Active Transfer Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

1 Introduction

2 Preliminaries

Markov Decision Processes We define a Markov decision process (MDP) as a tuple $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, p_0, \gamma \rangle$, where $\mathcal{S}$ is the state-space, $\mathcal{A}$ is a finite set of actions, $\mathcal{P}(\cdot|s,a)$ is the distribution of the next state s' given that action a is taken in state s, $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function, p_0 is the initial-state distribution, and $\gamma \in [0,1)$ is the discount factor. We assume the reward function to be uniformly bounded by a constant $R_{max} > 0$. A deterministic policy $\pi : \mathcal{S} \to \mathcal{A}$ is a mapping from states to actions. At the beginning of each episode of interaction, the initial state s_0 is drawn from p_0. Then, the agent takes the action $a_0 = \pi(s_0)$, receives a reward $\mathcal{R}(s_0, a_0)$, transitions to the next state $s_1 \sim \mathcal{P}(\cdot|s_0, a_0)$, and the process is repeated. The goal is to find the policy maximizing the long-term return over a possibly infinite horizon: $\max_\pi J(\pi) = \mathbb{E}[\sum_{t=0}^{\infty} \gamma^t r_t \mid \mathcal{M}, \pi]$. To this end, we define the optimal value function $Q^*(s,a)$ as the expected return obtained by taking action a in state s and following an optimal policy thereafter. Then, an optimal policy π^* is a policy that is greedy with respect to the optimal value function, i.e., $\pi^*(s) = \operatorname{argmax}_a Q^*(s,a)$ for all states s. It can be shown (e.g., [4]) that Q^* is the unique fixed-point of the optimal Bellman operator T defined by $TQ(s,a) = \mathcal{R}(s,a) + \gamma \mathbb{E}_{\mathcal{P}}[\max_{a'} Q(s',a')]$ for any value function Q. From now on, we adopt the term Q-function to denote any plausible value function, i.e., any function $Q : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ uniformly bounded by $\frac{R_{max}}{1-\gamma}$.

We define the Bellman residual of a Q-function Q as $B(Q) \triangleq TQ - Q$. Notice that a Q-function Q is optimal if, and only if, $B(Q)(s,a) = 0$ for all s, a.

Multitask Settings We represent tasks τ as MDPs with shared state and action spaces, but with potentially different values for all other parameters. We assume the existence of a distribution $\mathcal{D}$ over tasks, i.e., $\tau \sim \mathcal{D}$, and we suppose that we are able to sample from such distribution.

3 Approach

We start by noticing that the task distribution $\mathcal{D}$ clearly induces a distribution over optimal Q-functions. Then, our goal is to estimate such distribution from the set of source tasks and use it as a prior for speeding up the learning process in the target task.

Consider solving the target task given a dataset D of N samples. The posterior distribution over optimal Q-functions is:

$$P(Q \mid D) \propto P(D \mid Q) P(Q), \tag{1}$$

Submitted to 31st Conference on Neural Information Processing Systems (NIPS 2017). Do not distribute.

where $P(D \mid Q)$ is the likelihood of observing the dataset D given that Q is optimal and $P(Q)$ is the prior distribution induced by $\mathcal{D}$. Computing the likelihood requires knowing the transition and reward models of the current task (i.e., the MDP model), which are not available in practice. However, since we are conditioned on the fact that Q is optimal, we can derive a simple approximation. We consider the empirical Bellman error $||B(Q)||^p_{p,D}$ under the l_p-norm defined as:

$$||B(Q)||^p_{p,D} = \frac{1}{N}\sum_{i=1}^{N}\left|r_i + \gamma \max_{a'} Q(s'_i, a') - Q(s_i, a_i)\right|^p \tag{2}$$

Assuming states and actions to be drawn from a fixed joint distribution μ, we have $||B(Q)||^p_{p,\mu} = 0$ whenever Q is optimal. Then, the probability of observing a dataset D clearly decreases exponentially as $||B(Q)||^p_{p,D}$, for an optimal Q, increases. This can be seen, for instance, by applying Hoeffding's inequality. Thus, a natural way to model the likelihood $P(D \mid Q)$ is:

$$P(D \mid Q) \propto e^{-\lambda N ||B(Q)||^p_{p,D}} \tag{3}$$

where λ is a constant hyperparameter. Intuitively, this indicates that D is more likely when it induces low empirical Bellman error under an optimal Q-function. Furthermore, as the number of available samples N increases, the distribution becomes more peaked at zero since the empirical Bellman error converges to the true Bellman error. In the limit $N \to \infty$:

$$P(D \mid Q) \propto I_{\left\{||B(Q)||^p_{p,D}=0\right\}} \tag{4}$$

3.1 Regularized Bellman Residual Minimization with Gaussian Priors

Taking the maximum of the log-posterior, we obtain the following optimization problem:

$$\min_{Q \in \mathcal{Q}} ||B(Q)||^p_{p,D} - \log P(Q) \tag{5}$$

Let us specify a particular hypothesis space $\mathcal{Q}$. We consider Q-functions $Q_{\boldsymbol{w}}$ parameterized by the vector $\boldsymbol{w}$. Our optimization problem becomes:

$$\min_{\boldsymbol{w}} ||B(\boldsymbol{w})||^p_{p,D} - \log P(\boldsymbol{w}) \tag{6}$$

We model the prior distribution over the optimal parameters $\boldsymbol{w}$ as a Gaussian. That is, we assume:

$$P(\boldsymbol{w}) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \tag{7}$$

Then, our optimization, adopting the l_2-norm, becomes:

$$\min_{\boldsymbol{w}} ||B(\boldsymbol{w})||^2_D + ||\boldsymbol{w} - \boldsymbol{\mu}||_{\boldsymbol{\Sigma}} = \min_{\boldsymbol{w}} \frac{1}{N}\sum_{i=1}^{N}|y_i - Q_{\boldsymbol{w}}(s_i, a_i)|^2 + ||\boldsymbol{w} - \boldsymbol{\mu}||_{\boldsymbol{\Sigma}} \tag{8}$$

where $y_i = r_i + \gamma \max_{a'} Q_{\boldsymbol{w}}(s'_i, a')$ and $||\boldsymbol{x}||_{\boldsymbol{A}} \triangleq \boldsymbol{x}^T \boldsymbol{A}^{-1} \boldsymbol{x}$ for $\boldsymbol{A}$ positive definite matrix.

Linear model We assume a linear model for the Q-functions: $Q_{\boldsymbol{w}}(s, a) = \boldsymbol{w}^T \boldsymbol{\phi}(s, a)$. Here $\boldsymbol{\phi}$ is a K-dimensional feature vector. Then, the solution to the optimization problem of Eq. (8) can be computed in closed form as follows:

$$\boldsymbol{w}^* = \left(\boldsymbol{A}^T \boldsymbol{A} + \boldsymbol{\Sigma}^{-1}\right)^{-1} \left(\boldsymbol{A}^T \boldsymbol{b} + \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}\right) \tag{9}$$

where $\boldsymbol{A}$ is an $N \times K$ matrix containing the feature vectors at each data point (s_i, a_i) and $\boldsymbol{b}$ is an N-dimensional vector containing their targets y_i. The resulting algorithm is a regularized version of LSVI.

Neural network We model $Q_{\boldsymbol{w}}(s, a)$ as a neural network with parameters $\boldsymbol{w}$. Then, the gradient of the objective function of Eq. (8) can be easily computed by standard backpropagation. The resulting algorithm is a regularized version of NFQI.

3.2 Variational Inference for Efficient Exploration

A major drawback of the maximum-a-posteriori (MAP) approach is that it does not explicitly estimate the posterior distribution $P(Q \mid D)$. Thus, given a dataset D, the algorithm consistently chooses the same MAP Q-function, and the only way to allow learning is to introduce a simple form of exploration (e.g., ϵ-greedy or softmax), whose drawbacks are well-known. The advantages of a more time-coherent exploration strategy such as posterior sampling (or Thompson sampling) have already been proved by many existing works (CITE). In particular, direct posterior sampling of Q-functions (also known as value function randomization), when a distribution is available, have been proven to be very effective and scalable. Thus, we known extend our previous approach to explicitly estimate $P(Q \mid D)$, so as to allow efficient exploration of the target task. The main complication is that, as mentioned earlier, we cannot compute the likelihood $P(D \mid Q)$ explicitly, but we only have access to a model that is proportional to it. Then, estimating the posterior distribution is clearly intractable since it requires computing the marginal likelihood $P(D) = \int P(D \mid Q) dQ$. Thus, we resort variational inference (CITE) to approximate the intractable posterior with a simpler distribution from which we are able to sample.

The main idea behind variational inference is to approximate the posterior $P(Q|D)$ with a simpler distribution $q_{\boldsymbol{\phi}}(Q)$, parameterized by the vector $\boldsymbol{\phi}$, from which we can easily get samples. The best approximation is chosen in terms of the Kullback-Leibler (KL) divergence between the two distributions, that is:

$$\min_{\boldsymbol{\phi}} KL\left(q_{\boldsymbol{\phi}}(Q) \,||\, P(Q \mid D)\right) \tag{10}$$

It is well-known that minimizing the KL divergence is equivalent to maximizing the so-called *evidence lower bound* (ELBO), which is defined as:

$$\text{ELBO}(\boldsymbol{\phi}) = \mathbb{E}_{Q \sim q_{\boldsymbol{\phi}}}\left[\log P(D|Q)\right] - KL\left(q_{\boldsymbol{\phi}}(Q) \,||\, P(Q)\right) \tag{11}$$

As before, we assume a parametric form $Q_{\boldsymbol{w}}$ for representing Q-functions. Thus, we reduce distributions over functions to distributions over vectors $\boldsymbol{w} \in \mathbb{R}^K$. Given a specific form for the approximate posterior $q_{\boldsymbol{\phi}(\boldsymbol{w})}$ and the prior $P(\boldsymbol{w})$, our inference problem reduces to:

$$\min_{\boldsymbol{\phi}} \mathbb{E}_{\boldsymbol{w} \sim q_{\boldsymbol{\phi}}}\left[\lambda N ||B(\boldsymbol{w})||_D^2\right] + KL\left(q_{\boldsymbol{\phi}}(\boldsymbol{w}) \,||\, P(\boldsymbol{w})\right) \tag{12}$$

Intuitively, the approximate posterior trades-off between placing probability mass over those weights $\boldsymbol{w}$ that have low empirical Bellman error (first term), and staying close to the prior distribution that is learned from the set of source tasks (second term). The factor N multiplying $||B(\boldsymbol{w})||_D^2$ makes sure that, as the number of samples goes to infinity, only the first term is considered in the optimization. In such case, all probability mass go over the (approximate) fixed-point of the optimal Bellman operator. On the other hand, for small values of N, there is less evidence that the empirical Bellman error closely estimate the true one, thus giving more importance to staying close to the prior.

Depending on the choice of the prior and approximate posterior distributions, the optimization problem of Eq. (12) might still be very hard to solve. We now propose some suitable choices that make the problem efficiently solvable.

Gaussian distributions We model both the prior and the approximate posterior as Gaussian distributions: $P(\boldsymbol{w}) = \mathcal{N}(\bar{\boldsymbol{\mu}}, \bar{\boldsymbol{\Sigma}})$ and $q_{\boldsymbol{\phi}}(\boldsymbol{w}) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, respectively. For the sake of simplicity, we assume $\boldsymbol{\Sigma} = \text{diag}(\sigma_1^2, \ldots, \sigma_K^2)$ and $\bar{\boldsymbol{\Sigma}} = \text{diag}(\bar{\sigma}_1^2, \ldots, \bar{\sigma}_K^2)$. Notice, in this case, that the parameter vector $\boldsymbol{\phi}$ is $[\mu_1, \ldots, \mu_K, \sigma_1^2, \ldots, \sigma_K^2]$. The KL divergence between the two distributions can be computed in closed form as:

$$KL\left(q_{\boldsymbol{\phi}}(\boldsymbol{w}) \,||\, P(\boldsymbol{w})\right) = \frac{1}{2}\left(\log \frac{|\bar{\boldsymbol{\Sigma}}|}{|\boldsymbol{\Sigma}|} + \text{Tr}\left(\bar{\boldsymbol{\Sigma}}^{-1}\boldsymbol{\Sigma}\right) + (\boldsymbol{\mu} - \bar{\boldsymbol{\mu}})^T \bar{\boldsymbol{\Sigma}}^{-1}(\boldsymbol{\mu} - \bar{\boldsymbol{\mu}}) - K\right) \tag{13}$$

To solve the optimization problem of Eq. (12) we adopt stochastic backpropagation (CITE). Thus, we need to compute the gradient of the objective function $\mathcal{L}(\boldsymbol{\phi})$ of Eq. (12) with respect to each component of $\boldsymbol{\phi}$. This can be easily done for the second term:

$$\nabla_{\boldsymbol{\mu}} KL\left(q_{\boldsymbol{\phi}}(\boldsymbol{w}) \,||\, P(\boldsymbol{w})\right) = \bar{\boldsymbol{\Sigma}}^{-1}(\boldsymbol{\mu} - \bar{\boldsymbol{\mu}}) \tag{14}$$

$$\nabla_{\sigma_k^2} KL\left(q_{\boldsymbol{\phi}}(\boldsymbol{w}) \,||\, P(\boldsymbol{w})\right) = \frac{1}{2}\left(\frac{1}{\bar{\sigma}_k^2} - \frac{1}{\sigma_k^2}\right) \tag{15}$$

For the first term, we have to compute the gradient of an expectation with respect to the parameters of the Gaussian distribution under which the expectation is taken. For the mean, we can resort to Thm. X of (CITE Bonnet 1964):

$$\nabla_{\mu_k} \mathbb{E}_{\boldsymbol{w}\sim\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})} \left[\lambda N ||B(\boldsymbol{w})||_D^2\right] = \mathbb{E}_{\boldsymbol{w}\sim\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})} \left[\lambda N \nabla_{w_k} ||B(\boldsymbol{w})||_D^2\right] \tag{16}$$

For the covariance, we can resort to Thm. Y of (CITE Price 1958):

$$\nabla_{\sigma_k^2} \mathbb{E}_{\boldsymbol{w}\sim\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})} \left[\lambda N ||B(\boldsymbol{w})||_D^2\right] = \frac{1}{2} \mathbb{E}_{\boldsymbol{w}\sim\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})} \left[\lambda N \nabla^2_{w_k,w_k} ||B(\boldsymbol{w})||_D^2\right] \tag{17}$$

We refer the reader to (CITE Deepmind) for a simple proofs of these two theorems.

It only remains to compute the gradient and Hessian of the empirical Bellman error $||B(\boldsymbol{w})||_D^2$ with respect to $\boldsymbol{w}$. Recall that:

$$||B(\boldsymbol{w})||_D^2 = \frac{1}{N} \sum_{i=1}^{N} \Big(r_i + \gamma \max_{a'} Q_{\boldsymbol{w}}(s_i', a') - Q_{\boldsymbol{w}}(s_i, a_i)\Big)^2 \tag{18}$$

A major complication is that computing the gradient with respect to $\boldsymbol{w}$ requires differentiating through the maximum operator. In iterative approaches such as Fitted Q-Iteration (FQI) or Deep Q-Networks (DQNs), this is typically solved by computing the Bellman target using the previous weights and adapting the current weights so that the resulting Q-functions matches the target. In such case, no differentiation through the maximum operator is required. However, in our case there is no previous weight: we simply want to now what is the Bellman error of any given weight $\boldsymbol{w}$ and we are thus required to differentiate through the maximum operator when updating the approximate posterior, which is not possible in practice. To solve this issue, we replace the max with the *mellowmax* operator defined by:

$$\operatorname*{mm}_a Q_{\boldsymbol{w}}(s, a) = \frac{1}{\kappa} \log \frac{1}{|\mathcal{A}|} \sum_a e^{\kappa Q_{\boldsymbol{w}}(s,a)} \tag{19}$$

where κ is a hyperparameter and $|\mathcal{A}|$ is the number of actions. The mellowmax operator has several appealing properties that make it suitable for our settings: (i) it converges to the maximum as $\kappa \to \infty$, (ii) it has a unique fixed point, and (iii) it is *differentiable*. Replacing max with the mellowmax operator, we obtain a new approximation to the Bellman residual:

$$\widetilde{B}(\boldsymbol{w})(s, a) \triangleq \mathcal{R}(s, a) + \gamma \mathbb{E}_{\mathcal{P}}[\operatorname*{mm}_{a'} Q_{\boldsymbol{w}}(s', a')] - Q_{\boldsymbol{w}}(s, a) \tag{20}$$

and to the empirical Bellman error:

$$||\widetilde{B}(\boldsymbol{w})||_D^2 = \frac{1}{N} \sum_{i=1}^{N} \Big(r_i + \gamma \operatorname*{mm}_{a'} Q_{\boldsymbol{w}}(s_i', a') - Q_{\boldsymbol{w}}(s_i, a_i)\Big)^2 \tag{21}$$

Of fundamental importance, this last quantity is now differentiable with respect to $\boldsymbol{w}$.

Differentiating trough the mellowmax operator with respect to $\boldsymbol{w}$ is now very simple:

$$\begin{aligned}
\nabla_{\boldsymbol{w}_k} \operatorname*{mm}_a Q_{\boldsymbol{w}}(s, a) &= \nabla_{\boldsymbol{w}_k} \frac{1}{\kappa} \log \frac{1}{|\mathcal{A}|} \sum_a e^{\kappa Q_{\boldsymbol{w}}(s,a)} && (22)\\
&= \frac{1}{\kappa} \frac{1}{\frac{1}{|\mathcal{A}|}\sum_a e^{\kappa Q_{\boldsymbol{w}}(s,a)}} \frac{1}{|\mathcal{A}|} \sum_a \nabla_{\boldsymbol{w}_k} e^{\kappa Q_{\boldsymbol{w}}(s,a)} && (23)\\
&= \frac{1}{\kappa} \frac{1}{\frac{1}{|\mathcal{A}|}\sum_a e^{\kappa Q_{\boldsymbol{w}}(s,a)}} \frac{1}{|\mathcal{A}|} \sum_a e^{\kappa Q_{\boldsymbol{w}}(s,a)} \kappa \nabla_{\boldsymbol{w}_k} Q_{\boldsymbol{w}}(s, a) && (24)\\
&= \frac{\sum_a e^{\kappa Q_{\boldsymbol{w}}(s,a)} \nabla_{\boldsymbol{w}_k} Q_{\boldsymbol{w}}(s, a)}{\sum_a e^{\kappa Q_{\boldsymbol{w}}(s,a)}} && (25)
\end{aligned}$$

The diagonal of the Hessian can also be computed straightforwardly:

$$\nabla^2_{\boldsymbol{w}_k,\boldsymbol{w}_k} \operatorname*{mm}_a Q_{\boldsymbol{w}}(s,a) = \nabla_{\boldsymbol{w}_k} \frac{\sum_a e^{\kappa Q_{\boldsymbol{w}}(s,a)} \nabla_{\boldsymbol{w}_k} Q_{\boldsymbol{w}}(s,a)}{\sum_a e^{\kappa Q_{\boldsymbol{w}}(s,a)}} \tag{26}$$

$$= \frac{\nabla_{\boldsymbol{w}_k} \sum_a e^{\kappa Q_{\boldsymbol{w}}(s,a)} \nabla_{\boldsymbol{w}_k} Q_{\boldsymbol{w}}(s,a)}{\sum_a e^{\kappa Q_{\boldsymbol{w}}(s,a)}} + \nabla_{\boldsymbol{w}_k} \left(\frac{1}{\sum_a e^{\kappa Q_{\boldsymbol{w}}(s,a)}} \right) \sum_a e^{\kappa Q_{\boldsymbol{w}}(s,a)} \nabla_{\boldsymbol{w}_k} Q_{\boldsymbol{w}}(s,a) \tag{27}$$

$$= (\kappa+1) \frac{\sum_a e^{\kappa Q_{\boldsymbol{w}}(s,a)} \nabla^2_{\boldsymbol{w}_k,\boldsymbol{w}_k} Q_{\boldsymbol{w}}(s,a)}{\sum_a e^{\kappa Q_{\boldsymbol{w}}(s,a)}} - \kappa \frac{\left(\sum_a e^{\kappa Q_{\boldsymbol{w}}(s,a)} \nabla_{\boldsymbol{w}_k} Q_{\boldsymbol{w}}(s,a)\right)^2}{\left(\sum_a e^{\kappa Q_{\boldsymbol{w}}(s,a)}\right)^2} \tag{28}$$

4 Related Works

List of any paper that relates to our approach together with a brief description:

- [3]: the authors propose a method for efficient exploration via randomized value functions. The optimal Q-function is computed by bayesian LSVI and, after each update, parameters are sampled from the posterior. Then, the agent follows a greedy policy with respect to the sampled Q-function. Regret bounds are provided.
- [2]: the authors extend the idea of randomized value functions to drive exploration in deep RL. A posterior distribution over Q-functions is approximated via bootstrapping. In each episode, the agent acts greedily with respect to a Q-function sampled from the approximated posterior.
- [1]: the authors build on top of bootstrapped DQNs to provide UCB-like exploration bonuses. In a previous (?) version of the paper, exploration bonuses based on information gain are also proposed.

5 Experiments

6 Conclusion

References

[1] Richard Y Chen, John Schulman, Pieter Abbeel, and Szymon Sidor. Ucb and infogain exploration via q-ensembles. *arXiv preprint arXiv:1706.01502*, 2017.

[2] Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep exploration via bootstrapped dqn. In *Advances in neural information processing systems*, pages 4026–4034, 2016.

[3] Ian Osband, Benjamin Van Roy, and Zheng Wen. Generalization and exploration via randomized value functions. *arXiv preprint arXiv:1402.0635*, 2014.

[4] Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., New York, NY, USA, 1994.

A Proofs