BERMIN: A Model Selection Algorithm for Reinforcement Learning Problems

Amir-massoud Farahmand[*]
School of Computer Science
McGill University
Montreal, Quebec, Canada

Csaba Szepesvári
Department of Computing Science
University of Alberta
Edmonton, Alberta, Canada

1 Introduction

Most reinforcement learning algorithms rely on the use of some function approximation method. In general, their performance will be largely influenced by what function approximation method is used and how it is configured. Current practice is that the user of the algorithm decides about both the method and its configuration, such as the number and nature of basis functions for a linear function approximation method, and tune them in an ad hoc manner. Although good rules of thumb may exist of how to tune a particular method, or which method to use in a particular situation, there is no guarantee that a rule of thumb will give good results on the problem that the user wants to solve. A superior approach, however, is to choose and configure the method based on the data. We show that strong theoretical results can be proven if an appropriate selection procedure is used.[1]

In this paper we assume that we are given a representative dataset $\mathcal{D}_n$ of sampled transitions from a Markovian Decision Process (MDP), the goal being to find a good policy of the MDP [2]. Instead of directly working with policies, we consider the problem of finding an action-value function with a small (integrated, squared) *Bellman error*, which is supposed to facilitate the search for a good policy: When the Bellman error of an action-value function is zero (or very small) an optimal (respectively, good) policy can be obtained from the action-value function with minimal effort.

Suppose that we are given a list of action-value functions $Q_1, Q_2, \dots, Q_P$ (with the possibility of $P > n$, or even $P = \infty$) and a dataset[2]

$$\mathcal{D}_n = \{(X_1, A_1, R_1, X_1'), \dots, (X_n, A_n, R_n, X_n')\}.$$

Here $(R_i, X_i') \sim P(\cdot, \cdot|X_i, A_i)$, $A_i \sim \pi_b(\cdot|X_i)$, and $X_i \sim \nu_{\mathcal{X}}$ $(i = 1, \dots, n)$, where $\nu_{\mathcal{X}}$ is some fixed distribution over the states and π_b is some stochastic, stationary Markov policy, the so-called behavior policy. We shall denote by ν the common distribution underlying (X_i, A_i). We say that this dataset meets the *standard offline sampling assumption*.

Our goal is to devise a procedure that selects the action-value function amongst $\{Q_1, \dots, Q_P\}$ that has the smallest (integrated, squared) Bellman (optimality) error. Thus, the ideal procedure would return $Q_{\hat{k}}$, with $\hat{k} = \text{argmin}_{1 \le k \le P} \|Q_k - T^* Q_k\|_\nu^2$.

It is tempting to use a conventional supervised learning model selection approach, such as *complexity regularization* [3, 4], to devise a model selection procedure for reinforcement learning problems. A straightforward adoption of complexity regularization to our problem suggests the following procedure: First, assume that data $\mathcal{D}_n$ is independent of the candidates $Q_1, Q_2, \dots$. Further, assume that data-based estimates $\text{BE}_n(Q_k)$ of the respective Bellman errors (i.e., $\|Q_k - T^* Q_k\|_\nu$) of the candidates are available. Then choose

$$\hat{k} = \operatorname*{argmin}_{k \ge 1} \left[C_1 \, \text{BE}_n(Q_k) + C_2 \, \frac{\text{pen}(k)}{n} \right],$$

[*] Also affiliated with the Department of Computing Science, University of Alberta, Canada.

[1] This extended abstract is a brief summary of the accepted paper by Farahmand and Szepesvári [1].

[2] For the definition of standard symbols used in the RL literature, refer to the full version of the paper [1].

where $C_1 \geq 1$ and $C_2 > 0$ are appropriate constants and $\text{pen}(k)$ is a suitable complexity penalty, such as $\text{pen}(k) = \ln(k)$.

Unfortunately, the Bellman error is not easy to work with. This is because the Bellman optimality operator T^* is not available in the learning setting. Moreover, even though $R(X_i, A_i) + \gamma \max_{a' \in \mathcal{A}} Q(X_i', a')$ is an unbiased estimate of $T^*Q(X_i, A_i)$ (for $1 \leq i \leq n$), this operator cannot be used in a simple manner to estimate the Bellman error. This would be clear if one notice that for any fixed bounded measurable function Q,

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\left|Q(X_i, A_i) - [R_i + \gamma \max_{a' \in \mathcal{A}} Q(X_i', a')]\right|^2\right] =$$

$$\|Q - T^*Q\|_\nu^2 + \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\left|T^*Q(X_i, A_i) - [R_i + \gamma \max_{a' \in \mathcal{A}} Q(X_i', a')]\right|^2\right] \neq \|Q - T^*Q\|_\nu^2 ,$$

which shows that $\frac{1}{n}\sum_{i=1}^{n} |Q(X_i, A_i) - [R_i + \gamma \max_{a' \in \mathcal{A}} Q(X_i', a')]|^2$ is a biased estimate of the Bellman error. The bias is a function of Q itself, which makes it quite different from the variance of the noise in the standard regression setting, which is independent of the regression function.

The main contribution of this work is a procedure called BERMIN that has an oracle-like property in the sense that it selects the model with the minimum Bellman error up to a multiplicative constant and some additional terms that converge to zero.

2 Model Selection Algorithm for Bellman Error Minimization (BERMIN)

The basic idea behind our approach is that while the Bellman operator T^* itself is not accessible, one still may approximately learn the function T^*Q and use it to estimate the Bellman error. Since for any fixed, bounded, measurable, deterministic function Q and index $1 \leq i \leq n$, it holds that $\mathbb{E}\left[R_i + \gamma \max_{a' \in \mathcal{A}} Q(X_i', a') \,\middle|\, X_i, A_i\right] = T^*Q(X_i, A_i)$, the regression function underlying

$$\mathcal{D}_{n,k} = \left\{\left((X_1, A_1), R_1 + \gamma \max_{a' \in \mathcal{A}} Q_k(X_1', a')\right), \ldots, \left((X_n, A_n), R_n + \gamma \max_{a' \in \mathcal{A}} Q_k(X_n', a')\right)\right\}$$

is T^*Q_k Thus, we can feed $\mathcal{D}_{n,k}$ to a regression procedure which, ideally, returns a "good" approximation to T^*Q_k. As the regression method one can use any of the large number of state-of-the-art techniques (cf., Hastie et al. [5]).

Let the action-value function returned by the chosen regression algorithm be denoted by $\tilde{Q}_k$. If $\tilde{Q}_k$ is close to T^*Q_k, then by calculating $\|Q_k - \tilde{Q}_k\|_n^2 \triangleq \frac{1}{n}\sum_{i=1}^{n} |Q_k(X_i, A_i) - \tilde{Q}_k(X_i, A_i)|^2 \approx \|Q_k - \tilde{Q}_k\|_\nu^2 \approx \|Q_k - T^*Q_k\|_\nu^2$, one can select the action-value function with the smallest Bellman error based on computing $\text{argmin}_{1 \leq k \leq P} \|Q_k - \tilde{Q}_k\|_n^2$. The problem with this procedure is that it might be overly optimistic and thus it may result in an uncontrolled error. To see why, imagine that for some index k_0 whose associated Bellman error $\|Q_{k_0} - T^*Q_{k_0}\|_\nu^2$ is "large", the regression procedure returns an estimate such that $\|Q_{k_0} - \tilde{Q}_{k_0}\|_\nu^2 \ll \|Q_{k_0} - T^*Q_{k_0}\|_\nu^2$ (for example, because the regression procedure might be biased towards action-values close to zero, Q_{k_0} might be close to zero, while $T^*Q_{k_0}$ might be far from zero, cf. also Figure 1). As a result, the above procedure will likely select k_0, and thus might miss some other index with a lower Bellman error.

To avoid this problem, we must guard the procedure against the underestimation of the Bellman error. BERMIN achieves this by correcting $\|Q_k - \tilde{Q}_k\|_\nu^2$ with $\|T^*Q_k - \tilde{Q}_k\|_\nu^2$. Since $\|Q_k - T^*Q_k\|_\nu^2 \leq 2[\|Q_k - \tilde{Q}_k\|_\nu^2 + \|T^*Q_k - \tilde{Q}_k\|_\nu^2]$, the correction indeed prevents the choice of an overly optimistic estimate. The first term of the right-hand side can be estimated by $\|Q_k - \tilde{Q}_k\|_n^2$. We further assume that we are provided with a (tight) high-probability upper bound, $\bar{b}_k$, on $\|T^*Q_k - \tilde{Q}_k\|_\nu^2$, i.e., $\|T^*Q_k - \tilde{Q}_k\|_\nu^2 \leq \bar{b}_k$ with high probability. We propose to select the action-value function corresponding to the minimum of $\|Q_k - \tilde{Q}_k\|_n^2 + \bar{b}_k$. If $\bar{b}_k$ is a sufficiently tight bound, we expect that using $\bar{b}_k$ in place of $\|T^*Q_k - \tilde{Q}_k\|_\nu^2$ will not introduce any significant further bias.

We want to take care of one more detail. We would like our procedure to handle situations where the number of candidate action-value functions, P, is very large, or even potentially infinite. The latter

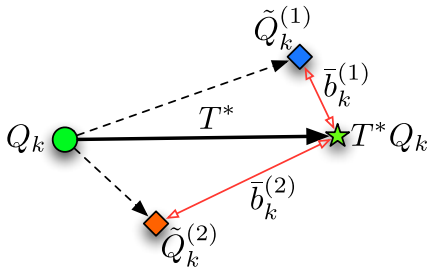

Figure 1: Consider the problem of estimating the Bellman error $\|Q_k - T^*Q_k\|_\nu^2$. If T^*Q_k is replaced by a surrogate $\tilde{Q}_k^{(1)}$, $\|Q_k - \tilde{Q}_k^{(1)}\|_\nu^2$ gives a relatively good estimate of this quantity because $\tilde{Q}_k^{(1)}$ is close to T^*Q_k. However, when $\tilde{Q}_k^{(2)}$ replaces T^*Q_k, the resulting estimate of the Bellman error becomes poor and $\|Q_k - \tilde{Q}_k^{(2)}\|_\nu^2$ would be an *underestimate* of the true Bellman error. This might lead to the unjust selection of the candidate Q_k. One way to protect oneself against such mistakes is to take into account how well the surrogate $\tilde{Q}_k$ approximates T^*Q.

Algorithm 1 BERMIN$(\{Q_k\}_{k=1,2,\dots}, \mathcal{D}_{(m,n)}, \text{REGRESS}(\cdot), \delta, a, B, \tau)$

Split $\mathcal{D}_{(m,n)}$ into two disjoint parts: $\mathcal{D}_{(m,n)} = \mathcal{D}'_m \cup \mathcal{D}''_n$.
Choose (C_k) such that $S = \sum_{k\geq 1} \exp(-\frac{(1-a)^2 a\, n}{16B^2\tau(1+a)} C_k) < \infty$.
Choose (δ'_k) such that $\sum_{k\geq 1} \delta'_k = \delta/2$.
for $k = 1, 2, \dots$ **do**
    $(\tilde{Q}_k, \bar{b}_k) \leftarrow \text{REGRESS}(\mathcal{D}'_{m,k}, \delta'_k)$
    $e_k \leftarrow \frac{1}{|\mathcal{D}''_n|} \sum_{(X,A)\in\mathcal{D}''_n} (Q_k(X,A) - \tilde{Q}_k(X,A))^2$
    $\mathcal{R}_k^{\text{RL}} \leftarrow \frac{1}{(1-a)^2}\, e_k + \bar{b}_k$
end for
$\hat{k} \leftarrow \text{argmin}_{k\geq 1} \left[\mathcal{R}_k^{\text{RL}} + C_k\right]$
return $\hat{k}$

situation arises when one transforms the algorithm into an anytime method, whose computation budget may or may not be limited, which keeps generating candidates if given more time. As a consequence of this, we add another penalty term that prevents optimistic selection bias and we will let $P = \infty$. If P is finite and small compared to n, this penalty term can safely be ignored.

BERMIN, shown as Algorithm 1, implements the described ideas. The algorithm's inputs are the candidate action-value functions, the dataset $\mathcal{D}_{(m,n)}$, a regression procedure REGRESS, a desired error probability δ, and three constants: $0 < a < 1$, B, and τ. Here a is a tuning parameter, the constant B is the bound on all functions involved (that is Q_k, $\tilde{Q}_k$, T^*Q_k, and $\bar{b}_k$), and τ is the forgetting time of the Markov chain (cf. [1]). The effect of these values on the quality of the solution is quantified in Theorem 1.

2.1 Theoretical Guarantee and Conclusions

In this section we state our main result, which shows that BERMIN has an oracle-like behavior. We prove the result under the following assumptions:

Assumption A1 Assume that the following hold:

1. The standard offline sampling assumption is satisfied by the data set $\mathcal{D}''_n = \{(X_1, A_1, R_1, X'_1), \dots, (X_n, A_n, R_n, X'_n)\}$. The time-homogeneous Markov chain $X_1, X_2, \dots, X_n$ uniformly quickly forgets its past with a forgetting time τ.

2. The functions Q_k, $\tilde{Q}_k$, T^*Q_k $(k \geq 1)$ are bounded by a deterministic quantity $B > 0$.

3. The functions Q_k $(k \geq 1)$ are deterministic.

4. For each k and for any $0 < \delta'_k < 1$, $(\tilde{Q}_k, \bar{b}_k) = \text{REGRESS}(\mathcal{D}'_{m,k}, \delta'_k)$ are $\sigma(\mathcal{D}'_m)$-measurable, $\bar{b}_k \in [0, 4B^2]$ and $\|\tilde{Q}_k - T^*Q_k\|_\nu^2 \le \bar{b}_k$ holds with probability at least $1 - \delta'_k$.

5. For $(X_i, A_i, R_i, X'_i) \in \mathcal{D}''_n$, the distribution of (X_i, A_i) given $\mathcal{D}'_m$ is ν: $\mathbb{P}\{(X_i, A_i) \in U | \mathcal{D}'_m\} = \nu(U)$ for any measurable set $U \subset \mathcal{X} \times \mathcal{A}$.

The following theorem is the main result of this work.

Theorem 1 (Model Selection for RL/Planning)**.** *Let Assumption A1 hold. Consider the* BERMIN *algorithm used with some $0 < a < 1$, $0 < \delta \le 1$, and $(C_k)_{k\ge1}$ such that $S \triangleq \sum_{k\ge1} \exp\left(-\frac{(1-a)^2 a\, n}{16B^2\tau\,(1+a)} C_k\right) < \infty$ holds. Let $\hat{k}$ be the index selected by* BERMIN*. Then, with probability at least $1-\delta$,*

$$\|Q_{\hat{k}} - T^*Q_{\hat{k}}\|_\nu^2 \le 4(1+a)\min_{k\ge1}\left\{\frac{2}{(1-a)^2}\|Q_k - T^*Q_k\|_\nu^2 + \frac{3}{(1-a)^2}\bar{b}_k + 2C_k\right\} + \frac{96B^2\tau\,(1+a)}{(1-a)^2 a\, n}\ln\left(\frac{4S}{\delta}\right).$$

Note that $C_k = \frac{32B^2\tau(1+a)}{(1-a)^2 an}\ln(k)$ satisfies $S < \infty$ (in particular, with this choice we get $S = \pi^2/6$). Detailed discussion of the assumptions and the theorem is presented in the full version of this paper [1].

Our main theoretical result, Theorem 1, is a finite-sample high-probability upper bound that shows that the Bellman error of the action-value function selected by BERMIN is almost as small as that of an oracle who has access to the true Bellman errors. This result can further be elaborated to show that BERMIN can be made adaptive in the sense that it can compete with an oracle who selects the model with the smallest error bounds [1]. As far as we know, this is the first work that considers adaptivity in a reinforcement learning scenario. The main message of our results is that just like in supervised learning, it is possible to learn almost as fast as if one had extra *a priori* information.

Although in this paper we made some progress toward reinforcement learning algorithms that require minimum human supervision, the problem is far from being solved. In particular, the following questions require further investigation:

- How to generate the list of candidate action-value functions $Q_1, Q_2, \ldots$?
- What is the relation between the quality of the solution of the fixed point of the Bellman optimality operator and the performance of the corresponding greedy policy?
- In the full version of this paper [1], we derived some data-dependent bounds on the excess-risk of a regression procedure that operates in a large function space which suited our immediate needs (i.e., $\bar{b}$ returned by REGRESS). However, the bound is asymptotic in nature and is potentially suboptimal. Can this bound be improved?

References

[1] Amir-massoud Farahmand and Csaba Szepesvári. Model selection in reinforcement learning. *Machine Learning Journal*, 85(3):299–332, 2011. URL `http://dx.doi.org/10.1007/s10994-011-5254-7`.

[2] Csaba Szepesvári. *Algorithms for Reinforcement Learning*. Morgan Claypool Publishers, 2010.

[3] Andrew R. Barron. Complexity regularization with application to artificial neural networks. In G. Roussas, editor, *Nonparametric Function Estimation and Related Topics*, pages 561–576. Kluwer Academic Publishers, 1991.

[4] Peter L. Bartlett, Stéphane Boucheron, and Gábor Lugosi. Model selection and error estimation. *Machine Learning*, 48(1-3):85–113, 2002.

[5] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2001.