Robust and Efficient Transfer Learning with Hidden Parameter Markov Decision Processes

Taylor W. Killian
Harvard University
Cambridge, MA 02148
taylorkillian@g.harvard.edu

George Konidaris
Brown University
Providence, RI 02912
gdk@cs.brown.edu

Finale Doshi-Velez
Harvard University
Cambridge, MA 02148
finale@seas.harvard.edu

Abstract

An intriguing application of transfer learning arises among tasks with similar, but not identical, dynamics. Hidden Parameter Markov Decision Processes (HiP-MDP) embed these tasks into a low-dimensional space; given the embedding parameters one can identify the MDP for a particular task. However, the original formulation of HiP-MDP had a critical flaw: the embedding uncertainty was modelled independently of the agent's state uncertainty, requiring an unnatural training procedure in which all tasks visited every part of the state space. In this work, we apply a Gaussian Process latent variable model to jointly model the dynamics and the embedding, leading to both a more elegant formulation and one that allows for better uncertainty quantification and thus more robust transfer. We demonstrate an initial promising result that our correction behaves as expected and illustrate it's use on three domains: acrobot, as well as HIV and a diabetes simulators.

Introduction

In a multitude of decision and control problems, there are instances where subtle variations in the underlying physical system can introduce a broad range of dynamics. These variations in unobserved and observed representations of the system can contribute to inefficiencies or, in some dramatic cases, failure in an agent's ability to learn an optimal control policy. This is particularly true when the agent is trained from data that does not account for unexpected variations.

With the growing availability of data sets generated by similar, but not identical, processes (e.g. healthcare, sensing networks, robotics) there is a compelling need to develop learning frameworks that include and account for system variations in an efficient and robust manner. In order to develop optimal treatment or control policies, it is undesirable and ineffectual to start afresh each time a new instance is encountered. Ideally, an agent tasked with developing an optimal control policy would be able to leverage the similarities across separate, but related, instances. This paradigm of learning introduces an intriguing use case for transfer learning.

The Hidden Parameter Markov Decision Process (HiP-MDP) (Doshi-Velez and Konidaris 2013) was introduced as a formalization of these domains with two primary features. **First**, that a bounded number of latent parameters, w, for a single task instance can fully specify the system dynamics, θ, if learned. That is, the dynamics dictating a system's transition between states can be expressed as $T(s'|s, a, \theta_b)$ for instance b. **Second**, that the system dynamics will not change during a task and an agent would be capable of determining when a change occurs. The HiP-MDP was shown to be able to rapidly identify the dynamics of a new task instance and flexibly adapt to the variations present therein. However, the original HiP-MDP formulation had a critical flaw: the embedding uncertainty of the latent parameter space was modelled independently from the agent's state uncertainty. This created an inefficient training procedure, requiring the agent to canvas the state space before identifying the variations present in the dynamics of the current instance.

We present an update to the original HiP-MDP that allows for more efficient training by embedding the latent parametrization in the observed data via a Gaussian Process latent variable model (GPLVM). This approach creates a unified Gaussian Process GP) model for both inferring the transition dynamics within a task instance but also in the transfer between task instances (Cao et al. 2010). Steps are taken to avoid negative transfer by selecting the most representative examples of the prior instances with regards to the latent parameter setting. This change in the model allows for better uncertainty quantification and thus more robust and direct transfer. We ground our approach with recent advances in the use of GP to approximate dynamical systems and in transfer learning. We then formalize the adjustments to the HiP-MDP framework and present the performance of the adjusted HiP-MDP on developing control policies for the acrobot domain, as well as HIV and Diabetes simulators.

Related Work

Inference with GP GP have increasingly been used to facilitate methods of Reinforcement Learning (RL) (Rasmussen and Kuss 2003),(Rasmussen and Williams 2006). Recent advances in modeling dynamical systems with GP have led to more efficient and robust formulations (Deisenroth and Rasmussen 2015),(Deisenroth and Rasmussen 2011), most particularly in the approximation and simulation of dynamical systems. The HiP-MDP approximates the underlying

dynamical system of the task through the training of a Gaussian Process dynamical model (Deisenroth and Mohamed 2012),(Wang, Fleet, and Hertzmann 2005) where only a small portion of the true system dynamics may be observed as is common in partially observable Markov Decision Processes (POMDP) (Kaelbling, Littman, and Cassandra 1998). In order to facilitate the transfer between task instances we embed a latent, low-dimensional parametrization to the states. By virtue of the GP (Lawrence 2004),(Urtasun and Darrell 2007), this latent embedding allows the HiP-MDP to infer across similar task instances and provide a better prediction of the currently observed system.

GP in Transfer Learning The use of GP to facilitate the transfer of previously learned information to new instances of the same or a similar task has a rich history (Bonilla, Chai, and Williams 2008)(Kaelbling, Littman, and Cassandra 1998),(Rasmussen and Kuss 2003). More recently, there have been advances in organizing how the GP is used to transfer, being constrained to only select previous task instances where positive transfer occurs (Cao et al. 2010),(Leen, Peltonen, and Kaski 2011). This adaptive approach to transfer learning helps to avoid previous instances that would otherwise negatively affect effective learning in the current instance. By selecting the most relevant instances of a current task for transfer, learning in the current instance becomes more efficient.

RL in healthcare The use of RL (and machine learning, in general) for the development of optimal control policies and decision making strategies in healthcare (Shortreed et al. 2011) is gaining significant momentum as methodologies have begun to adequately account for uncertainty and variations in the problem space. There have been notable efforts made in the administration of anesthesia (Moore et al. 2014), in personalizing cancer (Tenenbaum et al.) and HIV therapies (Ernst et al. 2006) and in understanding the causality of macro events in diabetes managment (Merck and Kleinberg 2015). Also, recent progress has been made to formalize routines to accommodate multiple sources of uncertainty in batch RL methods to better evaluate the effectiveness of treatments across subpopulations of patients (Marivate et al. 2014). We similarly attempt to address and identify the variations across subpopulations as well as the uncertainty present in the development treatment policies. We instead, attempt to account for these variations while developing effective treatment policies in an approximate online fashion.

A HiP-MDP with Joint Uncertainty

The HiP-MDP is described by a tuple: $\{S, A, \Theta, T, R, \gamma, P_\Theta\}$, where S and A are the sets of states s and actions a, and $R(s, a)$ is the reward function mapping the utility of taking action a from state s. The transition dynamics $T(s'|s, a, \theta_b)$ for each task instance b depends on the value of the hidden parameters $\theta_b \in \Theta$. Where the set of all possible parameters θ_b is denoted by with Θ and where P_Θ is the prior over these parameters. And finally, $\gamma \in (0, 1]$ is the factor by which R is discounted to express how influential immediate rewards are when learning a control policy. Thereby, the HiP-MDP describes a *class* of tasks; where particular instances of that class are obtained by independently sampling a parameter vector $\theta_b \in \Theta$ at the initiation of a new task instance b. We assume that θ_b is invariant over the duration of the instance, signaling distinct learning frontiers between instances when a newly drawn $\theta_{b'}$ accompanies observed additions to S and A.

The HiP-MDP presented in (Doshi-Velez and Konidaris 2013) provided a transition model of the form:

$$(s'_d - s_d) \sim \sum_k^K z_{kad} w_{kb} f_{kad}(s) + \epsilon$$
$$\epsilon \sim \mathcal{N}(0, \sigma^2_{nad})$$

which sought to learn weights w_{kb} based on the k^{th} latent factor corresponding to task instance b, filter parameters $z_{kad} \in \{0, 1\}$ denoting whether the k^{th} latent parameter is relevant in predicting dimension d when taking action a as well as task specific basis functions f_{kad} drawn from a GP. While this formulation is expressive, it presents a problematic flaw when trained. Due to the independence of the weights w_{kb} from the basis functions f_{kad}, training the HiP-MDP requires canvassing the state space S in order to infer the filter parameters z_{kad} and learn the instance specific weights w_{kb} for each latent parameter.

We bypass this flaw by applying a GPLVM (Lawrence 2004) to jointly represent the dynamics and the latent weights w_b corresponding to a specific task instance b. This leads to providing as input to the GP, with hyperparameters ψ, the augmented state $\tilde{s} =: [s^\intercal,\ a,\ w_b]^\intercal$. The approximated transition model then takes the form of:

$$s'_d \sim f_d(\tilde{s}) + \epsilon$$
$$f_d \sim GP(\psi)$$
$$w_b \sim \mathcal{N}(\mu_b, \Sigma_b)$$
$$\epsilon \sim \mathcal{N}(0, \sigma_{bd})$$

This approach enables the HiP-MDP to flexibly infer the dynamics of a new instance by virtue of the statistical similarities found in the learned covariance function between observed states of the new instance and those from prior instances. Another feature of formulating the HiP-MDP after this fashion is that we are able to leverage the marginal log likelihood of the GP to optimize the weight distribution and thereby quantify the uncertainty (Candela 2004),(Candela et al. 2003) of the latent embedding of w_b for θ_b. These two features of reformulating the HiP-MDP as a GPLVM allows for more robust and efficient transfer.

Inference

Parameter Learning and Updates We deploy the HiP-MDP when the agent is provided a large amount of batch observational data from several task instances and tasked with quickly performing well on new instances. With this observational data the GP transition functions f_d are learned and the individual weighting distributions for w_b are optimized.

However, the training of the f_d requires computing inverses of matrices of size $N = \sum_b n_b$ where n_b is the number of data points collected from instance b. To streamline the approximation of T we choose a set of support points s^* from S_b that sparsely approximate the full GP. Optimization procedures exist to select these points accurately (Snelson and Ghahramani 2005), (Rasmussen and Williams 2006) we however heuristically select these points to minimize the maximum reconstruction error within each batch.

Control Policy A control policy is learned for each task instance b following the procedure outlined in (Deisenroth and Rasmussen 2011) where a set of tuples $(s,\ a,\ s',\ r)$ are observed and the policy is periodically updated (as is the latent embedding w_b) in an online fashion, leveraging the approximate dynamics of T via the f_d^* to create a synthetic batch of data from the current instance. This generated batch of data from b is then used to improve the current policy via fitted-Q iteration (Ernst, Geurts, and Wehenkel 2005). Multiple episodes are run from each instance b to optimize the policy for completing the task under the hidden parameter setting θ_b. After doing so, the hyperparameters of the GP defining the f_d are updated before learning for another randomly manifest task instance.

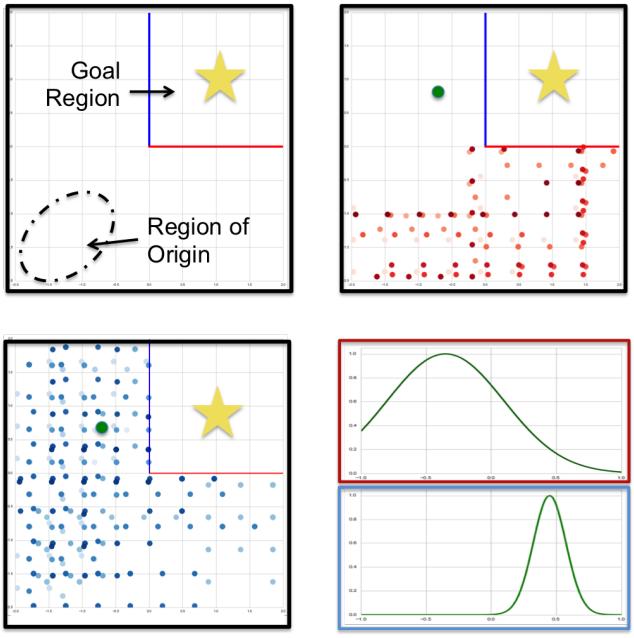

Figure 1: Toy Problem: (a) Schematic outlining the domain, (b) learned policy for "red" parametrization, (c) learned policy for "blue" parametrization, (d) uncertainty measure for input point according to separate latent classes.

Demonstration We demonstrate a toy example (see Figure 1) of a domain where an agent is able to learn separate policies according to a hidden latent parameter. Instances inhabiting a "blue" latent parametrization can only pass through to the goal region over the blue boundary while those with a "red" parametrization can only cross the red boundary. After a few training instances, the HiP-MDP is able to separate the two latent classes and develops individualized policies for each. We place an unclassified survey point in the top left quadrant, with a proposed action to move to the right, to gather information about the policy uncertainty given the two latent classes.

Future Experiments and Model Adjustments

Experiments using the HiP-MDP formulation

We highlight here the separate domains on which we will apply the adjusted HiP-MDP framework and procedure presented above. In all domains, we summarize the entire system with the tuple: $\{S, A, \Theta, T, R, \gamma, P_\Theta\}$ and apply fiited-Q iterations (Ernst, Geurts, and Wehenkel 2005) on synthesized batch data derived from limited observations of the true dynamical system.

Baselines We aim to benchmark the HiP-MDP framework in the HIV, Diabetes and Acrobot domains by observing how an agent would perform without transferring information from prior patients to aid in the efficient development of the treatment policy for a current patient. We do this by representing two ends of the precision medicine spectrum; a "one-size-fits-all" approach that learns a single treatment policy for all patients by using all previous patient data together and a "personally tailored" treatment plan where a single patient's data is all that is used to train the policy. We represent these baselines in environments where a model is present (with the simulators) or absent (utilizing the GP approximation).

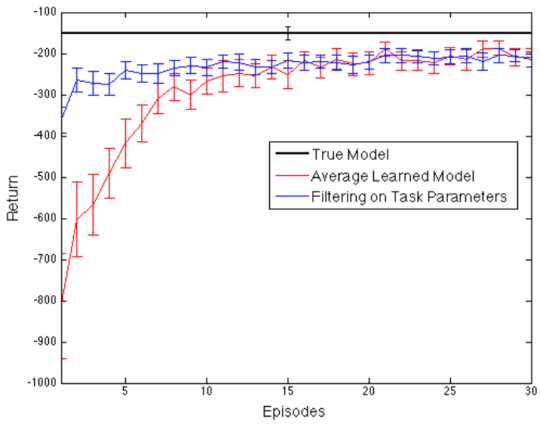

Figure 2: Desired baseline relationship between "one-size-fits-all" policy and HiP-MDP learned policy, copied from (Doshi-Velez and Konidaris 2013)

HIV Ernst, et.al. (Ernst et al. 2006) leverage the mathematical representation of how a patient responds to HIV treat-

ments (Adams et al. 2004) in developing an RL approach to find effective treatment policies using fitted-Q iteration. The learned treatment policies cycle on and off two different types of anti-retroviral medication in a sequence that maximizes long-term health.

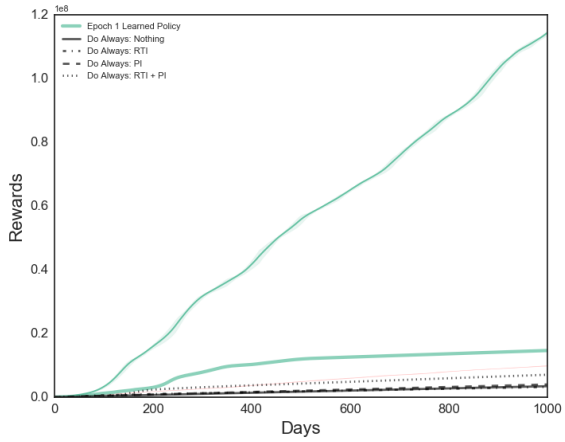

Figure 3: Example of gains made in the HIV domain in the fitted-Q optimized treatment policy over naive treatment baselines and randomized policy.

Diabetes Merck and Kleinberg (Merck and Kleinberg 2015) developed a model within which they could infer how a patient with type I diabetes responds to different environmental stimuli alongside the intrinsic glucose-insulin process. This model was developed to study causality in diabetes management while we adopt it to train an agent to effectively balance glucose and insulin levels over the course of a few hours. We assume that the patient heeds the direction an agent gives. The agent has the ability to suggest glucose intake or insulin injections as it determines to be appropriate.

Acrobot The acrobot, introduced in (Sutton and Barto 1998), features a double-pendulum. The agent can apply a positive, negative, or neutral torque to the hinge joint between the two legs of the pendulum. The goal is to apply a combination of torques in succession so as to swing the foot of the pendulum above a specified height above the hinge at the top of the pendulum.

Making the HiP-MDP more efficient and robust

There has been significant progress toward formalizing a more robust HiP-MDP by jointly modeling the state and latent embedding uncertainties. However, we have encountered significant computational and run-time difficulties when accounting for the full-data GP. We have begun investigating two approaches to bypass this bottleneck when developing a policy for a current task instance, based on previous data.

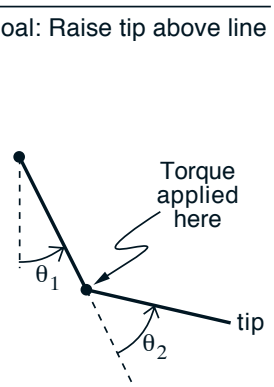

Figure 4: Schematic of Acrobot Domain, pulled from (Sutton and Barto 1998).

Using Bayesian Neural Networks for the Latent Variable Model In the near future we will be utilizing multiple approaches to make the HiP-MDP more efficient, most particularly in the transfer between prior task instances, and be enabled to fully quantify the uncertainty in the transfer between task instances. The computational load to train and infer from the GPLVM severely limits the amount of data one can use from prior task instances when training the current instance. This causes an incomplete transfer between instances and impacts the robustness of the HiP-MDP. To account for this and to more efficiently provide updates to the learned policy we aim to transition away from a GP-based approximation of the dynamics and adopt Bayesian Neural Networks. This added efficiency will allow us to perform more detailed testing within the domains presented here and to more accurately estimate the uncertainty in the latent weight distributions.

Adaptive Transfer Learning We also aim to introduce a non-parametric approach to selecting which of the previous instances to select as a representative set to use for transferring to the current instance. Currently we are using a hard-coded heuristic to accomplish this task by measuring the similarity between the latent representations (w) of each instance and choosing the closest few for transfer. We hope to train a separate "scheduler" that can identify features between task instances and can thus choose which previous examples are most relevant for our current example. This approach will rely heavily on the work done in (Cao et al. 2010) and (Leen, Peltonen, and Kaski 2011).

Acknowledgments

TWK is supported as a Lincoln Scholar from MIT Lincoln Laboratory, located in Lexington, Massachusetts. We are grateful to Christopher Merck and Samantha Kleinberg for providing the simulator used in the Diabetes domain.

References

Adams, B.; Banks, H.; Kwon, H.; and Tran, H. 2004. Dynamic multidrug therapies for hiv: optimal and sti control approaches. *Mathematical Biosciences and Engineering* 223–241.

Bonilla, E.; Chai, K.; and Williams, C. 2008. Multi-task Gaussian Process Prediction. In *Advances in Neural Information Processing Systems*, volume 20, 153–160.

Candela, J. Q.; Girard, A.; Murray-Smith, R.; and Rasmussen, C. 2003. Propagation of uncertainty in bayesian kernel models - application to multiple-step ahead time series forecasting. In *Proceedings of the ICASSP*, 701–704.

Candela, J. Q. 2004. *Learning with uncertainty - gaussian processes and relevance vector machines*. Ph.D. Dissertation, Technical University of Denmark.

Cao, B.; Pan, S.; Zhang, Y.; Yeung, D.; and Yang, Q. 2010. Adaptive transfer learning. In *AAAI*, volume 2, 7.

Deisenroth, M., and Mohamed, S. 2012. Expectation Propagation in Gaussian Process Dynamical Systems. In *Advances in Neural Information Processing Systems*, volume 25, 2618–2626.

Deisenroth, M., and Rasmussen, C. 2011. Pilco: A model-based and data-efficient approach to policy search. In *In Proceedings of the International Conference on Machine Learning*.

Deisenroth, M., and Rasmussen, C. 2015. Gaussian processes for data-efficient learning in robotics and control. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37.

Doshi-Velez, F., and Konidaris, G. 2013. Hidden parameter markov decision processes: A semiparametric regression approach for discovering latent task parametrizations. *CoRR* abs/1308.3513.

Ernst, D.; Stan, G.; Goncalves, J.; and Wehenkel, L. 2006. Clinical data based optimal sti strategies for hiv; a reinforcement learning approach. In *Proceedings of the 45th IEEE Conference on Decision and Control*.

Ernst, D.; Geurts, P.; and Wehenkel, L. 2005. Tree-based batch mode reinforcement learning. *Journal of Machine Learning Research* 6(Apr):503–556.

Kaelbling, L.; Littman, M.; and Cassandra, A. 1998. Planning and acting in partially observable stochastic domains. *Artificial intelligence* 101(1):99–134.

Lawrence, N. 2004. Gaussian process latent variable models for visualisation of high dimensional data. In *Advances in Neural Information Processing Systems*, volume 16, 329–336.

Leen, G.; Peltonen, J.; and Kaski, S. 2011. Focused multi-task learning using gaussian processes. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 310–325. Springer.

Marivate, V.; Chemali, J.; Brunskill, E.; and Littman, M. 2014. Quantifying uncertainty in batch personalized sequential decision making. In *Workshops at the Twenty-Eighth AAAI Conference on Artificial Intelligence*.

Merck, C., and Kleinberg, S. 2015. Causal explanation under indeterminism: A sampling approach.

Moore, B.; Pyeatt, L.; Kulkarni, V.; Panousis, P.; Padrez, K.; and Doufas, A. 2014. Reinforcement learning for closed-loop propofol anesthesia: a study in human volunteers. *Journal of Machine Learning Research* 15(1):655–696.

Rasmussen, C., and Kuss, M. 2003. Gaussian processes in reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 15.

Rasmussen, C., and Williams, C. 2006. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge.

Shortreed, S.; Laber, E.; Lizotte, D.; Stroup, T.; Pineau, J.; and Murphy, S. 2011. Informing sequential clinical decision-making through reinforcement learning: an empirical study. *Machine learning* 84(1-2):109–136.

Snelson, E., and Ghahramani, Z. 2005. Sparse gaussian processes using pseudo-inputs. In *Advances in Neural Information Processing Systems*, volume 17, 1257–1264.

Sutton, R., and Barto, A. 1998. *Reinforcement learning: An introduction*, volume 1. MIT Press, Cambridge.

Tenenbaum, M.; Fern, A.; Getoor, L.; Littman, M.; Manasinghka, V.; Natarajan, S.; Page, D.; Shrager, J.; Singer, Y.; and Tadepalli, P. Personalizing cancer therapy via machine learning.

Urtasun, R., and Darrell, T. 2007. Discriminative gaussian process latent variable model for classification. In *Proceedings of the 24th International Conference on Machine learning*, 927–934. ACM.

Wang, J.; Fleet, D.; and Hertzmann, A. 2005. Gaussian Process Dynamical Models. In *Advances in Neural Information Processing Systems*, volume 17, 1441–1448.