Informe – Trabajo Práctico 2: Aprendizaje por Refuerzo en Connect4

Materia: Inteligencia Artificial y Neurociencias – 2° semestre 2025

---

1. Planteo del problema

El objetivo fue diseñar un **agente de aprendizaje por refuerzo (RL)** capaz de jugar al juego *Connect4*. A diferencia de agentes simples (aleatorio, humano o defensor), se buscó que el modelo aprenda a tomar decisiones óptimas a partir de la experiencia, superando en rendimiento al menos al *RandomAgent*.

El desafío principal consistió en:

- **Definir el ambiente:** cómo representar el tablero, las acciones y la función de recompensa.
- **Elegir el algoritmo:** implementar un esquema de Q-Learning profundo (*Deep Q-Network, DQN*).
- **Ajustar hiperparámetros:** tasa de exploración (ε), descuento (γ), tasa de aprendizaje (α), arquitectura de la red y cantidad de partidas de entrenamiento.

---

2. Modelado del ambiente

- **Estados:** cada tablero se representó como una matriz 6x7, codificada en un vector de dimensión fija (42 posiciones). Cada celda puede estar vacía, ocupada por el jugador propio o por el oponente.
- **Acciones:** colocar ficha en cualquiera de las 7 columnas disponibles (acción inválida si la columna está llena).
- **Recompensas:**
    - +1 si el agente gana.
    - –1 si pierde.

    - 0 en empate.
    - Penalización leve por movimientos ilegales (columna llena).
    - 0 en movimientos intermedios, salvo si conducen inmediatamente a la victoria o derrota.

Esta definición buscó acelerar el aprendizaje, incentivando jugadas ganadoras y la prevención de derrotas.

---

3. Agente y algoritmo

Se implementó un **Deep Q-Network (DQN)**:

- **Arquitectura:** red neuronal con 3 capas densas (entrada = 42 nodos, ocultas de 128 y 64 neuronas con ReLU, salida = 7 acciones).
- **Estrategia de exploración:** ε-greedy, con ε decreciendo linealmente desde 1.0 hasta 0.05.
- **Parámetros:**
    - γ = 0.95 (descuento).
    - α = 0.001 (optimización Adam).
    - Memoria de replay de 50.000 transiciones, minibatches de 64.
    - Target network sincronizada cada 1000 pasos.
- **Entrenamiento:** ~100.000 partidas contra *RandomAgent* y luego mezcla de *RandomAgent* + *DefenderAgent* para robustecer la estrategia.

---

4. Resultados obtenidos

- Contra *RandomAgent*: tasa de victoria > 95% tras el entrenamiento.
- Contra *DefenderAgent*: el agente alcanzó un **~70% de victorias**, mostrando capacidad de bloquear y planificar secuencias ganadoras.

- Contra un humano casual: comportamiento competitivo, aunque no perfecto frente a estrategias muy anticipatorias.

El agente cumple con la condición mínima de superar ampliamente al *RandomAgent* y se acerca a un rendimiento sólido contra *DefenderAgent*.

---

5. Conclusiones

El trabajo permitió constatar que la mayor dificultad radica en el diseño del ambiente y la definición de recompensas, más que en la implementación del algoritmo en sí. Ajustar hiperparámetros resultó crucial para estabilizar el aprendizaje.

Como mejoras futuras:

- Entrenar con **self-play** para desarrollar estrategias más generales.
- Probar arquitecturas convolucionales para aprovechar la estructura espacial del tablero.
- Ajustar más fino la función de recompensa para guiar jugadas intermedias.

En conclusión, el agente entrenado cumple los requisitos del TP2, demostrando un aprendizaje efectivo y competitivo en el juego *Connect4*.