6.882 Proposal: Bayesian Reinforcement Learning

Vickie Ye and Alexandr Wang

1 Background and Motivation

Because of the recent excitement regarding DeepMind's AlphaGo AI, we wanted to explore reinforcement learning, in a Bayesian framework. We will use the Strens's ICML-2000 paper, *A Bayesian Framework for Reinforcement Learning* and Engel's ICML-2005 paper, *Reinforcement learning with Gaussian processes* as guidelines as to how to reason about the reinforcement learning problems with Bayesian formulations.

We will study a toy reinforcement learning problem of navigating an unknown maze. In the reinforcement learning problem, the agent must explore the surroundings and determine the behavior that maximizes the expected return. In this case, our agent must discover the structure of the maze while minimizing the time required to find the end. A Markov decision process (MDP) model is traditionally used to model the interactive system over S the set of states, A the set of actions, $R : S \times A \to \mathbb{R}$ the reward function, and T the transition probabilities defined as

$$T(s, a, s') = P(X_{t+1} = s' | X_t = s, Y_t = a). \tag{1}$$

We also define a quality function Q as

$$Q(s, a) = \mathbb{E}[R(s, a)] + \gamma \sum_{s'} \max_{a'} Q(s', a') \tag{2}$$

where γ is the time-discount factor (Strens, 2000, p.3).

Our optimal behavior is then the policy that maximizes the quality function. Many approaches for finding the best policy uses dynamic programming in order to estimate the transition probabilities. The maximum likelihood estimate for $T(s, a, s')$ is the proportion of times the action a in state s leads to s', and the estimate of $\mathbb{E}[R(s, a)]$ is the average reward received whenever (s, a) is taken. This requires the keeping track of all pairs of states and actions - storage of large sparse data. In Strens (2000), the author places a prior over and estimates the posterior transition probabilities. At each time step, the author uses the current model over the system to estimate the quality function Q^* and choose the best action.

We will also use Engel (2005) as a guideline for implementing Bayesian policy evaluation using Gaussian processes. The algorithm presents a Bayesian solution to policy evaluation termed Gaussian Process Temporal Difference (GPTD). It then extends the algorithm to a State-Action-Reward-State-Action (SARSA) based extension of GPTD termed GPSARSA which is a fully functioning agent that can select actions and gradually improve its policies.

2 Plans and Consideration

For the two of us, our plan to implement these models will be in the following tasks:

1. Implement Bayesian MDPs as described in Strens (2000). This includes implementing the representation of the posterior, generating the prior, and posterior updates (Vickie).
2. Implement hypothesis generation using the posteriors over the MDP parameters as described in Strens (2000) (Vickie).
3. Replicate paper results in three applications discussed in the paper ("Chain", "Loop", and "Maze" toy problems) (Alex).
4. Apply implementation to our maze problem and compare to non-Bayesian methods (in particular, traditional DPs) (Vickie).
5. Implement on-line Monte Carlo GPTD algorithm as described in Engel (2005) (Alex and Vickie).
6. Implement full GP State-Action-Reward-State-Action algorithm (GPSARSA) described in Engel (2005), an extension of the GPTD which implements a fully learning agent (Alex).
7. Apply implementation of GPSARSA to our maze problem and compare to our Bayesian DP method and non-Bayesian methods from before (Alex).

The major risks of this project come from successfully implementing the GP algorithms Engel (2005). The algorithm is relatively complicated and involves learning of many parameters, and therefore is very prone to bugs. In general, it seems difficult to test the performance of our algorithms modularly without testing the performance of our agent as a whole. Since our reinforcement learning methods involve multiple techniques working together, we expect challenges in clean, efficient implementations that we can properly test on our toy problems. We anticipate that this proposal contains more work than we will be able to complete, and plan to reevaluate our goals at the halfway point in the project.

References

Engel, Y. (2005). Reinforcement learning with gaussian processes. In *International Conference on Machine Learning*.

Strens, M. (2000). A bayesian framework for reinforcement learning. In *International Conference on Machine Learning*.