Learning multi-dimensional rules with probabilistic feedback via value-based serial hypothesis testing

Mingyu Song
Princeton Neuroscience Institute, Princeton University
Princeton, NJ 08540, United States
mingyus@princeton.edu

Yael Niv
Princeton Neuroscience Institute and Department of Psychology, Princeton University
Princeton, NJ 08540, United States
yael@princeton.edu

Ming Bo Cai
International Research Center for Neurointelligence, (WPI-IRCN), UTIAS, The University of Tokyo
Bunkyo City, Tokyo 113-0033, Japan
mingbo.cai@ircn.jp

Abstract

Learning the rules for reward is a ubiquitous and crucial task in daily life, where stochastic reward outcomes can depend on an unknown number of task dimensions. We designed a paradigm tailored to study such complex learning scenarios. In the experiment, participants configured three-dimensional stimuli by selecting features for each dimension and received probabilistic feedback, without being informed of the underlying rules. Through learning, participants were able to select more rewarding features over time. To investigate the learning process, we tested two learning strategies, feature-based reinforcement learning and serial hypothesis testing, and found evidence for both. The extent to which each strategy was engaged depended on the instructed task complexity: when instructed that there were fewer relevant dimensions (and therefore simpler and fewer possible rules) people tended to serially test hypotheses, whereas they relied more on learning feature values when more dimensions were relevant, demonstrating a strategic use of task information to balance the cost-and-benefit of the two learning systems.

1 Introduction

When interacting with a complex environment, it is crucial to figure out the underlying rules for obtaining rewards. Learning such rules, however, can be quite challenging, with numerous potentially relevant dimensions and uncertain outcomes. For example, when learning to bake breads, a collection of decisions needs to be made including the amount of yeast to use, the flour to water ratio, the proof time, the baking temperature, etc. An inexperienced baker can be clueless when facing these decisions, especially when the results are variable even following the same recipe. But after a few attempts on different combinations, they will hopefully figure out the rule for bread-making.

Few studies have investigate how humans learn about rules under the challenges of both multiple relevant dimensions and stochastic feedback (but see [1, 2]). In most multidimensional reinforcement learning (RL) tasks [3, 4, 5], only one dimension of a stimulus is relevant for reward, and participants

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

are explicitly informed so; in category learning tasks, rules often involve multiple dimensions, but they are often deterministic by design [6, 7]. Therefore, we developed a task aimed at studying probabilistic reward learning about multiple (or even an unknown number of) relevant dimensions.

2 The “build-your-own-stimulus" task

In this task, stimuli are characterized by features in three dimensions: color ({red, green, blue}), shape ({square, circle, triangle}) and texture ({plaid, dots, waves}). In each game, a subset of the three dimensions was relevant for reward, meaning that one feature (compared to the other two) in each of these dimensions made stimuli more rewarding.

To earn rewards and figure out the most rewarding features (abbreviated as “rewarding features" from here on) in the relevant dimensions, participants were asked to configure stimuli by selecting features for each dimension (Figure 1). They could also leave any dimension empty, in which case the computer would randomly select a feature in that dimension. The participant then saw the resulting stimulus and received probabilistic reward feedback (one or zero points) based on the number of rewarding features in the stimulus (Table 1). Participants' goal was to earn as many points as possible over the course of each game.

Table 1: Probability of reward for each game type, as a function of number of rewarding features in the stimulus

Game type	# rewarding features			
	0	1	2	3
1D-relevant	20 %	80%	–	–
2D-relevant	20 %	50%	80%	–
3D-relevant	20 %	40%	60%	80%

Each game had 1-3 relevant dimensions (corresponding to 1D, 2D and 3D-relevant conditions), and this number was either known or unknown to participants (“known" and “unknown" conditions), resulting in six game types in total.

Compared to the multidimensional RL tasks and categorization tasks in the literature where stimuli (i.e. the combination of features) are often pre-determined and where it is hard to isolate the participants' preference over single features, this task design enables us to directly probe participants' preference (or lack thereof) in each of the three dimensions.

2.1 Participants.

102 participants recruited through Amazon Mechanical Turk each played all six types of games (3 games of each type, 30 trials per game). Participants were told that there could be one, two or three dimensions that were important for reward, and were explicitly informed about the reward probabilities in Table 1. In “known" games, the number of relevant dimensions was instructed before the start of the game. Participants were never told which dimensions were relevant or which features were more rewarding.

2.2 Learning performance and choice behavior.

Across all six game types, participants' performance improved over the course of a game (Figure 2A). Games were harder (i.e. participants were less able to learn all the rewarding features) as the number of relevant dimensions increased; knowing the number helped performance when three dimensions were relevant (repeated measures ANOVA: $F(1, 101) = 11.3, p = .001$), but not for one or two relevant dimensions (1D: $F(1, 101) = 3.28, p = .073$; 2D: $F(1, 101) = 0.0007, p = .98$).

Participants also showed distinct choice behavior in the different game types (Figure 2B): in “known" (number of relevant dimensions) games, they systematically selected more features on each trial as more dimensions were relevant; in “unknown" games, the number of selected features was not different between game types.

In sum, participants learned the task and performed better than chance, and their performance and choice behavior depended on game conditions.

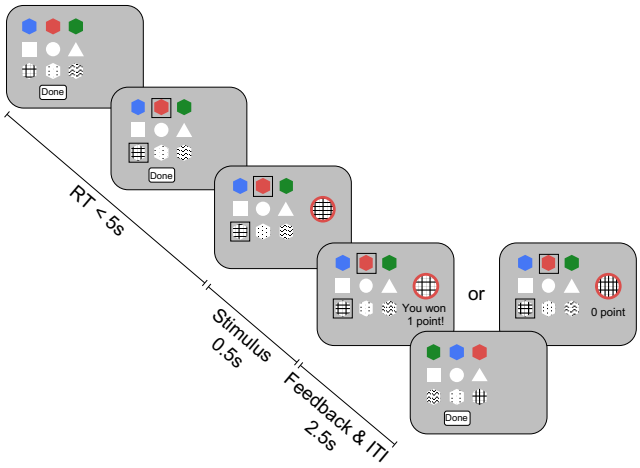

Figure 1: **The build-your-own-stimulus task.** Participants built stimuli by selecting a feature in each of 0-3 dimensions (marked by black squares). After hitting “Done", the stimulus showed up on the screen, with features randomly determined for any dimension without a selection (here, circle was randomly determined). Reward feedback was then shown.

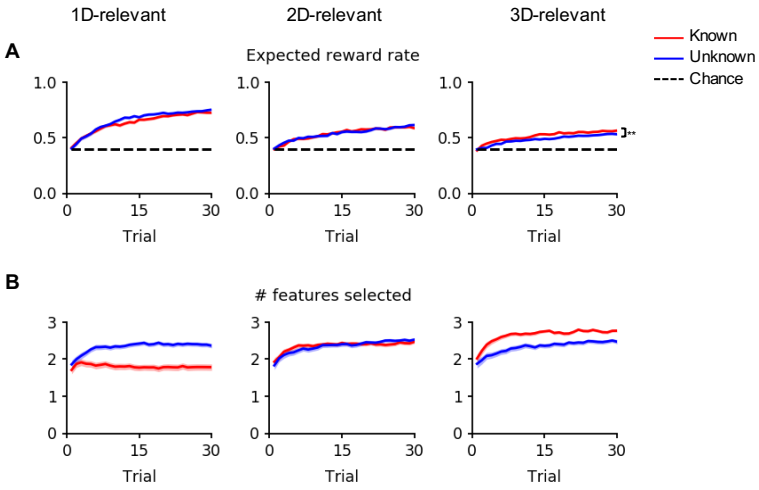

Figure 2: **Performance and choices by game type.** **(A)** The number of rewarding features in the configured stimulus, and **(B)** The number of features selected by the participants, over the course of 1D, 2D and 3D-relevant games (left, middle and right columns); red and blue curves represent the “known" and “unknown" conditions, respectively. Shaded areas represent 1 s.e.m. across participants. Dashed lines represent chance level for that type of game.

3 A hybrid of two learning systems

There is extensive evidence supporting the existence of two learning systems in representation learning [8, 9]: an incremental learning system that learns the value of stimuli based on feedback from trial-and-error experiences, and a rule-based learning system that explicitly represents possible rules and evaluates them. Both learning strategies have been observed in tasks similar to the current one. For instance, in probabilistic reward learning tasks, people seem to learn via trial-and-error to identify relevant dimensions, and gradually focus their attention onto the rewarding features in those dimensions [3, 4, 5]. In contrast, in some types of categorization tasks, people seem to evaluate the probability of all possible rules via Bayesian inference, with a prior belief favoring simpler rules [6]. Inspired by these prior works, we test and compare both learning strategies.

3.1 Reinforcement learning model

First, we consider a feature-based reinforcement learning (RL) model, similar to the feature RL with decay model in [3]. It learns the values of nine features using Rescorla-Wagner updating, with separate learning rates for features that were selected by the participant ($\eta = \eta_s$) and those that were randomly determined ($\eta = \eta_r$). Values for the features not in the current stimulus s_t are decayed towards zero with a factor $d \in [0, 1]$. η_s, η_r and d are free parameters.

$$V_t(f_{i,j}) = \begin{cases} V_{t-1}(f_{i,j}) + \eta(r_t - ER(c_t)), \text{ if } j = s_t^i \\ d \cdot V_{t-1}(f_{i,j}), \text{ if } j \neq s_t^i \end{cases} \tag{1}$$

where i and j index dimensions and features, respectively.

At decision time, the expected reward (ER) for each choice c is calculated as the sum of its feature values:

$$ER(c) = \sum_i V(f_{i,c^i}), \tag{2}$$

with the average value of all three features used for dimensions with no selected features.

The choice probability is then determined based on $ER(c)$ using a softmax function, with β as a free parameter:

$$P(c) = \frac{e^{\beta \cdot ER(c)}}{\sum_{c'} e^{\beta \cdot ER(c')}}. \tag{3}$$

3.2 Rule learning models

Unlike the value-based strategy that learns values for each feature independently and combines them additively at choice time, the rule-based strategy directly evaluates combinations of features. We considered each specification of the relevant dimension(s) and the corresponding rewarding feature(s) as a "rule". For "unknown" games, there were 63 possible rules in total; for "known" games, the total reduced to 9, 27 and 27 for 1D, 2D and 3D-relevant conditions, respectively.

There is little consensus on how people learn which rule is correct. One possibility is to consider all candidate rules, and use Bayes' rule to evaluate how likely each of them is; we term this a "Bayesian rule learning model". This model optimally utilizes feedback information to learn about candidate rules, and can serve as a reference model. However, Bayesian inference is computationally expensive and has a high memory load. A simpler alternative is serial hypothesis testing, with the assumption that people only test one rule at a time: if the evidence supports their hypothesis, they continue with that rule; otherwise, they switch to a different one, until the correct rule is found.

Bayesian rule learning model maintains a probabilistic belief distribution over all possible rules (denoted by h for hypotheses). After each trial, the belief distribution is updated according to Bayes' rule:

$$P(h|c_{1:t}, r_{1:t}) \propto P(r_t|h, c_t)P(h|c_{1:t-1}, r_{1:t-1}). \tag{4}$$

At decision time, the expected reward for each choice is calculated using the entire belief distribution:

$$ER(c) = \sum_h P(h)P(r|h, c). \tag{5}$$

The expected reward is then used to determine the choice probability as in Equation 3.

Serial hypothesis testing (SHT) models assume the participant tests one hypothesis at any given time. We do not directly observe what hypothesis the participant is testing, and must infer that from their choices. We do so by using the change point detection model in [10]. The detailed math of this approach is beyond the page limit, but the basic idea is to infer the current hypothesis using all the choices the participant made so far (in the current game) and their reward outcomes (together denoted by $D_{1:t-1}$). Different variants of the model differ in the assumptions they make about the hypothesis testing and switching policies (i.e., whether to switch hypotheses and which next hypothesis to switch to, respectively; these two policies together determine the transition from the hypothesis on the last trial to the current one), and the choice policy (the probability of choice given the current hypothesis).

Given this generative model of choices, we use Bayes' rule to calculate the posterior probability distribution over the current hypothesis h_t: $P(h_t|D_{1:t-1})$, and use this to predict the choice:

$$P(c_t|D_{1:t-1}) = \sum_{h_t} P(c_t|h_t)P(h_t|D_{1:t-1}) \tag{6}$$

Various choices can be made regarding the three policies, and the hypothesis space. As a baseline, we allow all $N_h = 63$ hypotheses in the hypothesis space, and consider the following hypothesis testing policy: On each trial, the participant estimates the reward probability of the hypothesis on last trial. With a uniform Dirichlet prior, this is equivalent to counting how many times they have been rewarded since they started testing this hypothesis. The estimated reward probability is then compared to a soft threshold θ to determine whether to stay with this hypothesis or to switch to a different one:

$$Pr(\text{stay}) = \frac{1}{1 + e^{-\beta_{\text{stay}}\left(\hat{P}_{\text{reward}} - \theta\right)}}, \tag{7}$$

where $\hat{P}_{\text{reward}} = \frac{\text{reward count} + 1}{\text{trial count} + 2}$ is the estimated probability of reward for hypothesis h_{t-1}, and β_{stay} and θ are free parameters. If the participant decides to switch, the model assumes that they will randomly switch to any other hypothesis:

$$P(h_t) = \begin{cases} Pr(\text{stay}), \text{ if } h_t = h_{t-1} \\ \frac{1}{N_h - 1}\left(1 - Pr(\text{stay})\right), \text{ if } h_t \neq h_{t-1} \end{cases} \tag{8}$$

Participants' choices are assumed to be aligned with their hypotheses most of the time, with a free-parameter lapse rate of λ.

We call this model the **random-switch SHT model**.

3.3 Hybridizing the two learning systems

So far we have considered the two learning systems separately. However, they are not necessarily exclusive. We thus consider a hybrid model by incorporating the feature values into the switch policy of the serial hypothesis testing model. Rather than choosing a new hypothesis randomly, this model favors hypotheses that contain recently rewarded features. It maintains a set of feature values updated according to feature RL with decay as in Equation 1 (but with a single learning rate), and calculates the expected reward for each alternative hypothesis by adding up its feature values, similar to Equation 2 but for h instead of c. The probability of switching to $h_t \neq h_{t-1}$ is:

$$P(h_t) = (1 - Pr(\text{stay})) \frac{e^{\beta_{\text{switch}} \cdot ER(h_t)}}{\sum_{h' \neq h_{t-1}} e^{\beta_{\text{switch}} \cdot ER(h')}}, \tag{9}$$

where β_{switch} is a free parameter. We call this model the **value-based SHT model**.

3.4 Model fitting and model comparison

We fit the models using maximum likelihood estimation with the minimize function (L-BFGS-B algorithm) in Python package scipy.optimize with 10 random starting points. We performed leave-one-game-out cross-validation. Model fits were evaluated with cross-validated trial-by-trial likelihood.

Model comparison results are shown in Figure 3A. Among the four models, the Bayesian rule learning model, even though optimal in utilizing the feedback information, showed the worst fit to participants' choices. This is potentially due to the large hypothesis space (up to 63 hypotheses), making it implausible that participants performed exact Bayesian inference. Both the feature RL with decay model and the random-switch SHT model showed much better fit. Compared to the Bayesian model, both have lower memory and computational loads: the RL model takes advantage of the fact that different dimensions are independent and the reward probabilities are additive, by learning nine feature values individually and later combining them; the random-switch SHT model limits the consideration of hypotheses to one at a time. The hybrid value-based SHT model, combining both learning systems, fit best, suggesting that participants used both strategies when solving this task.

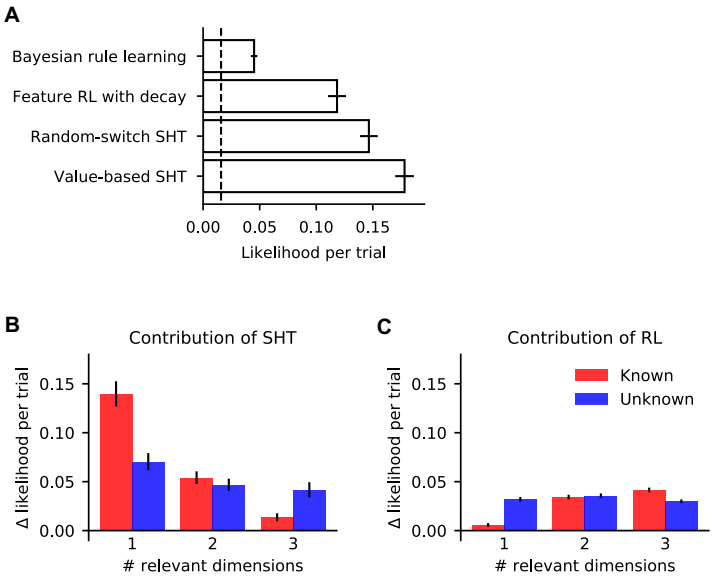

Figure 3: **Model comparison supports both learning strategies. (A)** Geometric average likelihood per trial for each model. Higher values indicate better model fits. Dashed lines indicate the chance level. **(B, C)** The difference in likelihood per trial between the hybrid value-based SHT model and **(B)** the feature RL with decay model (i.e. the contribution of serial hypothesis testing in the hybrid model), or **(C)** the random-switch SHT model (i.e. the contribution of feature value learning), by game type. Error bars represent 1 s.e.m. across participants.

Knowing that both learning systems were used in this task, the next question is how much each of them contributed. We address this question by comparing the hybrid model with the two component models, for each game condition separately: the additional likelihood per trial for the hybrid model as compared to each component is a proxy for the contribution of the other mechanism (Figure 3B and 3C). Our results show that participants' strategies were sensitive to task conditions. In "known" games, the contribution of hypothesis testing decreased with more relevant dimensions (estimated fixed effect slope -0.0631 ± 0.0051 in a mixed linear model with a random intercept, $p < .001$), and the contribution of value learning increased instead (estimated slope: 0.0178 ± 0.0013, $p < .001$). The different use of the two strategies can be explained by the their efficiency (considering both cost and benefit) under different task complexities. In lower-dimensional games, the candidate rules are simpler and thus less working-memory demanding to rehearse; the hypothesis space is also smaller, making it less inefficient to test one hypothesis at a time. Thus, it is more beneficial to use serial hypothesis testing strategy in these games, and this was indeed what the participants did. On the contrary, in higher-dimensional games with more complex rules and larger hypothesis spaces, serial hypothesis testing is less efficient and more costly on mental effort. As a result, it is more beneficial to learn all feature values in parallel, and participants indeed relied more on the value-learning strategy in these games. Taken together, these results suggested that participants took advantage of the task information and stroke a strategic balanced between the two learning systems. This was in contrast to "unknown" games when such task information was unavailable to participants: the contribution of both mechanisms differed less across game conditions (estimated slopes: -0.0144 ± 0.0042 for SHT, $p < .001$; -0.0011 ± 0.0012 for RL, $p = .389$).

4 Discussions

Our work shed light on the way humans learn about rules in complex and stochastic environments to help make better decisions. With limited cognitive resources, their strategies deviated from the optimal Bayesian model, yet performance was close to optimal (average reward across all games is 90.8% compared to the optimal solution). This was achieved by leveraging the existence of two learning systems: The serial hypothesis testing system focuses on learning only one possibility

at a time and reduces noise (locally) faster as a result. The reinforcement learning system learns universally about all features; therefore it takes longer but is more accurate and informative in the long run. The hybrid model (value-based SHT model) incorporates the advantages of both systems by testing single hypothesis within short time intervals and relying on feature values learned incrementally in longer time scales. This model was shown to fit best to participants' behavior. In addition, we showed that human participants were able to gauge which system was more suitable to use under different task conditions and demonstrated a strategic balance between them.

5 Acknowledgment

This work was supported by the National National Institute of Drug Abuse (Grant R01DA042065) and Army Research Office (Grant W911NF-14-1-0101). MBC was supported by World Premier International Research Center Initiative (WPI), MEXT, Japan.

References

[1] Oh-hyeon Choung, Sang Wan Lee, and Yong Jeong. Exploring feature dimensions to learn a new policy in an uninformed reinforcement learning task. *Scientific reports*, 7(1):17676, 2017.

[2] Katherine Duncan, Bradley B Doll, Nathaniel D Daw, and Daphna Shohamy. More than the sum of its parts: a role for the hippocampus in configural reinforcement learning. *Neuron*, 98(3):645–657, 2018.

[3] Yael Niv, Reka Daniel, Andra Geana, Samuel J Gershman, Yuan Chang Leong, Angela Radulescu, and Robert C Wilson. Reinforcement learning in multidimensional environments relies on attention mechanisms. *Journal of Neuroscience*, 35(21):8145–8157, 2015.

[4] Dimitrije Marković, Jan Gläscher, Peter Bossaerts, John O'Doherty, and Stefan J Kiebel. Modeling the evolution of beliefs using an attentional focus mechanism. *PLoS computational biology*, 11(10):e1004558, 2015.

[5] Klaus Wunderlich, Ulrik R Beierholm, Peter Bossaerts, and John P O'Doherty. The human prefrontal cortex mediates integration of potential causes behind observed outcomes. *Journal of neurophysiology*, 106(3):1558–1569, 2011.

[6] Ian Ballard, Eric M Miller, Steven T Piantadosi, Noah D Goodman, and Samuel M McClure. Beyond reward prediction errors: Human striatum updates rule values during learning. *Cerebral Cortex*, 28(11):3965–3975, 2017.

[7] Michael L Mack, Bradley C Love, and Alison R Preston. Dynamic updating of hippocampal object representations reflects new conceptual knowledge. *Proceedings of the National Academy of Sciences*, 113(46):13203–13208, 2016.

[8] F Gregory Ashby and W Todd Maddox. Human category learning. *Annu. Rev. Psychol.*, 56:149–178, 2005.

[9] Angela Radulescu, Yael Niv, and Ian Ballard. Holistic reinforcement learning: the role of structure and attention. *Trends in cognitive sciences*, 2019.

[10] Robert C Wilson and Yael Niv. Inferring relevance in a changing world. *Frontiers in human neuroscience*, 5:189, 2012.