Objective:

Given a model-free environment, find an optimal policy that dictates which actions to take in any given state. This policy is determined by training a machine to take actions that maximize cumulative (long-term) rewards.

Inputs:

states, and rewards

Outputs:

actions

Environment:

Defines the set of actions that can be taken at any given state. Each action has a reward associated with it, and actions potentially change the state.

Markov decision processes:

Outcomes are partly random.

Q-values:

For the current state and action, Q-value is the current estimate of cumulative rewards.

Q-table:

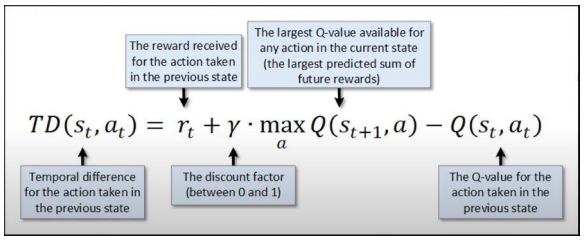
Q-values are stored in a Q-table, which has one row for each possible state and one column for each possible action. Q-table represents the policy for taking actions in the environment.

Temporal Differences (TDs):

If the new state provides a relatively good reward, the Q-value for the previous (most recent) action is increased, based on the idea that the maximum future reward for that action cannot exceed the sum of the immediate reward for taking that action and the maximum possible future reward in the next state. (heuristic)

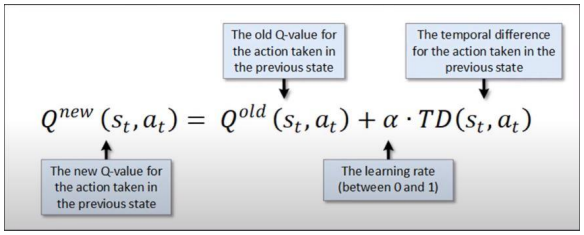
Discount Factor (gamma):

Between 0 and 1, somewhat close to 1.



Bellman Equation:

Determines the new Q-value based on actions taken in the previous state.



Step 1:

Initialize Q-table

Step 2:

Choose an action from the Q-table for the current state. Instead of always choosing an action that has the highest Q-value, we use the Epsilon-Greedy strategy here. Epsilon-Greedy algorithm encourages the agent to explore its environment by occasionally choosing a random action. (Exploration vs Exploitation)

Step 3:

Act, and update state

Step 4:

Receive reward for taking the most recent action, and compute TD for the action.

Step 5:

Update the Q-value for the most recent action based on the TD computed and the Bellman equation. Go back to step 2.

When we reach a terminal state, we reset the agent to an initial state, and restart episode. The new episode uses the updated Q-values from the previous episode.