Coding Deep Q learning | 18/08/20

Episode → change in point.

$$q^{new}(s,a) = (1-\alpha)\underbrace{q(s,a)}_{\text{old qvalue}} + \alpha\Big(\overbrace{R_{t+1} + \gamma \max_{a'} q(s',a')}^{\text{learned}}\Big)$$

Episode loop
- → reset env
- → ~~check wh~~ initialise flag for stopping
- → initialise reward

Time-step loop
- → take action based on exploitation, ~~exploitation~~ exploration
- → ~~comp~~ update q table
- → update reward
- → check if episode is compl.
- → add reward of curr episod to all rewards

Qlearning ≡ Using value iteration to compute optimal Q-value funcn.

Deep Q learning ≐ Use neural networks to approximate Q-value function

---

Experience replay

↓

we store the agent's experience at each time step in a DS called replay memory.

$$e_t = (s_t, a_t, r_{t+1}, s_{t+1})$$

Q Why we replay memory?

→ To break correlation.

Summary of steps

1. Initialise replay memory capacity
2. Initialise the network with random weights (Xavier initialisation)
3. For each episode
    1. Initialise the starting state
    2. For each time step
        1. Select an action
            → via exploration or exploitation
        2. Execute selected action in an emulator
        3. Observe reward and next state.
        4. Store experience in replay memory.

---

5. Sample random Batch from replay memory
6. Preprocess states from batch
7. Pass batch of pre-processed states to policy network
8. Calculate loss between target Q-values & Q-values
    - Requires second pass to the network for next state
9. Gradient descent updates weights in the policy network to minimize loss.

Target network

→ When calculating loss between present Q-values and target Q-values, we use the target network [a clone of the original network] to compute target Q-values.

→ The weights of target network are updated after every x - time steps where x is a hyperparameter

Coding deep Q learning | 18/03/20

Code :

- DQN class
  - init ()
  - forward()
- Experience class
- Replay memory class
  - init ()
  - push()
  - sample() & can-provide-sample()
- Epsilon greedy strategy
  - init ()
  - get-exploration-rate()
- RL Agent
  - init()
  - select-action()
- CartPoleEnv Manager ()

Scanned with CamScanner