project 4 udacity

In your report, mention what you see in the agent's behavior. Does it eventually make it to the target location?

Yes, given enough time random movements will eventually lead the cab to its destination (or any point on our grid for that matter). Although in our scenario, even neglecting the deadline, we still hit the hard time limit of deadline + 100 iterations occasionally.

Justify why you picked these set of states, and how they model the agent and its environment.

I'm using the status of the traffic light, whether there's traffic or not, and the next_waypoint as my states. Those three items represent the most critical (in my opinion) items for determining what the cab should do next.

What changes do you notice in the agent's behavior?
It's much more intelligent now, it 'chooses' actions better and makes its way to the destination more efficiently. It also continues to improve as we iterate, thus learning along the way.

Report what changes you made to your basic implementation of Q-Learning to achieve the final version of the agent. How well does it perform?
First I implemented some reporting features to be able not identify how well the Q-Learning algorithm was doing.

I also varied Initial Q, Gamma, and Alpha values, settling on Q_init = 3, gamma = 0.45, and alpha = 0.2

Does your agent get close to finding an optimal policy, i.e. reach the destination in the minimum possible time, and not incur any penalties?
Yes- my agent consistently improves and while not perfect increases its probability of reaching the goal destination in the minimum possible time.

Links
Discussion threads used during the project
https://discussions.udacity.com/t/next-state-action-pair/44902/19
https://discussions.udacity.com/t/how-to-get-started/171416/3
https://discussions.udacity.com/t/states-is-this-on-the-right-track/44273
Interesting code example that includes reporting features
https://github.com/jaycode/smartcab/blob/master/smartcab/smartcab/agent.py

Another student example of this project I referenced during coding
https://github.com/jaycode/smartcab/blob/master/smartcab/smartcab/agent.py
Q-Learning applied to Pong
http://www.danielslater.net/2016/05/pydatalondon-2016.html