Hackerrank Exam

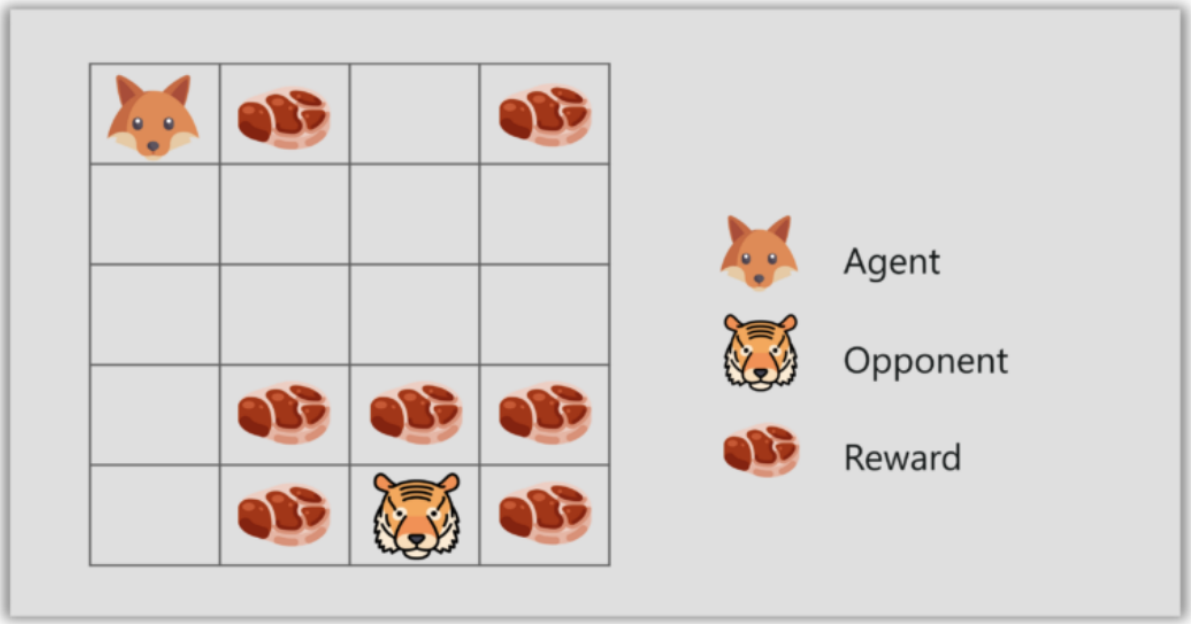
(Statistical Learning, reinforcement and Deep Learning)

Q1. What is exploitation and exploration trade-off?
Solution:

An important concept in reinforcement learning is the exploration and exploitation trade-off.

Exploration, like the name suggests, is about exploring and capturing more information about an environment. On the other hand, exploitation is about using the already known exploited information to heighten the rewards.

Eg case:



- Consider the fox and tiger example, where the fox eats only the meat (small) chunks close to him but he doesn't eat the bigger meat chunks at the top, even though the bigger meat chunks would get him more rewards.
- If the fox only focuses on the closest reward, he will never reach the big chunks of meat, this is called exploitation.
- But if the fox decides to explore a bit, it can find the bigger reward i.e. the big chunk of meat. This is exploration.

Q2. Explain the different algorithms used for hyperparameter optimization.

Solution:

Grid Search

Grid search trains the network for every combination by using the two set of hyperparameters, learning rate and the number of layers. Then evaluates the model by using Cross Validation techniques.

Random Search

It randomly samples the search space and evaluates sets from a particular probability distribution. For example, instead of checking all 10,000 samples, randomly selected 100 parameters can be checked.

Bayesian Optimization

This includes fine-tuning the hyperparameters by enabling automated model tuning. The model used for approximating the objective function is called the surrogate model (Gaussian Process). Bayesian Optimization uses Gaussian Process (GP) function to get posterior functions to make predictions based on prior functions.

Q3. What is Dropout?

Solution:

Dropout is a regularization technique to avoid overfitting thus increasing the generalizing power. Generally, we should use a small dropout value of 20%-50% of neurons with 20% providing a good starting point. A probability too low has minimal effect and a value too high results in under-learning by the network.

Use a larger network. You are likely to get better performance when dropout is used on a larger network, giving the model more of an opportunity to learn independent representations.

Q4. Write a bellman equation and what it defines?

Formally, Bellman equation defines the relationships between a given state (or state-action pair) to its successors. While many forms exist, the most common one usually encountered in Reinforcement Learning tasks is the Bellman equation for the optimal Q-Value , which is given by:

$$Q^*(s,a) = \sum_{s',r} p(s',r|s,a)\left[r + \gamma \max_{a'} Q^*(s',a')\right]$$

or when no uncertainty exists (meaning, probabilities are either 1 or 0):

$$Q^*(s,a) = r(s,a) + \gamma \max_{a'} Q^*(s',a')$$

where the asterisk sign indicates optimal value. Some algorithms, such as Q-Learning, are basing their learning procedure over it.

Q5. Why were we using Q-values? What is the advantage of learning state-action values (Q) compared to state values (V)? ?(Hint: consider action selection)

Ans:

The advantage is in the action selection. With Q-values, we can directly see the value of each available action, and use these in e.g. -greedy or softmax action selection. On the contrary, when we only have V-estimates, we need to 1) do extra calculations, and 2) have a transition model, so at each decision moment we can calculate the Q(s, a). for all available actions a. Especially when controlling a real-time system, your decisions have to be fast.

Q6. What is q-learning? Define below term. Exploration rate, learning rate and Discounting rate?

Ans:

Q-learning is an off policy reinforcement learning algorithm that seeks to find the best action to take given the current state. It's considered off-policy because the q-learning function learns from actions that are outside the current policy, like taking random actions, and therefore a policy isn't needed. More specifically, q-learning seeks to learn a policy that maximizes the total reward.

Exploration rate: This exploration rate is the probability that our agent will explore the environment rather than exploit it

Learning Rate: learning rate, often referred to as alpha or α, can simply be defined as how much you accept the new value vs the old value.

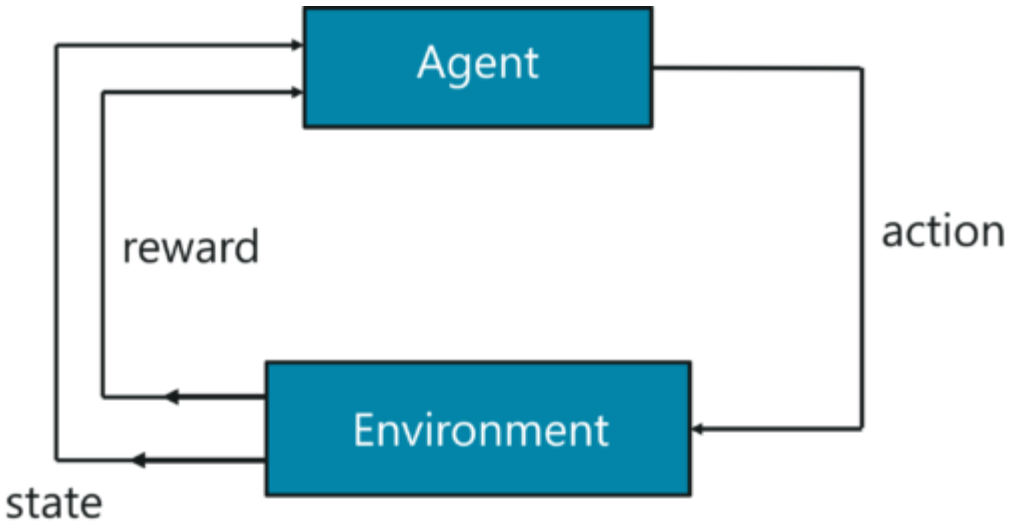
Discount factor: It's used to balance immediate and future reward.

Q7. Explain Markov's Decision Process with an example?

Ans:

The following parameters are used to attain a solution using MDP:

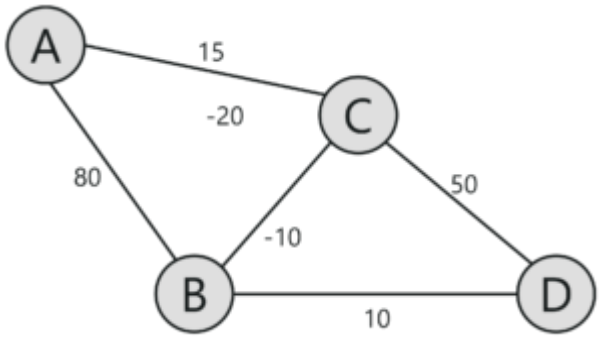
- Set of actions, A
- Set of states, S
- Reward, R
- Policy, π
- Value, V



Markov's Decision Process – Artificial Intelligence Interview Questions – Edureka

To briefly sum it up, the agent must take an action (A) to transition from the start state to the end state (S). While doing so, the agent receives rewards (R) for each action he takes. The series of actions taken by the agent, define the policy (π) and the rewards collected define the value (V). The main goal here is to maximize rewards by choosing the optimum policy.

To better understand the MDP, let's solve the Shortest Path Problem using the MDP approach:



Given the above representation, our goal here is to find the shortest path between 'A' and 'D'. Each edge has a number linked with it, this denotes the cost to traverse that edge. Now, the task at hand is to traverse from point 'A' to 'D', with minimum possible cost.

In this problem,

- The set of states are denoted by nodes i.e. {A, B, C, D}
- The action is to traverse from one node to another {A -> B, C -> D}
- The reward is the cost represented by each edge
- The policy is the path taken to reach the destination

You start off at node A and take baby steps to your destination. Initially, only the next possible node is visible to you, thus you randomly start off and then learn as you traverse through the network. The main goal is to choose the path with the lowest cost.

Q8. What is an activation function? Give an example of activation function

Ans: An activation function is something which helps us regulate the flow of information from the previous set of neurons to the next. It is a mathematical function which decides whether a particular neuron should be activated or not. Depending upon the type of activation values are shrunk in various ranges from -1 to +1 in case of tanh or 0 - 1 in case of sigmoid.

Example of activation function: Sigmoid $f(x) = 1 / (1 + e^{-x})$

Q9. What is the difference between feed forward neural network and recurrent neural network?

Ans:

A Feedforward Neural Network signals travel in one direction from input to output. There are no feedback loops; the network considers only the current input. It cannot memorize previous inputs (e.g., CNN).

A Recurrent Neural Network's signals travel in both directions, creating a looped network. It considers the current input with the previously received inputs for generating the output of a layer and can memorize past data due to its internal memory.

Q10. What is the difference between probability and likelihood?

Ans: Probability is the chance of an event to occur. Say we want to predict today's weather. If it rains it is (1) and does not rain is (0). So there is a 50% chance that it will rain and 50% for the other. Those are the only two true possible outcomes. Probability only plays on a set of possible finite possible outcomes of an event without any prior condition.

Likelihood is the conditional probability of an event occuring considering past occurrences. For our weather example, the likelihood that it will rain depends on the time of the year or the country in which we are talking about or many other factors and situations. There are many possible combinations and likelihood gives chance of either of them happening considering past success in events.