目录 1

# 目录

1	模版备用	2
2	Q learning 算法	3
3	Sarsa 算法	4
4	Policy Gradient 算法	5
5	DQN 算法	6
6	SoftQ 算法	7
7	SAC-S 算法	8
8	SAC 算法	9

模版备用 2

## 1 模版备用

算法	
开14	
1: 测试	

## 2 Q learning 算法

#### Q-learning 算法<sup>1</sup>

- 1: 初始化 Q 表 Q(s,a) 为任意值,但其中  $Q(s_{terminal},)=0$ ,即终止状态对应的 Q 值为 0
- 2: **for** 回合数 = 1, M **do**
- 3: 重置环境,获得初始状态  $s_1$
- 4: **for** 时步 = 1, t **do**
- 5: 根据  $\varepsilon$  greedy 策略采样动作  $a_t$
- 6: 环境根据  $a_t$  反馈奖励  $r_t$  和下一个状态  $s_{t+1}$
- 7: 更新策略:
- 8:  $Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha [r_t + \gamma \max_a Q(s_{t+1}, a) Q(s_t, a_t)]$
- 9: 更新状态  $s_{t+1} \leftarrow s_t$
- 10: end for
- 11: end for

 $<sup>^1\</sup>mathrm{Reinforcement}$  Learning: An Introduction

SARSA 算法 4

## 3 Sarsa 算法

#### Sarsa 算法<sup>1</sup>

- 1: 初始化 Q 表 Q(s,a) 为任意值,但其中  $Q(s_{terminal},)=0$ ,即终止状态 对应的 Q 值为 0
- 2: **for** 回合数 = 1, M **do**
- 3: 重置环境,获得初始状态  $s_1$
- 4: 根据  $\varepsilon$  greedy 策略采样初始动作  $a_1$
- 5: for 时步 = 1, t do
- 6: 环境根据  $a_t$  反馈奖励  $r_t$  和下一个状态  $s_{t+1}$
- 7: 根据  $\varepsilon$  greedy 策略  $s_{t+1}$  和采样动作  $a_{t+1}$
- 8: 更新策略:
- 9:  $Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha [r_t + \gamma Q(s_{t+1}, a_{t+1}) Q(s_t, a_t)]$
- 10: 更新状态  $s_{t+1} \leftarrow s_t$
- 11: 更新动作  $a_{t+1} \leftarrow a_t$
- 12: end for
- 13: end for

<sup>&</sup>lt;sup>1</sup>Reinforcement Learning: An Introduction

## 4 Policy Gradient 算法

## REINFORCE 算法: Monte-Carlo Policy Gradient<sup>1</sup>

- 1: 初始化策略参数  $\boldsymbol{\theta} \in \mathbb{R}^{d'}$ ( e.g., to  $\boldsymbol{0}$ )
- 2: for 回合数 = 1, M do
- 根据策略  $\pi(\cdot \mid \cdot, \boldsymbol{\theta})$  采样一个 (或几个) 回合的 transition
- 4:
- 5:
- for 时步 = 1, t do 计算回报  $G \leftarrow \sum_{k=t+1}^{T} \gamma^{k-t-1} R_k$ 更新策略  $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \alpha \gamma^t G \nabla \ln \pi \left( A_t \mid S_t, \boldsymbol{\theta} \right)$ 6:
- end for
- 8: end for

<sup>&</sup>lt;sup>1</sup>Reinforcement Learning: An Introduction

## 5 DQN 算法

6

#### DQN 算法<sup>1</sup>

- 1: 初始化策略网络参数  $\theta$
- 2: 复制参数到目标网络  $\hat{Q} \leftarrow Q$
- 3: 初始化经验回放 D
- 4: **for** 回合数 = 1, M **do**
- 5: 重置环境,获得初始状态 st
- 6: **for** 时步 = 1, t **do**
- 7: 根据  $\varepsilon$  greedy 策略采样动作  $a_t$
- 8: 环境根据  $a_t$  反馈奖励  $r_t$  和下一个状态  $s_{t+1}$
- 9: 存储 transition 即  $(s_t, a_t, r_t, s_{t+1})$  到经验回放 D 中
- 10: 更新环境状态  $s_{t+1} \leftarrow s_t$
- 11: 更新策略:
- 12: 从 D 中采样一个 batch 的 transition
- 13: 计算实际的 Q 值, 即  $y_i^2$
- 14: 对损失  $L(\theta) = (y_i Q(s_i, a_i; \theta))^2$  关于参数  $\theta$  做随机梯度下降<sup>3</sup>
- 15: end for
- 16: 每 C 个回合复制参数  $\hat{Q} \leftarrow Q^4$
- 17: end for

<sup>&</sup>lt;sup>1</sup>Playing Atari with Deep Reinforcement Learning

 $<sup>^{2}</sup>y_{i} = \begin{cases} r_{i} &$  对于终止状态 $s_{i+1} \\ r_{i} + \gamma \max_{a'} Q\left(s_{i+1}, a'; \theta\right) \end{cases}$  对于非终止状态 $s_{i+1}$ 

 $<sup>^{3}\</sup>theta_{i}\leftarrow\theta_{i}-\lambda\nabla_{\theta_{i}}L_{i}\left(\theta_{i}\right)$ 

 $<sup>^4</sup>$ 此处也可像原论文中放到小循环中改成每  $^{C}$  步,但没有每  $^{C}$  个回合稳定

SOFTQ 算法 7

#### 6 SoftQ 算法

#### SoftQ 算法

```
1: 初始化参数 \theta 和 \phi
 2: 复制参数 \theta \leftarrow \theta, \phi \leftarrow \phi
 3: 初始化经验回放 D
 4: for 回合数 = 1, M do
          for 时步 = 1, t do
 5:
              根据 \mathbf{a}_t \leftarrow f^{\phi}(\xi; \mathbf{s}_t) 采样动作,其中 \xi \sim \mathcal{N}(\mathbf{0}, \mathbf{I})
 6:
              环境根据 a_t 反馈奖励 s_t 和下一个状态 s_{t+1}
 7:
              存储 transition 即 (s_t, a_t, r_t, s_{t+1}) 到经验回放 D 中
 8:
              更新环境状态 s_{t+1} \leftarrow s_t
 9:
              更新 soft Q 函数参数:
10:
             对于每个 s_{t+1}^{(i)} 采样 \left\{\mathbf{a}^{(i,j)}\right\}_{j=0}^{M} \sim q_{\mathbf{a}'}
11:
             计算 empirical soft values V_{\text{soft}}^{\theta}(\mathbf{s}_t)^1
12:
              计算 empirical gradient J_Q(\theta)^2
13:
              根据 J_Q(\theta) 使用 ADAM 更新参数 \theta
14:
              更新策略:
15:
             对于每个 s_t^{(i)} 采样 \left\{\xi^{(i,j)}\right\}_{j=0}^{M} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) 计算 \mathbf{a}_t^{(i,j)} = f^{\phi}\left(\xi^{(i,j)}, \mathbf{s}_t^{(i)}\right)
16:
17:
              使用经验估计计算 \Delta f^{\phi}(\cdot; \mathbf{s}_t)^3
18:
              计算经验估计 \frac{\partial J_{\pi}(\phi;\mathbf{s}_t)}{\partial \phi} \propto \mathbb{E}_{\xi} \left[ \Delta f^{\phi}(\xi;\mathbf{s}_t) \frac{\partial f^{\phi}(\xi;\mathbf{s}_t)}{\partial \phi} \right], 即 \hat{\nabla}_{\phi} J_{\pi}
19:
              根据 \hat{\nabla}_{\phi}J_{\pi} 使用 ADAM 更新参数 \phi
20:
21:
          end for
22:
          每 C 个回合复制参数 \bar{\theta} \leftarrow \theta, \bar{\phi} \leftarrow \phi
23:
24: end for
```

$$\frac{1}{V_{\text{soft}}^{\theta}\left(\mathbf{s}_{t}\right) = \alpha \log \mathbb{E}_{\mathbf{q}_{\mathbf{a}'}}\left[\frac{\exp\left(\frac{1}{\alpha}Q_{\text{soft}}^{\theta}\left(\mathbf{s}_{t},\mathbf{a}'\right)\right)}{q_{\mathbf{a}'}(\mathbf{a}')}\right]$$

$$\frac{2}{J_{Q}(\theta)} = \mathbb{E}_{\mathbf{s}_{t} \sim q_{\mathbf{s}_{t}}, \mathbf{a}_{t} \sim q_{\mathbf{a}_{t}}}\left[\frac{1}{2}\left(\hat{Q}_{\text{soft}}^{\bar{\theta}}\left(\mathbf{s}_{t}, \mathbf{a}_{t}\right) - Q_{\text{soft}}^{\theta}\left(\mathbf{s}_{t}, \mathbf{a}_{t}\right)\right)^{2}\right]$$

$$\frac{\Delta f^{\phi}\left(\cdot; \mathbf{s}_{t}\right) = \mathbb{E}_{\mathbf{a}_{t} \sim \pi^{\phi}}\left[\kappa\left(\mathbf{a}_{t}, f^{\phi}\left(\cdot; \mathbf{s}_{t}\right)\right) \nabla_{\mathbf{a}'}Q_{\text{soft}}^{\theta}\left(\mathbf{s}_{t}, \mathbf{a}'\right)\right|_{\mathbf{a}' = \mathbf{a}_{t}}$$

$$+ \alpha \nabla_{\mathbf{a}'}\kappa\left(\mathbf{a}', f^{\phi}\left(\cdot; \mathbf{s}_{t}\right)\right)\Big|_{\mathbf{a}' = \mathbf{a}_{t}}\right]$$

SAC-S 算法 8

## 7 SAC-S 算法

#### SAC-S 算法<sup>1</sup>

```
1: 初始化参数 \psi, \bar{\psi}, \theta, \phi
         2: for 回合数 = 1, M do
                                                            for 时步 = 1, t do
         3:
                                                                                   根据 \mathbf{a}_t \sim \pi_{\phi}(\mathbf{a}_t \mid \mathbf{s}_t) 采样动作 a_t
         4:
                                                                                 环境 \mathbf{a}_t によった \mathbf{a}_t になった \mathbf{a}_t が \mathbf{a}_t が \mathbf{a}_t が \mathbf{a}_t が \mathbf{a}_t が \mathbf{a}_t を \mathbf{a}_t \mathbf{a}_t \mathbf{
        5:
         6:
                                                                                 更新环境状态 s_{t+1} \leftarrow s_t
         7:
         8:
                                                                                    更新策略:
                                                                                 \psi \leftarrow \psi - \lambda_V \hat{\nabla}_{\psi} J_V(\psi)
        9:
                                                                                 \theta_{i} \leftarrow \theta_{i} - \lambda_{Q} \hat{\nabla}_{\theta_{i}} J_{Q}(\theta_{i}) \text{ for } i \in \{1, 2\}

\phi \leftarrow \phi - \lambda_{\pi} \hat{\nabla}_{\phi} J_{\pi}(\phi) 

\bar{\psi} \leftarrow \tau \psi + (1 - \tau) \bar{\psi}

11:
12:
                                                            end for
13:
14: end for
```

 $<sup>^1{\</sup>rm Soft}$  Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor

SAC 算法 9

## 8 SAC 算法

#### SAC 算法<sup>1</sup>

```
1: 初始化网络参数 \theta_1, \theta_2 以及 \phi
 2: 复制参数到目标网络 \bar{\theta_1} \leftarrow \theta_1, \bar{\theta_2} \leftarrow \theta_2,
 3: 初始化经验回放 D
 4: for 回合数 = 1, M do
         重置环境,获得初始状态 st
 5:
         for 时步 = 1, t do
 6:
             根据 \mathbf{a}_t \sim \pi_{\phi}(\mathbf{a}_t \mid \mathbf{s}_t) 采样动作 a_t
 7:
             环境反馈奖励和下一个状态,\mathbf{s}_{t+1} \sim p(\mathbf{s}_{t+1} \mid \mathbf{s}_t, \mathbf{a}_t)
 8:
             存储 transition 到经验回放中,\mathcal{D} \leftarrow \mathcal{D} \cup \{(\mathbf{s}_t, \mathbf{a}_t, r(\mathbf{s}_t, \mathbf{a}_t), \mathbf{s}_{t+1})\}
 9:
             更新环境状态 s_{t+1} \leftarrow s_t
10:
11:
             更新策略:
             更新 Q 函数,\theta_i \leftarrow \theta_i - \lambda_Q \hat{\nabla}_{\theta_i} J_Q(\theta_i) for i \in \{1, 2\}^{23}
12:
             更新策略权重, \phi \leftarrow \phi - \lambda_{\pi} \hat{\nabla}_{\phi} J_{\pi}(\phi)^{4}
13:
             调整 temperature, \alpha \leftarrow \alpha - \lambda \hat{\nabla}_{\alpha} J(\alpha)^{5}
14:
             更新目标网络权重, \bar{\theta}_i \leftarrow \tau \theta_i + (1-\tau)\bar{\theta}_i for i \in \{1,2\}
15:
         end for
16:
17: end for
```

<sup>&</sup>lt;sup>2</sup>Soft Actor-Critic Algorithms and Applications  ${}^{2}J_{Q}(\theta) = \mathbb{E}_{(\mathbf{s}_{t}, \mathbf{a}_{t}) \sim \mathcal{D}} \left[ \frac{1}{2} \left( Q_{\theta} \left( \mathbf{s}_{t}, \mathbf{a}_{t} \right) - \left( r \left( \mathbf{s}_{t}, \mathbf{a}_{t} \right) + \gamma \mathbb{E}_{\mathbf{s}_{t+1} \sim p} \left[ V_{\bar{\theta}} \left( \mathbf{s}_{t+1} \right) \right] \right) \right)^{2} \right]$   ${}^{3}\hat{\nabla}_{\theta}J_{Q}(\theta) = \nabla_{\theta}Q_{\theta} \left( \mathbf{a}_{t}, \mathbf{s}_{t} \right) \left( Q_{\theta} \left( \mathbf{s}_{t}, \mathbf{a}_{t} \right) - \left( r \left( \mathbf{s}_{t}, \mathbf{a}_{t} \right) + \gamma \left( Q_{\bar{\theta}} \left( \mathbf{s}_{t+1}, \mathbf{a}_{t+1} \right) - \alpha \log \left( \pi_{\phi} \left( \mathbf{a}_{t+1} \mid \mathbf{s}_{t+1} \right) \right) \right) \right)$   ${}^{4}\hat{\nabla}_{\phi}J_{\pi}(\phi) = \nabla_{\phi}\alpha \log \left( \pi_{\phi} \left( \mathbf{a}_{t} \mid \mathbf{s}_{t} \right) \right) + \left( \nabla_{\mathbf{a}_{t}}\alpha \log \left( \pi_{\phi} \left( \mathbf{a}_{t} \mid \mathbf{s}_{t} \right) \right) - \nabla_{\mathbf{a}_{t}}Q \left( \mathbf{s}_{t}, \mathbf{a}_{t} \right) \right) \nabla_{\phi}f_{\phi} \left( \epsilon_{t}; \mathbf{s}_{t} \right), \mathbf{a}_{t} = f_{\phi} \left( \epsilon_{t}; \mathbf{s}_{t} \right)$ 

 $<sup>^{5}</sup>J(\alpha) = \mathbb{E}_{\mathbf{a}_{t} \sim \pi_{t}} \left[ -\alpha \log \pi_{t} \left( \mathbf{a}_{t} \mid \mathbf{s}_{t} \right) - \alpha \overline{\mathcal{H}} \right]$