Glossary

This is a community-created glossary. Contributions are welcomed!

Strategies to find the optimal policy

- **Policy-based methods.** The policy is usually trained with a neural network to select what action to take given a state. In this case it is the neural network which outputs the action that the agent should take instead of using a value function. Depending on the experience received by the environment, the neural network will be re-adjusted and will provide better actions.
- **Value-based methods.** In this case, a value function is trained to output the value of a state or a state-action pair that will represent our policy. However, this value doesn't define what action the agent should take. In contrast, we need to specify the behavior of the agent given the output of the value function. For example, we could decide to adopt a policy to take the action that always leads to the biggest reward (Greedy Policy). In summary, the policy is a Greedy Policy (or whatever decision the user takes) that uses the values of the value-function to decide the actions to take.

Among the value-based methods, we can find two main strategies

- **The state-value function.** For each state, the state-value function is the expected return if the agent starts in that state and follows the policy until the end.
- **The action-value function.** In contrast to the state-value function, the action-value calculates for each state and action pair the expected return if the agent starts in that state, takes that action, and then follows the policy forever after.

Epsilon-greedy strategy:

- Common strategy used in reinforcement learning that involves balancing exploration and exploitation.
- Chooses the action with the highest expected reward with a probability of 1-epsilon.
- Chooses a random action with a probability of epsilon.
- Epsilon is typically decreased over time to shift focus towards exploitation

- Involves always choosing the action that is expected to lead to the highest reward, based on the current knowledge of the environment. (Only exploitation)
- Always chooses the action with the highest expected reward.
- Does not include any exploration.
- Can be disadvantageous in environments with uncertainty or unknown optimal actions.

Off-policy vs on-policy algorithms

- **Off-policy algorithms:** A different policy is used at training time and inference time
- **On-policy algorithms:** The same policy is used during training and inference

Monte Carlo and Temporal Difference learning strategies

- **Monte Carlo (MC):** Learning at the end of the episode. With Monte Carlo, we wait until the episode ends and then we update the value function (or policy function) from a complete episode.

- **Temporal Difference (TD):** Learning at each step. With Temporal Difference Learning, we update the value function (or policy function) at each step without requiring a complete episode.

If you want to improve the course, you can open a Pull Request.

This glossary was made possible thanks to: