Reinforcement Learning with Limited Reinforcement: Using Bayes Risk for Active Learning in POMDPs
Finale Doshi, Joelle Pineau, and Nicholas Roy

Presented by Hamid R. Chinaei

Computer Science and Software Engineering Department, Laval University

October 24th, 2008
Outline

- Wheelchair Scenario
- Problems and Approaches
- Modeling POMDP Uncertainty
- Active Dialogue Model Learning
- Bayes Risk Action Selection
- Updating the Model Distribution
- Performance
- Conclusions and Future Works
Wheelchair Scenario

- start
- Go to Kitchen
- Go to Bathroom
- Go to Bedroom

... done
reset

Go to Bathroom

Bayes Risk for Active Learning in POMDPs
Finale Doshi, Joelle Pineau, and Nicholas Roy

3 / 19
Problems and Approaches

Problems

1. Gathering training data for supervised learning may be expensive.
2. Most approaches require the agent to experience a large penalty.
3. Accurate numerical reward is especially hard to obtain from people.
4. Inverse reinforcement learning is challenging, identifying the reward model without explicit reinforcement, is a challenge.

Approaches

1. Bayesian methods: since allow experts to incorporate domain knowledge into priors over models.
2. Active learning: since allows asking for which actions (instead of numerical feedback).
Modeling POMDP Uncertainty

Assumptions

1. The sets $S$, $A$, and $O$ are known
2. The true model is static

We know that $T$, $\Omega$, $R$ describe the dynamics of the problem

Good to know

1. Priors for $T$ and $\Omega$ captured from Dirichlet distributions
2. Priors for $R$ are uniform on the expert specified range
3. $S' = S \times M$, $M$ all valid values for the model parameters
   - $S'$ to reduce uncertainty on $M$ in the next step
POMDP model

The belief $b_{t+1}^a(s)$ is the belief after a Bayesian update using:

$$b_{t+1}^{a,o}(s) = \frac{\Omega(o|s', a) \sum_{s' \in S} T(s'|s, a) b(s)}{\sum_{\sigma \in S} \Omega(o|\sigma, a) \sum_{s \in S} T(\sigma|s, a) b_t(s)}$$

$$V_t(b) = \max_{a \in A} Q_t(b, a)$$

$$Q_n(b, a) = R(b, a) + \gamma \sum_{o \in O} \Omega(o|b, a) V_{n-1}(b^{a,o})$$
Sample POMDPs from a prior distribution
Complete a task choosing actions based on Bayes risk:
  - Use the POMDP samples to compute the action with minimal Bayes risk
  - If the risk is larger than a given $\xi$, perform a meta-query
  - Update each POMDP sample’s belief based on the observation received
Once a task is completed, update priors
  - Use a kernel incorporating action-observation history to propagate POMDP samples
  - Weight POMDPs based on meta-query history
Bayes Risk Action Selection

\[ L_m(a, a^*; b) = Q_m^*(b, a) - Q_m^*(b, a^*) \]

\[ BR(a) = \int_M (Q_m^*(b_m, a) - Q_m^*(b_m, a^*)) p_M(m) \]

Where \( M \) is the space of all possible dialogue models \( m \)
\( p(m) \) agent’s belief over user states and dialog models

\[ BR(a) = \int_M (Q(b_m, a)p_M(m) - \int_M Q(b_m, a^*)p_M(m) \]

\[ V_{BR} = \max \int_M (Q(b_m, a)p_M(m) \]

Hopefully the uncertainty over models will be resolved at the next time step
Approximation and bounds

\[ BR(a) \approx \sum_i (Q(b_i, a) - Q_i(b_i, a^*_i))p_M(m_i) \]

1. **Error due to Monte Carlo approximation**

\[
0 \leq (Q(b_i, a) - Q_m(b_m, a^*_i))p_M(m_i) \leq \frac{R_{max} - \min(R_{min}, \xi)}{1 - \gamma}
\]

Hoeffding bound, sampling error \( \epsilon \), and confidence \( \delta \), we require \( n_m \) samples:

\[
n_m = \frac{R_{max} - \min(R_{min}, \xi)}{2(1 - \gamma)^2 \epsilon^2} \ln(1/\delta)
\]

2. **Error due to the point-based approximation of** \( Q(b_i, a) \):

\( Q(b_i, a) - Q_m(b_m, a^*_i) \) may have error of up to

\[
\epsilon_{PB} = \frac{2(R_{max} - R_{min})\delta_B}{1 - \gamma^2}
\]
Updating the Model Distribution

\[ p_{M|h,Q}(m|h, Q) \propto p(Q|m)p(h|m)p_M(m) \]

1. During a task: Updating particle weights
2. Between tasks: Resampling Particles
During a task: Updating particle weights

\[ m_t \sim K(m_{t-1}, m_t) \]

\[ w_t = w_{t-1} \frac{p_{M,t}(m_t)}{p_{M,t-1}(m_{t-1})K(m_{t-1}, m_t)} \]

**Attention!**

- Sampling a new POMDP is expensive
- No sampling during a task
- \( k(m, m') = \delta_m(m') \) where \( \delta \) is Dirac delta function

If a meta Query happened at time \( t \):

\[ p_{M,t} \propto p(Q_t|m)p_{M,t-1}(m) \]

\[ w_t = w_{t-1}p(Q_t|m) \]
Meta Query

- I think $a_i$ is the best action, Should i do it?
- Then I think $a_j$ is best. Is that correct?

\[ w_t \propto \frac{p(Q|m')p_{M|h}(m')}{p(Q|m)p_M(m)K(m, m')} \]
Resampling, cont’d
Figure 1. Boxplot of POMDP learning performance with a discrete set of four possible models. The medians of the policies are comparable, but the active learner (left) makes fewer mistakes than the passive learner (center). The Bayes risk action selection criterion (right) does not cause the performance to suffer.
### Table 2. Mean difference between optimal (under the true model) and accrued rewards (smaller = better).

<table>
<thead>
<tr>
<th>Problem</th>
<th>States</th>
<th>Control</th>
<th>Passive</th>
<th>Active</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tiger</td>
<td>2</td>
<td>46.5</td>
<td>50.7</td>
<td>33.3</td>
</tr>
<tr>
<td>Shuttle</td>
<td>8</td>
<td>10.0</td>
<td>10.0</td>
<td>2.0</td>
</tr>
<tr>
<td>Gridworld-5</td>
<td>26</td>
<td>33.1</td>
<td>102</td>
<td>21.4</td>
</tr>
<tr>
<td>Hallway</td>
<td>57</td>
<td>1.0</td>
<td>1.0</td>
<td>0.08</td>
</tr>
</tbody>
</table>
Performance on Shuttle Problem

![Graph showing performance comparison between Active Learning, Passive Learning, and No Learning on the shuttle problem.](image)

**Figure 2.** Performance of the non-learner, passive learner, and active learner on the shuttle problem.
Conclusions and Future Works

Conclusion
- Use of Bayes risk for active learning in POMDPs
- Use of meta query to get more information when needed

Future Work
- How reduce the computational expense of finding dialogue model consistent with the meta query?
- More efficient sampling methods
- Heuristics for allocating more computation to more promising solutions
Questions?

1. How to avoid the assumptions?
   - The sets $S$, $A$, and $O$ are known
   - The true model is static

2. Other methods of aiding the dialogue manager when it is confused?

3. How to learn other parameters such as $\lambda$ and $p(Q|M)$?
Thanks!