Inverse POMDP: Inferring Internal Model and Latent Beliefs

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

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

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

Abstract

Complex behaviors are often driven by an internal model, which may reflect memories, beliefs, motivation, or arousal. Inferring the internal model is a crucial ingredient for understanding how the brain generates behaviors and interpreting neural activities of agents. 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. Assuming rationality of animals, we model the behaviors of the animals as a Partially Observable Markov Decision Process (POMDP). 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.

Keywords: POMDP, belief MDP, behavior modeling

Introduction

The brain evolved complex mechanisms to enable flexible behaviors in an uncertain and partially observable environment, 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.

Since the world state is not fully observed, the agent needs to create an internal representation of the state in the world. There is a one-to-one correspondence between a POMDP and a Marokv Decision Process (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.

Inverse reinforcement learning (IRL) tackles the problem of learning the motivation of an agent based on the behaviors (Russell, 1998), which is a set of reward functions that determine the instantaneous reward obtained through different actions in different states assuming known dynamics. Under suitably strong model constraints, we show that the agent's reward functions and assumed dynamics can be identified. The Inverse POMDP can be cast as a maximum-likelihood problem where the reward functions and latent dynamics parameters can be learned with gradient descent methods (Babes, Marivate, Subramanian, & Littman, 2011). We use Expectation-Maximization (EM) algorithm (Dempster, Laird, & Rubin, 1977), specially the Baum-Welch algorithm, to estimate the parameters of the internal model, and infer the posterior over the latent states.

Behavioral Modeling

Modeling Behavior as 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 sensory observations o according to the 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 a 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 (Bellman, 1957):

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

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'.

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 $\mathbf{b}$ 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,

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

where the sub-index $1:t$ denotes the time span of the data sample.

The fully observable belief state representation allows a POMDP problem to be mapped onto an MDP problem with the state-action value function on belief states, $Q_\pi(b,a)$. The policy π in this case is a mapping from the belief state to actions.

To make belief MDP problems more tractable, we can discretize the belief space, which will allow us to solve the problem with standard MDP algorithms (Bellman, 1957; Howard, 1964).

Internal Model Inference

The dynamics of the belief states and the policy are determined by a set of parameters θ. In our setting, the agent assumes they know the true parameters and acts accordingly, but we allow that they may be incorrect.

Inferring the agent's parameters θ can be viewed as a maximum likelihood estimation problem. The EM algorithm (Dempster et al., 1977) enables us to solve for the parameters that give best explanation of the observed data, while inferring unobserved states in the model. Denote by $l(\theta)$ the likelihood of the observed data, where θ 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. Let $\mathbf{b}$ be the vector of beliefs, which is the latent variable in our belief MDP model, and let $\mathbf{a}$ and $\mathbf{o}$ be the vector of actions and sensory information over time. According to the EM algorithm, we alternately update the parameters θ that improve the expected complete-data log-likelihood and the posterior over latent states based on the estimated parameter. In the E-step, we need to determine the posterior distribution of the latent variable given the observed data, $P_{\theta^{\text{old}}}(\mathbf{b}|\mathbf{a}_{1:T}, \mathbf{o}_{1:T})$, based on the estimated parameters θ^{old} from the previous iteration. In the M-step, we update the parameters by maximizing an auxiliary function that describes the expected complete data likelihood,

$$Q(\theta, \theta^{\text{old}}) = \langle \log P_\theta(\mathbf{b}_{1:T}, \mathbf{a}_{1:T}, \mathbf{o}_{1:T}) \rangle_{P_{\theta^{\text{old}}}(\mathbf{b}_{1:T}|\mathbf{a}_{1:T}, \mathbf{o}_{1:T})}. \tag{3}$$

Since the policy and transition probability depend implicitly on the parameters θ, we are unable to get a closed form of optimal solution for θ. Instead of solving for the optimal θ, we need to take the gradient of these terms with respect to the parameters θ, and use gradient descent to update the parameter θ in the M-step.

Here 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

$$\pi_{\text{sfm}}(a|s) = P_\theta(a|s) \sim \frac{e^{-Q_{\pi_{\text{sfm}}}(s,a)/\tau}}{\sum_{a'} e^{-Q_{\pi_{\text{sfm}}}(s,a')/\tau}}. \tag{4}$$

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:

$$Q_{\pi_{\text{sfm}}}(s,a) = \sum_{s'} P(s'|s,a) \Big[R(s',s,a) + \gamma \sum_{a'} \pi_{\text{sfm}}(a'|s') Q_{\pi_{\text{sfm}}}(s',a') \Big] \tag{5}$$

Differentiating with respect to θ on both sides, and reorganizing the terms, 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 EM algorithm to estimate the internal model parameters that best explain the observed data.

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.

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 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 with an associated cost 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). 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.

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 consider the food dynamic in a specific box when no action is taken as follows:

A_{t+1} \ A_t	0	1
0	1-γ	ε
1	γ	1- ε

With belief defined as $b_t^1 = P(A_t = 1 | \mathbf{o}_{1:t}, \mathbf{a}_{1:t})$ and $b_t^0 = P(A_t = 0 | \mathbf{o}_{1:t}, \mathbf{a}_{1:t})$, we can see that the belief has dynamics:

$$b_{t+1}^1 = \gamma + (1 - \varepsilon - \gamma) b_t^1 \tag{6}$$

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. For computational tractability, we discretize the continuous beliefs in each box into N states. With the transition matrices and reward functions for different states and actions, 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 (4).

Experiments

We now apply our learning method for solving an inverse POMDP for the foraging task. 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 at each discrete time step, the 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 $\varepsilon_1 = \varepsilon_2 = 0.01$. 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 relative 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 1, 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.

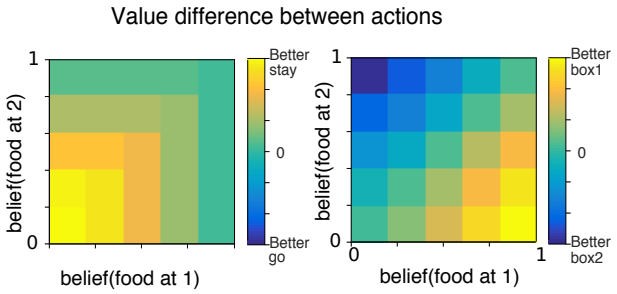

Figure 1: Visualization of value functions. Values of different actions and states under the optimal solution of the two-box task, when the agent starts between the two boxes and has specific beliefs (subjective probability) about whether reward is available at each box. Left: The value difference (colors) 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.

To allow for variability in action selection, we create an agent that uses a softmax policy (4) with temperature $\tau = 0.2$. This small temperature enables the agent to follow an approximately optimal policy based on state-action value $Q(s, a)$.

We track the agent's actions and sensory observations over $T = 5000$ time points. In Figure 2, we show an example of the task data.

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.

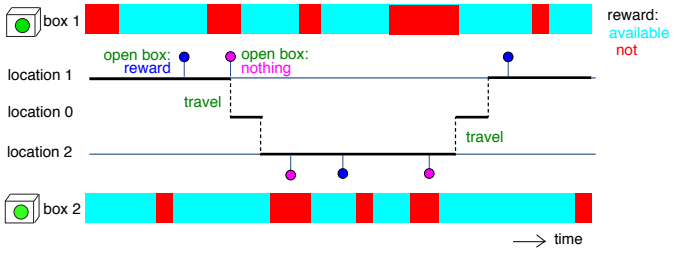

Figure 2: An example of task data. The reward availability in each of two boxes evolves according to a telegraph process, switching between available (cyan) and unavailable (red), and the animal may travel between the locations of the two boxes. When the box is opened, if there is food in it, the reward is obtained; otherwise, there is no reward.

Table 1: Comparison of true and estimated parameters

θ	γ_1	γ_2	ε_1	ε_2
True	0.1	0.1	0.01	0.01
Estimated	0.1225	0.1256	0.0077	0.0124

Reward	Groom	Travel	Button Press
True	0.05	-0.2	-0.3
Estimated	0.0360	-0.2080	-0.4424

In Figure 3, we show the results for inference based on a typical data set. The comparison between the true parameters and the estimated parameters are shown in Table 1.

Due to the limited amount of data, there is a small discrepancy between the true parameters and the estimated parameters. This discrepancy can be reduced with additional data. With the estimated parameters, we can then 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 (Figure 3B). 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.

Based on the estimated parameters, we create another simulated agent using the inferred internal model to compare the true and inferred model. Figure 4 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 intervals between pushing buttons or traveling. This demonstrates that our estimated agent's internal model generates behaviors that are consistent with behaviors of the agent from which it learned.

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 param-

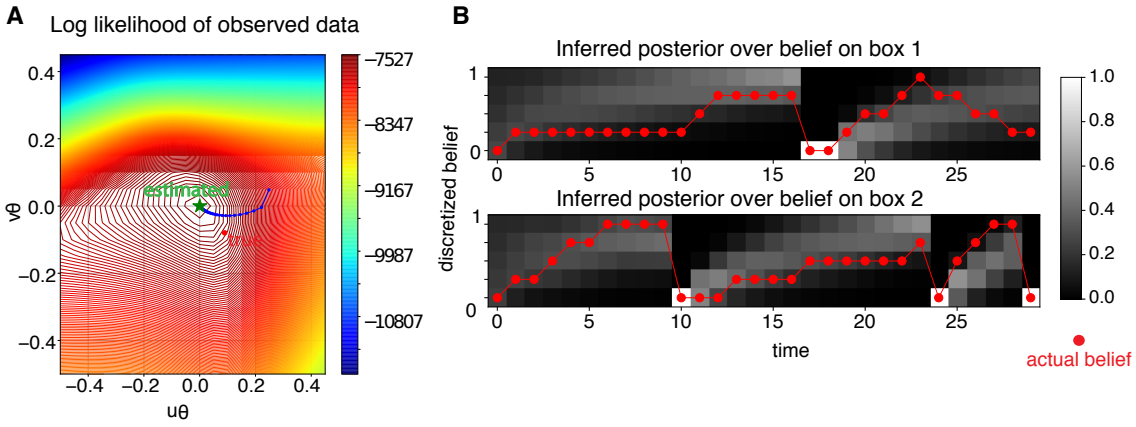

Figure 3: Inference of parameters of the internal model and the posterior over the latent belief states. **A**: The estimated parameters converge to the optimal point of the log-likelihood contour. Since the parameter space has high dimensions, we project them onto the first two principal components 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.

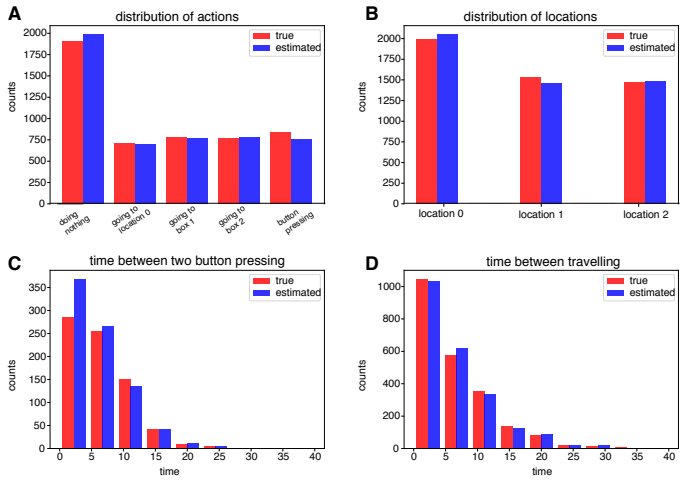

Figure 4: Comparing statistics of behaviors for the actual agent and the inferred agent. **A**: The distribution of actions. **B**: The distribution of time staying at each location. **C**: The distribution of time intervals between two button pressing actions. **D**: The distribution of time intervals between traveling actions.

eters 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, numerical 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.

The success of our method on simulated agents suggests our method could be fruitfully applied to experimental data from real animals performing such foraging tasks (Sugrue, Corrado, & Newsome, 2004). 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.

Acknowledgments

The authors thank Dora Angelaki, Baptiste Caziot, Neda Shahidi, Russell Milton, Valentin Dragoi for useful discussions. ZW, PS, and XP were supported by BRAIN Initiative grant NIH 5U01NS094368.

References

Babes, M., Marivate, V., Subramanian, K., & Littman, M. L. (2011). Apprenticeship learning about multiple intentions. In *Proceedings of the 28th international conference on machine learning (icml-11)* (pp. 897–904).

Bellman, R. (1957). *Dynamic programming: Princeton univ. press.* Princeton.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, 1–38.

Howard, R. A. (1964). *Dynamic programming and markov processes.* Wiley for The Massachusetts Institute of Technology.

Russell, S. (1998). Learning agents for uncertain environments. In *Proceedings of the eleventh annual conference on computational learning theory* (pp. 101–103).

Sugrue, L. P., Corrado, G. S., & Newsome, W. T. (2004). Matching behavior and the representation of value in the parietal cortex. *science*, *304*(5678), 1782–1787.