Reinforcement Learning

Foundations of Reinforcement Learning and Introduction to Open AI Gym

Learning Objectives

By the end of this lesson, you will be able to:

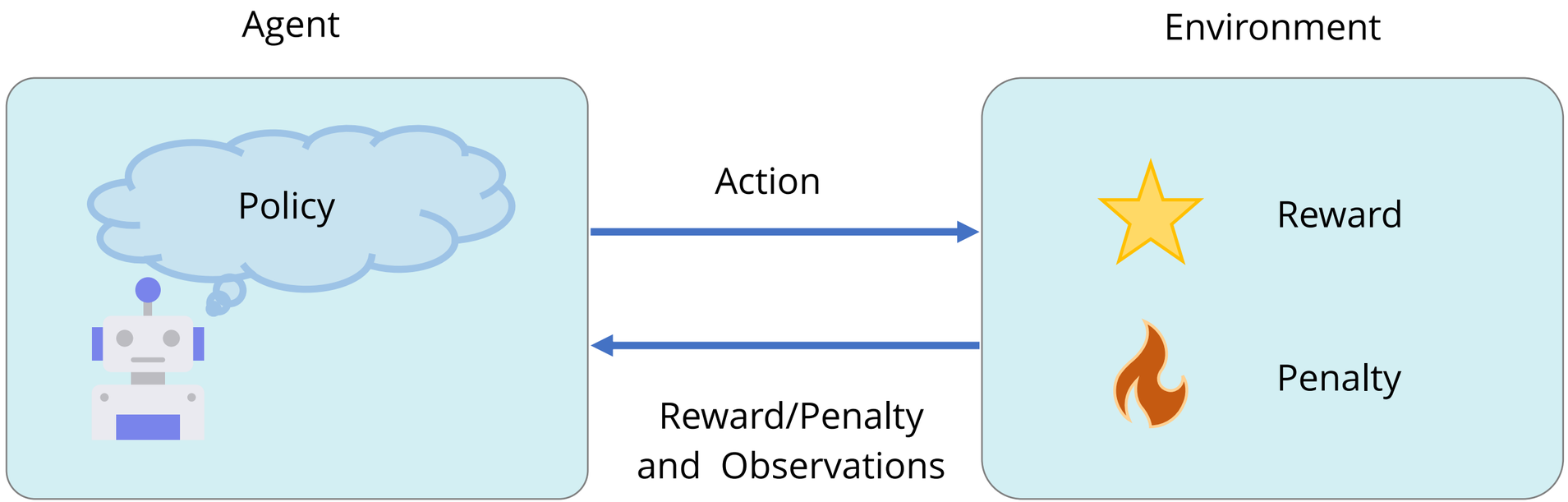
- Define the concept of policy in reinforcement learning
- Summarize the key steps for policy search in reinforcement learning
- Explore openAI gym tool for reinforcement learning
- Utilize openAI gym for policy search

Policy Search

What Is Policy Search?

Policy search focuses on identifying the optimal strategy (or policy) for an agent in an environment, aiming to maximize its reward.

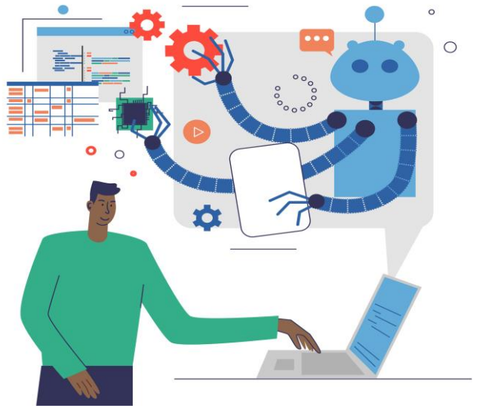
The method involves the direct identification of this strategy to guide the agent's actions over time.



Key Steps for Policy Search

Step 1: Define the policy

Firstly, a means to describe the policy is needed.



- This may encompass a set of rules or a mathematical function.
- The function take the current state of the environment as input and determine the action to be taken.

Key Steps for Policy Search

Step 2: Evaluate policies

Next, the policy is tested in the environment to assess its performance.

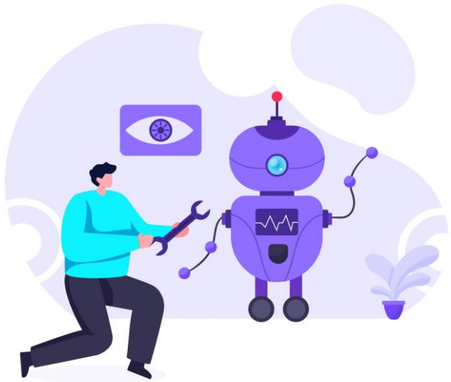
Using a robot as an example,

- It could be allowed to move around to evaluate how frequently it loses balance.
- The objective is to gather data on rewards, such as points awarded for walking without falling.

Key Steps for Policy Search

Step 3: Improve the policy

The policy is now updated to favor higher rewards.

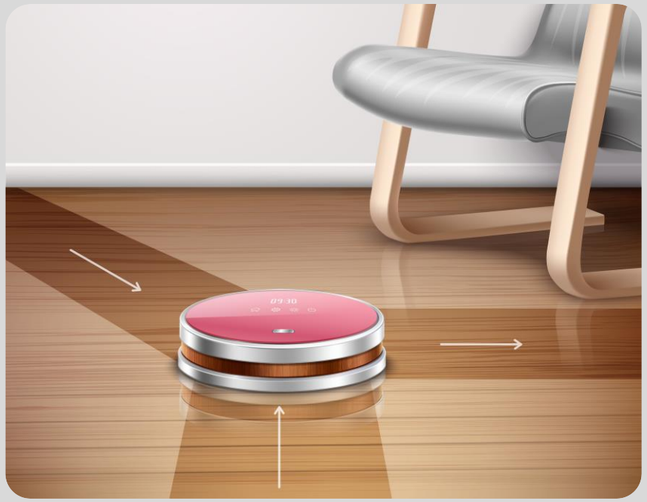


- Modifications are made to the policy based on its performance in an effort to achieve an improved score.
- This could involve adjusting some rules or fine-tuning the algorithm.

Example Study

Example: Robotic Vacuum Cleaner

Consider an example of a robotic vacuum cleaner where the reward is determined by the quantity of dust it collects within *T* minutes.



- Its policy might involve moving forward with a certain probability p every second or randomly rotating left or right with a probability of 1 - p.
- The rotation angle is determined as a random angle within the range of -r to +r.

Example: Robotic Vacuum Cleaner

To train such a robot, only two policy parameters can be adjusted: the probability *p* and the angle *r*. There may be three potential learning approaches.

1

Policy search

- Try out many different values for these parameters.
- And finally pick the combination that perform best.
- This is an example of policy search.

When policy space is too large, finding a good set of parameters this way is like searching a needle in haystack.

Example: Robotic Vacuum Cleaner

Another way to search the policy space is genetic algorithm.

2

Genetic algorithm

- A set of 100 initial policies is created randomly and tested.
- Subsequently, the 90 worst-performing policies are eliminated, and the 10 survivors generate 9 offspring each.
- An offspring is a duplicate of its parent with some random variations.

This iterative process continues until the best-performing policy is identified

Example: Robotic Vacuum Cleaner

Yet another approach is to use optimization techniques.

3

Policy gradient

- The gradients of rewards concerning the policy parameters are assessed.
- These parameters are adjusted by moving along the gradients toward higher rewards.
- This technique is referred to as the policy gradient technique.

OpenAI Gym

Need of OpenAI Gym

The development of gym aimed to address two main issues afflicting the field of reinforcement learning:

Standardization

Benchmarking

Experiments attempted by over 70 percent of researchers to reproduce another scientist's work were unsuccessful, and more than half were unable to replicate their own experiments.

Reference: https://www.nature.com/articles/533452a

What Is OpenAI Gym?

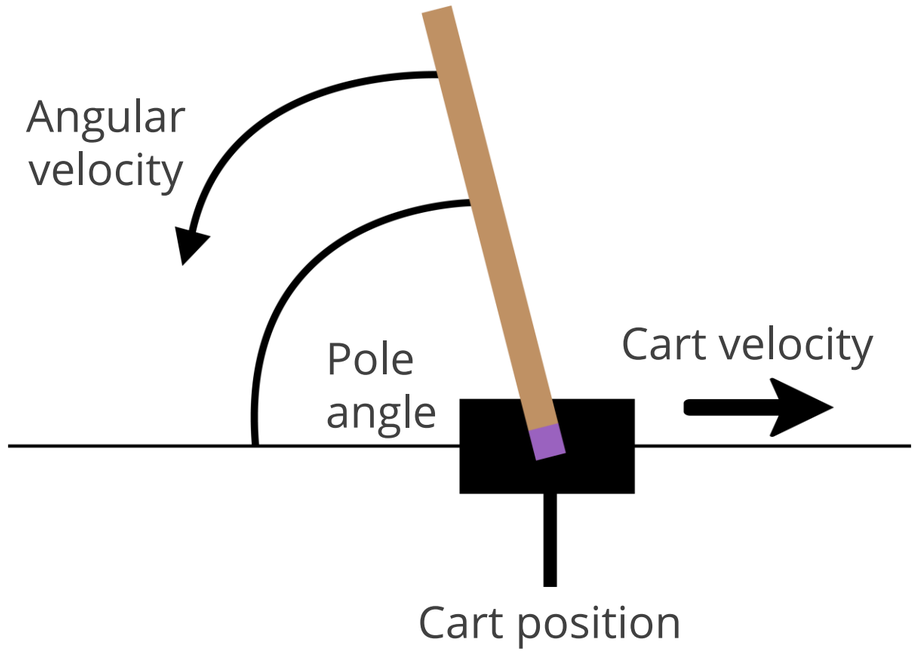
Gym is established to address the issue of non-standardization in papers and to create improved benchmarks by providing a versatile array of environments with easy setup.

```
# installing OpenAI gym
pip install --upgrade gym
```

It is an open-source toolkit that offers a diverse range of simulated environments, allowing agents to be trained, compared, or new reinforcement learning algorithms to be developed.

OpenAI Gym – Cart-Pole Environment

The cart-pole environment, also known as the **inverted pendulum**, is one of the classic reinforcement learning problems provided by OpenAI Gym.



It's a simulation of a pole standing upright on a cart, and the objective is to prevent the pole from falling over by moving the cart left or right.

Image source: Avoiding excess computation in asynchronous evolutionary algorithms by Eric O. Scott et.al.

Cart-Pole Environment - Key Elements

Key element of cart-pole Environment are as follows:

Objective

Observations

Actions

Rewards

Learning Challenge

- The agent needs to balance the pole on the cart by applying force to the cart's left or right side.
- The goal is to keep the pole balanced upright as long as possible.

Cart-Pole Environment - Key Elements

Key element of cart-pole Environment are as follows:

- Objective
- **Observations**
- Actions
- Rewards
- Learning Challenge

The agent observes four values from the environment:

- The position of the cart,
- The velocity of the cart,
- The angle of the pole, and
- The rotation rate of the pole.

Cart-Pole Environment - Key Elements

Key element of cart-pole Environment are as follows:

- Objective
- Observations
- **Actions**
- Rewards
- Learning Challenge

The agent can take one of two possible actions:

move the cart to the left

move the cart to the right

Cart-Pole Environment - Key Elements

Key element of cart-pole Environment are as follows:

- Objective
- Observations
- Actions
- **Rewards**
- Learning Challenge

- The agent receives a reward for each time step that the pole remains upright.
- The episode ends and the agent receives no further reward.
- Once the pole tilts more than a certain angle from vertical or the cart moves too far from the center.

Cart-Pole Environment - Key Elements

Key element of cart-pole Environment are as follows:

- Objective
- Observations
- Actions
- Rewards
- **Learning Challenge**

The key challenges in the cart-pole environment include:

- Agent must learn optimal actions.
- Consider cart's position, velocity, pole's angle, and rotation rate.
- Goal is to keep the pole balanced for maximum duration.

OpenAI Gym – Cart-Pole Environment

The following code can be utilized in Jupyter Notebook environment to create and initialize the environment after OpenAI Gym has been installed.

```
import gym
from PIL import Image
env = gym.make("CartPole-v1", render_mode =
"rgb_array")
obs = env.reset()
obs
```

For cart-pole environment, each observation is a 1D array with 4 floats.

Carts horizontal position	:	Where, the value 0.0 means center
Velocity	:	A positive value means right
Angle of the pole	:	Where, the value of 0.0 means vertical
Angular velocity of the pole	:	A positive value means clockwise

OpenAI gym – Policy Search on Cart-pole Problem

A simple policy can be hardcoded to accelerate left when the pole leans left and accelerate right when the pole leans right.

This policy can be executed for a reasonable duration, such as 500 episodes, with each episode comprising 200 steps.

Even with 500 attempts, this policy would **not** be able to keep the pole upright for 100 timesteps which makes it not a great policy.

OpenAI gym – Policy Search on Cart-pole Problem

To enhance the policy, deep reinforcement learning is to be used

For policy enhancement,

- Move away from a hardcoded pole policy.
- Adopt a deep neural network for policy formulation.
- Input observations, output corresponding actions.

Key Takeaways

- Policy search focuses on identifying the optimal strategy (or policy) for an agent in an environment, aiming to maximize its reward.
- The key steps for a policy search algorithm include defining, evaluating, and improving policies.
- Gym addresses standardization issues in research papers, offering diverse and easily configurable environments to enhance benchmarking.

Knowledge Check

Knowledge Check

1

In reinforcement learning, what is the main goal of a policy search algorithm?

A. Minimizing the state space.

B. Maximizing the reward function.

C. Learning optimal hyperparameters.

D. Discovering an effective strategy for the agent.

Knowledge Check 1

In reinforcement learning, what is the main goal of a policy search algorithm?

A. Minimizing the state space.

B. Maximizing the reward function.

C. Learning optimal hyperparameters.

D. Discovering an effective strategy for the agent.

The correct answer is **D**

The main goal of a policy search algorithm is to discover an effective strategy for the agent.

Knowledge Check 2

What role does the reward signal play in the context of policy search algorithms?

A. It guides the agent's exploration.

B. It determines the initial policy.

C. It limits the agent's action space.

D. It has no influence on policy search.

Knowledge Check 2

What role does the reward signal play in the context of policy search algorithms?

A. It guides the agent's exploration.

B. It determines the initial policy.

C. It limits the agent's action space.

D. It has no influence on policy search.

The correct answer is **A**

The reward signal guides the agent’s exploration in policy search algorithms

Knowledge Check 3

In the OpenAI Gym cart-pole environment, what are the key components that an agent needs to consider for successful task completion?

A. The color of the cart and the pole.

B. The time of day in the virtual environment.

C. The cart's position and velocity, and the pole's angle and rotation rate.

D. The number of episodes completed by the agent.

Knowledge Check 3

In the OpenAI Gym cart-pole environment, what are the key components that an agent needs to consider for successful task completion?

A. The color of the cart and the pole.

B. The time of day in the virtual environment.

C. The cart's position and velocity, and the pole's angle and rotation rate.

D. The number of episodes completed by the agent.

The correct answer is **A**

In the OpenAI Gym cart-pole environment, the key components to be considered are the cart's position and velocity, and the pole's angle and rotation rate.

Thank You