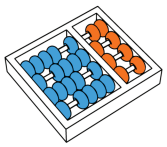
MC886A/MO444A - Machine Learninng
Institute of Computing - Unicamp
First Semester 2021



Prof. Esther Colombini
esther@ic.unicamp.br
PEDs
Alana Correia (alana.correia@ic.unicamp.br
Patrick Ferreira (patrickctrf@gmail.com)

Project 3 - Deadline: 04/07/2021

1 Goal

This assignment aims to apply Evolutionary Algorithms and Reinforcement Learning Methods to solve the Pac-Man game. We will use as the basic engine the environment available in `http://cs.brynmawr.edu/Courses/cs372/fall2017/Code/search.zip`. You should download the code and use the smallClassic, mediumClassic, and originalClassic layouts.

2 PART I - Evolutionary Model

In this part, your group should apply an evolutionary computing technique to solve the Pac-man problem. The work consists of finding an adequate solution to the problem, evaluating it according to different parameters. You must clearly define:

- The evolutionary model adopted
- Variations on parameters
- Fitness function adopted
- Population size
- Stop criteria
- Selection technique
- Crossover technique
- Mutation technique
- Replacement method

- Mutation rate
- Crossover rate

For the Evolutionary model, you should train your method for the 3 layouts mentioned earlier, computing the fitness score (best, worst, and average individual), number of actions, and score per generation.

3 PART II - Reinforcement Learning

In the second part of the work, your group should solve the Pac-Man game by implementing an RL algorithm. You can use libraries that implement the methods you want to use, clearly defining:

- The MDP formulation (states, actions, reward function)
- The discretization model adopted
- The number of training episodes
- The stop criteria
- The learning rate value and other parameter values used

For the RL experiments, you should train your method for the 3 layouts mentioned earlier, computing the reward, number of actions, and score per episode.

4 Results Discussion and Algorithms Comparison

After training your models, you should discuss your results, comparing the solutions for both classes of models. For each layout, run your best EA and RL models 10 times (assuming the ghosts appear in distinct locations each time) and compute the scores achieved. Compare and discuss the results.

A critical evaluation is expected on the relationship between adopted parameters x solution performance. Graphs and tables representing the evolution of the solutions are expected. Additional comparisons with the literature are welcome, although they are not mandatory.

5 Programming language

You should use Pyhton as programming language.

6 Groups

The groups must be composed of 2 members.

7 Report and Submission

The definition of the problem, the solution, and the results obtained must be presented in a report created as a Jupyter notebook. Please, make sure you put the graphs, tables, comparisons, and critical analysis in the notebook. The report should clearly indicate what the contribution of each team member was.

The reports must be submitted following the nomenclature:

$< RAmenor > \_ < RAmaior > .zip$

Example for a group whose members have RAs 166666, 175480 and 201234. The name of the submission would be:

166666_175480_201234.zip

8 Grading

This work will be evaluated according to the following criteria:

- Submission within deadline
- Quality of the solution employed
- Report and discussions
- Code analysis
- Individual student participation in the project

8.1 Penalty policy for late submission

You are not encouraged to submit your assignment after due date. However, in case you did, your grade will be penalized as follows:

- late submission one day after the deadline: grade * 0.75
- late submission two days after the deadline: grade * 0.5
- late submission three days after the deadline: grade * 0.25