COMP579: Reinforcement Learning - Sample midterm questions

March 10, 2026

1. Consider a sandwich shop in a small town with a fixed population of N people. Customers arrive at times governed by an unknown probability distribution. Each customer can order a sandwich with a certain type of bread (chosen from 3 types) and filling (chosen from 4 types). Customers pay a given price for each sandwich. If a customer cannot get the desired sandwich, he or she will never come back to the store again. Ingredients need to be discarded 3 days after purchase. The store owner wants to figure out a policy for buying ingredients in such a way as to maximize long-term profit, and hopes to use reinforcement learning for this task.

   - What are the state space, action space and reward function?
   - Would you use incremental reinforcement learning or dynamic programming for this problem?

2. Suppose you have an MDP and you transform the reward function by multiplying it with a constant: $r?(s, a) = wr(s, a)$.

   - Prove that for any fixed policy π, its action-value function is proportional to the old action-value function q^π.
   - Will the optimal policy stay the same? Justify your answer
   - What if instead of a constant we have a function dependent on the state, $w(s)$?

3. Sketch learning curves for a bandit algorithm for learning rate $\alpha = 0$, $\alpha = 0.01$ and $\alpha = 0.1$. Would any of these converge to the optimal policy? Justify your answer

4. Consider two function approximators, one which discretizes the state space into k bins, and one which takes each of those bins and splits it in half (hence producing a feature vector of size 2k). Suppose we want to use the approximators to estimate the value function of a fixed policy, using on-policy data. Which of these would you expect to work better with TD(λ) as opposed to TD(0)? Justify your answer

5. Why does PPO use clipping? What is the effect of the clipping parameter?

1. Solution:

   - States are represented as the quantity of each of the 3 types of bread and 4 types of filling that was purchased on each of the 3 previous days. The action space consists of the quantity of each ingredient to purchase today. The reward is the sum of: + price of every sandwich purchased, -cost of purchased ingredients.
   - Reinforcement learning would be more appropriate because the state space is very large (combinatorial) and also customers may have preferences which are not easy to model in a transition probability distribution

2. Solution:

   - Return with new reward is: $G'_t = wR_{t+1} + w\gamma R_{t+2} + \ldots = wG_t$. Therefore $q'_\pi(s,a) = E_\pi[G'_t|s,a] = E_\pi[wG_t|s,a] = wq_\pi(s,a)$
   - If $w > 0$, then action orderings for any policy will state the same, and therefore the optimal policy will be the same. If $w = 0$ all policies are optimal. If $w < 0$ action orderings flip so optimal policy will not be the same.
   - If you have $w(s)$, then the derivation in the first part does not hold and optimal policy also in general will not remain the same.

3. $\alpha = 0$ will give a flat wiggly line, $\alpha = 0.01$ will rise slowly and more stably to a higher value, $\alpha = 0.1$ will typically rise faster but more wiggly, typically flattens at a lower value

4. It would depend on the amount of data used in the learning as well as the density of rewards. If you have sparse rewards (eg only a reward at the goal) then $\lambda > 0$ is very useful for the $2k$ approximator, especially after small amounts of data. This is because the sparse reward will propagate more. If you have dense rewards, there is signal everywhere.

5. PPO uses clipping in order to reduce variance, but the clipping introduced some bias. So if you do not clip, you have more noisy learning curves, but you may end up with a better solution in the limit of a lot of data