
Experiments by: Ashish Budhiraja
Course: Advanced Computer Vision
Instructor: Jia-Bin Huang
Q-Learning

- **Q-Learning**: sample-based Q-value iteration
  \[
  Q_{k+1}(s, a) \leftarrow \sum_{s'} T(s, a, s') \left[ R(s, a, s') + \gamma \max_{a'} Q_k(s', a') \right]
  \]

- **Learn \(Q(s,a)\) values as you go**
  - Receive a sample \((s,a,s',r)\)
  - Consider your old estimate: \(Q(s, a)\)
  - Consider your new sample estimate:
    \[
    \text{sample} = R(s, a, s') + \gamma \max_{a'} Q(s', a')
    \]
  - Incorporate the new estimate into a running average:
    \[
    Q(s, a) \leftarrow (1 - \alpha)Q(s, a) + (\alpha)[\text{sample}]
    \]

[Demo: Q-learning – gridworld (L10D2)]
[Demo: Q-learning – crawler (L10D3)]
Approximate Q-Learning

\[ Q(s, a) = w_1 f_1(s, a) + w_2 f_2(s, a) + \ldots + w_n f_n(s, a) \]

- **Q-learning with linear Q-functions:**
  
  transition \( = (s, a, r, s') \)
  
  difference \( = [r + \gamma \max_{a'} Q(s', a')] - Q(s, a) \)

  \[ Q(s, a) \leftarrow Q(s, a) + \alpha \text{[difference]} \]

  \[ w_i \leftarrow w_i + \alpha \text{[difference]} f_i(s, a) \]

- **Exact Q’s**
- **Approximate Q’s**

- **Intuitive interpretation:**
  - Adjust weights of active features
  - E.g., if something unexpectedly bad happens, blame the features that were on: disprefer all states with that state’s features

- **Formal justification:** online least squares
Videos at different Learning Rate (Breakout)

77 epochs

200 epochs
Videos at different Learning Rate (Pong)

141 epochs

200 epochs
Videos at different Learning Rate (Seaquest)

178 epochs

200 epochs
Feature Visualization

file:///home/ashishkb/RL_ACV/neon/simple_dqn/results/breakout.html
Breakout

Graphs

Average reward

Epoch

Average Q-value

Epoch

Number of games

Epoch

Average loss

Epoch
Credits:

1) Prof. FRED G. MARTIN
   http://www.cs.uml.edu/ecg/index.php/Alfall16/PS3b


3) https://github.com/devsisters/DQN-tensorflow

4) https://github.com/tambetm/simple_dqn