The following document is a summarisation of https://towardsdatascience.com/reinforcement-learning-markov-decision-process-part-2-96837c936ec3. The figures also refers to the same blog post given in the link.

Before reading this document it would be better if you could watch the deepmind lecture:

DeepMind

2018
LECTURE 1

Introduction to Reinforcement Learning

REINFORCEMENT LEARNING

Value function and Q value

The expected value of a state is given by,

$$V_\pi(s_t) = \mathbb{E}[R_{t+1} + \gamma V_\pi(s_{t+1})]$$

according to the **bellman expectation equation.** In other words, the expected value at time step t is obtained by following the policy π. The expected value is calculated using the immediate reward the agent gets for following the policy π + the expected discounted future state values (Rewards) the agent will get by following the same policy π.

The expected state action value is given by,

$$q_\pi(s_t, a_t) = \mathbb{E}[R_{t+1} + \gamma q_\pi(s_{t+1}, a_{t+1})]$$

Similar to the value function the expected q value at a time step t is the immediate reward agent obtains when the action a_t is executed + the expected discounted future q values for taking some future actions

LET'S CONSIDER SCENARIO: ONE TRANSITION STEP (from state $s -> s'$)

Now let's see how the value function and action value function are related using a **backup** diagram. The empty circles are states and filled circles are actions. The diagram shows the state and action relationship.

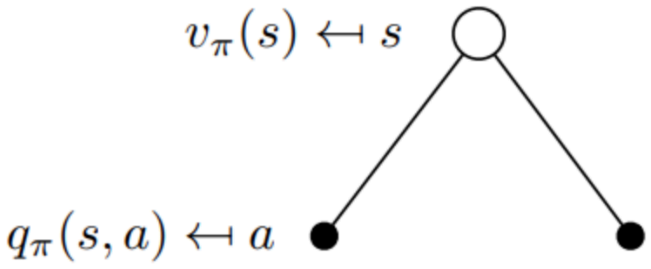
$v_\pi(s) \leftarrow\!\shortmid\, s$
$q_\pi(s,a) \leftarrow\!\shortmid\, a$

Fig-1: The diagram shows an agent at state **s** and executing the action **a**

In Fig-1, From the state s, there are two possible actions an agent can take with a certain probability. So the value of the state is calculated by averaging the state action values (expectation).

$$V_\pi(s) = \sum_{a \in A} \pi(a \,|\, s) q_\pi(s,a) - (1)$$

The value at state s is obtained by multiply the probability of taking an action (policy $a \sim \pi(\,.\,|\,s)$) with the value of taking that action (q value) and summing them together (hence expectation). This is how the state values are related to the action values

Now let's see how the action values are related to the next state values by using the following backup diagram.

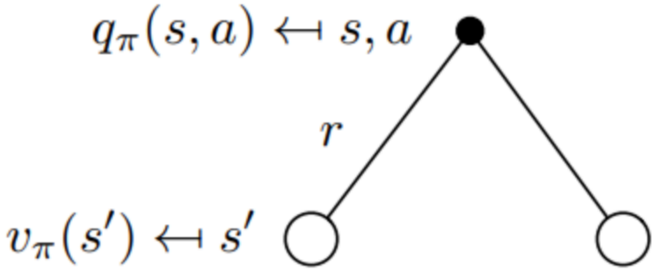
$q_\pi(s,a) \leftarrow\!\shortmid\, s,a$
r
$v_\pi(s') \leftarrow\!\shortmid\, s'$

Fig-2: Backup diagram that show the action value when an agent takes the action **a**

As shown in the diagram, if the agent takes an action a, it can end up in one of the future states with a given transition probability (In model free case agent has no idea about the transition probability of the MDP). This can be expressed mathematically as follows,

$$q_\pi(s,a) = R_s^a + \gamma \sum_{s' \in S} P_{s->s'}^a V_\pi(s') - (2)$$

The action value is calculated by summing the immediate reward the agent obtains by taking action a with the discounted expected value of the future state s'. The future value s' is weighted by the transition probability $P_{s->s'}^a$ (here the agent can end up in one of the two states by taking the action a)

Now to calculate the state value of state s obtained by the agent (when it follows the policy that would transit it from state $s -> s'$), the two backup diagrams (Fig-1 and Fig-2) can be stitched together.

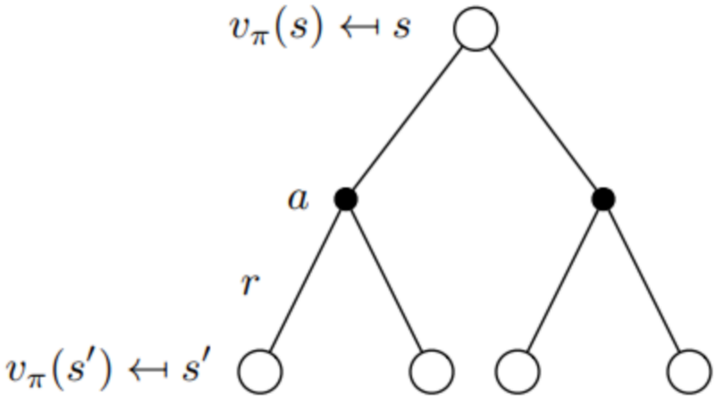
$v_\pi(s) \leftarrow\!\!\dashv s$
a
r
$v_\pi(s') \leftarrow\!\!\dashv s'$

Fig-3: Stitching the two graph together to find the value of a state given a policy

The equation of the graph shown in figure Fig-3 is as follows (a combination of equation 1 and 2),

$$V_\pi(s) = \sum_{a \in A} \pi(a \mid s)(R_s^a + \gamma \sum_{s' \in S} P_{s->s'}^a V_\pi(s')) - (3)$$

As shown in the diagram when transiting from state s to state s', an agent can take 2 action (with a probability determined by the policy. This probability is a property of the agent that it should learn). For each action the agent can end up in one of two possible states (the transit here is determined by the transit probability. The transit probability is a property of the environment).

Therefore, the overall value for taking an action is weighed by the policy and the next state transition is weighed by the transition probability and sum them together to find the expected state value. The immediate reward the agent gets is a solid value determined by the action taken

Similarly we can derive a backup diagram to calculate the action value for a given state action pair.

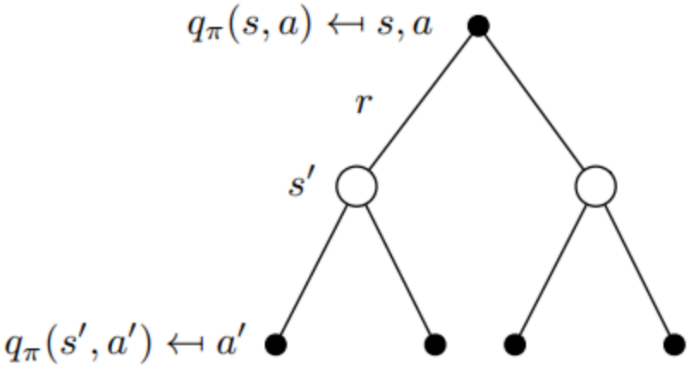
$q_\pi(s,a) \leftarrow\!\!\dashv s,a$
r
s'
$q_\pi(s',a') \leftarrow\!\!\dashv a'$

Fig-4: Stitching the two graph together to find the action value for a given state action pair

Similar to the state value case (3), the equations (1) and (2) can be combined to obtain the

$$q_\pi(s,a) = R_s^a + \gamma \sum_{s' \in S} P^a_{s->s'} \sum_{a \in A} \pi(a'|s') q_\pi(s',a') -$$ (4)

The figure Fig-4 shows that, the value of an action that is taken at s can transit the agent to one of the two states with a given transition probability. Once the agent ends up in the state s' it takes the action a'. The action value at state s' is weighed by the policy (**Following the Policy after taking an action**)

The above equations, equation 1 - 4 are derived considering the policies are stochastic. When the agent acts under a deterministic policy, then $\pi(a|s) = 1$ or $a = \mu(s)$ where μ is the deterministic policy.

The optimal value and policy function for a given MPD can be found as follows.

The definitions: Optimal value of a state is the maximum state value an agent can obtain. The policy that the agent would follow to obtain the optimal state value is called the optimal policy. Similar to optimal state value, the optimal state action value is the maximum action value an agent can obtain by taking the best action a. The optimal equations are as follows,

Optimal value function: $V*(s) = max_\pi V_\pi(s) -$ (5)
Optimal state action function: $q*(s,a) = max_\pi q_\pi(s,a) -$ (6)
Optimal Policy: for all states s; $\pi \geqslant \pi'$ if $V_\pi(s) \geqslant V_{\pi'}(s) -$ (7)

Note that for a given MDP there could be many optimal policies, but for each optimal policy, one gets same optimal state value and action state value.

Now lets combine equation (1) and (5) to find the optimal state value:
When the agent follows the optimal policy, the probability of an action for that policy is,

$\pi* = 1$ if $a = argmax_{a \in A} q(s,a)$ and $\pi* = 0$ otherwise. So from (1) and (5)
$V*(s) = max_a q(s,a) -$ (8)
here we shifted from sum (expectation) to the max (optimal) to find the optimal state value.
The optimal state values can be visualised using following backup diagram. The arc shows the optimal action

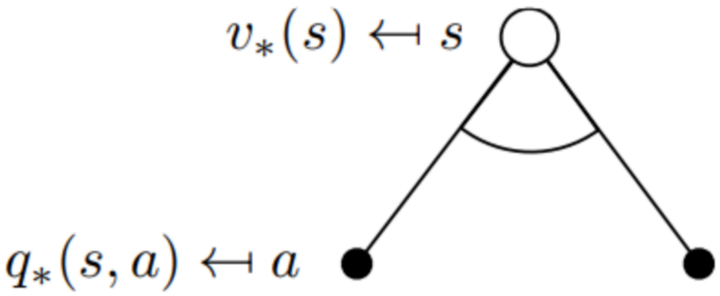

Fig-5: Optimal state value is calculated by finding the maximum q value

Now using (2) and (6) the optimal state action value can be found,

$$q*(s,a) = max_\pi(R_s^a + \gamma \sum_{s' \in S} P^a_{s->s'} V_\pi(s'))$$
$$q*(s,a) = R_s^a + \gamma \sum_{s' \in S} P^a_{s->s'} max_\pi(V_\pi(s'))$$

$$q*(s,a) = R_s^a + \gamma \sum_{s' \in S} P_{s->s'}^a V*(s') - \text{(9)}$$

The optimal state action value can be visualised in the following backup diagram;

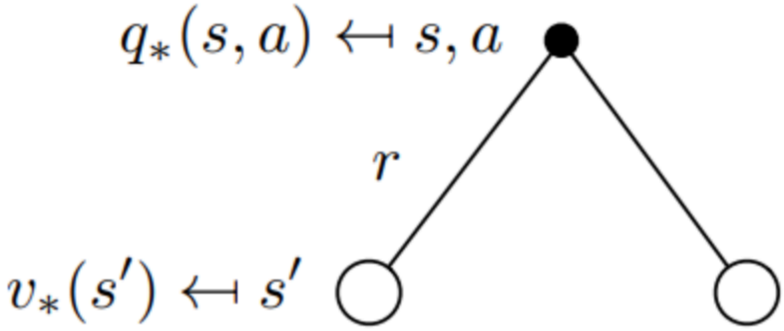

Fig-6: Optimal state action value backup diagram. Note that there is no arc here (max is taken at the stage of state not action in the diagram)

Now to calculate the optimal state value for following the optimal policy we can stitch the Fig-5 and 6 together.

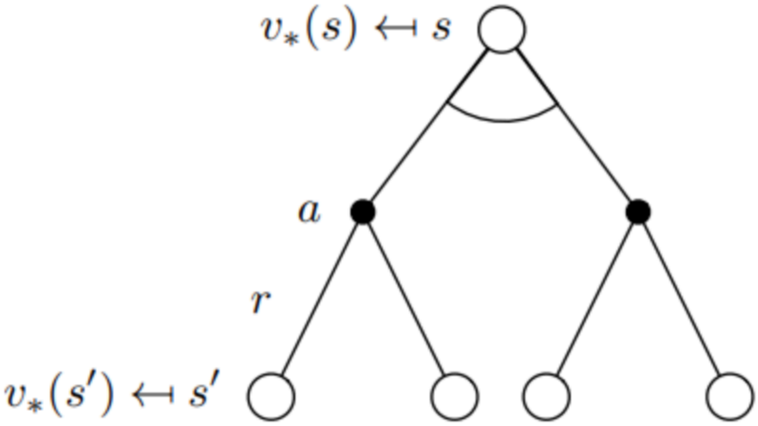

Fig-7: Optimal state value backup diagram. Take the action that max the q value and follow the policy

From (2) and (8) the optimal value at state s for following the optimal policy can be calculated.

$$V*(s) = max_a R_s^a + \gamma \sum_{s' \in S} P_{s->s'}^a max_\pi V_\pi(s')$$

$$V*(s) = max_a R_s^a + \gamma \sum_{s' \in S} P_{s->s'}^a V*(s')$$

So the optimal state value calculated by first finding the maximum immediate reward the agent get sum with the discounted optimal future state values. The state transition probability is considered since it's an environmental property in the MDP.

Similar to optimal state value, the optimal state action values can be illustrated using a backup diagram.

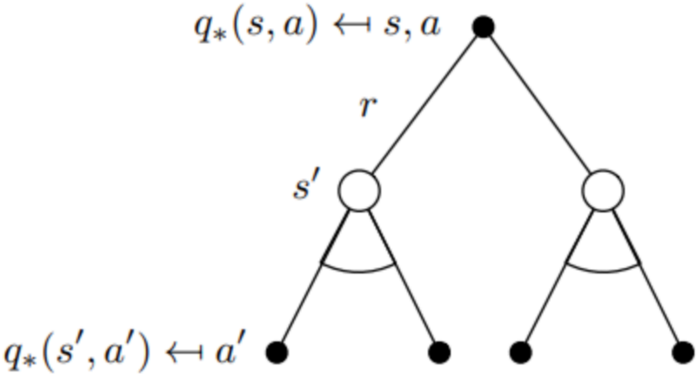

Fig-8: Optimal state action value backup diagram.

In this case, the agent takes an action a and lands in one of the state s’ with a transition probability. Then from that landed state, agent takes the optimal action that maximises the q value. This can be expressed using equations (8) and (9) as follows,

$$q^*(s,a) = R_s^a + \gamma \sum_{s' \in S} P^a_{s->s'} max_{a'} q(s',a')$$

Reference:

- https://towardsdatascience.com/reinforcement-learning-markov-decision-process-part-2-96837c936ec3

- On MDP please refer to chapter 1 to 3 in the book Reinforcement learning an Introduction Richard Sutton http://www.incompleteideas.net/book/RLbook2020.pdf

- A practical introduction to Q-Learning: https://blog.floydhub.com/an-introduction-to-q-learning-reinforcement-learning/

- On Deep Q-Learning: https://www.analyticsvidhya.com/blog/2019/04/introduction-deep-q-learning-python/ https://towardsdatascience.com/deep-q-learning-tutorial-mindqn-2a4c855abffc

- Original DQN paper from deep mind: https://www.cs.toronto.edu/~vmnih/docs/dqn.pdf