Assessing Generalization in Deep Reinforcement Learning.

Agent adaptation to a changing environment.

Oleg Dats,
Glusco Dmitri
Students at UKU
Lviv, Ukraine
2019

Importance of the problem

The problem of generalization is the fact that deep RL agents are commonly trained and tested in the same environment and are thus not encouraged to learn representations that generalize to previously unseen circumstances. Imagine you train a robot to walk on different types of ground. After successful validation tests it meets new frozen ground never seen before. The generalization is the tool responsible for successful overcoming such new obstacles.

Related work

We have analyzed the current progress of generalization in Deep Reinforcement Learning field. Charles Packer∗ and Katelyn Gao∗ from Berkeley's presented a good foundation to measure and analyze generalization. We decided to continue their work and follow presented approach. Train on Deterministic and Stochastic versions and test on Deterministic, Stochastic and Extremely Stochastic versions of the same environments. Generalization is a good performance of Stochastic and Extremely Stochastic versions.

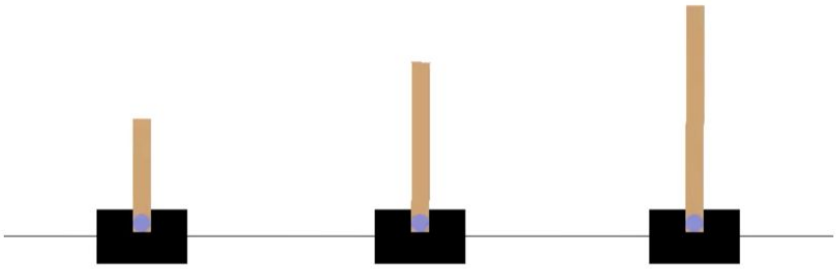
Main work

Unlike all previous work the state of our final env is based on pixels. Famous DQN by Mnih, Nature 2015 was trained and validated on deterministic env (the state of env was always the same). In our work we have added random noise. It means the state space was extended to almost infinity. The environment is always new for the agent.

Convention follows Charles Packer∗ and Katelyn Gao∗ from Berkeley's. D: Deterministic env, R: Stochastic, E: Stochastic with higher variance. DR: trained on D, validated on R.

Due to the hierarchical structure of our work we have decided to break the project on next pipeline:

1. Study RL and develop Q-learning: model free and the most basic value based learning algorithm. Verify on FrozenLake-v0 Gym. (Achieved mean: 0.77)
2. Continues state/action space. It is not possible to use Q-learning for CartPole. We have developed basic DQN to tackle this problem. (DD)
3. Add noise and show poor performance of the agent (DR, DE). Use SunblazeCartPole environments to introduce stochasticity.



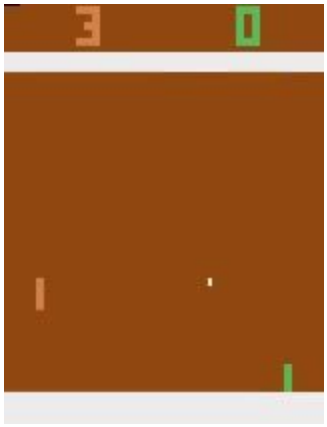
```
Loaded model from cartpole_model_d.pkl
DD 184.36
Loaded model from cartpole_model_d.pkl
DR 134.96
Loaded model from cartpole_model_d.pkl
DE 64.25
```

4. Retrain the agent on SunblazeCartPoleRandomNormal and show it works on RR and RE

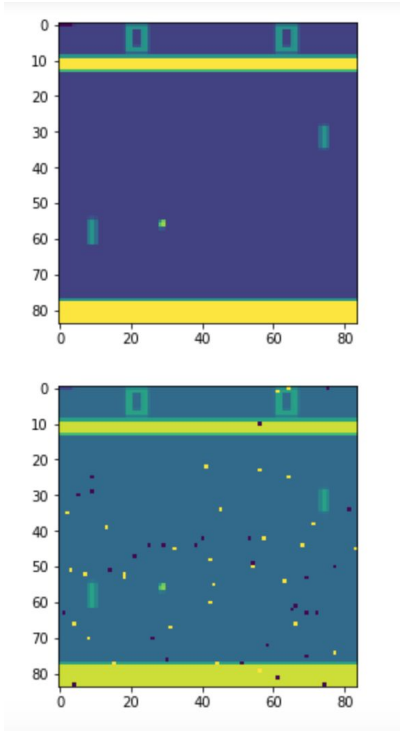
```
Loaded model from cartpole_model_r.pkl
RD 200.0
Loaded model from cartpole_model_r.pkl
RR 200.0
Loaded model from cartpole_model_r.pkl
RE 172.32
```

5. The main focus of our research is to tackle the environment with the state represented by pixels. We have analyzed Atari envs and found that Pong is the most optimal env

(pixels and complexity)



6. New production ready model. We now switch to much more complex task. Due to its complexity and state space represented by pixels we have to move to clouds and train on modern GPU. We have analyzed Google Dopamine and OpenAi Baselines as a perfect implementation of Dueling DQN and decided to go with OpenAi Baselines implementation. We had to extend OpenAi/baselines/deepq/deepq.py module to support our Stochastic environment.
7. Train agent and show it converges to a good policy. For Pong avg result ~20.
8. Develop basic noise algorithms v1 and show it can not play anymore.



9. Train on Stochastic env and show it takes to much time.
10. Develop optimized noise algorithms v2 and show performance improvements. Drop for one frame: 0.01 to 0.003. Almost 30 times less.

11. Train on Stochastic env and show satisfying performance for the Stochastic Pong env (RD,RR)

```
Loaded model from models/pong_model_d.pkl
DD 18.5
Loaded model from models/pong_model_d.pkl
DR -20.3
Loaded model from models/pong_model_r.pkl
RD 18.1
Loaded model from models/pong_model_r.pkl
RR 17.7
```

Summary

Training procedure is much more important than complicated algorithms. It confirms findings from previous researchers. As additional findings I want to highlight the importance of how the stochasticity procedure is implemented. Usually the training of simple Atari environment needs at least 1mln frames. Even a small delay in integration stochasticity can make training last for infinity.

The list of main sources

Reinforcement Learning: An Introduction, by Richard S. Sutton and Andrew G. Barto
https://bair.berkeley.edu/blog/2019/03/18/rl-generalization/
https://openai.com/blog/quantifying-generalization-in-reinforcement-learning/
https://openai.com/blog/openai-baselines-dqn/
https://deepmind.com/research/dqn/