A Theorical Analysis of Fitted Q-Iteration

Jacob Harder
University of Copenhagen

January 22, 2020

1 Introduction

1.1 Reinforcement Learning

In Reinforcement Learning (RL) we are concerned with finding an optimal policy for an agent in some environment. Typically (also in the case of Q-learning) this environment is a Markov decision process

Definition 1. A Markov decision process (MDP) $(\mathcal{S}, \mathcal{A}, P, R, \gamma)$ consists of

- $\mathcal{S}$ a set of states
- $\mathcal{A}$ a set of actions
- $P : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{S})$ its Markov transition kernel
- $R : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathbb{R})$ its immediate reward distribution
- $\gamma \in (0, 1)$ the discount factor

A policy (for an MDP) is a function

$$\pi : \mathcal{S} \to \mathcal{P}(\mathcal{A})$$

With this we can define the state-value function $V^\pi : \mathcal{S} \to \mathbb{R}$

$$V^\pi(s) = \mathbb{E}\left(\sum_{t\geq 0} \gamma^t R_t \mid R_t \sim R(S_t, A_t), S_t \sim P(S_{t-1}, A_{t-1}), A_t \sim \pi(S_t), S_0 = s\right)$$

And the state-action-value (Q-) function $Q^\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$

$$Q^\pi(s, a) = \mathbb{E}(R(s, a) + \gamma V^\pi(S_0) \mid S_0 \sim P(s, a))$$

The optimal Q-function is defined as

$$Q^*(s, a) = \sup_\pi Q^\pi(s, a)$$

One can show that there is a policy π^* such that $Q^* = Q^{\pi^*}$. This is the optimal policy - the goal of RL.

Note that V^π, Q^π and Q^* are usually infeasible to calculate to machine precision, unless $\mathcal{S} \times \mathcal{A}$ is finite and not very big.

1.2 Q-Learning

Let $\pi : \mathcal{S} \to \mathcal{P}(\mathcal{A})$ be a policy. We define the operator

$$(P^\pi Q)(s,a) = \mathbb{E}(Q(S',A') \mid S' \sim P(s,a), A' \sim \pi(S'))$$

Intuitively this operator yields the expected state-action-value function when looking *one step ahead* following the policy π and taking expectation over Q.

We define the operator T^π called the Bellman operator by

$$(T^\pi Q)(s,a) = \mathbb{E}R(s,a) + \gamma(P^\pi Q)(s,a)$$

This operator adjust the Q function to look more like Q^π making one "iteration" of "propagation of rewards" discounting with γ. Indeed it is easily seen that Q^π is a fixed point for T^π.

The *greedy* policy π with respect to a state-action value function Q is the one for which $\pi(s,a) = 1$ when $a = \operatorname{argmax}_a Q(s,a)$ and 0 otherwise.

$$(TQ)(s,a) = T^{\pi_Q} Q$$

called the Bellman optimality operator.

The Bellman optimality equation is says that $Q^* = TQ^*$.

1.3 Artificial Neural Networks

Definition 2. An **ANN** (Artificial Neural Network) with structure $\{d_i\}_{i=0}^{L+1} \subseteq \mathbb{N}$, activation functions $\sigma_i = (\sigma_{ij} : \mathbb{R} \to \mathbb{R})_{j=1}^{d_i}$ and weights $\{W_i \in M^{d_i \times d_{i-1}}, v_i \in \mathbb{R}^{d_i}\}_{i=1}^{L+1}$ is the function $F : \mathbb{R}^{d_0} \to \mathbb{R}^{d_{L+1}}$

$$F(x) = w_{L+1} \circ \sigma_L \circ w_L \circ \sigma_{L-1} \circ \cdots \circ w_1 x$$

where w_i is the affine function $x \mapsto W_i x + v_i$ for all i.

Here $\sigma_i(x_1, \ldots, x_{d_i}) = (\sigma_{i1}(x_1), \ldots, \sigma_{id_i}(x_{d_i}))$.

$L \in \mathbb{N}_0$ is called the number of hidden layers.

d_i is the number of neurons or nodes in layer i.

An ANN is called *deep* if there are two or more hidden layers.

1.4 Fitted Q-Iteration

We here present the algorithm which everything in this paper revolves around:

Algorithm 1: Fitted Q-Iteration Algorithm

Input: MDP $(\mathcal{S}, \mathcal{A}, P, R, \gamma)$, function class $\mathcal{F}$, sampling distribution ν, number of iterations K, number of samples n, initial estimator $\widetilde{Q}_0$

for $k = 0, 1, 2, \ldots, K-1$ **do**

Sample i.i.d. observations $\{(S_i, A_i), i \in [n]\}$ from ν obtain $R_i \sim R(S_i, A_i)$ and $S_i' \sim P(S_i, A_i)$

Let $Y_i = R_i + \gamma \cdot \max_{a \in \mathcal{A}} \widetilde{Q}_k(S_i', a)$

Update action-value function:

$$\widetilde{Q}_{k+1} \leftarrow \operatorname*{argmin}_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} (Y_i - f(S_i, A_i))^2$$

Define π_K as the greedy policy w.r.t. $\widetilde{Q}_K$

Output: An estimator $\widetilde{Q}_K$ of Q^* and policy π_K

1.5 Assumption 1: Holder Smoothness

Definition 3. Let $\mathcal{D} \subseteq \mathbb{R}^r$ be compact and $\beta, H > 0$. A function $f : \mathcal{D} \to \mathbb{R}$ we call Holder smooth if

$$\sum_{\alpha : |\alpha| < \beta} \|\partial^\alpha f\|_\infty + \sum_{\alpha : \|\alpha\|_1 = \lfloor \beta \rfloor} \sup_{x \neq y} \frac{|\partial^\alpha (f(x) - f(y))|}{\|x - y\|_\infty^{\beta - \lfloor \beta \rfloor}} \leq H$$

Where $\alpha = (\alpha_1, \ldots, \alpha_r) \in \mathbb{N}^r$. We write $f \in C_r(\mathcal{D}, \beta, H)$.

Definition 4. We consider families of *Compositions of Holder Functions*

$$\mathcal{G}(\{p_j, t_j, \beta_j, H_j\}_{j \in [q]})$$

where $t_j, p_j \in \mathbb{N}$, $t_j \leq p_j$ and $H_j, \beta_j > 0$, defined as containing f when $f = g_q \circ \cdots \circ g_1$ for $g_j : [a_j, b_j]^{p_j} \to [a_{j+1}, b_{j+1}]^{p_{j+1}}$ functions on some real hypercubes that only depend on t_j of their inputs for each of their components g_{jk}, and satisfies $g_{jk} \in C_{t_j}([a_j, b_j]^t_j, \beta_j, H_j)$.

Assumption 1. Let

$$\mathcal{G}_0 = \{f : \mathcal{S} \times \mathcal{A} \to \mathbb{R} : f(\cdot, a) \in \mathcal{G}(\{p_j, t_j, \beta_j, H_j\}_{j \in [q]}), \forall a \in \mathcal{A}\}$$

It is assumed that $Tf \in \mathcal{G}_0$ for any $f \in \mathcal{F}_0$.

I.e. when using the Bellman optimality operator on our sparse ReLU networks, we should stay in the class of compositions of Holder smooth functions.

1.6 Assumption 2: Concentration Coefficients

Assumption 2. Let $\nu_1, \nu_2 \in \mathcal{P}(\mathcal{S} \times \mathcal{A})$ be probability measures, Lebesgue-absolutely continuous in $\mathcal{S}$ Define

$$\kappa(m, \nu_1, \nu_2) = \sup_{\pi_1, \dots, \pi_m} \left[\mathbb{E}_{v_2} \left(\frac{\mathrm{d}(P^{\pi_m} \dots P^{\pi_1} \nu_1)}{\mathrm{d}\nu_2} \right)^2 \right]^{1/2}$$

Let ν be the sampling distribution from the algorithm, and mu the distribution over which we measure the error in the main theorem, then we assume

$$(1-\gamma)^2 \sum_{m \geq 1} \gamma^{m-1} m \kappa(m, \mu, \nu) = \phi_{\mu,\nu} < \infty$$

1.7 The Main Theorem

Theorem 1 (Yang, Xie, Wang)**.** For any $K \in \mathbb{N}$ let Q^{π_K} be the action-value function corresponding to policy π_K which is return by Algorithm 1, when run with a sparse ReLU network on the form

$$\mathcal{F}_0 = \{ f(\cdot, a) \in \mathcal{F}(L^*, \{d_j^*\}_{j=0}^{L^*+1}, s^*) \mid a \in \mathcal{A} \}$$

where

$$L^* \lesssim (\log n)^{\xi'}, d_0 = r, d_j^*, d_{L+1} = 1, \lesssim n^{\xi'}, s^* \asymp n^{\alpha^*} \cdot (\log n)^{\xi'}$$

Let μ be any distribution over $\mathcal{S} \times \mathcal{A}$. Under assumptions assumption 1 and **??** $\|Q^* - Q^{\pi_K}\|_{1,\mu} \leq C \cdot \frac{\phi_{\mu,\nu} \cdot \gamma}{(1-\gamma)^2} \cdot |\mathcal{A}| \cdot (\log n)^{\xi^*} \cdot n^{(\alpha^*-1)/2} + \frac{4\gamma^{K+1}}{(1-\gamma)^2} \cdot R_{\max}$
Here $\alpha^* \in (0,1), C, \xi', \xi^*, \phi_{\mu,\nu} \in \mathbb{R}_+$ are constants depending on the assumptions and $R_{\max}$ the maximum possible reward.

2 Disambiguation

- $I_m = [0,1]^m$.
- $C(X) = \{f : X \to \mathbb{R} \mid f \text{continuous}\}$.
- $C_{\mathbb{C}}(X) = \{f : X \to \mathbb{C} \mid f \text{continuous}\}$.
- ANN: artificial neural network see definition 2.

3 Universal Approximator Theorem

Theorem 2 (Universal Approximation Theorem for ANNs)**.** Let $\sigma : \mathbb{R} \to \mathbb{R}$ be non-constant, bounded and continuous function. Let $\varepsilon > 0$ and $f \in C(I_m)$. Then there exists a 1-layer ANN F with activation function σ such that

$$\|F - f\|_\infty < \varepsilon$$

4 Main theorem