Reinforcement Learning - Lectures 8-9

Contents

1 Planning in discounted MDPs

1.1 Now planning

We look for an optimal policy ie: $\pi^* : \mathcal{S} \to \mathcal{A}$ such that:

$$V^{\pi^*} \geq V^\pi \quad \forall \pi$$

Note

The existence is not obvious at first!

Definition: Bellman operator T^*

$$T^* : \mathbb{R}^k \to \mathbb{R}^k$$
$$\forall V \in \mathbb{R}^k, \quad (T^*V)(s) = \max_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} k(s,a,s')[r(s,a,s') + \gamma V(s')]$$

Proposition:

T^* is isotonic and a γ-contractant.
γ-contractant: $|\max(f) - \max(g)| \leq \max(|f-g|)$
Hence:

$$\begin{aligned}
&(T^*V_1)(s) - (T^*V_2)(s)\\
&= \left| \max_a \sum_{s'} k(s,a,s')[r(s,a,s') + \gamma V_1(s')] - \max_a \sum_{s'} k(s,a,s')[r(s,a,s') + \gamma V_2(s')] \right|\\
&\leq \max_a \left(\sum_{s'} k(s,a,s') \gamma (V_1(s') - V_2(s')) \right)\\
&\leq \max_a \left(\sum_{s'} k(s,a,s') \gamma \max_{s'} |V_1(s') - V_2(s')| \right)\\
&\leq \gamma ||V_1 - V_2||_\infty \max_a \left(\sum_{s'} k(s,a,s') \right)\\
&\Rightarrow ||T^*V_1 - T^*V_2||_\infty \leq \gamma ||V_1 - V_2||_\infty
\end{aligned}$$

T^* isotonic: exercise

Consequences: T^* has a unique fixed point V^* ie: $\forall V_0 \in \mathbb{R}^k, (T^*)^n V_0 \to V^*$ when $n \to +\infty$

Theorem (Bellman optimality theorem):

V^* is the optimal value function ie: $V^*(s) = \max_\pi V^\pi(s)$. A policy such that $T^\pi V^* = V^*$ is an optimal policy.

Definition: Greedy Policy

For every $V \in \mathbb{R}^k$, there exists at least one policy π such that $T^\pi V = T^* V$.
This policy is called a greedy policy with respect to V and is characterized by:

$$\forall s \in \mathcal{S}, \quad \pi(s) \in \arg\max_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} k(s,a,s')[r(s,a,s') + \gamma V(s')]$$

Theorem (Policy improvement lemma):

For any policy π, any greedy policy π' with respect to V^π improves on π ie: $V^{\pi'} \geq V^\pi$
Proof: $T^{\pi'}V^\pi = T^*V^\pi \geq T^\pi V^\pi = V^\pi$ (by definition of greedy policy and isotonicity of T^π)
$\Rightarrow (T^{\pi'})^2 V^\pi \geq T^{\pi'}V^\pi \geq V^\pi \Rightarrow \cdots \Rightarrow (T^{\pi'})^n V^\pi \geq V^\pi$
When $n \to +\infty$, $(T^{\pi'})^n V^\pi \to V^{\pi'}$ hence $V^{\pi'} \geq V^\pi$

Warning

Proof of Bellman optimality theorem:
Let π^* be a greedy policy with respect to V^*.

- For any policy π, $T^\pi \leq T^*$ hence: $V^\pi = (T^\pi)^n V^\pi \leq (T^*)^n V^\pi \to V^*$ hence $V^* \geq V^\pi$
- $T^{\pi^*}V^* = T^*V^* = V^*$ hence $V^{\pi^*} = V^*$

Any finite MDP has a deterministic optimal policy.

Algorithm: Value iteration algorithm

Input: $\epsilon > 0$ the precision parameter, $V_0 \in \mathbb{R}^k$ an initial value function
Output: an ϵ-approximation of V^*

1. $V \leftarrow V_0$
2. while $||T^*V - V||_\infty \geq \frac{\epsilon(1-\gamma)}{\gamma}$ do
   - $V \leftarrow T^*V$
3. end while
4. return T^*V

Proof: Let $V_n = (T^*)^n V_0$. We have:

$$\begin{aligned} ||V_n - V^*||_\infty &\leq ||(T^*)V^* - (T^*)V_n||_\infty + ||(T^*)V_n - V_n||_\infty \\ &\leq \gamma||V^* - V_n||_\infty + \gamma||V_n - V_{n-1}||_\infty \end{aligned}$$

because $V_n = (T^*)V_{n-1}$. Hence:

$$||V_n - V^*||_\infty \leq \frac{\gamma}{1-\gamma}||V_n - V_{n-1}||_\infty$$

Now if $||V_n - V_{n-1}||_\infty \leq \frac{\epsilon(1-\gamma)}{\gamma}$ then $||V_n - V^*||_\infty \leq \epsilon$.

Proposition:

The Value iteration algorithm requires at most $\frac{\log\left(\frac{M}{\epsilon(1-\gamma)}\right)}{1-\gamma}$ iterations where $M = ||T^*V_0 - V_0||_\infty$.
Proof:

$$||V_{n+1} - V_n||_\infty = ||(T^*)V_n - (T^*)V_{n-1}||_\infty \leq \gamma||V_n - V_{n-1}||_\infty \leq \cdots \leq \gamma^n ||T^*V_0 - V_0||_\infty$$

Hence if:

$$n \geq \frac{\log\left(\frac{M}{\epsilon(1-\gamma)}\right)}{1-\gamma} \geq \frac{\log\left(\frac{M}{\epsilon(1-\gamma)}\right)}{-\log(\gamma)}$$

then $\gamma^n \leq \frac{\epsilon(1-\gamma)}{M\gamma}$ hence $||V_{n+1} - V_n||_\infty \leq \frac{\epsilon(1-\gamma)}{\gamma}$

Algorithm: Policy iteration algorithm

Input: an initial policy π_0
Output: π^* an optimal policy

1. $\pi \leftarrow \pi_0$
2. $\pi' \leftarrow$ None
3. while $\pi \neq \pi'$ do
   - Compute V^π
   - $\pi' \leftarrow \pi$
   - $\pi \leftarrow$ a greedy policy with respect to V^π
4. end while
5. return π

Properties:

The policy iteration algorithm always returns an optimal policy in at most $|\mathcal{A}|^{|\mathcal{S}|}$ iterations (one can prove that the number of iterations can be bounded by $O\left(\frac{|\mathcal{A}|^{|\mathcal{S}|}}{|\mathcal{S}|}\right)$).

Lemma:

Let (U_n) be the sequence of value functions generated by the Value Iteration algorithm and (V_n) be the sequence of value functions generated by the above Policy Iteration algorithm. If $U_0 = V_0$ then $\forall n, U_n \leq V_n$
Proof by induction:
Assume that $U_n \leq V_n$ then:

$$U_{n+1} = T^*(U_n) \leq T^*(V_n) = T^{\pi_{n+1}}(V_n) \leq T^{\pi_{n+1}}(V_n) = V_{n+1}$$

Algorithm 3: Linear Programming

Let $\alpha : \mathcal{S} \to \mathbb{R}_+$ be a positive weight function over the states.
V^* is the solution of the linear program:

$$\min_{V \in \mathbb{R}^k} \sum_{s \in \mathcal{S}} \alpha(s) V(s)$$

subject to:

$$\forall s \in \mathcal{S}, a \in \mathcal{A}, \quad V(s) \geq \sum_{s' \in \mathcal{S}} k(s,a,s')[r(s,a,s') + \gamma V(s')]$$

Proof:

By Bellman's optimality equation $T^*(V^*) = V^*$ thus V^* satisfies the constraints with equality.
If V satisfies the constraint then let $W = V - V^*$

$$\forall s, a, \quad W(s) \geq \gamma \sum_{s'} k(s, a, s')W(s')$$

$\Rightarrow$ If $s_- \in \arg\min_s W(s)$ then:

$$W(s_-) \geq \gamma \sum_{s'} k(s_-, a, s')W(s') \geq \gamma W(s_-)$$

thus $W(s_-) \geq 0$ and $\forall s, W(s) \geq 0$ thus $V \geq V^*$
Therefore: $\sum_s \alpha(s)V(s) \geq \sum_s \alpha(s)V^*(s)$

2 Learning in finite discounted MDPs

Definition

State-action value function (Q-function):
The state-action value function $Q^\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ of a policy π is the expected return when first taking action a in state s, and then following policy π:

$$\begin{aligned} Q^\pi(s, a) &= \mathbb{E}^\pi \left[\sum_{t=1}^{\infty} \gamma^t R_t \mid S_1 = s, A_1 = a \right] \\ &= \sum_{s' \in \mathcal{S}} k(s, a, s')[r(s, a, s') + \gamma V^\pi(s')] \end{aligned}$$

Remark: $V^\pi(s) = Q^\pi(s, \pi(s))$

Policy improvement Lemma 2:

For any policies π, π':

$$\forall s \in \mathcal{S}, Q^\pi(s, \pi'(s)) \geq Q^\pi(s, \pi(s)) \Rightarrow \forall s \in \mathcal{S}, V^{\pi'}(s) \geq V^\pi(s)$$

Furthermore, if one of the inequalities in the hypothesis is strict, then at least one of the inequalities in the RHS is strict.
Proof: for all $s \in \mathcal{S}$:

$$\begin{aligned} V^\pi(s) &= Q^\pi(s, \pi(s)) \\ &\leq Q^\pi(s, \pi'(s)) \\ &= \sum_{s_1} k(s, \pi'(s), s_1)[r(s, \pi'(s), s_1) + \gamma V^\pi(s_1)] \end{aligned}$$

with $V^\pi(s_1) = Q^\pi(s_1, \pi(s_1)) \leq Q^\pi(s_1, \pi'(s_1))$

$$\begin{aligned}\Rightarrow V^\pi(s) \leq \sum_{s_1} & k(s,\pi'(s),s_1)[r(s,\pi'(s),s_1) \\ & + \gamma[\sum_{s_2} k(s_1,\pi'(s_1),s_2)[r(s_1,\pi'(s_1),s_2) + \gamma V^\pi(s_2)]]] \\ \leq \cdots &= V^{\pi'}(s)\end{aligned}$$

with $V^\pi(s_2) = Q^\pi(s_2, \pi(s_2)) \leq Q^\pi(s_2, \pi'(s_2))$

2.1 Q-table formulation of Bellman optimality equation

A policy π is optimal iff $\forall s \in \mathcal{S}, \pi(s) \in \arg\max_{a\in\mathcal{A}} Q^\pi(s,a)$

Proof: A policy π such that:

$$\pi(s) \in \arg\max_{a\in\mathcal{A}} Q^\pi(s,a) = \arg\max_{a\in\mathcal{A}} \sum_{s'\in\mathcal{S}} k(s,a,s')[r(s,a,s') + \gamma V^\pi(s')]$$

is a greedy policy with respect to V^π and then $T^*V^\pi = T^\pi V^\pi = V^\pi$ hence $V^\pi = V^*$.

If $\exists s_0 \in \mathcal{S}, a \in \mathcal{A}$ such that $Q^\pi(s_0, \pi(s_0)) \leq Q^\pi(s_0, a)$ then by policy improvement lemma 2, the policy π' defined by:

$$\pi'(s) = \begin{cases} \pi(s) & \text{if } s \neq s_0 \\ a & \text{if } s = s_0 \end{cases}$$

is preferable: $V^{\pi'}(s_0) \geq V^\pi(s_0)$

The Q-learning Algorithm

Input:

- Q_0 an initial guess for the Q-table (may be zero)
- s_0 an initial state
- π a learning policy (may be greedy wrt Q)
- T the number of iterations

1. $Q \leftarrow Q_0$
2. $s \leftarrow s_0$
3. for t in $0, 1, \ldots, T-1$ do
    - $a \leftarrow$ Select action $(\pi(Q, s))$
    - $r', s' \leftarrow$ observe reward and transition from state s using action a
    - $Q(s,a) \leftarrow Q(s,a) + \alpha_t(r' + \gamma(\max_{a'} Q(s',a')) - Q(s,a))$
    - $s \leftarrow s'$
4. return Q

Theorem: Convergence of Q-learning

Let $\alpha_t(s,a) = \alpha_t \mathbf{1}_{(s_t,a_t)=(s,a)}$
If for all $s \in \mathcal{S}, a \in \mathcal{A}$ it holds that $\sum_{t=0}^{\infty} \alpha_t(s,a) = \infty$ and $\sum_{t=0}^{\infty} \alpha_t(s,a)^2 < \infty$ then with probability 1 the Q-learning algorithm converges to Q^* the optimal Q-table as $t \to +\infty$.

SARSA algorithm

Input: Same

1. $Q \leftarrow Q_0$
2. $s \leftarrow s_0$
3. $a \leftarrow$ Select action $(\pi(Q,s))$
4. for t in $0, 1, \ldots, T-1$ do
   - $r', s' \leftarrow$ random reward and transition from s using action a
   - $a' \leftarrow$ Select action $(\pi(Q,s'))$
   - $Q(s,a) \leftarrow Q(s,a) + \alpha_t(r(s,a,s') + \gamma Q(s',a') - Q(s,a))$
   - $s \leftarrow s'$
   - $a \leftarrow a'$
5. end for
6. return Q

Note

Distributional policy evaluation for homework: we just need to optimize the expectation criteria for homework.

3 Approximate dynamic programming

When the state space $\mathcal{S}$ is large (or continuous), we have access only to simulation and not the exact dynamics.

We need a representation of the value function by an approximation space.

$V^*, V^\pi : \mathcal{S} \to \mathbb{R}$

We will approximate $\tilde{V^*}, \tilde{V^\pi}$ in a space $\bar{\mathcal{S}}$, a subset of $\mathbb{R}^{\mathcal{S}}$.

Sampling error: $r(s,a,s')$ and $k(s,a,s')$ will possibly not be known exactly. Instead you will have access to a simulation: $r(s,a,s') \sim \hat{r}(s,a,s'), k(s,a,s') \sim \hat{k}(s,a,s')$

- Sample $s \in \mathcal{S}$ with some probability distribution μ
- Given $s \in \mathcal{S}$, and an action $a \in \mathcal{A}$, sample s' according to $k(s,a,s')$

Today we focus on discounted infinite horizon MDPs.

3.1 Performance loss due to Value Function Approximation

$\pi^*(s) \in \arg\max_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} k(s,a,s')[r(s,a,s') + \gamma V^*(s')]$ is the optimal policy.

Now, assume that I only use V instead of V^*:

$\pi(s) \in \arg\max_{a\in\mathcal{A}} \sum_{s'\in\mathcal{S}} k(s,a,s')[r(s,a,s') + \gamma V(s')]$ is a suboptimal policy.
How much do I lose?
When V^π is the value function of policy π then:

Proposition:

$$||V^* - V^\pi||_\infty \leq \frac{2\gamma}{1-\gamma}||V^* - V||_\infty$$

Proof:

$$\begin{aligned}
||V^* - V^\pi||_\infty &\leq ||T^*V^* - T^*V||_\infty + ||T^\pi V - T^\pi V^\pi||_\infty \\
&\leq \gamma||V^* - V||_\infty + \gamma||V - V^\pi||_\infty \\
&= \gamma||V^* - V^\pi||_\infty + \gamma(||V - V^*||_\infty + ||V^* - V^\pi||_\infty) \\
&= 2\gamma||V^* - V||_\infty + \gamma||V^* - V^\pi||_\infty \\
&\leq \frac{2\gamma}{1-\gamma}||V^* - V||_\infty
\end{aligned}$$

(where T^π, T^* are γ-contractant)

Proposition:

There exists $\epsilon > 0$ such that if $||V^* - V||_\infty \leq \epsilon$ then π is optimal and $V^\pi = V^*$.
Proof: Let $\delta = \min_{\pi:V^\pi \neq V^*} ||V^\pi - V^*||_\infty$. We have $\delta > 0$ since the set of policies is finite. Let ϵ such that $\frac{2\gamma}{1-\gamma}\epsilon < \delta$. Then:

$$||V - V^*||_\infty \leq \epsilon \Rightarrow ||V^\pi - V^*||_\infty \leq \delta$$

hence $V^\pi = V^*$

3.2 Bellman residual minimization

Let $\mathcal{F}$ be a subset of $\mathbb{R}^{\mathcal{S}}$, a function space equipped with the norm $||\cdot||$.

For a function $V \in \mathbb{R}^{\mathcal{S}}$, the Bellman residual is defined as: $B(V) = ||V - T^*V||$. The optimal value function V^* satisfies $T^*V^* = V^*$ and hence $B(V^*) = 0$. This motivates to approximate V^* by $\inf_{V\in\mathcal{F}} B(V)$.

Here we first consider the norm $||\cdot||_\infty$ and we will express the suboptimality gap $||V^* - V||_\infty$ as a function of the Bellman Residual $B(V) = ||T^*V - V||_\infty$.

Proposition:

For any function $V \in \mathbb{R}^{\mathcal{S}}$:

1. $||V^* - V||_\infty \leq \frac{1}{1-\gamma}||T^*V - V||_\infty$
2. Let π be the greedy policy with respect to V, then:

$$||V^* - V^\pi||_\infty \leq \frac{2}{1-\gamma}||T^*V - V||_\infty$$

3. Let $V_{BR} \in \arg\min_{V\in\mathcal{F}} ||T^*V - V||_\infty$ then:

$$||T^*V_{BR} - V_{BR}||_\infty \leq (1+\gamma)\inf_{V\in\mathcal{F}} ||V^* - V||_\infty$$

Overall, if π_{BR} is greedy with respect to V_{BR} then:

$$||V^* - V^{\pi_{BR}}||_\infty \leq \frac{2(1+\gamma)}{1-\gamma} \inf_{V \in \mathcal{F}} ||V^* - V||_\infty$$

Proof:

1. $||V^* - V||_\infty \leq ||V^* - T^*V||_\infty + ||T^*V - V||_\infty \leq \gamma||V^* - V||_\infty + ||T^*V - V||_\infty$
   Thus: $(1-\gamma)||V^* - V||_\infty \leq ||T^*V - V||_\infty$

2. $$\begin{aligned} ||V^* - V^\pi||_\infty &\leq ||V^* - V||_\infty + ||V - V^\pi||_\infty \\ &\leq ||V - T^*V||_\infty + ||T^*V - V^\pi||_\infty \\ &\leq ||T^*V - V||_\infty + \gamma||V - V^\pi||_\infty \end{aligned}$$

   Thus: $(1-\gamma)||V^* - V^\pi||_\infty \leq ||T^*V - V||_\infty$
   And using (1) we get: $||V^* - V^\pi||_\infty \leq \frac{2}{1-\gamma}||T^*V - V||_\infty$

3. $||T^*V - V||_\infty \leq ||T^*V - V^*||_\infty + ||V^* - V||_\infty \leq (1+\gamma)||V^* - V||_\infty$
   hence: $||T^*V_{BR} - V_{BR}||_\infty \leq \inf_{V \in \mathcal{F}} ||T^*V - V||_\infty \leq (1+\gamma)||V^* - V||_\infty$

3.3 Minimizing the Bellman residual

Let $\mathcal{F} = \{f_\alpha, \alpha \in \Theta\}$ be a family of functions. We want to find $V_{BR} \in \arg\min_{V \in \mathcal{F}} B(V)$.

Minimizing B over $||\cdot||_\infty$ is computationally hard / impossible.

Even if we choose a distribution μ on $\mathcal{S}$ and minimize in $||\cdot||_{\mu,2}$ norm can be hard:

$$\alpha \to B(\alpha) = ||T^*V_\alpha - V_\alpha||^2_{\mu,2} = \int_{\mathcal{S}} (T^*V_\alpha(s) - V_\alpha(s))^2 d\mu(s)$$

If it is the case we resort to stochastic gradient descent: $\alpha \leftarrow \alpha - \eta \nabla_\alpha B(\alpha)$.

- We draw n states at random according to the distribution μ: $s_1, \ldots, s_n \sim \mu$
- We define the empirical Bellman residual for parameter α: $\hat{B}(\alpha) = \frac{1}{n}\sum_{i=1}^n (T^*V_\alpha(s_i) - V_\alpha(s_i))^2$
- We perform a gradient descent with:

$$\nabla_\alpha \hat{B}(\alpha) = \frac{2}{n}\sum_{i=1}^n (T^*V_\alpha(s_i) - V_\alpha(s_i))(\gamma P^{\pi_\alpha} - I)\nabla V_\alpha(s_i)$$

  where π_α is a greedy policy with respect to V_α and P^{π_α} is the transition matrix under policy π_α

3.4 Approximate Value Iteration (AVI)

VI is defined by: $V_{k+1} = T^*V_k$.

AVI is defined by: $V_{k+1} = \mathcal{A}(T^*V_k)$ where $\mathcal{A}$ is an approximation operator, typically a projection on a subspace $\mathcal{F}$ of $\mathbb{R}^{\mathcal{S}}$. Then:

$$V_{k+1} = \arg\min_{V \in \mathcal{F}} ||T^*V_k - V||_\infty$$

Proposition:

In that case, if $\mathcal{A} = \Pi_\infty$ is the projection operator in $||\cdot||_\infty$ then: $\mathcal{A}T^*$ is still a contraction and the iteration V_k of AVI converge.

Warning

Problem:
Impractical in general since the projection Π_∞ is hard to compute.

Proposition:

Let V^K be the K-th iterate of AVI and π^K be the corresponding greedy policy. Then:

$$||V^* - V^{\pi^K}||_\infty \leq \frac{2\gamma}{(1-\gamma)^2} \max_{k \leq K} ||T^*V_k - \mathcal{A}T^*V_k||_\infty + \frac{2\gamma^{K+1}}{(1-\gamma)}||V^* - V_0||_\infty$$

Proof: Let $\epsilon = \max_{k \leq K} ||T^*V_k - \mathcal{A}T^*V_k||_\infty$. We have:

$$\begin{aligned} ||V^* - V_{k+1}||_\infty &\leq ||T^*V^* - T^*V_k||_\infty + ||T^*V_k - V_{k+1}||_\infty \\ &\leq \gamma||V^* - V_k||_\infty + \epsilon \end{aligned}$$

And:

$$||V^* - V_k||_\infty \leq (1 + \gamma + \cdots + \gamma^{k-1})\epsilon + \gamma^k||V^* - V_0||_\infty \leq \frac{\epsilon}{1-\gamma} + \gamma^k||V^* - V_0||_\infty$$

From Proposition 1: $||V^* - V^{\pi^K}||_\infty \leq \frac{2}{1-\gamma}||V^* - V_K||_\infty$
hence:

$$||V^* - V^{\pi^K}||_\infty \leq \frac{2\gamma}{(1-\gamma)^2}\epsilon + \frac{2\gamma^{K+1}}{(1-\gamma)}||V^* - V_0||_\infty$$

3.5 Implementation of fitted Q-iterations

We assume that we have the generative model mentioned above:

- a sampler from distribution μ over $\mathcal{S}$
- a transition sampler: $S_{t+1} \mid S_t, A_t$

Then $Q^*(s,a) = \sum_{s'} k(s,a,s')[r(s,a,s') + \gamma V^*(s')]$ is the unique fixed point of $T^* : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$:

$$T^*Q(s,a) = \sum_{s'} k(s,a,s')[r(s,a,s') + \gamma \max_b Q(s',b)]$$

So the fitted Q-iteration algorithm can be written: $Q_{k+1} = \mathcal{A}(T^*Q_k)$
Let $\mathcal{F}$ be a vector space over $\mathcal{S} \times \mathcal{A}$ defined by a set of features $\phi_1, \ldots, \phi_d : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$.

$$\mathcal{F} = \left\{ Q_\alpha(s,a) = \sum_{j=1}^d \alpha_j \phi_j(s,a), \alpha \in \mathbb{R}^d \right\}$$

μ is a probability distribution over $\mathcal{S}$

$$Q_{k+1} = \arg\min_{Q \in \mathcal{F}} ||T^* Q_k - Q||^2_{\mu,2}$$

Algorithm: Fitted Q-Iteration

1. Start with $Q_0 : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ for example $= 0$
2. for $k = 1, \dots, K$ do
   - Sample $s_1, \dots, s_n \sim \mu$ and $a_1, \dots, a_n \sim$ uniform over $\mathcal{A}$
   - Use the generative transitions to get $R_1, \dots, R_n$ and $s'_1, \dots, s'_n$ (rewards and next states associated to (s_i, a_i))
   - Compute an estimation $T^* Q_k(s_i, a_i)$ as $Z_i = R_i + \gamma \max_{a \in \mathcal{A}} Q_k(s'_i, a)$
   - Compute Q_{k+1} by solving:

$$Q_{k+1} = \arg\min_{Q_\alpha \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} (Q_\alpha(s_i, a_i) - Z_i)^2$$

Since $\mathcal{F}$ is linear, this is a classical least square minimization problem.

Note

$Q(s, a) = \mathbb{E}[r(s, a, S') + \gamma \max_b Q(s', b)]$
If $Q = Q_\theta$ then:

$$Q_\theta(s, a) = \mathbb{E}[r(s, a, S') + \gamma \max_b Q_\theta(s', b)]$$

We can use dynamics for many pairs:

- $S_i \sim \mu$
- $A_i \sim$ uniform
- Sample S'_i as next state, R_i as reward $(r(S_i, A_i, S'_i))$

$$l(\theta) = \sum_{i=1}^{n} (Q_\theta(S_i, A_i) - (R_i + \gamma \max_b Q_\theta(S'_i, b)))^2$$

Is the general iterating scheme:

- Start from $Q_0(s, a) = 0$
- $Q_{k+1} = Q_{\theta_{k+1}}$ where $\theta_{k+1} = \arg\min_\theta l(\theta)$

Parametric form: $Q_\theta(s, a) = \langle \phi(s, a), \theta \rangle$
Where $\phi(s, a) : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^d$ is a feature vector.

3.6 Non-parametric regressors

K-nearest neighbors:

- Sample $s_1, \dots, s_n \sim \mu$
- For any $a \in \mathcal{A}$, sample $S'_{i,a}, R_{i,a}$ next state and reward according to the dynamics

- Start with $Q_0(s,a) = 0$
- For every i:

$$Q_{k+1}(s_i, a_i) = R_{i,a_i} + \gamma \max_b Q_k(S'_{i,a_i}, b)$$

Where $Q_k(s,b) = \frac{1}{K}\sum_{j=1}^{K} Q_k(s_{I_j}, b)$ where $d(s, s_{I_1(s)}) < \cdots < d(s, s_{I_K(s)})$ are the K nearest neighbors of s in the training set $s_1, \ldots, s_n$.

3.7 Neural Network approximation

We can also try to approximate the Q-function with a Neural Network: $Q_\theta(s,a)$ where θ are the weights of the NN.

Algorithm: Deep Q-Learning

1. epochs = 1000, θ_0 = random
2. for k in range(epochs):
   - Sample (S_i, A_i, S'_i) $(i = 1, \ldots, n)$
   - $l_\theta = \sum_i [Q_\theta(S_i, A_i) - (R_i + \gamma \max_b Q_{\theta_k}(S'_i, b))]^2$
   - $\theta_{k+1} = \theta_k - \eta \nabla_\theta l(\theta_k)$