Week8 . 강화학습논문리뷰

<Playing Atari with Deep Reinforcement Learning> (DQN)

0. Abstract 초록	고차원 sensory에서 강화학습을 사용하여 제어정책 (control policies)을 학습하는 딥러닝 모델 제시.
	합성곱 (convolutional) 신경망으로 Q-learning을 train 된다.
	input은 raw pixels 이고 output은 미래 reward 추정하는 가치함수 (value function) (그림그대로 넣음)
	7개의 Atrari 2600 게임에 적용 했고 6개 게임이 이전접근을 능가하고 3개 게임에서 인간 전문가 능가함을 알아냄.
1. Introduction	현재 동향과 RL의 문제점 (해결점) 3가지, 이 논문에서 해결한 방법
	강화학습에 딥러닝 적용시 문제점 ① labelled training data 불필요 ② highly correlated states 필요 (independent data X) ③ 변화하는 데이터분포
	해결 : Q-learning 과 SGD로 학습, ① 이전 변화 무작위 추출로 correlated data와 non-stationary distributions 해결
2. Background	사용 스테이션 및 공식설명.
	(Stochastic) environment $\mathcal{E}$, action a_t, set of action A, reward r_t, π
	future reward maximize 하는 action 선택, action-value function $Q(s,a)$ (state, action) 사용
	최적의 action-value function은 Bellman 따름
	각 시퀀스 나누어서 추정하는 것 대신 근사값 사용 (function approximator)
	: $Q(s,a;\theta) \approx Q^*(s,a)$
	: θ weight는 Q-network로 학습 됨
	Loss function 구하기 위해 gradient 사용하는데 SGD 사용
	이과정거치고 나면 Q-learning algorithm 형태됨. (model-free, off-policy, greedy strategy)
3. Related Work	기존 모델들 소개 및 한계점, 다른점
	TD-gamma, NFQ, HyperNEAT
4. Deep Reinforcement Learnin	RL을 딥러닝에 적용 하기 위해 experience replay 사용했다. 에피소드 pool인 data-set D, replay memory를 저장한다. agent는 ε-greedy policy 따라 action 선택한다. function ϕ 사용
	이러한 Deep Q-learning 사용함으로써 3가지 이점이 생긴다
	First, each step of experience is potentially used in many weight updates, which allows for greater data efficiency.
	randomizing the samples breaks these correlations and therefore reduces the variance of the updates. Third, when learning on-policy the current parameters determine the next data sample that the parameters are trained on.
4-1 Processing and Model Architecture	input은 축소시키고 square로 만들어서 84x84로 전처리, nn 사용해서 Q parameterizing
	hidden layer 1 : convolves 16 8x8, stride 4, rectifier nonlinearity
	2 : 〃 32 4x4, stride 2, 〃
	3 : fully connected, consists of 256 rectifier units
	output layer : fully connected linear layer with a single output for each valid action (4-18)
	convolutional network는 DQN으로 train

5. Experiments	리워드 설정 +1, -1, 0 RMSProp 사용, ε=0.1 설정 및 train data 수 simple frame-skipping technique 사용했는데 agent는 모든 k번째 frame 마다 action 선택하고 매번 action은 skipped frames에서 반복.
5-1. Trainning and Stability	training을 periodically 하게 계산
5-2. Visualizing the Value function	
5-3. Main Evaluation	비교
6. Conclustion.	