



Министерство науки и высшего образования  
Российской Федерации  
Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
"Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)"  
(МГТУ им. Н.Э. Баумана)

---

ФАКУЛЬТЕТ ИНФОРМАТИКА, ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА \_\_\_\_\_СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ (ИУ5)\_\_\_\_\_

## ОТЧЕТ

### Лабораторная работа №7 «Алгоритмы Actor-Critic»

по курсу «Методы машинного обучения»

ИСПОЛНИТЕЛЬ:

группа ИУ5-21М

Рабцевич К. Р.

ФИО

\_\_\_\_\_

подпись

"\_\_" \_\_\_\_\_ 2023 г.

ПРЕПОДАВАТЕЛЬ:

Гапанюк Ю.Е.

ФИО

\_\_\_\_\_

подпись

"\_\_" \_\_\_\_\_ 2023 г.

---

Москва - 2023

## 1. Задание

- Реализуйте любой алгоритм семейства Actor-Critic для произвольной среды.

## 2. Текст программы

```
1 #!/usr/bin/env python
2
3 import gymnasium as gym
4 import numpy as np
5 from itertools import count
6 from collections import namedtuple
7
8 import torch
9 import torch.nn as nn
10 import torch.nn.functional as F
11 import torch.optim as optim
12 from torch.distributions import Categorical
13
14 # Cart Pole
15 CONST_ENV_NAME = 'Acrobot-v1'
16 env = gym.make(CONST_ENV_NAME)
17 GAMMA = 0.99
18 SavedAction = namedtuple('SavedAction', ['log_prob', 'value'])
19
20 class Policy(nn.Module):
21     def __init__(self):
22         super(Policy, self).__init__()
23         self.affine1 = nn.Linear(6, 128)
24
25         # actor's layer
26         self.action_head = nn.Linear(128, 3)
27
28         # critic's layer
29         self.value_head = nn.Linear(128, 1)
30
31         # action & reward buffer
32         self.saved_actions = []
33         self.rewards = []
34
35     def forward(self, x):
36         x = F.relu(self.affine1(x))
37
38         # actor: choses action to take from state s_t
39         # by returning probability of each action
40         action_prob = F.softmax(self.action_head(x), dim=-1)
41
42         # critic: evaluates being in the state s_t
43         state_values = self.value_head(x)
44
45         # return values for both actor and critic as a tuple of 2 values:
46         # 1. a list with the probability of each action over the action space
47         # 2. the value from state s_t
48         return action_prob, state_values
49
50 model = Policy()
51 optimizer = optim.AdamW(model.parameters(), lr=1e-3)
52 eps = np.finfo(np.float32).eps.item()
53
```

```

54 def select_action(state):
55     state = torch.from_numpy(state).float()
56     probs, state_value = model(state)
57
58     # create a categorical distribution over the list of probabilities of actions
59     m = Categorical(probs)
60
61     # and sample an action using the distribution
62     action = m.sample()
63
64     # save to action buffer
65     model.saved_actions.append(SavedAction(m.log_prob(action), state_value))
66
67     # the action to take (left or right)
68     return action.item()
69
70 def finish_episode():
71     """
72     Training code. Calculates actor and critic loss and performs backprop.
73     """
74     R = 0
75     saved_actions = model.saved_actions
76     policy_losses = [] # list to save actor (policy) loss
77     value_losses = [] # list to save critic (value) loss
78     returns = [] # list to save the true values
79
80     # calculate the true value using rewards returned from the environment
81     for r in model.rewards[::-1]:
82         # calculate the discounted value
83         R = r + GAMMA * R
84         returns.insert(0, R)
85
86     returns = torch.tensor(returns)
87     returns = (returns - returns.mean()) / (returns.std() + eps)
88
89     for (log_prob, value), R in zip(saved_actions, returns):
90         advantage = R - value.item()
91
92         # calculate actor (policy) loss
93         policy_losses.append(-log_prob * advantage)
94
95         # calculate critic (value) loss using L1 smooth loss
96         value_losses.append(F.smooth_l1_loss(value, torch.tensor([R])))
97
98     # reset gradients
99     optimizer.zero_grad()
100
101     # sum up all the values of policy_losses and value_losses
102     loss = torch.stack(policy_losses).sum() + torch.stack(value_losses).sum()
103
104     # perform backprop
105     loss.backward()
106     optimizer.step()
107
108     # reset rewards and action buffer
109     del model.rewards[:]
110     del model.saved_actions[:]
111
112 def main():
113     running_reward = -500
114

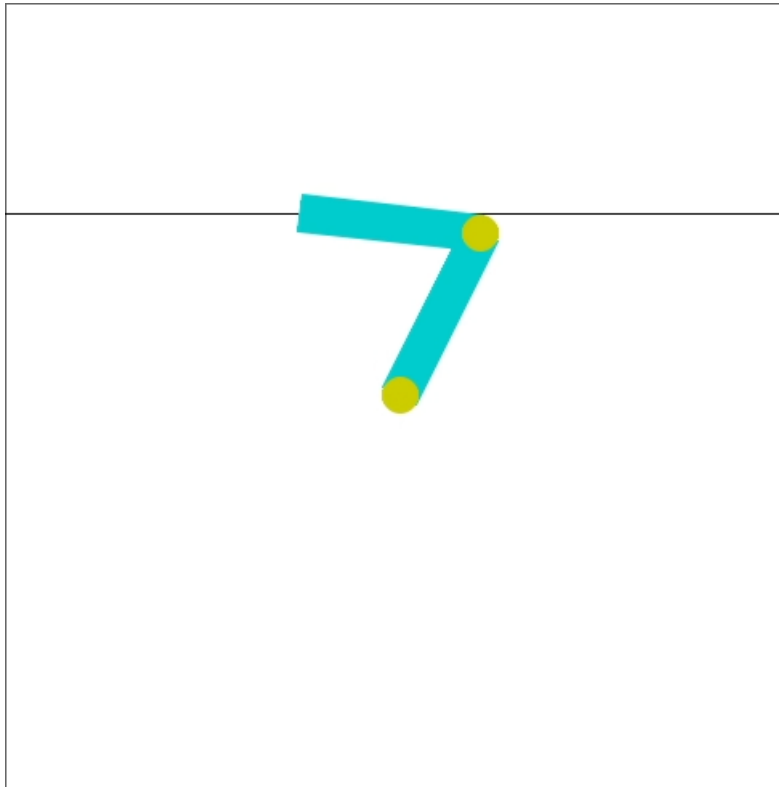
```

```

115 # run infinitely many episodes
116 for i_episode in count(1):
117     #print(running_reward)
118     # reset environment and episode reward
119     state, _ = env.reset()
120     ep_reward = 0
121
122     # for each episode, only run 9999 steps so that we don't
123     # infinite loop while learning
124     for t in range(1, 9999):
125         # select action from policy
126         action = select_action(state)
127
128         # take the action
129         state, reward, done, truncated, _ = env.step(action)
130
131         model.rewards.append(reward)
132         ep_reward += reward
133         if done or truncated:
134             break
135
136     print(ep_reward)
137     # update cumulative reward
138     running_reward = 0.05 * ep_reward + (1 - 0.05) * running_reward
139
140     # perform backprop
141     finish_episode()
142
143     # log results
144     if i_episode % 10 == 0:
145         print(f"Episode {i_episode}\tLast reward: {ep_reward:.2f}\tAverage reward:
146             {running_reward:.2f}")
147
148     # check if we have "solved" the cart pole problem
149     if running_reward > env.spec.reward_threshold*2:
150         print(f"Solved! Running reward is now {running_reward} and the last episode runs to {t}
151             time steps!")
152         break
153
154     env2 = gym.make(CONST_ENV_NAME, render_mode='human')
155
156     # reset environment and episode reward
157     state, _ = env2.reset()
158     ep_reward = 0
159
160     # for each episode, only run 9999 steps so that we don't
161     # infinite loop while learning
162     for t in range(1, 10000):
163         # select action from policy
164         action = select_action(state)
165         # take the action
166         state, reward, done, _, _ = env2.step(action)
167         model.rewards.append(reward)
168         ep_reward += reward
169         if done:
170             break
171
172 if __name__ == '__main__':
173     main()

```

### 3. Экранные формы с примерами выполнения программы



```
1 -500.0
2 -500.0
3 -500.0
4 -500.0
5 -500.0
6 -500.0
7 -500.0
8 -500.0
9 -500.0
10 -500.0
11 Episode 10      Last reward: -500.00    Average reward: -500.00
12 -500.0
13 -500.0
14 -500.0
15 -500.0
16 -500.0
17 -500.0
18 -500.0
19 -500.0
20 -500.0
21 -500.0
22 Episode 20      Last reward: -500.00    Average reward: -500.00
23 -500.0
24 -500.0
25 -500.0
26 -500.0
27 -500.0
28 -500.0
29 -500.0
30 -410.0
31 -500.0
32 -292.0
```

|               |                      |                         |
|---------------|----------------------|-------------------------|
| 33 Episode 30 | Last reward: -292.00 | Average reward: -485.54 |
| 34 -232.0     |                      |                         |
| 35 -335.0     |                      |                         |
| 36 -500.0     |                      |                         |
| 37 -500.0     |                      |                         |
| 38 -500.0     |                      |                         |
| 39 -500.0     |                      |                         |
| 40 -500.0     |                      |                         |
| 41 -500.0     |                      |                         |
| 42 -461.0     |                      |                         |
| 43 -500.0     |                      |                         |
| 44 Episode 40 | Last reward: -500.00 | Average reward: -475.57 |
| 45 -500.0     |                      |                         |
| 46 -500.0     |                      |                         |
| 47 -500.0     |                      |                         |
| 48 -424.0     |                      |                         |
| 49 -362.0     |                      |                         |
| 50 -391.0     |                      |                         |
| 51 -350.0     |                      |                         |
| 52 -500.0     |                      |                         |
| 53 -257.0     |                      |                         |
| 54 -500.0     |                      |                         |
| 55 Episode 50 | Last reward: -500.00 | Average reward: -454.83 |
| 56 -500.0     |                      |                         |
| 57 -384.0     |                      |                         |
| 58 -500.0     |                      |                         |
| 59 -481.0     |                      |                         |
| 60 -500.0     |                      |                         |
| 61 -292.0     |                      |                         |
| 62 -500.0     |                      |                         |
| 63 -342.0     |                      |                         |
| 64 -428.0     |                      |                         |
| 65 -468.0     |                      |                         |
| 66 Episode 60 | Last reward: -468.00 | Average reward: -447.79 |
| 67 -500.0     |                      |                         |
| 68 -329.0     |                      |                         |
| 69 -472.0     |                      |                         |
| 70 -500.0     |                      |                         |
| 71 -412.0     |                      |                         |
| 72 -299.0     |                      |                         |
| 73 -500.0     |                      |                         |
| 74 -500.0     |                      |                         |
| 75 -284.0     |                      |                         |
| 76 -455.0     |                      |                         |
| 77 Episode 70 | Last reward: -455.00 | Average reward: -437.99 |
| 78 -241.0     |                      |                         |
| 79 -333.0     |                      |                         |
| 80 -500.0     |                      |                         |
| 81 -315.0     |                      |                         |
| 82 -339.0     |                      |                         |
| 83 -292.0     |                      |                         |
| 84 -258.0     |                      |                         |
| 85 -408.0     |                      |                         |
| 86 -273.0     |                      |                         |
| 87 -297.0     |                      |                         |
| 88 Episode 80 | Last reward: -297.00 | Average reward: -392.21 |
| 89 -451.0     |                      |                         |
| 90 -387.0     |                      |                         |
| 91 -268.0     |                      |                         |
| 92 -317.0     |                      |                         |
| 93 -255.0     |                      |                         |

|     |             |                      |                         |
|-----|-------------|----------------------|-------------------------|
| 94  | -223.0      |                      |                         |
| 95  | -352.0      |                      |                         |
| 96  | -267.0      |                      |                         |
| 97  | -258.0      |                      |                         |
| 98  | -242.0      |                      |                         |
| 99  | Episode 90  | Last reward: -242.00 | Average reward: -353.33 |
| 100 | -208.0      |                      |                         |
| 101 | -273.0      |                      |                         |
| 102 | -283.0      |                      |                         |
| 103 | -165.0      |                      |                         |
| 104 | -231.0      |                      |                         |
| 105 | -195.0      |                      |                         |
| 106 | -237.0      |                      |                         |
| 107 | -306.0      |                      |                         |
| 108 | -213.0      |                      |                         |
| 109 | -267.0      |                      |                         |
| 110 | Episode 100 | Last reward: -267.00 | Average reward: -307.42 |
| 111 | -242.0      |                      |                         |
| 112 | -149.0      |                      |                         |
| 113 | -236.0      |                      |                         |
| 114 | -258.0      |                      |                         |
| 115 | -196.0      |                      |                         |
| 116 | -204.0      |                      |                         |
| 117 | -152.0      |                      |                         |
| 118 | -366.0      |                      |                         |
| 119 | -251.0      |                      |                         |
| 120 | -285.0      |                      |                         |
| 121 | Episode 110 | Last reward: -285.00 | Average reward: -279.45 |
| 122 | -210.0      |                      |                         |
| 123 | -143.0      |                      |                         |
| 124 | -185.0      |                      |                         |
| 125 | -231.0      |                      |                         |
| 126 | -142.0      |                      |                         |
| 127 | -253.0      |                      |                         |
| 128 | -251.0      |                      |                         |
| 129 | -322.0      |                      |                         |
| 130 | -160.0      |                      |                         |
| 131 | -162.0      |                      |                         |
| 132 | Episode 120 | Last reward: -162.00 | Average reward: -250.42 |
| 133 | -313.0      |                      |                         |
| 134 | -205.0      |                      |                         |
| 135 | -186.0      |                      |                         |
| 136 | -162.0      |                      |                         |
| 137 | -186.0      |                      |                         |
| 138 | -186.0      |                      |                         |
| 139 | -217.0      |                      |                         |
| 140 | -124.0      |                      |                         |
| 141 | -171.0      |                      |                         |
| 142 | -208.0      |                      |                         |
| 143 | Episode 130 | Last reward: -208.00 | Average reward: -227.24 |
| 144 | -214.0      |                      |                         |
| 145 | -207.0      |                      |                         |
| 146 | -149.0      |                      |                         |
| 147 | -125.0      |                      |                         |
| 148 | -245.0      |                      |                         |
| 149 | -204.0      |                      |                         |
| 150 | -175.0      |                      |                         |
| 151 | -243.0      |                      |                         |
| 152 | -135.0      |                      |                         |
| 153 | -459.0      |                      |                         |
| 154 | Episode 140 | Last reward: -459.00 | Average reward: -225.08 |

155 -175.0  
156 -163.0  
157 -212.0  
158 -177.0  
159 -166.0  
160 -157.0  
161 -175.0  
162 -140.0  
163 -156.0  
164 -197.0  
165 Episode 150      Last reward: -197.00      Average reward: -203.49  
166 -241.0  
167 -193.0  
168 -154.0  
169 -156.0  
170 Solved! Running reward is now -199.8982282342606 and the last episode runs to 157 time steps!