FRANKFURT UNIVERSITY OF APPLIED SCIENCES

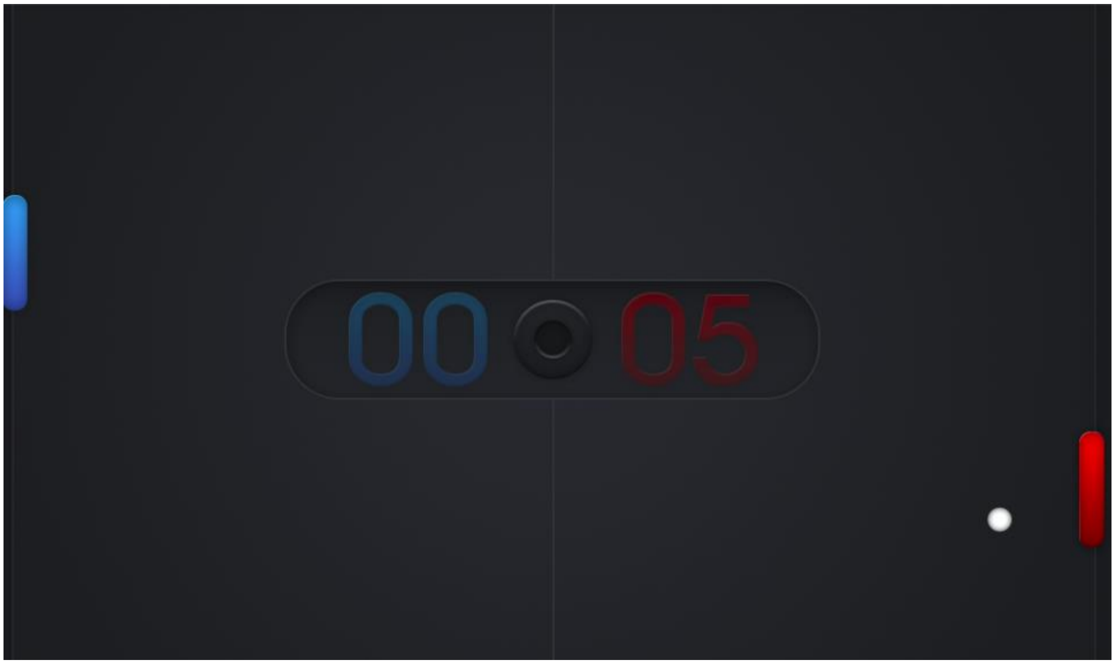
Visualizing a Self-Learning Agent to Play Pong Game

Supervisor

Prof. Doina Logofatu
Lam Gia Thuan

Description

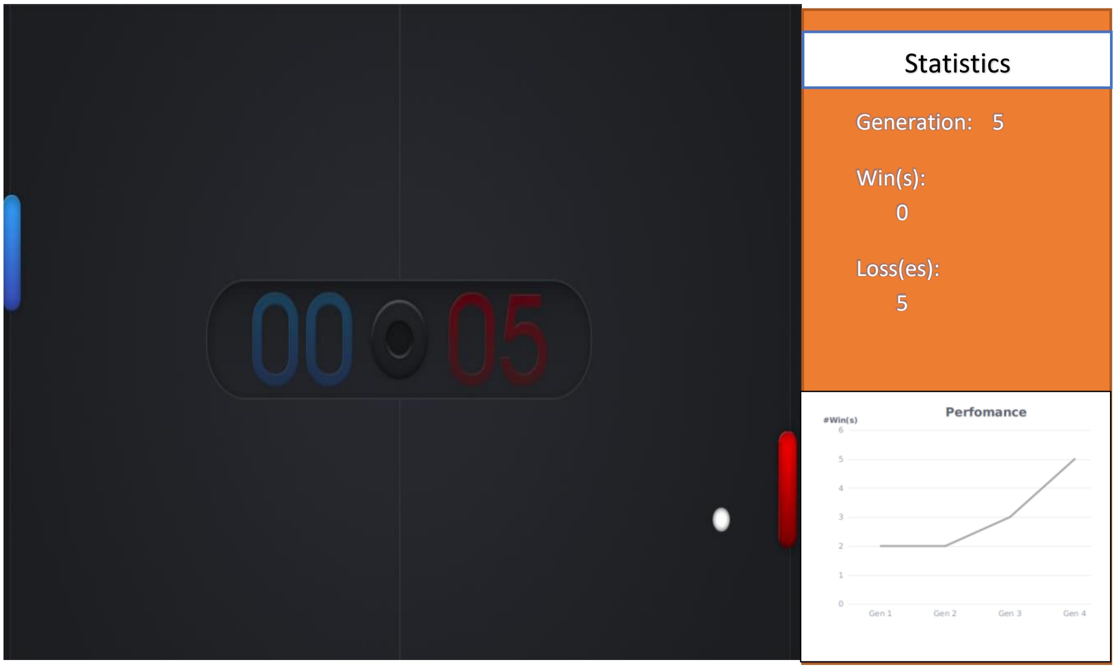
The Pong Game is one of the most popular computer games that almost everyone used to play once. In this project, your task is to create a Pong Game application in Java that allows **human playing against machine and machine playing against machine**. Most importantly, you are required to train your self-playing agents using at least **2 different algorithms such as Machine Learning or Evolutionary Algorithms** and visualize the complete learning process.



Example 1: A Pong Game

The visualization is to ensure that you are not going to use any simple deterministic algorithms in this project. It can be done by displaying how your agent performance improvement over time. Example 2 shows one way to do it. The process consists of multiple games. In each game, you can see the interactions on the left and the statistic on the right. On the right, there is a small graph changing after every game, showing the performance improvement of your agent over time. This is only one example of visualization. You are free to choose whichever way you feel comfortable on the condition that **your machine performance improvement over time is clearly displayed and we can see the interactions in each game**.

You are also free to choose any method you know of. One popular way is Reinforcement Learning. Most importantly, that you are supposed to use at least 2 methods does not imply that you need to learn too much. For example, Reinforcement Learning with Deep Neural Network is one method. If you use Reinforcement Learning with Shallow Neural Network, it will also count as the second. There are plenty libraries, so it is very unlikely that you need to implement the basics yourself. Hence, the main complexity of this task is implementation of the pong game and visualization. Your program must also allow human interaction (Playing with Mouse and/or Arrow Keys).



Generation: 5

Win(s):
0

Loss(es):
5

Example 2: An example of how to visualize the learning process

Application Input and Output

The GUI must ask the user what he or she would like to do. If the user wants to play himself, the program must let him choose one of the trained agents to play with. If the user would like to see how the agents are trained, the program must visualize the whole learning process of the chosen agent. Otherwise, the program will display a game between trained agents playing against each other.

References

Andrej Karpathy. (2016). “Deep Reinforcement Learning: Pong from Pixels.” Accessed on 13 November 2017. Available at: http://karpathy.github.io/2016/05/31/rl/.

Arthur Juliani. (2016). “Simple Reinforcement Learning with Tensorflow Part 0: Q-Learning with Tables and Neural Networks.” Accessed on 13 November 2017. Available at: https://medium.com/emergentfuture/simple-reinforcement-learning-with-tensorflow-part-0-q-learning-with-tables-and-neural-networks-d195264329d0

Stephan Rauh. (2013). “Which Is the Hottest UI Framework in the Java World: JSF or JavaFX?” Accessed on 13 November 2017. Available at: https://www.beyondjava.net/blog/hottest-gui-framework-java-world-jsf-javafx/.

Andrew Ng. (2017). “Coursera Online Course: Machine Learning.” Accessed on 13 November 2017. Available at: https://www.coursera.org/learn/machine-learning.