

Witam gitas

Kamil Koziół, Magdalena Sadowska, Wiktor Leszczyński

1 Introduction

This project intends to compare 4 different AI implementations which can be used when making bots that play games

1.1 Podsekcja

ziut DAWAJ TEKST

2 Conditions

We have now added a title, author and date to our first \LaTeX document!

2.1 Podsekcja

ziut

3 Genetic algorithm

Genetic Algorithm is a machine learning technique that uses principles of natural selection to find the optimal solution to a problem.

We can determine how good individual is doing by calculating his fitness.

3.1 Algorithm

Each of these steps is repeated untill conditions are met and then the current generation value is increased

3.1.1 Initialization

The first step is initializing a population using random neural networks, where each solution represents a possible strategy for playing the game.

Done only once at the first generation

3.1.2 Evaluation

Each solution is evaluated by playing the game using the corresponding neural network. Fitness score of the solution is calculated based on its performance in the game. The fitness score is a measure of how well the solution performs in achieving the objective of the game.

3.1.3 Selection

Then probabilities of picking each individual are made. Those probabilities range from $\langle 0, 1 \rangle$

3.1.4 Mutation

To introduce diversity into the population, picked by random individuals undergo mutation, where random changes are made to their weights and biases. This process helps to prevent the population from converging too quickly to a local optimum.

Every weight and bias has a change of being mutated based on *mutationRate*

If mutated those are modified using this formula:

$$x = x + \text{randomGaussian}(\text{initMean}, \text{initStdev}) * \text{mutationPower}$$

Then being kept at $\langle \text{minValue}, \text{maxValue} \rangle$

3.1.5 Crossover

The selected solutions are then combined through crossover, where random pairs of individuals exchange weights and biases of neural network to create new offspring solutions. This process mimics the natural process of sexual reproduction, where genes from two parents combine to produce offspring with a mix of genetic traits.

3.1.6 Elites

Some of the best-performing solutions from the previous generation are also added to the new generation

3.2 Fitness function

Fitness function is calculated by this formula:

$$fitness = \begin{cases} age^2 * (2^{apples}) & apples < 10 \\ age^2 * (2^{10}) * (apples - 9) & apples \geq 10 \end{cases}$$

Each snake in the game has a hunger variable, denoted as H . Upon consumption of an apple, H is replenished to a maximum value of H_{\max} . At every time step, H decreases and once it reaches 0, the snake perishes.

3.3 Hyperparameters

Hyperparameters can be modified in file *settings.json*.

3.3.1 Neural network

Neural network used to train the snakes is described by this model:

$$Model = \begin{matrix} InputLayer(28, linear) \\ DenseLayer(20, relu) \\ DenseLayer(12, relu) \\ OutputLayer(4, softmax) \end{matrix}$$

Description:

$$Layer(neurons, activationFunction)$$

4 NEAT

We have now added a title, author and date to our first L^AT_EX document!

4.1 Podsekcja

ziut

5 DeepQ

We have now added a title, author and date to our first L^AT_EX document!

5.1 Podsekcja

ziut

6 Supervised Classification

We have now added a title, author and date to our first L^AT_EX document!

6.1 Podsekcja

ziut

7 Summary

This project intends to compare 4 different AI implementations which can be used when making bots that play games

7.1 Podsekcja

ziut DAWAJ TEKST