A Note on Double Q-Learning

WeiChe Chang
Department of Computer Science
National Chiao Tung University
destiny10191019.cs05@nctu.edu.tw

1 Introduction

Q-learning is an off-policy method which can used in the Markov Decision Processes(MDPs). It uses the information observed to approximate the optimal function, from which one can construct the optimal policy. The update of Q-learning is

$$Q_{t+1}(s_t, a_t) = Q_t(s_t, a_t) + \alpha(r_t + \gamma \max_a Q_t(s_{t+1}, a) - Q_t(s_t, a_t)) \tag{1}$$

In the equation (1), $Q_t(s_t, a_t)$ is the Q-value of the action a in the state s at time t, r_t is the reward at time t. The discount rate is $\gamma \in [0, 1)$, and the learning rate is $\alpha \in [0, 1]$.

The update of Q-learning uses the Q-value of the highest value action in the next state, $\max_a Q_t(s_{t+1}, a)$. If all the Q-values are very close, we can say that the Q-values are stored with noise. Using the maximum of serveral noisy Q-values is an overestimate as,

$$E[\max(\vec{a})] \geq \max[E(\vec{a})] \tag{2}$$

where $\vec{a}$ is a vector and E is the expectation operator. In the Q-learning case,

$$E[\max_a Q_t(s_{t+1}, a)] \geq \max_a E[Q_t(s_{t+1}, a)] \tag{3}$$

When updating the Q-learning, we will overestimate the Q-value of the next state. The systematic overestimation will cause the poor performace. In order to fix the poor performance caused by large overestimation of Q-values, we introduce an other way to approximate the maximum action value. Instead of using the single estimator in the Q-learning, we apply the double estimator to Q-learning to contruct Double Q-learning.

2 Problem Formulation

2.1 Markov Decision Processes

Reinforcement learning can be used to find optimal solutions for many problems. Before starting, we need to model this problems, so the framework of Markov Decision Processes(MDPs) is used. The notation in the MDP:

- S is a set of states, where $s_t \in S$ denotes the state the agnet in at time t.
- $A(s)$ is a set of available actions in state s, where $a_t \in A(s_t)$ denotes the action the agent performs at time t
- $P : S \times A \times S \to [0, 1]$ is a transition function where $P_{sa}^{s'}$ denotes the probability of ending up in state s' when performing action a in state s.
- $R : S \times A \times S \to \mathbb{R}$ is a reward function where $R_{sa}^{s'}$ denotes the expected reward when the agent transitions from state s to state s' after performing action a. The actual reward that is witnessed by the agent after performing action at and on transitioning to state s_{t+1} may contain noise and is denoted as r_{t+1}, where $E(r_{t+1}|(s, a, s') = (s_t, a_t, s_{t+1})) = R_{sa}^{s'}$.

Preprint. Under review.

- $\gamma \in [0, 1]$ is a discount factor.
- $Q(s, a)$ is the action-value of taking action a in the state s.

2.2 Single Estimator

Let the set of N random variables $X = \{X_1, X_2, ..., X_N\}$ and $u = \{u_1, u_2, ..., u_N\}$ be a set of unbiased estimators such that $E(u_i) = E(X_i)$ for all i. Assume that $D = \cup_{i=1}^{N} D_i$ is a set of samples, where D_i is the subset containing at least one sample for the variable X_i, and the samples in D_i are i.i.d. The single estimator method uses the value $\max\limits_i u_i(D)$ as an estimator of $\max\limits_i E(X_i)$, where $u_i(D) = \frac{1}{|D_i|} \sum\limits_{d \in D_i} d$ is an unbiased estimator for the value of $E(X_i)$. However, $\max\limits_i E(u_i(D)) \geq \max\limits_i E(X_i)$. We will show that the single estimator would cause overestimation in the Lamma 3.1. To avoid overestimation in the single estimator, we proposed the double estimator estimator approach.

2.3 Double Estimator

The key difference between single estimator and double estimator is that double estimator divides the sample set D into two disjoint subsets, D^U and D^V. Let $u^U = \{u_1^U, u_2^U, ..., u_N^U\}$ and $u^V = \{u_1^V, u_2^V, ..., u_N^V\}$ be two sets of unbiased estimators such that $E(u_i^U) = E(u_i^V) = E(X_i)$ for all i.The two sample subsets are used to learn two independent estimates, $u_i^U(D) = \frac{1}{|D_i^U|} \sum\limits_{d \in D_i^U} d$ and $u_i^V(D) = \frac{1}{|D_i^V|} \sum\limits_{d \in D_i^V} d$. The optimal action in the $u_i^U(D)$ is $a^* \in \arg\max\limits_i u_i^U(D)$, and the optimal action in the $u_i^V(D)$ is $a^* \in \arg\max\limits_i u_i^V(D)$. Hence, we obtain the euqation that $E(u_{a^*}^U(D)) = E(X_{a^*})$ and $E(u_{a^*}^V(D)) = E(X_{a^*})$. Since $E(X_{a^*}) \leq \max\limits_i E(X_i)$, and we can get $E(u_{a^*}^V(D)) \leq \max\limits_i E(X_i))$. We will show this property in Lamma 3.2.

Algorithm 1 Double Q-learning

```
 1: Initialize $Q^U$, $Q^V$, s
 2: while s is not terminal state do
 3:     Choose action $a$ from state $s$ based on $Q^U$ and $Q^V$
 4:     Take action $a$, observe reward $r$, next state $s'$
 5:     Choose one table to update($Q^U$ or $Q^V$)
 6:     if choose update $Q^U$ then
 7:         $u^* = \arg\max_a Q^U(s', a)$
 8:         $Q^U(s, a) = Q^U(s, a) + \alpha(r + \gamma Q^V(s', u^*) - Q^U(s, a))$
 9:     else if choose update $Q^V$ then
10:         $v^* = \arg\max_a Q^V(s', a)$
11:         $Q^V(s, a) = Q^V(s, a) + \alpha(r + \gamma Q^U(s', v^*) - Q^V(s, a))$
12:     end if
13:     $s = s'$
14: end while
```

2.4 Double Q-learning

Double Q-learning, as shown in Algorithm 1, applys the double estimator to estimate the maximum expected value of the Q function for the next state, which is $\max\limits_{a'} E(Q(s', a'))$. It contains two Q functions, Q^U and Q^V. Double Q-learning uses two separate subsets of experience samples to learn the two Q functions. The difference between Q-learning and double Q-learning is in the line 8. We update $Q^U(s, a)$ uses $Q^V(s', a^*)$ instead of $Q^U(s', a^*)$. Because Q^V is updated by other set of experience samples. We can get that $Q^V(s', a^*)$ is an unbiased estimate Q-value in the state s and action a^*. With this property, we know that $E(Q^V(s', a^*)) = E(Q(s', a^*))$. There is same property in the line 11. We will show that double Q-learning will converge in the limit in theorem 1.

3 Theoretical Analysis

Definition 3.1 (Optimal estimators) *An estimator m_j and the corresponding random variable Y_j with expected value $u_j = E(Y_j)$ are called optimal for a given set of random variables Y if $u_j = \max_i u_i$. An index $j \in \{1, ..., M\}$ is called optimal if the corresponding random variable Y_j is optimal. The set of optimal indices is denoted by $\mathcal{O}$ and is thus defined by*

$$\mathcal{O} \overset{def}{=} \{j | u_j = \max_i u_i\} \tag{4}$$

Definition 3.2 (Maximal estimators) *An estimator m_j is called maximal for a given set of samples X if $m_j(X) \geq m_i(X)$ for all i. An index $j \in \{1, ..., M\}$ is called maximal if the corresponding estimator is maximal and the set of maximal indices is denoted by*

$$\mathcal{M}(x) \overset{def}{=} \{j | m_j(X) = \max_i m_i(X)\} \tag{5}$$

Lemma 3.1 *Let $Y = \{Y_1, Y_2, ..., Y_M\}$ be a set of random variables with expected values $u_i, ..., u_M$ and let $m = \{m_1, ..., m_M\}$ be a set of unbiased estimators such that $E(m_i) = u_i$, for all i. Assume that a set of samples X contains at least one sample for each of the variables in Y. Let O be the set of optimal estimators as defined in Definition 3.1 and let $M(X)$ be the set of maximal indices for X as defined in Definition 3.2. Then*

$$\forall j \in \mathcal{O} : E(\max_i m_i) \geq E(m_j) = u_j = \max_i u_i \tag{6}$$

Proof 1 *Assume $j \in \mathcal{O}$, such that by Definition 3.1 m_j is an optimal estimator. Then*

$$\begin{aligned} E(\max_i m_i) &= P(j \in \mathcal{M})E(\max_i m_i | j \in \mathcal{M}) + P(j \notin \mathcal{M})E(\max_i m_i | j \notin \mathcal{M}) \\ &= P(j \in \mathcal{M})E(m_j | j \in \mathcal{M}) + P(j \notin \mathcal{M})E(\max_i m_i | j \notin \mathcal{M}) \\ &\geq P(j \in \mathcal{M})E(m_j | j \in \mathcal{M}) + P(j \notin \mathcal{M})E(m_j | j \notin \mathcal{M}) \\ &= E(m_j) = u_j = \max_i u_i \end{aligned} \tag{7}$$

Lemma 3.2 *Let $Y = \{Y_1, Y_2, ..., Y_M\}$ be a set of random variables with expected values $u_1, u_2, ..., u_M$ and let $m^A = \{m_1^A, ..., m_M^A\}$ and $m^B = \{m_1^B, ..., m_M^B\}$ be two sets of unbiased estimators such that $E\{m_i^A\} = E\{m_i^B\} = u_i$ for all i. Let $a^* \in \mathcal{M}^A(X)$ denote a maximal element in $m^A(X)$ and let $\mathcal{O}$ be the set of optimal indices, as defined in Definition 3.1. Then*

$$E(m_j^B | j \in \mathcal{M}^A) = E(Y_{a^*}) \leq \max_i u_i \tag{8}$$

Proof 2 *Assume a^* is optimal, such that $a^* \in \mathcal{O}$. Then, because $m_{a^*}^B$ is an unbiased estimator for u_{a^*}, and by definition of $\mathcal{O}$, we can get the equation:*

$$E(m_{a^*}^B | a^* \in \mathcal{O}) = E(u_{a^*} | a^* \in \mathcal{O}) = \max_i u_i \tag{9}$$

If the a^* is not optimal, such that $a^* \notin \mathcal{O}$. Then we can get:

$$E(m_{a^*}^B | a^* \notin \mathcal{O}) = E(u_{a^*} | a^* \notin \mathcal{O}) < \max_i u_i \tag{10}$$

Because we know one of these situations must hold, so combine these:

$$E(m_{a^*}^B) \leq \max_i u_i \tag{11}$$

Lemma 3.3 *A stochastic process (ζ_t, Δ_t, F_t), where $\zeta_t, \Delta_t, F_t : X \to \mathbb{R}$ satify the equation:*

$$\Delta_{t+1}(x_t) = (1 - \zeta_t(x_t))\Delta_t(x_t) + \zeta_t(x_t)F_t(x_t) \tag{12}$$

where $x_t \in X andt = 0, 1, 2.....$ Let P_t be a sequence of increasing σ-fields such that ζ_0 and Δ_0 are P_0-measurable and $\zeta_t, \Delta_t andF_{t-1}$ are P_t-measurable, $t \geq 1$. Assume that the following hold:

1. *the set X is finite.*
2. $\zeta_t \in [0,1]$, $\sum_t \zeta_t(x_t) = \infty$, $\sum_t (\zeta_t(x_t))^2 < \infty$*w.p.1 and* $\forall x \neq x_t : \zeta_t(x) = 0$
3. $||E(F_t|P_t)|| \leq k||\Delta_t|| + c_t$, *where* $k \in [0,1)$ *and* c_t *converages to zero w.p.1.*
4. $Var(F_t(x_t)|P_t) \leq K(1 + k||\Delta_t||)^2$, *where* K *is constant.*

where $||\cdot||$ *denotes a maximum norm. Then* Δ_t *converges to zero with probability one.*

Theorem 1 *Given an ergodic MDP, both* Q^U *and* Q^V *as updated by Double Q-learning as described in Algorithm 1 will converge to the optimal value function* Q^* *with probability one if an infinite number of experiences in the form of rewards and state transitions for each state action pair are given by a proper learning policywhen the six conditions are fulfilled.*

1. *The MDP is finite, i.e.* $|S \times A| < \infty$
2. $\gamma \in [0,1)$
3. *the action values are stored in a lookup table*
4. *both* Q^U *and* Q^V *receive an infinite number of updates*
5. $\alpha_t(s,a) \in [0,1)$, $\sum_t \alpha_t(s,a) = \infty$, $\sum_t (\alpha_t(s,a))^2 < \infty$ *w.p.1,* $\forall s, a \neq s_t, a_t : \alpha_t(s,a) = 0$
6. $\forall s, a, s' : Var(R_{sa}^{s'}) < \infty$

Proof 3 *Because We update* Q^U *and* Q^V *symmetrically, it can show convergence for either of these. We use Lamma 3.3 to prove Theorem 1. Let* $P_t = \{Q_0^U, Q_0^V, s_0, a_0, \alpha_0, r_1, s_1, ..., s_t, a_t\}$, $X = S \times A$, $\Delta_t = Q_t^U - Q^*$, $\zeta^U = \alpha$, *and* $F_t(s_t, a_t) = r_t + \gamma Q_t^V(s_{t+1}, u^*) - Q_t^*(s_t, a_t)$, *where* $u^* = \arg\max_a Q^U(s_{t+1}, a)$. *The first two conditions of the lemma can be easily hold because of the first condition and the fifth conditon in the theorem 1.The fourth condition of the lemma holds as a consequence of the boundedness condition on the variance of the rewards in the theorem. This, together with the condition that the discount factor is lower than 1, ensures that the Q values are bounded. The theorem can be extended to undiscounted MDPs that have a non-zero probability of eventually terminating for all states and actions. Hence, we only need to prove that the third condition of tje expected contraction of* F_t *holds. First, we can write:*

$$F_t(s_t, a_t) = F_t^Q(s_t, a_t) + \gamma(Q_t^V(s_{t+1}, u^*) - Q_t^U(s_{t+1}, u^*)) \tag{13}$$

where $F_t^Q = r_t + \gamma(Q_t^U(s_{t+1}, a^*) - Q_t^*(s_t, a_t)$ *is the value of* F_t *if normal Q-learning would be under consideration. We know that* $E(F_t^Q|P_t) \leq \gamma||\Delta_t||$, *so we can get:*

$$C_t = \gamma E(Q_t^V(s_{t+1}, u^*) - Q_t^U(s_{t+1}, u^*)|P_t) \tag{14}$$

Now, we define a new function:

$$\Delta_t^{VU}(s,a) = Q_t^V(s_t, a) - Q_t^U(s_t, a) \tag{15}$$

In order to prove the convergence of c_t *to zero, we just need to prove that* Δ_t^{VU} *will convergence to zero. There in two conditions need to handle, whether* Q^V *or* Q^U *is updated. Hence, we can get* Δ_t^{VU} *is either:*

$$\begin{aligned} \Delta_{t+1}^{VU}(s_t, a_t) &= \Delta_t^{VU}(s_t, a_t) + \alpha_t(s_t, a_t) F_t^V(s_t, a_t), or \\ \Delta_{t+1}^{VU}(s_t, a_t) &= \Delta_t^{VU}(s_t, a_t) - \alpha_t(s_t, a_t) F_t^U(s_t, a_t) \end{aligned} \tag{16}$$

where

$$\begin{aligned} F_t^U(s_t, a_t) &= r_t + \gamma Q_t^V(s_{t+1}, u^*) - Q_t^U(s_t, a_t) and \\ F_t^V(s_t, a_t) &= r_t + \gamma Q_t^U(s_{t+1}, v^*) - Q_t^V(s_t, a_t) \end{aligned} \tag{17}$$

Then we can get:

$$\begin{aligned}
&E(\Delta_{t+1}^{VU}(s_t, a_t)|P_t) \\
&= \Delta_t^{VU}(s_t, a_t) + \frac{\alpha_t(s_t, a_t)}{2} E(F_t^V(s_t, a_t) - F_t^U(s_t, a_t)|P_t) \\
&= (1 - \zeta_t^{VU}(s_t, a_t)\Delta_t^{VU}(s_t, a_t) + \zeta_t^{VU}(s_t, a_t)E(F_t^{VU}(s_t, a_t))|P_t)
\end{aligned} \tag{18}$$

when $\zeta_t^{VU}(s, a) = \frac{\alpha_t(s,a)}{2}$*, we can get:*

$$E(F_t^{VU}(s_t, a_t))|P_t) = \gamma E(Q_t^U(s_{t+1}, v^*) - Q_t^V(s_{t+1}, u^*)|P_t) \tag{19}$$

Now, we need to show that $||E(F_t^{VU}|P_t)|| \leq k||\Delta_t^{VU}||$ *for all* $k \in [0, 1)$*. To prove that, we can get two conditions.*
First, let $E(Q_t^U(s_{t+1}, v^*)|P_t) \geq E(Q_t^V(s_{t+1}, u^*|P_t))$*, where* v^* *and* u^* *are defined in line 7 and line 10 of the algorithm 1. So we can get* $Q_t^U(s_{t+1}, u^*) = \max_a Q_t^U(s_{t+1}, a) \geq Q_t^U(s_{t+1}, v*)$*.*
Then:

$$\begin{aligned}
|E(F_t^{VU}s_t, a_t|P_t)| &= \gamma E(Q_t^U(s_{t+1}, v*) - Q_t^V(s_{t+1}, u*)|P_t) \\
&\leq \gamma E(Q_t^U(s_{t+1}, u*) - Q_t^V(s_{t+1}, u*)|P_t) \leq \gamma||\Delta_t^{UV}||
\end{aligned} \tag{20}$$

Second, let $E(Q_t^U(s_{t+1}, v^*)|P_t) < E(Q_t^V(s_{t+1}, u^*|P_t))$*, we can get* $Q_t^V(s_{t+1}, v^*) = \max_a Q_t^V(s_{t+1}, a) \geq Q_t^V(s_{t+1}, u*)$*.*
Then:

$$\begin{aligned}
|E(F_t^{VU}s_t, a_t|P_t)| &= \gamma E(Q_t^V(s_{t+1}, u*) - Q_t^U(s_{t+1}, v*)|P_t) \\
&\leq \gamma E(Q_t^V(s_{t+1}, v*) - Q_t^U(s_{t+1}, v*)|P_t) \leq \gamma||\Delta_t^{UV}||
\end{aligned} \tag{21}$$

Now, we know the result that $||E(F_t^{VU}|P_t)|| \leq k||\Delta_t^{VU}||$*. Because of the lemma 3.3,* Δ_t^{VU} *will converge to zero. So we can ensure that the original process will converge in the limit.*

4 Conclusion

We presented the problem of the overestimation in the original Q-learning, and proved that the single estimator has a positive bias in proof 1. We introduced an new method- double Q-learning, using the double estimator, and proved double Q-learning also converges in the limit in proof 3. But there have some limitations in double Q-learning. Double Q-learning somtimes underestimates the action values(proof 2). In latest, Deep reinforcement learning with double q-learning(AAAI, 2016) constructs a new algorithm called Double DQN based on double Q-learning, which can get more accurate value estimates, and much higher scores on several games.