IES Pere Maria Orts

Sistemas de Aprendizaje Automático

Práctica 1_2: OXO - Inteligencia débil y fuerte

Autor:
Kenny Berrones

Profesor:
David Campoy Miñarro

iesperemariaorts

GENERALITAT VALENCIANA

Índice

1. Introducción **2**

2. OXO con Inteligencia Débil **2**

3. OXO con Inteligencia Fuerte **3**

4. OXO con Red Neuronal **6**

5. Conclusiones **9**

1. Introducción

En esta práctica se nos pide que probemos distintos algoritmos para jugar al videojuego OXO. Este juego es más conocido como el Tic-Tac-Toe, se juega en un tablero 3x3 y el objetivo es completar una fila, columna o diagonal con la misma ficha del jugador.

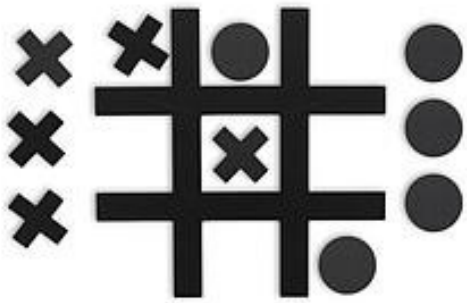

Figura 1: Ejemplo partida Tic-Tac-Toe

2. OXO con Inteligencia Débil

La versión con "inteligencia" débil se trata de una versión en la que devuelve un valor aleatorio de las celdas libres, por lo tanto vemos que no tenemos una inteligencia como tal, en la siguiente imagen vemos que ganamos prácticamente a la primera:

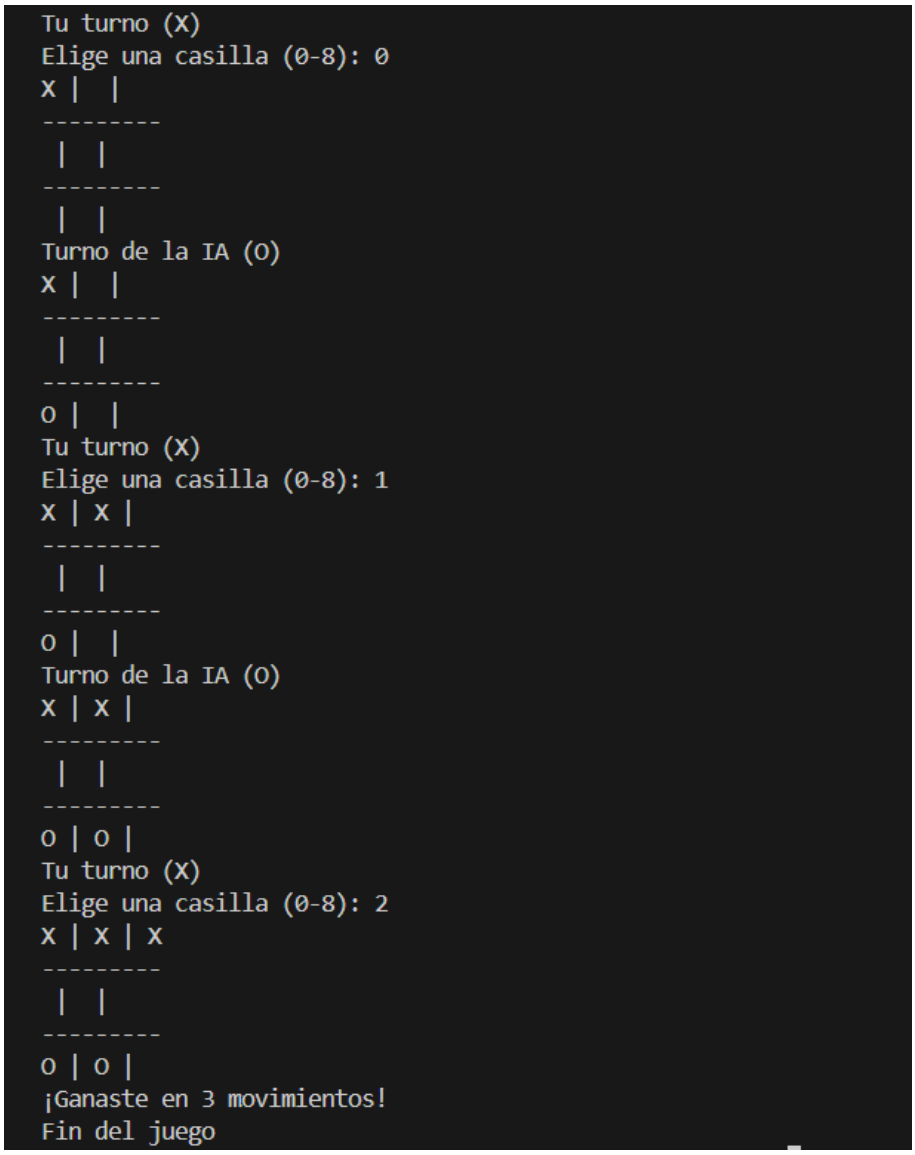

Figura 2: Resultados OXO con Inteligencia Débil

3. OXO con Inteligencia Fuerte

En esta versión de OXO, vamos a utilizar el algoritmo Q-Learning. Este algoritmo se trata de una técnica de Aprendizaje por Refuerzo (RL) que enseñará a un agente a actuar en un escenario. El agente aprende a decidir maximizando una función de recompensa a largo plazo. En una tabla Q se almacenan los valores de recompensa esperados para cada par estado-acción. La idea es que el agente aprende que acción tomar para cada estado, con el fin de maximizar sus recompensas.

Hay que destacar que es importante explicar dos hiper parámetros de este algoritmo: el learning rate y el discount factor:

- **Learning Rate:** Este parámetro indica cuánto influye la nueva información en la actualización del valor de Q. Un valor alto significa que al agente se adaptará rápidamente.
- **Discount Factor:** Este parámetro mide la importancia de las recompensas futuras.

Cuanto más cerca de 1 más se enfocará en las recompensas a largo de plazo.

En la siguiente tabla podemos apreciar que escenarios se plantean con distintos valores para cada parámetro:

Tasa de Aprendizaje	**Factor de Descuento**	**Efecto en el Q-Learning**
0.01	0.9	El aprendizaje es muy lento y prioriza las recompensas futuras.
0.9	0.1	El aprendizaje es rápido pero se enfoca en las recompensas inmediatas.
0.01	0.1	El aprendizaje es muy lento y se enfoca en las recompensas inmediatas.
0.5	0.5	El aprendizaje es equilibrado entre recompensas inmediatas y futuras.
0.9	0.9	El aprendizaje es rápido y prioriza las recompensas futuras.

Cuadro 1: Efectos de diferentes combinaciones de Tasa de Aprendizaje y Factor de Descuento.

En la siguiente imagen se aprecia una partida que realicé:

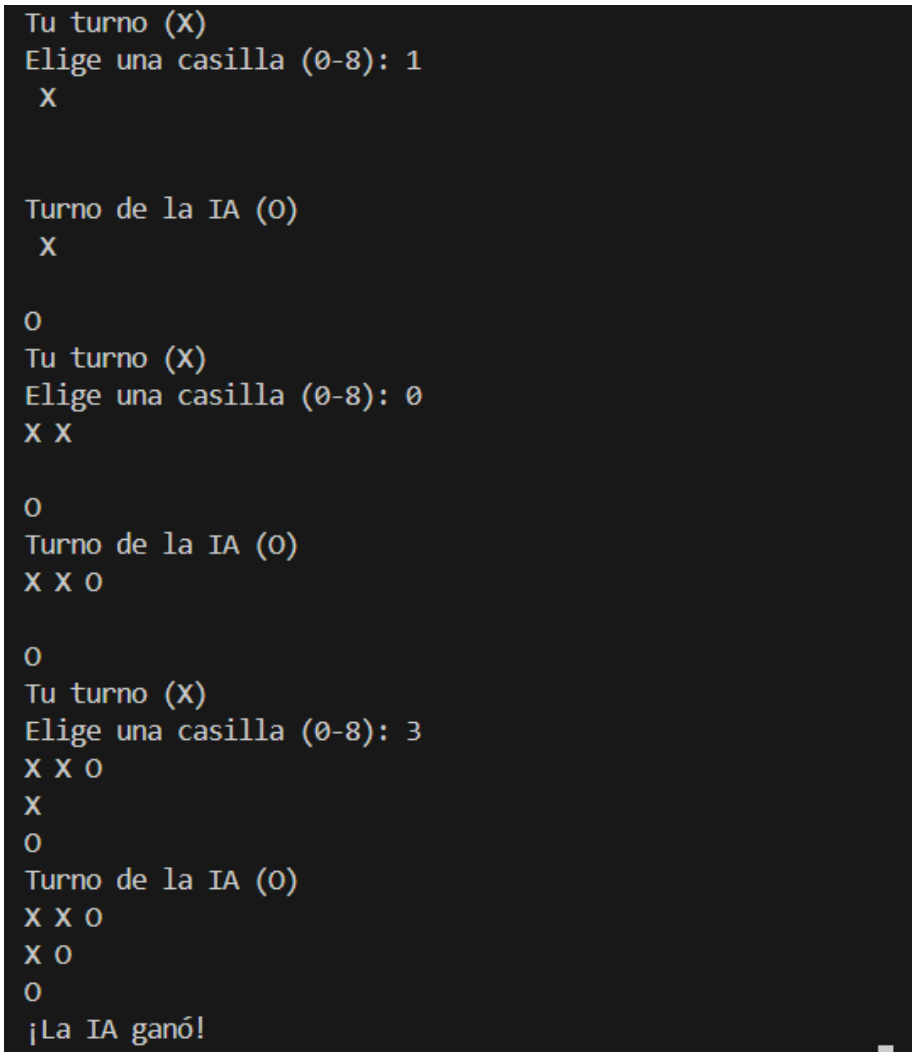

Figura 3: Resultado de una partida con inteligencia fuerte

En esta partida la IA me gana, pero igualmente le podría haber ganado fácilmente, por lo que vemos que no "existe" esa inteligencia que este algoritmo promete.

Si modificamos el hiper parámetro de los episodios observamos la siguiente gráfica:

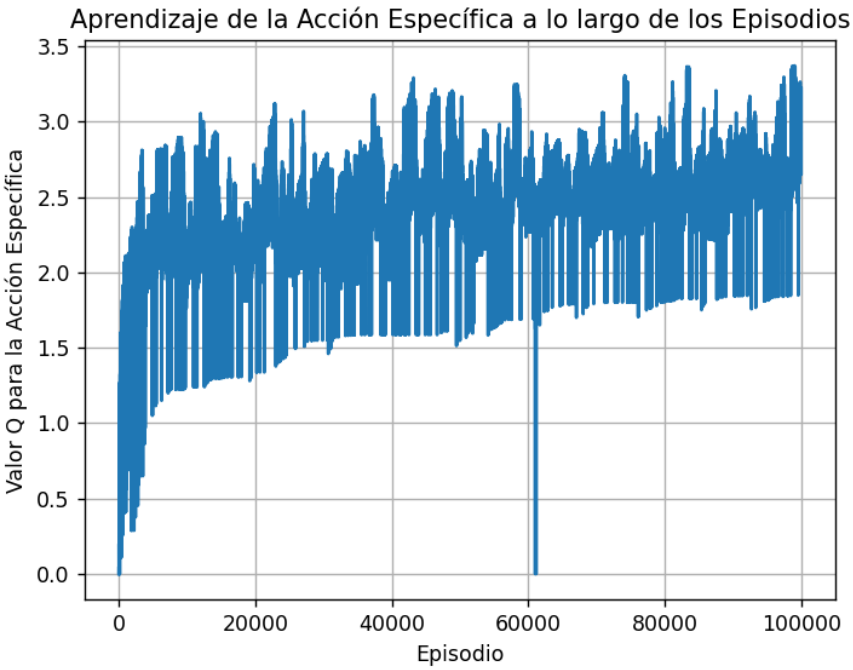

Figura 4: Evolución del aprendizaje aumentando el valor de los episodios

Si vemos la gráfica anterior, vemos que a medida que aumentamos el número de episodios vemos que también el valor de Q aumenta para la acción específica, por lo que fijándonos en la gráfica creo que un valor para el número de episodios sería de alrededor de 20000.

4. OXO con Red Neuronal

Finalmente, tenemos la versión de OXO que emplea una red neuronal para entrenar el algoritmo que luego jugará contra nosotros, este algoritmo aprenderá en su entrenamiento a jugar. Este aprenderá de 1000 posibles juegos, además, iremos guardando el error del modelo a la hora de predecir.

Una vez se ha entrenado el modelo, podremos jugar contra la IA, esta a la hora de hacer su movimiento hará una predicción en base al tablero actual.

Por ejemplo, tras entrenar durante 100 iteraciones la IA nos ganó:

```
Tu turno (X)
Elige una casilla (0-8): 0
X


Turno de la IA (O)
X

  O
Tu turno (X)
Elige una casilla (0-8): 1
X X
```

```
  O
Turno de la IA (O)
X X O

  O
Tu turno (X)
Elige una casilla (0-8): 3
X X O
X
  O
Turno de la IA (O)
X X O
X
O  O
Tu turno (X)
Elige una casilla (0-8): 4
X X O
X X
O  O
Turno de la IA (O)
X X O
X X
O O O
¡La IA ganó!
```

Como se aprecia en la partida anterior, vemos que hemos intentado ganar en la primera fila, pero la IA nos ha bloqueado cuando teníamos intención de hacer el tercer movimiento en la celda 2. Entonces, vemos que ahora si que tiene inteligencia, esto debido a que ha ganado conocimiento sobre las distintas partidas que ha jugado por su cuenta.

En la siguiente gráfica vemos la tasa de error:

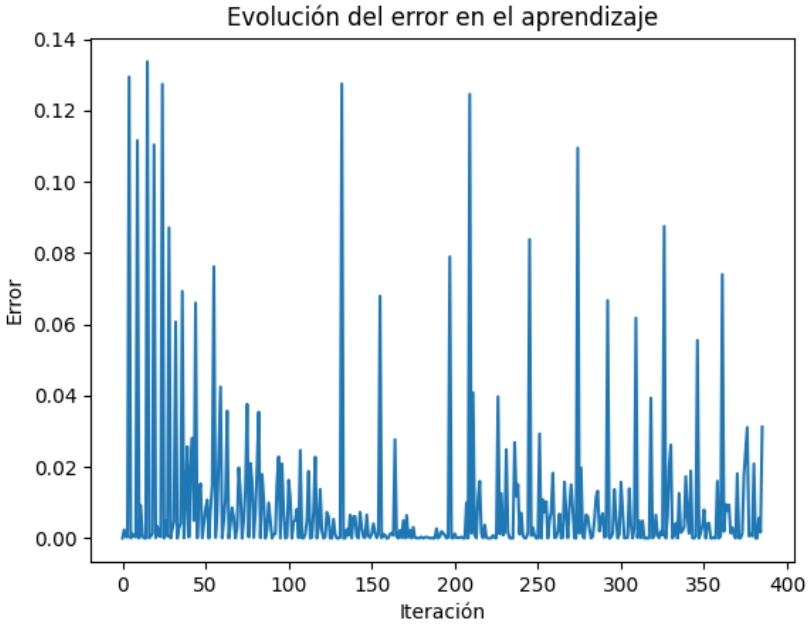

Figura 5: Error del modelo a través de las distintas iteraciones

Además, también hemos modificado el código para sacar la tasa de acierto a lo largo de las distintas iteraciones:

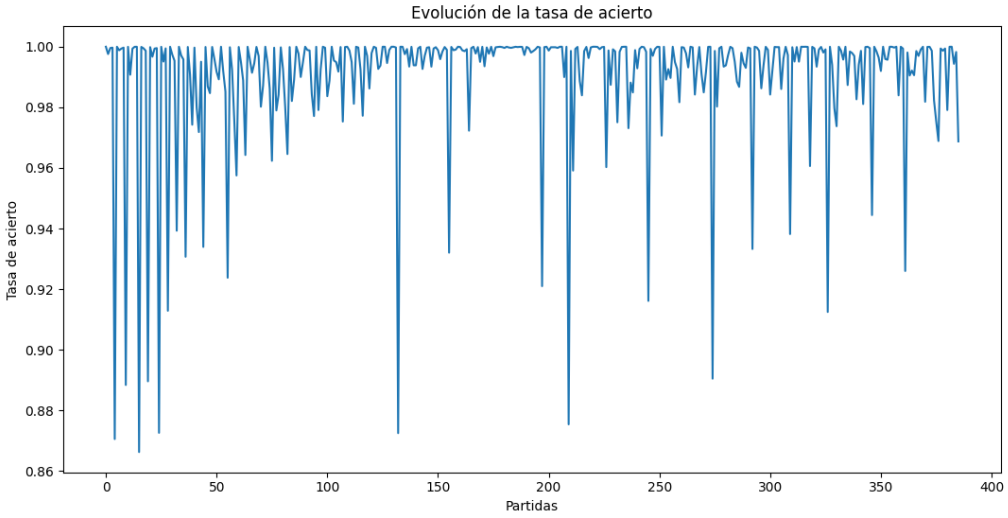

Figura 6: Acierto del modelo a través de las distintas iteraciones

Como vemos, la tasa de acierto mejora hasta las 120 primeras partidas, luego empieza a fallar en ciertas partidas, así que lo mejor es que entrenemos el modelo para 120 partidas.

5. Conclusiones

En esta práctica, se han implementado distintas aproximaciones para jugar al OXO (Tic-Tac-Toe), desde una inteligencia débil basada en movimientos aleatorios hasta una inteligencia fuerte que utiliza Q-Learning y redes neuronales. A partir de los resultados obtenidos, se pueden extraer las siguientes conclusiones:

- **OXO con Inteligencia Débil:** La IA que elige movimientos aleatorios no es competitiva. Carece de estrategia, lo que permite al jugador humano ganar fácilmente, mostrando las limitaciones de este enfoque básico.
- **OXO con Inteligencia Fuerte (Q-Learning):** El algoritmo Q-Learning mostró mejoras notables, aprendiendo a maximizar recompensas a largo plazo. Sin embargo, requiere ajustes finos en sus hiperparámetros y muchas iteraciones (aproximadamente 20,000) para ser eficaz.
- **OXO con Red Neuronal:** La IA basada en redes neuronales mostró un aprendizaje rápido y una mejora clara en sus decisiones, como bloquear al oponente. A partir de 120 iteraciones, el rendimiento es óptimo, pero entrenamientos adicionales pueden degradar el desempeño.
- **Comparación de Enfoques:** La inteligencia débil es insuficiente. Tanto Q-Learning como las redes neuronales permiten un aprendizaje efectivo, aunque las redes neuronales ofrecen mejores resultados en menos iteraciones.

En resumen, los experimentos demuestran que es posible entrenar una IA para que juegue al OXO de manera competitiva utilizando técnicas de aprendizaje por refuerzo o redes neuronales. Aunque ambos enfoques presentan ventajas y limitaciones, el uso de redes neuronales parece ofrecer un mayor control sobre el proceso de aprendizaje y mejores resultados en un menor número de iteraciones.