Bayesian Nonparametric Approaches for Reinforcement Learning in Partially Observable Domains

Finale Doshi-Velez
AAAI Doctoral Consortium 2010
Motivation

Specifying models is often difficult and tedious, yet is needed for lots of problems.

Remote Patient Monitoring
Need models of vital signs.

Assisted Living for the Elderly
Need models of resident behavior
Motivation

Learning as we go helps bias us toward parts of the model needed for good performance.

Remote Patient Monitoring

Common health regime of patient governs what vital sign deviations matter.

Assisted Living for the Elderly

Knowing what activities that a resident enjoys helps detect deviations.
Motivation

Different domains also usually come with various ways of gathering knowledge.

Remote Patient Monitoring

Nurses can suggest what vital signs are likely to be important.

Assisted Living for the Elderly

Caretakers are familiar with needs, personalities of residents.
Goal

Enable agents to learn how to act in partially observable environments without known models.
Goal

Enable agents to learn how to act in partially observable environments without known models.

We also want to assume as little as possible about the model (because specifying models is hard!); models should scale with sophistication of the data. the kinds of information available; agents should be able to combine multiple sources of information.
Formalizing the Problem
General Reinforcement Learning Framework
Defining the World

Assume that the world is a (discrete) partially observable Markov decision process (POMDP)
The Agent's Goal

Learn enough about how the world works to maximize expected discounted rewards

Agent

Representation of how the world works
Belief over world state now

action

observation, reward

State of the world
True world dynamics

World
Challenges

Delayed rewards
Hidden world state
Noisy observations and transitions
Many unknowns to reason about
Many sources of information
Challenges

Delayed rewards
Hidden world state
Noisy observations and transitions
Many unknowns to reason about
Many sources of information

} Lots of RL work
Challenges

Delayed rewards
Hidden world state
Noisy observations and transitions
Many unknowns to reason about
Many sources of information

Our Focus
Expected Contributions

Building flexible models for reinforcement learning in (discrete) partially observable domains with Bayesian nonparametric (BNP) techniques.

Developing inference techniques to manage computational complexity (BNP models generally give good sample complexity).
A Successful Example
Rewards while learning the POMDP Shuttle

Evolution of Total Reward for Shuttle

Final Reward for Shuttle
How Bayesian Nonparametrics Help

Let the agent reason about its uncertainty
Scale sophistication of the model with structure in the data
Incorporate multiple sources of information
How Bayesian Nonparametrics Help

Let the agent reason about its uncertainty

Scale sophistication of the model with structure in the data

Incorporate multiple sources of information

Common to all Bayesian approaches
How Bayesian Nonparametrics Help

Let the agent reason about its uncertainty

Scale sophistication of the model with structure in the data

Incorporate multiple sources of information

Finite-state POMDPs with many (infinite) states, but a strong bias to
Growing Representations: Infinite POMDP (built from iHMM)

Input: actions  
(observed, discrete)

Infinitely many states  
(hidden, discrete)

Output: observations,  
rewards (observed)
Growing Representations: Infinite POMDP (built from iHMM)

Input: actions
(observed, discrete)

Infinitely many states
(hidden, discrete)

but a few popular states
most likely to be visited

Output: observations,
rewards (observed)
Generative Process

First, sample overall popularities, observation and reward distributions for each state.

<table>
<thead>
<tr>
<th>State Index</th>
<th>Overall Popularity (Stick-breaking Process)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>State Index</th>
<th>Observation Distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>State Index</th>
<th>Reward Distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>
Generative Process

Next, sample transition matrix using the state popularities as a base distribution.
Inference

Use beam sampling to efficiently draw samples from our posterior belief over models.

Construct “infinite” model; only visited states have parameters initialized.

Slice “infinite” model to get a finite model.

Run forward-filtering, backward sampling (not EM) on finite model to get visited states.
Planning with Sampled Models
Planning with Sampled Models

models $b(s)$ $a_1$
Planning with Sampled Models
Planning with Sampled Models

Plan in model-space using a basic forward search; each model updates its belief $b(s)$. 

weight of models $b(m)$
Planning with Sampled Models

Plan in model-space using a basic forward search; each model updates its belief $b(s)$.

Leaves use values of beliefs for each model; sampled models are small, quick to solve.
Example Model Learned

Lineworld

Loopworld

Number of States in Lineworld POMDP

Number of States in Loopworld POMDP
Results on Standard Problems
How Bayesian Nonparametrics Help

Let the agent reason about its uncertainty

Scale sophistication of the model with structure in the data

Incorporate multiple sources of information

Policy priors allow us to incorporate knowledge from self-exploration and
Leveraging Expert Trajectories

Often, an expert (could be another planning algorithm) can provide near-optimal trajectories. However, combining expert trajectories with data from self-exploration is challenging:

Experience provides direct information about the dynamics, which indirectly suggests a policy.

Experts provide direct information about the policy, which indirectly suggests dynamics.
Policy Prior Model
(joint work with David Wingate)

- **Policy Prior**: Used FSMs with iPOMDP prior: bias to simple controllers
- **Dynamics Prior**: Used iPOMDP prior: bias to simple models
- **Expert Policy**: Dynamics Model
- **Agent Policy**: Dynamics Model

- **Expert Data**
- **Agent Data**

(joint work with David Wingate)
Inference #1: Model-Based

Biasing lots of agent data with a little expert data
Sample models \( m \) from \( p( m | \text{agent's data} ) \)
Sample policies \( \pi \) from \( p( \pi | \text{expert's data} ) \)
Apply likelihood weights on \( m \) of the form:

\[
p(\pi|m) \propto \exp \left( V_m(\pi) \right)
\]

(Expert likely to provide high-valued policies)
Inference #2: Policy-Based

Fill-in policies from expert data with models based on agent data.

Sample models $m$ from $p( m \mid \text{agent's data} )$

Sample policies $\pi$ from $p( \pi \mid \text{expert's data} )$

Update policies $\pi$ with models $m$ (one step of bounded policy iteration).
Results on Several Problems
Vision for Future Work

Improved planning and inference

Fully online inference for the iPOMDP

Planning algorithms for deeper search in model space

Incorporating more expert information, such as written instructions.

More structured nonparametric model priors, such as a factored or first-order iPOMDP.

Applying models to healthcare domains.
IPOMDP: Different Planning Approaches
Policy Prior: What it means

Mass of models with simple dynamics

Model Space
Policy Prior: What it means

Mass of models with simple control policies.
Policy Prior: What it means

Joint Prior: models with few states, also easy to control.

Model Space