SAC 논문 구현

환경 구축

Lunar rander environment 불러오기

Replay buffer 제작

$[s_t, a_t, r_t, s_{t+1}]$을 저장할 수 있도록

모델 구현

- parameterized state value function $V_\psi\left(\mathbf{s}_t\right)$ # 8, 1
- soft Q-function $\mathbf{Q}_\theta\left(\mathbf{s}_t, \mathbf{a}_t\right)$, If continuous action space policy as a Gaussian with mean and covariance
- tractable policy $\pi_\phi\left(\mathbf{a}_t \mid \mathbf{s}_t\right)$

학습 코드 구현

Algorithm 1 Soft Actor-Critic

Initialize parameter vectors ψ, $\bar{\psi}$, θ, ϕ.
for each iteration **do**
  for each environment step **do**
    $\mathbf{a}_t \sim \pi_\phi(\mathbf{a}_t|\mathbf{s}_t)$
    $\mathbf{s}_{t+1} \sim p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)$
    $\mathcal{D} \leftarrow \mathcal{D} \cup \{(\mathbf{s}_t, \mathbf{a}_t, r(\mathbf{s}_t, \mathbf{a}_t), \mathbf{s}_{t+1})\}$
  end for
  for each gradient step **do**
    $\psi \leftarrow \psi - \lambda_V \hat{\nabla}_\psi J_V(\psi)$
    $\theta_i \leftarrow \theta_i - \lambda_Q \hat{\nabla}_{\theta_i} J_Q(\theta_i)$ for $i \in \{1, 2\}$
    $\phi \leftarrow \phi - \lambda_\pi \hat{\nabla}_\phi J_\pi(\phi)$
    $\bar{\psi} \leftarrow \tau\psi + (1 - \tau)\bar{\psi}$
  end for
end for

Loss function(objective function)

- Value function

  At below equation, actions are sampled according to the current policy, instead of the replay buffer

  $J_V(\psi) = \mathbb{E}_{\mathbf{s}_t \sim \mathcal{D}}\left[\frac{1}{2}\left(V_\psi(\mathbf{s}_t) - \mathbb{E}_{\mathbf{a}_t \sim \pi_\phi}\left[Q_\theta(\mathbf{s}_t, \mathbf{a}_t) - \log \pi_\phi(\mathbf{a}_t \mid \mathbf{s}_t)\right]\right)^2\right]$

  Target value network $V_{\bar{\psi}}$, we can update the target weights to match the current value function weights periodically

- Q function

  $J_Q(\theta) = \mathbb{E}_{(\mathbf{s}_t, \mathbf{a}_t) \sim \mathcal{D}}\left[\frac{1}{2}\left(Q_\theta(\mathbf{s}_t, \mathbf{a}_t) - \hat{Q}(\mathbf{s}_t, \mathbf{a}_t)\right)^2\right]$

  $where\ \hat{Q}(\mathbf{s}_t, \mathbf{a}_t) = r(\mathbf{s}_t, \mathbf{a}_t) + \gamma \mathbb{E}_{\mathbf{s}_{t+1} \sim p}\left[V_{\bar{\psi}}(\mathbf{s}_{t+1})\right]$

- Policy

  $J_\pi(\phi) = \mathbb{E}_{\mathbf{s}_t \sim \mathcal{D}}\left[\mathrm{D}_{\mathrm{KL}}\left(\pi_\phi(\cdot \mid \mathbf{s}_t) \,\|\, \frac{\exp(Q_\theta(\mathbf{s}_t, \cdot))}{Z_\theta(\mathbf{s}_t)}\right)\right]$

Hyperparameters

D. Hyperparameters

Table 1 lists the common SAC parameters used in the comparative evaluation in Figure 1 and Figure 4. Table 2 lists the reward scale parameter that was tuned for each environment.

Table 1. SAC Hyperparameters

Parameter	Value
Shared	
optimizer	Adam (Kingma & Ba, 2015)
learning rate	$3 \cdot 10^{-4}$
discount (γ)	0.99
replay buffer size	10^6
number of hidden layers (all networks)	2
number of hidden units per layer	256
number of samples per minibatch	256
nonlinearity	ReLU
SAC	
target smoothing coefficient (τ)	0.005
target update interval	1
gradient steps	1
SAC (hard target update)	
target smoothing coefficient (τ)	1
target update interval	1000
gradient steps (except humanoids)	4
gradient steps (humanoids)	1

테스트 코드 구현

n번 시행해서 평균 점수를 얻을 수 있도록

n번 시행의 비디오 저장(이어서)

Future work

- Use of two Q-functions to mitigate positive bias

  we parameterize two Q-functions, with parameters θ_i, and train them independently to optimize $J_Q(\theta_i)$. We then use the minimum of the Q-functions for the value gradient and policy gradient

- Continuous action space task에서 문제 해결
- 모델의 고도화 using LSTM(논문과 동일한 구조 사용)