# CS 747 Assignment 3

Sheel Shah, 19D070052

Task 1

For this task, I discretized the continuous valued states by splitting the 2d space into grids, and assigning a Boolean feature which was one only for the grid in which the state is in, and zero for the others. There were 20 partitions along x, and 10 partitions along v, implying the state vector was of length 200.

The training plot generated is as follows:
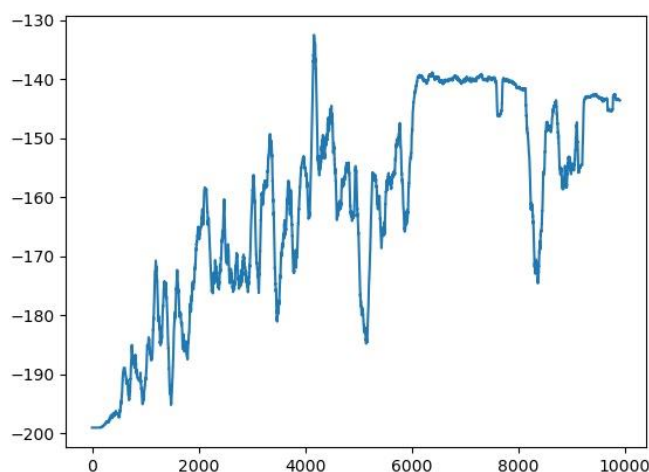
As is visible, the average reward steadily increases, and then it stabilizes around the value of -140. The testing gave an average score of **-142.86** over 100 runs, which is well above the -160 threshold of the problem statement.

Observations:

I noted that the 20x10 discretization worked best. I initially tried a 100x20 split but that resulted in very little training, since there were too many weights to be learned for 10000 training episodes. Trying a 10x5 discretization resulted in a worse final reward, and hence I settled for 20x10. Epsilon is chosen to be 0.01, and the learning rate is 0.1.

Task 2

For this task, I discretized the continuous valued states by using RBF coding. 20x10 RBF centers were uniformly spaced across the 2d space, and when coding some (x, v) to its state vector, the value of each feature was taken to be gaussian RBF with each of the centers. Hence the centers that were close to the point led to a significant feature value, whilst far away centers had very small feature values.

This is the training plot generated:
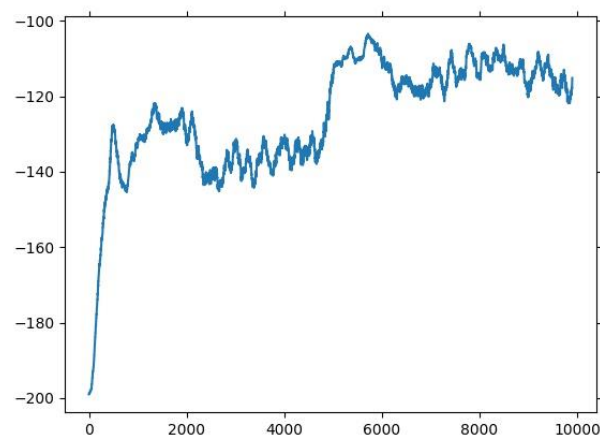
As is visible, the reward very quickly improves to match the best-case value of task 1, in under 1000 training episodes. Then the improvement slows down before there is another bump at 5000 episodes, after which the reward settles close to -110, which again satisfies our requirement of it being at least -130. The testing score was **-101.13**.

Observations:

We see that the state representation is just as long as in task 1, but since the features are no longer Boolean, and there is more generalization, the learning is much faster in the beginning. The scaling factor for the gaussian RBF was chosen to be 100, as this resulted in only nearby centers to contribute to the net state, hence not over generalizing. Epsilon was chosen to be 0.03, and the learning rate was 0.01.