Capstone Project Reinforcement Learning: Tetris

Andrew Falcone
10/17/2022

Project Summary:

Program Tetris and program an agent to play the game.

1) Create the game Tetris playable for humans (N64 color schema)

2) Create an agent able to play the game optimally (in progress)

- Limitations: agent is only able to choose to swap, rotate and then position piece. No rotations or slides upon landing to reduce training complexity.

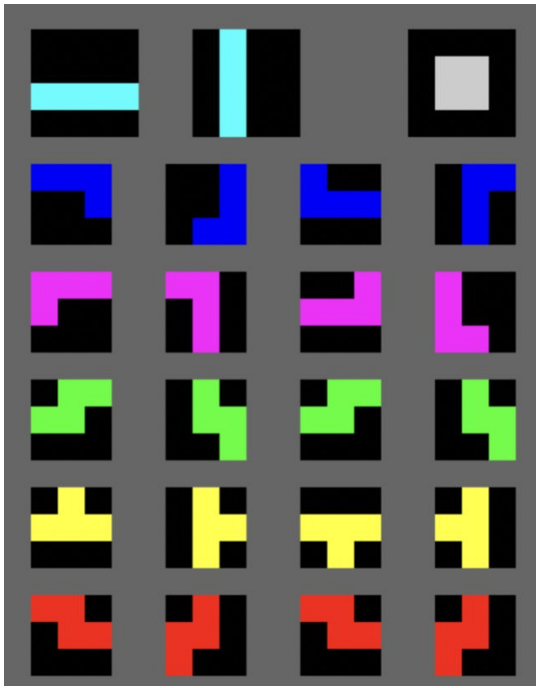
Motivation

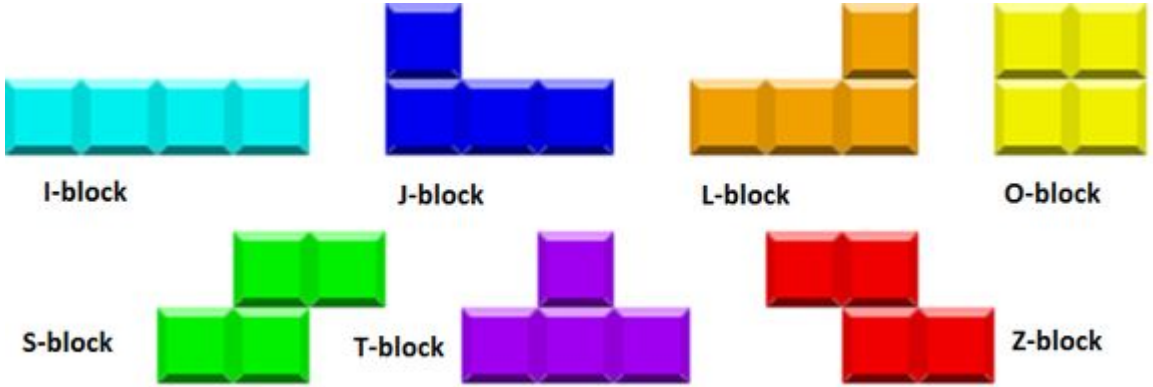
-Love of games and desire to practice implementing a reinforcement learning algorithm!

-DeepMind’s recent breakthroughs using reinforcement learning algorithms including:

- AlphaGo 2016
- AlphaZero 2017
- AlphaFold 2020

Meet the Pieces & Rotations





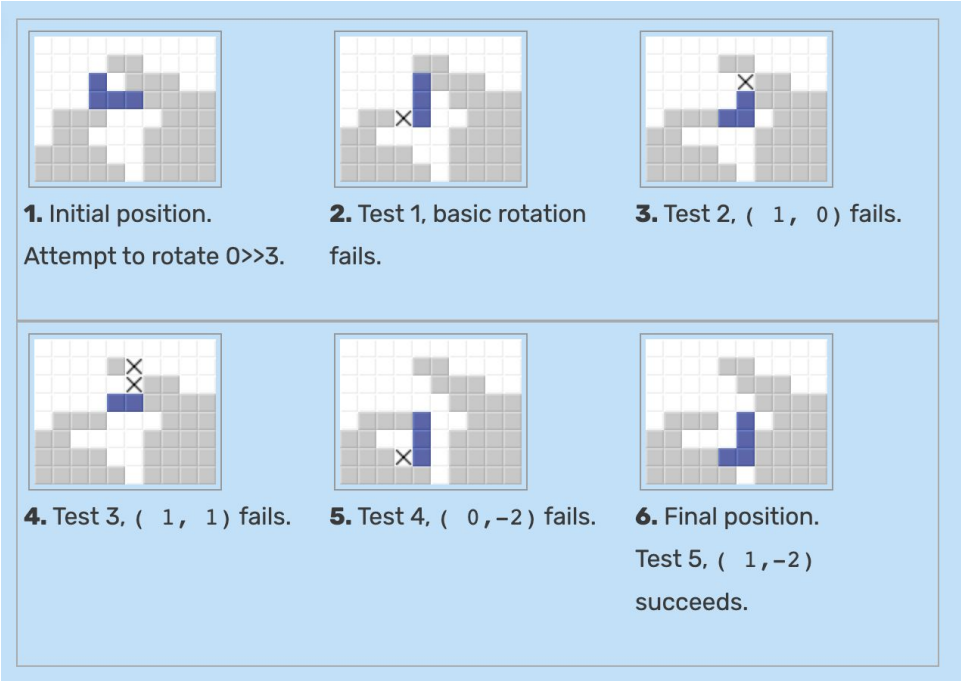
Goal of Playing the Game

1) CLEAR LINES
2) More Lines, the better (demo)
3) 4 Lines at once is a Tetris and worth more points (depends on version actual score)

Coding Challenges in Game Creation

1) Started with starter code game to learn about Pygame clocks and rendering (https://levelup.gitconnected.com/writing-tetris-in-python-2a16bddb5318)
2) Fixed color scheme from random to N64
3) Implemented SRS rotation system
4) Added shadow block for piece placement
5) Added piece queue to see in coming pieces and the ability to swap piece
6) Added debugging features such as choosing which block to come next and adding lines/levels for free

SRS: Super Rotational System



1. Initial position.
Attempt to rotate 0>>3.

2. Test 1, basic rotation fails.

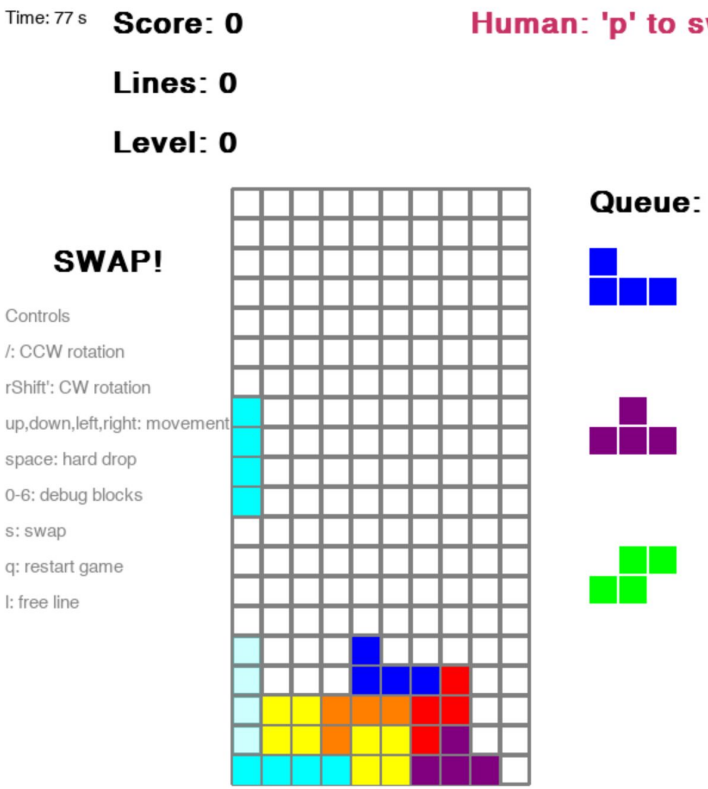
3. Test 2, `( 1, 0)` fails.

4. Test 3, `( 1, 1)` fails.

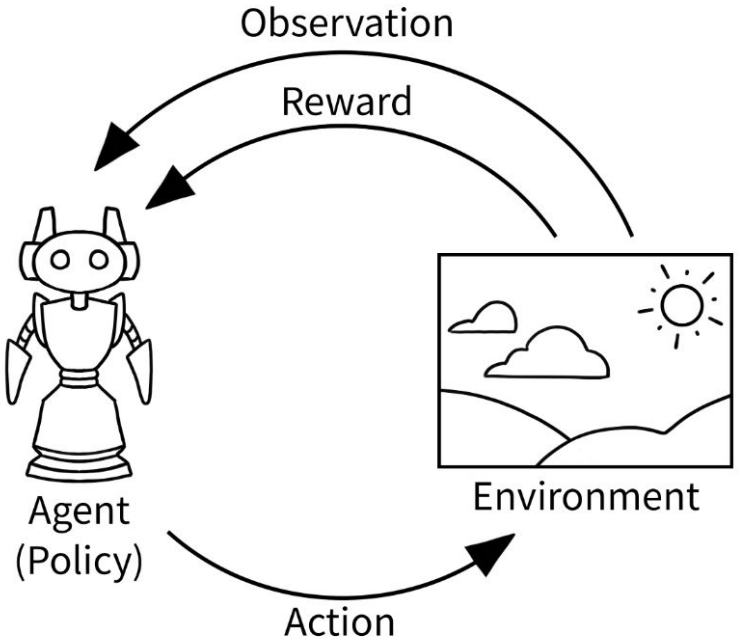
5. Test 4, `( 0,-2)` fails.

6. Final position.
Test 5, `( 1,-2)` succeeds.

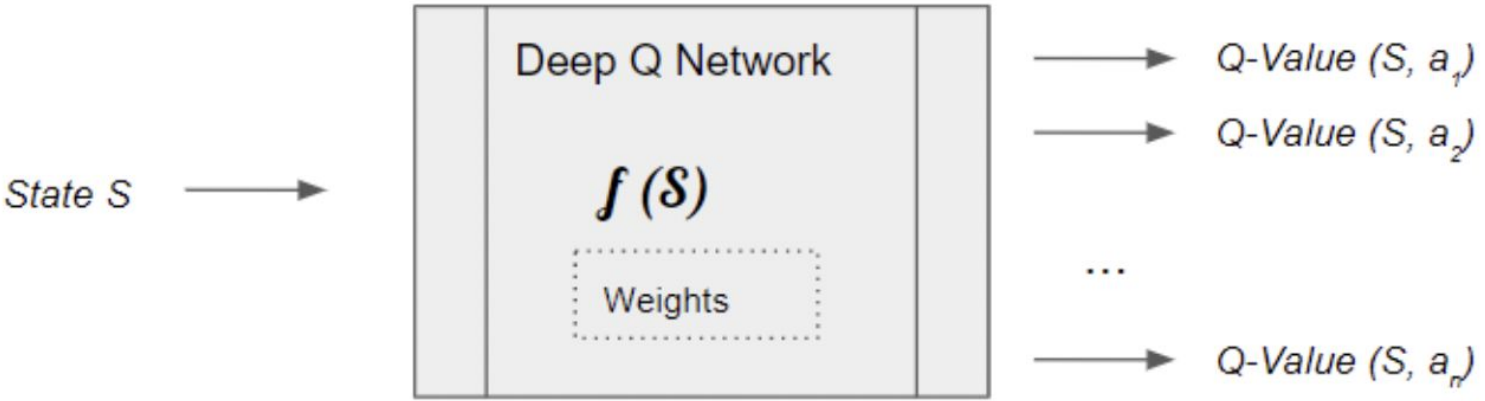
Shadow Piece



Reinforcement Learning

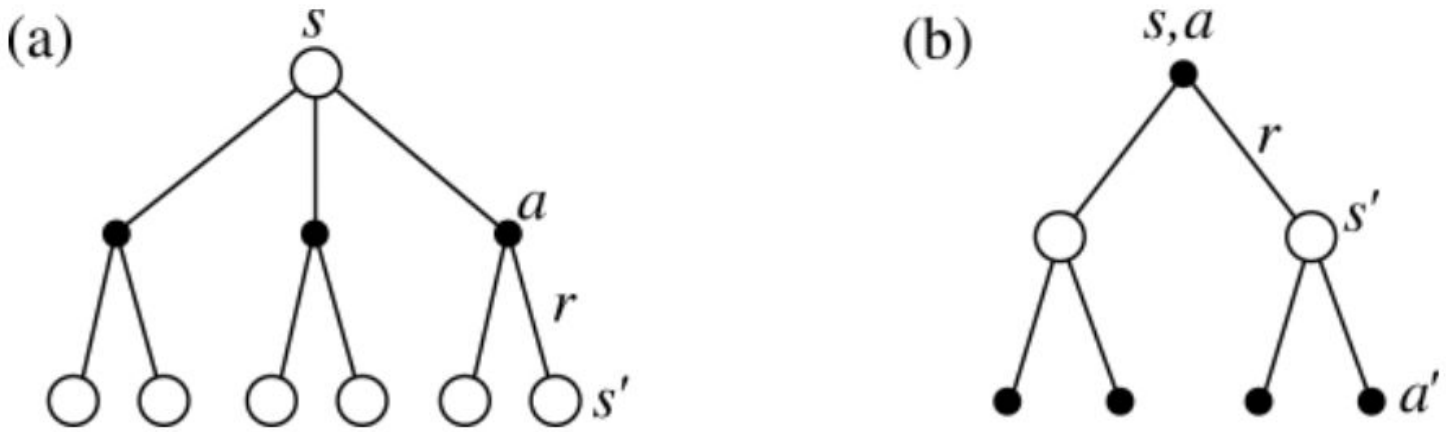


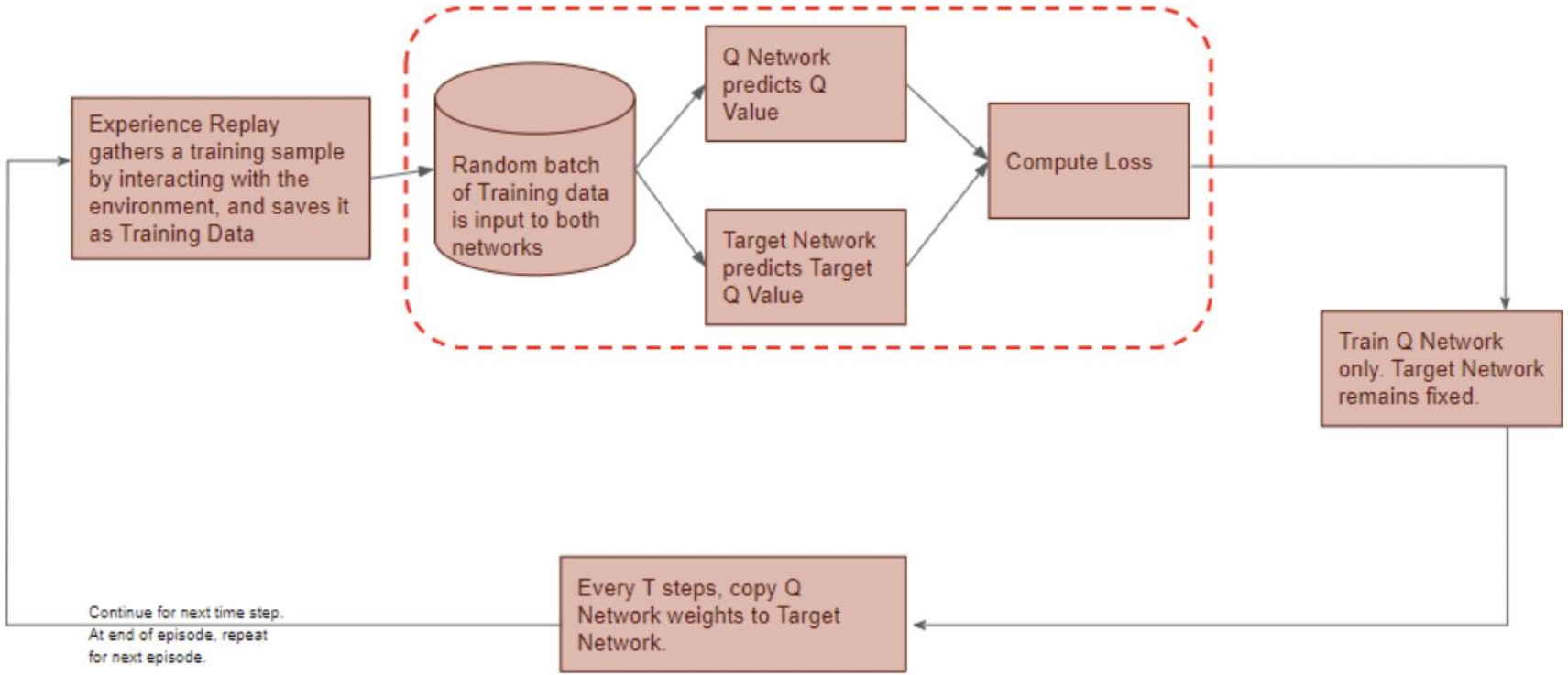
Q Network



https://towardsdatascience.com/reinforcement-learning-explained-visually-part-5-deep-q-networks-step-by-step-5a5317197f4b

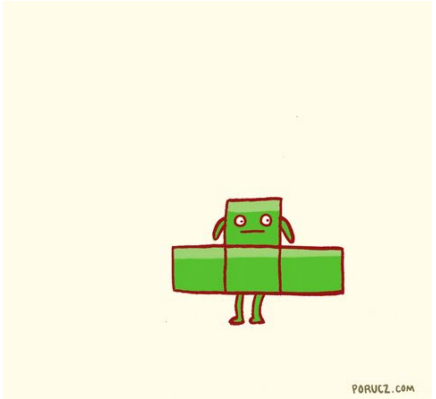
Value Functions and Q Functions





https://towardsdatascience.com/reinforcement-learning-explained-visually-part-5-deep-q-networks-step-by-step-5a5317197f4b

Challenges



1) Working with OpenAI Gym- version stability and kernel crashes
2) Bugs mid-game are difficult to find and fix! Simple indexing issues are a nightmare
3) Exploitation vs Exploration
    a) Agent has difficulty learning good states
4) Training time (agent is still training)
5) Getting all the pieces of code to literally to fit together
    a) Separating human vs computer play
    b) Getting pertinent data into neural network
    c) Implementing code in OOP style
6) Using too much data (whole game matrix initially), did not show any signs of converging or learning

Future Goals

Finish modeling and I hope to share a running model soon!

Learn more reinforcement learning and hopefully add extensions.

Questions???