Reinforcement Learning with Misspecified Bayesian Nonparametric Model Classes

Joshua Joseph, Alborz Geramifard, Jonathan P. How and Nicholas Roy
Massachusetts Institute of Technology
77 Massachusetts Ave., Cambridge, MA 02139 USA
{jmjoseph, agf, jhow, nickroy}@mit.edu

Abstract

Decision making in complex, real-world domains often involves a model of the world dynamics. When this model is unknown, we commonly fit a model from a model class based on past data. Unfortunately, the class of models used to capture the world dynamics is often misspecified, or unable to capture the true dynamics. Although Bayesian nonparametric models are often turned to for difficult modeling problems, they are still vulnerable to this problem of misspecification. In [1], we introduced Reward Based Model Search (RBMS), an approach for learning misspecified parametric models and demonstrated its effectiveness over the standard maximum likelihood model selection metric. In this work we extend RBMS to Bayesian nonparametric models.

1 Introduction

Planning in real-world domains commonly involves some form of dynamics model, to predict the future evolution of the world. Often in these domains, the dynamics model is unknown and is therefore learned from data. To enable learning, a designer typically specifies a model class from which a model is chosen based on the collected data. Unfortunately, these representation classes often can only approximate the true dynamics. In these cases, we call the model class misspecified.

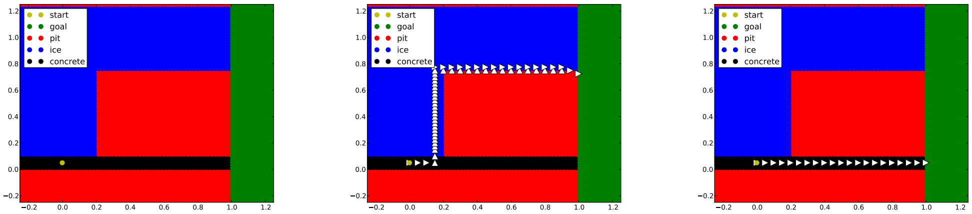
Previously, in [1], we have worked to overcome misspecification for reinforcement learning [2, 3] with parametric models. To accomplish this, we introduced an algorithm called Reward Based Model Search (RBMS), which explicitly chooses the model, whose policy, results in the highest performance. This is in contrast to the common approach of maximum likelihood (ML) model selection which selects the model that best explains the data, without consideration of the planning problem being solved.

Recently, Bayesian nonparametric models (BNMs) have been used extensively in real-world domains [4, 5, 6, 7]. Although BNM classes are generally more flexible than standard parametric models, they are still vulnerable to the problem of misspecification. In this work, we extend the RBMS algorithm to BNM classes.

2 Parametric Reward Based Model Search

In this section we briefly describe Reward Based Model Search (RBMS) [1]. RBMS takes a batch of training data and returns a policy which it estimates to achieve the highest return. To determine the highest return policy, there are two main components of the algorithm: the policy evaluation step and the model improvement step.

For policy evaluation, RBMS uses a method called Model-Free Monte Carlo (MFMC) [8]. MFMC first pieces together on-policy episodes of data from off-policy batch data. The MFMC algorithm



(a) (b) (c)

Figure 1: The domain (a) and the episode of data produced by the policy from the MAP model (b) and the RBMS model (c).

then estimates the return of the policy by averaging the return of each of episode, similar to standard Monte Carlo policy evaluation [3].

The model improvement step is performed by gradient ascent in the model classes' parameter space, which attempts to maximize return by adjusting the model. In [1] we observed that although this optimization is not only non-convex but discontinuous, standard gradient ascent with random restarts proved sufficient..

3 Bayesian Nonparametric Reward Based Model Search

The difficulty in applying RBMS to BNM classes lies in the model improvement step. In BNMs, the model is not only a function of some hyperparameters, but also of the data, so it is not clear what taking the gradient in the model classes' parameter space would mean for BNM classes. The purpose of the model improvement step is to search through the model class in the direction of increasing return. Therefore, following this purpose, we propose three approaches for searching through the model class: removing data from the model, sampling new data from the model, and treating the data as parameters themselves to be adjusted. Note that these approaches may be used together or separately. These are in addition to using the standard gradient technique with the BNM classes' hyperparameters, which can also be used to search through the space.

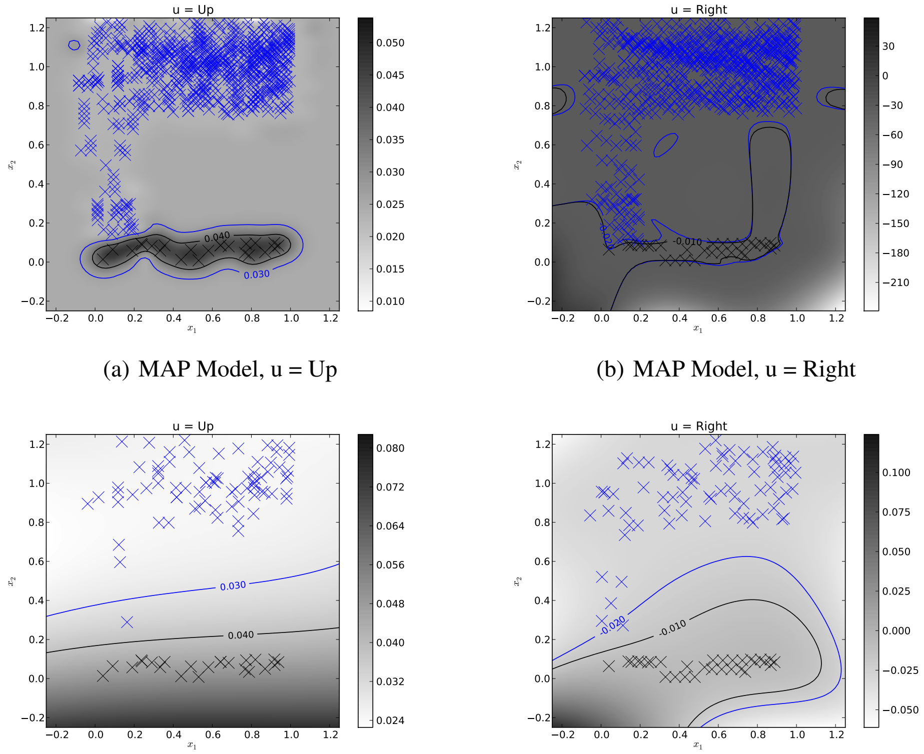
In Section 4 we experimentally test RBMS with the approach of removing data from the model for model improvement with BNM classes. We leave it to future work to further explore the other two approaches and the trade-offs between all the approaches and their combinations

4 Results

We empirically validated RBMS for Bayesian nonparametric models on the domain shown in Figure 1(a). In the domain the agent starts at the yellow point and uses available actions $\{up, right\}$ to try and reach the goal (green) while avoiding falling in any of the pits (red). The agent experiences two different dynamics across the world, concrete (black) and ice (blue). On the concrete the agent's actions achieve their desired outcome and on the ice the agent will move in the chosen direction but will also "slide" south. The dynamics are deterministic and the reward function is -1 for any action, -100 for falling in a pit, and the episode ends when the agent either falls in a pit or reaches the goal. One hundred episodes of training data were generated by randomly starting an agent in a state and randomly choosing actions until the episode ended.

To model the dynamics, two separate Gaussian processes (GPs) were used to model the transitions for the up action and the $right$ action. We used the standard squared exponential form of the covariance function for both GPs. Figures 1(b) and 1(c) show the episode of data generated by the policy for the maximum a posteriori (MAP) model and the RBMS model, respectively. For the RBMS model improvement step we used the strategy of stochastically removing data, which proved sufficient.

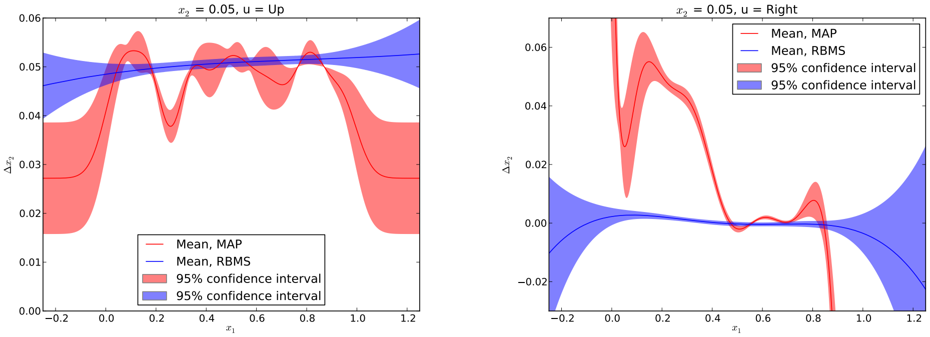
Figures 2 is a visualization of the mean for both the learned MAP model and RBMS model. Data points included in each model are shown as Xs and colored blue if they were on the ice and black if



(a) MAP Model, u = Up (b) MAP Model, u = Right

(c) RBMS Model, u = Up (d) RBMS Model, u = Right

Figure 2: The GPs' mean of the MAP model (a,b) and the MBRS model (c,d) for both actions.



(a) MAP and RBMS Models, u = Up, $x_2 = 0.05$ (b) MAP and RBMS Models, u = Right, $x_2 = 0.05$

Figure 3: The GPs' means and confidence intervals with $x_2 = 0.05$.

on the concrete. The sharp contour of the MAP GP around $x_2 = 0.1$ shows each GP attempting to cope with the discontinuity in the dynamics, in contrast to the smooth dynamics found by RBMS.

Figures 3 shows the GPs' means and confidence intervals, plotted over x_1 with $x_2 = 0.05$ to show the learned variances of the models in the center of the concrete. These figures demonstrate how the GPs, which assume the dynamics are smooth, cannot cope with the discontinuity at $x_2 = 0.1$. In other words, the discontinuity of the dynamics violates the implicit assumption of the GPs' covariance function and leads to their misspecification. RBMS mitigated this problem by searching for the model which achieved the highest return, not by attempting to fit the data well.

5 Conclusion and Future Work

In this work we presented a Bayesian nonparametric extension of Reward Based Model Search (RBMS), a method for learning misspecified Bayesian nonparametric models. We demonstrated experimentally the potential benefit for using RBMS to fit Bayesian nonparametric models on a simulated domain with discontinuous dynamics. As discussed in Section 3, there is an open question regarding how to perform the model improvement step. While we suggested three potential methods for performing model improvement, and implemented one of them, a great deal of future work is need to understand both the sample and computational complexity trade-offs between them.

References

[1] J. Joseph, A. Geramifard, J. W. Roberts, J. P. How, and N. Roy, "Reinforcement learning with misspecified model classes," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2013)*, Under Review.

[2] L. P. Kaelbling, M. L. Littman, and A. P. Moore, "Reinforcement Learning: A Survey," *Journal of Artificial Intelligence Research*.

[3] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. MIT Press, May 1998.

[4] E. Fox, E. Sudderth, M. Jordan, and A. Willsky, "A Sticky HDP-HMM with Application to Speaker Diarization," *Annals of Applied Statistics*, vol. 5, no. 2A, pp. 1020–1056, 2011.

[5] Y. W. Teh and M. I. Jordan, "Hierarchical Bayesian nonparametric models with applications," in *Bayesian Nonparametrics: Principles and Practice*, N. Hjort, C. Holmes, P. Müller, and S. Walker, Eds. Cambridge University Press, 2010.

[6] J. Joseph, F. Doshi-Velez, A. S. Huang, and N. Roy, "A Bayesian Nonparametric Approach to Modeling Motion Patterns," *Autonomous Robots*, vol. 31, no. 4, pp. 383–400, 2011.

[7] J. Joseph, F. Doshi-Velez, and N. Roy, "A bayesian nonparametric approach to modeling battery health," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2012)*, St Paul, MN, 2012.

[8] R. Fonteneau, S. A. Murphy, L. Wehenkel, and D. Ernst, "Model-free monte carlo-like policy evaluation," *Journal of Machine Learning Research - Proceedings Track*, vol. 9, pp. 217–224, 2010.