Московский государственный технический университет им. Н.Э. Баумана
Кафедра «Системы обработки информации и управления»

Лабораторная работа №7
по дисциплине
«Методы машинного обучения»
на тему

«Алгоритмы Actor-Critic.»

Выполнил:
студент группы ИУ5-22М
Лун Сыхань

Москва — 2024г.

1. Цель лабораторной работы

Ознакомление с базовыми методами обучения с подкреплением на основе алгоритмов Actor-Critic.

2. Задание

Реализуйте любой алгоритм семейства Actor-Critic для произвольной среды.

3. Ход выполнения работы

```
import gym
import numpy as np
import tensorflow as tf
from tensorflow.keras.layers import Dense
from tensorflow.keras.optimizers import Adam
```

```
def train(env, actor_critic, optimizer, gamma=0.99, max_episodes=1000):
    for episode in range(max_episodes):
        state = env.reset()
        episode_reward = 0

        # 在每个episode开始时创建一个新的GradientTape
        with tf.GradientTape() as tape:
            while True:
                state_tensor = tf.expand_dims(tf.convert_to_tensor(state), 0)
                logits, value = actor_critic(state_tensor)
                action_probs = tf.nn.softmax(logits)
                action = np.random.choice(env.action_space.n, p=np.squeeze(action_pro
                next_state, reward, done, _ = env.step(action)
                episode_reward += reward

                next_state_tensor = tf.expand_dims(tf.convert_to_tensor(next_state),
                _, next_value = actor_critic(next_state_tensor)
                td_target = reward + gamma * next_value * (1 - done)
                td_error = td_target - value

                # 计算actor和critic损失
                action_prob = tf.gather(action_probs[0], action)
                actor_loss = -tf.math.log(action_prob) * td_error
                critic_loss = td_error ** 2
                total_loss = actor_loss + critic_loss

                if done:
                    break
                else:
                    state = next_state

        # 计算梯度并更新参数
        grads = tape.gradient(total_loss, actor_critic.trainable_variables)
        optimizer.apply_gradients(zip(grads, actor_critic.trainable_variables))

        if episode % 10 == 0:
            print("Episode {}: Total Reward = {}".format(episode, episode_reward))

# 初始化环境和Actor-Critic模型
env = gym.make('CartPole-v1')
num_actions = env.action_space.n
actor_critic = ActorCritic(num_actions)
optimizer = Adam(learning_rate=0.01)

# 训练Actor-Critic模型
train(env, actor_critic, optimizer)
```

```
Episode 0: Total Reward = 31.0
Episode 10: Total Reward = 37.0
Episode 20: Total Reward = 33.0
Episode 30: Total Reward = 33.0
Episode 40: Total Reward = 19.0
Episode 50: Total Reward = 11.0
Episode 60: Total Reward = 9.0
Episode 70: Total Reward = 14.0
Episode 80: Total Reward = 19.0
Episode 90: Total Reward = 36.0
Episode 100: Total Reward = 18.0
Episode 110: Total Reward = 11.0
Episode 120: Total Reward = 14.0
Episode 130: Total Reward = 13.0
Episode 140: Total Reward = 11.0
Episode 150: Total Reward = 12.0
Episode 160: Total Reward = 10.0
Episode 170: Total Reward = 14.0
Episode 180: Total Reward = 9.0
Episode 190: Total Reward = 11.0
Episode 200: Total Reward = 13.0
Episode 210: Total Reward = 17.0
Episode 220: Total Reward = 11.0
Episode 230: Total Reward = 18.0
Episode 240: Total Reward = 9.0
Episode 250: Total Reward = 15.0
Episode 260: Total Reward = 16.0
Episode 270: Total Reward = 10.0
Episode 280: Total Reward = 11.0
Episode 290: Total Reward = 9.0
Episode 300: Total Reward = 16.0
Episode 310: Total Reward = 14.0
Episode 320: Total Reward = 8.0
Episode 330: Total Reward = 12.0
Episode 340: Total Reward = 21.0
Episode 350: Total Reward = 16.0
Episode 360: Total Reward = 10.0
Episode 370: Total Reward = 8.0
Episode 380: Total Reward = 12.0
Episode 390: Total Reward = 14.0
Episode 400: Total Reward = 20.0
Episode 410: Total Reward = 12.0
Episode 420: Total Reward = 10.0
Episode 430: Total Reward = 13.0
Episode 440: Total Reward = 10.0
Episode 450: Total Reward = 15.0
Episode 460: Total Reward = 13.0
Episode 470: Total Reward = 23.0
Episode 480: Total Reward = 21.0
Episode 490: Total Reward = 14.0
Episode 500: Total Reward = 10.0
Episode 510: Total Reward = 20.0
Episode 520: Total Reward = 21.0
Episode 530: Total Reward = 23.0
Episode 540: Total Reward = 18.0
Episode 550: Total Reward = 10.0
Episode 560: Total Reward = 13.0
Episode 570: Total Reward = 9.0
Episode 580: Total Reward = 18.0
Episode 590: Total Reward = 16.0
Episode 600: Total Reward = 10.0
Episode 610: Total Reward = 9.0
Episode 620: Total Reward = 12.0
Episode 630: Total Reward = 19.0
Episode 640: Total Reward = 13.0
Episode 650: Total Reward = 12.0
Episode 660: Total Reward = 10.0
Episode 670: Total Reward = 14.0
Episode 680: Total Reward = 11.0
Episode 690: Total Reward = 9.0
Episode 700: Total Reward = 12.0
Episode 710: Total Reward = 14.0
Episode 720: Total Reward = 8.0
Episode 730: Total Reward = 12.0
Episode 740: Total Reward = 9.0
Episode 750: Total Reward = 10.0
Episode 760: Total Reward = 12.0
Episode 770: Total Reward = 8.0
Episode 780: Total Reward = 10.0
Episode 790: Total Reward = 9.0
Episode 800: Total Reward = 11.0
Episode 810: Total Reward = 10.0
Episode 820: Total Reward = 9.0
Episode 830: Total Reward = 8.0
Episode 840: Total Reward = 10.0
Episode 850: Total Reward = 10.0
Episode 860: Total Reward = 11.0
Episode 870: Total Reward = 9.0
Episode 880: Total Reward = 9.0
Episode 890: Total Reward = 9.0
Episode 900: Total Reward = 12.0
Episode 910: Total Reward = 10.0
Episode 920: Total Reward = 11.0
Episode 930: Total Reward = 11.0
Episode 940: Total Reward = 10.0
Episode 950: Total Reward = 9.0
Episode 960: Total Reward = 12.0
Episode 970: Total Reward = 9.0
Episode 980: Total Reward = 10.0
Episode 990: Total Reward = 8.0
```