UNIVERSIDAD CENTRAL DE CHILE
FACULTAD DE INGENIERÍA
ESCUELA DE COMPUTACIÓN E INFORMÁTICA

ANÁLISIS DE APRENDIZAJE POR REFUERZO EXPLICABLE BASADO EN OBJETIVOS EN UN ENTORNO CONTINUO SIMULADO.

Memoria para optar al título profesional de
Ingeniero Civil en Computación e Informática.

Profesor Guía: **Francisco Cruz**
Profesor Informante: **Rodolfo Canelón**
Profesor Informante: **Alejandro Sanhueza**

Ernesto Portugal

Santiago, Chile
2021

Esta tesis está dedicada a mis padres que me han apoyado incondicionalmente en mis decisiones, a mis hermanas por regalarme buenos momentos y al profesor Francisco por guiarme en la realización de este trabajo.

El autor.

Resumen

Actualmente la Inteligencia Artificial esta en un importante periodo de crecimiento, debido al crecimiento de la tecnología es posible resolver problemas que antes no se podía. Por ejemplo, es posible que máquinas o agentes inteligentes puedan realizar tareas sin intervención del humano a través del aprendizaje automático, es decir, aprenden por si mismos. Sin embargo, por esto el humano no es capaz de entender como la máquina o el agente pudo llegar a la respuesta. Bajo estas circunstancias, la Inteligencia Artificial Explicable busca poder eliminar esta brecha.

El presente proyecto consiste en poner a prueba dos métodos de explicabilidad en entornos continuos. Los métodos de aprendizaje e introspección ocupan la probabilidad de éxito para poder explicar el comportamiento del agente, estos ya fueron puestos a prueba en entornos discretos. El entorno continuo utilizado es Car-Racing de la librería de Python Open AI Gym, el cual es un juego de carrera simulado. Los agentes dentro de este entorno son entrenados con el algoritmo Deep Q Network (DQN) y de forma paralela se implementan los métodos de explicabilidad. El proyecto abarca una propuesta para la adaptación e implementación de estos métodos para llevarse a cabo dentro de los entornos continuos. Además, se evalúa el rendimiento de los métodos en el uso de la memoria y uso del procesador.

Ambos métodos fueron posibles de adaptar al entorno, siendo el método de aprendizaje el que tuvo mayores cambios, implementado por medio de una red neuronal artificial. Las probabilidades de ambos fueron consistentes a lo largo de los experimentos. Teniendo un resultado de probabilidad mayor en el método de aprendizaje. El uso de recursos computacionales por parte del método de introspección fue ligeramente mejor que en su contra parte.

Índice general

Índice de figuras

Índice de tablas

Capítulo 1

Introducción

1.1. Motivación

El aprendizaje automático se ha masificado dentro de la vida cotidiana de las personas, sin darnos cuenta cada vez más algoritmos están analizando y aprendiendo a ayudarnos en ciertas cosas, por ejemplo, recomendaciones de películas o música personalizada, en búsqueda de correo basura, reconocimiento facial y también procesamiento de imágenes o vehículos autónomos. Por ello, en ciertas circunstancias es importante que el usuario pueda comprender lo que estos sistemas, máquinas o robots nos están informando, ya que de forma contraria no sería efectivo. Por ejemplo, no sería de utilidad tener un algoritmo que recomiende películas que no gustan, o también sería grave que un sistema de diagnóstico médico fallara al evaluar como negativo a un paciente con cáncer. En este contexto, el área de la inteligencia artificial explicable juega un papel importante, ya que consiste en herramientas, técnicas y algoritmos que permiten dotar al agente de la capacidad de explicar su actuar al humano de forma intuitiva (Das and Rad, 2020).

En particular en esta tesis, el problema se abordará dentro del área del aprendizaje por refuerzo explicable basado en objetivos, ocupando entornos continuos para simular una situación similar al mundo real. A través de una investigación por el método científico se busca contrastar los resultados de los métodos de explicabilidad implementados en los agentes, comparando la probabilidad de éxito obtenida por cada uno de ellos.

1.2. Definición del Problema

En escenarios de interacción de humano-robot, es posible que un usuario no experto no comprenda por que el robot tomo cierta decisión. Entonces, surgen ciertas preguntas que el usuario podría realizar bajo estas circunstancias: por qué, por qué no, como, y si, que (Lim et al., 2009).

En el contexto del Aprendizaje por Refuerzo Explicable basado en objetivos (Explainable Goal-Driven Reinforcement Learning, XGDRL, siglas en inglés), se puede dar la situación de, por ejemplo, un robot dentro de un laberinto que dobla a la derecha en la intersección A, pero el usuario no logra entender por qué o como llego a esa solución. Por lo que le pregunta "¿Por qué has doblado a la derecha?", entonces el robot podría dar la siguiente respuesta: "He doblado a la derecha porque es la opción con más probabilidades de llegar a la meta".

El problema tiene lugar con la comprensión y confianza que se establezca entre un sistema autónomo y un humano no experto al momento de interactuar, es decir, si es que el usuario puede comprender las razones que le entrega una inteligencia artificial de la conclusión a la que llego (Gunning, 2017). En esta situación, se particulariza el problema dentro del área del aprendizaje por refuerzo explicable basado en objetivos, en donde según Sado et al. (2020) todavía faltan estudios en este ámbito. Por ello, aumentar la explicabilidad de las inteligencias artificiales beneficiaria estos sistemas, ya que cualquier usuario sería capaz de comprender las acciones tomadas y por lo tanto facilitaría la confianza.

1.3. Propuesta

El proyecto consiste en desarrollar y evaluar algoritmos de aprendizaje por refuerzo en entornos continuos que puedan explicar su comportamiento. Se evaluarán dos métodos de explicabilidad: aprendizaje e introspección. Cada uno de ellos, utilizará la probabilidad de éxito para explicar sus resultados. Además, se considerarán recursos computacionales, en particular, la memoria utilizada.

1.4. Objetivos

1.4.1. Objetivo General

Evaluar algoritmos de aprendizaje por refuerzo en entornos continuos que puedan explicar su comportamiento de acuerdo a su objetivo a través de la probabilidad

de éxito, considerando recursos computacionales.

1.4.2. Objetivos Específicos

En el presente trabajo, se han definido los siguientes objetivos específicos:

- Estudiar sobre aprendizaje por refuerzo y entornos continuos.
- Seleccionar entornos continuos para evaluar a los agentes inteligentes.
- Desarrollar algoritmos de aprendizaje por refuerzo para entornos continuos que permitan explicar su comportamiento basados en los métodos de aprendizaje e introspección.
- Entrenar agentes inteligentes en base a los algoritmos propuestos
- Analizar resultados obtenidos según probabilidad de éxito y uso de recursos.

1.5. Hipótesis

El algoritmo de aprendizaje por refuerzo con el método de introspección es más efectivo al uso de los recursos computacionales, en particular la memoria, según el orden del algoritmo.

1.6. Metodología de Trabajo

La metodología del proyecto será el método científico (Nola y Sankey, 2014, citado en Moreira (2019)).

- Planteamiento del problema: Un agente inteligente bajo el aprendizaje por refuerzo, interactuando en un entorno continuo, ¿está apto para explicar su comportamiento?
- Examen y análisis de los enfoques existentes: Revisar el estado del arte sobre aprendizaje por refuerzo explicable.
- Construcción del escenario experimental: Se desarrollarán los algoritmos con los distintos métodos y se definirá los escenarios a utilizar.
- Revisión de resultados y análisis: Se evaluará la probabilidad de éxito con respecto a cada método y también el uso de la memoria.
- Resultados del informe: Los resultados quedaran reflejados en la tesis.

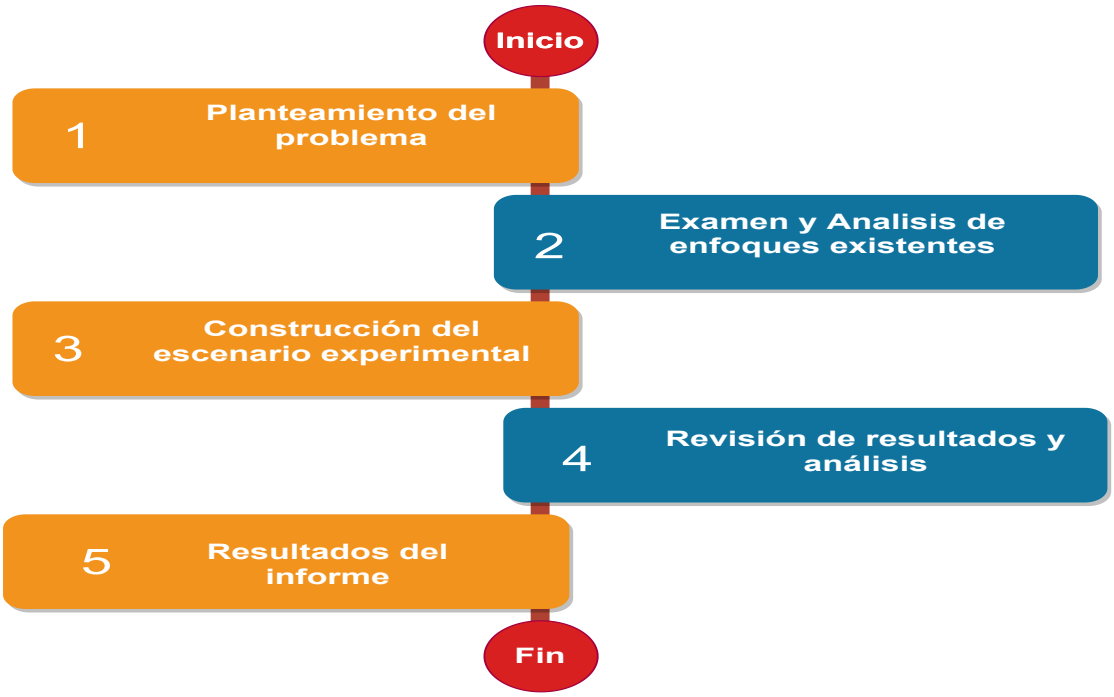


Figura 1.1: Los cinco pasos llevados a cabo en el método científico. Adaptada desde (Nola y Sankey, 2014, citado en Moreira (2019)).

1.6.1. Cronograma

El cronograma del proyecto se encuentra en el Apéndice A y se compone de los siguientes puntos.

1. Planteamiento del problema.
    a) Establecer el problema.
    b) Establecer herramientas para resolver el problema.
2. Examen y análisis de los enfoques existentes.
    a) Revisar el estado del arte.
    b) Analizar desde una perspectiva critica los resultados obtenidos previamente.
3. Construcción del escenario experimental.
    a) Estudio de simuladores.
    b) Definir el simulador.

   c) Configurar el entorno continuo.

   d) Estudio de algoritmos de aprendizaje por refuerzo.

   e) Desarrollo de los algoritmos.

   f) Entrenamiento de los agentes.

4. Análisis de resultados

5. Resultado del informe

   a) Finalización de la Tesis.

   b) Publicación del código en repositorio de acceso público.

1.7. Alcances y Limitaciones

1.7.1. Alcances

- La investigación se centra en los métodos que son capaces de hacer que el agente inteligente pueda explicar su comportamiento. No se tendrá en cuenta el rendimiento de aprendizaje.

- El entorno continuo a definir se enfocará en investigar un escenario en el cual el agente pueda interactuar y así explicar su comportamiento. No se hará análisis sobre el funcionamiento o rendimiento del entorno.

1.7.2. Limitaciones

Las pruebas se realizarán solamente en entornos simulados, debido a que no se cuenta con robots o máquinas físicas para crear agentes de aprendizaje por refuerzo en un entorno real.

1.8. Estructura del Documento

Se presenta un resumen de la composición del presente documento.

1. Introducción: Capitulo actual que presenta el contexto, la motivación, el problema a tratar y la metodología empleada para afrontar el problema.

2. Marco Teórico y Estado del Arte: Se abarca dentro del marco teórico dos temas principales, el aprendizaje por refuerzo y la inteligencia artificial explicable.
3. Métodos de Explicabilidad: Se expondrán los métodos de explicabilidad a analizar dentro del trabajo.
4. Implementación: Este capítulo se describe el escenario experimental y la adaptación de los métodos de explicabilidad a los entornos continuos.
5. Resultados y Análisis: Se mostraran los resultados de los experimentos y un análisis de estos.
6. Conclusiones: Este capítulo resume las principales ideas del trabajo realizado, discute los resultados.

A. Cronograma: Este apéndice contiene el cronograma del proyecto.

Capítulo 2

Marco Teórico y Estado del Arte

En este capitulo se expone los temas que están relacionados al trabajo de esta investigación. El contexto de la problemática se enfoca en el área de Inteligencia Artificial. En particular, se tratará el tema del Aprendizaje por Refuerzo, subárea del Aprendizaje Automático, la cual es utilizada como base de los agentes inteligentes dentro de la investigación. Luego, se hablará de la Inteligencia Artificial Explicable (Explainable Artificial Intelligence, XAI).

2.1. Aprendizaje por Refuerzo

El aprendizaje por refuerzo (Reinforcement Learning, RL) es un enfoque dentro del área de aprendizaje automático (Machine Learning), que se caracteriza por el aprendizaje a través de la interacción entre un agente y su entorno para lograr metas de largo alcance (Sutton and Barto, 2018). Se basa en la forma como animales obtienen experiencia para la toma de decisiones y planificación, es decir, realizar acciones con el propósito de obtener una recompensa que entrega el entorno, y evaluar esta recompensa según el objetivo final. Por ejemplo, dos perros cachorros juegan y uno de ellos es más agresivo que el otro, llamémoslo A y el otro perro se llamará B, por lo tanto, al momento de jugar el perro A muerde al otro y el perro B llora por ello, alejándose y no queriendo jugar de nuevo. Así, el perro A se da cuenta de que no debe morder tan fuerte cuando juegan. El agente es aquel que aprende y realiza la toma de decisiones. Por otro lado, todo lo que rodea al agente y con lo que puede interactuar es el entorno. En la Fig. 2.1 se muestra un diagrama de la interacción entre el agente y el entorno.

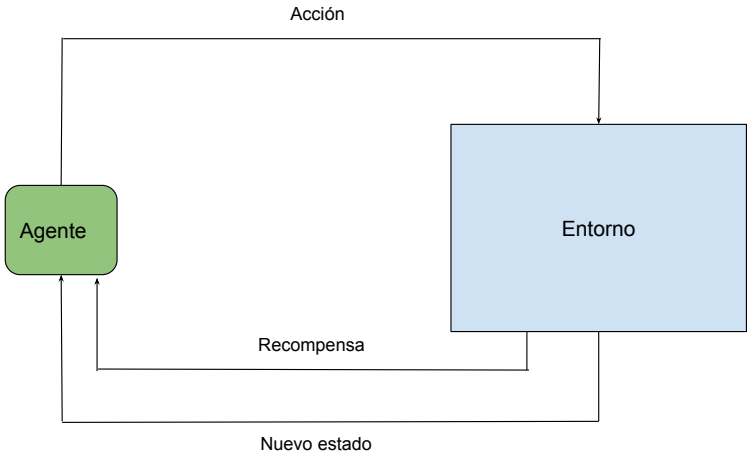


Figura 2.1: Un agente toma alguna acción y el entorno responde al acto. Con ello el agente va interactuando con el entorno y generando el aprendizaje. Adaptada de (Sutton and Barto, 2018)

Sutton and Barto (2018) menciona que este enfoque se compone de 4 subelementos, aparte del agente y el entorno, los cuales son: una política, una señal de recompensa, una función de valor y un modelo sobre el entorno.

- La política determina cómo se comporta el agente dado algún estado dentro del entorno. En general, las políticas pueden ser estocásticas.

- La señal de recompensa, proporcionada por el entorno, le da un estímulo al agente de acuerdo a la acción tomada. Con ello el agente puede reconocer si es que la acción elegida es buena o mala, el objetivo principal del agente es maximizar la recompensa. Esta señal puede ser estocásticas.

- La función de valor determina si es que las acciones tomadas son buenas a largo plazo. Prácticamente esta función entrega el total de recompensa que se puede aspirar en un determinado estado.

- El modelo del entorno permite al agente planificar. Este es una representación del entorno de tal forma que el agente pueda predecir cómo se comportara el entorno de acuerdo a alguna acción. El modelo es opcional, hay métodos que usan modelos que se catalogan como métodos basados en modelos, y en cambio, hay métodos sin ellos, los cuales se llaman métodos sin modelo que son de prueba y error.

2.1.1. Aprendizaje de Diferencia-Temporal

El aprendizaje de Diferencia-Temporal ocupa la experiencia para resolver el problema. De acuerdo a alguna experiencia según la política π, se debe actualizar el estimado del valor V para cada estado no terminal. Los métodos de Diferencia Temporal estiman este valor en cada paso de tiempo (t). Mediante la actualización de la función de valor $V(S_t)$ según la Eq. 2.1 (Sutton and Barto, 2018)

$$V(S_t) \leftarrow V(S_t) + \alpha \left[R_{t+1} + \gamma V(S_{t+1}) - V(S_t)\right] \tag{2.1}$$

Dentro de este contexto, se destaca el método Q Learning el cual considera la función de valor de acción (q_π) en vez de la función de valor de estado. Como se muestra en la Eq. 2.2

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha \left[R_{t+1} + \gamma \underset{a}{max} Q(S_{t+1}, a) - Q(S_t, A_t)\right] \tag{2.2}$$

2.1.2. Deep Q Network

Deep Q Network (DQN) combina el método Q Learning con una red neuronal artificial (Sutton and Barto, 2018). La red es una red neuronal multicapa que procesa un estado y da como resultado una predicción de los valores Q para cada acción ($Q(s, \cdot; \theta)$, donde θ son los parámetros de la red neuronal) . Hay dos importantes aspectos para el funcionamiento del algoritmo DQN, el uso de una red neuronal objetivo (target network) y el uso de la repetición de las experiencias. Los parámetros de la red neuronal objetivo se denotan por θ_t^- y se copian cada τ pasos desde la red base. El objetivo usado por el algoritmo se muestra en la Eq. 2.3 (Van Hasselt et al., 2016)

$$Y_t^{DQN} \equiv R_{t+1} + \gamma \underset{a}{max} Q(S_{t+1}, a; \theta_t^-) \tag{2.3}$$

2.2. Inteligencia Artificial Explicable

2.2.1. Caja Negra

El problema de la caja negra dentro del área de la inteligencia artificial, se refiere a la dificultad del sistema para poder entregar una explicación apropiada para la respuesta que obtuvo (Adadi and Berrada, 2018). Los modelos empíricos no revelan de que manera generan las decisiones, por lo tanto lo que se conoce son

Figura 2.2: El problema de la caja negra gráficamente representado. Se conocen las entradas y salidas, pero se desconoce que ocurre dentro del sistema.

las entradas y las salidas del sistema. Por ejemplo, los celulares inteligentes son pequeñas máquinas que realizar muchas tareas: llamar a otro teléfono, conectarse a Internet y navegar por ella, sacar fotos y vídeos, reproducir música, etc. Sin embargo, los usuarios de estos no conocen como funcionan por dentro. Se da por hecho que funciona, pero en general no se conocen los procesos que se llevan a cabo. En la Inteligencia artificial, esto puede presentar un problema, ya que en ciertas ocasiones es necesario entender el proceso interno dentro del sistema. En la Fig. 2.2 se presenta este problema gráficamente.

2.2.2. Interacción Humano Robot

La interacción humano-robot (HRI sigla en inglés) es el estudio aplicado a sistemas robóticos que trabajan con o para humanos (Goodrich and Schultz, 2007). Este campo requiere de comunicación entre las partes, por ello se puede clasificar en: interacción remota; humano y robot no se encuentran físicamente en el mismo lugar, o incluso en el mismo tiempo, y en interacción próxima; al contrario de la anterior, se encuentran en el mismo lugar físico y temporal.

2.2.3. Explicabilidad

En escenarios HRI en donde es importante la confianza, la explicabilidad en la inteligencia artificial toma relevancia. Mediante la transparencia en los sistemas, se logra que los usuarios pueden ser capaces de comprender y confiar en las decisiones tomadas dentro del sistema de un agente inteligente (Gunning, 2017). La inteligencia artificial explicable será fundamental para los sistemas cooperativos entre humano y máquina, aumentando la efectividad de estos.

2.3. Estado del Arte

Diversos trabajos se han estado realizando dentro del campo de la inteligencia artificial explicable, Adadi and Berrada (2018) realizaron un estudio sobre esta área

recabando términos comunes y clasificando los métodos de explicabilidad existentes. Se destaca el dominio de aplicación, habiendo trabajos dentro de áreas como: transporte, salud, legal, finanzas y militar. Por otro lado, los autores Lamy et al. (2019) postulan una inteligencia artificial explicable a través del método de razonamiento basado en casos (Case-Based Reasoning, CBR) ocupando el algoritmo de k vecinos más próximo ponderado (Weighted k Nearest Neighbor, WkNN), aplicado al diagnóstico de cáncer de mamas.

En cuanto al aprendizaje por refuerzo explicable se busca dotar al agente de un método para explicarse dentro del proceso de aprendizaje. Por ejemplo, Madumal et al. (2020) propusieron un enfoque que aprende un modelo causal estructural durante el proceso de aprendizaje por refuerzo. El modelo sirve para generar explicaciones mediante análisis contrafactual. Lo aplicaron a participantes que veían a agentes jugar un juego de estrategia en tiempo real (Starcraft II) y luego proporcionaban explicaciones del comportamiento. Sequeira and Gervasio (2020) proponen un marco para agentes de aprendizaje por refuerzo por medio de análisis introspectivo. Se sugiere tres niveles de análisis, en principio a través de recolección de información de elementos de interés en el entorno. En segundo lugar, analizar la interacción con el entorno. Como tercer paso, mezclar los resultados obtenidos realizando un último análisis.

En específico hay un área sobre explicabilidad enfocado en objetivos, es decir, el agente inteligente puede dar respuestas sobre su comportamiento de acuerdo a su meta. Por ejemplo, el agente puede decir que tiene 60 % de probabilidad de lograr la meta si toma el camino "A", y en cambio tiene 40 % de probabilidad si toma el "B". Ferreyra et al. (2019) proponen un marco de un sistema de inteligencia artificial explicable basado en lógica difusa (en específico, big bang-big crunch interval type-2 fuzzy logic system, BB-BC IT2FLS) simulado en base a los objetivos para ayudar en el dominio de asignación de personal, en particular, dentro de la industria de telecomunicaciones. En primer lugar, proponen un sistema basado en lógica difusa para imitar como el humano piensa y, en segundo lugar, realizaron la simulación en consideración con los objetivos.

Cruz et al. (2019) trabajaron en aprendizaje por refuerzo explicable basado en memoria. Este enfoque se basa en una memoria episódica que permite poder explicar sus decisiones con respecto a la probabilidad de éxito y la cantidad de pasos para llegar al objetivo. Sin embargo, se presentan problemas en escenarios más amplios, debido a que la memoria es finita. Nuevamente, Cruz et al. (2020) expanden su trabajo aumentando el número de enfoques, se considera el mismo enfoque basado

en memoria y agrega los enfoques basados en aprendizaje e introspección. Se utiliza un escenario episódico, con transiciones deterministas y estocásticas. Por ello, no es aplicable a situaciones de mundo real.

2.4. Discusión

Dentro de la revisión del Marco Teórico se habla sobre conceptos como: Aprendizaje por Refuerzo, Algoritmo DQN, Inteligencia Artificial Explicable, los cuales son conceptos clave para la resolución del problema. Los agentes utilizados ocuparan el algoritmo DQN para aprender a realizar su tarea, el cual esta basado en el algoritmo Q Learning y redes neuronales artificiales. Además, se utiliza enfoques de la Inteligencia Artificial Explicable para que los agentes puedan justificar su comportamiento.

El Estado del Arte permite saber que hacen falta trabajos con respecto al tema de Aprendizaje por Refuerzo Explicable. También dando el pie de partida para el uso de los métodos propuestos para poner a prueba en entornos continuos.

Capítulo 3

Métodos de Explicabilidad

Los métodos utilizados para dotar de explicabilidad al agente son el método de aprendizaje y el método de introspección. Estos métodos se agregan al algoritmo del aprendizaje por refuerzo, pero cada uno tiene una manera distinta de calcular la probabilidad de éxito. Los métodos han sido probados dentro de entornos discretos, por lo que dentro de este capítulo se verá su definición.

3.1. Método de Aprendizaje

Este método consiste en aprender la probabilidad de éxito de forma similar a como se aprende los valores Q, la probabilidad se va actualizando a medida que se ejecutan las acciones y se asigna la probabilidad para cada par estado-acción, en la Eq. 3.1 se muestra el cálculo de la probabilidad (línea 10) . Este proceso se va ejecutando de manera estática, es decir, se mantiene una tabla que pueda almacenar las probabilidades.

$$\mathbb{P}(s_t, a_t) \leftarrow \mathbb{P}(s_t, a_t) + \alpha[\phi_{t+1} + \mathbb{P}(s_{t+1}, a_{t+1}) - \mathbb{P}(s_t, a_t)] \tag{3.1}$$

En la Eq. 3.1, $\mathbb{P}(s_t, a_t)$ hace referencia a la probabilidad de acuerdo al estado y acción, α es la tasa de aprendizaje, ϕ_{t+1} corresponde a un valor binario correspondiente a si es que la tarea fue completada o no. En el Alg. 3.1 se muestra el pseudocódigo correspondiente al método de aprendizaje para entornos discretos.

Algorithm 3.1. Enfoque de Inteligencia Artificial Explicable para computar la probabilidad de éxito usando el método de aprendizaje. (Cruz et al., 2020)

1: Inicializar $Q(s,a), \mathbb{P}(s_t, a_t)$
2: **for** cada episodio **do**
3: Inicializar s_t
4: Elegir una acción a_t según s_t
5: **repeat**
6: Realizar acción a_t
7: Observar recompensa r_{t+1} y el estado siguiente s_{t+1}
8: Elegir siguiente acción a_{t+1} usando el método de selección softmax
9: $Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \left[r_{t+1} + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)\right]$
10: $\mathbb{P}(s_t, a_t) \leftarrow \mathbb{P}(s_t, a_t) + \alpha \left[\phi_{t+1} + \mathbb{P}(s_{t+1}, a_{t+1}) - \mathbb{P}(s_t, a_t)\right]$
11: $s_t \leftarrow s_{t+1}; a_t \leftarrow a_{t+1}$
12: **until** s_t es terminal (meta o estado aversivo)
13: **end for**

3.2. Método de Introspección

Este método utiliza el valor de Q para hacer una estimación con respecto a la probabilidad de éxito.

$$\hat{P_S} \approx \left[(1-\sigma)\cdot(\frac{1}{2}\cdot\log\frac{Q^*(s,a)}{R^T}+1)\right]_{\hat{P_S}\geq 0}^{\hat{P_S}\leq 1} \tag{3.2}$$

En la Eq. 3.2 se presenta la estimación utilizada, σ representa la estocástica de las transiciones, $Q^*(s,a)$ representa el valor Q óptimo, R^T corresponde a la recompensa total y debido a que $\hat{P_S}$ corresponde a una probabilidad ($\hat{P_S} \in [0,1]$) se hace una rectificación denotada por $[...]_{\hat{P_S}\geq 0}^{\hat{P_S}\leq 1}$.

En el Alg. 3.2 se tiene el algoritmo para el método de introspección. Se destaca la línea 12 del algoritmo, en donde se presenta la fórmula ocupada para calcular la probabilidad de éxito.

Algorithm 3.2. Enfoque de Inteligencia Artificial Explicable para computar la probabilidad de éxito usando el método de introspección. (Cruz et al., 2020)

Inicializar $Q(s,a), \hat{P}_S$
for cada episodio **do**
    Inicializar s_t
    Elegir una acción a_t según s_t
    repeat
        Realizar acción a_t
        Observar recompensa r_{t+1} y el estado siguiente s_{t+1}
        Elegir siguiente acción a_{t+1} usando el método de selección softmax
        $Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha\left[r_{t+1} + \gamma Q(s_{t+1},a_{t+1}) - Q(s_t,a_t)\right]$
        $s_t \leftarrow s_{t+1}; a_t \leftarrow a_{t+1}$
    until s_t es terminal (meta o estado aversivo)
    $\hat{P}_S \approx \left[(1-\sigma)\cdot(\frac{1}{2}\cdot\log\frac{Q^*(s,a)}{R^T}+1)\right]_{\hat{P}_S\geq 0}^{\hat{P}_S\leq 1}$
end for

Capítulo 4

Implementación

4.1. Escenario Experimental

El escenario experimental es desarrollado usando la librería Open AI Gym (Open AI, 2021), la cual facilita el desarrollo de algoritmos de aprendizaje por refuerzo. Esta librería cuenta con muchos entornos con los que poder trabajar, en particular se eligió el entorno que simula el juego "Car Racing" en la versión 0, ya que cumple con la condición de ser un entorno continuo. Por ello, se hace uso de la implementación de Deep Q Network (DQN) publicada en el sitio web Github por parte de Wu, Andy (2021), en donde utiliza TensorFlow y Keras [1] para crear la red neuronal. Además, la elección de acciones esta dada por el método de decaying-epsilon-greedy.

[1] TensorFlow es una plataforma de código abierto para el aprendizaje automático. Keras es un entorno de trabajo construido sobre TensorFlow que ofrece simplicidad en el uso de las APIs.

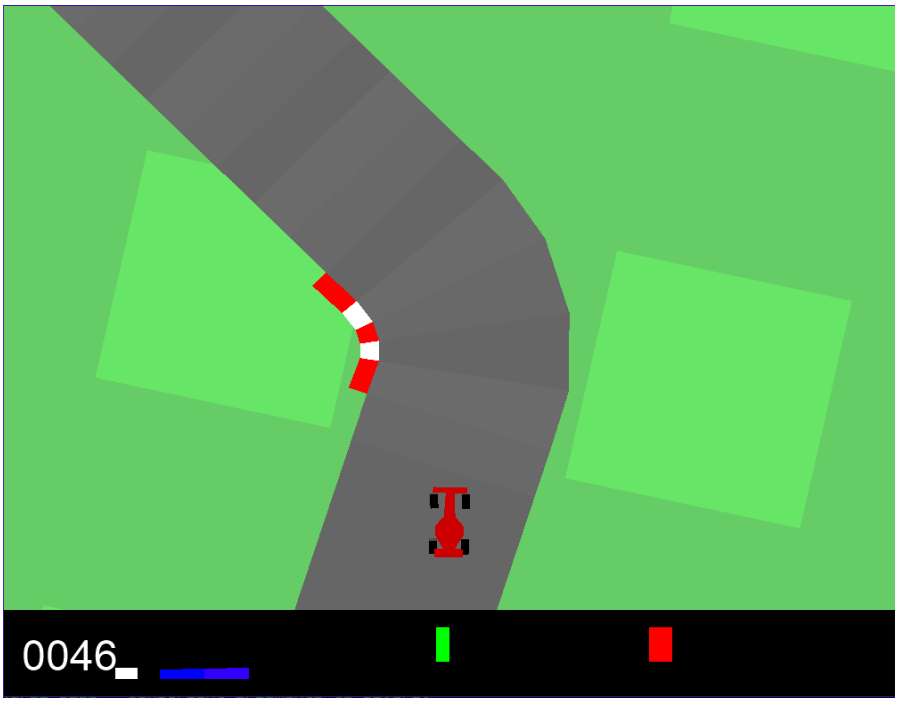
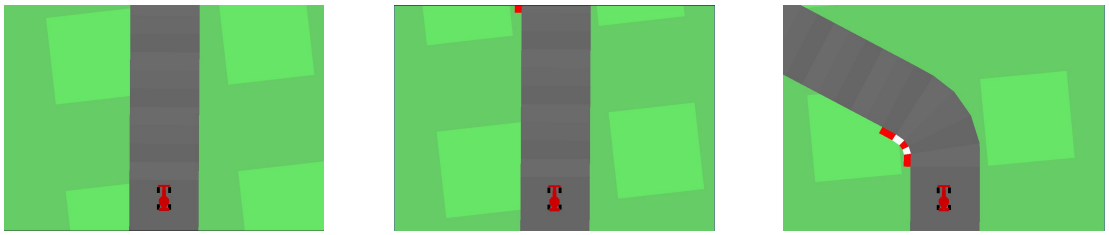


Figura 4.1: Escenario experimental. En la parte inferior, de izquierda a derecha, se muestran en colores los indicadores de velocidad (blanco), los cuatro sensores ABS (azul y morado), la posición del manubrio (verde) y el giroscopio (rojo).

El entorno del juego consiste en una pista de carrera en donde el jugador controla en automóvil. El objetivo del juego es visitar todas las casillas de la pista. Para ello, el jugador o en este caso el agente tendrá 12 acciones posibles para realizar las cuales son las combinaciones de 3 aspectos: dirección del manubrio, aceleración y freno. Los dos últimos son binarios, mientras que la dirección del manubrio tiene 3 opciones: izquierda, centro y derecha.

Los estados se representan a través de 3 imágenes consecutivas del juego (Fig. 4.2), ya que cada imagen se conforma por una matriz de 96 x 96 píxeles, el estado queda representado por una matriz de $96 \times 96 \times 3$. Sin embargo, en este caso el color no cumple un rol importante, por lo que las imágenes son previamente procesadas para que queden en la escala de grises.



(a) t=0 **(b)** t=1 **(c)** t=2

Figura 4.2: El estado se representa a través de 3 imágenes consecutivas del juego "Car Racing" de 96 x 96 (matriz de 96 × 96 × 3, las imágenes dentro de la figura son a modo de ejemplo, ya que previamente se procesan para que queden en escala de grises).

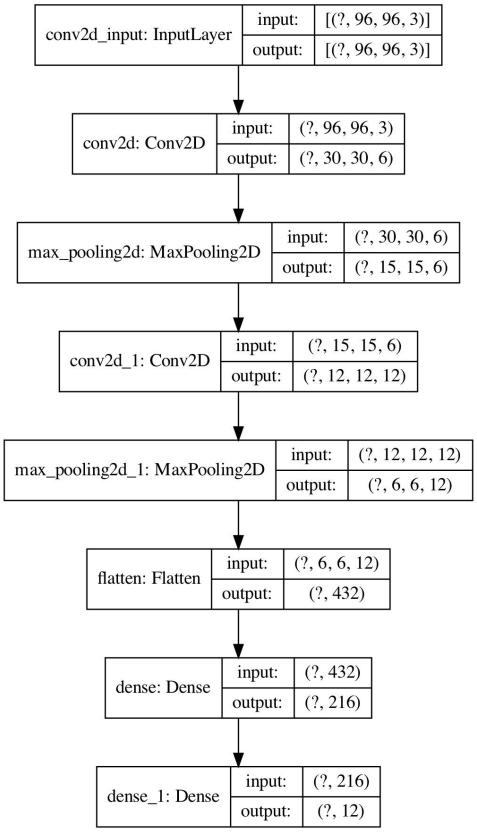


Figura 4.3: Diagrama de la red neuronal utilizada para el cálculo de los valores Q dentro del algoritmo de DQN en el repositorio de Andy Wu. Recibe como entrada 3 imágenes consecutivas del juego "Car Racing" en escala de grises de 96 x 96.

La estructura de red neuronal se muestra en la Fig. 4.3, en el diagrama se ocupan capas convolucionales para el procesamiento de imagen, además de capas de agru-

pación máxima para reducir la información. Con esta red se ingresa 3 imágenes consecutivas del juego y el resultado consiste en un valor estimado de Q para cada una de las 12 acciones.

4.2. Adaptación de los Métodos

Siguiendo los algoritmos propuestos por Cruz et al. (2020), se propone poner bajo estudio los métodos de aprendizaje e introspección para que el agente sea capaz de expresar sus decisiones bajo términos comprensibles para el ser humano dentro de entornos continuos.

4.2.1. Aprendizaje

Para el método de aprendizaje se presenta el problema de la infinidad de estados posibles, por ello el método no se puede trabajar con una matriz de par estado-acción para ir aprendiendo la probabilidad de éxito. Se propone que de forma similar al algoritmo DQN que implementa una red neuronal para aprender los valores de Q, se ocupa una red similar para aprender la probabilidad de éxito.

La red neuronal ocupada para que aprenda la probabilidad de éxito se compone de la misma estructura que tiene la del algoritmo DQN (Fig. 4.3). En principio, el objetivo de la red se cambió para seguir la ecuación presente en la Eq. 3.1. En ella se efectúa un método de aprendizaje similar al de los algoritmos de diferencia temporal, con la diferencia de que el factor de descuento este fijo en 1, además que no se ocupa la recompensa para actualizar los valores, en cambio se ocupa un variable binaria que corresponde a 0 si es que no se ha completado la tarea, y un 1 si es que se ha completado, denotado por la letra griega minúscula ϕ.
Luego de unos experimentos se notaron inconsistencias en los resultados, como la estructura de la red neuronal tenia capas con funciones de activación de unidad lineal rectificada (ReLU), la probabilidad no estaba acotada entre 0 y 1, por ello se cambio la función de activación, a la última capa, a una sigmoidea. Con el propósito de obtener valores en el intervalo $[0, 1]$. Este cambio tampoco tuvo buenos resultados, a lo largo de los episodios la probabilidad se mantenía estática dentro de un valor cercano a 0,6.

Finalmente, se propone ocupar la misma estructura (Fig. 4.3) y objetivo de la red de DQN (Eq. 2.3), con la última capa con la función de activación sigmoidea.

4.2.2. Introspección

El método de introspección esta dado por el cálculo de la probabilidad de éxito a través de la ecuación presente en la Eq. 3.2. Dado que este método requiere el valor Q en el estado terminal junto a la recompensa que se ha recabado hasta ese estado. Se intentó implementar de acuerdo a la fórmula, pero hubieron resultados que sugerían que la acotación superior propuesta estaba ocultando parte de los resultados. Por ello, se propuso una normalización de los datos para que quedaran en el intervalo $[0, 1]$. Por ello, en la Eq. 3.2 la notación $[...]_{\hat{P_S}\geq 0}^{\hat{P_S}\leq 1}$ que representa la rectificación se reemplaza por la función de normalización que se refleja en la Eq. 4.1. En donde $\hat{P}$ corresponde a un conjunto de datos de probabilidades de éxito calculadas a partir de la transformación del valor Q y R^T.

$$\begin{aligned} \hat{P_S} &\approx \hat{p_S} \in \hat{P} \\ \Rightarrow \hat{p_S} &= \frac{\hat{p} - \text{mín}(\hat{P})}{\text{máx}(\hat{P}) - \text{mín}(\hat{P})} \\ \Rightarrow \hat{p} &= (1-\sigma) \cdot (\frac{1}{2} \cdot \log \frac{Q^*(s,a)}{R^T} + 1) \end{aligned} \tag{4.1}$$

Capítulo 5

Resultados y Análisis

En este capítulo se muestran los resultados de los experimentos. Para el primer experimento, enfocado en la adaptación de los métodos, se ejecutaron 5 agentes a través del algoritmo DQN para que puedan aprender a manejar en el entorno de carreras del escenario experimental, calculando la probabilidad de éxito con ambos métodos. En el segundo experimento, que tiene por objetivo analizar el uso de recursos, se efectuó la ejecución de 3 agentes exclusivamente para el método de aprendizaje y otros 3 agentes para el método de introspección.

5.1. Valores Iniciales

El entrenamiento de los agentes tiene los parámetros que se indican en la Tabla 5.1. El primer experimento tiene un entrenamiento de 500 episodios, mientras que el segundo experimento tiene 200 episodios por agente.

Tabla 5.1: Parámetros de aprendizaje por refuerzo.

	Epsilon inicial	*Epsilon Decay*	*Tasa de aprendizaje*
Valor	1.0	0.9999	0.001

Además, en la Tabla 5.2 se muestran los parámetros con respecto a la red neuronal del algoritmo DQN.

Tabla 5.2: Parámetros de la red neuronal del Algoritmo DQN.

	Tamaño del lote	Tamaño de la memoria
Valor	64	5000

5.2. Valores Q y Recompensa

En la Fig. 5.1 se muestran los resultados del promedio de los valores Q de entre las 12 acciones posibles de los agentes y en la Fig. 5.2 se muestra la recompensa total, ambos gráficos muestran los valores para cada agente a lo largo de los 500 episodios. Ambas gráficas están suavizadas a través del método de media móvil por convolución linear discreta de un conjunto de treinta datos. Estos valores se muestran como referencia para poder analizar los resultados de las probabilidades de éxito.

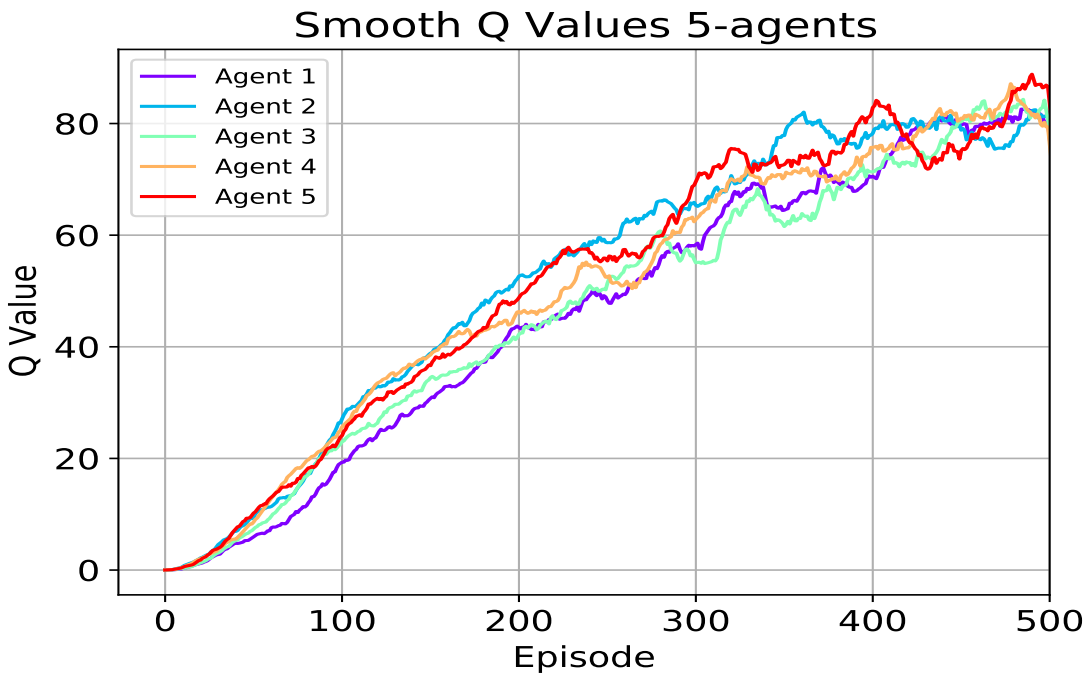


Figura 5.1: Valores promedio de Q para 5 agentes en los 500 episodios.

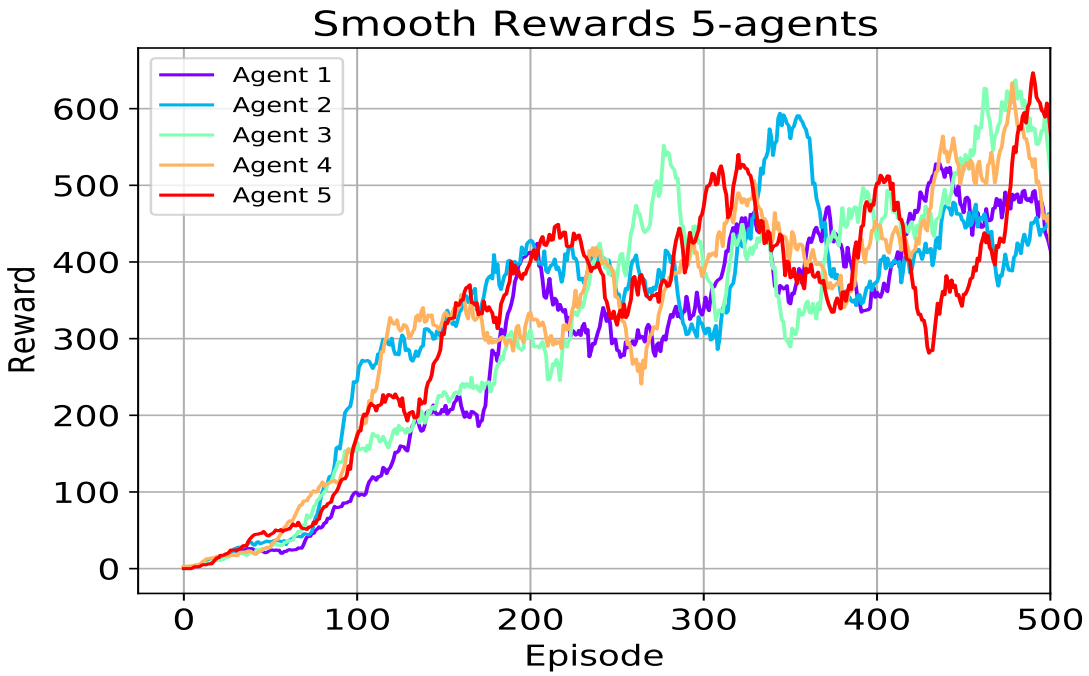


Figura 5.2: Valores de la recompensa total obtenida para 5 agentes en los 500 episodios.

5.3. Método Aprendizaje

De acuerdo a la adaptación planteada en la Sección 4.2, se obtuvieron resultados como se muestra en la Fig. 5.3, éstos se componen de la media de las probabilidades de éxito entre los agentes, en donde para cada agente se obtuvo el promedio de las probabilidades entre todas las acciones de acuerdo a lo computado por la red neuronal artificial dedicada. En los primeros 75 episodios se mantiene el promedio de la probabilidad relativamente constante cercano a $\mathbb{P} \approx 0{,}5$. Alrededor del episodio 75, los agentes muestran una mejora en la probabilidad, pero se detiene en aproximadamente el episodio 100, teniendo un valor cercano a $\mathbb{P} \approx 0{,}75$. Luego, la probabilidad de éxito fluctúa en lo que resta del entrenamiento pero manteniéndose cercano al mismo valor mencionado. Esto se interpreta como: el agente alrededor del episodio 500 tiene un 75 % de probabilidades de completar la tarea. Por lo cual si, por ejemplo, un jugador de Car-Racing quisiera preguntarle al agente por qué en la primera curva de la pista dobló hacia la izquierda en vez de la derecha, si el agente responde a con la probabilidad de éxito diría: "Tengo 75 % de probabilidades de completar la pista si elijo esta acción". En cambio, si respondiera con el valor de Q: "Tengo un valor $Q \approx 80$", lo cual para el jugador no tendría ningún

sentido.

Aun cuando la adaptación propuesta esta relacionada directamente con los valores Q, se aprecia un comportamiento escalonado para el caso de la probabilidad mientras que en el caso de los valores Q es similar a un comportamiento lineal creciente.

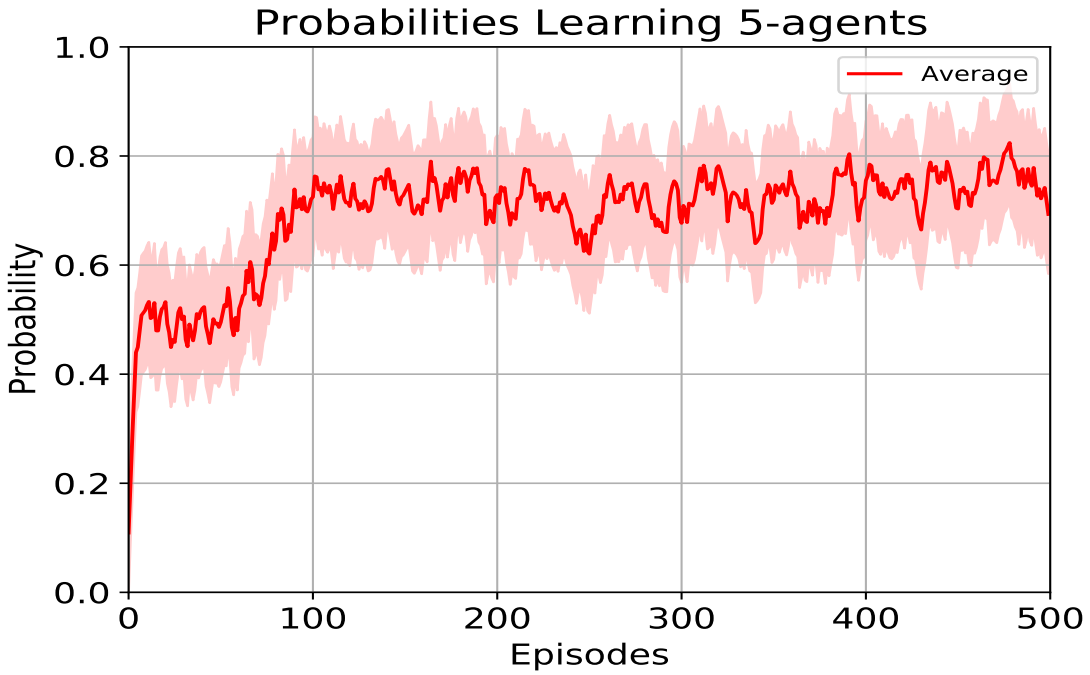


Figura 5.3: Valores promedio de la probabilidad de éxito de 5 agentes en los 500 episodios de acuerdo al método de aprendizaje. El área sombreada corresponde a la desviación estándar.

5.4. Método Introspección

Los resultados para el método de introspección están reflejados en la Fig. 5.4, éstos se componen de la media de las probabilidades de éxito entre los agentes, en donde para cada agente se obtuvo el promedio de las probabilidades entre todas las acciones calculadas con la Eq. 4.1. Se presenta un aumento del promedio de la probabilidad de éxito entre los primeros 75 episodios alcanzando un valor cercano a $\hat{P}_S \approx 0{,}5$. Luego, sigue creciendo sutilmente para finalizar con una probabilidad de éxito con un valor $\hat{P}_S \approx 0{,}62$. Por lo tanto, se puede interpretar como: el agente al episodio 500 tiene un 62 % de probabilidades de completar la tarea. Se aprecia un comportamiento logarítmico, que está respaldado por la función presentada en

la Eq. 3.2. De igual manera, el agente podría explicarle a un usuario que tiene, por ejemplo, 62 % de probabilidades de completar la pista si es que sigue derecho en el último tramo, en vez de doblar a la derecha. Lo importante en este sentido, es que el usuario pueda entender la toma de decisiones.

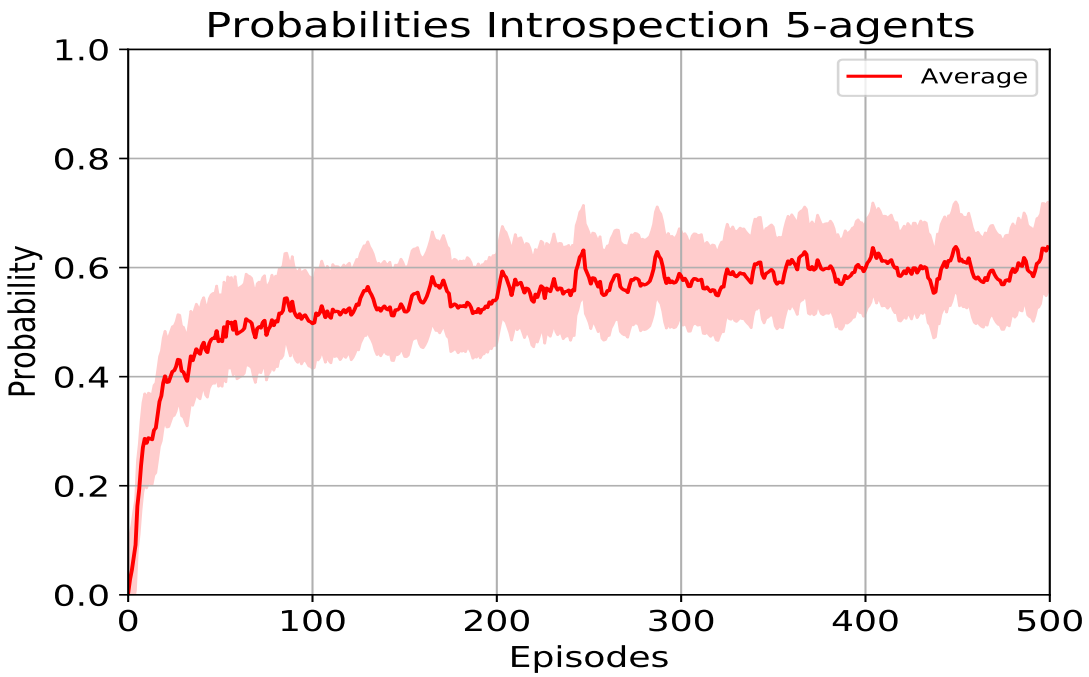


Figura 5.4: Valores promedio de la probabilidad de éxito de 5 agentes en los 500 episodios de acuerdo al método de introspección. El área sombreada corresponde a la desviación estándar.

5.5. Uso de Recursos

Se realiza un segundo experimento para medir el uso de recursos de los métodos adaptados a los entornos continuos. Este experimento cuenta con 3 agentes para cada método que se entrenan a través de 200 episodios cada uno. Con el objetivo de medir el uso de memoria RAM y uso de procesador (CPU) se utilizó la librería de Python psutil.

En el entrenamiento del agente en el entorno se procesan y almacenan imágenes, por ello la cantidad de memoria utilizada por este proceso es abundante. Debido a esto, este experimento (incluso el anterior) se realiza en periodos de 40 episodios hasta completar los 200 episodios (en el anterior hasta 500). Con estas condiciones para cada agente se registró la memoria utilizada y el uso de CPU en cada uno de

estos periodos. En el caso de la memoria se sumó los registros de todos los periodos y para el caso de la CPU se promedió.

El computador utilizado para estos experimentos cuenta con sistema operativo Windows 10 Pro, procesador Intel® Core™ i5-8500 @3GHz y dos memorias RAM HyperX® FURY 8GB DDR4 2666MHz.

Los resultados se muestran en la Fig. 5.5 en donde cada método tienen valores similares en el uso de CPU. En cuanto a la memoria utilizada por los métodos hay un leve diferencia a favor del método de introspección ocupando en promedio un $93,8\,\%$ de la memoria ocupada por el método de aprendizaje, en conjunto con una desviación estándar menor que da indicios de mayor estabilidad al uso de este recurso.

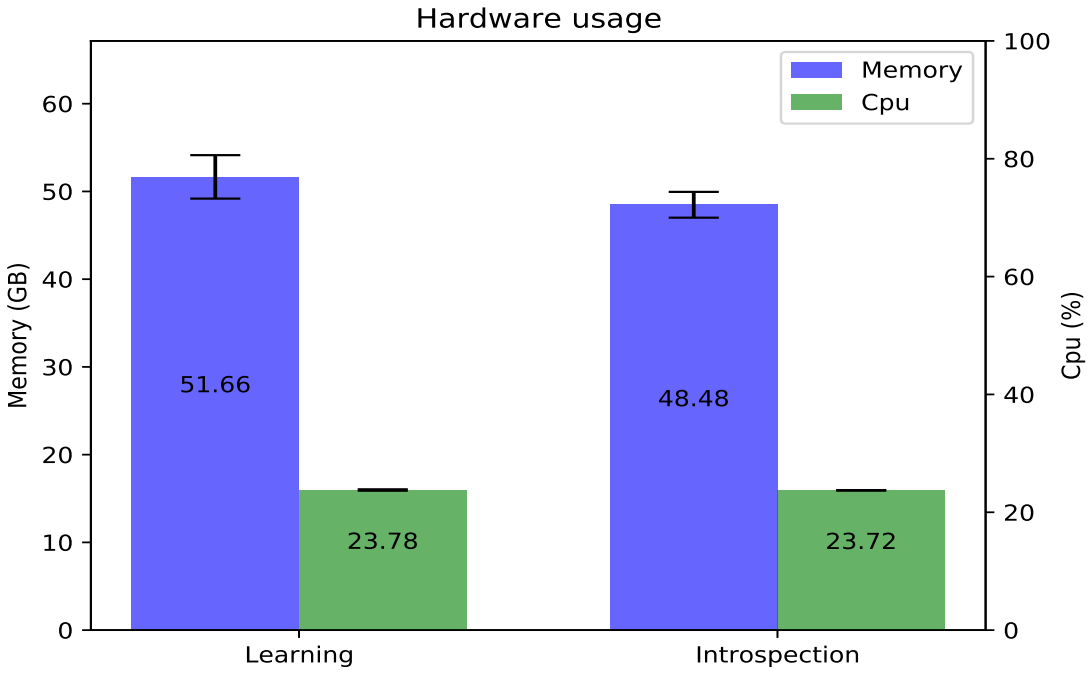


Figura 5.5: Valores de uso de recursos, considerando memoria RAM y porcentaje de uso de procesador (CPU). Se muestran los valores de los promedios de los 3 agentes durante 200 episodios para cada método, incluyendo la desviación estándar. El eje y izquierdo corresponde a la escala de gigabytes y el derecho es el porcentaje de uso de CPU. Se observa una desviación estándar menor para el método de introspección en el uso de la memoria lo cual indica más estabilidad en la utilización de este recurso. Para el caso de la CPU ambos métodos mantienen una desviación estándar no nula cercana a 0.

5.6. Análisis

En comparativa ambos métodos tuvieron resultados similares. Sin embargo, dado las características del experimento, con el método de aprendizaje se predice una mayor probabilidad de éxito que en el de introspección. Debido a la naturaleza logarítmica de la fórmula del introspección no se estima que haya aumento sustantivo de la probabilidad que puede calcular. En cambio, el método de aprendizaje como está relacionado directamente con los valores Q, es posible tener cierta mejora.

Con respecto al uso de recursos, la implementación de una red neuronal aparte para el aprendizaje de la probabilidad de éxito en el método de aprendizaje hace una leve diferencia con respecto al uso de la memoria en comparación con el método de introspección. Esto se explica por la fase de entrenamiento de la red, que requiere unas transiciones de ejemplo (64 por el tamaño del lote) para realizar el proceso, las que obtiene de la misma memoria de transiciones ocupada por el algoritmo DQN (ver Tabla 5.2). Por otra parte, el uso de CPU presenta una diferencia, al igual con la memoria, a favor del método de introspección. Sin embargo, esta diferencia es tan baja que no es determinante para catalogar a este último método como más efectivo en la utilización de la CPU. El hecho de tener 2 redes neuronales (aprendizaje) contra 1 (introspección) no afecta al uso de la CPU ya que no aumenta el orden de complejidad del algoritmo. Por la propiedad de los ordenes de complejidad, al sumar la complejidad de ambas redes neuronales se mantiene el orden de solo una (como cuando se tienen dos *for* consecutivos no anidados dentro de un código, resulta un orden de complejidad de $O(n)$).

Capítulo 6

Conclusiones

6.1. Resumen del Trabajo Expuesto

En el presente trabajo se pusieron a prueba los métodos de explicabilidad propuesto por Cruz et al. (2020) dentro de un entorno continuo. Un método basado en el aprendizaje y otro basado en la introspección. Para ello, fue necesario adaptar los métodos para que fuera posible ejecutarlos dentro del entorno continuo. Siendo el método de aprendizaje el que sufrió mayores cambios, debido al remplazo de la tabla de probabilidades (discreto) a una red neuronal artificial (continuo) para el aprendizaje de las probabilidades, también la forma de calcular la probabilidad es propuesta de forma similar a la del algoritmo DQN. En el método de introspección se propone cambiar la rectificación de la probabilidad por una normalización de acuerdo a los valores obtenidos.

Se realizaron dos experimentos, el primero fue compuesto de 5 agentes a través de 500 episodios, a la vez que se calculaba la probabilidad de éxito para ambos métodos. El segundo, con el propósito de verificar el uso de recursos computacionales, contaba con 3 agentes entrenados durante 200 episodios cada uno bajo el método de aprendizaje y otros 3 agentes para el método de introspección con la misma cantidad de episodios por agente.

6.2. Discusión

A partir de la investigación llevada a cabo, resalta el hecho de la modificación propuestas para los métodos. Por un lado, el método de introspección su base, que corresponde a la fórmula para el cálculo de la probabilidad (Eq. 3.2), se vio

levemente modificada y el único problema era la rectificación. Por lo que habla de cierta versatilidad del método con respecto al cambio de entorno. Por otro lado, en el método de aprendizaje se propone una nueva forma de calcular la probabilidad de éxito, ya no siendo aplicable lo propuesto en el trabajo de Cruz et al. (2020).

6.3. Trabajos Futuros

Según lo visto en la sección Discusión, se genera la duda ¿el método propuesto para el aprendizaje puede ser general para entornos continuos?. El método propuesto se basa en como el algoritmo de aprendizaje por refuerzo realiza el calculo de los valores Q y con ello sacar una aproximación, pero en particular en este trabajo se utilizó el algoritmo DQN, entonces ¿habrán resultados consistentes para cualquier algoritmo de aprendizaje por refuerzo?. Por ello, como futuro trabajo, se puede plantear evaluar una serie de entornos como lo mostrado por Mnih et al. (2013), también evaluar la eficiencia en la probabilidad de éxito basada en otros algoritmos de aprendizaje por refuerzo, tal como Proximal Policy Optimization (PPO) o Soft Actor-Critic.

6.4. Conclusión

Se adaptaron y evaluaron los métodos de aprendizaje e introspección en un entorno continuo. El método de aprendizaje mantuvo la idea de aprender la probabilidad de éxito a través del entrenamiento, pero con un cambio en la forma en que lo hace, alcanzando un promedio de $\mathbb{P} = 0{,}75$ al finalizar el entrenamiento. Por otra parte, el método de introspección su base se mantiene y cambia la rectificación por una normalización de los datos, dando un promedio de $\hat{P_S} = 0{,}62$. Si bien con ciertos cambios, se logró dar la capacidad de explicar el comportamiento del agente. A partir de estos avances, se puede enriquecer la interacción entre Humano-Robot de modo que en escenarios más cercanos a la realidad pueda generarse la confianza necesaria para que puedan funcionar plenamente.

A priori se estimaba que el método de aprendizaje sería más costoso en el uso de recursos, pero a partir de la propuesta para este método, se mantiene cercano el uso de memoria RAM al método de introspección. Dado que en el algoritmo 3.1 se ocupa una tabla de las mismas dimensiones que la tabla de los valores Q, en la adaptación del método de aprendizaje se reemplaza por una red neuronal artificial, lo que sustituye el uso de memoria por uso de procesamiento. Aun cuando es una

mejora para el uso de la memoria, sigue necesitando más memoria que el método de introspección.

Apéndice A

Cronograma

En la Tabla A.1 se muestra el cronograma de trabajo según la metodología del método científico.

Tabla A.1: Cronograma del proyecto

		Oct.				Nov.				Dic.				Ene.				Feb.				Mar.				Abr.				May.				Jun.				Jul.				Ago			
	Actividades \Semanas	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1. Planteamiento del problema	1.1. Establecer el problema.				x	x	x	x																																					
	1.2. Establecer herramientas para resolver el problema.							x	x	x	x																																		
Primera entrega FIP						x																																							
2. Examen y análisis de los enfoques existentes	2.1. Revisar el estado del arte.											x	x	x	x																														
	2.2. Estudiar y analizar el estado del arte.														x	x	x	x																											
Entrega Final FIP														x																															
3. Construcción del escenario experimental	3.1. Estudio de simuladores.																	x	x	x	x																								
	3.2. Definir el simulador.																				x	x	x																						
	3.3. Configurar el entorno continuo.																						x	x																					
Avance informe PT2																										x																			
3. Construcción del escenario experimental	3.4. Estudio de algoritmos de aprendizaje por refuerzo.																										x	x	x	x	x														
	3.5. Desarrollo de los algoritmos.																															x	x	x	x	x									
	3.6. Entrenamiento de los agentes.																																x	x	x	x	x	x	x	x					
Segundo Avance PT2																																							x						
4. Análisis de resultados																																					x	x	x	x	x	x			
5. Resultado del Informe	5.1. Finalización de la Tesis.																																							x	x	x		
	5.2. Publicación del código en repositorio de acceso público.																																											x
Entrega Final Informe PT2																																									x			

Bibliografía

Adadi, A. and Berrada, M. (2018). Peeking inside the black-box: a survey on explainable artificial intelligence (xai). *IEEE access*, 6:52138–52160.

Cruz, F., Dazeley, R., and Vamplew, P. (2019). Memory-based explainable reinforcement learning. In *Australasian Joint Conference on Artificial Intelligence*, pages 66–77. Springer.

Cruz, F., Dazeley, R., and Vamplew, P. (2020). Explainable robotic systems: Understanding goal-driven actions in a reinforcement learning scenario. *arXiv preprint arXiv:2006.13615*.

Das, A. and Rad, P. (2020). Opportunities and challenges in explainable artificial intelligence (xai): A survey. *arXiv preprint arXiv:2006.11371*.

Ferreyra, E., Hagras, H., Kern, M., and Owusu, G. (2019). Depicting decision-making: A type-2 fuzzy logic based explainable artificial intelligence system for goal-driven simulation in the workforce allocation domain. In *2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–6. IEEE.

Goodrich, M. A. and Schultz, A. C. (2007). Human-robot interaction: A survey, foundations and trends in human-computer interaction, vol. 1.

Gunning, D. (2017). Explainable artificial intelligence (xai)(2017). *Seen on*, 1.

Lamy, J.-B., Sekar, B., Guezennec, G., Bouaud, J., and Séroussi, B. (2019). Explainable artificial intelligence for breast cancer: A visual case-based reasoning approach. *Artificial intelligence in medicine*, 94:42–53.

Lim, B. Y., Dey, A. K., and Avrahami, D. (2009). Why and why not explanations improve the intelligibility of context-aware intelligent systems. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 2119–2128.

Madumal, P., Miller, T., Sonenberg, L., and Vetere, F. (2020). Explainable reinforcement learning through a causal lens. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 2493–2500.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M. (2013). Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*.

Moreira, I. (2019). *Análisis del aprendizaje por refuerzo profundo para un asistente doméstico en un entorno simulado*. PhD thesis, UNIVERSIDAD CENTRAL DE CHILE.

Open AI (2021). Openai gym. `https://gym.openai.com/`. última vez visto en May, 2021.

Sado, F., Loo, C. K., Liew, W. S., Kerzel, M., and Wermter, S. (2020). Explainable goal-driven agents and robots–a comprehensive review. *arXiv preprint arXiv:2004.09705*.

Sequeira, P. and Gervasio, M. (2020). Interestingness elements for explainable reinforcement learning: Understanding agents' capabilities and limitations. *Artificial Intelligence*, 288:103367.

Sutton, R. S. and Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

Van Hasselt, H., Guez, A., and Silver, D. (2016). Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30.

Wu, Andy (2021). Openai gym carracing dqn. `https://github.com/andywu0913/OpenAI-GYM-CarRacing-DQN`. última vez visto en May, 2021.