ADL HW3 Report
B04705026 資管四 林彥廷

- ## Q1: Basic Performance

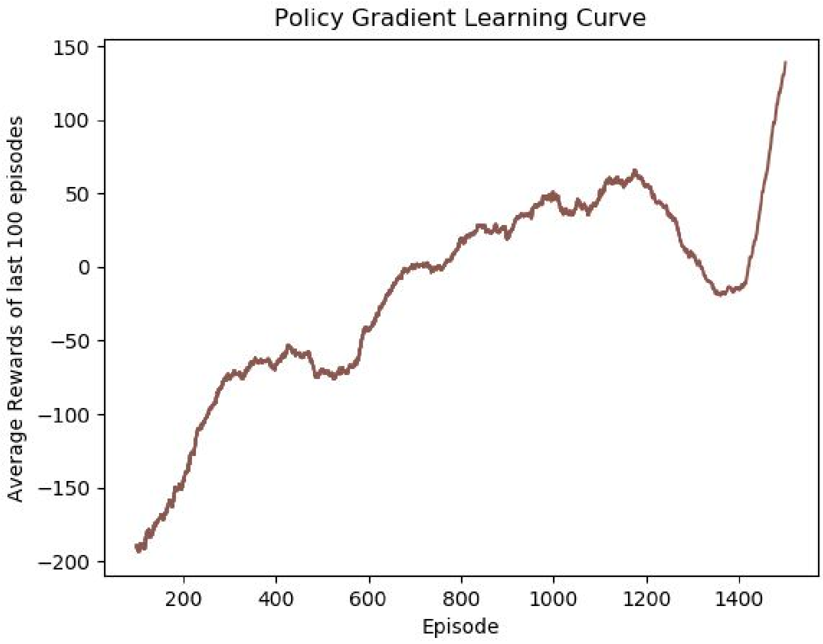
1. ### Policy Gradient Model

   - Model Architecture

Layer	Input Dim	Output Dim	Notes
Fully Connected	Dim of state	512	
Fully Connected	512	# Actions	
Softmax Norm.	# Actions	# Actions	Normalized to sum=1

   - Results and Learning Curve on Lunar Lander

Mean scores in 30 episodes: 119.87216564645134



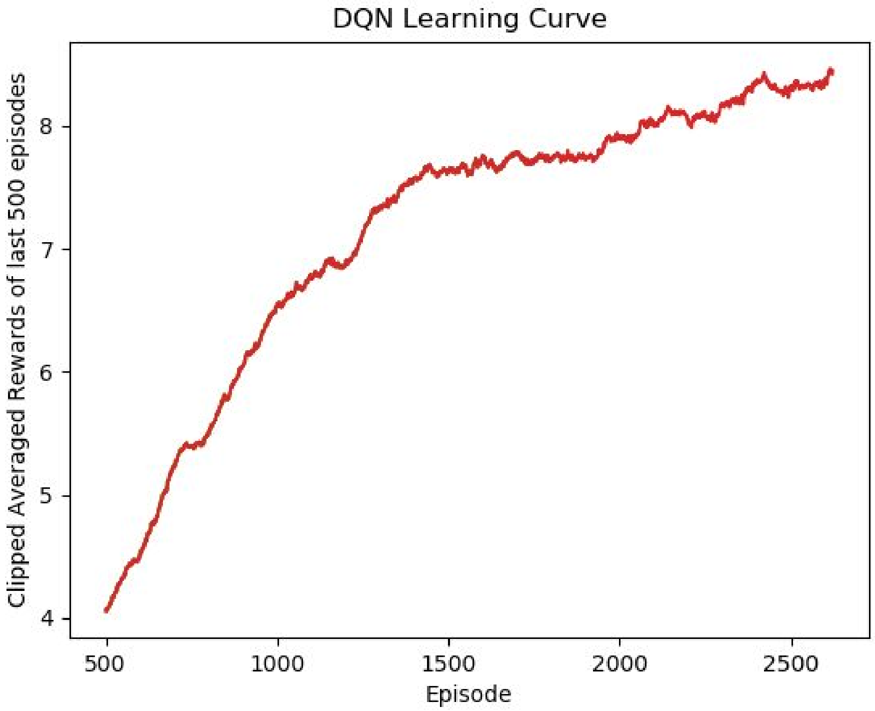
2. DQN Model

- Loss: Smooth L1 Loss
- Model Architecture

Layer	Kernel / Input Dim	Stride / Output Dim
Conv	8	4
Conv	4	2
Conv	3	1
Fully Connected	3136, 512	512
Fully Connected	512	#Ations

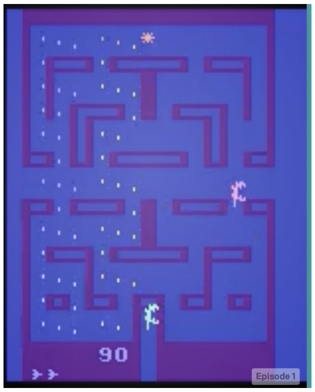
- Result and Learning Curve on Assault

  Clipped Mean scores in 100 episodes: 230.56



Q2: DQN Hyper-parameters

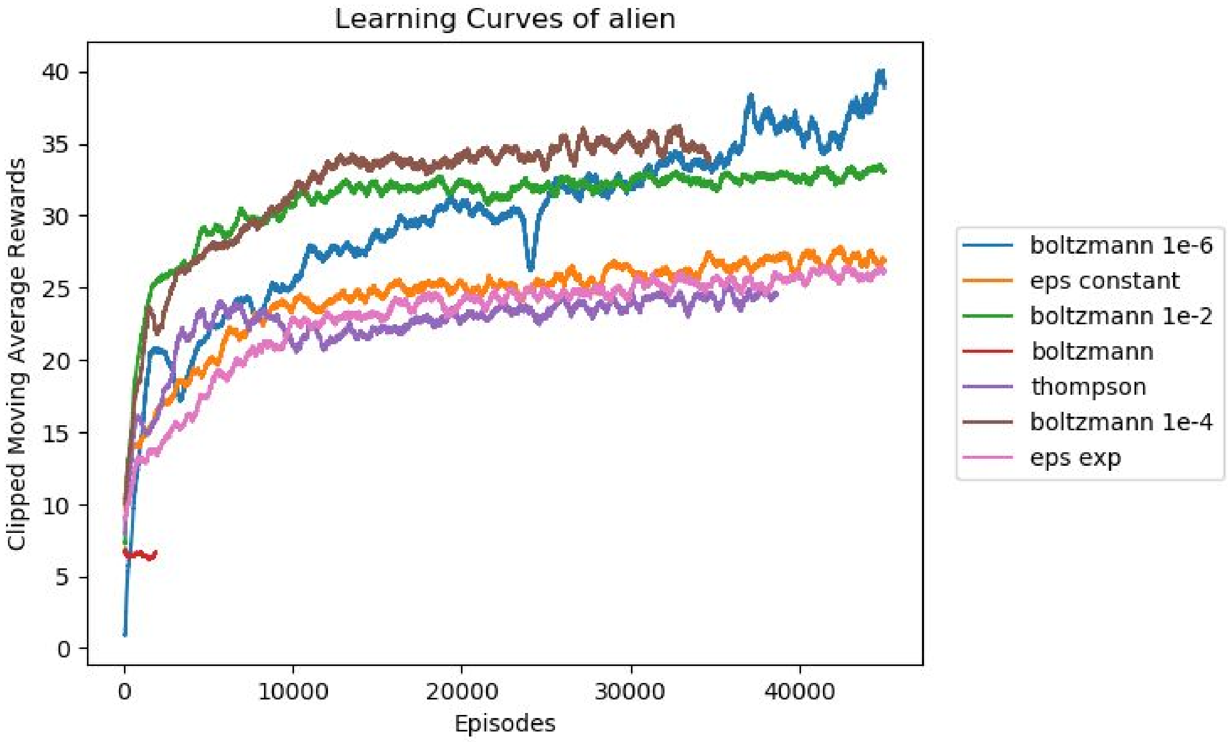
1. Environment: Alien-v0



2. Hyperparameters: Exploration Rules
   - Candidates
     - Greedy
     - ϵ-greedy
     - Boltzmann
     - Bayesian Neural Network / Thompson sampling (by Dropout)
   - Why?
     - The Atari game-Alien is a maze-like task that heavily relies on exploration methods in training.
     - Agent has to sufficiently explore the environment to find (approximately) optimal strategy, while agent also has to exploit current information to do effective move.
       The trade-o□ff between exploration and exploitation demands efficient exploration capabilities□ maximizing the effect of learning while minimizing the costs of exploration□.
3. Learning Curves

Method	Note *(Temp = temperature parameters in Boltzmann Dist.)*
Boltzmann	*Temp* = 1
Boltzmann	*Temp* = 0.01
Boltzmann	*Temp* = 0.001
Boltzmann	*Temp* = 0.0001

ϵ-greedy	ϵ = 0.1
ϵ-greedy	ϵ exponentially decrease from 0.9 to 0.1 in first 200 episodes
Thompson	Dropout Rate = 0.3 to approximate Bayesian Neural Network



Learning Curves of alien

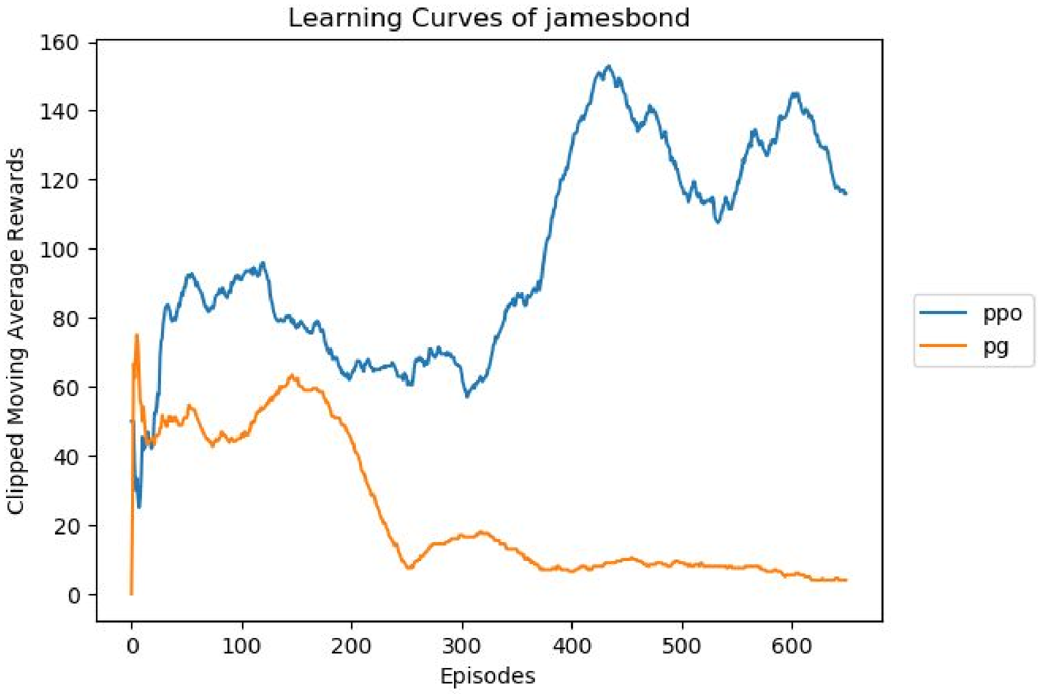
4. How exploration rules affect learning?

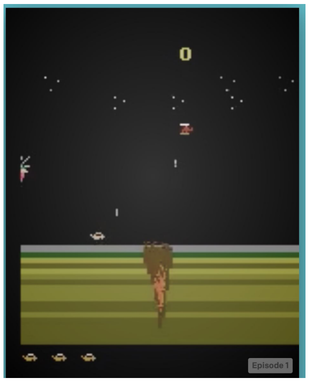
- Well-tune Boltzmann >> ϵ-greedy > Thompson sampling
- Tuning temperature parameter is **VITAL** in the Boltzmann method.
Temperature = 0 => Exploitation
Temperature = inf => Exploration
(However, Temp =1 failed to learn anything)
- ϵ scheduling in ϵ-greedy merely affects performance and speed.
- Thompson sampling use dropout to approximate Bayesian Neural Network. When sampling size is large, Bayesian style is better than single point estimate, but the improvement is insignificant in this task.

Q3: Improvements to PG & DQN

1. *Proximal Policy Gradients*

- Environment: Jamesbond-ramNoFrameskip-v0
- Learning Curves: *PPO vs Policy Gradients*

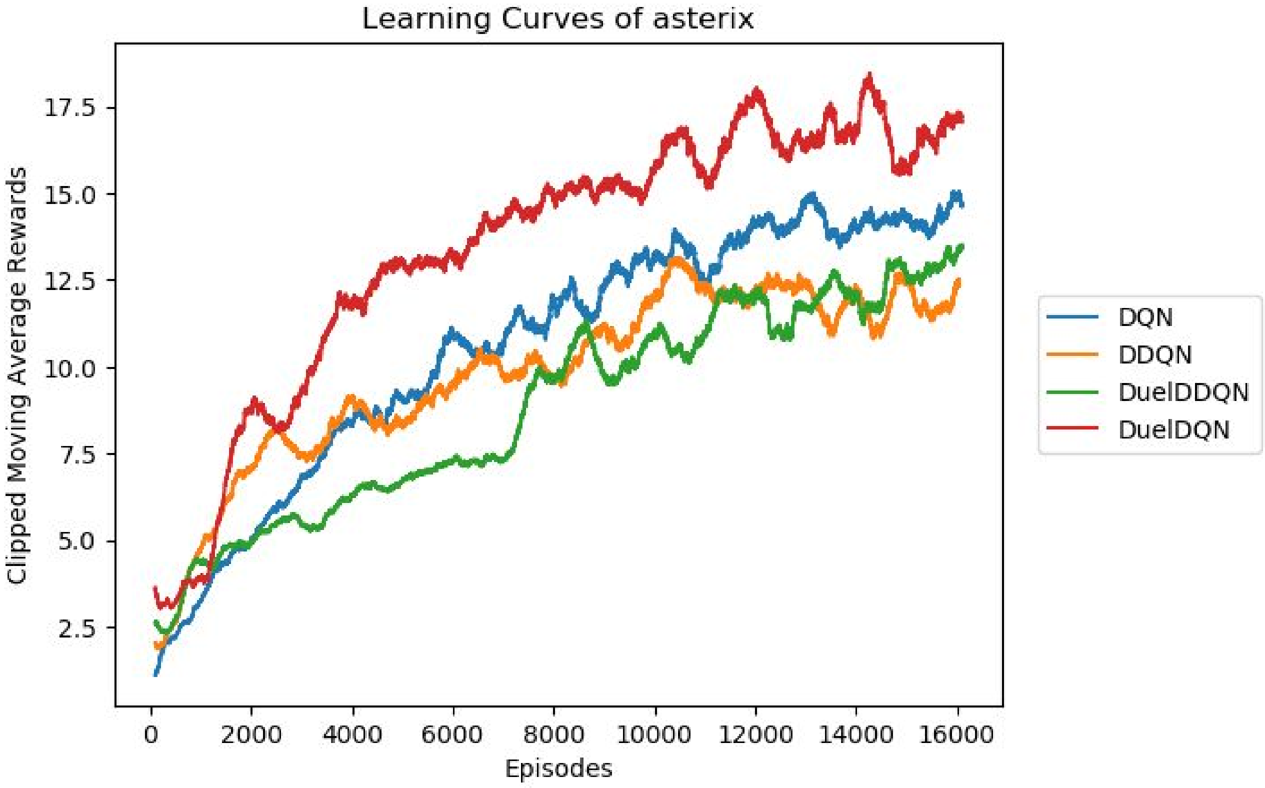


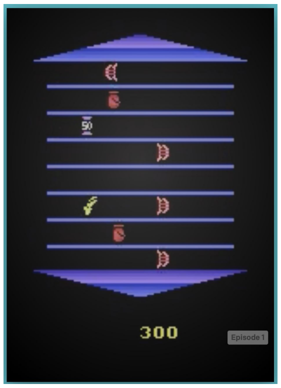


- Why?
  - My claim:
    - PPO solved the problem of catastrophic collapse during training.
  - In vanilla PG, the problem of credit assignment causes huge noise in data which leads to a unstable optimization process and ends up catastrophic failing the latter phase of training in the consequence of on-policy settings.
  - PPO converts the training process to off-policy to increase the stability in each batch update by Importance Sampling and is more robust to noisy data.
  - PPO optimizes for surrogate loss which takes “balancing exploration vs exploitation” and “minimizing distributionally differences between two policy network”.

2. ***Duel DQN*** *vs* ***Double DQN*** *vs* ***Duel Double DQN*** *vs* ***DQN***

- Environment: AsterixNoFrameskip-v0
- Learning Curves





- Why?
    - My claim:
        - **Duel DQN explicitly detach states' and actions' contributions to rewards, while Double DQN and vanilla DQN do not address this problem.**
    - Duel DQN clearly outperformed the other counterparts on this task.
    - Duel DQN modifies the network architecture and clearly separates the state values and action advantages.
      State value is independent of action which is useful when the action makes no difference to the rewards (eg. the agent is 100% to die regardless of which action taken)
    - In the environment - Asterix, one of top improvement environments in original paper, a lot of reward seem to be little related or even independent of action taken next.