



Deep Q-Learning applied on Snake - Artificial Intelligence Project

9.02.2018

Florian Kleinicke, Bsc. Bsc.
Heidelberg University

Rules of Snake:

Find Food
Grow by one if it's found
Move up, down, left or right
Die when the wall is hit
Die wenn the snake is hit

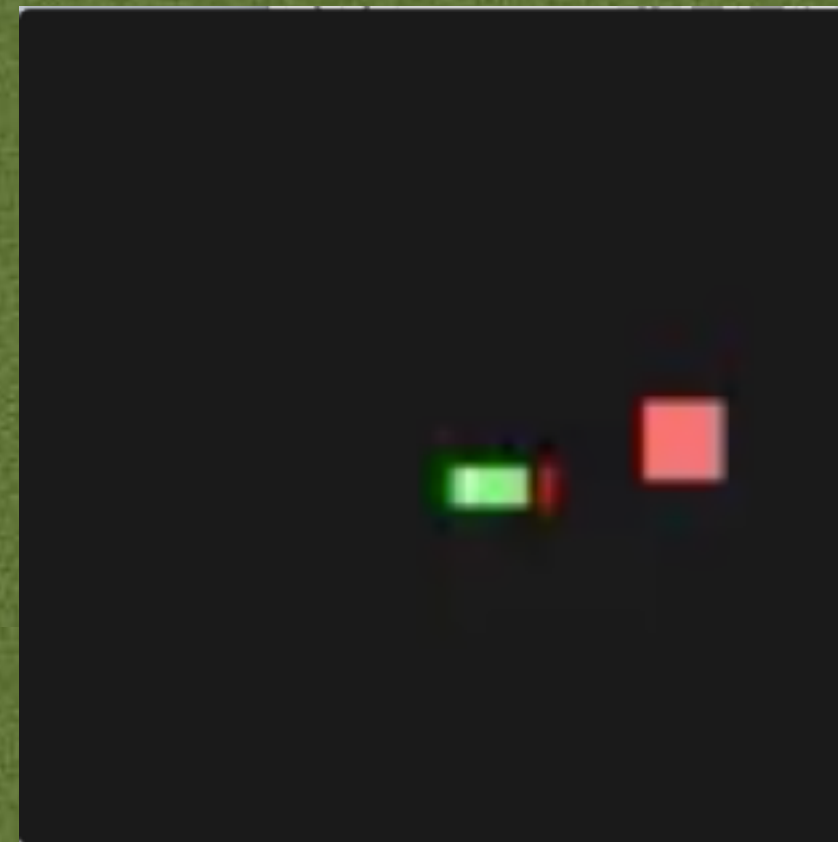


Figure 1. Snake at beginning pose.

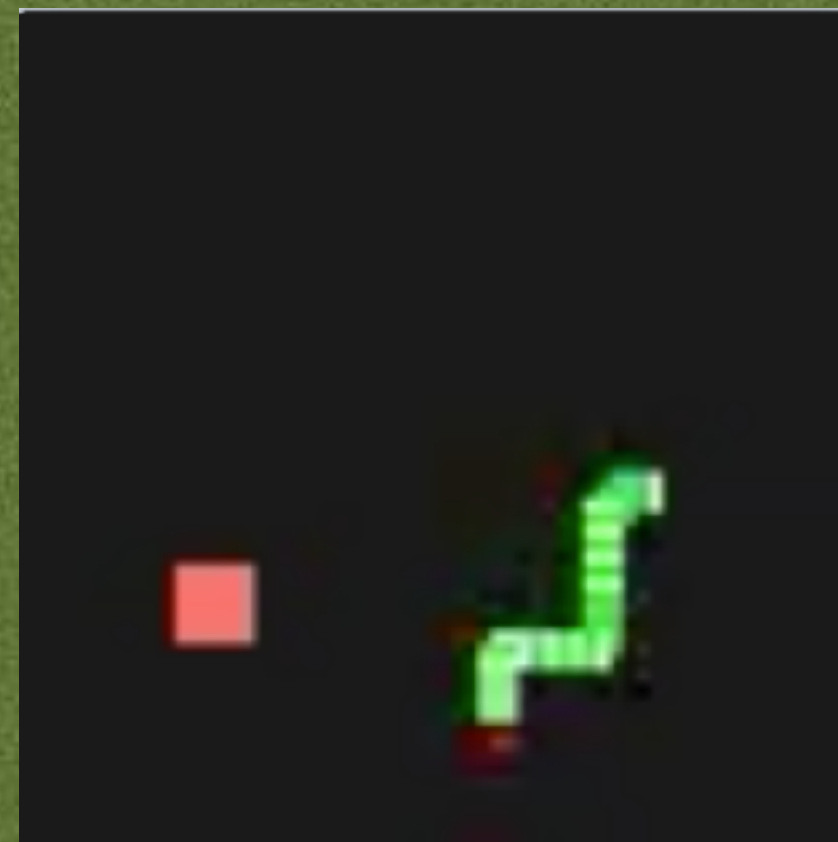


Figure 2. A longer snake.

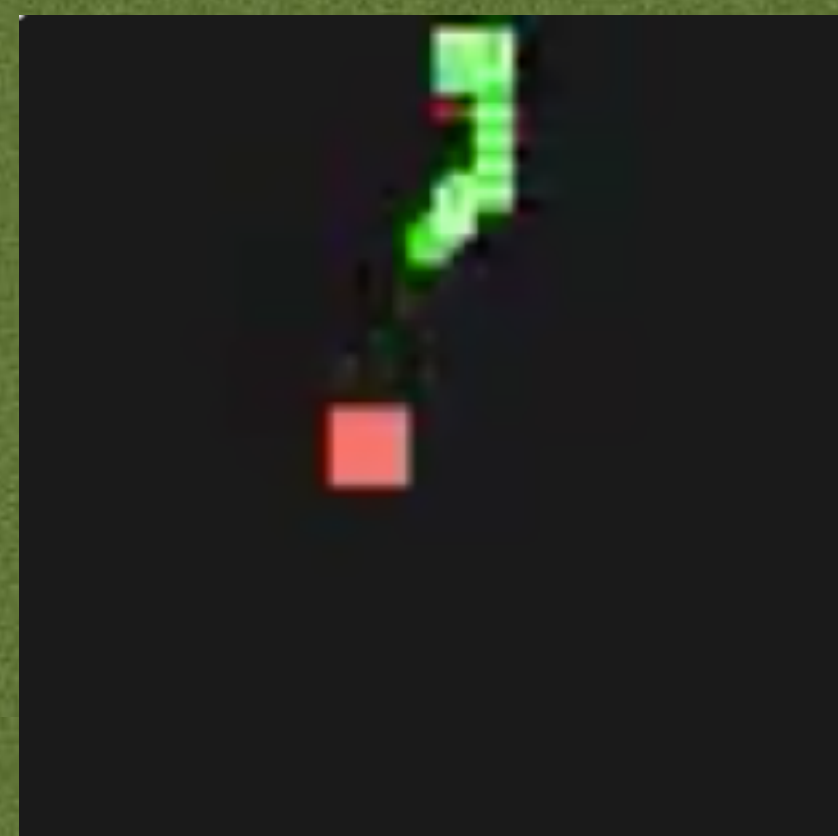


Figure 3. A dying snake

CONTACT

Florian Kleinicke
Heidelberg University
Email: kleinicke@stud.uni-heidelberg.de

Network

- Snake environment Open AI Gym
- Rewards: Food +1, death -5
- Lossfunction: smooth_l1_loss
- Optimizer: Adam
- Learningrate: 0.0001

Improved Features and Learning rates

- Additional Features added to fc-layer
- Position of head, food and distances
- No significant improvements

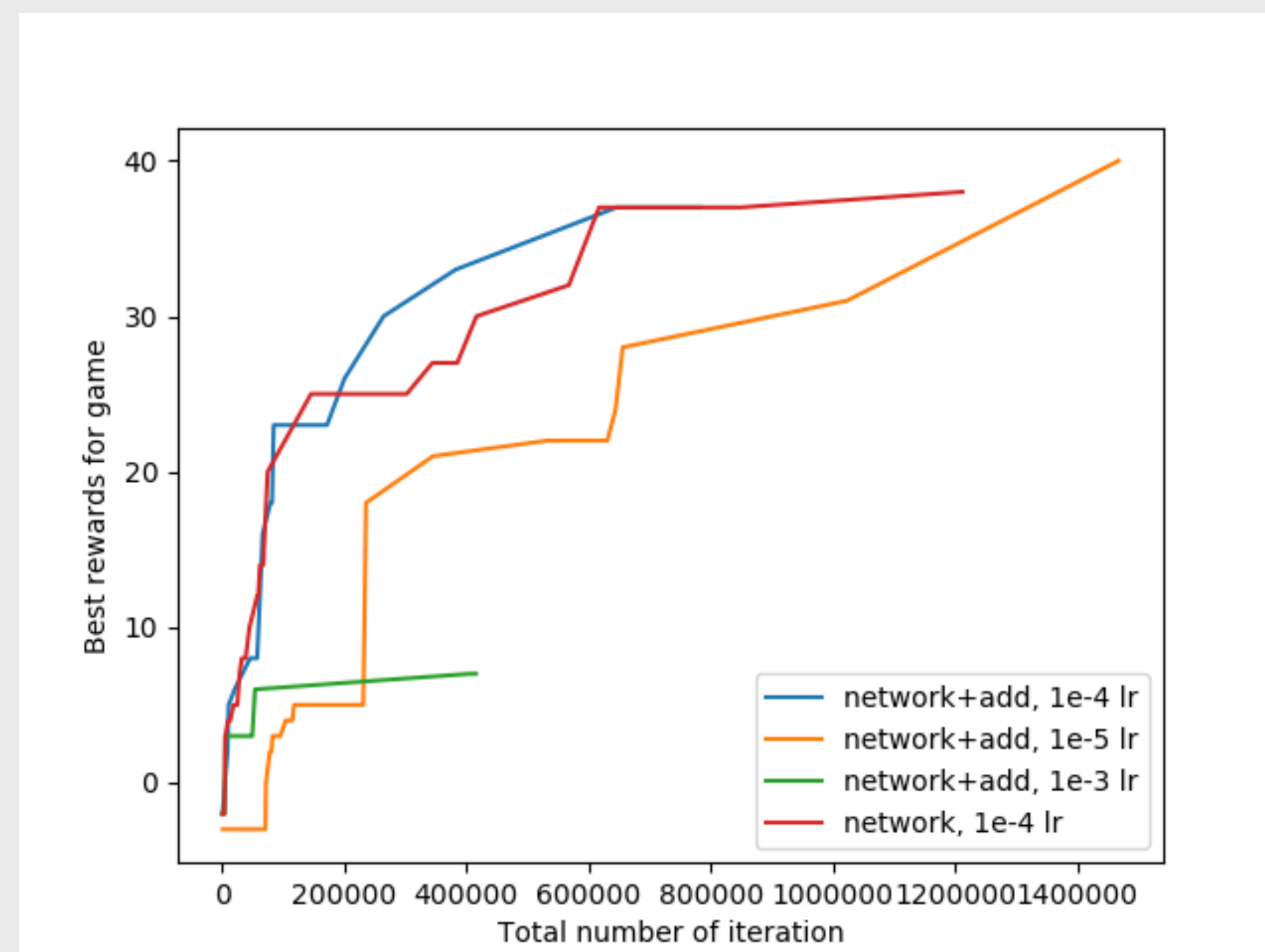


Figure 4. Compare different learning rates and additional features.

First Problems

- First try failed
- Network ran, but no improvement
- Error still unknown

Nonlinear learning progresss

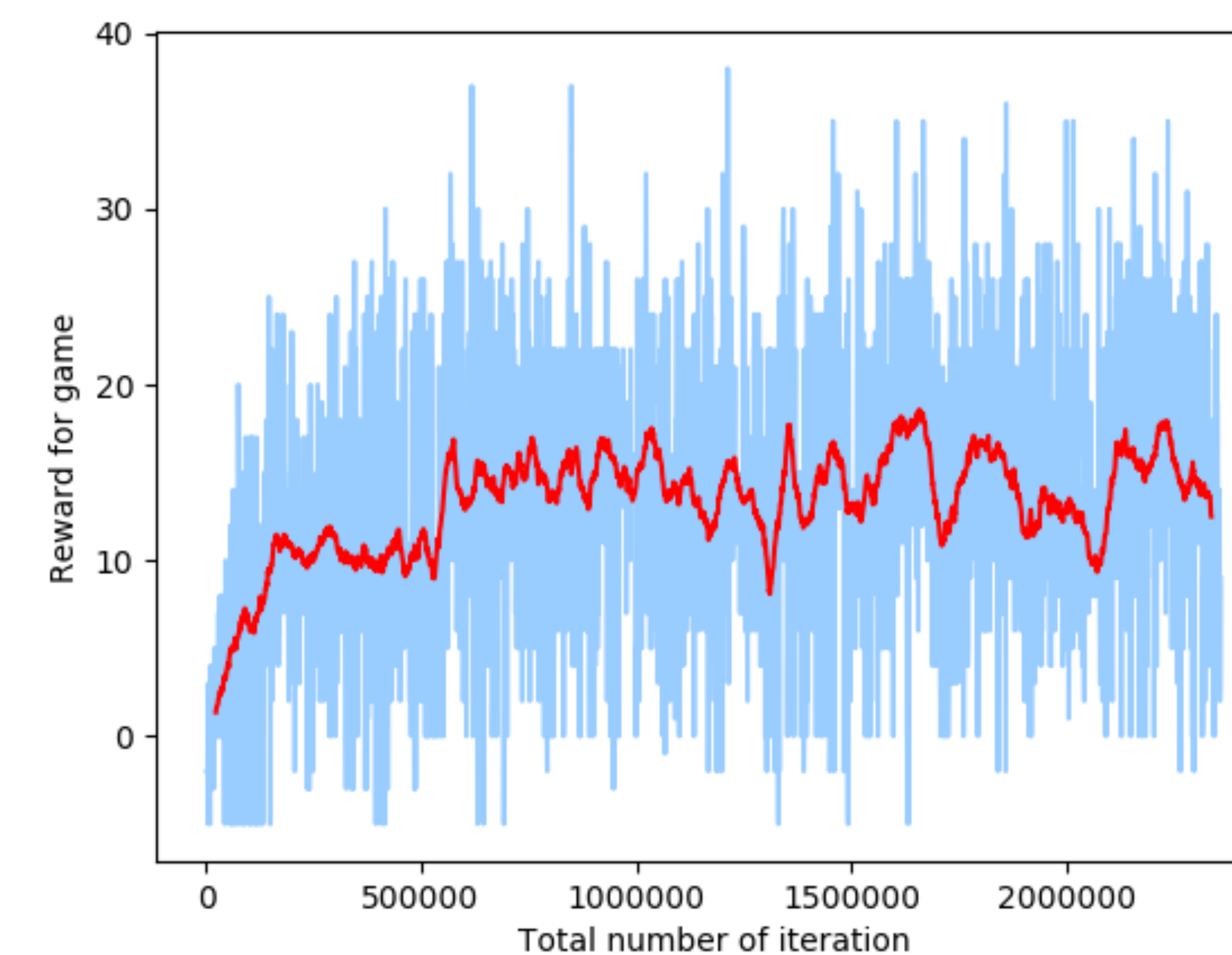


Figure 5. Typical Learning progress (blue) with average (red).

Current Problems

- Long training periods (>24 hours)
- Non-monotonically increasing rewards
- Most experiments with similar results

Multiple NN-Architectures

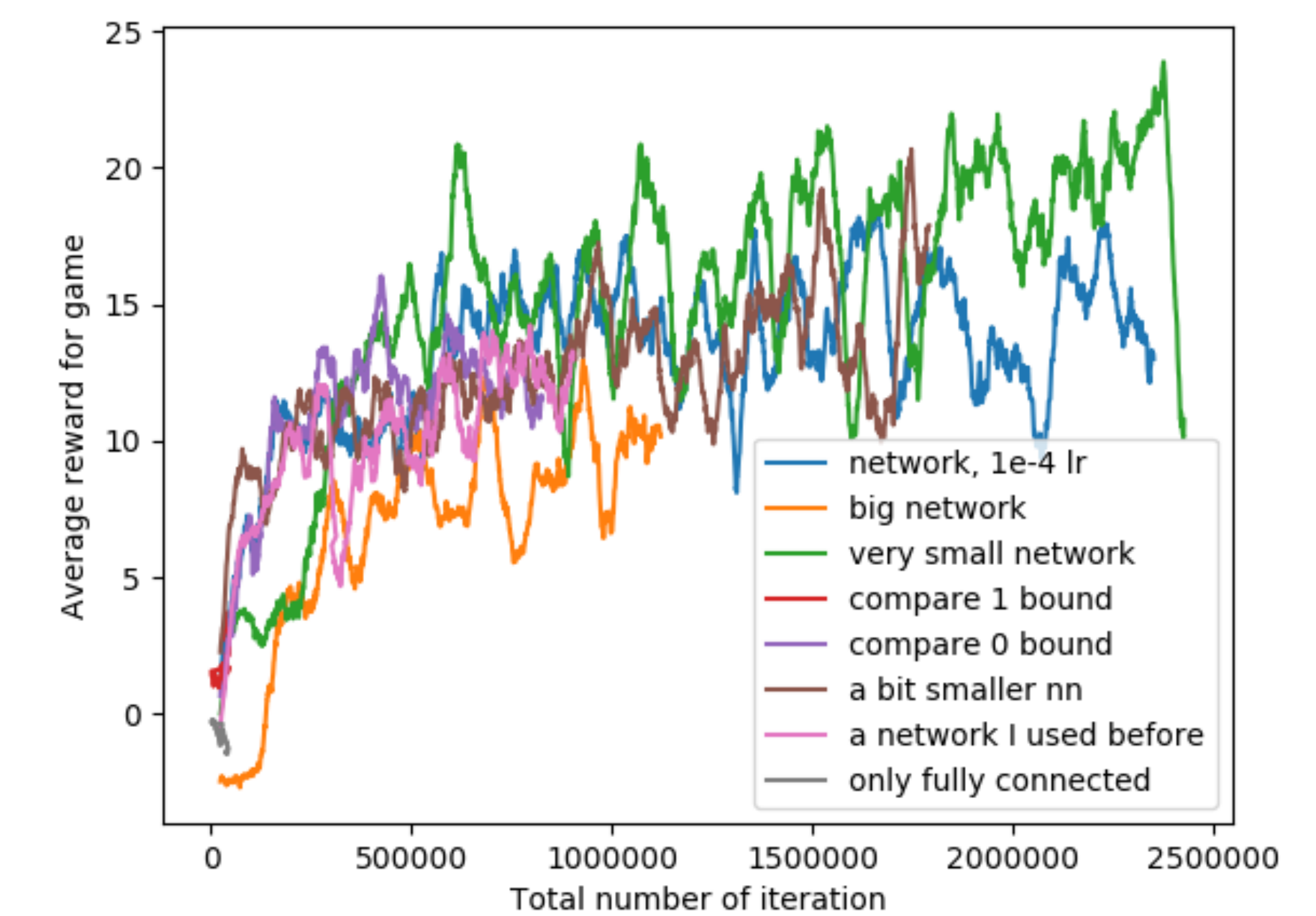


Figure 6. Multiple self designed NN performance compared

- Normal network had 3 Conv. Layer
- Followed by 2 FC- Laver

Next steps

- Try out other than q-learning
- Double q-learning, A3C, ...

REFERENCES

Snake [https://en.wikipedia.org/wiki/Snake_\(video_game_genre\)](https://en.wikipedia.org/wiki/Snake_(video_game_genre))
OpenAI Gym Snake <https://gym.openai.com/envs/Snake-v0/>
Code base <https://github.com/AndersonJo/dqn-pytorch/blob/master/dqn.py>