x_{HP} is whether the mapping is the high popularity mapping and x_{train} is whether the context is a training or test context.

A similar model was fit to the mean reaction time per trial (expressed in frequency space),

$$\frac{1}{RT} = \beta_{subj} + \beta_t x_t + \beta_r x_r + \beta_{HP} x_{HP} + \beta_{train} x_{train} \tag{B.3}$$

with equivalent terms. The models make no reaction time predictions, but it is plausible to expect reaction to correlate more with mapping learning than goal learning as it reflects the motor component of the task (it is nonetheless impossible to isolate the effects of goals from mappings in reaction time).

In experiment 1, training was associated with less efficient paths (i.e. more responses needed to reach the goal; β_{train} 95% HDI = [0.014, 0.054], but little evidence to suggest the high popularity mapping was associated with more efficient navigation ($\Pr(\beta_{HP} > 0) < 0.75$ 95% HDI = [-0.010, 0.021].) Reaction times were also slower in training (RT: β_{train} 95% HDI = [-0.385, -0.319]) and there was also an effect of mapping on reaction time ($\Pr(\beta_{HP} > 0) < 0.73$ 95% HDI =[0.003, 0.053]).

For experiment 2, training was also associated with less efficient paths (β_{train} 95% HDI = [0.048, 0.089] as was the high popularity mapping ($\beta_{HP} > 0$ 95% HDI = [0.011, 0.037]). Reaction times were also slower in training (RT: β_{train} 95% HDI = [-0.590, -0.522]) but there was no meaningful effect of mapping on reaction time ($\Pr(\beta_{HP} > 0) < 0.73$ 95% HDI =[-0.029, 0.014]).

Overall, the difference in reaction time and path efficiency suggest subjects were generalizing mappings from the training contexts to the test contexts, but the effects of the high popularity mapping on both measures was not as reliable across both studies.

Bibliography

Abler, B., Walter, H., Erk, S., Kammerer, H., and Spitzer, M. (2006). Prediction error as a linear function of reward probability is coded in human nucleus accumbens. *NeuroImage*, 31(2):790–5.

Acuña, D. and Schrater, P. (2010). Structure Learning in Human Sequential Decision-Making. *PLoS computational biology*, (December).

Aldous, D. (1985). Exchangeability and related topics. In *École d'été de probabilités de Saint-Flour XIII*. Springer., Berlin.

Alexander, G. E., DeLong, M. R., and Strick, P. L. (1986). Parallel Organization of Functionally Segregated Circuits Linking Basal Ganglia and Cortex. *Annual review of neuroscience*, 9.

Austerweil, J. L., Griffiths, T. L., and Griffithsberkeleyedu, T. (2009). Analyzing human feature learning as nonparametric Bayesian inference. *Advances in Neural Information Processing Systems*, 21:1–8.

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

Badre, D. and Frank, M. J. (2012). Mechanisms of hierarchical reinforcement learning in cortico-striatal circuits 2: evidence from FMRI. *Cerebral cortex (New York, N.Y. : 1991)*, 22(3):527–36.

Badre, D., Kayser, A. S., and D'Esposito, M. (2010). Frontal cortex and the discovery of abstract action rules. *Neuron*, 66(2):315–26.

Bakker, B. (2001). Reinforcement Learning with Long Short-Term Memory. *NIPS*.

Balleine, B., Daw, N. D., and O'Doherty, J. P. (2008). Multiple forms of value learning and the function of dopamine. *Neuroeconomics: Decision Making and the Brain*.

Balleine, B. W. and Dickinson, a. (1998). Goal-directed instrumental action: contingency and incentive learning and their cortical substrates. *Neuropharmacology*, 37(4-5):407–19.

Batra, O., McGuire, J. T., and Kable, J. W. (2013). The valuation system: A coordinate-based meta-analysis of BOLD fMRI experiments examining neural correlates of subjective value. *Neuroimage*, 1(76):412–427.

Bayer, H. H. M., Lau, B., and Glimcher, P. P. W. (2007). Statistics of midbrain dopamine neuron spike trains in the awake primate. *Journal of Neurophysiology*, 98(3):1428–39.

Beeler, J. a., Frank, M. J., McDaid, J., Alexander, E., Turkson, S., Bernandez, M. S., McGehee, D. S., and Zhuang, X. (2012). A role for dopamine-mediated learning in the pathophysiology and treatment of Parkinson's disease. *Cell reports*, 2(6):1747–61.

Behrens, T. E., Woolrich, M. W., Walton, M. E., and Rushworth, M. F. S. (2007). Learning the value of information in an uncertain world. *Nature neuroscience*, 10(9):1214–21.

Bellman, R. (1957). *Dynamic Programming*. Princeton University Press.

Borsa, D., Graepel, T., and Shawe-Taylor, J. (2016). Learning Shared Representations in Multi-task Reinforcement Learning.

Botvinick, M. M., Niv, Y., and Barto, A. C. (2009). Hierarchically organized behavior and its neural foundations: a reinforcement learning perspective. *Cognition*, 113(3):262–80.

Bouton, M. E. and Bolles, R. C. (1979). Contextual control of the extinction of conditioned fear. *Learning and Motivation*, 10(4):445–466.

Bouton, M. E. and King, D. a. (1983). Contextual control of the extinction of conditioned fear: tests for the associative value of the context. *Journal of experimental psychology. Animal behavior processes*, 9(3):248–65.

Braver, T. S. and Cohen, J. D. (2000). On the control of control: The role of dopamine in regulating prefrontal function and working memory. In Monsell, S. and Driver, J., editors, *Attention and Performance XVIII*, pages 713–737. MIT Press, London.

Bunge, S. a., Kahn, I., Wallis, J. D., Miller, E. K., and Wagner, A. D. (2003). Neural circuits subserving the retrieval and maintenance of abstract rules. *Journal of neurophysiology*, 90(5):3419–28.

Campbell, M., Hoane Jr., a. J., and Hsu, F.-h. (2002). Deep Blue. *Artificial Intelligence*, 134(1-2):57–83.

Cassandra, A. R. (1998). *Exact and approximate algorithms for partially observable Markov decision processes*. PhD thesis.

Cavanagh, J. F., Figueroa, C. M., Cohen, M. X., and Frank, M. J. (2012). Frontal theta reflects uncertainty and unexpectedness during exploration and exploitation. *Cerebral cortex (New York, N.Y. : 1991)*, 22(11):2575–86.

Chomsky, N. (1968). *Language and mind*. Harcourt, Brace & World, New York.

Chrisman, L. (1992). Reinforcement learning with perceptual aliasing: The perceptual distinctions approach. *AAAI*.

Cockburn, J. (2015). *What you should do (and what you really do) when you don't know what to do: Information seeking during value based decision making*. PhD thesis, Brown University.

Cohen, J. D., McClure, S. M., and Yu, A. J. (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. Series B, Biological sciences*, 362(1481):933–42.

Collins, A. G. E., Cavanagh, J. F., and Frank, M. J. (2014). Human EEG Uncovers Latent Generalizable Rule Structure during Learning. *The Journal of neuroscience*, 34(13):4677–85.

Collins, A. G. E. and Frank, M. J. (2013). Cognitive control over learning: creating, clustering, and generalizing task-set structure. *Psychological review*, 120(1):190–229.

Collins, A. G. E. and Frank, M. J. (2014). Opponent Actor Learning (OpAL): modeling interactive effects of striatal dopamine on reinforcement learning and choice incentive. *Pyschological Review*, 121(3):337–366.

Collins, A. G. E. and Frank, M. J. (2016a). Motor Demands Constrain Cognitive Rule Structures. *PLoS Computational Biology*, 12(3):1–18.

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

Cornwell, B. R., Johnson, L. L., Holroyd, T., Carver, F. W., and Grillon, C. (2008). Human hippocampal and parahippocampal theta during goal-directed spatial navigation predicts performance on a virtual Morris water maze. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 28(23):5983–90.

Crick, F. (1989). The recent excitement about neural networks. *Nature*.

D'Ardenne, K., Eshel, N., Luka, J., Lenartowicz, A., Nystrom, L. E., and Cohen, J. D. (2012). Role of prefrontal cortex and the midbrain dopamine system in working memory updating. *Proceedings of the ....*

Daw, N. D. (2012). Model-Based Reinforcement Learning as Cognitive Search: Neurocomputational Theories. *Cognitive Search: Evolution Algorithms and the Brain*.

Daw, N. D. and Doya, K. (2006). The computational neurobiology of learning and reward. *Current opinion in neurobiology*, 16(2):199–204.

Daw, N. D., Gershman, S. J., Seymour, B., Dayan, P., and Dolan, R. J. (2011). Model-based influences on humans' choices and striatal prediction errors. *Neuron*, 69(6):1204–15.

Dayan, P. and Balleine, B. W. (2002). Reward, motivation, and reinforcement learning. *Neuron*, 36:285–298.

Dayan, P. and Niv, Y. (2008). Reinforcement learning: the good, the bad and the ugly. *Current opinion in neurobiology*, 18(2):185–96.

Dayan, P. and Yu, A. J. (2003). Uncertainty and learning. *IETE Journal of Research*, 49:171–181.

D'Esposito, M., Detre, J. A., Alsop, D. C., Shin, R. K., Atlas, S., and Grossman, M. (1995). The neural basis of the central executive system of working memory. *Nature*.

Devan, B. D., Goad, E. H., and Petri, H. L. (1996). Dissociation of hippocampal and striatal contributions to spatial navigation in the water maze. *Neurobiology of learning and memory*, 66(3):305–23.

DiCarlo, J. J. and Cox, D. D. (2007). Untangling invariant object recognition. *Trends in cognitive sciences*, 11(8):333–41.

Diuk, C., Cohen, A., and Littman, M. L. (2000). An Object-Oriented Representation for Efficient Reinforcement Learning.

Donoso, M., Collins, A. G. E., and Koechlin, E. (2014). Human cognition. Foundations of human reasoning in the prefrontal cortex. *Science (New York, N.Y.)*, 344(6191):1481–6.

Doshi-Velez, F. (2009). The infinite partially observable Markov decision process. *Neural Information Processing Systems*, pages 1–10.

Doya, K. (1999). What are the Computations of the Cerebellum, the Basal Gangila, and the Cerebral Cortex? *Neural Networks*, 12(7):961–974.

Ekstrom, A., Kahana, M., Caplan, J., Fields, T., EA, I., Newman, E., and Fried, I. (2003). Cellular networks underlying human spatial navigation. *Nature*, 425(September).

Elman, J. (1990). Finding structure in time. *Cognitive Science*, 14(2):179–211.

Fiser, J., Berkes, P., Orbán, G., and Lengyel, M. (2010). Statistically optimal perception and learning: from behavior to neural representations. *Trends in cognitive sciences*, 14(3):119–30.

Fodor, J. and Pylyshyn, Z. (1988). Connectionism and cognitive architecture: A critical analysis. *Cognition*.

Frank, M. J. (2005). Dynamic dopamine modulation in the basal ganglia: a neurocomputational account of cognitive deficits in medicated and nonmedicated Parkinsonism. *Journal of cognitive neuroscience*, 17(1):51–72.

Frank, M. J. and Badre, D. (2012). Mechanisms of hierarchical reinforcement learning in corticostriatal circuits 1: computational analysis. *Cerebral cortex (New York, N.Y. : 1991)*, 22(3):509–26.

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

Frank, M. J., Seeberger, L. C., and O'reilly, R. C. (2004). By carrot or by stick: cognitive reinforcement learning in parkinsonism. *Science (New York, N.Y.)*, 306(5703):1940–3.

Franklin, N. T. and Frank, M. J. (2015). A cholinergic feedback circuit to regulate striatal population uncertainty and optimize reinforcement learning. *eLife*, 10.7554/eL:1–29.

Friedman, J. H. (1997). On Bias , Variance , 0 / 1 Loss , and the Curse-of-Dimensionality. 77:55–56.

Fusi, S., Miller, E. K., and Rigotti, M. (2016). Why neurons mix: High dimensionality for higher cognition.

Gelman, A. (2006). Prior distribution for variance parameters in hierarchical models. *Bayesian Analysis*, 1(3):515–533.

Geman, S. and Doursat, R. (1992). Neural Networks and the Bias-Variance Dilemma. *Neural computation*, 58.

Gershman, S. J., Blei, D. M., and Niv, Y. (2010). Context, learning, and extinction. *Psychological review*, 117(1):197–209.

Gershman, S. J. and Niv, Y. (2010). Learning latent structure : carving nature at its joints. *Current opinion in neurobiology*.

Gershman, S. J. and Niv, Y. (2013). Perceptual estimation obeys Occam's razor. *Frontiers in Psychology*, 4(SEP):1–11.

Gittins, J. (1979). Bandit Processes and Dynamic Allocation Indicies. *Journal of the Royal Statistical Society.*, 41(2):148–177.

Gläscher, J., Daw, N. D., Dayan, P., and O'Doherty, J. P. (2010). States versus rewards: dissociable neural prediction error signals underlying model-based and model-free reinforcement learning. *Neuron*, 66(4):585–95.

Goodman, N. D., Roy, D. M., Leslie, P., Tenenbaum, J. B., Policy, B., Wingate, D., Goodman, N. D., Roy, D. M., Kaelbling, L. P., and Tenenbaum, J. B. (2011). Bayesian Policy Search with Policy Priors .

Green, C. S., Benson, C., Kersten, D., and Schrater, P. (2010). Alterations in choice behavior by manipulations of world model. *Proceedings of the National Academy of Sciences of the United States of America*, 107(37):16401–6.

Griffiths, T. L., Chater, N., Kemp, C., Perfors, A., and Tenenbaum, J. B. (2010). Probabilistic models of cognition: exploring representations and inductive biases. *Trends in cognitive sciences*, 14(8):357–64.

Gureckis, T. M. and Love, B. C. (2009a). Learning in Noise: Dynamic Decision-Making in a Variable Environment. *Journal of mathematical psychology*, 53(3):180–193.

Gureckis, T. M. and Love, B. C. (2009b). Short-term gains, long-term pains: how cues about state aid learning in dynamic environments. *Cognition*, 113(3):293–313.

Hamid, A. A., Pettibone, J. R., Mabrouk, O. S., Hetrick, V. L., Schmidt, R., Vander Weele, C. M., Kennedy, R. T., Aragona, B. J., and Berke, J. D. (2016). Mesolimbic dopamine signals the value of work. *Nature Neuroscience*, 19(1):117–126.

Hazy, T. E., Frank, M. J., and O'reilly, R. C. (2007). Towards an executive without a homunculus: computational models of the prefrontal cortex/basal ganglia system. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 362(1485):1601–13.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*.

Huys, Q. J. M., Eshel, N., O'Nions, E., Sheridan, L., Dayan, P., and 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.

James, W. (1890). The principals of psychology.

Johnson, K. (2004). On the systematicity of langague and thought. *The Journal of Philosophy*, 101(3):111–139.

Kaelbling, L., Littman, M., and Moore, A. (1996). Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285.

Kaelbling, L. P., Littman, M. L., and Cassandra, A. R. (1998). Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101(1-2):99–134.

Kahneman, D. and Tversky, A. (1984). Choices, values, and frames. *American psychologist*.

Kansky, K., Silver, T., Mely, D. A., Ekdawy, M., Lazaro-Gredilla, M., Lou, X., Dorfman, N., Szymon, S., Phoenix, S., and George, D. (2017). Schema Networks: Zero-shot Transfer with a Generative Causal Model of Intuitive Physics. *International Conference on Machine Learning*.

Knox, W. B., Otto, a. R., Stone, P., and Love, B. C. (2011). The nature of belief-directed exploratory choice in human decision-making. *Frontiers in psychology*, 2(January):398.

Konidaris, G. (2016). Constructing Abstraction Hierarchies Using a Skill-Symbol Loop. In *IJCAI International Joint Conference on Artificial Intelligence*, pages 1648–1654.

Konidaris, G. and Barto, A. (2007). Building portable options: Skill transfer in reinforcement learning. *IJCAI International Joint Conference on Artificial Intelligence*, pages 895–900.

Kool, W., Cushman, F. A., and Gershman, S. J. (2016). Reinforcement Learning Trade-offs. *PLoS Computational Biology*.

Kravitz, A. V., Freeze, B. S., Parker, P. R. L., Kay, K., Thwin, M. T., Deisseroth, K., and Kreitzer, A. C. (2010). Regulation of parkinsonian motor behaviours by optogenetic control of basal ganglia circuitry. *Nature*, 466(7306):622–6.

Kravitz, A. V., Tye, L. D., and Kreitzer, A. C. (2012). Distinct roles for direct and indirect pathway striatal neurons in reinforcement. *Nature neuroscience*, 15(6):816–8.

Kriete, T., Noelle, D. C., Cohen, J. D., and O'Reilly, R. C. (2013). Indirection and symbol-like processing in the prefrontal cortex and basal ganglia. *Proceedings of the National Academy of Sciences of the United States of America*, 110(41):16390–16395.

Kruschke, J. K. (2015). *Doing Bayesain data anlaysis: A tutorial with R, JAGS and Stan*. Academic Press, 2nd edition.

Kulkarni, T. D., Saeedi, A., Gautam, S., and Gershman, S. J. (2016). Deep Successor Reinforcement Learning. *arXiv*.

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., and Gershman, S. J. (2016). Building Machines That Learn and Think Like People. (c):1–89.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.

Leffler, B. R., Littman, M. L., and Edmunds, T. (2007). Efficient reinforcement learning with relocatable action models. *Proceedings of the 22nd AAAI Conference on Artificial Intelligence*, 1(2005):572–577.

Littman, M. (1996). *Algorithms for sequential decision making*. PhD thesis.

Mahmud, M. M. H., Hawasly, M., Rosman, B., and Ramamoorthy, S. (2013). Clustering Markov Decision Processes For Continual Transfer.

Maia, T. V. and Frank, M. J. (2011). From reinforcement learning models to psychiatric and neurological disorders. *Nature neuroscience*, 14(2):154–62.

Marcus, G. F., Marblestone, A., and Dean, T. (2014). Frequently Asked Questions for: The Atoms of Neural Computation. *arXiv preprint*, pages 1–12.

McClure, S. M., Berns, G. S., and Montague, P. R. (2003a). Temporal prediction errors in a passive learning task activate human striatum. *Neuron*, 38(2):339–46.

McClure, S. M., Daw, N. D., and Read Montague, P. (2003b). A computational substrate for incentive salience. *Trends in Neurosciences*, 26(8):423–428.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M. (2013). Playing Atari with Deep Reinforcement Learning. pages 1–9.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., and Hassabis, D. (2015). Human-level control through deep reinforcement learning. *Nature*.

Momennejad, I., Russek, E. M., Cheong, J. H., Botvinick, M. M., Daw, N., and Gershman, S. J. (2017). The successor representation in human reinforcement learning. *bioRxiv*.

Montague, P. R., Dayan, P., and Sejnowski, T. J. (1996). A Framework for Mesencephalic Predictive Hebbian Learning. *The Journal of Neuroscience*, 76(5):1936–1947.

Mushiake, H., Saito, N., Sakamoto, K., Itoyama, Y., and Tanji, J. (2006). Activity in the lateral prefrontal cortex reflects multiple steps of future events in action plans. *Neuron*, 50(4):631–41.

Nassar, M. R., Wilson, R. C., Heasly, B., and Gold, J. I. (2010). An approximately Bayesian delta-rule model explains the dynamics of belief updating in a changing environment. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 30(37):12366–78.

Newell, A., Rosenbloom, P., and Laird, J. (1989). Symbolic architectures for cognition. In *Foundations of Cognitive Science*.

Newell, A. and Simon, H. A. (1959). The Simulation of Human Thought.

Noorani, I. and Carpenter, R. H. S. (2016). The LATER model of reaction time and decision. *Neuroscience and Biobehavioral Reviews*, 64:229–251.

O'Doherty, J. P., Dayan, P., Friston, K., Critchley, H., and Dolan, R. J. (2003). Temporal difference models and reward-related learning in the human brain. *Neuron*, 38(2):329–37.

O'Reilly, R. C. and Frank, M. J. (2006). Making working memory work: a computational model of learning in the prefrontal cortex and basal ganglia. *Neural computation*, 18(2):283–328.

Pagnoni, G., Zink, C. F., Montague, P. R., and Berns, G. S. (2002). Activity in human ventral striatum locked to errors of reward prediction. *Nature neuroscience*, 5(2):97–8.

Payzan-LeNestour, E. and Bossaerts, P. (2011). Risk, unexpected uncertainty, and estimation uncertainty: Bayesian learning in unstable settings. *PLoS computational biology*, 7(1).

Peshkin, L., Meuleau, N., and Kaelbling, L. (1999). Learning policies with external memory. *Sixteenth International Converence on Machine Learning*, pages 307–314.

Pinker, S. (2001). Four decades of rules and associations, or whatever happened to the past tense debate. In *Language, the brain, and cognitive development: Papers in honor of Jacques Mehler*, pages 157–179.

Plate, T. A. (1995). Holographic Reduced Representations. *IEEE Transactions on Neural Networks*, 6(3):623–641.

Retailleau, A., Etienne, S., Guthrie, M., and Boraud, T. (2012). Where is my reward and how do I get it? Interaction between the hippocampus and the basal ganglia during spatial learning. *Journal of physiology, Paris*, 106(3-4):72–80.

Rogers, T. T. and McClelland, J. L. (2004). *Semantic cognition: A parallel distributed processing approach.* MIT Press, Cambridge, MA.

Rosman, B., Hawasly, M., and Ramamoorthy, S. (2016). Bayesian policy reuse. *Machine Learning*, 104(1):99–127.

Ross, S., Pineau, J., Paquet, S., and Chaib-Draa, B. (2008). Online Planning Algorithms for POMDPs. *The journal of artificial intelligence research*, 32(2):663–704.

Rougier, N. P., Noelle, D. C., Braver, T. S., Cohen, J. D., and O'Reilly, R. C. (2005). Prefrontal cortex and flexible cognitive control: rules without symbols. *Proceedings of the National Academy of Sciences of the United States of America*, 102(20):7338–43.

Rumelhart, D. E. and McClelland, J. L. (1985). *On learning the past tenses of English verbs.* Institute for Cognitive Science, University of California, San Diego.

Rumelhart, D. E. and McClelland, J. L. (1986). *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, volume 1.

Russek, E. M., Momennejad, I., Botvinick, M. M., Gershman, S. J., and Daw, N. D. (2017). Predictive representations can link model-based reinforcement learning to model-free mechanisms. *bioRxiv*.

Sanborn, A. N., Griffiths, T. L., and Navarro, D. J. (2006). A More Rational Model of Categorization. *Proceedings of the 28th Annual Conference of the Cognitive Science Society*, pages 1–6.

Sanborn, A. N., Griffiths, T. L., and Navarro, D. J. (2010). Rational approximations to rational models : Alternative algorithms for category learning. *Psychological review*, 117(4):1144–1167.

Schölkopf, B. (2015). Artificial intelligence: Learning to see and act. *Nature*, 518(7540):486–487.

Schuck, N. W., Cai, M. B., Wilson, R. C., and Niv, Y. (2016). Human Orbitofrontal Cortex Represents a Cognitive Map of State Space. *Neuron*, 91(6):1402–1412.

Schultz, W. (1998). Predictive Reward Signal of Dopamine Neurons Predictive Reward Signal of Dopamine Neurons. *Journal of Neurophysiology*, pages 1–27.

Schultz, W. (2010). Dopamine signals for reward value and risk: basic and recent data. *Behavioral and brain functions : BBF*, 6:24.

Shafto, P., Kemp, C., Mansinghka, V., and Tenenbaum, J. B. (2011). A probabilistic model of cross-categorization. *Cognition*, 120(1):1–25.

Shen, W., Flajolet, M., Greengard, P., and Surmeier, D. J. (2008). Dichotomous dopaminergic control of striatal synaptic plasticity. *Science (New York, N.Y.)*, 321(5890):848–51.

Siegelmann, H. and Sontag, E. (1995). On the Computational Power of Neural Nets.

Siegelmann, H. T. and Sontag, E. D. (1991). Turing Computability with Neural Nets. *Applied Mathematics Letters*, 4(6):77–80.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., and Hassabis, D. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489.

Sloman, S. a. (1996). The empirical case for two systems of reasoning. *Psychological Bulletin*, 119(1):3–22.

Smolensky, P. (1990). Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial intelligence*, 46(1-2):159–216.

Solway, A., Diuk, C., Córdova, N., Yee, D., Barto, A. G., Niv, Y., and Botvinick, M. M. (2014). Optimal Behavioral Hierarchy. *PLoS Computational Biology*, 10(8):e1003779.

Stachenfeld, K. L., Botvinick, M. M., and Gershman, S. J. (2017). The hippocampus as a predictive map. *bioRxiv*.

Stalnaker, T. A., Berg, B., Aujla, N., and Schoenbaum, G. (2016). Cholinergic Interneurons Use Orbitofrontal Input to Track Beliefs about Current State. *J. Neurosci.*, 36(23):6242–6257.

Starkweather, C. K., Babayan, B. M., Uchida, N., and Gershman, S. J. (2017). Dopamine reward prediction errors reflect hidden-state inference across time. *Nature Neuroscience*, 20(4):581–589.

Summerfield, C., Behrens, T. E., and Koechlin, E. (2011). Perceptual classification in a rapidly changing environment. *Neuron*, 71(4):725–36.

Surmeier, D. J., Carrillo-Reid, L., and Bargas, J. (2011). Dopaminergic modulation of striatal neurons, circuits, and assemblies. *Neuroscience*, 198:3–18.

Sutton, R. S. and Barto, A. G. (1998). *Reinforcement Learning: An Introduction*. MIT Press, Cambridge.

Sutton, R. S., Precup, D., and Singh, S. (1999). Between MDPs and Semi-MDPs: A Framework for Temporal Abstraction in Reinforcement Learning. *Artificial Intelligence*, 122(1-2):181–211.

Tenenbaum, J. B., Kemp, C., Griffiths, T. L., and Goodman, N. D. (2011). How to grow a mind: statistics, structure, and abstraction. *Science (New York, N.Y.)*, 331(6022):1279–85.

Todd, M., Niv, Y., and Cohen, J. D. (2008). Learning to use working memory in partially observable environments through dopaminergic reinforcement. *Advances in neural information processing systems*.

Tsai, H.-C., Zhang, F., Adamantidis, A., Stuber, G. D., Bonci, A., de Lecea, L., and Deisseroth, K. (2009). Phasic firing in dopaminergic neurons is sufficient for behavioral conditioning. *Science (New York, N.Y.)*, 324(5930):1080–4.

Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, 59(236):433–460.

Wallis, J. D., Anderson, K. C., and Miller, E. K. (2001). Single neurons in prefrontal cortex encode abstract rules. *Nature*, 411(6840):953–6.

Watanabe, S. (2010). Asymptotic Equivalence of Bayes Cross Validation and Widely Applicable Information Criterion in Singular Learning Theory. *Journal of Machine Learning Research*, 11:3571–3594.

Watkins, C. J. C. H. and Dayan, P. (1992). Q-learning. *Machine Learning*, 8(3-4):279–292.

Werchan, D. M., Collins, A. G., Frank, M. J., and Amso, D. (2016). Role of Prefrontal Cortex in Learning and Generalizing Hierarchical Rules in 8-Month-Old Infants. *The Journal of Neuroscience*, 36(40):10314–10322.

Wiecki, T. V., Riedinger, K., von Ameln-Mayerhofer, A., Schmidt, W. J., and Frank, M. J. (2009). A neurocomputational account of catalepsy sensitization induced by D2 receptor blockade in rats: context dependency, extinction, and renewal. *Psychopharmacology*, 204:265–277.

Wilson, A., Fern, A., and Tadepalli, P. (2012). Transfer Learning in Sequential Decision Problems: A Hierarchical Bayesian Approach. In *ICML Unsupervised and Transfer Learning*, pages 217–227.

Wilson, R. and Finkel, L. (2009). A neural implementation of the Kalman filter. *Advances in neural information ...*, 22:2062–2070.

Wilson, R. C., Geana, A., White, J. M., Ludvig, E. A., and Cohen, J. D. (2014a). Humans Use Directed and Random Exploration to Solve the ExploreExploit Dilemma. *Journal of experimental psychology: General*.

Wilson, R. C., Nassar, M. R., and Gold, J. I. (2010). Bayesian online learning of the hazard rate in change-point problems. *Neural computation*, 22(9):2452–76.

Wilson, R. C. and Niv, Y. (2011). Inferring relevance in a changing world. *Frontiers in human neuroscience*, 5(January):189.

Wilson, R. C., Takahashi, Y. K., Schoenbaum, G., and Niv, Y. (2014b). Orbitofrontal cortex as a cognitive map of task space. *Neuron*, 81(2):267–278.

Yamins, D. L. K., Hong, H., and Cadieu, C. (2013). Hierarchical Modular Optimization of Convolutional Networks Achieves Representations Similar to Macaque IT and Human Ventral Stream. *Advances in Neural Information Processing Systems*, (October):1–9.

Yoshida, W. and Ishii, S. (2006). Resolution of uncertainty in prefrontal cortex. *Neuron*, 50(5):781–9.

Yu, A. J. and Cohen, J. D. (2008). Sequential effects: superstition or rational behavior? *Advances in neural information . . .*, pages 1–8.

Yu, A. J. and Dayan, P. (2002). Acetylcholine in cortical inference. *Neural networks : the official journal of the International Neural Network Society*, 15(4-6):719–30.

Yu, A. J. and Dayan, P. (2005). Uncertainty, neuromodulation, and attention. *Neuron*, 46(4):681–92.

Yu, Y., Fitzgerald, T. H. B., and Friston, K. J. (2013). Working memory and anticipatory set modulate midbrain and putamen activity. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 33(35):14040–7.

Zacks, J. M. and Tversky, B. (2001). Event structure in perception and conception. *Psychological bulletin*, 127(1):3–21.

Zaghloul, K., Blanco, J., and Weidemann, C. (2009). Human Substantia Nigra Neurons Encode Unexpected Financial Rewards. *Science*, (March):1496–1500.