

# Learning Breakout using NEAT

James T. Murphy III<sup>1</sup> Skyler Thomas<sup>2</sup>

The University of Texas at Austin

December 10, 2017

---

<sup>1</sup>jamesmurphy@math.utexas.edu

<sup>2</sup>skylertthomas0@gmail.com

# A neat application of a neat algorithm

In 2015 youtuber SethBling trained a neural network to play and successfully beat, the first level of Super Mario World.



# How

This was done using the Neural Evolution of Augmenting Technologies AlgOrithm (NEAT-O) from the 2002 paper by Stanley and Mikkulainen.

# NEAT, the algorithm

- 1 NEAT is a type of genetic algorithm that follows the biological metaheuristics of “fitness begets evolution”.

# Breakout

- ▶ In the game, a layer of bricks lines the top of the screen. A ball travels across the screen, bouncing off the top and side walls of the screen. When a brick is hit, the ball bounces away and the brick is destroyed. The player loses a turn when the ball touches the bottom of the screen. To prevent this from happening, the player has a movable paddle to bounce the ball upward, keeping it in play.

# Null Hypothesis & Experiment

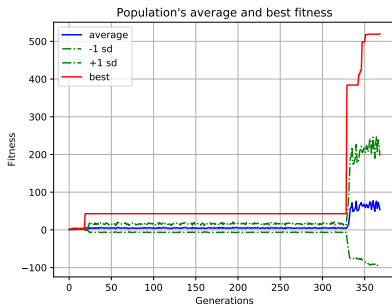
- ▶ Null Hypothesis: NEAT will learn something.
- ▶ Experiment: A copy of NEAT was made using the python pygame package and a NEAT network was made using then python NEAT package.

# Results

Qualitatively, it appears there are three eras

1. Pre-ball-tracking arc: The network does nothing or makes small and arbitrary movements
2. Ball-tracking arc: It is apparent the ball is tracking the ball, but there are cases the network hasn't learned to do deal with yet.
3. Aiming-arc: The network actively moves to and aims the ball to clear all 520 blocks.

# Results



**Figure:** *A successful training experiment.* In this case, the pre-ball-tracking era lasts until around generation 325, where the network abruptly learns how to track the ball. The ball-tracking era then lasts until around generation 350, where the network learns to avoid missing stray blocks. The aiming era is reached in the last few generations. Generations of 200 individuals took an average time of 2.835 sec to evaluate on an Intel Core i7-4790K @ 4.00GHz in parallel on 6 cores. Total runtime  $\approx$  17.5 minutes.



# Discussion

- ▶ It is difficult to predict what the network will and will not learn with respect to the inputs to the network and more so with respect to the hyperparameters.
- ▶ The network is incredibly sensitive to parameters and hyperparameters
- ▶ The number of generations required to successfully complete the task has a critical point.

# Future

- ▶ Explore the relationship between fitness and solution
- ▶ Compare with DQN