

Deep Reinforcement Learning for play Atari 2600 game

Pedro Torres

INF 2064 - MODVIS

Pontifícia Universidade Católica (PUC) - Rio de Janeiro

19 de novembro, 2019

Objetivo

Utilizar um algoritmo de *Deep reinforcement learning*, DQN, para jogar Space Invaders (Atari 2600) no ambiente da ferramenta OpenAI Gym.

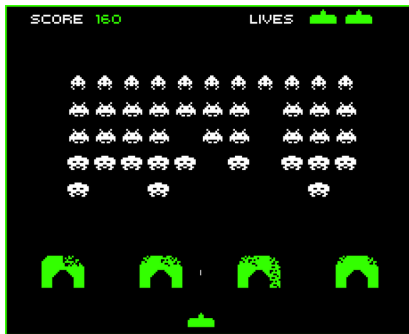


Figura: Frame capturado do jogo Space Invaders

Ambiente OpenAI Gym

Nesse ambiente, cada observação é dada por uma imagem RGB com *shape* (210, 160, 3). Cada ação realizada é repetidamente propagada em uma duração de k frames, onde $k \in \{2, 3, 4\}$.

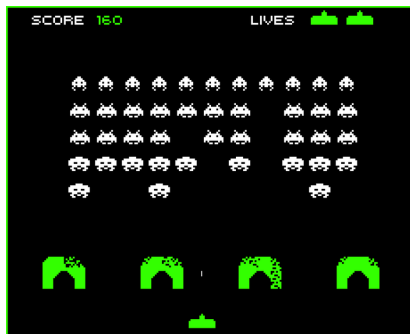


Figura: Frame capturado do jogo Space Invaders

- gym 0.15.3
- keras 2.2.4
- numpy 1.17.2
- opencv-python 4.1

Reinforcement learning é uma abordagem baseada em Processos de decisões de Markov para tomar decisões.

- *Rewards*:
 - Positivo
 - Negativo
- Quais ações o agente deve tomar a fim de maximizar o *reward*?

Q-Table é uma matriz que correlaciona o estado em que o agente se encontra com as possíveis ações que podem ser tomadas.

State	Right	Left	Up	Down
1	0	0.31	0.12	0.87
2	0.98	-0.12	0.01	0.14
3	1	0.10	0.12	0.31
4	0.19	0.14	0.87	-0.12

Figura: Exemplo de *Q-table*

A *Deep Q-Network* (DQN) foi o primeiro algoritmo de RL que mostrou-se capaz de trabalhar diretamente com imagens (*raw data*) em uma variedade de ambientes.

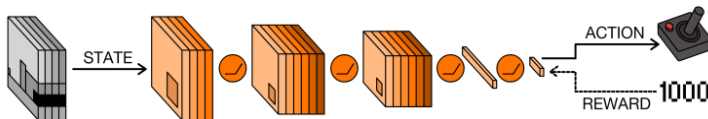


Figura: Arulkumaran, Kai (2017)¹

¹A brief survey of deep reinforcement learning, 2017

Em uma DQN, a rede seleciona e avalia cada ação. Isso pode resultar em uma superestimação do *Q-value*.

$$Q_{\max} = \begin{cases} r_{j+1} & \text{if episode terminates at } j+1 \\ r_{j+1} + \gamma \max_{a_{j+1}} Q_{\text{target}}(s_{j+1}, a_{j+1}; \theta^-) & \text{otherwise} \end{cases}$$

Figura: Estimativa do Q-value em uma DQN

A fim de resolver esse problema, um novo modelo foi proposto: DDQN (Double deep q-network).

- Q-network para escolher uma ação.
- Q-network' (target) para avaliar uma ação.

Implementação

► Implementation