Example: Off-policy TD Control - Q-learning for Simple Grid World

Consider a 3x3 grid world where the agent can move left, right, up, or down. The grid has a reward of -1 for each step and a reward of $+10$ for reaching the goal state. The discount factor γ is set to 0.9.

The Q-learning algorithm updates the Q-values based on observed transitions and rewards. The update rule for Q-values is given by:

$$Q(s,a) \leftarrow Q(s,a) + \alpha \left[R + \gamma \max_{a'} Q(s',a') - Q(s,a)\right]$$

where:

- $Q(s,a)$ is the Q-value for state s and action a,
- α is the learning rate,
- R is the reward received after taking action a in state s,
- γ is the discount factor,
- s' is the next state,
- a' is the next action.

Here's a step-by-step example of how Q-learning can estimate the Q-values for the grid world:

1. **Initialization**: Start with initial Q-values of zero for all state-action pairs.

State	Action (left)	Action (right)
$S1$	0	0
$S2$	0	0
G	0	0

2. **Agent's Action**: The agent selects an action based on an ϵ-greedy policy. Let's say the agent selects to move right from state S1 ($S1 \to S2$).

3. **Transition**: The agent transitions to state S2 and receives a reward of -1.

4. **Update Q-Value for State S1 and Action Right**:

$$\begin{aligned} Q(S1, \text{right}) &\leftarrow Q(S1, \text{right}) + \alpha \left[-1 + \gamma \max_{a'} Q(S2, a') - Q(S1, \text{right})\right] \\ &\leftarrow 0 + \alpha \left[-1 + 0.9 \times 0 - 0\right] \\ &\leftarrow \alpha \times (-1) \end{aligned}$$

5. **Update Q-Value for State S2 and Action Left**:

   Since there are no further actions in this episode, the Q-value for state S2 remains unchanged.

6. **End of Episode**: The episode ends.

7. **Repeat**: Repeat the above steps for multiple episodes to update Q-values and improve the policy.

The Q-learning algorithm iteratively updates the Q-values based on observed transitions and rewards, gradually learning the optimal policy for the grid world.