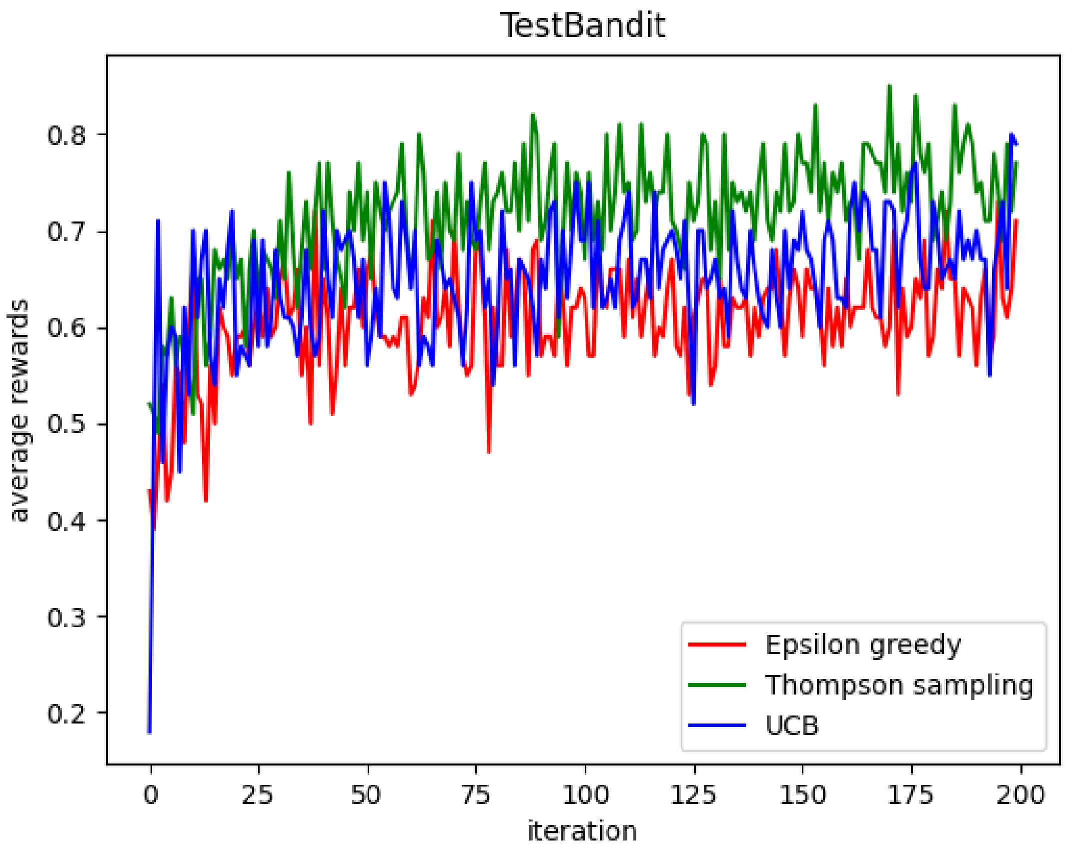
Assignment 2
CS885
Instructor: Pascal Poupart

Qiqin Fang
student number: 20625251

Part I

1.

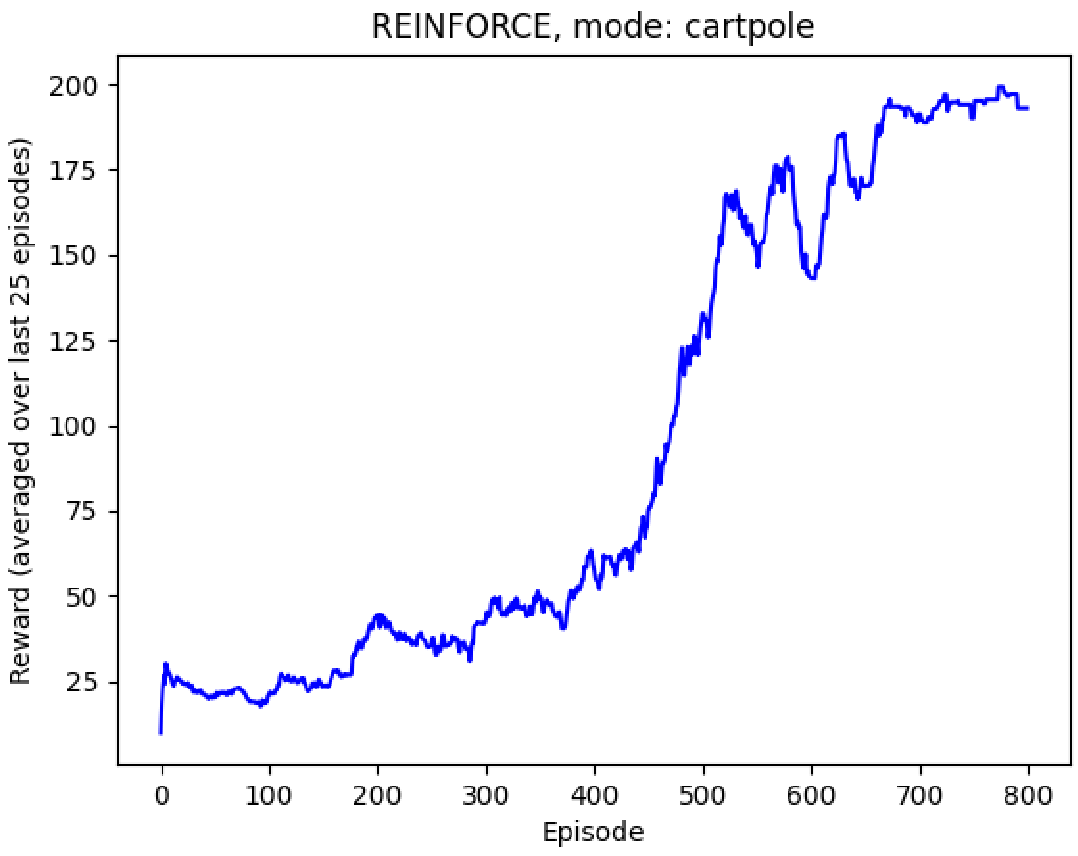


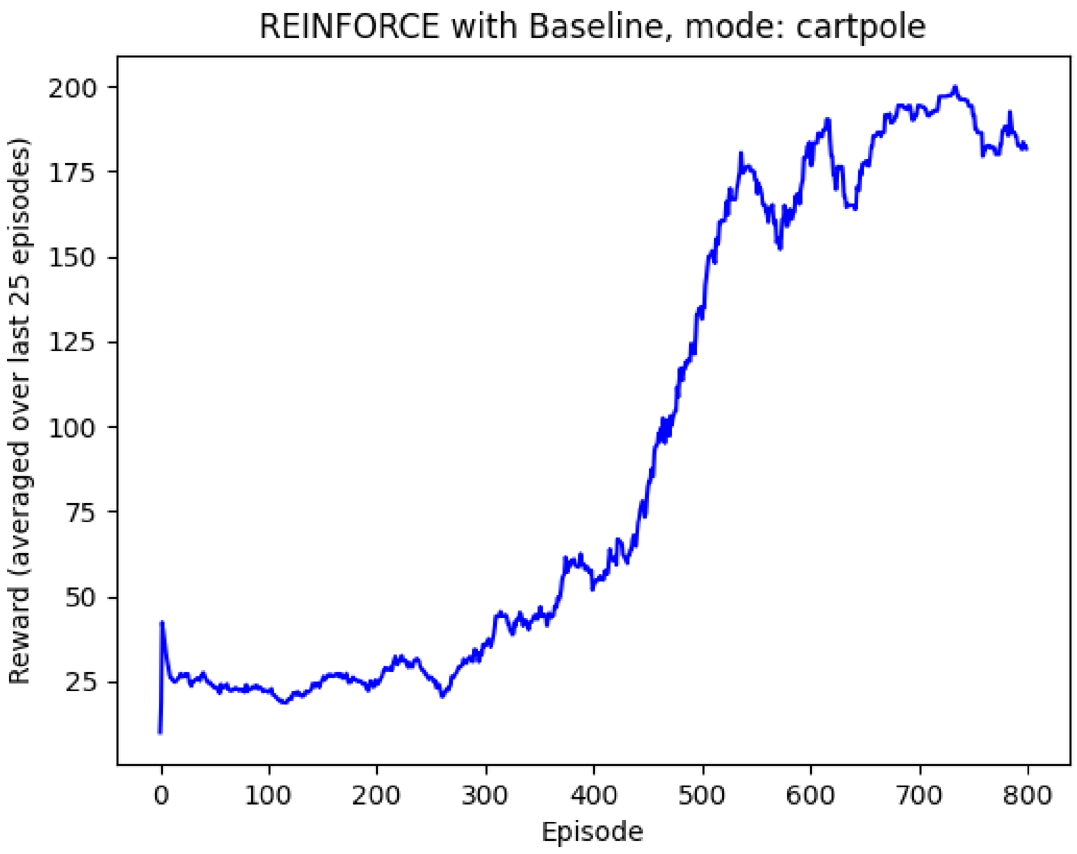
2.

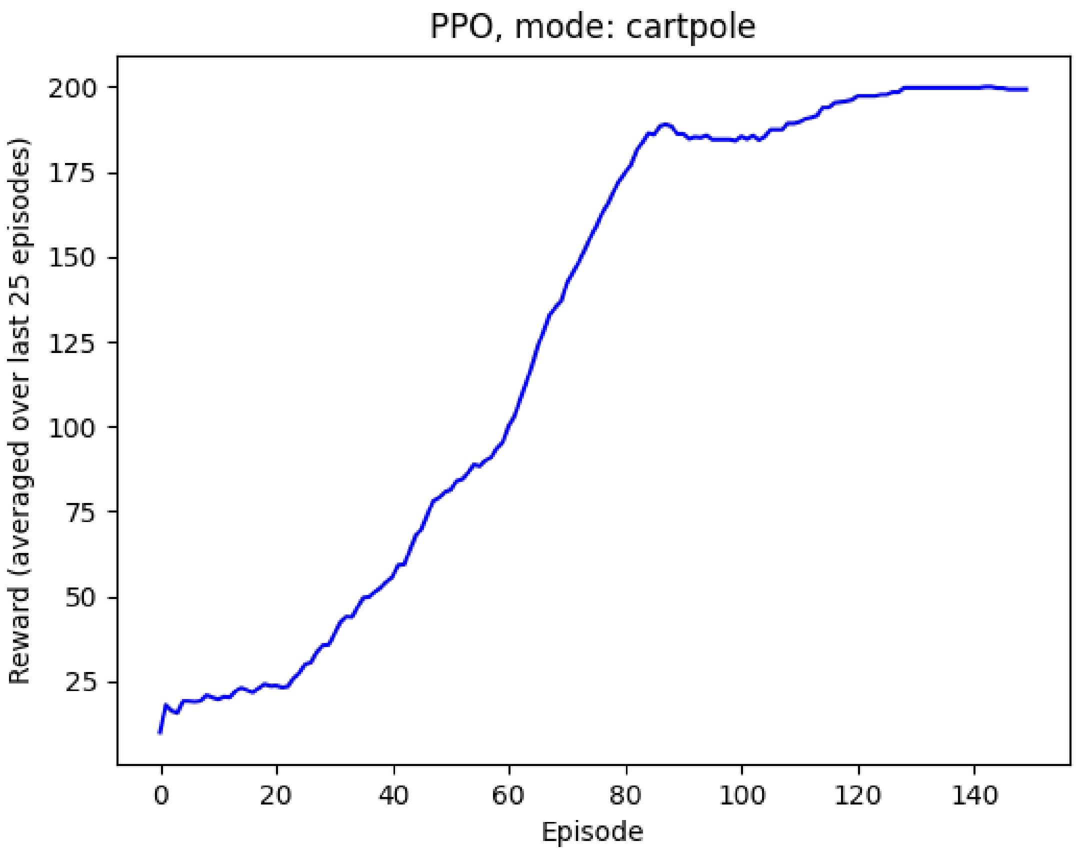
The result is not surprising. Epsilon greedy eventually reach to a point where it is very unlikely to do exploration, so it may get stucked at a local minimum if the rate of decent of epsilon is too quick. For UCB, the benefit is that the an action will always be chosen at some point because 2 * log(n) / na will become very large at some point. On the downside that it may has longer convergence time because we are not choosing the action which brings us the best reward but an upper bound estimates, and that leads to more exploration, depending on the setting of the upper bound. For Thompson sampling, it seems like it perform the best in this case. It sample the emperical mean based on prior distribution and update the prior distribution based on the observed reward. When the sample size is low, it will try to explore other actions (as it will be less likely to sample to the mean of the best action), However, once we have enough sample, it became more certain which action is the best to take, so it will more and more likely to stick to the best action required.

Part II

1.




REINFORCE with Baseline, mode: cartpole
Reward (averaged over last 25 episodes)
200
175
150
125
100
75
50
25
0
100
200
300
400
500
600
700
800
Episode

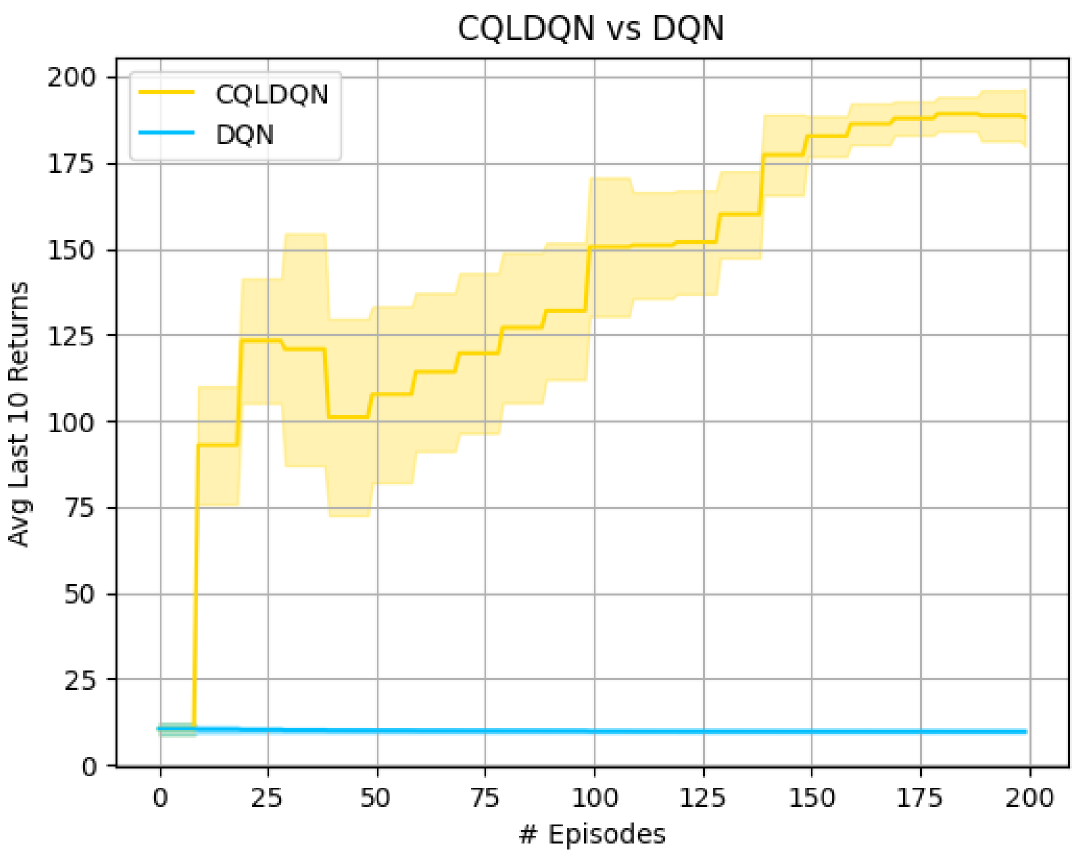


2.

In comparison between REINFORCE with and without baseline, with baseline gives a faster convergence curve because the gradient has a baseline which will help guiding the gradient to the right direction. For PPO, an approximation of TRPO, which utilize the second derivative which emperically help guiding the step direction by approximate local region with a quadratic function, and use trust region to constrain the step size, which improves the efficiency of optimization numerically without zigzagging in local valley, from the curve we can see clearly a much smoother learning curve presented in PPO than the previous two.

Part III

1.



2.

The DQN algorithm doesn't have any regularization thus will "overfit" the data distribution. However the CQL algorithm will try constraining the Q value such that it will panalize the action with the greatest value however not penalize the current choice of action. This will help the algorithm to generalize to the test set.