

Neural Network 의 문제점

1. Correlations between sample

- 학습시키는 sample 값들이 서로 강하게 연관되는 문제
- sample 을 어떻게 주느냐에 따라 학습 모델이 다르게 만들어지는 것이 문제임!

2. Non-stationary targets

$$\min_{\theta} \sum_{t=0}^T \left[\overbrace{\hat{Q}(s_t, a_t)}^{\text{예측값 } Q_{\text{pred}}}(\theta) - (r_t + \gamma \max_{a'} \overbrace{\hat{Q}(s_{t+1}, a')}^{\text{타겟 } \gamma})(\theta) \right]^2$$

같은 네트워크를 사용한다

- 같은 네트워크를 사용하기 때문에 예측값-타겟의 값을 최소화하기 위해 예측값의 네트워크를 업데이트 하면 타겟의 네트워크도 (강제로) 업데이트 된다.
- 한마디로 **타겟은** 고정되어야 하는데 (예측 값이 네트워크를 업데이트 할 때마다) **계속 바뀐다!**
- 과녁이 자꾸 움직이는 거예요 $\pi\pi\pi$

해결법

1. **go deep** : 네트워크를 deep 하게 만들 것
2. **experience replay** : correlation between sample 해결법

```

Initialize replay memory  $\mathcal{D}$  to capacity  $N$ 
Initialize action-value function  $Q$  with random weights
for episode = 1,  $M$  do
  Initialize sequence  $s_1 = \{x_1\}$  and preprocessed sequenced  $\phi_1 = \phi(s_1)$ 
  for  $t = 1, T$  do
    With probability  $\epsilon$  select a random action  $a_t$ 
    otherwise select  $a_t = \max_a Q^*(\phi(s_t), a; \theta)$ 
    Execute action  $a_t$  in emulator and observe reward  $r_t$  and image  $x_{t+1}$ 
    Set  $s_{t+1} = s_t, a_t, x_{t+1}$  and preprocess  $\phi_{t+1} = \phi(s_{t+1})$ 
    Store transition  $(\phi_t, a_t, r_t, \phi_{t+1})$  in  $\mathcal{D}$ 
    Sample random minibatch of transitions  $(\phi_j, a_j, r_j, \phi_{j+1})$  from  $\mathcal{D}$ 
    Set  $y_j = \begin{cases} r_j & \text{for terminal } \phi_{j+1} \\ r_j + \gamma \max_{a'} Q(\phi_{j+1}, a'; \theta) & \text{for non-terminal } \phi_{j+1} \end{cases}$ 
    Perform a gradient descent step on  $(y_j - Q(\phi_j, a_j; \theta))^2$  according to equation 3
  and for
  
```

experience replay

- 각 state 에 따른 결과를 가지고 바로 학습시키지 않고 버퍼에 쌓아놓는다.
- 다 쌓인 버퍼에서 랜덤하게 꺼내 그걸로 학습시킨다

3. **Separate target network & copy network** : non stationary targets 해결법

$$\min_{\theta} \sum_{t=0}^T [\hat{Q}(s_t, a_t(\theta) - (r_t + \gamma \max_{a'} \hat{Q}(s_{t+1}, a(\bar{\theta})))^2]$$

네트워크를 분리한다

- 예측하는 네트워크와 타겟 네트워크를 분리해서 예측 네트워크만 업데이트시킨다
- 학습이 끝나면 예측 네트워크를 복사해서 타겟 네트워크에 붙혀 넣어 교체한다.