18.3 Projected Equation Methods

The direct methods of section 18.2 are perhaps the most straightforward way of fitting an approximation architecture for policy evaluation based on simulated samples. An alternative approach for policy evaluation, which is actually preferred and referred to as *indirect approximation method*, is to try to solve a *projected form of Bellman's equation* $J = T_\mu J$ on the subspace

$$S = \text{span}\{\phi_1, \ldots, \phi_m\},$$

i.e., $S = \text{im } \Phi$ for finite-state spaces. With these methods, we aim to find a weight vector r_μ such that

$$\Phi r_\mu = \Pi T_\mu(\Phi r_\mu), \tag{18.19}$$

where Π is a linear projection on the subspace S. We view Φr_μ as an approximation of J_μ. Note that (18.19) is linear in r, and we solve equations in a smaller-dimensional space (dimension m) than when attacking Bellman's equation for J_μ directly (dimension $n >> m$). This approach is actually a popular technique in numerical analysis[5], but here it is coupled with stochastic simulation ideas. In this section, we consider exclusively linear approximation architectures $\tilde{J}(x, r) = \phi(x)^T r, x \in \mathsf{X}$.

Since we assume that the policy μ to be evaluated is fixed, the state evolves as a Markov chain. Let us consider a finite state space $\mathsf{X} = \{1, \ldots, n\}$ with the following assumptions

1. The Markov chain has steady-state probabilities $\xi = [\xi_1, \ldots, \xi_n]$ that are positive, i.e., for all $i = 1, \ldots, n$

$$\lim_{N\to\infty} \frac{1}{N} \sum_{k=1}^{N} P(x_k = j | x_0 = i) = \xi_j, \; j = 1, \ldots, n.$$

2. The matrix Φ as full column rank.

Assumption 1 is equivalent to assuming the the Markov chain is irreducible, i.e., has a single communication class (hence recurrent, and there are no transient states). Assumption 2 is equivalent to the basis functions $\phi_i, i = 1, \ldots, s$ being linearly independent and imply that any vector $J \in \text{im } \Phi$ as a *unique* decomposition $J = \Phi r$. We will also use the weighted Euclidian norm, defined for $w = [w_1, \ldots, w_n]^T$, with $w_i > 0$ for all i, by

$$\|J\|_{2,w} = \left(\sum_{i=1}^{n} w_i (J(i))^2 \right)^{1/2} = \sqrt{J^T W J},$$

[5] see e.g. Galerkin methods for continuous operator problems such as differential equations.

with $W = \text{diag}(w)$. Let Π_w be the orthogonal projection onto S with respect to this norm. $\Pi_w J$ is then the unique vector $\hat{J}$ in S that minimizes $\|J - \hat{J}\|_{2,v}$ over all vectors in S. Because of the full column rank assumption on Φ, we can write uniquely $\hat{J} = \Phi r_J$ where

$$r_J = \arg\min_{r \in \mathbb{R}^s} \|J - \Phi r\|_{2,v}.$$

In fact, it is not hard to see in this case that we have $r_J = (\Phi^T W \Phi)^{-1} \Phi^T W J$.

The first question we need to address is that of the existence of a fixed point for the equation

$$\Phi r = \Pi_w T_\mu(\Phi r).$$

For the standard Bellman equation, this followed from the fact that T_μ is an α-contraction for $\|\cdot\|_\infty$, see theorem 6.5.1. We would like to follow the same idea here. However, because of the composition with the orthogonal projection Π_w, it becomes more convenient to work with the weigthed 2-norm. Note first that Π_w is *nonexpansive* for $\|\cdot\|_{2,w}$, i.e.

$$\|\Pi_w J - \Pi_w \bar{J}\|_{2,w} \leq \|J - \bar{J}\|_{2,v}, \ \forall J, \bar{J} \in \mathbb{R}^n. \tag{18.20}$$

Exercise 18. Prove the property (18.20) (hint: Pythagorean theorem).

Hence if we can prove that that T_μ is contraction with respect to $\|\cdot\|_{2,w}$, then this is also true for the composition $\Pi_w T_\mu$. Unfortunately the contraction property of T_μ does not hold in general for the (weighted) 2-norm. In fact, the iteration

$$\Phi r_{k+1} = \Pi_w T_\mu(\Phi r_k),$$

can even diverge. For an example, fix the discount factor $\alpha \in (0,1)$ and take an uncontrolled two-state Markov chain with transition matrix

$$P = \begin{bmatrix} \epsilon & 1-\epsilon \\ \epsilon & 1-\epsilon \end{bmatrix},$$

and stage costs $c(1) = c(2) = 0$. Hence the total cost is $J^* = [0,0]^T$. Next, take just one basis function $\phi_1 = [1,2]^T$. We have

$$T\Phi r_k = \alpha P \phi_1 r_k = \alpha e \begin{bmatrix} \epsilon & 1-\epsilon \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix} r_k = (\alpha(2-\epsilon) r_k) e, \ \ e = [1,1]^T.$$

Next, consider the orthogonal projection with respect to the standard Euclidean norm

$$r_{k+1} = \arg\min_r \{(r - (\alpha(2-\epsilon)r_k))^2 + (2r - (\alpha(2-\epsilon)r_k))^2\} = \frac{3}{5}\alpha(2-\epsilon) r_k.$$

Hence the sequence $\{r_k\}_k$ diverges if ϵ is close to 0 and α close to 1.

However, an important case where T_μ turns out to be a contraction with respect to $\|\cdot\|_{2,w}$ is when $w = \xi$, i.e., the weights are the chain's steady-state probabilities. In the example above, the steady-state chain spends only the proportion ϵ of its time in state 1. It thus seems more sensible to weight the two states differently in the cost criterion. The contraction property essentially follows from the following lemma.

Lemma 18.3.1. *Let P be a $n \times n$ stochastic matrix with stationary distribution ξ such that $\xi_i > 0, i = 1, \ldots, n$. Then*

$$\|PJ\|_{2,\xi} \leq \|J\|_{2,\xi}, \ \forall z \in \mathbb{R}^n.$$

Proof.

$$\begin{aligned}
\|PJ\|_{2,\xi}^2 &= \sum_{i=1}^n \xi_i([PJ]_i)^2 = \sum_{i=1}^n \xi_i(E[J(X_1)|X_0 = i])^2 \\
&\leq \sum_{i=1}^n \xi_i E[(J(X_1))^2|X_0 = i] \quad \text{(Jensen's inequality)} \\
&= E_\xi[J(X_1))^2] = E_\xi[J(X_0))^2] = \|J\|_\xi^2 \quad \text{(definition of } \xi\text{, stationary).}
\end{aligned}$$

□

Hence we have immediately the following proposition.

Proposition 18.3.2. *The mappings T_μ and $\Pi_\xi T_\mu$ are α-contractions for the norm $\|\cdot\|_{2,\xi}$, where ξ is the stationary distribution of the Markov chain corresponding to μ.*

Proof. Since Π_w is nonexpansive, it is sufficient to prove the result for T_μ. Recall that $T_\mu J = c_\mu + \alpha P_\mu J$. Hence for all $J, \bar{J} \in \mathbb{R}^n$

$$\|T_\mu J - T_\mu \bar{J}\|_{2,\xi} = \alpha \|P_\mu (J - \bar{J})\|_{2,\xi} \leq \alpha \|J - \bar{J}\|_{2,\xi},$$

using lemma 18.3.1, and we are done. □

Since $\Pi_\xi T_\mu$ is an α-contraction for $\|\cdot\|_{2,\xi}$ and the space of functions $\mathsf{X} \to \mathbb{R}$ with finite weighted 2-norm is complete (it is a Hilbert space), we conclude that $\Pi_\xi T_\mu$ has a unique fixed point $\hat{J}_\mu \in \text{im } \Phi$. By our assumption on Φ having full column rank, there is a unique $r_\mu \in \mathbb{R}^s$ such that $\hat{J}_\mu = \Phi r_\mu$. We would like now to have an estimate of the error in approximating J_μ by $\hat{J}_\mu$. Note that the best we can expect to achieve is the projection $\Pi_\xi J_\mu$ of J_μ on S, that is, the performance cannot be good if the choice of approximation architecture is poor. In general, $\hat{J}_\mu$ is not equal to the projection $\Pi_\xi J_\mu$, but we have the following bound.

Proposition 18.3.3. *Let $\hat{J}_\mu$ be the unique fixed point of $\Pi_\xi T_\mu$. Then we have the error bound*

$$\|J_\mu - \hat{J}_\mu\|_{2,\xi} \leq \frac{1}{\sqrt{1 - \alpha^2}} \|J_\mu - \Pi_\xi J_\mu\|_{2,\xi}.$$

Proof. Write

$$\begin{aligned}\|J_\mu - \hat{J}_\mu\|_{2,\xi}^2 &= \|J_\mu - \Pi_\xi J_\mu\|_{2,\xi}^2 + \|\Pi_\xi J_\mu - \hat{J}_\mu\|_{2,\xi}^2 \text{ (Pythagorean theorem)} \\ &= \|J_\mu - \Pi_\xi J_\mu\|_{2,\xi}^2 + \|\Pi_\xi T_\mu J_\mu - \Pi_\xi T_\mu \hat{J}_\mu\|_{2,\xi}^2 \text{ (def. of } J_\mu \text{ and } \hat{J}_\mu) \\ &\leq \|J_\mu - \Pi_\xi J_\mu\|_{2,\xi}^2 + \alpha^2 \|J_\mu - \hat{J}_\mu\|_{2,\xi}^2 \ \ (\Pi_\xi T_\mu \ \alpha\text{-contraction}).\end{aligned}$$

□

Looking at the projected Bellman's equation in matrix form for a finite state space, writing $\Xi = \text{diag}(\xi)$ and $\hat{J}_\mu = \Phi r_\mu$, we know that r_μ also verifies

$$r_\mu = \arg\min_{r \in \mathbb{R}^s} \|\Phi r - (c_\mu + \alpha P_\mu \hat{J}_\mu)\|_{2,\xi}^2. \tag{18.21}$$

Note the somewhat subtle fact here that (18.21) is *not*

$$\arg\min_{r \in \mathbb{R}^s} \|\Phi r - (c_\mu + \alpha P_\mu \Phi r)\|_{2,\xi}^2. \tag{18.22}$$

In fact, (18.22) is the Bellman equation error approach mentioned earlier and discussed in section 18.4. By setting the gradient of the expression in (18.21) to 0 we see that r_μ must satisfy the following linear system of equations

$$C r_\mu = d, \ \text{ with } C := \Phi^T \Xi (I - \alpha P_\mu)\Phi, \ d := \Phi^T \Xi c_\mu. \tag{18.23}$$

Under assumption 2 this system has a unique solution $r_\mu = C^{-1} d$ and

$$\hat{J}_\mu = \Phi r_\mu = \Phi(\Phi^T \Xi (I - \alpha P_\mu)\Phi)^{-1} \Phi^T \Xi c_\mu.$$

Compare to the original Bellman's equation

$$J_\mu = (I - \alpha P_\mu)^{-1} c_\mu,$$

which requires solving an $n \times n$ system of linear equations, whereas computing r_μ now involves a typically much smaller $m \times m$ system. However, explicitly computing C and d using their definitions in (18.23) still requires computing inner products of size n, which can be impractical. Maybe more crucially, we do not now the stationary distribution Ξ in general and computing it directly is usually extremely difficult! Simulation is used to address both problems.

Before introducing simulation however, let us describe the analog of the value iteration algorithm for the projected equation. We start with a vector r_0 and compute the iterates

$$\Phi r_{k+1} = \Pi_\xi T_\mu (\Phi r_k).$$

This can theoretically be accomplished by first computing $T_\mu(\Phi r_k)$ (this is not practical because this vector lives in an n-dimensional space) and then projecting on S using Π_ξ (this is not practical since this requires the knowledge of ξ), i.e.,

$$r_{k+1} \in \arg\min_{r \in \mathbb{R}^s} \|\Phi r - (c_\mu + \alpha P_\mu \Phi r_k)\|_{2,\xi}^2. \tag{18.24}$$

We call this theoretical algorithm *projected value iteration* (PVI). Its convergence to $\hat{J}_\mu$ follows from the fact that $\Pi_\xi T_\mu$ is a contraction. Under assumption 2, the solution r_{k+1} is unique and satisfies

$$\begin{aligned}
&\Phi^T \Xi \Phi r_{k+1} - \Phi^T \Xi (c_\mu + \alpha P_\mu \Phi r_k) = 0 \\
&\Phi^T \Xi \Phi r_{k+1} - d + C r_k - \Phi^T \Xi \Phi r_k = 0 \\
&r_{k+1} = r_k - (\Phi^T \Xi \Phi)^{-1} (C r_k - d). \qquad (18.25)
\end{aligned}$$

Note that r_μ is the unique fixed point of (18.25). Now (18.25) can be seen as an example of more general iterative algorithms to solve the system $Cr = d$, of the form

$$r_{k+1} = r_k - \gamma_k D_k^{-1} (C r_k - d), \qquad (18.26)$$

where γ_k is a positive stepsize and D_k is a positive definite symmetric matrix. Fixing $\gamma_k = \gamma$ and $D_k = D$ for all k, the iterates (18.26) converge to the solution of $Cr = d$ is and only if the eigenvalues of $I - \gamma D^{-1} C$ are strictly within the unit circle. This turns out to be true for any D positive definite and γ small enough, see [Ber07b, prop. 6.3.3].

Simulation Based Approximations

In practice we can form approximations of C and d using simulations, and then use these approximations in $r = C^{-1} d$, as well as in the PVI algorithm (18.25) or (18.26). Treating the simulation variations as noise, we obtain stochastic approximation algorithms for solving the equation $f(r) := Cr - d = 0$, where we can only measure $f(r)$ up to noise entering through the coefficients C and d. Recall the definitions

$$\begin{aligned}
C &= \Phi^T \Xi (I - \alpha P_\mu) \Phi \\
d &= \Phi^T \Xi c_\mu .
\end{aligned}$$

Rewritten more explicitly, and recalling the definition of the feature vector $\phi(x) = [\phi_1(x), \ldots, \phi_m(x)]^T$ ($\phi^T(x)$ is a row of Φ, so $\phi(x)$ is a column of Φ^T), we have (below we write $[P_\mu \phi](i) = \sum_{j=1}^n p_{ij}(\mu(i)) \phi(j)$, which is the expected

feature vector of the next state given that the current state is i[6])

$$\begin{aligned}
C &= \sum_{i=1}^{n} \xi(i)\phi(i)(\phi(i) - \alpha[P_\mu \phi](i))^T \\
&= E_\xi\Big[\phi(x)\Big(\phi(x) - \alpha E[\phi(x_1)|x_0 = x, u_0 = \mu(x)]\Big)^T\Big], \\
&= E_\xi^\mu\Big[\phi(x_0)\Big(\phi(x_0) - \alpha\phi(x_1)\Big)^T\Big], \\
d &= \sum_{i=1}^{n} \xi(i)\phi(i)c_\mu(i) = E_\xi[\phi(x)c_\mu(x)] \\
&= E_\xi\Big[\phi(x)E[c(x_0, u_0, x_1)|x_0 = x, u_0 = \mu(x)]\Big] \\
&= E_\xi^\mu\Big[\phi(x_0)c(x_0, \mu(x_0), x_1)\Big].
\end{aligned}$$

Here $E_\xi^\mu[f(x_0, x_1)]$ is the expectation operator for the Markov chain (with the policy fixed to μ) assuming that x_0 is distributed according to ξ. Now consider a simulated trajectory $(x_0, x_1, \ldots)$. When x_k is generated, we can compute its feature vector $\phi(x_k)$ and when a transition (x_k, x_{k+1}) is generated we can compute the transition cost $c(x_k, \mu(x_k), x_{k+1})$. After $k+1$ such transitions are generated, consider the empirical versions of C and d above:

$$C_k = \frac{1}{k+1}\sum_{t=0}^{k} \phi(x_t)(\phi(x_t) - \alpha\phi(x_{t+1}))^T \tag{18.27}$$

$$d_k = \frac{1}{k+1}\sum_{t=0}^{k} \phi(x_t)c(x_t, \mu(x_t), x_{t+1}). \tag{18.28}$$

The law of large numbers, which is assumed to hold for our Markov chain, says that $C_k \to C$ and $d_k \to d$ almost surely. Note also that we have the recursive updates formulas for C_k and d_k, $k \geq 1$,

$$\begin{aligned}
C_k &= C_{k-1} + \frac{1}{k+1}\Big[\phi(x_k)(\phi(x_k) - \alpha\phi(x_{k+1}))^T - C_{k-1}\Big] \\
d_k &= d_{k-1} + \frac{1}{k+1}\Big[\phi(x_k)c(x_k, \mu(x_k), x_{k+1}) - d_{k-1}\Big].
\end{aligned}$$

These formulas allow us to update the value of C_k and d_k after a transition (x_k, x_{k+1}) is generated.

Least Squares Temporal Differences (LSTD)

This method uses the empirical (simulation-based) versions C_k and d_k of C and d to construct a simulation-based approximate solution

$$\hat{r}_k = C_k^{-1} d_k.$$

[6]Note that $\phi(i)$ is a vector of length m here, not a number.

As we saw above, C_k and d_k can be updated recursively (in fact, we update C_k^{-1} recursively using the matrix inversion lemma, see below for LSPE), but this method is not a true recursive method since we do not use $\hat{r}_{k-1}$ to compute $\hat{r}_k$. Using (18.27) and (18.28), we can also write the equation $C_k r_k = d_k$ as

$$\sum_{t=0}^{k} \phi(x_t) q_{k,t} = 0,$$

where $q_{k,t}$ is the temporal difference associated with r_k and the transition (x_t, x_{t+1})

$$q_{k,t} = \phi(x_t)^T r_k - \alpha \phi(x_{t+1})^T r_k - c(x_t, \mu(x_t), x_{t+1}).$$

As usual with linear systems of equations, a difficulty arises in LSTD if C_k and C are nearly singular, since then the solution is strongly sensitive to changes in the problem data, rounding errors and in our case the simulation-induced error

$$\hat{r}_k - r = C_k^{-1} d_k - C^{-1} d$$

is greatly amplified. If the discount factor α is significantly smaller than 1, this is not a problem when the number of samples is sufficiently large. But standard LSTD can run into serious singularity issues for C_k as α becomes close to 1 or for nondiscounted problems (e.g. stochastic shortest path or average cost problems). The standard solution to this problem is to use some form of regularized regression, which works even if the matrices are singular, at the cost of introducing some bias in the estimate. That is, we choose r_k by solving the least squares problem

$$r_k \in \arg\min_r \|d_k - C_k r\|^2_{2,\Sigma^{-1}} + \|r - \bar{r}\|^2_{2,\Gamma^{-1}} \tag{18.29}$$

$$\text{i.e.,} \quad r_k \in \arg\min_r \Big\{ (d_k - C_k r)^T \Sigma^{-1} (d_k - C_k r) + (r - \bar{r})^T \Gamma^{-1} (r - \bar{r}) \Big\},$$

where $\bar{r}$ is some *a priori* estimate of $r^* = C^{-1} d$, and Σ, Γ are some positive definite symmetric matrices. Here $\bar{r}$ may be chosen based on intuition about the problem or may correspond to the cost $\Phi \bar{r}$ of a similar policy (e.g., a preceding policy in approximate policy iteration). The quadratic term $\|r - \bar{r}\|^2_{2,\Gamma^{-1}}$ is known as a *regularization term* and biases the estimate $\hat{r}_k$ towards the a priori guess $\bar{r}$. Typically we take $\Gamma^{-1} = \beta I$, with $\beta > 0$ chosen by trial-and-error. A large β reduces the effect of near singularity of C_k and the sensitivity to simulation errors, but may cause a large bias.

The explicit solution to (18.29) is

$$\hat{r}_k = (C_k^T \Sigma^{-1} C_k + \Gamma^{-1})^{-1} (C_k^T \Sigma^{-1} d_k + \Gamma^{-1} \bar{r}).$$

Writing the projected Bellman's equation using simulation as $d = C r_k - e_k$, with the simulation noise $e_k = (C - C_k) r_k + d_k - d$, a suitable choice for Σ in the regression is an estimate of the covariance of e_k. Let

$$W_t = \phi(x_t)(\phi(x_t) - \alpha \phi(x_{t+1}))^T$$
$$v_t = \phi(x_t) c(x_t, \mu(x_t), x_{t+1}).$$

These quantities can be viewed as samples of C_k and d_k, and we can view a vector

$$y_t = W_t \tilde{r} - v_t,$$

as a sample of the error e_k, where $\tilde{r}$ is another guess (perhaps different from $\bar{r}$ above) of the solution. Note that y_t has sample mean $C_k\tilde{r} - d_k$. We use its sample covariance matrix in the regression

$$\begin{aligned}\Sigma &= \frac{1}{k+1}\sum_{t=0}^{k}(y_t - C_k\tilde{r} + d_k)(y_t - C_k\tilde{r} + d_k)^T \\ &= \frac{1}{k+1}\sum_{t=0}^{k}((W_t - C_k)\tilde{r} + (d_k - v_t))((W_t - C_k)\tilde{r} + (d_k - v_t))^T.\end{aligned}$$

The error $\hat{r}_k - r_\mu$ made using the regularized regression (18.29) can be bounded in probability, following classical arguments developed for linear regression (perhaps the most well-studied statistical method). The analysis is based on the fact that for a large number of samples, the errors $d - Cr_k$ are asymptotically normal, see [Ber07b, prop. 6.3.4]. The bound involves a term that decreases to 0 as more samples are used, and a second term due to the bias error that cannot be made arbitrarily small (but which diminishes with β). The choice of Σ to be close to the covariance matrix of $d - Cr_k$ also comes from this analysis of regression errors.

Least Squares Policy Evaluation (LSPE)

As an alternative to LSTD, we obtain a true iterative method by using the approximations C_k and d_k in the Projected Value Iteration recurrence (18.26), to get

$$r_{k+1} = r_k - \gamma_k D_k^{-1}(C_k r_k - d_k), \tag{18.30}$$

where D_k is a positive definite matrix, γ_k is a positive stepsize, and C_k, d_k are given by (18.27), (18.28). In terms of temporal differences, we have

$$r_{k+1} = r_k - \frac{\gamma_k}{k+1} D_k^{-1} \sum_{t=0}^{k} \phi(x_t) q_{k,t} \tag{18.31}$$

Regarding the choice of γ_k and D_k, a first guideline is that if say $\gamma_k = \gamma$, $D_k \to D$, $C_k \to C$ and $d_k \to d$ such that $I - \gamma D^{-1}C$ has its eigenvalues strictly within the unit circle, then we generally have $r_k \to r_\mu = C^{-1}d$ (recall the convergence result mentioned for PVI). Also motivated by the first PVI equation (18.25), we could choose $\gamma_k = 1$ for all k and take D_k to be a simulation based approximation of $\Phi^T \Xi \Phi = E_\xi[\phi(x_0)\phi(x_0)^T]$, possibly corrected to ensure positive definiteness:

$$D_k = \frac{1}{k+1}(\beta I + \sum_{t=0}^{k} \phi(x_t)\phi(x_t)^T), \text{ with } \beta \geq 0.$$

With this choice of D_k, the method is known as the Least Squares Policy Evaluation (LSPE) method. We have the recursion

$$\begin{aligned} D_k &= \frac{k}{k+1} D_{k-1} + \frac{1}{k+1} \phi(x_k)\phi(x_k)^T \\ &= D_{k-1} + \frac{1}{k+1}(\phi(x_k)\phi(x_k)^T - D_{k-1}). \end{aligned} \tag{18.32}$$

Among possible variations, we could update D_k only periodically instead of doing so after every new sample to save computations. On the other hand, since we are interested in D_k^{-1}, we can use the matrix inversion lemma

$$(A - BD^{-1}C)^{-1} = A^{-1} - A^{-1}B(D - CA^{-1}B)^{-1}CA^{-1}$$

to get

$$\begin{aligned} D_k^{-1} &= \left(\frac{k}{k+1} D_{k-1} + \frac{1}{k+1} \phi(x_k)\phi(x_k)^T \right)^{-1} \\ D_k^{-1} &= \frac{k+1}{k} \left[D_{k-1}^{-1} + \frac{D_{k-1}^{-1}\phi(x_k)\phi(x_k)^T D_{k-1}^{-1}}{k + \phi(x_k)^T D_{k-1}^{-1} \phi(x_k)} \right]. \end{aligned}$$

Another possibility to simplify the matrix inversion is to use diagonal matrices D_k, such as a diagonal approximation of $\Phi^T \Xi \Phi$, for example by discarding the off-diagonal elements in (18.32).

Note that the choice of γ and D significantly affects the convergence rate of the deterministic PVI algorithm. However, for the simulation based version (18.30), the slower speed of simulation (i.e., the rate at which $C_k \to C$ and $d_k \to d$) dominates the faster (linear) convergence rate of PVI. In consequence the *asymptotic* rate of convergence of (18.30) does not depend on the choice of γ_k and D_k, as long as $I - \gamma D^{-1}C$ is a contraction. However, the short-term convergence rate may be significantly affected.

TD(0) Method

This method is the TD based version of LSPE (18.31) where we only keep the latest sample and take $D_k = I$

$$r_{k+1} = r_k - \gamma_k \phi(x_k) q_{k,k}. \tag{18.33}$$

Note that we recover the case of TD(0) encountered earlier in (18.16). Writing $f(r) = Cr - d$, we see that $\phi(x_k)q_{k,k}$ is a noisy sample of $f(r_k)$ which uses just one sampled state x_k instead of the average of the past samples used to compute C_k and d_k in LSPE. Hence TD(0) is essentially the simplest form of a Robbins-Monro scheme for solving the equation $f(r) = 0$ (see section 15.1), whereas LSPE uses averaging of the past samples. In general, the convergence of TD(0) is much slower than that of LSPE, and it requires $\gamma_k \to 0$ to deal with the nondiminishing noise in the term $\phi(x_k)q_{k,k}$. On the hand, it is easier to compute.

Optimistic Versions

Optimistic versions of LSTD and LSPE are discussed in [Ber07b, section 6.3.5].

LSTD(λ), LSPE(λ) and TD(λ)

Consider the operator

$$T_\mu^{(\lambda)} = (1-\lambda)\sum_{l=0}^{\infty}\lambda^l\, T_\mu^{l+1},$$

for $\lambda \in [0,1)$. For $\lambda = 0$, this is just the usual operator T. Corresponding to this operator is a weighted multistep Bellman equation

$$J = T_\mu^{(\lambda)}J = c_\mu^{(\lambda)} + \alpha P_\mu^{(\lambda)}J,$$

with

$$P_\mu^{(\lambda)} = (1-\lambda)\sum_{l=0}^{\infty}\alpha^l\lambda^l P_\mu^{l+1}, \quad c_\mu^{(\lambda)} = \sum_{l=0}^{\infty}\alpha^l\lambda^l P^l c_\mu = (I-\alpha\lambda P_\mu)^{-1}c_\mu.$$

Exercise 19. Verify the correctness of the expression of $T_\mu^{(\lambda)}$ above. Note that we have

$$T_\mu^{l+1}J = \alpha^{l+1}P_\mu^{l+1}J + \sum_{k=0}^{l}\alpha^k P_\mu^k c_\mu.$$

Note that the operators T_μ^l and $T_\mu^{(\lambda)}$ have the same fixed point J_μ. Hence we can apply the preceding algorithms to $T_\mu^{(\lambda)}$ in place of T_μ. The projected equations become

$$C^{(\lambda)}r_\mu^{(\lambda)} = d^{(\lambda)},$$

where

$$C^{(\lambda)} = \Phi^T\Xi(I-\alpha P^{(\lambda)})\Phi, \quad d^{(\lambda)} = \Phi^T\Xi c_\mu^{(\lambda)}.$$

The motivation for replacing T with $T^{(\lambda)}$ is that the modulus of contraction of $T^{(\lambda)}$ is smaller, resulting in a tighter error bound. We have

Proposition 18.3.4. *The mappings $T_\mu^{(\lambda)}$ and $\Pi_\xi T_\mu^{(\lambda)}$ are contractions with respect to $\|\cdot\|_\xi$, of modulus*

$$\alpha_\lambda = \frac{\alpha(1-\lambda)}{1-\alpha\lambda}.$$

Hence we have the error bound

$$\|J_\mu - \Phi r_\mu^{(\lambda)}\|_\xi \le \frac{1}{\sqrt{1-\alpha_\lambda^2}}\|J_\mu - \Pi_\xi J_\mu\|_\xi, \tag{18.34}$$

where $\Phi r_\mu^{(\lambda)}$ is the fixed point of $\Pi_\xi T_\mu^{(\lambda)}$.

Proof. The proof follows from the result of lemma 18.3.1, since

$$\begin{aligned}\|P_\mu^{(\lambda)} z\|_\xi &\le (1-\lambda)\sum_{l=0}^{\infty}\alpha^l\lambda^l\|P_\mu^{l+1}z\|_\xi \\ &\le (1-\lambda)\sum_{l=0}^{\infty}\alpha^l\lambda^l\|z\|_\xi \\ &= \frac{1-\lambda}{1-\alpha\lambda}\|z\|_\xi.\end{aligned}$$

□

Note that α_λ decreases and the error bound (18.34) becomes better as λ increases, with $\alpha_\lambda \to 0$ as $\lambda \to 1$. However, as λ increases, it turns out that the "simulation noise" becomes more pronounced. Another consequence of proposition 18.3.4 and of the equivalence of norms in $\mathbb{R}^n$ is that for any set of weights w, $T_\mu^{(\lambda)}$ is a contraction for $\|\cdot\|_{2,w}$ provided λ is sufficiently close to 1. Coming back to the simulation algorithms, note that

$$\begin{aligned}C^{(\lambda)} &= E_\xi\Big[\phi(x_0)\Big(\phi(x_0) - \alpha(1-\lambda)\sum_{l=0}^{\infty}\alpha^l\lambda^l\phi(x_{l+1})\Big)^T\Big] \\ &= E_\xi\Big[\phi(x_0)\Big(\sum_{t=0}^{\infty}(\alpha\lambda)^t(\phi(x_t) - \alpha\phi(x_{t+1}))\Big)^T\Big], \\ d^{(\lambda)} &= E_\xi\Big[\phi(x_0)\Big(\sum_{l=0}^{\infty}(\alpha\lambda)^l c(x_l,\mu(x_l),x_{l+1})\Big)\Big].\end{aligned}$$

Consider a simulation path $x_0, x_1, \ldots$. When the transition (x_k, x_{k+1}) is observed, the simulation approximations are then

$$\begin{aligned}C_k^{(\lambda)} &= \frac{1}{k+1}\sum_{t=0}^{k}\phi(x_t)\sum_{m=t}^{k}(\alpha\lambda)^{m-t}(\phi(x_m) - \alpha\phi(x_{m+1}))^T, \\ d_k^{(\lambda)} &= \frac{1}{k+1}\sum_{t=0}^{k}\phi(x_t)\sum_{m=t}^{k}(\alpha\lambda)^{m-t}c(x_m,\mu(x_m),x_{m+1}).\end{aligned}$$

If we replace C_k and d_k by $C_k^{(\lambda)}$ and $d_k^{(\lambda)}$, we obtain the so-called LSTD(λ) and LSPE(λ) methods. Again one can streamline the computations by introducing the vector

$$z_m = \sum_{t=0}^{m}(\alpha\lambda)^{m-t}\phi(x_t),$$

which evolves as

$$z_{m+1} = \alpha\lambda z_m + \phi(x_{m+1}).$$

Interchanging the sums in the definitions of $C_k^{(\lambda)}$ and $d_k^{(\lambda)}$, we have

$$\begin{aligned} C_k^{(\lambda)} &= \frac{1}{k+1}\sum_{t=0}^{k}\phi(x_t)\sum_{m=t}^{k}(\alpha\lambda)^{m-t}(\phi(x_m)-\alpha\phi(x_{m+1}))^T, \\ &= \frac{1}{k+1}\sum_{m=0}^{k}\left(\sum_{t=0}^{m}(\alpha\lambda)^{m-t}\phi(x_t)\right)(\phi(x_m)-\alpha\phi(x_{m+1}))^T \\ &= \frac{1}{k+1}\sum_{m=0}^{k}z_m(\phi(x_m)-\alpha\phi(x_{m+1}))^T, \end{aligned}$$

and similarly

$$d_k^{(\lambda)} = \frac{1}{k+1}\sum_{m=0}^{k}z_m c(x_m,\mu(x_m),x_{m+1}).$$

This allows us to easily update $C_k^{(\lambda)}, d_k^{(\lambda)}$ recursively as well, and the rank-one update of $\left(C_k^{(\lambda)}\right)^{-1}$ can be done efficiently using the matrix inversion lemma. Finally the iteration for LSPE(λ)

$$r_{k+1} = r_k - \gamma D_k^{-1}(C_k^{(\lambda)}r_k - d_k^{(\lambda)}), \quad D_k = \frac{1}{k+1}\sum_{t=0}^{k}\phi(x_t)\phi(x_t)^T,$$

can also be written

$$r_{k+1} = r_k - \frac{\gamma}{k+1}D_k^{-1}\sum_{t=0}^{k}z_t q_{k,t},$$

where $q_{k,t}$ is the usual temporal difference

$$q_{k,t} = \phi(x_t)^T r_k - \alpha\phi(x_{t+1})^T r_k - c(x_t,\mu(x_t),x_{t+1}).$$

Just as with TD(0), we can view the algorithm TD(λ) as a truncated version of LSPE(λ), which takes the form

$$r_{k+1} = r_k - \gamma_k z_k q_{k,k},$$

where γ_k is a stepsize parameter. This amounts to approximating $C^{(\lambda)}$ and $d^{(\lambda)}$ by one sample instead of $k+1$ samples.

Convergence of TD(0)

The TD(0) algorithm with a linear architecture has the simple form

$$r_{k+1} = r_k + \gamma_k\phi(x_k)(c(x_k,x_{k+1}) + \alpha\phi(x_{k+1})^T r_k - \phi(x_k)^T r_k),$$

where we use $c(x,y) := c(x,\mu(x),y)$ for notational simplicity. This is a type of stochastic approximation algorithm, where the noise is a function of the Markov chain $\{x_t\}_t$. We can rewrite it as

$$r_{k+1} = r_k + \gamma_k \Big[\bar{f}(r_k) + \Big(f(r_k, x_k, x_{k+1}) - \bar{f}(r_k) \Big) \Big],$$

with

$$\begin{aligned} f(r_k, x_k, x_{x+1}) &= \phi(x_k)(c(x_k, x_{k+1}) + \alpha \phi(x_{k+1})^T r_k - \phi(x_k)^T r_k), \\ \bar{f}(r) &= E_\xi[f(r, x_k, x_{k+1})] = \sum_{x,y} \xi_x P^\mu_{xy} f(r, x, y). \end{aligned}$$

In particular in the second expression note that we have taken the average with respect to the steady state distribution of the Markov chain. Although this is slightly more general than the situation described in chapter 15 (martingale difference noise), you can imagine that under appropriate conditions, in particular the same decreasing step-size conditions for γ_k, the iterates will asymptotically track the ODE

$$\dot{r} = \bar{f}(r). \tag{18.35}$$

Now we can write

$$\bar{f}(r) = \Phi^T \Xi (c_\mu + \alpha P_\mu \Phi r - \Phi r) = \Phi^T \Xi (T_\mu \Phi - \Phi) r.$$

An equilibrium point of the ODE (18.35) satisfies

$$\Phi^T \Xi (I - \alpha P_\mu) \Phi r = \Phi^T \Xi c_\mu,$$

and under our assumption 2, this equation has a unique solution, which is r_μ. This equilibrium is globally asymptotically stable for the ODE (18.35). Indeed, consider the Lyapunov function $V(r) = \frac{1}{2}\|r - r_\mu\|_2^2$. Its Lie derivative along the vector field is

$$\begin{aligned} \langle r - r_\mu, \bar{f}(r) \rangle &= \langle r - r_\mu, \Phi^T \Xi (T_\mu \Phi r - \Phi r) \rangle \\ &= \langle \Phi(r - r_\mu), T_\mu \Phi r - \Phi r \rangle_\xi \\ &= \langle \Phi(r - r_\mu), \Pi_\xi T_\mu \Phi r - \Phi r \rangle_\xi \\ &= \langle \Phi(r - r_\mu), \Pi_\xi T_\mu \Phi r - \Pi_\xi T_\mu \Phi r_\mu \rangle_\xi - \langle \Phi(r - r_\mu), \Phi r - \Phi r_\mu \rangle_\xi \\ &\leq -(1 - \alpha) \|\Phi r - \Phi r_\mu\|_{2,\xi}^2 \\ &< 0. \end{aligned}$$

In the third equality above, the introduction of Π_ξ is valid due to the fact that the orthogonal components does not contribute to the scalar product. In the fourth equality, we simply use the definition of the fixed point r_μ. Note the similarity with the convergence proof for the "fixed-point ode" of theorem 15.2.1.