Wrapper 2: Vedant Bhat

Note: I was unable to download the original word doc, so I copy and pasted the questions into my own document.

Question 1: Watch the video and explain why the agent's policy has learned this circling behavior instead of progressing to the end of the course. Explain the behavior in terms of utility and reward.

The boat Agent found that it was able to gain high rewards if it continued collecting powerups and doing tricks in a loop. The consistent high rewards set higher utility values to this state, and in the end the Agent continued looping because the player score kept increasing, which increased the rewards and utilities at the states near the loop. Also, it seems as though the agents reward function does not place enough emphasis on getting to the finish, compared to collecting score, so it loops infinitely with no regard for finishing.

Question 2: When humans play, the rules for scoring are the same. Score is a way for games to give feedback about how well the player is doing. Why do humans play differently, always completing the course? That is, why don't humans circle in the same spot in the course if they are receiving the same score feedback as the agent?

The agent's reward function is quite different from a human's mental reward function. The agent focuses more on score, and it realizes that it can continuously increase score by driving in loops and doing tricks. On the other hand, humans place more value on finishing first and only set slight "rewards" for completing side missions/getting score. The agent is narrower minded as it focuses on state by state transitions while humans are able to see the bigger picture quickly and place more emphasis on winning the race. Also, a humans reward function can change at any time based on the circumstances (even mid race).

Question 3: The agent's original reward function is R(st,a) = game_score(st) – game_score(st-1). Describe—in terms of utility, reward, and score—two (2) ways one could modify the reward function to get the agent to behave more like a human player. That is, what do we need to change to make the agent complete the course every single time? Assume the agent has access to state information such as the position and speed of the boat and rival racers, but we cannot change how the game itself provides scores.

To improve the reward function, we would have to include an element in the method that accounts for the boat's position relative to the finish line. That is, we have to give the agent more reward for transitioning to a state that is closer to the finish line. This could be represented by some variation of:

$R(st,a) = game\_score(st) - game\_score(st\text{-}1) - (DistFromFinish(s_{t-1}) - DistFromFinish(s_t))$

Thus, if the agent is getting closer to the finish line reward increases and if it is getting further away reward decreases.

Wrapper 2: Vedant Bhat

For a second reward function, we could take into account the position of the other racers who are close to the finish line. This could be separated into 2 separate scenarios: If the agent is in first (closest to finish line compared to competitors) or if it is not in first.

If the agent is in first, the reward function should ignore the position of the other racers and attempt to continuously minimize the agent's distance from the finish. If the agent is not in first, the reward function should look at the distance between the agent and the player in first. As the agent closes on the player in first, the reward should increase. This could be mathematically represented something like:

[IF FIRST] $R(st,a) = game\_score(st) – game\_score(st-1) - (DistFromFinish(s_{t-1}) – DistFromFinish(s_t))$

[IF NOT FIRST] $R(st,a) = game\_score(st) – game\_score(st-1) - DistFromFirst(s_{t-1}) – DistFromFirst(st) + (DistFromFinish(s_{t-1}) – DistFromFinish(s_t))$

Question 4: Self-driving cars do not use reinforcement learning for a variety of reasons, including the difficulty of teaching RL agents in the real world (instead of a simulation or computer game). Suppose however, that you tried to make a reinforcement learning agent that drove a taxi. The agent is given reward based on how much fare is paid, including tips. Describe a scenario in which, after the taxi agent has learned a policy, the autonomous car might choose to do an action that could put either the rider, pedestrians, or other drivers in danger. If you think there is not such a scenario, explain how the reward function might be altered to cause the autonomous car to learn a policy that endangers the rider, pedestrians, or other drivers.

This policy is dangerous for many reasons. It will be ineffective because agents will be able to maximize rewards by driving as slow as possible (since fare is determined by length of travel in this case) and by taking as inefficient a route as possible. A majority of the time, the fastest route from point A to point B will be to drive on the paved road. But if the RL Agent wants to maximize the fare, it may go off road, take back roads, or even get into an accident, as all of these actions increase the trip length/fare.