Tarea 4

Inteligencia Artificial

Facultad de Ingeniería y Ciencias
Universidad Adolfo Ibáñez
Profesores: Mauricio Valle, Camilo Ramirez, Mauricio Figueroa

Fecha Distribución: 03 de noviembre de 2025
Fecha Entrega: 28 de noviembre de 2025

Objetivo

El objetivo de la tarea es implementar y analizar un agente de aprendizaje por refuerzo con Q-Learning en un MDP discreto, formalizando estados, acciones y recompensas, diseñando una política de exploración-explotación (ε-greedy), entrenando por episodios con actualización de Q(s,a) y evaluando el desempeño por retorno acumulado, eficiencia y estabilidad de la política; además, se espera discutir la sensibilidad a α, γ, ε y asegurar la reproducibilidad del experimento.

1. Contexto

El laboratorio de IA del curso dispone de un entorno experimental discreto tipo grid, compuesto por celdas con coordenadas (i,j) y elementos identificados (recursos y riesgos). Algunos espacios contienen trofeos y recompensas, otros presentan restricciones como bloques, puertas que requieren llave, y zonas peligrosas con agentes hostiles. En este contexto, la tarea se centra en que un agente autónomo aprenda, mediante Q-Learning, a planificar un recorrido eficiente que maximice el retorno: recolectar objetivos y alcanzar la meta minimizando pasos y evitando riesgos, bajo una función de recompensas que incorpora penalización por movimiento. El trabajo enfatiza la formalización del MDP (estados, acciones, recompensas), el entrenamiento por episodios y la evaluación de la política en mapas de distinta complejidad.

2. Problema

Se requiere desarrollar una solución con Q-Learning para que un agente autónomo navegue de forma segura y eficiente en un entorno discreto tipo grid, tomando como referencia los mapas y reglas definidos en el notebook. El agente debe recolectar todos los trofeos y alcanzar la meta, gestionando restricciones (p. ej., puertas que requieren llave y bloques) y evitando zonas peligrosas (enemigos), mientras optimiza su política para minimizar el número de pasos y la penalización acumulada (tiempo/energía). La solución debe aprender por episodios sin conocimiento previo de la dinámica y evidenciar mejoras en el retorno al comparar la política aprendida con líneas base simples.

3. Instrucciones de Implementación

A continuación, se presentan instrucciones resumidas y ordenadas para implementar la solución con Q-Learning en el mismo notebook; se recomienda resolver P1→P7 en secuencia y dejar evidencia mínima (gráficos/tablas/comentarios) por cada paso.

P1 — Entorno (MDP)

Objetivo: Definir el grid, estados, acciones, recompensas y condiciones terminales; implementar reset/step con semilla para garantizar reproducibilidad.

P2 — Inicialización

Objetivo: Crear la tabla Q y fijar parámetros operativos (tasa de aprendizaje, descuento, exploración, episodios y pasos máximos) dejando todo listo para entrenar.

P3 — Política de exploración-explotación

Objetivo: Implementar la política epsilon-greedy y su decaimiento para equilibrar búsqueda de nuevas acciones con el aprovechamiento del conocimiento aprendido.

P4 — Entrenamiento Q-Learning

Objetivo: Ejecutar el bucle de episodios/pasos, actualizar la tabla Q en función de la recompensa observada y del valor estimado futuro, registrando retorno y pasos por episodio.

P5 — Curvas de aprendizaje

Objetivo: Visualizar retorno por episodio, tasa de éxito y pasos, describiendo brevemente las tendencias y el punto en que el aprendizaje se estabiliza.

P6 — Sensibilidad de hiperparámetros

Objetivo: Comparar configuraciones (tasa de aprendizaje, descuento, exploración/decay) y, si aplica, variantes de mapa, para analizar impacto en desempeño y estabilidad.

P7 — Evaluación y política final

Objetivo: Extraer la política final (greedy respecto de Q), evaluarla en varios episodios de prueba y compararla con una línea base simple, reportando métricas clave.

4. Entregable

Un único archivo Jupyter Notebook (.ipynb) que contenga todo el código que cumpla con las instrucciones, visualizaciones y documentación del proyecto, con el nombre grupoX_tarea4_IA.ipynb (donde X corresponde al identificador del grupo).

5. Consideraciones

- Los equipos de trabajo deben estar conformados por un máximo de 4 integrantes, y la corrección de la tarea se verificará en la plataforma Colab o Jupyter.

- Se debe subir a webcursos un notebook en formato **grupoX_tarea4_IA.iypnb**, donde **X** es el identificador del grupo (ej. Apellidos).
- Por cada día de atraso, se descontarán 0.5 puntos de la nota final de la tarea.