

# 1 Overview

This project demonstrates the use of deep Q-Networks for reinforcement learning. Reinforcement learning is a branch of Machine learning where an agent is trained to correctly behave in a particular environment. The Agent interacts with the environment by choosing some action  $a$ . In response to the action the environment returns a new environment state  $s$  and a reward  $r$ . The goal of the agent is to maximize the cumulative reward.

In this particular project the agent uses Q-Networks to be able to easily deal with a big set of possible states  $S$ . Q-Network's are deep neural networks that serve as a nonlinear function approximator and are trying to find the best action value function. This means the Q-Network tries to estimate the expected sum of future rewards (discounted by  $S'\gamma$ ) *for a given state action pair*.

Due to the unstable nature of nonlinear functions there are multiple additions of the Deep Q-learning Algorithm to increase the stability. This project implements the following two:

**Replay Buffer:** Use a buffer to store  $N$  most recent Experiences  $(S, A, R, S')$  tuples). Sample  $n$  random elements from the buffer at  $x$ th every time step and learn from them.

**Double DQN:** Use two neural networks to select the best action in order to prevent the algorithm from propagating incidental high rewards received by chance.

## 2 Results

## 3 Ideas for future improvements