**Question 4**: Briefly summarize your experience working on this project. You can use the following prompts for ideas.

* What was the hardest part of the project? (e.g. getting started, plotting, specifying the task, etc.)
* Did you find anything interesting in how the quadcopter or your agent behaved?

**Answer**: Understanding how to implement the DDPG and Actor Critic algorithm was a challenge for me. I replayed to study the lecture videos numerous times, and worked to understand the examples. Reading articles on towardsdatascience was very helpful in providing understanding. This combined with the time to train the agent made this project the most challenging in the machine learning nanodegree.

After getting the code to work, I estimated I spent a lot of of the time tweaking *the hyper*-parameters, then stepwise changing the NN units, and then modifying the reward function by running the training again and again.  I persisted and landed on a set of hyper-parameters that produced the best results amongst all the training episodes. Finding the right combination seemed very elusive.

It was interesting to see the results when changing the reward function. The reward function is the key for a successful learning rate. After trying different methods to normalize and to incentify right behavior correctly, I noticed a drastic change in the learning curve of the agent. At first, it seemed random, but after applying different incentives, the performance increased in a satisfying manner.

What I used as an indicator of performance was the plot show how the quadcopter performed during the simulation. Changing the reward parameter affected the stability of the performance dramatically after 200 episodes in my early testing and then after 1250 episodes in my later testing.

It would have been helpful to have a deeper understanding of the DDPG algorithm prior to attempting this project.

Methods that could be done to help greatly improve the hyperparameter tuning and reward function design would be to use a quadcopter visualization script. A python visualization method can be found at <https://github.com/AtsushiSakai/PythonRobotics> under Aerial Navigation <https://github.com/AtsushiSakai/PythonRobotics/tree/master/AerialNavigation/drone_3d_trajectory_following>