Online Variants of Value Iteration and Approximation Via State Aggregation

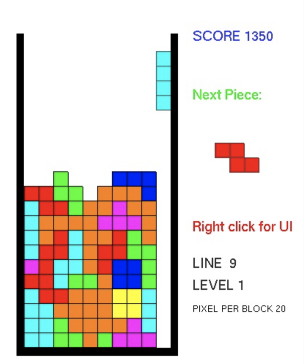
Daniel Russo

March 23, 2020

Columbia University

Table of Contents

Where are we heading? Value function approximation



- State $s \in \{0,1\}^{10\times 20}$ is a board configuration.
- One approach is to fit a linear approximation: $J^*(s) \approx J_\theta(s) := \phi(s)^\top \theta$.
- $\phi(s)$ encodes features. E.g. column heights, inter-column height differences, max height etc.
- Greedy action under J_θ is simple: move the piece to the position with minimal estimated cost-to-go.

Fitted Value Iteration (FVI)

Regression based approximation to Bellman updates.

- Define the weighted norm $\|J\|_{2,\nu} = \sqrt{\mathbb{E}_{s\sim\nu}[J(s)^2]}$.
- Fitted value iteration is the scheme: for $k = 1, 2, \cdots$

$$\theta_{k+1} \in \operatorname*{argmin}_{\theta\in\Theta} \|J_\theta - TJ_{\theta_k}\|_{2,\nu}$$

- Equivalently, this can be viewed as a projected value iteration:

$$J_{k+1} = \Pi_{\mathcal{F},\nu} TJ_k$$

  where $\mathcal{F} = \{J_\theta \mid \theta \in \Theta\}$ is the space of value functions approximations and $\Pi_{\mathcal{F},\nu} J = \operatorname{argmin}_{f\in\mathcal{F}} \|f - J\|_{2,\nu}$ projects onto $\mathcal{F}$ in a weighted norm.

Sample based approximations to FVI

In practice, we typically approximate expectations by random sampling.

1. Sample n states $s_1, \cdots, s_n \sim \rho(\cdot)$.
2. For each state s_i, and each control $u \in U(s)$, draw m samples of the successor states $s_{i,u}^{(1)}, \cdots, s_{i,u}^{(m)} \sim P_{s_i,\cdot}(u)$,
3. For $k = 0, 1, 2, \cdot$

$$\theta_{k+1} \in \operatorname*{argmin}_{\theta} \frac{1}{n} \sum_{i=1}^{n} \left(J_\theta(s_i) - \underbrace{\left(\min_u g(s_i, u) + \gamma \frac{1}{m} \sum_{j=1}^{m} J_{\theta_k}(s_{i,u}^{(j)}) \right)}_{\text{sample approx. to } TJ_{\theta_k}(s_i)} \right)$$

Comments:

(a) If state transitions are very sparse, $TJ(s)$ can be computed exactly.

(b) Statistical learning theory bounds the error from random sampling.

Fitted Q-iteration

It is simpler to approximate Bellman updates to Q-functions.

- Define the state-action value function:

$$Q^*(s,u) = g(s,u) + \alpha \sum_{s' \in S} P_{ss'}(u) J^*(s')$$

- Obeys the Bellman $Q^* = FQ^*$ where

$$FQ(s,u) := g(s,u) + \alpha \sum_{s' \in S} P_{ss'}(u) \min_{u'} Q(s',u')$$

- Fitted Q iteration is the scheme: for $k = 1, 2, \cdots$

$$\theta_{k+1} \in \underset{\theta \in \Theta}{\operatorname{argmin}} \| Q_\theta - FQ_{\theta_k} \|_{2,\nu} \tag{1}$$

  where ν is a distribution over state, control pairs.

The Q-learning algorithm essentially makes a stochastic gradient updates rather than solving (1) *exactly.*

Does this work?

1. Convergence to a fixed point $\widehat{J} = \Pi_{\rho,\mathcal{F}} T \widehat{J}$?
2. Does it produce an accurate approximation if J^* is "close to" the function class $\mathcal{F}$?
3. Is the resulting policy near optimal?
4. How should we set the state-importance-weights ν?

Unfortunately . . . this procedure does not converge in general. Existing guarantees on typically performance require extremely strong assumptions.

Plan for today

1. State-aggregation: a simple case of value function approximation with which fitted-value-iteration is convergent.
2. Understanding the state-importance-weights and how this should be adapted over time.

State-Aggregation (a.k.a state abstraction)

- We believe $J^*(s) \approx J^*(\tilde{s})$ if s and $\tilde{s}$ are "similar."
- More formally, take $\phi : \mathcal{S} \to \mathcal{S}$ to a be a mapping that associates $s \in \mathcal{S}$ with a representative state $\phi(s) \in \mathcal{S}$.
  - Simple case is $\mathcal{S} = [0, 1]$ and ϕ maps $s \in [0, .01)$ to $\phi(s) = .005$, $s \in [.01, .02)$ to $\phi(s) = .015$ and so on.
- State aggregated value functions are

$$
\begin{aligned}
\mathcal{F}_\phi &= \{f | f(s) = f(\phi(s)) \, \forall s\} \\
&= \{f | f(s) = f(\tilde{s}) \quad \text{if } \phi(s) = \phi(\tilde{s})\}
\end{aligned}
$$

  - In the simple case described above, $J \in \mathcal{F}_\phi$ is defined by the 99 values $J(.005), J(.015), \cdots, J(.995)$.
- $\Phi = \{\phi(s) : s \in \mathcal{S}\} = \{1, \cdots, m\}$. $(\ldots \text{w.l.o.g })$
  Set $\mathcal{S}_i = \{s \in \mathcal{S} : \phi(s) = i\}$ for $i = 1, \cdots, m$.

Table of Contents

Convergence of FVI with state-aggregation

Theorem
Consider the iteration

$$J_{k+1} = \Pi_{\mathcal{F}_\phi,\nu} T J_k \quad k = 0, 1, \cdots$$

where $\nu(s) > 0$ *for all* $s \in \mathcal{S}$*. Then,*

$$\|J_k - \widehat{J}\|_\infty \leq \alpha^k \|J_0 - \widehat{J}\|_\infty$$

where $\widehat{J}$ *solves the projected Bellman equation*

$$\widehat{J} = \Pi_{\mathcal{F}_\phi,\nu} T \widehat{J}.$$

Convergence proof

Crucial fact: $\Pi_{\mathcal{F}_\phi,\nu}$ is a non-expansion in $\|\cdot\|_\infty$.

Proof.

$$\Pi_{\mathcal{F}_\phi,\nu} J \in \operatorname*{argmin}_{\widehat{J}\in\mathcal{F}_\phi} \mathbb{E}_{s\sim\nu}\left[\left(\widehat{J}(s) - J(s)\right)^2\right]$$

It is solved by the conditional mean

$$\widehat{J}(i) = \mathbb{E}_{s\sim\nu}\left[J(s) \mid s \in \mathcal{S}_i\right]$$

$\square$

Another fact: $\Pi_{\mathcal{F}_\phi,\nu}$ is linear.

Convergence proof

We show the projected Bellman update is a max-norm contraction. The convergence result follows immediately.

<u>Lemma</u> $\Pi_{\mathcal{F}_\phi,\nu} T$ is a contraction w.r.t $\|\cdot\|_\infty$ with modulus of contraction α.

Proof.

$$\begin{aligned}
\|\Pi_{\mathcal{F}_\phi,\nu} TJ - \Pi_{\mathcal{F}_\phi,\nu} T\bar{J}\|_\infty &= \|\Pi_{\mathcal{F}_\phi,\nu} \left(TJ - T\bar{J} \right) \|_\infty \\
&\leq \|TJ - T\bar{J}\|_\infty \\
&\leq \alpha \|J - \bar{J}\|_\infty
\end{aligned}$$

$\square$

Approximation error bound

Is $\hat{J}$ an effective approximation to J^*? We compare its quality to the best possible approximation possible with state-aggregation.

Theorem
Set $\epsilon = \|\Pi_{\mathcal{F}_\phi,\nu} J^* - J^*\|_\infty$. *Then*

$$\|\widehat{J} - J^*\|_\infty \leq \frac{\epsilon}{1-\alpha}$$

Proof.

$$\begin{aligned}\|\widehat{J} - J^*\|_\infty &\leq \|\widehat{J} - \Pi_{\mathcal{F}_\phi,\nu} J^*\|_\infty + \|\Pi_{\mathcal{F}_\phi,\nu} J^* - J^*\|_\infty \\ &= \|\Pi_{\mathcal{F}_\phi,\nu} T\widehat{J} - \Pi_{\mathcal{F}_\phi,\nu} TJ^*\|_\infty + \epsilon \\ &\leq \alpha\|\widehat{J} - J^*\|_\infty + \epsilon.\end{aligned}$$

□

Comment: We can't directly compute the projection of J^*, since we don't know it. We instead solve a projected version of Bellman's equation, which leads the error bound to expand by $1/(1-\alpha)$.

Performance loss bound

How effective is the greedy policy computed with respect to $\widehat{J}$?

Theorem

Let $\mu \in G(\widehat{J})$ *and* $\epsilon = \|\Pi_{\mathcal{F}_\phi,\nu} J^* - J^*\|_\infty$. *Then,*

$$\|J_\mu - J^*\|_\infty \leq \frac{\alpha \|\widehat{J} - J\|_\infty}{1-\alpha} \leq \frac{\alpha \epsilon}{(1-\alpha)^2}$$

Proof.
The first inequality was shown last class. $\square$

Tightness of the Approximation Error Bound

The dependence on $1/(1-\alpha)^2$ is highly problematic.

Unfortunately, it is tight due to an example discussed in detail in Van Roy (2006).

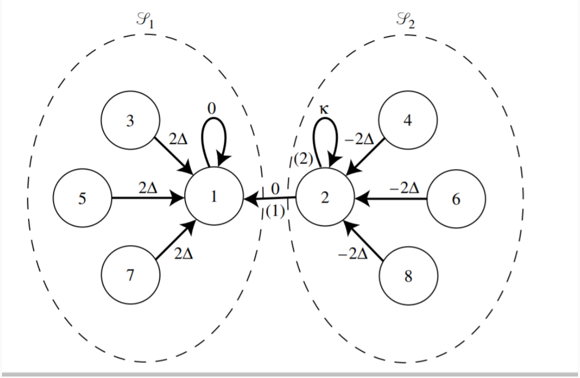


Table of Contents

Reducing performance loss via state-relevance weighting

Recall the discounted state occupancy measure

$$d_\infty^\mu = (1-\alpha)\sum_{t=0}^{\infty} \alpha^t d_0 P_\mu^t.$$

We will show a natural online algorithm converges toward a fixed point

$$\widehat{J} = \Pi_{\mathcal{F}_\phi, d_\infty^\mu} T\widehat{J} \qquad \mu \in G(\widehat{J}).$$

This satisfies the performance loss bound

$$\mathbb{E}_{s\sim d_0}\left[J_\mu(s) - J^*(s)\right] \le \frac{\epsilon}{1-\alpha}$$

Comments:

- Saves a factor of $1/(1-\alpha)$ in the worst-case.
- The state-relevance-weighting is the fraction of time spent in a given state under the selected policy.
- This is a fixed point in $\{$the space of cost-to-functions$\} \times \{$the space of policies$\}$.

Table of Contents

Real time value iteration

Real Time Value Iteration

- Input J_0
- For episode $k = 0, 1, \cdots$
  1. Select greedy policy $\mu_k \in G(J_k)$
  2. Draw random initial state $s_0^{(k)} \sim d_0$.
  3. Sample episode length $\tau \sim \mathrm{Geom}(1-\alpha)$
  4. Sample $(s_0^{(k)}, \cdots, s_\tau^{(k)})$ by applying μ_k for τ timesteps.
  5. Make Bellman update at $s_\tau^{(k)}$ (…Or at all visited states)

$$J_{k+1}(s) \leftarrow \begin{cases} TJ_k(s) & \text{if } s = s_\tau^{(k)} \\ J_k(s) & \text{if } s \neq s_\tau^{(k)} \end{cases}$$

Real time value iteration with Q functions

- Closer to what is used in RL, since no knowledge of the environment is required to compute a greedy policy w.r.t Q.

Real Time Value Iteration with Q functions

- Input Q_0
- For episode $k = 0, 1, \cdots$
  1. Select greedy policy $\mu_k \in G(Q_k)$
  2. Draw random initial state $s_0^{(k)} \sim d_0$.
  3. Sample episode length $\tau \sim \mathrm{Geom}(1-\alpha)$
  4. Sample $(s_0^{(k)}, \cdots, s_\tau^{(k)})$ by applying μ_k for τ timesteps.
  5. Make Bellman update at $s_\tau^{(k)}$ (...Or at all visited states)

$$Q_{k+1}(s,u) \leftarrow \begin{cases} FQ_k(s,u) & \text{if } s = s_\tau^{(k)}, u = \mu_k(s_\tau^{(k)}) \\ Q_k(s,u) & \text{if otherwise} \end{cases}$$

Q-learning

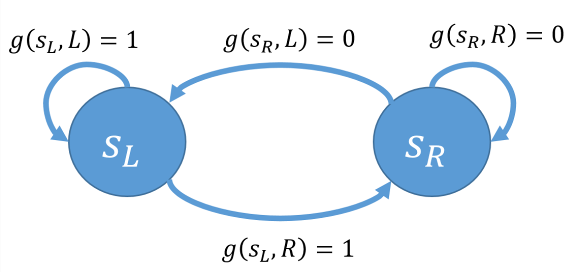
Dropping the k superscripts for ease of notation

Q-learning with greedy exploration

- Input initial Q
- For episode $k = 0, 1, \cdots$
  1. Select greedy policy $\mu \in G(Q)$
  2. Draw random initial state $s_0 \sim d_0$.
  3. Sample episode length $\tau \sim \mathrm{Geom}(1-\alpha)$
  4. Sample $(s_0, \cdots, s_\tau, s_{\tau+1})$ by applying μ for τ timesteps.
  5. Define $u_t = \mu(s_t)$ for $t = 0, \cdots, \tau + 1$.
  6. Make soft noisy Bellman update at s_τ (...Or $s_0 \cdots s_\tau$)

$$Q(s_\tau, \mu(s_\tau)) \leftarrow (1-\beta_k) Q(s_\tau, \mu(s_\tau)) + \beta_k \underbrace{\left[g(s_\tau, u_\tau) + \gamma \min_u Q(s_{\tau+1}, u) \right.}_{\text{Unbiased obsrvation of } FQ(s_\tau, u_\tau)}$$

Convergence of Real-time Value Iteration?



- Suppose the initial state is always L ($d_0 = (1, 0)$).
- Suppose $J = (1, 2)/(1 - \alpha)$,
  - Greedy policy $\mu \in G(J)$ satisfies $\mu(s_L) = L$.
  - The system always stays in state L
  - Also $TJ(s_L) = J(s_L)$.
- Suppose $J(s_R) \leq 0$. [. . . *Optimism in the face of uncertainty*]
  - If $G(J) = (R, R)$ then the induced policy is optimal.
  - Otherwise, we must have $J(s_L) < J^*(s_L)$.
    Real-time VI increases the estimate $J(s_L)$.

Table of Contents

Error bounds depend on the state distribution

Notation: Define $J(d) = \sum_{s \in \mathcal{S}} d(s) J(s)$.

Lemma from last class. For any $J \in \mathbb{R}^n$ and policy μ',

$$J - J_{\mu'} = (I - \alpha P_{\mu'})^{-1}(J - T_{\mu'} J)$$

$$J(d_0) - J_{\mu'}(d_0) = \frac{1}{1-\alpha}(J - T_{\mu'} J)(d_\infty^{\mu'})$$

Performance loss bounds that depend on the state distribution

Lemma: For any $J \in \mathbb{R}^n, \mu \in G(J)$ and optimal policy μ^*,

$$
\begin{aligned}
0 \preceq \quad & J_\mu - J^* \preceq \left[(I - \alpha P_{\mu^*})^{-1} - (I - \alpha P_\mu)^{-1}\right](J - TJ) \\
0 \preceq \quad & J_\mu(d_0) - J^*(d_0) \preceq (J - TJ)(d_\infty^{\mu^*}) - (J - TJ)(d_\infty^{\mu})
\end{aligned}
$$

Proof.
Applying the earlier Lemma with $\mu' = \mu^*$ gives

$$J - J^* = (I - \alpha P_{\mu^*})^{-1}(J - T_{\mu^*}J) \preceq (I - \alpha P_{\mu^*})^{-1}(J - TJ)$$

Since $\mu \in G(J)$, we also have

$$J - J_\mu = (I - \alpha P_\mu)^{-1}(J - T_\mu J) = (I - \alpha P_\mu)^{-1}(J - TJ)$$

Subtracting yields the result. $\square$

Comment: For our current purposes, the dependence on μ^* is problematic, as it's unknown.

Optimism to the rescue

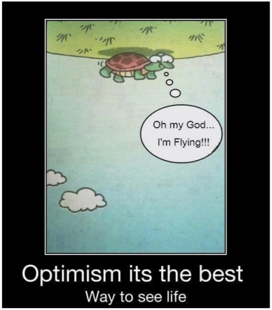
Performance loss is bounded by *on policy* Bellman errors.

Lemma: If $J \preceq J^*$, $\mu \in G(J)$,

$$0 \preceq J_\mu - J^* \preceq (I - \alpha P_\mu)^{-1}(TJ - J)$$

$$0 \preceq J_\mu(d_0) - J^*(d_0) \preceq (TJ - J)(d_\infty^\mu)$$

Proof.
$J_\mu - J^* \preceq J_\mu - J = (I - \alpha P_\mu)^{-1}(TJ - J)$. $\square$



Regret Style Analysis of RTVI

Assumption: Optimistic initiation $J_0 = \frac{-M}{1-\alpha}$. (Recall $M = \|g\|_\infty$)

Keys to analysis:

- **On policy sampling**: The state $s_\tau^{(k)}$ is sampled from $d_\infty^{\mu_k}$.

$$\begin{aligned}\mathbb{P}(s_\tau^{(k)} = s) = \mathbb{E}\left[\mathbb{P}(s_\tau^{(k)} = s \mid \tau)\right] &= \mathbb{E}\left[(d_0 P_{\mu_k}^\tau)(s)\right] \\ &= (1-\alpha)\sum_{t=0}^{\infty} \alpha^t (d_0 P_{\mu_k}^t(s)).\end{aligned}$$

- **Monotonicity**: Since $J_0 \preceq TJ_0$, we have

$$J_0 \preceq TJ_0 \preceq T^2 J_0 \preceq \cdots \preceq J^*$$

and one can similarly show the iterates of RTVI are monotone:

$$J_0 \preceq J_1 \preceq J_2 \cdots$$

Regret Style Analysis of RTVI

Set $\mathrm{Sum}(J) = \sum_{s \in \mathcal{S}} J(s)$. By our earlier Lemma,

$$
\begin{aligned}
(1-\alpha)(J_{\mu_0}(d_0) - J^*(d_0)) \le (TJ_0 - J_0)(d_\infty^{\mu_0}) &= \mathbb{E}\left[(TJ_0 - J_0)(s_\tau^{(0)})\right] \\
&= \mathbb{E}\left[(J_1 - J_0)(s_\tau^{(0)})\right] \\
&= \mathbb{E}\left[\mathrm{Sum}(J_1) - \mathrm{Sum}(J_0)\right]
\end{aligned}
$$

Applying this for each episode k, and using the tower property:

$$
\begin{aligned}
\mathrm{Regret(K)} :&= \mathbb{E}\left[\sum_{k=1}^{K} J_k(d_0) - J^*(d_0)\right] \\
&\le (1-\alpha)^{-1}\mathbb{E}\left[\sum_{k=0}^{K} (\mathrm{Sum}(J_k) - \mathrm{Sum}(J_{K+1}))\right] \\
&= (1-\alpha)^{-1}\mathrm{Sum}(J_0) - \mathrm{Sum}(J_{k+1}) \\
&\le (1-\alpha)^{-1}\mathrm{Sum}(J_0 - J^*) \\
&\le \frac{2M|\mathcal{S}|}{(1-\alpha)^2}.
\end{aligned}
$$

Sketch of asymptotic analysis of RTVI

We converge to a cost-to-go function satisfying Bellman's equation at all states visited with positive probability.
Optimism implies optimality.

- Since $\{J_k\}$ is a monotone sequence, it has a limit $J_\infty \preceq J^*$.
- Let $\mu_\infty \in G(J_\infty)$ be the corresponding policy, and assume it's unique so $\mu_k = \mu_\infty$ for sufficiently large k.
- Since the limit exists, we must have

$$\|J_k - J_{k+1}\|_\infty \to 0$$

- This implies

$$J_k(s) - TJ_k(s) \to 0$$

  for s visited with positive probability ($d_\infty^{\mu_\infty}(s) > 0$). So

$$J_{\mu_\infty}(d_0) - J^*(d_0) \leq \left(TJ_\infty - J_\infty\right)\left(d_\infty^{\mu_\infty}\right) = 0.$$

Table of Contents

Which algorithm attains this performance bound?

We use require $\epsilon = \sup_s \|J^*(\phi(s)) - J^*(s)\|$.

Real Time Value Iteration with State Aggregation

- Input J_0 with $J_0(s) = \frac{-M}{1-\alpha}$ $\forall s$. (…Recall $M = \|g\|_\infty$)
- For $k = 0, 1, \cdots$
  1. Sample $s_k \sim d_\infty^{\mu_k}$ where $\mu_k \in G(J_k)$
  2. Make a conservative update to the representative state $\phi(s_k)$:

$$J_{k+1}(s) \leftarrow \begin{cases} TJ_k(s_k) - \epsilon & \text{if } \phi(s) = \phi(s_k) \\ J_k(s) & \text{if } \phi(s) \neq \phi(s_k) \end{cases}$$

Comments:

- Step 1 can be executed by simulating a state trajectory under a greedy policy with respect to J_k, as shown before.
- An efficient implementation only needs to store the value at the representative state.
- With Q-function variants it is easier to apply a greedy policy.

Regret style analysis of RTVI with state-aggregation

Notation: Representative states: $\Phi = \{\phi(s) : s \in \mathcal{S}\}$
$\mathrm{Sum}(J) := \sum_{s \in \Phi} J(s)$.

The next result establishes optimism for the iterates J_k of RTVI.
Lemma (Optimism): $J_k \preceq J^*$ for each k.

Proof.
We have $J_0 \preceq J^*$ by definition.
For every s with $\phi(s) = \phi(s_0)$ (...the states we update ...),

$$J_1(s) = TJ_0(s_0) - 2\epsilon \leq TJ^*(s_0) - 2\epsilon \leq J^*(s)$$

where the inequality used that

$$TJ^*(s) = J^*(s) \leq J^*(s_0) + |J^*(s) - J^*(\phi(s_k)) + |J^*(s_0) - J^*(\phi(s_k))|.$$

$\square$

Regret style analysis of RTVI with state-aggregation (continued)

As before, we have the bound in each episode:

$$\begin{aligned}(1-\alpha)\left[J_{\mu_0}(d_0) - J^*(d_0)\right] &\leq (TJ_0 - J_0)\,(d_\infty^{\mu_0}) \\ &= \mathbb{E}\left[(TJ_0 - J_0)\,(s_0)\right] \\ &\overset{(*)}{=} \mathbb{E}\left[(J_1 - J_0)\,(\phi(s_0))\right] + \epsilon \\ &= \mathbb{E}\left[\mathrm{Sum}(J_1) - \mathrm{Sum}(J_0)\right] + \epsilon\end{aligned}$$

The equality (*) uses that by the definition of the algorithm

$$J_1(\phi(s_0)) = TJ_0(s_0) - \epsilon.$$

Regret style analysis sof RTVI (continued)

Applying this for each episode k, and using the tower property:

$$\begin{aligned}
\mathrm{Regret(K)} &:= \mathbb{E}\left[\sum_{k=1}^{K} J_{\mu_k}(d_0) - J^*(d_0)\right] \\
&\leq \frac{1}{1-\alpha}\mathbb{E}\left[\sum_{k=0}^{K} (\mathrm{Sum}(J_{k+1}) - \mathrm{Sum}(J_K))\right] + \frac{K\epsilon}{1-\alpha} \\
&= \frac{\mathrm{Sum}(J_{k+1}) - \mathrm{Sum}(J_0)}{1-\alpha} + \frac{K\epsilon}{1-\alpha} \\
&\leq \frac{\mathrm{Sum}(J^* - J_0)}{1-\alpha} + \frac{K\epsilon}{1-\alpha} \\
&\leq \frac{2M|\Phi|}{(1-\alpha)^2} + \frac{K\epsilon}{1-\alpha}.
\end{aligned}$$

We see that average regret scales as

$$\limsup_{K\to\infty} \frac{\mathrm{Regret(K)}}{K} \leq \frac{\epsilon}{1-\alpha}$$