Curiosity-driven Variational Autoencoder for Deep Q Network

Gao-Jie Han, Xiao-Fang Zhang, Hao Wang, and Chen-Guang Mao

School of Computer Science and Technology, Soochow University, Suzhou, China

xfzhang@suda.edu.cn

Outline

1.Introduction

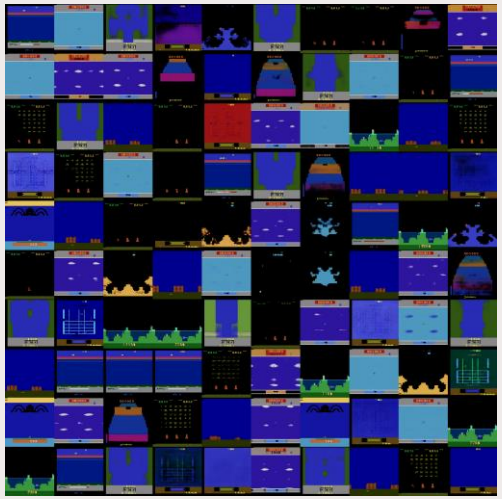
2.Curiosity-driven Variational Autoencoder

3.Experiment

4.Conclusion

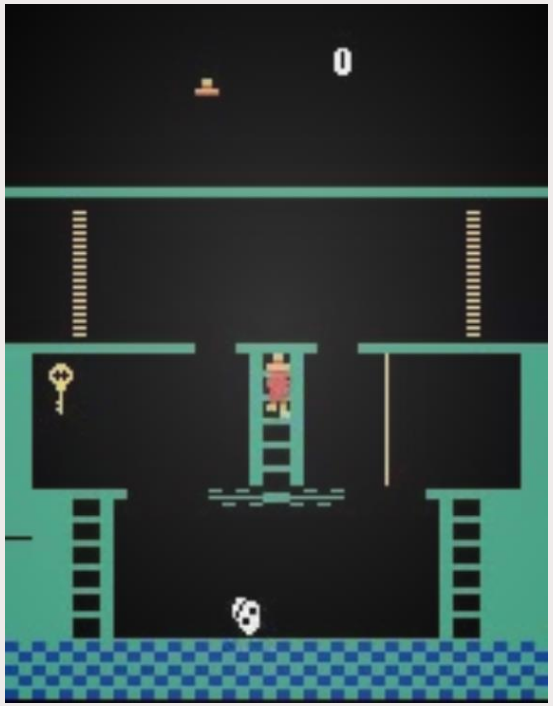
1.Introduction

- Deep Reinforcement Learning(DRL) has tremendous success across various fields.
- In some scenarios, the training sample is difficult to obtain.
  - DRL algorithm needs millions of training samples, but in some scenarios, the training sample is difficult or time-consuming to obtain.



1.Introduction

- Simple heuristic exploration strategies often leads to insufficient exploration.



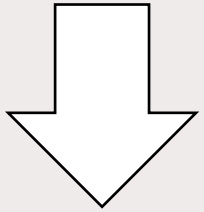
Montezuma's Revenge

- The Deep Q Network(DQN) relies on simple ε-greedy strategy, which often leads to insufficient exploration;
- For example, the *Montezuma's Revenge*(left) game has 24 scenes, but the DQN algorithm with ε-greedy strategy only explore 2 scenes in general.

1.Introduction

Existing Work

- Some researcher attempts to represent the actual environment by generative model to improve the sample efficiency.



Limitation

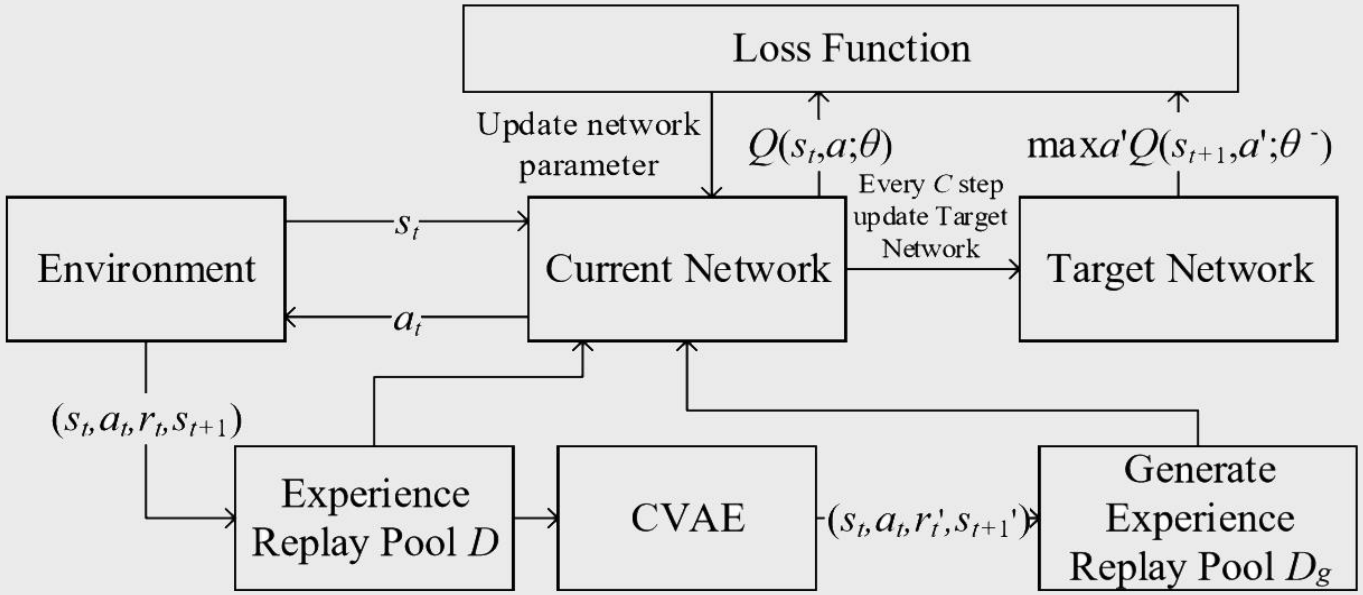
- The generative model may become inaccurate if the agent doesn't make a sufficient exploration.

1.Introduction

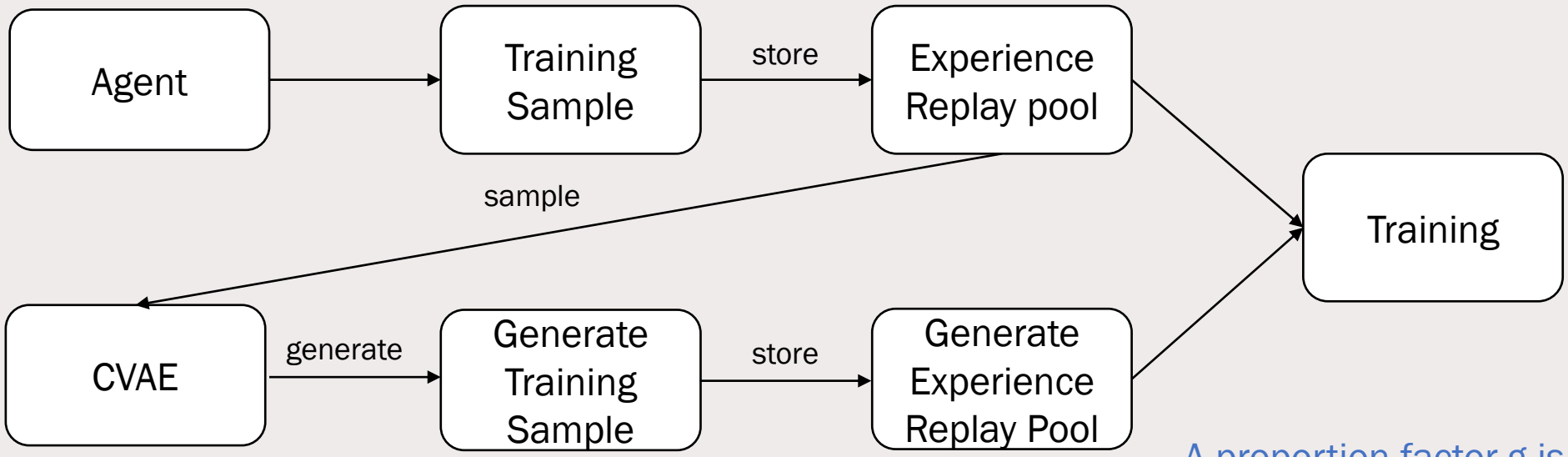
- In this work, we propose a new algorithm called Curiosity-driven Variational Autoencoder(CVAE).
- CVAE model uses a generative model to represent the actual environment and generate training samples. Then, the model uses an intrinsic reward to drive the agent to explore the unexplored region, which can improve the quality of the generate training samples.

2.Curiosity-driven Variational Autoencoder

Overview of CVAE



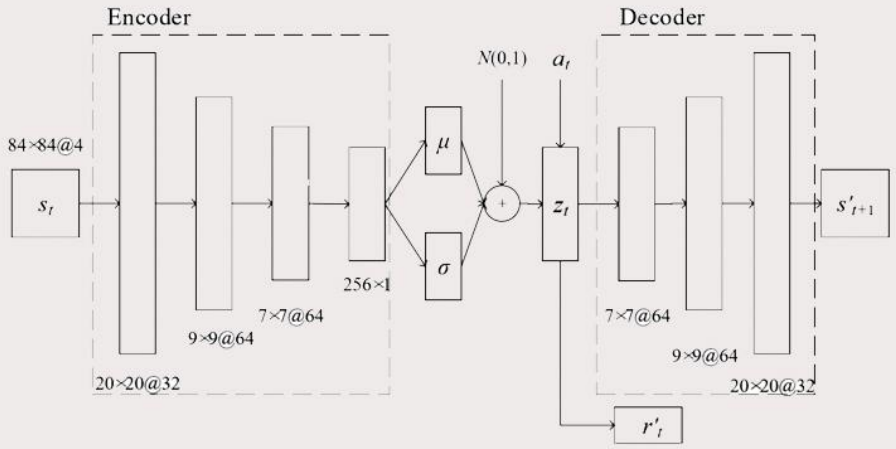
2.Curiosity-driven Variational Autoencoder



Flow Chart for CVAE

A proportion factor g is used to control the ratio of the two experience replay pools.

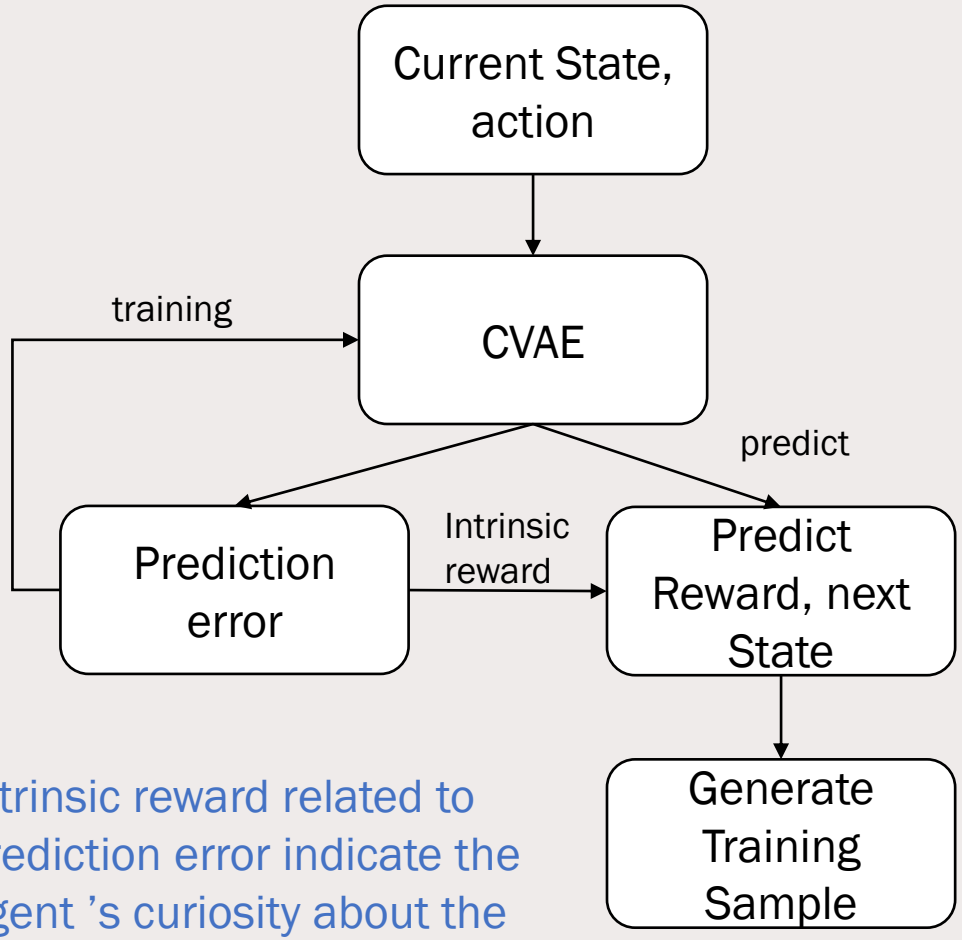
2.Curiosity-driven Variational Autoencoder



There are 4 convolutional layers, 2 full-connected layers and 4 deconvolutional layers in CVAE model.

The 84×84@4 means input size is 84×84, 4 channel.

Use the KL divergence as prediction error



Intrinsic reward related to prediction error indicate the agent 's curiosity about the training sample.

2.Curiosity-driven Variational Autoencoder

Usage of CVAE

Algorithm 1 DQN-CVAE

Initialize replay memory D with capacity N, generate replay memory D_g with capacity N_g, minibatch size M, proportion factor g
Initialize the action-value function Q with random weight θ
Initialize the target action-value function Q with weight θ^-
for episode=1, I **do**
  Observe state s_0
  for t=1,T **do**
    Choose an action a_t based on ϵ-greedy policy
    Observe transition(s_t, a_t, r_t, s_{t+1})
    Store transition(s_t, a_t, r_t, s_{t+1}) in D
    /*CVAE part*/
    Sample random minibatch of transition (s_t, a_t, r_t, s_{t+1}) from D
    Generate transition $(s_t, a_t, r'_t, s'_{t+1})$
    Compute the prediction error e_t
    Store transition $(s_t, a_t, r'_t + \beta e_t, s'_{t+1})$ in D_g
    /*DQN part*/
    Random sample $M \times (1-g)$ of transition (s_j, a_j, r_j, s_{j+1}) from D
    Random sample $M \times g$ of transition (s_j, a_j, r_j, s_{j+1}) from D_g
    Set

$$y_j = \begin{cases} r_j & \text{if episode terminates at step } j{+}1 \\ r_j + \gamma max_{a'} Q(s_{j+1}, a'; \theta^-) & \text{otherwise} \end{cases}$$

    Perform a gradient descent step on $(y - Q(s, a; \theta))^2$ with respect to the network parameters θ
    Every C step update $\theta^- = \theta$
  end for
end for

Generate a training sample from CVAE

Proportion factor g is used to control the ratio of the two experience replay pools.

3.Experiment

- Research Questions
  - RQ1: Does the DQN-CVAE improve the performance of the DQN?
  - RQ2: Does the DQN-CVAE improve the performance of other DQN variants?
  - RQ3: How does the proportion factor g affect the performance of the DQN-CVAE?

3.Experiment

- Experiment Environment
  - We use Atari 2600 game as experiment environment, and 5 games were selected in our experiment: Alien, Beam Rider, Kangroo, Seaquest and Space Invaders.

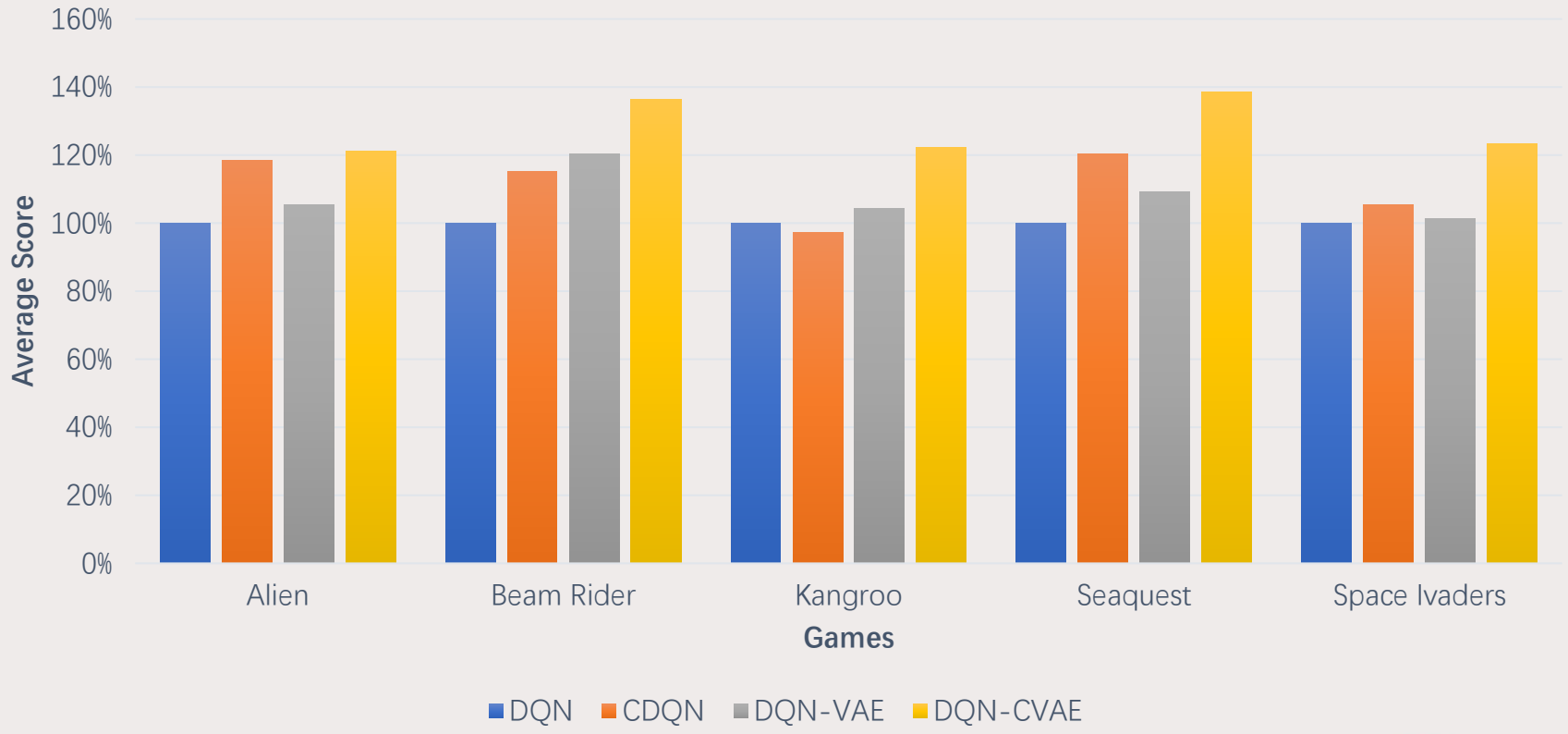
Game	Action Number	Introduction
Alien	18	Agent avoids enmenies and reach the target point
Beam Rider	9	Agent avoid bullet and hit moving enemies
Kangroo	18	Agent climbs through stairs and avoids obstacles
Seaquest	18	Agent evades obstacles and attacks enemies under water
Space Invaders	6	Agent evades and attacks the enemies

3.Experiment

- Experiment Setup
  - We use 200 epoch as the training periods, 100,000,000 steps are trained.
- Comparison Algorithms
  - DQN: benchmark algorithm;
  - CDQN:DQN with curiosity-driven exploration;
  - DQN-VAE:DQN with a VAE model to generate training samples;
  - DQN-CVAE:DQN with a CVAE model.

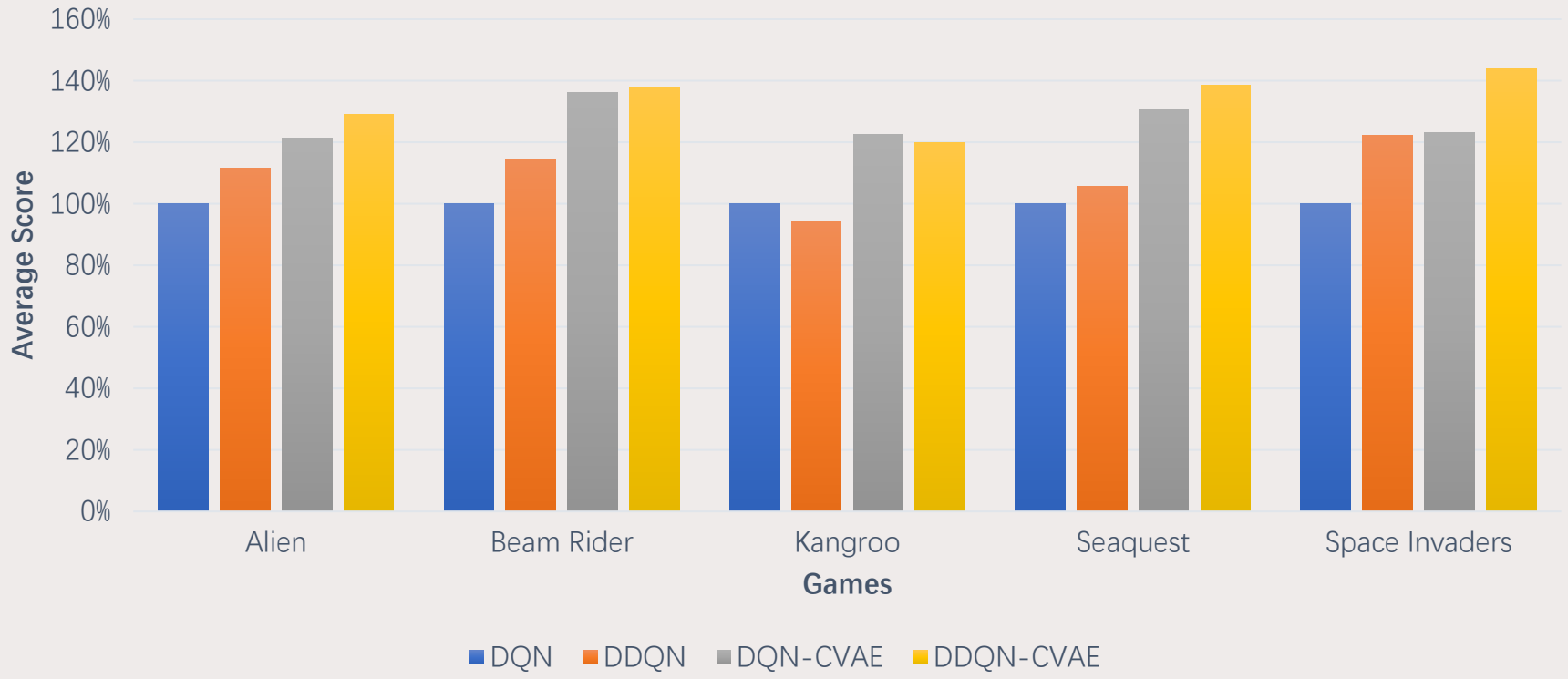
3.Experiment

RQ1: Does the DQN-CVAE improve the performance of the DQN?



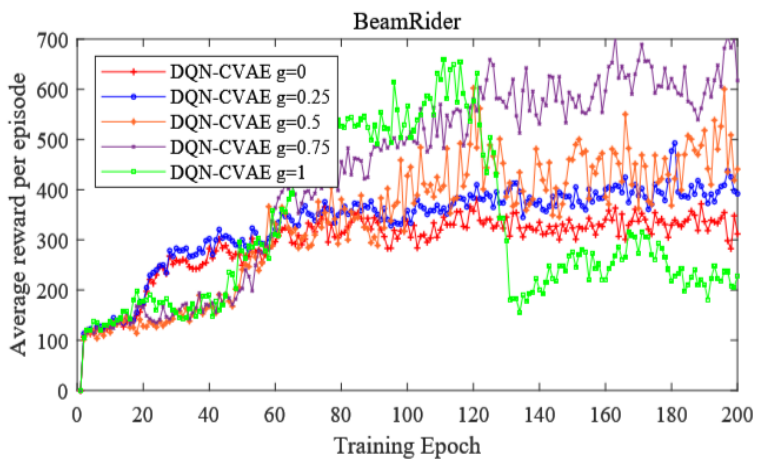
3.Experiment

RQ2: Does the DQN-CVAE improve the performance of other DQN extensions?

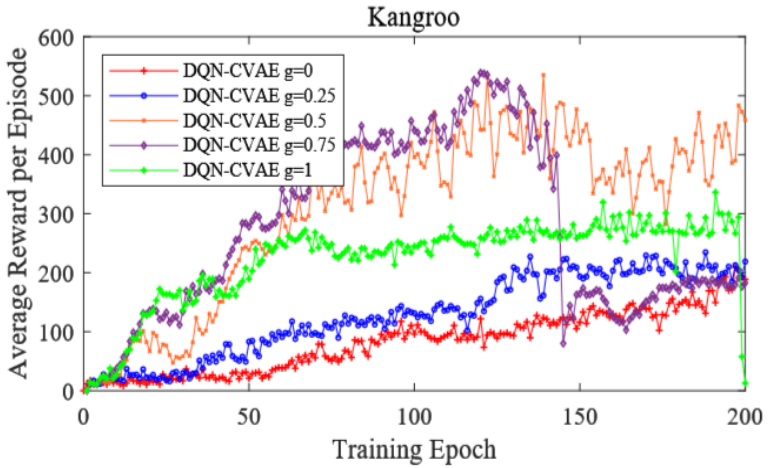


3.Experiment

RQ3: How does the proportion factor g affect the performance of the DQN-CVAE?



(a) BeamRider



(b) Kangroo

4.Conclusion

- The CVAE algorithm can improve performance of the agent.
- The CVAE algorithm can be easily applied in model-free DRL algorithms.
- In furture work
  - the priority can be used to select to generate training samples;
  - make g become a dynamic learnable parameter with the use of neural networks.