

# Least Squares Temporal Difference Learning and Galerkin's Method

Csaba Szepesvári

September 11, 2011

## Abstract

The problem of estimating the value function underlying a Markovian reward process is considered. As it is well known, the value function underlying a Markovian reward process satisfied a linear fixed point equation. One approach to learning the value function from finite data is to find a good approximation to the value function in a given (linear) subspace of the space of value functions. We review some of the issues that arise when following this approach, as well as some results that characterize the finite-sample performance of some of the algorithms.

## 1 Markovian Reward Processes

Let  $\mathcal{X}$  be a measurable space and consider a stochastic process

$$(X_0, R_1, X_1, R_2, X_2, \dots),$$

where  $X_t \in \mathcal{X}$  and  $R_{t+1} \in \mathbb{R}$ ,  $t = 0, 1, 2, \dots$ . The process is called a *Markovian Reward process* if

- $(X_0, X_1, \dots)$  is a Markov process, and
- for any  $t \geq 0$ , given  $X_t, X_{t+1}$  the distribution of  $R_{t+1}$  is independent of the history of the process.

Here,  $X_t$  is called the *state* of the system at time  $t$ , while  $R_{t+1}$  is the *reward* associated to transitioning from  $X_t$  to  $X_{t+1}$ . We shall denote by  $\mathcal{P}$  the Markovian kernel underlying the process: Thus, the distribution of  $(X_{t+1}, R_{t+1})$  given  $X_t$  is given by  $\mathcal{P}(\cdot, \cdot | X_t)$ ,  $t = 0, 1, \dots$ .

Fix the so-called *discount factor*  $0 \leq \gamma \leq 1$  and define the (total discounted) *return* associated to the process

$$\mathcal{R} = \sum_{t=0}^{\infty} \gamma^t R_{t+1}$$

and the value function

$$V^*(x) = \mathbb{E}[\mathcal{R} | X_0 = x], \quad x \in \mathcal{X}.$$

For simplicity, assume that the rewards are bounded with probability one, say,  $\sup_{t \geq 0} |R_{t+1}| \leq 1$  a.s. Further, assume that the support of the distribution of  $X_0$  is the entire state space  $\mathcal{X}$ . Under these conditions and if  $\gamma < 1$ ,  $V^* : \mathcal{X} \rightarrow \mathbb{R}$  is well-defined and depends solely on the transition probability kernel  $\mathcal{P}$ . When  $\gamma = 1$ , further conditions are necessary to ensure that  $V$  is well-defined. Hence, for simplicity in the theoretical developments below we shall assume that  $\gamma < 1$ .

## 2 Markovian Value Prediction Problems

The *Markovian value prediction problem* is to estimate  $V$  given data that follows the law of the underlying Markovian Reward Process. This problem arises in a number of applications (e.g., predicting long term

values in financial applications, predicting whether a biped robot is going to fall, or as a subproblem of approximate dynamic programming algorithms).

By overloading some of the letters, let  $D_n = ((X_1, R_2, X'_2), \dots, (X_n, R_{n_1}, X'_n))$  be the data available for the statistician, where the distribution of  $(X'_t, R_{t+1})$  given the history  $((X_1, R_2, X'_2), \dots, (X_{t-1}, R_t, X'_t))$  and  $X_t$  is  $\mathcal{P}(\cdot|X_t)$ . Two important special cases are the following:

- (i) The process  $((X_t, R_{t+1}, X'_{t+1}))_{1 \leq t \leq n}$  is i.i.d.
- (ii) For all  $i$ ,  $X_{t+1} = X'_{t+1}$ .

The second case arises in applications where only a single *trajectory* is available for learning, while the first is mainly used as a convenient simplifying assumption when studying the performance of learning methods (a situation, when the first assumption is met in practice is when a simulator of the Markovian Reward Process is available). In practice, one might also have a dataset that consists of a number of trajectories whose start states are independently sampled from each other.

If the training data was an infinitely long trajectory, one could approach estimating  $V^*$  by casting it as a regression problem. Indeed, when  $n = \infty$ ,  $\mathcal{R}_t = \sum_{j=t}^{\infty} \gamma^{j-t} R_{j+1}$  provides an unbiased estimate of  $V^*(X_t)$ :  $V^*(X_t) = \mathbb{E}[\mathcal{R}_t|X_t]$ . Of course, in practice  $n$  is finite. Then, one is forced to truncate the estimates  $\mathcal{R}_t$  and use  $\mathcal{R}_t^{(n)} = \sum_{j=t}^n \gamma^{j-t} R_{j+1}$ . This way, however, a bias of size  $O(\gamma^{n-t})$  is introduced at sample  $t$ . Thus, for  $t$  large, the bias introduced is considerable, whereas for  $t$  small, the variance of  $\mathcal{R}_t^{(n)}$  can be large. The approach can be extended to the case when the data consists of a set of long trajectories. However, this approach will not work when the data consists of short trajectories (i.e., case i above).

### 3 Least squares temporal difference learning: A Statistical Galerkin Method

The approach proposed by Bradtke and Barto (1996) allows one to deal with such sequences of short trajectories. This approach was inspired by earlier work by Sutton (1988). In fact, it can be viewed as the least-squares analogue of the stochastic approximation algorithm proposed by Sutton (1988). Here, we show how this approach can be viewed as a “statistical Galerkin method”. Thus our approach to introduce this method will be different than the usual approach. Although the connection to Galerkin methods has been acknowledged beforehand (e.g., Yu and Bertsekas (2010)), the connection is not very well known amongst reinforcement learning researchers and hence we find it useful to explain it. For yet another alternative derivation of the method, as well as a statistical analysis, see Antos et al. (2008). For the analysis of non-parametric, regularized versions of the method, see, e.g., Farahmand et al. (2009).

As it is well known, the value function  $V^*$  is the solution to the fixed-point equation

$$TV^* = V^*, \quad (1)$$

where  $T : L^\infty(\mathcal{X}) \rightarrow L^\infty(\mathcal{X})$  is the so-called *Bellman-operator*:

$$TV(x) = \int (r + \gamma V(x')) d\mathcal{P}(x', r|x), \quad V \in L^\infty(\mathcal{X}).$$

As it can be seen from the definition,  $T$  is an affine linear operator, so (1) is a linear fixed point equation. Also,  $T$  is a bounded and is in fact a  $\gamma$ -contraction.

The *projection method* for solving (1) consists of choosing two finite dimensional linear subspaces  $\mathcal{F}, \mathcal{G}$  of a Banach space  $\mathcal{B}$  of functions over  $\mathcal{X}$ ,  $\mathcal{F}, \mathcal{G}$  sharing a common dimension  $d \in \mathbb{N}$ , and then solving the projected equation

$$\Pi_{\mathcal{G}} TV = \Pi_{\mathcal{G}} V \quad \text{for } V \in \mathcal{F}, \quad (2)$$

where  $\Pi_{\mathcal{G}} : \mathcal{B} \rightarrow \mathcal{F}$  is a projection operator (Kirsch, 2011). It can be easily seen that this leads to a  $d \times d$  linear system of equations once one fixes some bases for  $\mathcal{F}$  and  $\mathcal{G}$ . When  $\mathcal{B}$  is a pre-Hilbert space,  $\Pi_{\mathcal{G}}$  is the corresponding orthogonal projection and  $T : \mathcal{F} \rightarrow \mathcal{B}$  is bounded, we arrive at a *Galerkin method*. In this case, (2) is equivalent to

$$\langle TV, g \rangle = \langle V, g \rangle \quad \text{for all } g \in \mathcal{G}. \quad (3)$$

Write  $TV = \bar{r} + \gamma \mathcal{P}V$ , where  $\bar{r}(x) = \int r d\mathcal{P}(x', r|x)$  and  $\mathcal{P} : L^\infty(\mathcal{X}) \rightarrow L^\infty(\mathcal{X})$  is defined by  $\mathcal{P}V = \int V(x') d\mathcal{P}(x', r|x)$ . Then, assuming that  $\mathcal{F} = \text{span}(f_1, \dots, f_d)$ ,  $\mathcal{G} = \text{span}(g_1, \dots, g_d)$ , writing  $V = \sum_{i=1}^d \alpha_i f_i$ , (3) leads to the linear system of equations

$$\sum_{i=1}^d \alpha_i \langle (I - \gamma \mathcal{P})f_i, g_j \rangle = \langle \bar{r}, g_j \rangle, \quad 1 \leq j \leq d.$$

or

$$A\alpha = b,$$

where  $A_{ij} = \langle (I - \gamma \mathcal{P})f_i, g_j \rangle$ ,  $b_j = \langle \bar{r}, g_j \rangle$ . For the error analysis of Galerkin's method, see, e.g., Chapter 3 of (Kirsch, 2011). These results should be compared to that of Yu and Bertsekas (2010) and Scherrer (2010).

Now consider the case when we are given a sample  $D_n$  as described in the previous section. Further, for simplicity, assume that  $(X_t)_{1 \leq t \leq n}$  is stationary and let  $\mu$  be the common distribution underlying  $(X_t)$ . Choose  $\mathcal{B}$  to be the Hilbert space  $L^2(\mu)$ . Given the sample  $D_n$ , approximate  $A_{ij}$  by

$$\hat{A}_{ij}^{(n)} = \frac{1}{n} \sum_{t=1}^n (f_i(X_t) - \gamma f_i(X'_t)) g_j(X_t)$$

and approximate  $b_j$  by

$$\hat{b}_j^{(n)} = \frac{1}{n} \sum_{t=1}^n R_t g_j(X_t).$$

The method of Bradtke and Barto (1996) corresponds to the case when  $\mathcal{F} = \mathcal{G}$  (the corresponding "exact" method is the so-called Bubnov-Galerkin method).

Under various additional assumptions (e.g., independence of the snippets),  $\hat{A}_{ij}^{(n)}$  and  $\hat{b}_{ij}^{(n)}$  converge with probability one to  $A_{ij}$  and  $b_j$ , respectively. A stability analysis of the equation  $A\alpha = b$  can then be used to derive bounds on the quality of solutions obtained by solving  $\hat{A}^{(n)}\alpha = \hat{b}^{(n)}$ . One can further introduce appropriate regularization to appropriately stabilize the estimation process in the case of large dimensions ( $d$ ) and small sample sizes ( $n$ ). Details are available in the forthcoming paper (Pires and Szepesvári, 2011).

## 4 Conclusions

We have discussed the connection between a popular method in reinforcement learning, the so-called least-squares temporal difference (LSTD) method, and Galerkin's method. Although this connection was recognized before, we found it useful to explain it as it is lesser known within the reinforcement learning community and because the connection leads to new insights into the issues related to statistical performance of this important algorithm. We hope that this short abstract will foster further research to explore this and other connections between reinforcement learning and statistical inverse problems.

## References

- Antos, A., Szepesvári, C., and Munos, R. (2008). Learning near-optimal policies with Bellman-residual minimization based fitted policy iteration and a single sample path. *Machine Learning*, 71(1):89–129. Published Online First: 14 Nov, 2007.

- Bradtke, S. J. and Barto, A. G. (1996). Linear least-squares algorithms for temporal difference learning. *Machine Learning*, 22:33–57.
- Farahmand, A., Ghavamzadeh, M., Szepesvári, C., and Mannor, S. (2009). Regularized policy iteration. In Koller, D., Schuurmans, D., Bengio, Y., and Bottou, L., editors, *NIPS-21*, pages 441–448. MIT Press.
- Kirsch, A. (2011). *An Introduction to the Mathematical Theory of Inverse Problems*. Springer, 2nd edition.
- Pires, B. and Szepesvári, C. (2011). Statistical linear estimation with penalized estimators: an application to reinforcement learning. under preparation.
- Scherrer, B. (2010). Should one compute the temporal difference fix point or minimize the Bellman residual? The unified oblique projection view. In Fürnkranz, J. and Joachims, T., editors, *ICML 2010*, pages 959–966.
- Sutton, R. S. (1988). Learning to predict by the method of temporal differences. *Machine Learning*, 3(1):9–44.
- Yu, H. and Bertsekas, D. (2010). Error bounds for approximations from projected linear equations. *Mathematics of Operations Research*, 35(2):306–329.