Context model inference for large or partially observable MDPs

Christos Dimitrakakis CHRISTOS.DIMITRAKAKIS@GMAIL.COM
Frankfurt Institute for Advanced Studies

Abstract

We describe a simple method for exact online inference and decision making for partially observable and large Markov decision processes. This is based on a closed form Bayesian update procedure for certain classes of models exhibiting a special conditional independence structure, which can be used for prediction, and consequently for planning.

1. Introduction

We consider estimation of a class of context models that can approximate large or partially observable Markov decision processes. This is closely related to the context tree weighting algorithm for discrete sequence prediction (Willems et al., 1995). We present a *constructive* definition of a context process, extending the one proposed in (Dimitrakakis, 2010a) for the estimation of variable order Markov models, and apply it to prediction, state representation and planning in partially observable Markov decision processes.

We consider discrete-time decision problems in unknown environments, with a known set of actions $\mathcal{A}$ chosen by the decision maker, and a set of observations $\mathcal{Z}$ drawn from some unknown process μ, to be made precise later. At each time step $t \in \mathbb{N}$, the decision maker observes $z_t \in \mathcal{Z}$, selects an action $a_t \in \mathcal{A}$ and receives a reward $r_t \in \mathbb{R}$.

The environment μ is a (partially observable) *Markov decision process* ((PO)MDP) with state $s_t \in \mathcal{S}$. The process is defined by the following conditional distributions: the set of transition and reward distributions $\mathcal{T}_\mu \triangleq \{Pr_\mu(s_{t+1} \mid s_t = i, a_t = j) : i \in \mathcal{S}, j \in \mathcal{A}\}$ and $\mathcal{R}_\mu \triangleq csetPr_\mu(r_{t+1} \mid s_t = i, a_t = j) i \in \mathcal{S}, j \in \mathcal{A}$. For POMDPs, observations z_t are sampled from $\mathcal{O}_\mu^{i,j} \triangleq \mathbb{P}_\mu(z_{t+1} \mid s_t = i, a_t = j)$. For MDPs, $\mathcal{Z} = \mathcal{S}$ and $z_t = i$ iff $s_t = i$.

Submitted to the ICML 2010 Workshop on Reinforcement Learning and Search in Very Large Spaces.

The decision maker has a policy π for choosing actions, which indexes a set of probability measures on actions. Jointly π and μ index a set of probability measures $\mathbb{P}_{\mu,\pi}(z_{t+1}, s_{t+1}, r_{t+1}, a_t \mid s_t)$ on actions, states, rewards and observations. The goal is to find a policy π maximising the expected utility:

$$U_t \triangleq \sum_{k=1}^{T-t} \gamma^k r_{t+k}. \qquad \text{(utility)}$$

The decision maker is usually uncertain about the true MDP μ. We adopt a subjective decision-theoretic viewpoint (DeGroot, 1970) and assume a set $\mathcal{M}$ of MDPs contains μ, then define a *prior* probability measure ξ_0 on $(\mathcal{M}, \mathfrak{B}_\mathcal{M})$, such that for any $M \in \mathfrak{B}_\mathcal{M}$:

$$\xi_{t+1}(M) \triangleq \xi_t(M \mid z_{t+1}, r_{t+1}, a_t, z_t) \qquad (1)$$

is defined for all t and sequences of s_t, a_t, r_t. We now must find a policy π maximising:

$$\mathbb{E}_{\xi_t,\pi} U_t = \int_\mathcal{M} \mathbb{E}_{\mu,\pi}(U_t) \xi_t(\mu) \, \mathrm{d}\mu, \qquad (2)$$

the expected utility under our current belief ξ_t. The decision problem can be seen as an MDP whose state space is the product of $\mathcal{S}$ and the set of probability measures on $(\mathcal{M}, \mathfrak{B}_\mathcal{M})$. However, since there is an infinite number of beliefs, approximations are required even under full observability (Duff, 2002; Wang et al., 2005; Dimitrakakis, 2009). Nevertheless, such methods are also extensible to the partially observable case (Veness et al., 2009; Ross et al., 2008).

When dealing with large or partially observable MDPs, even (1) is not closed-form. In this paper, we extend a specific formulation of variable order Markov model estimation (Dimitrakakis, 2010a) to variable order *or* large MDPs. We experimentally show that this can not only provide accurate predictions, but that the internal state of the process closely tracks the state of the system, even though no explicit state estimation is being performed. This can be used to implement standard reactive learning algorithms, value iteration, or even decision-theoretic planning, as was done in (Veness et al., 2009; Ross et al., 2008)

2. Inference for context MDPs

One can use context models to perform closed-form inference for either discrete variable order MDPs or for continuous MDPs. This is done by constructing a context graph, such that for each observation history $x^t = (x_k)_{k=1}^t$, there exists a set of contexts forming a chain on the graph. We can then perform a walk which stops with probability w_k^t on the k-th node of the chain, and generates the next observation.

Let $\mathcal{X} \triangleq \mathcal{Z} \times \mathcal{A}$ be the action-observation product space and let us denote the set of possible histories by $\mathcal{X}^* = \bigcup_{k=0}^{\infty} \mathcal{X}^k$ and let $\mathscr{F}$ be a σ-algebra on $\mathcal{X}^*$. Let the context set $\mathscr{C}$ be the set of all sequences of elements in $\mathscr{F}$ and consider a function $C : \mathcal{X}^* \to \mathscr{C}$, for which we write $\mathbf{c}^t = C(x^t)$. Each context $\mathbf{c}_k^t \in \mathscr{F}$ is associated to a sequence of probability measures ϕ_k^t

$$\phi_k^t(z_{t+1} \in Z) = \mathbb{P}(z_{t+1} \in Z \mid \mathbf{c}_k^t, x^t). \tag{3}$$

We wish to perform online estimation of:

$$\mathbb{P}(z_{t+1} \mid x^t) = \sum_k \phi_k^t(z_{t+1})\, \mathbb{P}(\mathbf{c}_k^t \mid x^t). \tag{4}$$

For a given x^t, let B_k^t denote the event that the next observation will be generated by one of the contexts in $\{\mathbf{c}_1^t, \dots, \mathbf{c}_k^t\}$. Then it holds that:

$$\mathbb{P}(z_{t+1}|B_k^t, x^t) = \phi_k^t(z_{t+1})w_k^t + \mathbb{P}(z_{t+1}|B_{k-1}^t, x^t)(1-w_k^t), \tag{5}$$

where the weight $w_k^t \triangleq \mathbb{P}(\mathbf{c}_k^t \mid x^t, B_k^t)$, is a stopping probability. To perform inference, we only need to update ϕ and the weights. The former depends on the details of the model at each context. For the weights, we have the following procedure, which is is a direct outcome of Theorem 1 in (Dimitrakakis, 2010a).

$$w_k^{t+1} = \frac{\phi_k^t(z_{t+1})w_k^t}{\phi_k^t(z_{t+1})w_k^t + \mathbb{P}(z_{t+1}|x^t, B_{k-1}^t)(1-w_k^t)}. \tag{6}$$

2.1. The context structure

In general x^t is a concatenation of observation-action pairs, i.e. $(z_k, a_k) \circ (z_{k+1}, a_{k+1})$. The main question is what the context structure should be. If, for any sequence $x^t \in \mathcal{X}^*$, $\mathbf{c}^t = C(x^t)$ is such that $\mathbf{c}_{k+1}^t \subset \mathbf{c}_k$, then the random walk starts from the deepest matching context. If in addition, the contexts correspond to suffixes of $\mathcal{X}^*$ and there are Dirichlet-multinomial models at each context, then we obtain a mixture of variable order Markov decision processes (VMDP)[1]

One may alternatively consider fully observable but large spaces. Let us restrict $\mathscr{F}$ to an algebra generated by some subsets of $\mathcal{X}$. Let $X(x^t) \triangleq \{\mathbf{c} \in \mathscr{F} : x_t \in \mathbf{c}\}$, and define $C(x^t) = (\mathbf{c}_k^t : k = 1, \dots)$ such that $\mathbf{c}_k \in X(x^t)$ and ordered such that $\mathbf{c}_{k+1}^t \subset \mathbf{c}_k^t$ for all k. Now C defines a chain of contexts for each observation, where each deeper context is a smaller subset of $\mathcal{X}$.[2] Since in many reinforcement learning problems $\mathcal{A}$ is discrete, the main difficulty is how to partition the state space $\mathcal{S}$. However, once this (admittedly hard) obstacle has been overcome, perhaps with some heuristics, it is straightforward to update conditional probabilities in the same manner as for discrete, partially observable problems. We, however, are not tackling this problem explicitly in this paper.

3. Action selection

Furthermore, we need to incorporate a reward model. To do this, we shall simply add a reward distribution $\mathbb{P}(r_{t+1} \mid \mathbf{c})$ to each context $\mathbf{c}$.[3] In our model, we maintain a distribution over contexts. It follows by elementary probability, that the expected utility can be written in terms of the utility of each context:

$$\mathbb{E}(U_t \mid x^t) = \sum_{\mathbf{c}} \mathbb{E}(U_t \mid \mathbf{c}, x^t)\, \mathbb{P}(\mathbf{c} \mid x^t). \tag{7}$$

Maximising the above results in a method to select the optimal (in a decision-theoretic sense) action and is the analogue of (2). The solution, however, requires solving an augmented Markov decision process. In this paper, we shall only look at methods for approximating the values of nodes by fixing the belief parameters.

3.1. Approximate methods

Given x^t, we fix the context predictions to $\hat{\phi} = \phi^t$, so that for any $k > 0$ and $\mathbf{x} \in \mathcal{X}^*$, $\mathbb{P}(z_{t+k}, r_{t+k} \mid \mathbf{c}, \mathbf{x}) = \mathbb{P}(z_{t+k}, r_{t+k} \mid \mathbf{c}) = \hat{\phi}(z_{t+k}, r_{t+k})$, while we fix the context weights to $\hat{w} = w^t$, thus also fixing the conditional distribution over contexts, to $\mathbb{P}(\mathbf{c} \mid \mathbf{x})$. Substituting the above in (7), we obtain:

$$Q_t(\mathbf{c}) = \mathbb{E}(r_{t+1}|\mathbf{c}) \tag{8}$$
$$+ \gamma \sum_{z_{t+1}} \mathbb{P}(z_{t+1}|\mathbf{c}) \max_{a_{t+1}} \sum_{\mathbf{c}'} \mathbb{P}(\mathbf{c}'|x^t, a_{t+1}, z_{t+1}) Q_{t+1}(\mathbf{c}').$$

This immediately defines a value iteration procedure, since were are only updating the Q_t. If, for all x_t, there is some a unique $\mathbf{c}$ such that $\mathbb{P}(\mathbf{c} \mid x^t) = 1$, then this procedure becomes identical to the one proposed by McCallum (1995). Alternatively, we may use an algorithm such as Q-learning, shown in Algorithm 1.

[1] The reader is referred to (Dimitrakakis, 2010a;b) for a complete presentation.

[2] This is different from simply discretising the space and using VMDP estimation.

[3] In this section we shall omit model, context and weight subscripts when there is no ambiguity.

Algorithm 1 Weighted context Q-learning with stochastic steepest gradient descent

WCQL$(K, \mathcal{W}, \Theta, S, x^t, r_{t+1}, z_{t+1}\hat{Q}_t, \eta)$
for $\mathbf{c} \prec x^t$ **do**
  $\zeta := \mathbb{P}(\mathbf{c} \mid x^t)$, $p(\mathbf{c}'|a) := \mathbb{P}[\mathbf{c}' \mid x^t \circ (z_{t+1}, a)]$.
  $\tilde{U}_t := r_{t+1} + \gamma \max_a \sum_{\mathbf{c}'} \hat{Q}_t(\mathbf{c}')p(\mathbf{c}'|a)$
  $\hat{Q}_{t+1}(\mathbf{c}) = \hat{Q}_t(\mathbf{c}) + \eta\zeta \left(\tilde{U}_t - \hat{Q}_t(\mathbf{c})\right)$
end for

4. Experiments

4.1. Prediction

We compared the accuracy of the predictions of a VMDP (of maximum order D), a mixture of MDPs (MMDP), as well as a single k-order MDP, all estimated with closed-form Bayesian updating, on a number of tasks. Each task is an unknown POMDP μ. There were $n = 10^3$ runs performed to a horizon $T = 10^6$ for each μ. For the i-th run, we select a policy π and generate a sequence of observations $z^t(i)$ and actions $a_1^t(i)$ with distribution $\mathbb{P}_{\mu,\pi}$. For any model ν, with posterior predictive distribution $\mathbb{P}_\nu(z_{t+1}|x^t)$ at time t we calculate the average accuracy at time t: $u_t(\nu) \triangleq \frac{1}{n}\mathbb{P}_\nu(z_{t+1} = z_{t+1}^{(i)} \mid x_{1,t} = x_{1,t}^{(i)})$. Figure 1 shows the results on a stochastic maze task

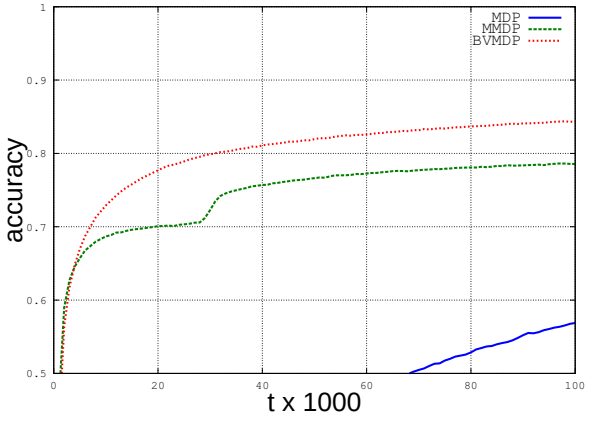

Figure 1. 8×8 maze, $Z = 16$, $D = 4$, $\epsilon = 0.1$. Predictive accuracy on mazes averaged over 10^3 runs and smoothed over 10^3 steps, showing D-order MDP model (**MDP**), mixture of MDP orders (**MMDP**), variable order Markov model (**BVMDP**).

with $Z = 16$ observations, which represent a binary encoding of the occupancy of neighbouring grid-points by a wall. In that case, we used a policy which with some probability $\epsilon > 0$, or whenever a wall was detected, took a random action, and otherwise took the same action as in the previous time step. The VMDP and MMDP were found to be superior to the MDP.

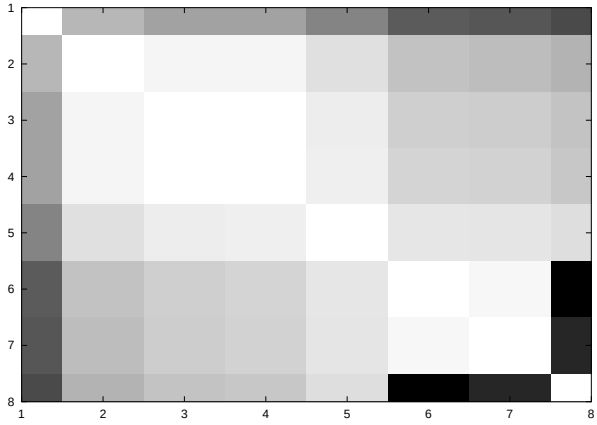

Figure 2. State similarity matrix of an 8-state $1D$-maze problem, obtained by calculating the L_1 distance of the BVMDP context distribution at each actual state. Similar states are lighter.

This was also the case in other tested environments. In general, the MMDP approach exhibits step-wise performance increases due to the fact that a distribution over model orders is maintained.

4.2. State representation

The model creates an internal representation of the current system state. To see this, consider the probability of each context conditioned on the current history, $\mathbb{P}(\mathbf{c}|x^t)$. This will be zero for non-matching contexts, and will depend on the weights w_k^t for all the matching contexts. Thus, if there are N contexts, the effective state space is $\mathbb{R}_+^N$. Figure 2 shows the L_1 distance between context distributions between each state for a corridor task with 8 states.

4.3. Planning

In this paper we do not examine decision-theoretic planning. However, Q-learning is easily implemented on top of the state representation implicitly defined by the context distribution (Alg. 1). Figure 3 shows how performance on a POMDP maze task increases with the depth of the context tree, for a maze task with $z_t = 1$ when a wall is hit and 0 otherwise, with observation noise 0.1.

5. Conclusion

We outlined how efficient, online, closed-form inference procedure can be used for estimating large or partially observable MDPs. A similar structure, proposed by Hutter (2005), used a random walk that started from the complete set and branched out to subsets.

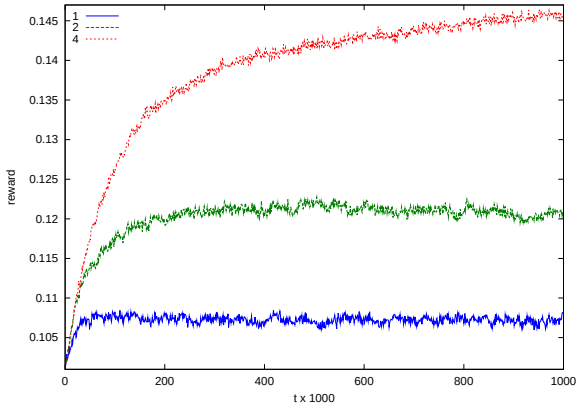

Figure 3. 4×4 maze, $Z = 2$, $\epsilon = 0.01$. VMDP reward with Q-learning, averaged over 10^3 runs, for increasing D.

This makes the approach more suitable for density estimation, in this author's view. It appears possible that branching should also be feasible for the class of context models presented here, though this is an open question. It would be interesting to combine the two approaches for *conditional* density estimation. Such an approach should remain tractable.

Nevertheless, the *crucial* problem is how to partition a space when no "natural" partitioning (such as the tree of suffixes for discrete sequences, or the binary partition for intervals) exists. This is more pronounced for *controlled* processes, because one cannot rely on the statistics of the observations to create an effective partition. For such problems, perhaps entirely new methods would have to be developed.

The simplicity of the inference makes it application of approximate decision-theoretic action selection methods (DeGroot, 1970) possible. In the point-based methods(Poupart et al., 2006), planning in an augmented-action MDP (Auer et al., 2008; Asmuth et al., 2009), sparse sampling (Wang et al., 2005), *Monte Carlo* tree search (Veness et al., 2009) or stochastic branch and bound (Dimitrakakis, 2009) methods have been suggested. It is an open question which of these is best for such planning problems.

References

Asmuth, J., Li, L., Littman, M. L., Nouri, A., and Wingate, D. A Bayesian sampling approach to exploration in reinforcement learning. In *UAI 2009*, 2009.

Auer, P., Jaksch, T., and Ortner, R.. Near-optimal regret bounds for reinforcement learning. In *Proceedings of NIPS 2008*, 2008.

DeGroot, Morris H. *Optimal Statistical Decisions*. John Wiley & Sons, 1970.

Dimitrakakis, C. Complexity of stochastic branch and bound for belief tree search in Bayesian reinforcement learning. In *2nd international conference on agents and artificial intelligence (ICAART 2010)*, pp. 259–264, Valencia, Spain, 2009. ISNTICC, Springer.

Dimitrakakis, C. Bayesian variable order Markov models. AISTATS 2010. 2010a.

Dimitrakakis, C. Variable order Markov decision processes: Exact Bayesian inference with an application to POMDPs. Technical Report. 2010b. http://fias.uni-frankfurt.de/∼dimitrakakis/papers/tr-fias-10-05.pdf.

Duff, M. *Optimal Learning Computational Procedures for Bayes-adaptive Markov Decision Processes*. PhD thesis, University of Massachusetts at Amherst, 2002.

Hutter, M. Fast non-parametric Bayesian inference on infinite trees. In *AISTATS 2005*, 2005.

McCallum, A. Instance-based utile distinctions for reinforcement learning with hidden state. In *ICML*, pp. 387–395, 1995.

Poupart, P., Vlassis, N., Hoey, J., and Regan, K. An analytic solution to discrete Bayesian reinforcement learning. In *ICML 2006*, pp. 697–704. ACM Press New York, NY, USA, 2006.

Ross, S., Chaib-draa, B., and Pineau, J. Bayes-adaptive POMDPs. In Platt, J.C., Koller, D., Singer, Y., and Roweis, S. (eds.), *Advances in Neural Information Processing Systems 20*, Cambridge, MA, 2008. MIT Press.

Veness, J., Ng, K.S., Hutter, M., and Silver, D. A Monte Carlo AIXI approximation. Arxiv preprint arXiv:0909.0801, 2009.

Wang, T., Lizotte, D., Bowling, M., and Schuurmans, D. Bayesian sparse sampling for on-line reward optimization. In *ICML '05*, pp. 956–963, New York, NY, USA, 2005. ACM. ISBN 1-59593-180-5. doi: http://doi.acm.org/10.1145/1102351.1102472.

Willems, F.M.J., Shtarkov, Y.M., and Tjalkens, T.J. The context tree weighting method: basic properties. *IEEE Transactions on Information Theory*, 41 (3):653–664, 1995.