|  |  |
| --- | --- |
| UBA | Facultad de Ingeniería  aprendizaje por refuerzo  Desafío Final | Abstract  En este trabajo se aborda la resolución de un problema simple mediante Aprendizaje por Refuerzo, utilizando el algoritmo Q-Learning.  Charaf, Christopher – Villanueva, Azul  CEIA |

# 1. Introducción

Se diseña un entorno cuadrado 5x5 en el cual un agente inicia en la posición (0,0) y debe llegar a la posición (4,4). El entorno contiene casillas seguras (F), pozos (H) y una meta (G).

El agente puede moverse en cuatro direcciones y recibe +1 al alcanzar la meta, -1 si cae en un hueco y -0.01 en otros casos para incentivar un camino óptimo.

El episodio termina si cae en un pozo o si llega al objetivo. La política se aprende mediante Q-Learning.

A screen shot of a computer program

AI-generated content may be incorrect.

# 2. Implementación del Entorno y Algoritmo

Se desarrolló el entorno herendando de gymnasium.Env, utilizando espacios de observación y acción discretos. La tabla Q se inicializa en ceros y se actualiza mediante la ecuación clásica del algoritmo Q-Learning con política ε-greedy. Los parámetros utilizados fueron: tasa de aprendizaje 0.1, factor de descuento 0.99, ε inicial de 1.0 con decay hasta 0.01.

A screen shot of a computer program

AI-generated content may be incorrect.

# 3. Resultados y Gráfico de Convergencia

Se entrenó el agente durante 5000 episodios y se observó convergencia en la política aprendida, reflejada en un aumento progresivo de la recompensa media. A continuación, se presenta el gráfico de convergencia, pasos por episodio y tasa de éxito por cada 100 episodios, hasta 5000.

A graph with a green line

AI-generated content may be incorrect.

# 4. Política Aprendida

Una vez entrenado el agente, se imprime la política aprendida representada con flechas en la grilla, indicando la mejor acción en cada estado según la tabla Q. Las casillas con pozos se indican como 'H', la meta como 'G', y el resto con las direcciones ← ↓ → ↑.

A screenshot of a computer

AI-generated content may be incorrect.

# 5. Desafíos encontrados

* Inicialmente, el entorno definido contenía demasiados pozos y caminos poco accesibles, lo que impedía que el agente aprendiera una política útil. Esto fue resuelto rediseñando el mapa para que mantuviera obstáculos, pero permitiera rutas viables hacia la meta, en este caso hay una potencialidad, dados los parámetros correctos de entrenamiento e hiperparámetros, de aumentar la complejidad del entorno.
* El gráfico de convergencia se mostraba plano porque el agente no lograba alcanzar la meta en ningún episodio. Este comportamiento fue resuelto tras los cambios en el entorno y una mejora en la configuración de recompensas.
* La recompensa inicial del entorno era de tipo binario (1 si ganaba, 0 en todo otro caso). Se introdujeron penalizaciones pequeñas por paso y penalizaciones más fuertes por caer en pozos, mejorando el aprendizaje.
* Hubo varias oportunidades donde se ajustaron los hiperparámetros para acelerar la convergencia.

# 6. Conclusiones

Para concluir, el experimento demostró la efectividad del algoritmo Q-Learning en un entorno personalizado. El agente fue capaz de aprender una política óptima y generalizable sin el uso de librerías obsoletas.

El diseño modular permite extender el entorno y probar otros algoritmos como SARSA o Monte Carlo fácilmente.

# 6. Repositorio del código

[Github](https://github.com/christophcharaf/aprendizaje_por_refuerzo)