Problema del 8-Puzzle

Objetivos de aprendizaje:

- Comprender y aplicar los conceptos de aprendizaje por refuerzo y el algoritmo Q-learning.
- Implementar un agente de aprendizaje por refuerzo que resuelva el problema del 8-puzzle.
- Evaluar el rendimiento del agente y ajustar los hiperparámetros para mejorar la eficacia de aprendizaje.

Descripción de la tarea:

1. **Introducción al problema del 8-puzzle**:

   El problema del 8-puzzle consiste en una cuadrícula de 3x3 con 8 piezas numeradas del 1 al 5 y un espacio vacío. El objetivo es reordenar las piezas de modo que coincidan con un estado objetivo predefinido, por ejemplo:

   1 2 3
   4 5 6
   7 8

- **Configuración del entorno de aprendizaje**:

Implementa un entorno para el 8-puzzle donde el agente pueda mover las piezas en las cuatro direcciones posibles: arriba, abajo, izquierda y derecha. Asegúrate de definir las reglas de movimiento (es decir, que no puede moverse fuera de los límites de la cuadrícula) y de penalizar movimientos inválidos.

- **Definición de recompensas y estados**:

  - Cada estado es una configuración única del tablero.
  - El objetivo del agente es llegar al estado objetivo.
  - Asigna una recompensa negativa para cada paso para incentivar al agente a resolver el puzzle en el menor número de movimientos posible. Otorga una recompensa positiva significativa al alcanzar el estado objetivo.

- **Implementación del algoritmo Q-learning**:

  - Implementa la tabla Q para almacenar las recompensas para cada acción en cada estado.
  - Usa una política ε-greedy para equilibrar la exploración y explotación.
  - Aplica la actualización de la función Q con la ecuación:

$$Q(s,a) = Q(s,a) + \alpha \left[r + \gamma \max_{a'} Q(s',a') - Q(s,a) \right]$$

1.

donde:

- s y s′ son el estado actual y el siguiente.
- a y a′ son las acciones actuales y las posibles en s′.
- α es la tasa de aprendizaje.

- γ es el factor de descuento.

2. **Entrenamiento del agente**:
   - Entrena al agente con diferentes configuraciones iniciales del 8-puzzle.
   - Ajusta los hiperparámetros (ε, α, γ) para optimizar el rendimiento.
   - Registra el número de pasos que toma el agente para resolver el puzzle desde diferentes estados.
3. **Evaluación y análisis de resultados**:
   - Evalúa el rendimiento del agente en el 8-puzzle.
   - Presenta gráficas que muestren cómo cambia la tasa de éxito y el número de pasos promedio para resolver el puzzle a lo largo del tiempo.
   - Reflexiona sobre el impacto de cada hiperparámetro en el rendimiento del agente y su capacidad de generalización desde estados iniciales aleatorios.
4. **Entrega**:
   - Código implementado con explicaciones de cada parte.
   - Un informe que incluya:
     - Gráficas de rendimiento.
     - Análisis de los resultados.
     - Reflexiones sobre los desafíos encontrados y las decisiones de diseño tomadas.

Evaluación:

- Estructura, funcionamiento correcto y claridad del código. (3 Puntos)
- Documentación y explicaciones. (3 Puntos)
- Calidad del análisis de resultados. (2 Puntos)
- Ajuste y justificación de los hiperparámetros. (2 Puntos)

AYUDA:

Definimos la tabla de transiciones generando todos los estados posibles como permutaciones del número 123456789 siendo el hueco el número 9.

Irían desde el estado solución 123456789 hasta el 987654321.

Para generar la tabla sabemos que hay cuatro posibles acciones que puede realizar el hueco a0 (subir), a1 (derecha), a2 (abajo) y a3 (izquierda). Partiendo del estado solución no todos los estados son alcanzables.

Realizamos un proceso sistemático generando un array bidimensional T de (987654321-123456789) filas y 4 columnas rellenadas a -1. Una vez creada recorremos cada fila y columna y vamos rellenando la tabla con los estados alcanzables. (Se generan dos grafos no conexos de la mitad de nodos cada uno) Con un recorrido en anchura desde el estado solución podría marcar los alcanzables.