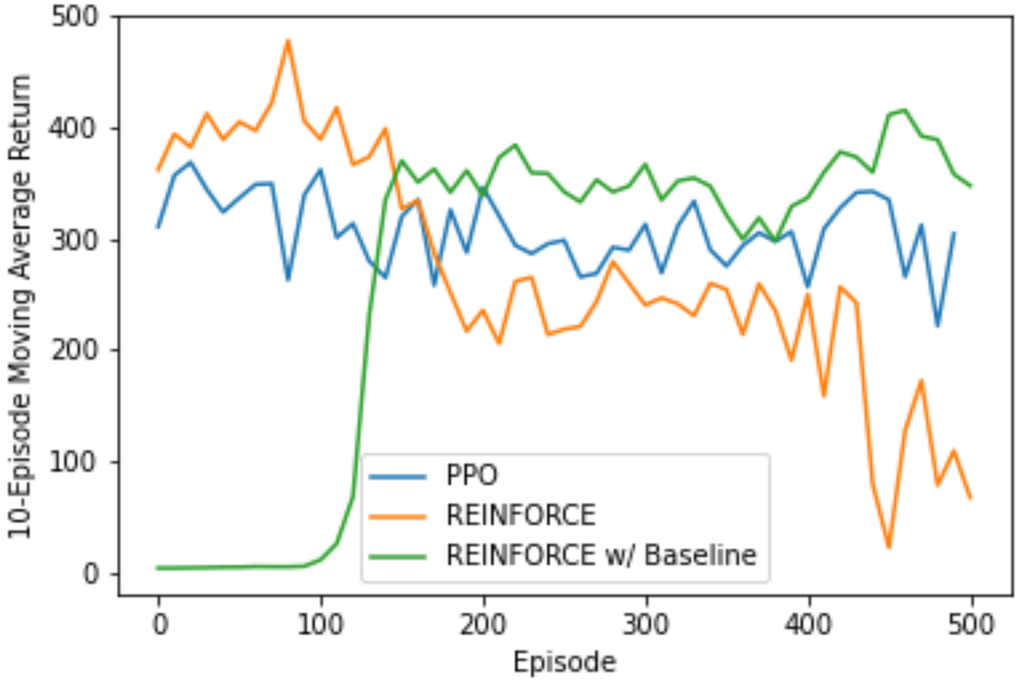
Deep Reinforcement Learning – Homework 1

To begin, I had some medical issues and so I was only able to begin working on this on this past Thursday. The only issue this cause me was that I didn't really have time to mess around with parameters and Network Architectures to try and improve the training I was doing on the environment. Due to this, my results are a little strange, but I am going to discuss them both practically and theoretically to demonstrate my understanding. I also apologize that my code is not cleaned up very well, I normally would take the time to really clean up the syntax and organization of my code, but I ran out of time trying to get the PPO agent made correctly.

FIGURE:



DISCUSSION:

The first thing I noticed, which makes sense both practically and in theory was the variance of the rewards/returns I was seeing by episode for each of these algorithms. The REINFORCE implementation varied a lot through the 500,000 steps and the REINFORCE with a baseline varied a fair amount as well, but not to the extent of the original REINFORCE algorithm. This makes sense, as the baseline adds a level of stability to the improvements being made using the value function. By using the baseline, the updates to policy are limited in their extremity thus lowering the variance in the rewards/returns for different episodes.

Then, the PPO returns varied even less and seemed much more stable than either of the REINFORCE algorithms. It stays relatively near the previous values, but when it fluctuated to a lower return in one episode it would bounce back up a little, seemingly back towards the higher reward.

The agents are exploring their action and observation spaces based on an optimization, or loss, function to point them towards the most beneficial actions. However, these spaces are highly complex and finding the optimal combinations of actions can take a very long time. Then, the learning is occurring in the Networks for the value and policy functions so the learning rate, number of nodes, and number of hidden layers also can have a huge impact. I think not being able to explore different networks is the biggest reason for my results not aligning with the expected theoretical result. Theoretically, the REINFORCE algorithm would learn the slowest (inefficient with data and high variance), the REINFORCE with baseline would be slightly faster (still inefficient, but less variance) and PPO would be the fastest (more data efficient and lower variance).