Actividad 2

Proceso de Decisión de Markov (MDP)

TIC

TALENTO TECH

Universidad Tecnológica de Pereira

faceIT

Proceso de Decisión de Markov (MDP)

El Proceso de Decisión de Markov (MDP) es un modelo matemático fundamental en el aprendizaje por refuerzo (RL). Esta unidad proporciona una descripción detallada del MDP y sus componentes clave, que incluyen estados, acciones, la función de transición y las recompensas. Se exploran los conceptos de Markov y la propiedad de recompensa, que son fundamentales para comprender cómo los agentes toman decisiones en entornos dinámicos y estocásticos.

El Proceso de Decisión de Markov (MDP) es un modelo matemático ampliamente utilizado en el aprendizaje por refuerzo (RL). En esencia, un MDP describe un entorno en el que un agente toma decisiones secuenciales para maximizar una recompensa acumulada a lo largo del tiempo.

Propiedad de Recompensa:

La propiedad de recompensa se refiere a cómo se asignan las recompensas a las acciones tomadas por el agente. En un MDP, la recompensa puede ser determinista o estocástica, dependiendo del entorno y la tarea específica.

Propiedades de Markov:

Un MDP tiene la propiedad de Markov si la probabilidad de transición entre estados depende únicamente del estado actual y la acción tomada, no de la historia completa de estados y acciones anteriores. Esta propiedad es esencial para simplificar el modelado y el cálculo en RL.

Conceptos clave:

Los estados representan situaciones o configuraciones en el entorno. Las acciones son las opciones disponibles para el agente en cada estado. La función de transición define las probabilidades de transición entre estados cuando se toma una acción. La función de recompensa asigna una recompensa numérica a cada estado-acción o estado transición.

Descripción del MDP:

En un MDP, tenemos un conjunto de estados, un conjunto de acciones disponibles para el agente en cada estado, una función de transición que describe la probabilidad de moverse de un estado a otro dado una acción, y una función de recompensa que proporciona la recompensa inmediata recibida por el agente por realizar una acción en un estado dado.

Descripción del MDP

Ejercicio: De ne un MDP básico con tres estados (A, B, C), dos acciones posibles en cada estado (Arriba, Abajo), una función de transición aleatoria y recompensas aleatorias asociadas a cada acción en cada estado.

```python
import numpy as np

# Definición de estados, acciones y recompensas

estados = ['A', 'B', 'C']

acciones = ['Arriba', 'Abajo']

recompensas = np.random.randint(0, 10, size=(len(estados),

len(acciones)))


# Función de transición aleatoria

def transicion_aleatoria():

      return np.random.choice(estados)
```

```python
# Generación de datos

estado_actual = np.random.choice(estados)

accion = np.random.choice(acciones)

nuevo_estado = transicion_aleatoria()

recompensa = recompensas[estados.index(estado_actual),

acciones.index(accion)]


print("Estado actual:", estado_actual)

print("Acción tomada:", accion)

print("Nuevo estado:", nuevo_estado)

print("Recompensa:", recompensa)
```

Conceptos de MDP

Ejercicio: Escribe una función para calcular la función de valor de un estado dado un MDP, utilizando el algoritmo de iteración de valor.

```
def calcular_valor_estado(mdp, gamma=0.9, theta=0.01):
      valores = {estado: 0 for estado in mdp.estados}
      while True:
      delta = 0
      for estado in mdp.estados:
            valor_previo = valores[estado]
            valores[estado] =
sum(mdp.transiciones[estado][accion][nuevo_estado] *

(mdp.recompensas[estado][accion][nuevo_estado] + gamma *
valores[nuevo_estado])
                                    for accion in mdp.acciones for
nuevo_estado in mdp.estados)
            delta = max(delta, abs(valor_previo - valores[estado]))
      if delta < theta:
            break
      return valores


# Ejemplo de uso
valores_estados = calcular_valor_estado(mdp)
print("Valores de los estados:", valores_estados)
```

Propiedades de Markov

```
def verificar_propiedad_markov(mdp):
      for estado in mdp.estados:
      for accion in mdp.acciones:
            suma_probabilidades =
sum(mdp.transiciones[estado][accion].values())
            if not np.isclose(suma_probabilidades, 1):
                  return False
      return True


# Ejemplo de uso
print("Cumple con la propiedad de Markov:",
verificar_propiedad_markov(mdp))
```

Ejercicio: Escribe una función para veri car si un MDP dado cumple con la propiedad de Markov.

Propiedad de Recompensa

```
def calcular_recompensa_promedio(mdp):
      recompensa_total = 0
      total_acciones = 0
      for estado in mdp.estados:
      for accion in mdp.acciones:
            for nuevo_estado in mdp.estados:
                  recompensa_total +=
mdp.recompensas[estado][accion][nuevo_estado]
                  total_acciones += 1
      return recompensa_total / total_acciones


# Ejemplo de uso
print("Recompensa promedio por acción:",
calcular_recompensa_promedio(mdp))
```

Ejercicio: Escribe una función para calcular la recompensa promedio por acción en un MDP.