**Task 3**

**Solution :-**

* For manual policy, I have used very simple logic and that is, try to minimize the dz value as much as possible by taking the suitable action. So, if relative distance is negative then go to the left side and if relative distance is positive then go to right side else if relative distance is very close to 0 then don’t do anything.
* For RL solution, I have done several tasks:
  + Discretization of state space as it is required.
  + Selected a good TD model. As per my experience DDQN is good model and it is very advance model So, I finalized this one.
  + Implemented the DDQN model code via tensorflow library.
  + Tuned the parameters of DDQN model like gamma, epsilon, etc
  + Reward function shaping for giving the direction to the agent.
* I just trained my agent for just 170 episodes as the number of episode doesn’t matter in this task. I tried to capture whether my agent is learning or not.
* For this, I created a training reward plot

Chart, scatter chart

Description automatically generated

* As we can see that, the cumulative reward is increasing after each episode which says that our agent is getting improve after some episodes.
* Test average = 82.64
* Test Standard deviation = 11.65
* My training time was around 5-7 mins.
* If we increase the no. of states, then there are 2 possibilities:
  + It will improve the performance of agent
  + It will degrade the performance
* And it depends on the state whether your performance will degrade or improve. If state doesn’t have any meaning in taking decision then it will act as a noise and can degrade the performance and if state is playing a key role in decision making process then, it will upgrade the performance. In our case, if we increase the level of the environment, the additional state i.e. Y coordinate of the ball has meaning in decision making So, it will upgrade the performance.