Inverse POMDP: Inferring What You Think from What You Do

Zhengwei Wu
Baylor College of Medicine, Rice University
Houston, TX 77030
Zhengwei.Wu@bcm.edu

Paul Schrater
University of Minnesota
Minneapolis, MN 55455
schrater@umn.edu

Xaq Pitkow
Baylor College of Medicine, Rice University
Houston, TX 77030
xaq@rice.edu

Abstract

Complex behaviors are often driven by an internal model, which integrates sensory information over time and facilitates long-term planning. Inferring the internal model is a crucial ingredient for interpreting neural activities of agents and is beneficial for imitation learning. Here we describe a method to infer an agent's internal model and dynamic beliefs, and apply it to a simulated agent performing a foraging task. We assume the agent behaves rationally according to their understanding of the task and the relevant causal variables that cannot be fully observed. We model this rational solution as a Partially Observable Markov Decision Process (POMDP). However, we allow that the agent may have wrong assumptions about the task, and our method learns these assumptions from the agent's actions. Given the agent's sensory observations and actions, we learn its internal model by maximum likelihood estimation over a set of task-relevant parameters. The Markov property of the POMDP enables us to characterize the transition probabilities between internal states and iteratively estimate the agent's policy using a constrained Expectation-Maximization algorithm. We validate our method on simulated agents performing suboptimally on a foraging task, and successfully recover the agent's actual model.

1 Introduction

In an uncertain and partially observable environment, animals learn to act based on their limited sensory information. The brain evolved complex mechanisms to enable flexible behaviors in this uncertain world, yet its computational strategies remain unclear. To better understand behaviors and interpret the associated neural activities, it would be beneficial to estimate the internal model that explains behavioral strategies of animals. In this paper, we use Partially Observed Markov Decision Processes (POMDP) to model animal behavior as that of rational agents acting under possibly incorrect assumptions about the world. We then solve an inverse POMDP problem to infer these internal assumptions.

In a POMDP, since the world is not fully observed, the agent must create an internal representation of latent states in the world. Identifying the content of such an internal representation will help to identity how these task-relevant variables are encoded in neural responses. Instead of solving directly for the POMDP, we use the mapping between POMDP and a Belief Markov Decision Processes (Belief MDP) to solve for the policies that describe the best actions for the animal's internal model.

Preprint. Work in progress.

Inverse reinforcement learning (IRL) tackles the problem of learning the motivation of an agent based on the behaviors [1]. Specifically, the motivation is a set of reward functions that determine the instantaneous reward obtained through different actions in different states. In addition to learning the reward functions that drive the behaviors, we want to explore the parameters that determine the internal model and the dynamics of the latent states, which is typically described as Inverse Optimal Control [2]. Without a model, inferring both the rewards and latent dynamics is an underdetermined problem. However, under suitably strong model constraints, we show that the agent's reward functions and assumed dynamics can be identified. We therefore combine Inverse Reinforcement Learning and Inverse Optimal Control into an Inverse POMDP problem.

The Inverse POMDP can be cast as a maximum-likelihood problem where the reward functions and latent dynamics can be learned with gradient descent methods [3] to identify the parameters that best explain the animal behaviors under a specific task structure. We use Expectation-Maximization (EM) algorithm [4], specially the Baum-Welch algorithm, to estimate the parameters of the internal model, and infer the posterior of the latent states.

In Section 2, we describe how behaviors are modeled by viewing a POMDP as a Belief MDP. The inference of the internal model is explained in Section 3.1. We applied our method to a naturalistic foraging task described in Section 4. Results of numerical experiments are shown in Section 5, followed by a brief discussion.

2 Behavioral modeling

There is a one-to-one correspondence between a POMDP and a MDP operating on the space of beliefs (a Belief MDP). By using this equivalence, we are able to define belief states and their dynamics, and further to compute the rational policy by which an artificial agent chooses actions, given its reward function and action costs.

2.1 Modeling behavior as a POMDP

In a POMDP in discrete time, the state of the world, s, follows dynamics described by transition probability $T(s', s, a) = P(s'|s, a)$, where s' is the new state, s is the current state, and a is the action selected by an agent. However, the agent does not have direct access to the world state s, but must infer it from measurements o. The sensory information for the agent depends on the action and the world state following probability distribution $P(o|s, a)$. Upon taking action a, the agent receives an immediate reward $r = R(s', s, a)$. The goal of an agent solving a POMDP is to choose actions that maximize the long-term expected reward $E[\sum_{t=0}^{\infty} \gamma^t r_t]$ based on a temporal discount factor $0 < \gamma < 1$. The policy $\pi(a|s)$ describes the probability of choosing an action from a certain state s. We use the "state-action value", $Q_\pi(s, a)$, to quantify how much total future reward can be obtained by taking action a from state s and then following a particular policy π from subsequent states. This value function under optimal policy π^* can be expressed in a recursive form using the Bellman equation [5]:

$$Q_{\pi^*}(s, a) = \sum_{s'} P(s'|s, a) \Big[R(s', s, a) + \gamma \max_{a'} Q(a', s') \Big] \tag{1}$$

where γ is the temporal discount factor, and $R(s', s, a)$ is the instantaneous net reward for taking action a from state s and reaching state s'.

2.2 Belief MDP

In a partially observable environment, an agent can only act on the basis of past actions and observations. The concept of *belief*, which is a posterior distribution over world states s given sensory information, concisely summarizes the information that can be used by agents during decision making. Mathematically, we write the belief $\boldsymbol{b}$ as a vector with length equal to the number of states. The i-th element of the belief vector b^i is the probability that the current state at time t is $s = i$ given the sensory information until now,

$$b_t^i = P(s_t = i|o_{1:t}) \tag{2}$$

The belief state representation allows a POMDP problem to be mapped onto an MDP problem with fully observable states. Here the state is now a belief state instead of true world state, and the process

is known as a *Belief MDP*. Correspondingly, the state-action value function can be defined on belief states as $Q_\pi(b, a)$, with the world state s replaced by the belief state b. The policy π in this case describes the strategy of an agent, which is a mapping from the belief state to actions.

In a Belief MDP, the belief state is a probability distribution and thus takes on continuous values. To make the problem more tractable, we discretize the belief space. This will allow us to solve the belief MDP problem with standard MDP algorithms [5, 6].

3 Internal model inference

The dynamics of the belief states and the policy are dependent on a set of parameters $\boldsymbol{\theta}$. In our setting, these are presumed by the agent to relate to the task setting, but may be incorrect. Inferring the agent's parameters $\boldsymbol{\theta}$ enables us to better understand the internal model of the agents that explains the behavior, and further infer the latent belief of the agent. This can be viewed as a maximum likelihood estimation problem. Due to the Markov property of the belief MDP model, this estimation problem can be analyzed using a hidden Markov model (HMM) where the belief state is a latent variable.

3.1 EM algorithm for inverse POMDP

The EM algorithm [4] enables us to solve for the parameters that give best explanation of the observed data, while inferring unobserved states in the model. Denote by $l(\boldsymbol{\theta})$ the likelihood of the observed data, where $\boldsymbol{\theta}$ are the parameters of the model which include both assumptions about the world dynamics and the parameters determining the sizes of rewards and action costs. We alternately update the parameters $\boldsymbol{\theta}$ that improve the expected complete-data log-likelihood and the posterior over latent states based on the estimated parameter. Let $\boldsymbol{b}$ be the vector of beliefs, which is the latent variable in our belief MDP model, and let $\boldsymbol{a}$ and $\boldsymbol{o}$ be the vector of actions and sensory information over time. The sub-index of the variables $1:t$ means the data sample is from time 1 to t. According to the EM algorithm, in the E-step the estimated parameters $\boldsymbol{\theta}^{old}$ from the previous iteration determine the posterior distribution of the latent variable given the observed data $P_{\boldsymbol{\theta}^{old}}(\boldsymbol{b}|\boldsymbol{a}_{1:T}, \boldsymbol{o}_{1:T})$. In the M-step, the observed data log-likelihood function to be maximized reduces to

$$l(\boldsymbol{\theta}) = \mathcal{Q}(\boldsymbol{\theta}^{old}, \boldsymbol{\theta}) + H(P_{\boldsymbol{\theta}^{old}}(\boldsymbol{b}|\boldsymbol{a}_{1:T}, \boldsymbol{o}_{1:T})). \tag{3}$$

To be consistent with [7], we use $\mathcal{Q}(\boldsymbol{\theta}^{old}, \boldsymbol{\theta})$ as the auxiliary function that describes the expected complete data likelihood, and $H(\cdot)$ is the entropy of the posterior of the latent variable.[1]

During iterations of EM algorithm, the value of the log-likelihood $l(\boldsymbol{\theta})$ always increases to a (possibly local) maximum. Within the M-step, with fixed parameters $\boldsymbol{\theta}^{old}$ from the previous iteration, the entropy of the latent state $H(P_{\boldsymbol{\theta}^{old}}(\boldsymbol{b}|\boldsymbol{a}_{1:T}, \boldsymbol{o}_{1:T}))$ is fixed. As a result, we need to update parameter $\boldsymbol{\theta}$ that maximizes function $\mathcal{Q}(\boldsymbol{\theta}^{old}, \boldsymbol{\theta})$ in the new iteration.

For the Belief MDP, the expected complete data log likelihood can be decomposed into transition probabilities and policies at each time due to the Markov property. The $\mathcal{Q}$-auxiliary function can therefore be expressed as:

$$\begin{aligned}
\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) &= \langle \log P_\theta(\boldsymbol{b}_{1:T}, \boldsymbol{a}_{1:T}, \boldsymbol{o}_{1:T}) \rangle_{P_{\theta^{\text{old}}}(\boldsymbol{b}_{1:T}|\boldsymbol{a}_{1:T}, \boldsymbol{o}_{1:T})} && (4)\\
&= \sum_i \log P_{\boldsymbol{\theta}}(b_0 = i) P_{\boldsymbol{\theta}^{old}}(b_0 = i|\boldsymbol{a}_{1:T}, \boldsymbol{o}_{1:T}) && (5)\\
&+ \sum_t \sum_{i,j} \log P_{\boldsymbol{\theta}}(b_{t+1} = j, o_{t+1}|b_t = i, o_t, a_t) P_{\boldsymbol{\theta}^{old}}(b_t = i, b_{t+1} = j|\boldsymbol{a}_{1:T}, \boldsymbol{o}_{1:T})\\
&+ \sum_t \sum_i \log P_{\boldsymbol{\theta}}(a_t|b_t = i, o_t) P_{\boldsymbol{\theta}^{old}}(b_t = i|\boldsymbol{a}_{1:T}, \boldsymbol{o}_{1:T})
\end{aligned}$$

Since the policy and transition probability depend implicitly on the parameters $\boldsymbol{\theta}$, we are unable to get a closed form of optimal solution for $\boldsymbol{\theta}$. Instead of solving for the optimal $\boldsymbol{\theta}$, we use gradient

[1] Unfortunately, the conventional notations in EM and reinforcement learning collide here, both using the same letter: this $\mathcal{Q}(\boldsymbol{\theta}^{old}, \boldsymbol{\theta})$ auxiliary function is different from the Q-value function in the MDP model, and is denoted in the Calligraphic font.

descent to update the parameter $\boldsymbol{\theta}$ in the M-step. In (5), the terms $P_{\boldsymbol{\theta}}(b_{t+1} = j, o_{t+1}|b_t = i, o_t, a_t)$ depend only on the parameters describing the bayesian update for the state dynamics and the agent's uncertainty about it, while the policy terms $P_{\boldsymbol{\theta}}(a_t|b_t = i, o_t)$ depend on both the dynamic parameters and reward functions. To perform gradient descent, we need to take the gradient of these terms with respect to the parameters $\boldsymbol{\theta}$.

3.2 Derivatives of policy

When the policy is optimal, the term $P_{\boldsymbol{\theta}}(a_t|b_t = i, o_t)$ is a delta function, so the derivative of the policy does not exist. As a result, we approximate the optimal policy using a softmax or Boltzmann policy with a small learnable temperature τ. The softmax introduces an additional sub-optimality of the agent: instead of choosing the action that brings the maximal expected reward, the agent has some chance of choosing a reward that yields a lesser reward, depending on the state-action value Q. Under the softmax policy, the actions under state s follow the distribution

$$\pi_{\text{sfm}}(a|s) = P_{\boldsymbol{\theta}}(a|s) \sim \frac{e^{-Q_{\pi_{\text{sfm}}}(s,a)/\tau}}{\sum_{a'} e^{-Q_{\pi_{\text{sfm}}}(s,a')/\tau}} \tag{6}$$

If we can calculate the derivative of the Q-value function with respect to the parameter set $\boldsymbol{\theta}$, we are able to get the policy derivatives. Similarly to the Bellman equation (1) based on the optimal policy, the Q-value function under a softmax policy can also be expressed in a recursive way, replacing the max with an average:

$$Q_{\pi_{\text{sfm}}}(s,a) = \sum_{s'} P(s'|s,a)\Big[R(s',s,a) + \gamma \sum_{a'} \pi_{\text{sfm}}(a'|s') Q_{\pi_{\text{sfm}}}(s',a')\Big] \tag{7}$$

For simplicity, we omit the temporal subindices in the following derivation. Denote the vectorized version of $Q(s,a)$ and $\pi(s|a)$ as Q^V and π^V. Differentiating with respect to $\boldsymbol{\theta}$ on both sides gives us:

$$\frac{\partial Q^V}{\partial \boldsymbol{\theta}} = \mathbf{c}_1^V + \gamma \Gamma(P(s'|s,a))\Big(\text{diag}(Q^V(s',a'))\frac{\partial \pi^V}{\partial Q^V} + \text{diag}(\pi^V(s',a'))\Big)\frac{\partial Q^V}{\partial \boldsymbol{\theta}} \tag{8}$$

where $\mathbf{c}_1^V$ is a vectorized version of matrix $\mathbf{c}_1(s,a)$ with

$$\begin{aligned}\mathbf{c}_1(s,a) = &\sum_{s'} \frac{\partial P(s'|s,a)}{\partial \boldsymbol{\theta}}\Big[R(s',s,a) + \gamma \sum_{a'} \pi(a'|s') Q(a',s')\Big] \\ &+ \sum_{s'} P(s'|s,a)\frac{\partial R(s',s,a)}{\partial \boldsymbol{\theta}}\end{aligned}$$

and $\Gamma(P(s'|s,a))$ is a function containing repeated blocks of the transition probabilities $P(s'|s,a)$.

By reorganizing equation (8), we can see that the derivative of the Q-value function with respect to the parameters can be solved analytically as a linear function of the known quantities. Using the chain rule, the gradient of the policy can be obtained in this way. We then use this gradient in the Expectation-Maximization algorithm to estimate the internal model parameters that best explain the observed data.

4 Application to foraging

We applied our method to the specific setting of a task in which an animal can forage at either of two locations ('feeding boxes') which may have hidden food rewards that appear with a certain rate. A few discrete actions are available to the animal: it can open a box to get reward or observe its absence, go to the other box, or stay in a certain place.

To define the Belief MDP for this 'two-box' task, we need to define the states, actions and rewards. The states must represent the agent's location, whether it has obtained food from the boxes, and also whether food is available in each box. Since the agent knows its location exactly, and knows whether it has obtained food, we only need a belief representation for the unobserved food availability in each box.

We assume there are three possible locations for the agent: the positions of boxes 1 and 2, and a middle location 0. The actions are defined in a mutually exclusive way as: doing nothing, going to location 0/1/2, and pressing a button on the closest box to retrieve food (if available). Each action has an associated cost, such as the traveling cost, and the button pressing cost. This cost disincentivizes the agent from rapidly repeating actions that might access reward. We also include a small 'grooming' reward for staying at the middle location 0 to encourage the agent to stop and think.

In addition to the cost of actions, there are several parameters that are related to the experiment setting. The food availability in each box follows a telegraph process: the food becomes available following a Poisson process with rate γ, and then becomes unavailable following another Poisson process with a different transition rate ϵ. We assume the agent knows these dynamics, but may mistakenly assume different values for the transition rates. Let $A_{i,t} \in \{0, 1\}$ be the food availability for box $i \in \{1, 2\}$ at time t. By omitting the box index i, we now consider the dynamics of food availability at a specific box. When the animal takes no action, denoted by $a = \varnothing$, the food availability transitions according to $p(A_{t+1}|A_t, a = \varnothing)$:

		A_t	
		0	1
A_{t+1}	0	$1-\gamma$	ϵ
	1	γ	$1-\epsilon$

(9)

For a single box, the belief dynamics has the form $\boldsymbol{b}_{t+1} = T\boldsymbol{b}_t$, where T is the transition matrix, and $\boldsymbol{b}$ contains two elements $b_t^1 = P(A_t = 1|o_{1:t}, a_{1:t})$ and $b_t^0 = P(A_t = 0|o_{1:t}, a_{1:t})$. Since the availability is binary, $b_t^1 = 1 - b_t^0$, we only need to track b_t^1. According to the box dynamics (9),

$$b_{t+1}^1 = \gamma + (1 - \epsilon - \gamma)b_t^1 \tag{10}$$

In a Belief MDP, the belief state is a probability distribution over the unknown world states $s \in \mathcal{S}$, and if there are $|S|$ possible values then the beliefs states take on continuous values in an $|S| - 1$-dimensional simplex. For computational tractability, we discretize beliefs in each box into N states. This is sensible also computationally, since it is unlikely that an animal will maintain arbitrary precision about its uncertainty: it is difficult to distinguish between 70% and 80% confidence.

We then define the transition matrix in the discretized belief space by binning the transition matrix, integrating the transition probabilities for the continuous belief space within a given bin (Figure 1). To approximate this length, we assume a multivariate Gaussian distribution with mean at the center of each bin. The probability at the point, which is the orthogonal projection of the center point onto the dynamic line, reflects the length of the red line shown in (Figure 1). By adjusting the covariance of the multivariate Gaussian distribution, we control the diffusion between neighboring bins, which reflects additional belief stochasticity.

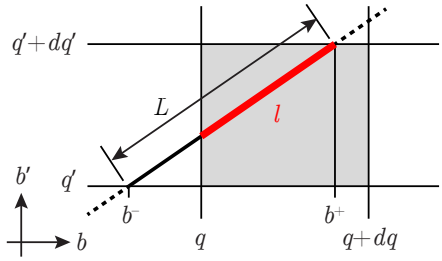

Figure 1: Quantization of deterministic relationship between $b_t = b$ and $b_{t+1} = b'$ within a bin $b \in [q, q+dq]$ and $b' \in [q', q'+dq']$. The diagonal line reflects the deterministic transition probability $P(b'|b)$ for the continuous belief space, which is a deterministic function. The mass within a bin on belief space is proportional to the length l of the red line. This length l can be approximated with the probability density of a two-dimensional multivariate Gaussian distribution with mean at the center of the bin.

When a button-press action is taken to open a box, any available reward there is acquired. Afterwards, the animal knows there is no more food available now in the box (since it was either unavailable or consumed) and the belief is reset to zero.

With the transition matrices and reward functions for different actions for the internal model, the animal has an optimal policy that is based on the value of different actions. To allow for variability of actions, we assume that the animal uses a softmax policy (6).

5 Experiments

We now apply the learning method for solving an inverse POMDP (Section 3.1) to the foraging task (Section 4). The goal is to estimate a simulated agent's internal model and belief dynamics from its sensory observations and chosen actions.

For simplicity we assume that reward availability at both boxes follows a telegraph process with the same appearance probability of $\gamma_1 = \gamma_2 = 0.1$ and disappearance probability of $\epsilon_1 = \epsilon_2 = .01$, per discrete time step. Although here the two boxes have identical dynamics, our model estimation algorithm will also work in cases where the two boxes have different dynamics.

Without loss of generality, we measure gains and losses compared to the food reward at one box, thus defining the reward as $r = 1$. In that currency, the cost (negative reward) of pressing the button is 0.3, and that of traveling is 0.2 (switching between boxes requires two steps for a total cost of 0.4). We also allow a small reward for waiting of $r = 0.05$ at the center location (*e.g.* while grooming).

We assume the action of an agent taking optimal strategy is determined rationally according to the value function (1). In Figure 2A, we show some properties of the value function under the optimal solution of this task given the agent's incorrect assumptions. Qualitatively, we see that the policy of the agent is to go to the box that has higher expected value, consistent with our intuition. Figure 2B shows the same value functions based on the internal model that we estimate by our method.

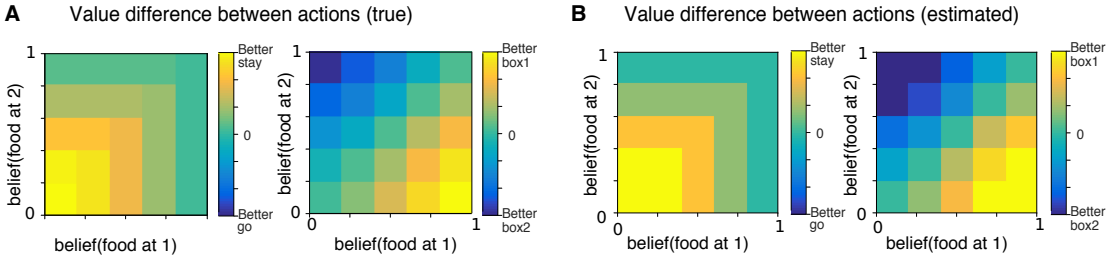

Figure 2: Visualization of real and inferred value functions. **A**: Values of different actions and states under the optimal solution of the two-box task, when the agent starts between the two boxes and has specific beliefs (subjective probability) about whether reward is available at each box. Left: The value difference (colors) between the actions 'stay' and 'go'. When the belief in food availability at *either* box 1 or 2 is high, the value of 'go' is higher than that of 'stay', and the animal chooses to go. Right: The optimal agent places a higher value on the box where the belief in available food is highest. **B**: Values of different actions and states based on inferred internal model, under optimal solution.

To allow for variability in action selection, we create an agent that uses a softmax policy (6) with temperature $\tau = 0.2$. This small temperature enables the agent to follow an approximately optimal policy based on state-action value $Q(s, a)$.

The actions and sensory evidence (locations and rewards) obtained by the agent all constitute observations for the experimenter's learning of the agent's internal model. Based on these observations over time, we use the EM algorithm to infer the parameters of the internal model that can best explain the behavioral data.We track the agent's actions and sensory observations over $T = 5000$ time points. In Figure 3, we show an example of the task data. When the animal pushes the button and opens the box at a time when there is food in the box, the food is acquired; when there is no food available at that time, the animal receives no reward. The animal may travel between the two boxes to maximize the expected future reward; it may also stay at a certain place whenever beneficial.

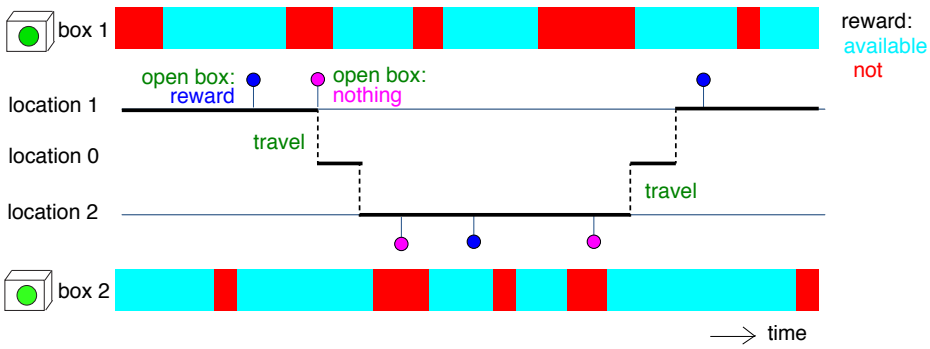

Figure 3: An example of task data. The reward availability in each of two boxes evolves according to a telegraph process, switching between available (cyan) and unavailable (red), and the animal may travel between the locations of the two boxes. When the box is opened, if there is food in it, the reward is obtained; otherwise, there is no reward.

In Figure 4, we show the results for inference based on a typical set of data. With EM algorithm, we are able to solve for the parameters that have the largest likelihood given the observations. The comparison between the true parameters and the estimated parameters are shown in table 5.

Comparison of true and estimated parameters

$\boldsymbol{\theta}$	γ_1	γ_2	ϵ_1	ϵ_2	Grooming	Traveling	Pressing button
True	0.1	0.1	0.01	0.01	0.05	-0.2	-0.3
Estimated	0.1225	0.1256	0.0077	0.0124	0.0360	-0.2080	-0.4424

Due to the limited amount of data, there is discrepancy between the true parameters and the estimated parameters. This discrepancy can be reduced with larger amount of data. With the estimated parameters, we are able to infer dynamics of the posterior over the latent states, which are the beliefs on the two boxes. Note that this is an experimenter's posterior over the agent's subjective posterior. The inferred posteriors have similar dynamics as the true latent belief states. Consistent with the true probability of the food availability in each box according to the underlying telegraph process, the inferred posteriors exhibit exponentially shaped time series.

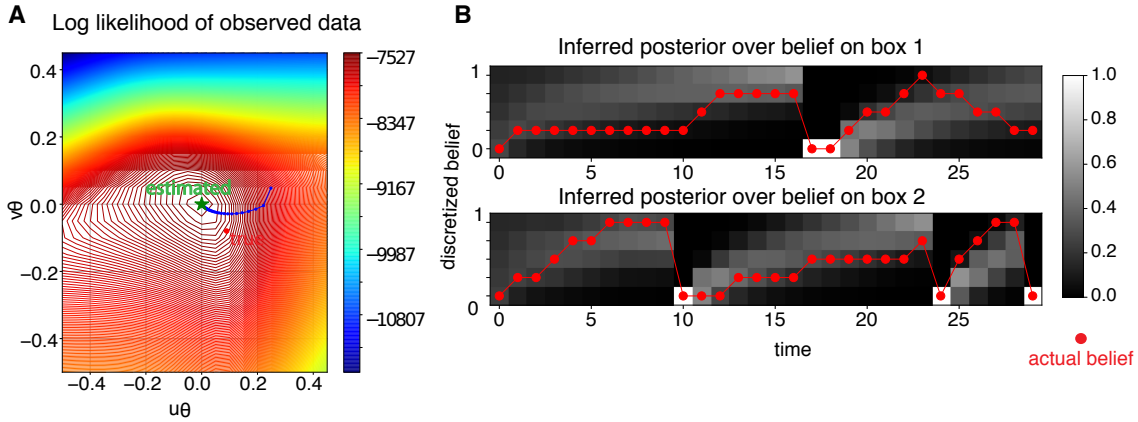

Figure 4: Inference of parameters of the internal model and the posterior over the latent belief states. **A**: The estimated parameters converge to the optimal point of the log-likelihood contour. Since the parameter space has high dimensions, we project them onto the first two principal components of the trajectory. **B**: Inferred posterior of the latent states. The greyscale indicates the probability over the possible beliefs, and the red dots are the true belief states of the agent over time. The posteriors are consistent with the the dynamics of the true beliefs.

Since our inferred model parameters differ slightly from the agent's true parameters, we examine how those two internal models differ. Based on the estimated parameters, we therefore create another simulated agent using the inferred internal model. Under the optimal policy for the given model, the new agent reveals a similar Q-value function. Figure 2 shows that starting from the middle location, both agents have almost the same preferences of actions under the different belief states. Figure 5 shows that under softmax near-optimal policies, the two agents choose actions with similar frequencies, occupy the three locations for the same fraction of time, and wait similar amounts of time between pushing buttons or travelling. This demonstrates that our estimated agent's internal model generates behaviors that are consistent with behaviors of the agent from which it learned.

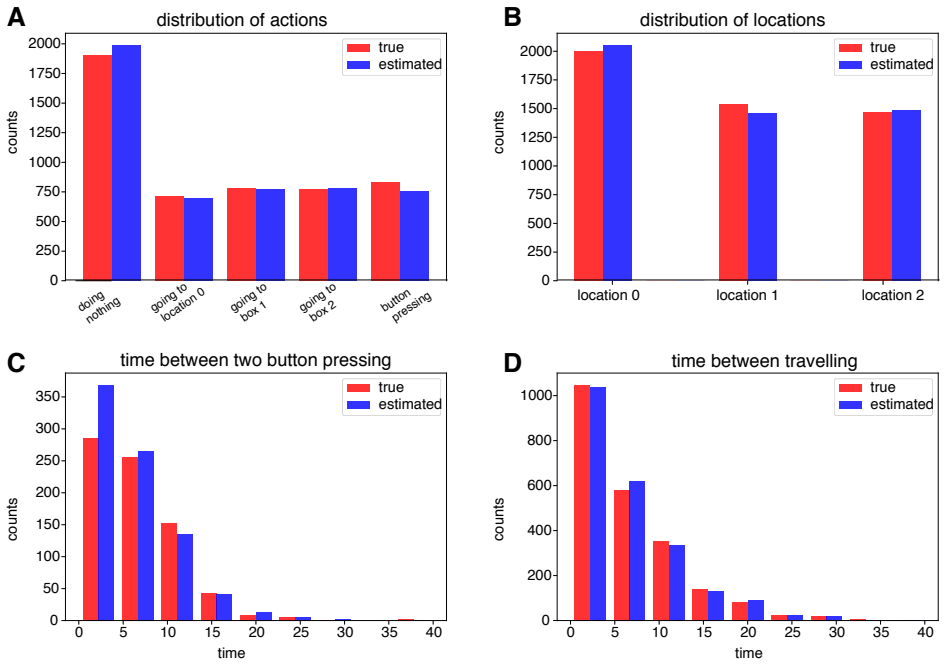

Figure 5: Comparing statistics of behaviors for the actual agent and the inferred agent. **A**: The distribution of actions. **B**: The distribution of time staying at each location. **C**: The distribution of time intervals between two button pressing actions. **D**: The distribution of time intervals between traveling actions.

6 Conclusions

We presented a method to infer the internal model of a rational agent who collects rewards in a task by following a Partially Observable Markov Decision Process. Given that an agent chooses actions in this way, the estimation of its internal model parameters can be formulated as a maximum likelihood problem, and the parameters can be inferred using the EM algorithm. When we applied our method to a foraging task, experiments showed that the parameters that best explain the behavior of the agent nicely matched the internal parameters of that agent. The estimated internal model and the true internal model produced similar value functions and behavioral statistics.

This approach generalizes previous meta-Bayesian work that imputes beliefs to rational observers [8, 9]. Though similar in spirit, these past approaches were based on simpler trial-structured tasks like perceptual decision-making, whereas our method can infer models that agents use to make long-term plans and choose long sequences of actions.

Our framework is quite general, and can be applied to still more complex tasks. It can be used to infer false beliefs derived from incorrect or incomplete knowledge of task parameters. It can also be used to infer incorrect *structure* within a given model class. For example, it is natural for animals to assume that some aspects of the world, such as reward rates at different locations, is not fixed, even if

an experiment uses fixed rates [10]. Similarly, an agent may have a superstition that different reward sources are correlated even when the true process in independent. Given a model class that includes such counterfactual relationships between task variables, our method could test whether an agent holds these incorrect assumptions about the task structure.

The success of our method on simulated agents suggests our method could be usefully applied to experimental data from real animals performing such foraging tasks [11]. Accurate estimation of dynamic belief states would provide useful targets for interpreting dynamic neural activity patterns, which could help identity the neural substrates of task-relevant thoughts.

References

[1] Stuart Russell. Learning agents for uncertain environments. In *Proceedings of the eleventh annual conference on Computational learning theory*, pages 101–103. ACM, 1998.

[2] Krishnamurthy Dvijotham and Emanuel Todorov. Inverse optimal control with linearly-solvable mdps. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 335–342, 2010.

[3] Monica Babes, Vukosi Marivate, Kaushik Subramanian, and Michael L Littman. Apprenticeship learning about multiple intentions. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 897–904, 2011.

[4] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38, 1977.

[5] R Bellman. Dynamic programming: Princeton univ. press, 1957.

[6] Ronald A Howard. *Dynamic programming and Markov processes*. Wiley for The Massachusetts Institute of Technology, 1964.

[7] Christian Robert. Machine learning, a probabilistic perspective, 2014.

[8] Jean Daunizeau, Hanneke EM Den Ouden, Matthias Pessiglione, Stefan J Kiebel, Klaas E Stephan, and Karl J Friston. Observing the observer (i): meta-bayesian models of learning and decision-making. *PLoS One*, 5(12):e15554, 2010.

[9] Ferenc Huszár, Uta Noppeney, and Máté Lengyel. Mind reading by machine learning: A doubly bayesian method for inferring mental representations. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 32, 2010.

[10] Christopher M Glaze, Alexandre LS Filipowicz, Joseph W Kable, Vijay Balasubramanian, and Joshua I Gold. A bias–variance trade-off governs individual differences in on-line learning in an unpredictable environment. *Nature Human Behaviour*, 2(3):213, 2018.

[11] Leo P Sugrue, Greg S Corrado, and William T Newsome. Matching behavior and the representation of value in the parietal cortex. *science*, 304(5678):1782–1787, 2004.