Page 1, Values and policy

Formally it could be written as $\pi(s) = \arg\max_a Q(s, a)$, which means that the result of our policy π at every state s is the action with the largest Q.

In math notation, policy is usually represented as $\pi(s)$, well use this notation as well.

Page 4, Policy gradients

We define policy gradient as $\nabla J \approx \mathbb{E}[Q(s, a)\nabla \log \pi(a|s)]$. Of course, there is a strong proof of this, but its not that important. Which is much more important is the semantic of this expression.

From the practical point of view, policy gradient methods could be implemented as performing optimisation of this loss function: $\mathcal{L} = -Q(s, a) \log \pi(a|s)$.

Page 5, REINFORCE

1. Initialize the network with random weight.
2. Play N full episodes, saving their (s_i, a_i, r_i, s_i) transitions.
3. For every step t of every episode k calculate discounted total reward for subsequent steps $Q_{k,t} = \sum_{i=0} \gamma^i r_i$
4. Calculate loss function for all transitions $\mathcal{L} = -\sum_{k,t} Q_{k,t} \log(\pi(s_{k,t}, a_{k,t}))$
5. Perform SGD update of weights minimizing the loss.
6. Repeat from step 2 until converged.

Page 6, CartPole Example

To do this efficiently, we calculate the reward from the end of the local reward list. Indeed, the last step of the episode will have the total reward equal to its local reward. The step before the last will have the total reward of $r_{t-1} + \gamma r_t$ (if t is an index of the last step).

Page 12, Full episodes are required

When we talked about DQN weve seen that in practice its fine to replace the exact value for discounted reward with our estimation using 1-step Bellman equation $Q(s, a) = r_a + \gamma V(s')$.

Page 13, High gradient variance

In policy gradients formula $\nabla J \approx \mathbb{E}[Q(s, a)\nabla \log \pi(a|s)]$ we have gradient proportional to the discounter reward from the given state.

Page 13, Exploration

In math notation, entropy of the policy is defined as: $H(\pi) = -\sum \pi(a|s) \log \pi(a|s)$

Page 18, Results

The entropy is decreasing over time from 0.69 to 0.52. The starting value corresponds to the maximum entropy with two actions, which is approximately 0.69:

$$H(\pi) = -\sum_a \pi(a|s) \log \pi(a|s) = -(\frac{1}{2} \log(\frac{1}{2}) + \frac{1}{2} \log(\frac{1}{2})) \approx 0.69$$

The fact that the entropy is decreasing during the training, as indicated by the following chart, shows that our policy is moving from the uniform distribution to more deterministic actions.