REINFORCEMENT LEARNING AND MARKOV DECISION PROCESSES FOR DECISION-MAKING IN SELF-DRIVING CARS AT INTERSECTIONS

SANJAY R

2348055

1st October 2024

MDS472

Reinforcement Learning

Department of Statistics and Data Science

CHRIST (Deemed to be University)

Department of Statistics and Data Science

Course: MDS572C -ReinforcementLearning **Practice Problem_1**

R.SANJAY **2348055**

Reinforcement Learning and Markov Decision Processes for Decision-Making in Self-Driving Cars at Intersections

Imagine a self-driving car approaching an intersection. The car can either stop at the red light or drive through if the light is green. The objective is to maximize the car's progress while avoiding accidents and obeying traffic rules.

1. Define the States
2. Define the Actions
3. Define the Transition States
4. Formulate MDP
5. Discuss the Solutions

✧ *Introduction:*

Self-driving cars have rapidly advanced in recent years, driven by the combination of sophisticated algorithms and extensive data collection. One of the key challenges for autonomous vehicles is navigating complex real-world environments such as intersections, where decision-making must be fast, reliable, and safety-focused. Autonomous vehicles must make split-second decisions based on traffic light signals and the surrounding environment while ensuring compliance with traffic laws and preventing accidents.

In this report, we will model the decision-making process of a self-driving car approaching an intersection using a Markov Decision Process (MDP) framework. The goal is to optimize the car's behavior to maximize its progress while maintaining safety and adhering to traffic regulations. Additionally, we will discuss how Reinforcement Learning techniques can be employed to learn and improve decision-making through experience.

✧ Reinforcement Learning and Markov Decision Processes:

At the core of autonomous decision-making is the concept of Reinforcement Learning (RL), where agents, such as self-driving cars, learn to make optimal decisions based on feedback from their environment. A key framework within RL is the Markov Decision Process (MDP), which models decision-making by defining states, actions, rewards, and transition probabilities. In the case of a self-driving car, each state represents a combination of factors, such as the traffic light status and the car's distance from the intersection. By formulating the problem as an MDP, the self-driving car can decide on the best course of action to ensure safe and efficient navigation.

✧ *Key Concepts:*

The key components covered in this report include:

- ✓ **States:** Defining the different environmental and vehicle-related conditions.
- ✓ **Actions:** The available decisions the car can make.
- ✓ **Transition States:** The probabilistic nature of moving from one state to another based on actions.
- ✓ **MDP Formulation:** How to structure the Markov Decision Process.
- ✓ **Solutions and Approaches:** Exploring algorithms and methods such as Value Iteration and Policy Iteration to solve the MDP, along with an introduction to Reinforcement Learning.

✧ DEFINING THE STATES:

In the context of a self-driving car approaching an intersection,the states in an MDP represent the possible situations that the self-driving car can encounter while approaching an intersection. Each state includes information about the traffic light's current status and the car's proximity to the intersection. These states guide the car's actions, helping it decide whether to stop, proceed, or accelerate.

The states of the self-driving car at an intersection can be defined using two main variables:

i. **Traffic Light Status:** This includes the possible colors of the traffic light (red, yellow, green).
ii. **Distance to Intersection: T**his considers how far the car is from the intersection (far, medium, near).

These two variables are combined to form a complete state space, representing every possible scenario the car might face as it approaches the intersection.

❖ **Traffic Light Status:**

The traffic light status is critical in determining whether the car should proceed or stop. The self-driving car must continuously monitor the traffic light and adjust its behavior accordingly.

- **Red Light (T_R):** The red light signals the car to stop at the intersection. Failure to stop at a red light results in violations and potential accidents.
- **Yellow Light (T_Y):** The yellow light warns that the light will soon turn red. The car must decide whether to stop if it's close to the intersection or proceed if it's far enough.
- **Green Light (T_G):** The green light indicates the car can safely pass through the intersection without stopping.

❖ **Distance to the Intersection:**

The distance to the intersection affects the car's ability to make decisions. The closer the car is to the intersection, the less time it has to stop safely or accelerate.

- **Far Distance (D_F):** The car is far from the intersection, providing ample time to react to the traffic light's status.
- **Medium Distance (D_M):** The car is at a moderate distance, where quick decisions need to be made depending on the traffic light.
- **Near Distance (D_N):** The car is close to the intersection, requiring immediate actions like stopping or accelerating based on the light's status.

✧ **State Space:**

The combination of traffic light status and distance forms the state space for the self-driving car. This state space can be expressed as:

S = { (T_R, D_F), (T_R, D_M), (T_R, D_N), (T_G, D_F), (T_G, D_M), (T_G, D_N), (T_Y, D_F), (T_Y, D_M), (T_Y, D_N) }

Each state represents a specific scenario where the car must make decisions to ensure safety and compliance with traffic regulations. Below is a detailed explanation of each state in the space.

✧ **List of all Possible States:**

State 1: (Red Light, Far Distance) — (T_R, D_F) :

- In this state, the car observes a red light and is far from the intersection.
- The car has sufficient time to decelerate and stop before reaching the intersection.
- The primary action would be to slow down and prepare for a complete stop.

State 2: (Red Light, Medium Distance) — (T_R, D_M):

- The car faces a red light and is at a moderate distance from the intersection.
- Here, the car needs to decelerate more aggressively to stop before reaching the intersection.
- It is important for the car to start slowing down to comply with traffic rules.

State 3: (Red Light, Near Distance) — (T_R, D_N):

- In this scenario, the car is near the intersection with a red light.
- The car must stop immediately to avoid running the light, which could cause a collision or violation.
- No other actions are acceptable in this state except for stopping.

State 4: (Green Light, Far Distance) — (T_G, D_F):

- The traffic light is green, and the car is far from the intersection.
- The car can maintain or increase its speed to proceed through the intersection safely.
- There is no need to decelerate unless other factors, such as obstacles, are present.

State 5: (Green Light, Medium Distance) — (T_G, D_M)

- The car encounters a green light at a medium distance.
- It is safe for the car to continue moving forward and pass through the intersection.
- Acceleration might be a suitable action to ensure the car crosses before the light changes.

State 6: (Green Light, Near Distance) — (T_G, D_N)

- In this state, the car is near the intersection with a green light.
- The car should proceed without delay as the light is still green.
- Minimal decision-making is required here, and the primary action is to continue moving.

State 7: (Yellow Light, Far Distance) — (T_Y, D_F)

- The yellow light is on, and the car is far from the intersection.
- The car should slow down, anticipating the light turning red by the time it reaches the intersection.
- This state demands caution, as speeding through may not be safe.

State 8: (Yellow Light, Medium Distance) — (T_Y, D_M):

- The car observes a yellow light and is at a medium distance.
- A quick decision needs to be made—either to slow down and stop or speed up to pass through the intersection before the light turns red.
- The car's ability to decide here is crucial to avoid violations or accidents.

State 9: (Yellow Light, Near Distance) — (T_Y, D_N);

- In this scenario, the car is near the intersection with a yellow light.
- The car must decide between speeding up to cross the intersection before the light turns red or stopping abruptly.
- Given the closeness to the intersection, actions must be quick and accurate.

✧ DEFINING THE ACTIONS:

The self-driving car's actions are the decisions it must take based on the current state it is in, including the traffic light status and distance to the intersection. These actions guide the car's movement to ensure safety and compliance with traffic rules. The car must choose an appropriate action for every state, affecting the next state it will transition into.
The key actions available for the car include:

i. **Stop (A_Stop):** The car comes to a halt.
ii. **Proceed (A_Proceed):** The car moves forward, either maintaining or slightly increasing speed.
iii. **Accelerate (A_Accelerate):** The car speeds up to pass through the intersection quickly.

These actions will vary depending on the traffic light's status and how close the car is to the intersection.

✧ List of Actions for Each State

State 1: (Red Light, Far Distance)
Action: Stop (A_Stop)
The car is far from the intersection, and the light is red. The safest and most appropriate action is to begin slowing down to come to a stop before the intersection.

State 2: (Red Light, Medium Distance)
Action: Stop (A_Stop)
The car is moderately close to the intersection with a red light. The car needs to decelerate faster and prepare for a complete stop.

State 3: (Red Light, Near Distance)
Action: Stop (A_Stop)
The car is very close to the intersection, and the red light demands that it immediately stop. Failure to stop could result in violations or accidents.

State 4: (Green Light, Far Distance)
Action: Proceed (A_Proceed)
Since the light is green and the car is far from the intersection, the car can continue moving forward safely at the current speed or with a slight acceleration.
State 5: (Green Light, Medium Distance)
Action: Proceed (A_Proceed)

The car can continue moving at a moderate pace, but accelerating may help ensure that the car passes the intersection before the light changes.

State 6: (Green Light, Near Distance)
Action: Proceed (A_Proceed)
The car is close to the intersection, and the green light is still on. The car should keep moving through the intersection without any hesitation.

State 7: (Yellow Light, Far Distance)
Action: Stop (A_Stop)
The car is far from the intersection, and the yellow light indicates that it should prepare to stop. It's safer to slow down and come to a stop since the light will likely turn red before reaching the intersection.

State 8: (Yellow Light, Medium Distance)
Action: Stop or Accelerate (A_Stop / A_Accelerate)
This is a decision-making state. The car needs to evaluate whether it has enough time to speed up and pass through the intersection before the light turns red. If not, the car should stop. The choice depends on the car's speed and distance.

State 9: (Yellow Light, Near Distance)
Action: Accelerate (A_Accelerate)
The car is very close to the intersection with a yellow light. The safest action would be to accelerate quickly and pass through the intersection before the light turns red.

✧ DEFINING THE TRANSITION STATES:

Transition states describe how the self-driving car moves between different conditions after taking an action. These transitions are probabilistic, meaning the outcome of taking a specific action from a given state is not always certain and depends on factors such as speed, distance, and traffic light status. The car transitions from one state to another based on the probability
$P(s' \mid s,a)$, which is the likelihood of moving to a new state s', given that the car is in state s and takes action a.

By considering these transitions, the self-driving car can model uncertainty and make informed decisions that account for both immediate and future outcomes. Below are detailed explanations of the transition states for each of the 9 key states:

1. State: (Red Light, Far Distance)
Action: Stop
Transition:The car chooses to decelerate and stop because the light is red, and it is far from the intersection. The probability of successfully transitioning to (Red Light, Near Distance) and eventually coming to a stop is very high, close to 1.0, as there is ample time to decelerate.
There is a negligible chance the car will remain in the (Red Light, Far Distance) state, as it is already reducing speed.

2. State: (Red Light, Medium Distance)
Action: Stop
Transition:When the car is moderately close to the intersection, stopping is still the best course of action. The car will likely transition to (Red Light, Near Distance) with a high probability, preparing for a complete stop.There is a small chance the car could remain at (Red Light, Medium Distance), but this is unlikely, as the action to stop would mean continued deceleration.

3. State: (Red Light, Near Distance)
Action: Stop
Transition:At this point, the car is very close to the intersection, and the red light requires the car to stop immediately. The transition to the final state of (Red Light, Stopped) is almost certain (probability near 1.0), as failure to stop would result in a violation or accident.

4. State: (Green Light, Far Distance)
Action: Proceed
Transition:Since the light is green, the car can safely proceed. If the car continues at its current speed or slightly accelerates, it will most likely transition to (Green Light, Medium Distance) with a high probability.
There is a small chance the car may remain at (Green Light, Far Distance) if external factors slow it down unexpectedly.

5. State: (Green Light, Medium Distance)
Action: Proceed
Transition:With the green light still on and the car at a medium distance, it can proceed through the intersection. The car is likely to transition to (Green Light, Near Distance) with a probability close to 0.8, as long as it maintains speed or accelerates.
There is a smaller probability (around 0.2) that the car could remain in (Green Light, Medium Distance) due to slower speeds or external factors.

6. State: (Green Light, Near Distance)
Action: Proceed
Transition:The car is very close to the intersection with a green light, so the most likely outcome is that it successfully moves through the intersection. The probability of transitioning to a new state beyond the intersection (e.g., Passed Intersection) is very high (close to 1.0), as there is little to impede its progress.
The chance of staying in (Green Light, Near Distance) is minimal unless the car unexpectedly slows down.

7. State: (Yellow Light, Far Distance)
Action: Stop
Transition:When the car is far from the intersection and the light turns yellow, it is safer to stop. The car will likely transition to (Yellow Light, Medium Distance) or (Yellow Light, Near Distance) as it decelerates, preparing to stop.
There is a small chance that if the car is moving too fast, it could remain in (Yellow Light, Far Distance) for a brief moment longer, but stopping is the recommended action.

8. State: (Yellow Light, Medium Distance)
Action: Stop or Accelerate
Transition:In this situation, the car needs to decide between stopping and accelerating. If it chooses to stop, there is a high probability (around 0.7) that it will transition to (Red Light, Near Distance) or a complete stop.
If the car accelerates, there is a chance (0.3) that it will move through the intersection before the light turns red, transitioning to (Passed Intersection).
This decision is highly dependent on factors like speed and distance.

9. State: (Yellow Light, Near Distance)
Action: Accelerate
Transition:The car is very close to the intersection, and accelerating through the yellow light is likely the safest action to avoid stopping abruptly. The probability of transitioning to (Passed Intersection) is high (around 0.9), as the car can quickly pass through before the light turns red.There is a small probability that the car might fail to pass the intersection before the light changes, but this is less likely at such a close distance.

✧ FORMULATING THE MDP:

A Markov Decision Process (MDP) provides a mathematical framework for modeling decision-making in environments where outcomes are partly random and partly under the control of a decision-maker. In the context of a self-driving car at an intersection, an MDP consists of the following elements:

i. States (S):
The states represent the various conditions in which the car can find itself. In this scenario, we have 9 states defined based on traffic light status and distance to the intersection:

- ✓ State 1: (Red Light, Far Distance)
- ✓ State 2: (Red Light, Medium Distance)
- ✓ State 3: (Red Light, Near Distance)
- ✓ State 4: (Green Light, Far Distance)
- ✓ State 5: (Green Light, Medium Distance)
- ✓ State 6: (Green Light, Near Distance)
- ✓ State 7: (Yellow Light, Far Distance)
- ✓ State 8: (Yellow Light, Medium Distance)
- ✓ State 9: (Yellow Light, Near Distance)

ii. Actions (A):
The actions available to the self-driving car are:

- ✓ A_Stop: The car comes to a halt.
- ✓ A_Proceed: The car moves forward through the intersection.
- ✓ A_SlowDown: The car reduces its speed.
- ✓ A_Accelerate: The car increases its speed.

iii. Transition Function (P):
The transition function defines the probabilities of moving from one state to another after taking a specific action. For example, the transition from state (Green Light, Medium Distance) to state (Green Light, Near Distance) occurs with a probability of 0.8 when the car takes the action A_Proceed.

iv. Reward Function (R):
The reward function assigns a value to each state-action pair, indicating the immediate reward received after taking an action in a given state. For instance:
Successfully passing through the intersection while the light is green may yield a high reward.
Stopping at a red light might yield a neutral reward (0).

Running a red light could result in a negative reward due to safety concerns and traffic violations.

iv. **Discount Factor γ:** The discount factor ($0 < \gamma \leq 1$), which determines the present value of future rewards. A higher γ places more emphasis on long-term rewards.

The MDP framework enables the self-driving car to systematically evaluate its actions based on the current state, potential transitions, and the expected rewards associated with each action, allowing it to make informed decisions to navigate intersections safely.

✧ **SOLUTIONS:**

Value Iteration, Policy Iteration, and Reinforcement Learning

The goal of the MDP is to find an optimal policy that tells the car which action to take in each state in order to maximize its expected cumulative reward. This can be achieved using the following techniques:

a. Value Iteration

Value iteration is a dynamic programming technique used to compute the optimal policy in a Markov Decision Process (MDP). The main objective of value iteration is to iteratively update the value of each state based on the expected future rewards of possible actions. The core principle involves evaluating each state by calculating the maximum expected future reward achievable from that state, given a specific action. Mathematically, this is represented as:

$$V(s)=\max_a \sum_{s'} P(s' \mid s,a)[R(s,a,s')+\gamma V(s')]$$

In this equation, V(s) denotes the value of state s, while a represents the action taken. The term P(s'|s,a) signifies the probability of transitioning to a new state s′ given the current state s and action a. The reward received when transitioning from state s to state s′ after taking action a is denoted by R(s,a,s'). Lastly, γ is the discount factor, which prioritizes immediate rewards over distant future rewards. This iterative process continues until the value function converges, at which point the optimal policy can be derived by selecting the action that yields the highest expected reward at each state.

b. Policy Iteration:

Policy iteration is another effective method for solving MDPs, consisting of two main steps: policy evaluation and policy improvement. Initially, a policy—a mapping from states to actions—is defined. The evaluation step calculates the expected value of each state under this current policy, providing a baseline for improvements. In mathematical terms, the value function for the current policy π can be expressed as:

$$V\pi(s)=s'\sum \quad P(s' \mid s,\pi(s))[R(s,\pi(s),s')+\gamma V\pi(s')]$$

Here, $V\pi(s)$ represents the value of state s under the policy π. During the improvement step, the policy is updated by selecting the action that maximizes the expected value, represented by:

$$\pi(s)=\mathrm{argamax} \quad s'\sum \quad P(s' \mid s,a)[R(s,a,s')+\gamma V\pi(s')]$$

This process continues, alternating between evaluating and improving the policy until the policy stabilizes, meaning further iterations yield no changes. The outcome is an optimal policy that prescribes the best action for each state based on expected cumulative rewards.

c. Reinforcement Learning

Reinforcement Learning (RL) is particularly advantageous in environments where the transition probabilities and reward structures are unknown. In this framework, the self-driving car learns by interacting with its environment and adjusting its actions based on received rewards. One of the key algorithms in RL is Q-Learning, which aims to estimate the optimal action-value function, denoted as $Q(s,a)$. This function quantifies the expected utility of taking action a in state s and following the optimal policy thereafter.

The update rule for Q-Learning is represented as:

In this equation, α represents the learning rate, which determines the degree to which new information overrides old information. The term R is the immediate reward received after taking action a in state s, leading to the new state s'. The discount factor γ adjusts the importance of future rewards. Over time, through repeated interactions with the environment, the car refines its Q-values, eventually developing an optimal policy that maximizes cumulative rewards.

✧ *Conclusion:*

This report has outlined how a self-driving car's decision-making at an intersection can be modeled as an MDP. By defining states, actions, and transitions, the car can be guided to make optimal decisions based on probabilistic outcomes. Solving the MDP with methods like Value Iteration, Policy Iteration, or Reinforcement Learning allows the car to safely and efficiently navigate through intersections, while adhering to traffic rules and maximizing its progress.

In real-world implementations, reinforcement learning holds the potential for self-driving cars to autonomously learn and adapt to various driving environments, continually improving their decision-making capabilities.