

define reinforcement learning

Important terms in reinforcement learning

Agent = the learner and decision maker, Agent (neural network) will receive state of the game

environment = the thing the agent will be interacting with (here level 1-1)

state = a snapshot of the environment at a given time (what agent sees)

state will be 4 no of consecutive snapshot frames from the game to track motion of the enemy is coming towards it or moving away

action = the agents available inputs for the environment (like controls of the game)

eg: no button pressed, right, A= jump-longer it's held the higher Mario will jump and then B is used to move faster

Agent gets negative "-1" point for each sec of the game timer if it stands still in one place, it is done this is to disincentivize the agent from standing

still and then the agent will get a massive negative "-15" points if it dies

once the environment gives the agent a state and the agent responds with an

action to the environment also gives the agent a reward a reward essentially

helps the agent determine whether or not the action it took inside of that state was good or bad

reward = A numerical value given by the environment to the agent to indicate how good the agents action was

episode = A sequence of interactions from start to terminal state

policy = A function mapping states to actions

ddqn algorithm implementation we'll be using the Epsilon greedy approach

value function = value function takes in a state and returns

how valuable that state is in reinforcement learning it's natural to

view some states as more valuable than others for example the state where you're right next to the flag is probably a little bit more valuable than the one where you're falling down a hole we're not going to be working directly

with the value function

Action value function = A function that returns the value of a state,action pair, it just return how valuable the pair is.

Replay-Buffer = It is essentially a storage for our past experience and reward to train the neural network

Here epsilon value at start will be 1 to have maximum exploration scope for the agent to try and learn and its value gradually decreases from 1 but remains greater than 0, to have scope for agent to keep exploring

two types of network:

1 online network that will be actually be trained

2 target network the one we'll be using for the ground Truth for our predictions to give to our loss function the target

Network won't be trained however we'll be intermittently copying the weights from the online network over to the

Target Network this is to increase stability during the training process

gym super mario bros

wrappers

tensordict

PyTorch is a Python package that provides two high-level features:

Tensor computation (like NumPy) with strong GPU acceleration

Deep neural networks built on a tape-based autograd system