Final Report: Comparison between Backpropagation and Genetic Algorithm for "Don't Touch the Spikes"

I. Introduction

In this study, we aimed to compare two machine learning approaches, namely the backpropagation algorithm and the genetic algorithm, to solve the specific problem of the "Don't Touch the Spikes" game. The goal was to determine which method was more effective in optimizing the bird's performance in this game.

2. Machine Learning Approaches

2.1 Backpropagation Algorithm

We utilized the backpropagation algorithm to train a neural network to solve the game problem. This approach was particularly suitable due to the availability of large quantities of input data and the complexity of the relationship between inputs and outputs.

2.2 Genetic Algorithm

The genetic algorithm was chosen to optimize the solution to the game problem. This method proved relevant, especially when the solution's topology is not known in advance, and the search space is complex.

3. Problem to Solve

3.1 Description of the "Don't Touch the Spikes" Game

The game challenges players to guide a small bird through a perilous environment filled with spikes. The bird is subject to gravity, and the player must tap the screen to make it jump (or not) to dodge the spikes. Each time the bird touches a wall without hitting the spikes, it earns a point. The goal is to score as many points as possible without touching any spikes.

3.2 Difficulty Progression

As the game progresses, the difficulty increases with more spikes generated on each wall, the bird's speed increasing, and the spikes starting to move more quickly.

4. Implementation of AI

A neural network was associated with the bird, with a single hidden layer of 4 neurons to simplify calculations and a single output neuron. The output neuron's value determines the likelihood of the bird jumping. The same input data were used for both approaches, including bird position, spike positions, ceiling and floor positions, and the bird's horizontal speed.

5. Performance Evaluation

5.1 Evaluation Criteria

The performance of both the backpropagation and genetic algorithm approaches was assessed based on the following criteria:

Backpropagation Algorithm:

- **Score Achieved:** The backpropagation algorithm achieved a score of 38 after 2 minutes of training.
- **Time Efficiency:** It demonstrated a shorter training time compared to the genetic algorithm.
- **Results Quality:** Despite the shorter training time, the backpropagation algorithm yielded inferior results.
- **Dataset Construction:** The algorithm required manual construction of a dataset, posing a challenge in generating a diverse and comprehensive dataset.

Genetic Algorithm:

- **Score Achieved:** The genetic algorithm achieved a higher score of 82 after 18 minutes of training spanning 21 generations.
- **Training Time:** It took longer to train compared to the backpropagation algorithm.
- **Unsupervised Learning:** The genetic algorithm employed unsupervised learning, allowing for adaptability to various scenarios without predefined datasets.
- **No Theoretical Score Limit:** Theoretically, there is no upper limit to the score that the genetic algorithm can achieve, providing flexibility in potential performance.

5.2 Results

The comparison between the two algorithms reveals distinct trade-offs in performance:

Backpropagation Algorithm:

- **Advantages:**
- Faster training time (2 minutes).
- Lower computational cost.
- **Drawbacks:**
- Limited by manually constructed datasets.
- Inferior performance, achieving a score of 38.

Genetic Algorithm:

- **Advantages:**
- Achieved a higher score of 82.
- Unsupervised learning capability.
- Theoretically, no limit to achievable scores.
- **Drawbacks:**
- Longer training time (18 minutes).
- Requires more computational resources.

5.3 Conclusion

The choice between the backpropagation algorithm and the genetic algorithm depends on the specific priorities of the task. While the backpropagation algorithm is more time-efficient, the genetic algorithm demonstrates the potential for superior performance and adaptability, making it suitable for scenarios where the dataset construction is challenging or where the score has no predefined limit.

The trade-offs between training time, computational cost, and performance must be carefully considered based on the specific requirements and constraints of the application.

6. Conclusion

In summary, the comparison between the backpropagation and genetic algorithms for "Don't Touch the Spikes" game revealed distinctive trade-offs.

The backpropagation algorithm, efficient in time, achieved a score of 38 in 2 minutes. However, its reliance on manually constructed datasets and the capped score underscored limitations.

In contrast, the genetic algorithm, with a longer 18-minute training period, excelled with a high score of 82. Unsupervised learning and limitless score potential make it appealing for adaptable scenarios.

The choice between them hinges on priorities; backpropagation for time efficiency and genetic for adaptability. Balancing training time, resources, and performance remains essential, tailoring the selection to specific application needs. This study informs decisions in employing machine learning for similar gaming scenarios.

Source code: https://github.com/leDarron/MachineLearning_FinalProject