

Jairo Iván Vélez Bedoya

Integración de la Inferencia Causal en el Aprendizaje Automático: Experimentos y Aplicaciones

Director/es

Castillo Ossa, Luis Fernando
González Bedia, Manuel Andrés

<http://zaguan.unizar.es/collection/Tesis>



Universidad de Zaragoza
Servicio de Publicaciones

ISSN 2254-7606

Tesis Doctoral

INTEGRACIÓN DE LA INFERENCIA CAUSAL EN EL APRENDIZAJE AUTOMÁTICO: EXPERIMENTOS Y APLICACIONES

Autor

Jairo Iván Vélez Bedoya

Director/es

Castillo Ossa, Luis Fernando
González Bedia, Manuel Andrés

UNIVERSIDAD DE ZARAGOZA
Escuela de Doctorado

Programa de Doctorado en Ingeniería de Sistemas e Informática

2024

Tesis Doctoral

Integración de la inferencia causal en el
aprendizaje automático: experimentos y
aplicaciones

Autor

Jairo Iván Vélez Bedoya

Director/es

Manuel González Bedia
Luis Fernando Castillo Ossa

Escuela de Ingeniería y Arquitectura
2024

Integración de la Inferencia Causal en el Aprendizaje Automático: Experimentos y Aplicaciones

por

Jairo Iván Vélez Bedoya

Director

Manuel González Bedia PhD

Codirector

Luis Fernando Castillo Ossa PhD

Una tesis presentada para obtener el título de
Doctor por la Universidad de Zaragoza



Departamento de Informática e Ingeniería de Sistemas
Universidad de Zaragoza
España
Septiembre 2024

Resumen

La tesis doctoral se enfoca en la integración de la inferencia causal dentro del aprendizaje de máquina para mejorar la explicabilidad y comprensión de fenómenos complejos en diversas aplicaciones. Para alcanzar este objetivo, se ha desarrollado un marco metodológico que abarca desde una revisión exhaustiva de la literatura hasta la presentación de los resultados y conclusiones. Este marco se basa en una planificación detallada, la ejecución de experimentos y un análisis riguroso de los resultados.

La metodología empleada se estructura en torno a la definición de un marco conceptual claro, la formulación de objetivos específicos, el diseño experimental, la configuración de escenarios de prueba, la generación de datos y el análisis de resultados. Se destaca la importancia de cada paso para garantizar la calidad y fiabilidad de los resultados obtenidos.

Se destacan cinco experimentos clave que exploran diferentes aspectos de la integración de la inferencia causal en el aprendizaje de máquina. Estos experimentos abordan la estimación de iluminancia en escenarios 3D, la determinación de causas de fenómenos de sombra en imágenes, el modelado cognitivo de agentes en entornos virtuales y la mejora del aprendizaje por refuerzo mediante inferencia causal.

Los resultados obtenidos en cada experimento son analizados detalladamente, considerando tanto sus implicaciones teóricas como prácticas. Se resalta el valor de la inferencia causal en la comprensión de fenómenos complejos y en la mejora del rendimiento de algoritmos de aprendizaje de máquina en diversas aplicaciones.

Finalmente, se presentan conclusiones significativas que subrayan el potencial y relevancia de la integración de la inferencia causal en procesos de aprendizaje de máquina. Se identifican áreas prometedoras para futuras investigaciones y se destaca la contribución de la tesis al avance teórico y práctico en este campo. En conjunto, la tesis ofrece una visión integral de cómo la inferencia causal puede potenciar el aprendizaje de máquina y mejorar la explicabilidad en una variedad de aplicaciones.

Abstract

This doctoral thesis focuses on the integration of causal inference within machine learning to enhance the explainability and understanding of complex phenomena in various applications. To achieve this objective, a methodological framework has been developed that encompasses a comprehensive literature review to the presentation of results and conclusions. This framework is based on detailed planning, the execution of experiments, and rigorous analysis of the results.

The methodology employed is structured around the definition of a clear conceptual framework, the formulation of specific objectives, experimental design, configuration of test scenarios, data generation, and result analysis. The importance of each step is highlighted to ensure the quality and reliability of the results obtained.

Five key experiments are highlighted, exploring different aspects of integrating causal inference into machine learning. These experiments address illuminance estimation in 3D scenarios, the determination of causes of shadow phenomena in images, the cognitive modeling of agents in virtual environments, and the enhancement of reinforcement learning through causal inference.

The results obtained in each experiment are analyzed in detail, considering both their theoretical and practical implications. The value of causal inference in understanding complex phenomena and improving the performance of machine learning algorithms in various applications is emphasized.

Finally, significant conclusions are presented that underline the potential and relevance of integrating causal inference into machine learning processes. Promising areas for future research are identified, and the thesis's contribution to theoretical and practical advancements in this field is highlighted. Overall, the thesis offers a comprehensive view of how causal inference can enhance machine learning and improve explainability in a variety of applications.

Dedicatoria

a Ligia

En memoria de ti, mi amada abuela, cuya presencia llena mis recuerdos y cuya ausencia se siente en cada logro.

Agradecimientos

Quiero expresar mi sincero agradecimiento a la Universidad de Zaragoza y al Banco Santander de España por su generoso apoyo financiero y logístico que hizo posible la realización de mi formación doctoral y, por ende, este proyecto de investigación. Su contribución ha sido fundamental para llevar a cabo este trabajo doctoral y estoy profundamente agradecido por su confianza en mi trabajo y su compromiso con mi formación académica.

Muy especialmente y con total gratitud y amor, quiero reconocer el apoyo incondicional de mi familia durante esta aventura enriquecedora. Una aventura puede ser enriquecedora no solo a nivel académico, sino también en el ámbito personal y profesional, que me ofreció experiencias y aprendizajes valiosos que trascienden las fronteras del conocimiento formal. A Ximena y a Gabriela por todo el amor, su comprensión, apoyo y compañía permanente desde que inicié esta aventura académica y de vida. A mis padres y demás familiares, les agradezco por su apoyo y aliento constante a lo largo de los años.

Agradezco sinceramente al Dr. Manuel González Bedia por su dedicación, orientación experta y apoyo continuo a lo largo de este proyecto de investigación. Su profundo conocimiento y su compromiso fueron fundamentales para dar forma a esta tesis. Al Dr. Luis Fernando Castillo Ossa, le agradezco por su constante acompañamiento y estímulo. A ambos, gracias por la confianza y el cariño que me tienen.

Al Dr. Francisco Serón Arbeloa, le expreso mi gratitud por compartir generosamente su conocimiento, por sus valiosos consejos y por inculcarme el rigor académico que le caracteriza. Su compañía fue un consuelo en momentos de soledad y vulnerabilidad al inicio de este proceso. Al Dr. José Merseguer, director del programa de doctorado, a Asun y Araceli de la sección de Relaciones Internacionales, les expreso mi gratitud por la orientación y diligencia en asuntos cruciales tanto para el desarrollo como para la culminación de este proceso.

Finalmente, a mis amigos cercanos: Sarah, Diego, Deybid y Jeferson, les agradezco de corazón por estar a mi lado durante los momentos de dificultad y por celebrar conmigo cada logro alcanzado. Su amistad y apoyo han sido una fuente de fortaleza y alegría. Gracias, Marcelo, por tus Gin tonics, que a veces trajeron calma en medio de la tormenta y siempre hicieron que cada conversación fuera un verdadero regalo. A Maricarmen y Sergio, gracias por su hospitalidad y aprecio. A Doña Lucía y Don Jorge, gracias por su dulzura y empatía. En general, siento una profunda gratitud hacia el pueblo Zaragozano, que acogió a mi esposa, mi hija y a mí en muchos sentidos.

Índice general

I	Introducción	11
1.	Introducción	12
2.	Metodología general	17
2.1.	Definición del marco conceptual	18
2.2.	Formulación de los objetivos de la experimentación	18
2.3.	Diseño experimental	18
2.4.	Configuración del escenario de pruebas	18
2.5.	Generación de datos	19
2.6.	Resultados y conclusiones	19
2.7.	Documentación y presentación	19
3.	Marco conceptual	21
3.1.	Machine Learning: Fundamentos y aplicaciones	21
3.1.1.	Aprendizaje supervisado	22
3.1.2.	Aprendizaje no supervisado	23
3.1.3.	Aprendizaje por refuerzo	25
3.2.	Inferencia causal en Machine Learning	26
3.3.	Trabajos relacionados	28
3.3.1.	Estimación de la iluminancia	28
3.3.2.	Inferencia causal para la explicación de eventos captados en imágenes . .	29
3.3.3.	Inferencia causal para determinar el comportamiento de un agente inteligente	30
3.3.4.	Inferencia causal para el muestreo priorizado en modelos de aprendizaje por refuerzo	31
3.4.	Conclusiones	33
II	Experimentación	35
4.	Simulación y visualización del fenómeno interacción luz-materia	36
4.1.	Objetivo	36
4.2.	Método	36
4.3.	Resultados	39
4.4.	Conclusiones	42

5. Estimación de valores de iluminancia mediante Deep Learning	44
5.1. Objetivo	44
5.2. Método	44
5.3. Resultados	54
5.4. Conclusiones	60
6. Inferencia causal para determinar la causa de un fenómeno de sombra en una imagen	62
6.1. Objetivo	62
6.2. Método	62
6.3. Resultados	68
6.4. Conclusiones	70
7. Inferencia causal para el modelo de comportamiento de un agente inteligente	71
7.1. Objetivo	71
7.2. Método	71
7.3. Resultados	81
7.4. Conclusiones	85
8. Incorporación de la inferencia causal en el procedimiento de muestreo de una Deep Q-Network para resolver problemas de control clásico	86
8.1. Objetivo	86
8.2. Método	86
8.3. Resultados	93
8.4. Conclusiones	101
III Resultados, conclusiones y trabajo futuro	103
9. Resultados	104
9.1. Experimento 1: Simulación y visualización del fenómeno interacción luz-materia	104
9.2. Experimento 2: Estimación de valores de iluminancia mediante Deep Learning .	105
9.3. Experimento 3: Inferencia causal para determinar la causa de un fenómeno de sombra en una imagen	105
9.4. Experimento 4: Inferencia causal para el modelo de comportamiento de un agente inteligente	106
9.5. Experimento 5: Incorporación de la inferencia causal en el procedimiento de muestreo de una Deep Q-Network	106
9.6. Generalización de resultados	107
9.7. Análisis comparativo	108
9.8. Relación entre los Resultados de los Experimentos y los Objetivos de la Investigación	110
10. Conclusiones	112
11. Trabajo futuro	114

12. Contribuciones de la investigación	117
12.1. Contribuciones metodológicas y empíricas	117
12.1.1. Avance en la Estimación de Iluminancia mediante Aprendizaje Profundo	117
12.1.2. Fortalecimiento de la Detección de Sombras a través de la Inferencia Causal	117
12.1.3. Mejora en la Toma de Decisiones de Agentes Inteligentes mediante Inferencia Causal	117
12.1.4. La integración de la inferencia causal en el muestreo de una Deep Q-Network ha mostrado resultados prometedores relacionados con su rendimiento en problemas de control clásico	118
12.2. Contribuciones a la literatura	118
12.2.1. Otras publicaciones	120
A. Código Fuente del Software Desarrollado	132
A.1. Introducción	132
A.2. Software 1: ML-Atlas	132
A.3. Software 2: DL-Illuminance	133
A.4. Software 3: SR-Illuminance	133
A.5. Software 4: CI-Shadow	134
A.6. Software 5: CI-Based Behaviour	134
A.7. Software 6: CI-Based Sampling	135
B. Regresión simbólica para determinar las ecuaciones que gobiernan los conjuntos de datos de iluminancia	136
B.1. Introducción	136
B.2. Método	136
B.2.1. Obtención y Preprocesamiento de Datos	136
B.2.2. Desarrollo de Modelos Predictivos de Deep Learning	137
B.2.3. Implementación de la Regresión Simbólica	137
B.3. Resultados	138
B.3.1. Ecuaciones obtenidas	138
B.4. Conclusiones	140

Índice de figuras

2.1. Proceso metodológico para cada experimento.	17
4.1. Cortes definidos en un plano	37
4.2. Iluminancia horizontal (E_H) y vertical (E_V) para el cálculo de E.	37
4.3. Distribución uniforme a escala real:(a) y (b) corresponden a un corte transversal por la cara superior del escenario debajo de donde se encuentra la fuente de luz. (c) y (d) corresponden a un corte perpendicular por el centro del escenario. . . .	39
4.4. Distribución uniforme a escala real evitando la singularidad: (a) y (b) corresponden a un corte transversal por la cara superior del escenario debajo de donde se encuentra la fuente de luz. (c) y (d) corresponden a un corte perpendicular por el centro del escenario.	40
4.5. Distribución uniforme a escala logarítmica evitando singularidad: (a) y (b) corresponden a un corte transversal por la cara superior del escenario debajo de donde se encuentra la fuente de luz. (c) y (d) corresponden a un corte perpendicular por el centro del escenario.	41
4.6. Distribución no uniforme a escala logarítmica evitando singularidad: (a) y (b) corresponden a un corte transversal por la cara superior del escenario debajo de donde se encuentra la fuente de luz. (c) y (d) corresponden a un corte perpendicular por el centro del escenario.	42
5.1. Proceso de adquisición del modelo.	45
5.2. Escenarios con distintas configuraciones de iluminación.	46
5.3. Representación detallada de la distribución de la luz en un escenario con una única fuente de luz	48
5.4. Representación detallada de la distribución de la luz en un escenario con dos fuentes de luz	49
5.5. Arquitectura de las FFNNs.	51
5.6. Gráficos de aprendizaje de FFNN para diferentes configuraciones de iluminación.	51
5.7. Arquitectura de las CNN.	53
5.8. Gráficos de aprendizaje de redes neuronales convolucionales para diferentes configuraciones de iluminación.	53

5.9.	Estimación de la red Feed-Forward (FFNN). Los valores esperados para dos escenarios iluminados por una y dos fuentes de luz se muestran en las subfiguras A y D. Las predicciones de iluminancia de FFNN para ambos escenarios se encuentran en las subfiguras B y E. Las subfiguras C y F muestran las discrepancias entre los valores esperados de cada escenario y las proyecciones realizadas.	55
5.10.	Estimaciones de la red neuronal convolucional (CNN). Se muestran los valores esperados para dos escenarios iluminados por una y dos fuentes de luz en las subfiguras A y D. Las predicciones de iluminancia de la CNN para ambos escenarios se presentan en las subfiguras B y E. Las subfiguras C y F ilustran las discrepancias entre los valores esperados de cada escenario y las predicciones. . .	56
5.11.	Comparación de resultados: CNN vs. método tradicional.	57
6.1.	Posibles escenas para generación de datos.	63
6.2.	Modelo causal estructural e independencia de variables.	65
6.3.	Distribución de probabilidad condicional para cada variable del modelo.	66
6.4.	Modelos descubiertos por el algoritmo NOTEARS.	68
6.5.	Resultados de la detección de sombras.	69
7.1.	Rutas configuradas en el entorno virtual.	72
7.2.	Modelo Arquitectónico para la ejecución del CBOT.	73
7.3.	Proceso de ejecución de una partida. (a) Consola. (b) Servidor. (c) Escenario vacío. (d) Vista GBOT. (e) Vista CBOT.	74
7.4.	Ejemplo de una partida y los datos generados.(a) Ubicación inicial de los agentes. (b) Dataset generado con las variables de una partida.	79
7.5.	Modelo causal estructural.	80
7.6.	Efecto de las intervenciones.	82
7.7.	Resultados de validación.	83
7.8.	Tasa de éxito según el estado del sensor de vista.	84
7.9.	Consumo de energía según el estado del sensor de vista.	84
7.10.	Consumo de energía Vs. Identificación de los EBOT.	85
8.1.	Arquitectura de la DQN	88
8.2.	Entornos de prueba CartPole y MountainCar.	95
8.3.	Recompensa Acumulativa: Muestreo Basado en Inferencia Causal vs. Muestreo Aleatorio en Deep Q-Network	97
8.4.	Porcentaje de Exploración: Muestreo Basado en Inferencia Causal vs. Muestreo Aleatorio en Deep Q-Network	98
8.5.	Porcentaje de Explotación: Muestreo Basado en Inferencia Causal vs. Muestreo Aleatorio en Deep Q-Network	99
8.6.	Pérdida Acumulativa: Muestreo Basado en Inferencia Causal vs. Muestreo Aleatorio en Deep Q-Network	100
8.7.	Entropía de la Política aprendida: Muestreo Basado en Inferencia Causal vs. Muestreo Aleatorio en Deep Q-Network	101

Índice de tablas

4.1. Casuística de los ejercicios de visualización	38
5.1. Ajuste óptimo de parámetros para una red feed-forward en una configuración de iluminación simétrica.	50
5.2. ajuste óptimo de parámetros para una red feed-forward en una configuración de iluminación asimétrica.	50
5.3. Ajuste óptimo de parámetros para una red neuronal convolucional en una configuración de iluminación simétrica.	52
5.4. Ajuste óptimo de parámetros para una red neuronal convolucional en una configuración de iluminación asimétrica.	52
5.5. Comparación entre los modelos FFNN y CNN	56
6.1. Proporciones de cada escena en el conjunto de datos	64
6.2. Papel de las variables del modelo estructural en el proceso de inferencia causal. .	66
6.3. Inferencia causal de la intervención $P(Y \text{ — do}(B = 0))$	69
7.1. Estructura del conjunto de datos.	76
7.2. Distribucion de probabilidad condicional del modelo causal estructural.	82
7.3. Resultado de evaluación del desempeño.	83
8.1. Métricas de rendimiento de modelos con muestreo basado en inferencia causal y muestreo aleatorio.	94
9.1. Análisis Comparativo de Experimentos	109
B.1. Parametrización del regresor simbólico	138
B.2. Precisión de los modelos para cada conjunto de datos	138
B.3. Descripción de las componentes de las ecuaciones generadas por el proceso de Regresión Simbólica.	139

Parte I

Introducción

Introducción

En el ámbito del aprendizaje de máquina, la inferencia causal desempeña un papel fundamental en una amplia gama de aplicaciones. Este trabajo se propone brindar una visión abarcativa de cómo la inferencia causal puede potenciar diversas aplicaciones de aprendizaje de máquina. Estas aplicaciones fueron seleccionadas meticulosamente basadas en el criterio proporcionado por la revisión de literatura, evaluando aspectos como la popularidad, la vigencia, la aplicabilidad y la alta complejidad de cada área. Particularmente, el énfasis experimental se centró en el cálculo de iluminancia, el aprendizaje por refuerzo, los modelos cognitivos de agentes en entornos virtuales y la explicabilidad de imágenes. Estas áreas fueron elegidas debido a su relevancia y potencial para avanzar en el campo del aprendizaje de máquina.

Aunque se reconoce que existen otras técnicas que podrían verse impactadas con la inclusión de la inferencia causal, estas fueron priorizadas por su importancia y su capacidad para proporcionar una comprensión más profunda y útil en diferentes contextos de aplicación.

La aplicación de la inferencia causal en el cálculo de la iluminancia permite una comprensión más detallada de cómo diferentes factores influyen en la distribución de la luz en un entorno específico, abriendo la puerta a ajustes precisos y adaptativos en configuraciones de iluminación del mundo real. Los hallazgos subrayan el potencial del aprendizaje profundo en la estimación de la iluminancia y señalan la importancia de abordar desafíos relacionados con la variación del número de fuentes de luz, lo que presenta un área interesante para la investigación futura.

La integración de la inferencia causal en el aprendizaje por refuerzo mejora el rendimiento de los modelos en problemas de control clásicos, permitiendo soluciones más flexibles y resilientes a problemas complejos del mundo real. Estos hallazgos proporcionan una comprensión más profunda de la capacidad de los modelos para ajustarse y aplicarse a diversas circunstancias, lo que presenta oportunidades para futuras mejoras en algoritmos de aprendizaje por refuerzo mediante Deep Learning.

La aplicación de la inferencia causal en agentes inteligentes ofrece una ventaja significativa sobre metodologías alternativas, facilitando la toma de decisiones más acertadas en entornos diversos. Además, la representación visual de un modelo causal mejora la comprensión del problema y los roles de los eventos en su resolución, aunque se requiere un conjunto de datos con una estructura más intrincada para su aplicación.

Objetivos de la investigación

El objetivo general de esta investigación es explorar y analizar cómo la integración y complementación del aprendizaje de máquina y la inferencia causal pueden mejorar la resolución de problemas en los que la explicabilidad es fundamental para aportar valor a su solución. Para conseguir este propósito precisamos específicamente:

- Desarrollar un modelo de aprendizaje de máquina que, mediante redes neuronales FFNN y CNN, estime el valor de iluminancia de un escenario 3D con distintas características de iluminación.
- Desarrollar un modelo que aplique la inferencia causal para determinar la causa de un fenómeno de sombra captado en una imagen.
- Desarrollar un modelo cognitivo, basado en inferencia causal, que apoye el proceso de toma de decisiones de un agente que debe cumplir una tarea compleja en un entorno virtual.
- Desarrollar un modelo de aprendizaje de máquina que, mediante una Deep Q-Network, aprenda a solucionar un problema de control clásico usando inferencia causal para el proceso de muestreo del replay buffer.

Inferencia causal como potenciador de algunas aplicaciones de aprendizaje de máquina

Luminotecnia

Es esencial comprender que aunque la inferencia causal es crucial en muchas aplicaciones de aprendizaje de máquina, no siempre es aplicable en todas las situaciones. Un ejemplo claro de esta limitación se encuentra en el cálculo de iluminancia, donde la ecuación utilizada es simétrica, y por lo tanto, la causalidad no se aplica directamente. Sin embargo, es importante reconocer que la comprensión de la naturaleza de la inferencia causal y sus aplicaciones en otras áreas sigue siendo esencial. La interacción entre la luz y la materia es un fenómeno crucial en diversos campos, como la fotografía, la luminotecnia, el diseño de iluminación y la cinematografía [1]. La luminotecnia se centra en el control de la luz artificial, abarcando la iluminación y la iluminancia. La iluminación se refiere al brillo percibido de una superficie, mientras que la iluminancia mide la luz incidente sobre una superficie [1]. Conseguir configuraciones de iluminación adecuadas para una escena es un complejo proceso iterativo que implica ajustes en varias características, incluida la iluminancia [2].

Explicabilidad

La explicabilidad de los modelos de aprendizaje de máquina, particularmente en el ámbito de las imágenes, está adquiriendo una importancia cada vez mayor en aplicaciones del mundo real. La inferencia causal proporciona una comprensión más profunda de cómo los modelos toman decisiones, lo que facilita la interpretación de los resultados y la identificación de posibles sesgos o errores. Al considerar cualquier imagen, más allá de verla como un contenedor de objetos, entre otras cosas, es natural que un ser humano le otorgue un significado o infiera la explicación de algún evento de interés plasmado en ella, pero ¿cómo se puede llegar a tal inferencia a

través de la inteligencia artificial? La inferencia causal puede aplicarse en muchas áreas de la ciencia y la tecnología, como la economía, la epidemiología, el procesamiento de imágenes o la conducción autónoma, ámbitos en los que es crucial tomar decisiones precisas. En la actualidad, existen métodos ampliamente estudiados que, a través de la correlación, reconocen y clasifican objetos utilizando conjuntos de datos como [3] que tiene el tamaño y la información suficientes para garantizar una alta precisión en dichas tareas [4]. Sin embargo, en la última década, como señalan [5], se ha propuesto la IA explicable (XAI) para responder a la necesidad planteada por importantes contribuciones en inteligencia artificial, que han llevado a una creciente complejidad de los algoritmos y falta de transparencia de los modelos, y para avanzar en la adopción de la IA en dominios críticos. Entonces, para obtener la explicación que buscamos sobre un evento capturado en una imagen, tendríamos que considerar relaciones causales que bien pueden ser inferidas a través del conocimiento experto [6] o bien intervenir dichos conjuntos de datos a través de la experimentación como se indica en [7] teniendo en cuenta que, en lenguaje probabilístico, no tener una forma de distinguir entre dar valor a una variable y observarla, impide modelar relaciones causa-efecto [8]. Así, tomando el modelado como un paso esencial para lograr la inferencia causal, [9] discuten el papel de la inferencia causal para mejorar la interpretabilidad y robustez de los métodos de aprendizaje automático, y destacan las oportunidades en el desarrollo de modelos de aprendizaje automático con capacidad causal adaptados para el análisis de la movilidad considerando imágenes o datos secuenciales. En el caso puntual, sobre inferencia causal aplicada a imágenes, [10] proponen utilizar coeficientes neuronales de causalidad (NCCs) que se calculan aplicando redes neuronales convolucionales (CNNs) a los píxeles de una imagen, de forma que la aparición de causalidad entre variables sugiere que existe un vínculo causal entre las propias entidades del mundo real, [11] han propuesto un enfoque estadístico -entropía de transferencia- para descubrir y cuantificar la relación entre el movimiento de la cámara y el movimiento de un objeto rastreado para predecir la ubicación del objeto rastreado, [12] presentaron un modelo de gramática bayesiana (C-AOG) para las relaciones causales percibidas por el ser humano que se pueden aprender de un vídeo, y [13] utilizan el método de causalidad, complementado con técnicas de visión por ordenador y aprendizaje automático, para determinar si un vídeo se está reproduciendo hacia delante o hacia atrás observando la "flecha del tiempo." en una secuencia temporal.

Modelos cognitivos para agentes inteligentes

Los agentes inteligentes, impulsados por la inteligencia artificial y el aprendizaje automático, son cada vez más frecuentes en diversas tareas de toma de decisiones en campos como los videojuegos [14], la conducción autónoma [15] y los sistemas IoT [16], entre otros. Un agente inteligente percibe su entorno, emprende acciones autónomas para alcanzar objetivos y puede mejorar su rendimiento mediante el aprendizaje y la adquisición de conocimientos [17]. Están diseñados para aprender de los datos recopilados, adaptarse a su entorno y tomar decisiones informadas para llevar a cabo las tareas.

En el advenimiento de la cuarta revolución industrial, la falta de transparencia de los sistemas basados en la inteligencia artificial representa un obstáculo fundamental para su uso, lo que ha llevado a la aplicación de la IA explicable (XAI) [18] a estos sistemas. Por ejemplo, los agentes inteligentes convencionales se entrenan para realizar tareas sin tener en cuenta las relaciones causales inherentes que subyacen al problema que deben resolver. Este es un problema que debe abordarse.

Aprendizaje por refuerzo

En el contexto del aprendizaje por refuerzo y los modelos cognitivos de agentes en videojuegos, la inferencia causal juega un papel crucial. Al analizar las relaciones causales entre las acciones y sus consecuencias, los algoritmos de aprendizaje por refuerzo pueden tomar decisiones más informadas y mejorar su rendimiento, especialmente en entornos complejos como los videojuegos. El aprendizaje profundo por refuerzo (Deep Reinforcement Learning) se ha convertido en una poderosa disciplina dentro de los campos de la inteligencia artificial y el aprendizaje automático. Su importancia radica en su capacidad para entrenar a las máquinas para que aprendan y tomen decisiones autónomas en entornos complejos y dinámicos. Los avances en esta área han encontrado diversas aplicaciones, como por ejemplo en la robótica [19], la automatización industrial [20], los videojuegos [21] y la sanidad [22]. En consecuencia, el aprendizaje por refuerzo implica que los agentes adquieran representaciones eficientes del entorno a partir de entradas sensoriales de alta dimensión y las utilicen para generalizar experiencias pasadas a situaciones nuevas como lo afirman [23] y [24]. En este contexto, la Deep Q-Network introducida por [25] se ha convertido en un punto de referencia fundamental para gran parte de la investigación en aprendizaje profundo por refuerzo.

El enfoque de replay memory, o replay buffer, es una parte importante de la mejora del aprendizaje por refuerzo. Este método, que implica el almacenamiento y posterior utilización de experiencias previas, ha demostrado ser fundamental para aumentar la velocidad y la estabilidad del proceso de aprendizaje en múltiples situaciones y algoritmos. Por ejemplo, en el caso de los algoritmos basados en Deep Q-Network, como se analiza en [25], el uso de la replay memory ha permitido un entrenamiento más eficaz en entornos muy complejos. Del mismo modo, la replay memory ha surgido como una herramienta importante para abordar problemas en situaciones dinámicas y no estacionarias en sistemas actor-critic [26]. Además, esta estructura es fundamental para limitar la sobreestimación de los valores de las acciones en el marco del Double Deep Q-Learning [27]. Esto demuestra la relevancia y adaptabilidad de la técnica de replay memory en una amplia gama de algoritmos y aplicaciones de aprendizaje por refuerzo.

En el campo del aprendizaje por refuerzo, la priorización de experiencias ha sido un tema de investigación crucial para mejorar la eficiencia y la capacidad de aprendizaje de los algoritmos de aprendizaje por refuerzo. Uno de los trabajos pioneros en este ámbito priorizó los métodos de muestreo para estos algoritmos [28]. El principio subyacente de estos métodos es proporcionar a la experiencia una mayor probabilidad de selección cuando el error de diferencia temporal indica un mayor cambio de valor. Como se muestra en [29], una serie de experimentos han demostrado que este método funciona, mostrando que la priorización puede acelerar en gran medida el proceso de aprendizaje en entornos de este tipo de aprendizaje.

En la literatura también se han investigado métodos alternativos de ponderación de experiencias en algoritmos de aprendizaje por refuerzo. Por ejemplo, algunos estudios han examinado la posibilidad de priorizar las experiencias en función de las recompensas adquiridas en lugar de los errores de diferencia temporal, como se muestra en [30]. Estos estudios han demostrado que la elección precisa de las medidas de priorización puede influir en la eficacia de los algoritmos de aprendizaje por refuerzo para diversas tareas.

Por otra parte, uno de los retos centrales del aprendizaje por refuerzo reside en la toma de decisiones autónoma dentro de entornos complejos y dinámicos. En este contexto, el concepto de inferencia causal ha surgido como un componente prometedor para mejorar la eficacia de los algoritmos de aprendizaje por refuerzo. Según [31], la inferencia causal puede abordar el reto fundamental de la generalización en el aprendizaje por refuerzo. Esta capacidad permite a los agentes comprender las relaciones causa-efecto en su entorno, lo que es crucial para tomar decisiones informadas. Un agente debe ser capaz de discernir cuáles de sus acciones pasadas, o lecciones aprendidas, contribuyeron a los resultados observados. Al incorporar la inferencia causal a los algoritmos de aprendizaje por refuerzo, los agentes pueden generalizar más eficazmente sus conocimientos a nuevas situaciones, lo que se traduce en una toma de decisiones más precisa.

Estructura de la Tesis

Este documento se compone de tres partes:

Primera parte (Introducción): Se explica la motivación para realizar esta investigación, se formulan los objetivos, se expone la causalidad desde el punto de vista de la ciencia computacional, se explican las consideraciones metodológicas, se detalla el desarrollo de cada uno de los experimentos realizados y se presenta el marco conceptual en relación con el aprendizaje de máquina y la inferencia causal, aportando una revisión de la literatura relacionada con la aplicación de la inferencia causal y el descubrimiento causal en el ámbito computacional.

Segunda parte (Experimentación): Compuesta por cinco capítulos, cada uno de ellos destinado al desarrollo de un experimento en particular. Cada capítulo documenta el propósito, el método, los resultados con su respectiva discusión y las conclusiones obtenidas.

Tercera parte (Conclusiones): Se presentan los resultados generales de la tesis, se formulan las conclusiones de esta investigación y se proponen trabajos futuros derivados. Además, se destacan las contribuciones metodológicas, empíricas y teóricas a la literatura resultantes de este estudio.

Metodología general

Para llevar a cabo esta investigación, se implementó una metodología sólidamente fundamentada en diversas fuentes académicas relevantes en el ámbito del aprendizaje automático (ML) y disciplinas afines. Nos basamos en los enfoques metodológicos presentados en estudios previos, como los realizados por [32], [33], [34], [35], [36], entre otros. Estos estudios abordan una amplia gama de aplicaciones del ML, desde el aprendizaje por refuerzo en videojuegos hasta el análisis de imágenes médicas y la predicción de resultados deportivos, entre otros. La figura 2.1 detalla gráficamente el proceso metodológico que se adoptó para cada experimento.

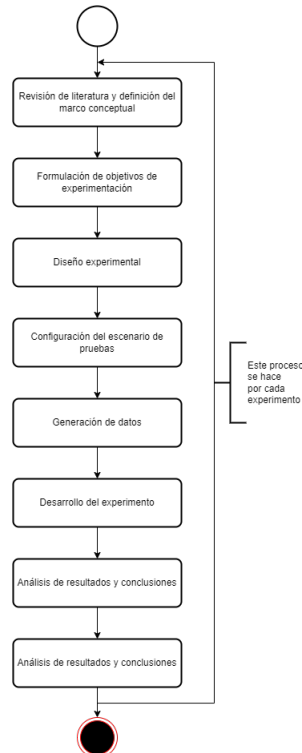


Figura 2.1. Proceso metodológico para cada experimento.

La metodología de esta investigación se estructuró en torno a la planificación, ejecución y análisis detallado de una serie de experimentos diseñados para abordar el objetivo general del

estudio. Este enfoque potenció un marco de trabajo coherente y robusto que aseguró la calidad y fiabilidad de los resultados obtenidos. A continuación, se detalla la metodología general empleada en la realización de cada uno de los experimentos:

2.1. Definición del marco conceptual

Se elaboró un marco conceptual que delineó los conceptos fundamentales y las teorías pertinentes que respaldan la presente investigación. Esto implicó una revisión exhaustiva de la literatura relacionada con el tema de estudio, con el fin de comprender el estado actual del conocimiento, identificar posibles brechas y establecer una base teórica sólida.

2.2. Formulación de los objetivos de la experimentación

La documentación detallada de cada experimento se inicia con la definición detallada de sus objetivos, los cuales actúan como directrices fundamentales para orientar tanto el diseño como la ejecución de cada fase experimental. Estos objetivos fueron concebidos de manera cuidadosa con el fin de abordar con precisión las preguntas de investigación planteadas, asegurando así la consecución del propósito general planteado en este estudio. La claridad y la precisión en la definición de estos objetivos no solo garantizan una alineación coherente con los lineamientos del proyecto, sino que también facilitan una evaluación efectiva de los resultados obtenidos en relación con los objetivos establecidos.

2.3. Diseño experimental

Se elaboraron experimentos detallados con el propósito de indagar en aspectos específicos del problema de investigación. Este proceso incluyó la identificación de variables relevantes, la selección meticulosa de métodos de recolección de datos y la planificación de los procedimientos experimentales. Específicamente, los experimentos llevados a cabo abordaron los siguientes aspectos:

- Simulación del fenómeno interacción luz-materia
- Estimación de valores de iluminancia mediante Deep Learning
- Inferencia causal para determinar la causa de un fenómeno de sombra en una imagen
- Inferencia causal para el modelo de comportamiento de un agente inteligente
- Incorporación de la inferencia causal en el procedimiento de muestreo de una Deep Q-Network para resolver problemas de control clásico

2.4. Configuración del escenario de pruebas

Se creó un entorno de pruebas acorde al propósito de cada experimento, de manera que asegurara coherencia y que hiciera que cada experimento fuera realizable y reproducible. Este

proceso implicó definir las dimensiones, características y condiciones de cada entorno experimental. Por ejemplo, se incluyeron entornos virtuales y modelos de escenarios en 3D para garantizar la fidelidad y la adecuación de los entornos de prueba a las necesidades específicas de cada experimento.

2.5. Generación de datos

Se procedió a la recopilación de datos empleando técnicas y herramientas adecuadas para cada experimento. Este proceso involucró la generación de muestras de datos siguiendo los procedimientos experimentales previamente establecidos. Específicamente, en los primeros tres experimentos (capítulos 4, 5, 6), los datos fueron generados sintéticamente. Por otro lado, en los dos últimos experimentos (capítulos 7 y 8), los datos se obtuvieron mediante un proceso de muestreo del conjunto de datos generado durante el entrenamiento de un agente.

2.6. Resultados y conclusiones

En el marco de cada experimento, se procedió a la presentación detallada y análisis de sus resultados. Su interpretación se dio en relación con el marco conceptual previamente establecido, considerando tanto sus implicaciones teóricas como prácticas. A través de este análisis, fue posible alcanzar conclusiones fundamentadas y proporcionar aportes significativos al campo de estudio abordado.

Además, se llevó a cabo una comparación entre los resultados obtenidos en los experimentos y teorías existentes, lo que contribuyó a una comprensión más profunda y diferenciada del fenómeno bajo estudio, y proporcionó una base sólida para futuras investigaciones y desarrollos en el área.

2.7. Documentación y presentación

Una parte esencial del proceso de investigación fue la documentación exhaustiva de todos los aspectos relevantes del estudio, que abarcan desde la metodología hasta los resultados y las conclusiones obtenidas. Esta documentación se realizó siguiendo estándares rigurosos para garantizar la claridad y la precisión de la información recopilada.

La metodología utilizada, junto con sus fundamentos teóricos y prácticos, se describió detalladamente, permitiendo así una comprensión completa de los procedimientos llevados a cabo en la investigación. Además, se incluyeron detalles sobre la selección de técnicas, herramientas y metodologías específicas empleadas en cada etapa del experimento.

En cuanto a los resultados obtenidos, se detallaron los hallazgos más relevantes y las tendencias observadas durante el análisis de los datos. Se incluyeron tablas, gráficos y otros elementos visuales pertinentes para facilitar la comprensión y la interpretación de los resultados.

Por último, se llevó a cabo una divulgación científica de los hallazgos de la investigación en varios journals especializados, con el objetivo de comunicar los resultados a audiencias tanto

especializadas como no especializadas. Estas publicaciones se seleccionaron cuidadosamente para asegurar que los hallazgos alcanzaran la mayor visibilidad y relevancia dentro de la comunidad académica y científica.

Marco conceptual

En esta sección, se explora la conexión entre la inferencia causal y el machine learning, dos campos cruciales en la ciencia de datos contemporánea. Se inicia revisando la importancia del machine learning en diversas aplicaciones, desde la medicina hasta las finanzas y la industria [37, 38, 39].

A continuación, se adentra en el concepto de inferencia causal y su importancia dentro del ámbito del machine learning. A diferencia del enfoque del machine learning en la predicción basada en correlaciones detectadas en los datos, la inferencia causal facilita la comprensión de las relaciones causa-efecto subyacentes. Al incorporar la inferencia causal en los modelos de machine learning, se aumenta la capacidad de interpretar los resultados y se reducen los sesgos no deseados [40, 41].

Se indaga en investigaciones pertinentes que examinan cómo la inferencia causal ha sido empleada para mejorar los modelos de machine learning, abordando aspectos como la interpretación de modelos y la toma de decisiones en condiciones de incertidumbre.

Con esta revisión de literatura, se busca proporcionar una comprensión profunda de la interacción entre la inferencia causal y el machine learning, así como su importancia en la investigación actual en ciencia de datos y sus aplicaciones prácticas.

3.1. Machine Learning: Fundamentos y aplicaciones

El campo del machine learning se basa en tres paradigmas principales: el aprendizaje supervisado, el aprendizaje no supervisado y el aprendizaje por refuerzo. De acuerdo a [42] el aprendizaje supervisado es fundamental en numerosas aplicaciones del mundo real, como reconocimiento de voz, diagnóstico médico, y detección de fraudes, ya que permite entrenar modelos para hacer predicciones precisas basadas en datos históricos. Por otra parte [43] sostiene que el aprendizaje no supervisado desempeña un papel crucial en la exploración y comprensión de conjuntos de datos complejos, facilitando tareas como la segmentación de clientes, la detección de anomalías y la reducción de la dimensionalidad. En cuanto al aprendizaje por refuerzo, [44] y [23] concuerdan en que es esencial en áreas como la robótica, los juegos y la optimización de procesos, ya que permite que los agentes aprendan a tomar decisiones secuenciales en entornos dinámicos para lograr objetivos específicos. Inscritos en estos tipos de aprendizaje, existen dos clases de modelos: los modelos de regresión, son herramientas estadísticas empleadas para entender la relación entre una variable dependiente y una o más variables independientes. Estos

modelos buscan establecer la relación entre las variables mediante una función matemática que pueda predecir el valor de la variable dependiente en función de los valores de las variables independientes [45] y, por otra parte, los modelos de clasificación, que son herramientas en el campo del aprendizaje automático y la estadística que se utilizan para predecir la clase o categoría a la que pertenecen los datos. Estos modelos se entrenan utilizando ejemplos de datos previamente etiquetados con su clase correspondiente y luego se emplean para predecir la clase de nuevos datos no etiquetados [46].

3.1.1. Aprendizaje supervisado

El aprendizaje automático supervisado implica la tarea de extraer significado de datos de entrenamiento etiquetados, los cuales consisten en un conjunto de ejemplos de entrenamiento [47]. El objetivo es que el modelo aprenda a predecir la salida correcta para nuevas entradas basadas en ejemplos previamente etiquetados.

Aplicaciones comunes del aprendizaje supervisado

El aprendizaje supervisado tiene una amplia gama de aplicaciones en la práctica en diversas industrias. Entre otros autores, [42] y [48] consideran que algunas de las aplicaciones más comunes incluyen:

- Reconocimiento de voz y procesamiento de lenguaje natural: Utilizado en sistemas de reconocimiento de voz como Siri de Apple, asistentes virtuales como Alexa de Amazon y en el procesamiento de lenguaje natural para la traducción automática y la generación de resúmenes de texto.
- Visión por computadora: Empleado en aplicaciones como reconocimiento facial, detección de objetos y segmentación de imágenes en campos que van desde la seguridad hasta la medicina.
- Diagnóstico médico: Aplicado en la detección temprana de enfermedades mediante el análisis de imágenes médicas, datos genéticos y registros de salud electrónicos.

Principales modelos del aprendizaje supervisado

1. **Regresión lineal:** Un modelo simple pero efectivo para problemas de regresión que busca establecer una relación lineal entre las características de entrada y la variable de salida.
2. **Regresión logística:** Utilizado para problemas de clasificación binaria, la regresión logística estima la probabilidad de que una instancia pertenezca a una clase particular.
3. **Random Forest:** Un algoritmo de ensemble que combina múltiples árboles de decisión para realizar predicciones más precisas. Es robusto frente al sobreajuste y capaz de manejar conjuntos de datos grandes con alta dimensionalidad.
4. **Redes Neuronales de propagación hacia adelante (FFNN):** son modelos en los que la información se mueve en una sola dirección, de la entrada a través de las capas ocultas hacia la salida. Son utilizadas para una variedad de tareas de aprendizaje automático,

incluyendo la clasificación, la regresión, el reconocimiento de patrones y la generación de secuencias, entre otras.

5. **Redes Neuronales Convolucionales (CNN):** Especialmente eficaces en problemas de visión por computadora, las CNN son capaces de aprender automáticamente características jerárquicas de las imágenes, como bordes, formas y texturas.
6. **Redes Neuronales Recurrentes (RNN):** Utilizadas en tareas de procesamiento de secuencias, como el procesamiento de lenguaje natural y la traducción automática, las RNN pueden manejar entradas de longitud variable y capturar dependencias temporales en los datos.
7. **Support Vector Machines (SVM) kernelizados:** Una extensión de las SVM que utiliza funciones kernel para mapear los datos en un espacio de mayor dimensionalidad, permitiendo la separación de clases no linealmente separables.
8. **K-Nearest Neighbors (KNN):** Un algoritmo simple pero poderoso que clasifica nuevas instancias según la mayoría de las etiquetas de las k instancias más cercanas en el espacio de características.

Ventajas y desventajas del aprendizaje supervisado

Varios autores como [49] y [50], han identificado tanto ventajas como desventajas del aprendizaje supervisado. Entre las ventajas, destacan las predicciones altamente precisas que estos modelos pueden realizar, basadas en datos de entrenamiento. Además, los resultados del modelo son interpretables, lo que facilita la comprensión de las relaciones entre las características y las salidas. En muchos casos, también es posible acceder a grandes conjuntos de datos etiquetados para el entrenamiento del modelo.

Por otro lado, se consideran algunas limitaciones. Estos modelos requieren grandes cantidades de datos etiquetados para su entrenamiento efectivo, lo que puede conducir al sobreajuste a los datos de entrenamiento y a una generalización deficiente a nuevos datos. Además, existe la dificultad inherente para capturar relaciones complejas y no lineales entre las características y las salidas.

3.1.2. Aprendizaje no supervisado

Este tipo de aprendizaje permite obtener representaciones a partir de datos sin etiquetar o experiencias que puedan ser empleadas para un aprendizaje más efectivo de tareas posteriores, incluso con cantidades modestas de datos etiquetados [51]. El objetivo es explorar la estructura oculta de los datos y generar conocimiento útil a partir de ellos.

Aplicaciones comunes del aprendizaje no supervisado

El aprendizaje no supervisado tiene una amplia gama de aplicaciones en diversas áreas. Una de las aplicaciones más comunes es el clustering, que implica agrupar automáticamente datos similares en grupos o clústeres. Este enfoque se utiliza ampliamente en la segmentación de clientes para estrategias de marketing, así como en el agrupamiento de documentos para

la organización eficiente y la búsqueda de información relevante [46] y [43]. Otra aplicación importante es la reducción de dimensionalidad, que consiste en disminuir el número de variables o características en un conjunto de datos mientras se conserva la mayor cantidad posible de información. Esta técnica se aplica en el análisis de datos de alta dimensionalidad, como imágenes o datos genómicos, así como en la visualización de datos para comprender la estructura y las relaciones entre variables [46] y [50]. Además, el aprendizaje no supervisado se utiliza en la detección de anomalías, que implica identificar instancias inusuales o atípicas en un conjunto de datos. Esta aplicación es fundamental en la detección de fraudes en transacciones financieras y en la monitorización de sistemas para la detección de fallos o intrusiones [52] y [53]. Estas aplicaciones son solo algunas de las muchas formas en que el aprendizaje no supervisado se aplica en la práctica para extraer información útil y descubrir patrones en los datos sin la necesidad de etiquetas explícitas.

Principales modelos del aprendizaje no supervisado

1. **Clustering:** Los algoritmos de clustering, como K-Means [54], DBSCAN [55] y Hierarchical Clustering [56], agrupan automáticamente datos similares en grupos o clústeres sin la necesidad de etiquetas previas. Estos algoritmos son ampliamente utilizados en la segmentación de datos y la exploración de patrones en conjuntos de datos no etiquetados [46].
2. **Reducción de dimensionalidad:** Estos modelos, como PCA (Análisis de Componentes Principales)[57] y t-SNE (t-Distributed Stochastic Neighbor Embedding) [58], reducen la dimensionalidad de un conjunto de datos conservando la mayor cantidad posible de información. Son útiles para visualizar datos de alta dimensionalidad y para encontrar representaciones más compactas de los datos.
3. **Detección de anomalías:** Los modelos de detección de anomalías identifican instancias inusuales o atípicas en un conjunto de datos [59]. Algunos algoritmos comunes incluyen el método de detección de outliers basado en la desviación estándar, el algoritmo de bosques aleatorios y los modelos de mezclas gaussianas [53].
4. **Asociación:** Estos modelos, como Apriori [60] y FP-Growth [61], identifican patrones de asociación entre diferentes variables en un conjunto de datos. Son utilizados en la minería de datos para descubrir reglas de asociación entre elementos en bases de datos transaccionales [50].

Ventajas y desventajas del aprendizaje no supervisado

El aprendizaje no supervisado ofrece varias ventajas significativas en el análisis de datos. Una de las principales ventajas es su capacidad para explorar y descubrir patrones en conjuntos de datos sin la necesidad de etiquetas previas, lo que facilita la exploración inicial de datos y la identificación de estructuras subyacentes [46]. Además, el aprendizaje no supervisado es altamente flexible y adaptable a una amplia variedad de problemas y tipos de datos, ya que no requiere etiquetas de salida específicas [62]. Esto lo hace especialmente útil en situaciones donde no se dispone de datos etiquetados o donde la naturaleza de los datos es desconocida. Otra ventaja importante es su capacidad para descubrir conocimiento oculto en los datos, lo que puede llevar a una mejor comprensión de los datos y a nuevas ideas o descubrimientos [50].

Sin embargo, el aprendizaje no supervisado también presenta desventajas. Por ejemplo, es difícil evaluar la calidad y el rendimiento de los modelos de aprendizaje no supervisado, ya que no hay una medida clara de éxito como en el aprendizaje supervisado [46]. Además, la interpretación de los resultados puede ser complicada y subjetiva, ya que los patrones descubiertos pueden carecer de un significado claro o ser difíciles de interpretar sin el contexto adecuado [62]. Existe también el riesgo de sobreajuste, donde los algoritmos pueden extraer patrones espurios o irrelevantes, especialmente en conjuntos de datos grandes y complejos [50].

3.1.3. Aprendizaje por refuerzo

El aprendizaje por refuerzo es un paradigma de aprendizaje automático donde un agente aprende a tomar decisiones secuenciales con el fin de maximizar una recompensa acumulativa. En este enfoque, el agente interactúa con un entorno, observa su estado actual, selecciona una acción y recibe una recompensa (o castigo) como resultado de esa acción. El objetivo del agente es aprender una política óptima, es decir, una estrategia que maximice la recompensa acumulativa a lo largo del tiempo [31]. El objetivo es aprender una política óptima que maximice la recompensa acumulada a lo largo del tiempo.

Aplicaciones comunes del aprendizaje por refuerzo

El aprendizaje por refuerzo se ha destacado en diversas aplicaciones, respaldado por investigaciones significativas en la literatura. En el ámbito de los juegos y estrategias, el trabajo seminal de [44] sobre el dominio del Go mediante redes neuronales profundas y búsqueda de árboles marcó un hito en la capacidad de los agentes de aprendizaje automático para enfrentar desafíos complejos. Asimismo, [23] demostraron el potencial del aprendizaje profundo en el control humano-nivel en juegos de Atari. En robótica, el aprendizaje por refuerzo ha permitido el desarrollo de agentes autónomos capaces de aprender tareas complejas, como la navegación en entornos desconocidos o la manipulación de objetos [63] y [64]. Además, en sistemas de recomendación y personalización, el aprendizaje por refuerzo se utiliza para adaptar y mejorar las recomendaciones de productos o contenido para usuarios individuales [31] y [25]. Por último, en la gestión de recursos y planificación, [65] han investigado el uso del aprendizaje por refuerzo en la administración eficiente de energía en edificios inteligentes, mientras que [66] han propuesto algoritmos de aprendizaje profundo para la asignación óptima de recursos en redes de comunicación. Estas investigaciones destacan el potencial del aprendizaje por refuerzo en una variedad de aplicaciones prácticas, desde juegos y robótica hasta sistemas de recomendación y gestión de recursos.

Principales modelos del aprendizaje por refuerzo

1. **Q-Learning:** Este algoritmo es uno de los más básicos y ampliamente utilizados en aprendizaje por refuerzo. A través de la actualización de una función de valor llamada Q-valor, el agente aprende a tomar decisiones óptimas en un entorno desconocido [67].
2. **SARSA (State-Action-Reward-State-Action):** Similar a Q-learning, SARSA es otro algoritmo fundamental en aprendizaje por refuerzo. A diferencia de Q-learning, SARSA actualiza el valor Q utilizando la acción tomada por el agente, lo que lo convierte en un método de aprendizaje on-policy [68].

3. **Deep Q-Networks (DQN):** Introducido por [23], DQN es un método que combina el aprendizaje profundo con Q-learning para aprender a jugar videojuegos de Atari directamente desde píxeles en bruto. Utiliza una red neuronal convolucional profunda para aproximar la función Q.
4. **Policy Gradient Methods:** Estos métodos, como REINFORCE [69] y TRPO (Trust Region Policy Optimization) [70], aprenden directamente la política óptima de un agente mediante el cálculo de gradientes de rendimiento con respecto a los parámetros de la política.
5. **Actor-Critic Methods:** Estos métodos combinan elementos de los métodos de gradiente de políticas y los métodos de función de valor. Un ejemplo es A3C (Asynchronous Advantage Actor-Critic) [71], que utiliza una red neuronal profunda para aproximar tanto la política del agente como la función de valor.

Ventajas y desventajas del aprendizaje por refuerzo

El aprendizaje por refuerzo ofrece una serie de ventajas significativas en el ámbito del aprendizaje automático. Una de las principales ventajas es su capacidad para aprender de la interacción directa con el entorno, lo que lo hace adecuado para problemas donde los datos son escasos o costosos de obtener [31]. Además, el aprendizaje por refuerzo puede generalizarse a una amplia gama de problemas, desde juegos y robótica hasta gestión de recursos y planificación [72]. Otra ventaja importante es su capacidad para aprender políticas óptimas de comportamiento a través de la maximización de una señal de recompensa, lo que lo hace adecuado para problemas de toma de decisiones secuenciales [31]. Sin embargo, el aprendizaje por refuerzo también tiene algunas limitaciones. Por ejemplo, puede requerir una cantidad significativa de tiempo y recursos computacionales para entrenar agentes que puedan desempeñarse bien en entornos complejos [72]. Además, la estimación de las señales de recompensa puede ser difícil en algunos problemas del mundo real, lo que puede llevar a un aprendizaje subóptimo o a comportamientos inesperados del agente [31].

3.2. Inferencia causal en Machine Learning

Aunque las estructuras causales están implícitas en los modelos de probabilidad, abarcan información adicional que está ausente en estos últimos. El proceso de obtener conclusiones a partir de un modelo causal se denomina razonamiento causal, que es análogo a cómo la teoría de la probabilidad nos permite razonar sobre los resultados de experimentos aleatorios. Por otro lado, el razonamiento causal es más efectivo que el razonamiento probabilístico debido a que los modelos causales contienen más información que los modelos probabilísticos. Esto se debe a que el razonamiento causal nos permite examinar el impacto de intervenciones o cambios en la distribución [41]. La inferencia causal desempeña un papel crucial en el fortalecimiento de los modelos de machine learning al ofrecer un sólido marco teórico para comprender las relaciones subyacentes de causa y efecto en los datos. Según [40], estos métodos pueden ayudar a identificar variables confusoras y mitigar sesgos en los modelos de machine learning, mejorando así la precisión y la capacidad de generalización de los resultados. Además, la integración de la inferencia causal en estos modelos puede potenciar la interpretación de los resultados al proporcionar una

comprensión más profunda de las relaciones entre las características y las salidas, como sugiere [10].

Aplicaciones de la Inferencia Causal en Machine Learning

La inferencia causal ha emergido como una herramienta poderosa en el campo del machine learning, ofreciendo una forma de comprender y manipular las relaciones causa-efecto en los datos. En la interpretación de modelos, la inferencia causal puede ayudar a identificar relaciones causales subyacentes que no son evidentes mediante técnicas puramente correlativas. Por ejemplo, en un estudio reciente, [73] utilizaron métodos de inferencia causal para desentrañar las relaciones entre las características de un modelo de machine learning y sus predicciones, proporcionando una comprensión más profunda de cómo y por qué el modelo toma ciertas decisiones. Este enfoque no solo mejora la interpretabilidad del modelo, sino que también puede ayudar a identificar posibles sesgos o efectos indeseados.

En cuanto a la generación de datos sintéticos, la inferencia causal puede utilizarse para simular datos realistas y controlados que reflejen las relaciones causales en el mundo real. Por ejemplo, [74] integración de la inferencia causal y el machine learning se evidencia en enfoques innovadores como el uso de generative adversarial networks (GANs) para generar datos sintéticos justos y libres de sesgos. Este enfoque incorpora explícitamente el modelo causal estructural en el proceso de generación de datos, lo que permite reconstruir cada variable condicionada a sus padres causales. Al hacerlo, no solo se asegura la reducción del sesgo en la generación de datos, sino que también se demuestra cómo la comprensión de las relaciones causa-efecto subyacentes puede mejorar significativamente la calidad de los modelos de machine learning. Esta capacidad para generar datos sintéticos basados en relaciones causales subyacentes puede ser invaluable en situaciones donde los datos reales son escasos o costosos de obtener.

Además, en la toma de decisiones bajo incertidumbre, la inferencia causal puede proporcionar un marco teórico sólido para evaluar y mitigar riesgos potenciales. Por ejemplo, en un contexto de salud pública, [75] utilizaron métodos de inferencia causal para evaluar el impacto causal de diferentes intervenciones en la propagación de una enfermedad infecciosa, lo que ayudó a informar decisiones de política pública. Al considerar las relaciones causales subyacentes, los investigadores pueden tomar decisiones más informadas y efectivas en entornos complejos y dinámicos.

Técnicas y enfoques para aplicar la inferencia causal

1. **Modelos de Ecuaciones Estructurales:** Estos modelos representan relaciones causales entre variables observadas y no observadas utilizando ecuaciones simultáneas. Se utilizan principalmente en estudios longitudinales y de panel.
2. **Métodos de Propensity Score Matching:** Estos métodos se utilizan para reducir el sesgo de selección en estudios observacionales mediante la creación de grupos comparables basados en la probabilidad de pertenencia a un tratamiento.
3. **Análisis de Instrumentos:** Esta técnica se utiliza para estimar el efecto causal de una variable de interés en presencia de endogeneidad utilizando variables instrumentales que están relacionadas con la variable de tratamiento pero no con el resultado.

4. **Estudio de Regresión Discontinua:** Se utiliza cuando la asignación a un tratamiento está determinada por una regla de asignación basada en un umbral, lo que permite estimar el efecto causal del tratamiento cerca de este umbral.
5. **Diseños Experimentales:** Los diseños experimentales, como los ensayos clínicos aleatorizados en medicina, son considerados el mejor método para establecer relaciones causales, ya que permiten controlar los sesgos mediante la asignación aleatoria de tratamientos.
6. **Modelos de Grafos Causales:** Estos modelos representan las relaciones causales entre variables como un grafo dirigido acíclico, lo que permite visualizar y analizar las relaciones de causa y efecto de manera explícita.

Ventajas y desventajas de la inferencia causal aplicada al machine learning

La aplicación de la inferencia causal en el campo del machine learning presenta una serie de ventajas y desventajas que han sido ampliamente discutidas en la literatura. En cuanto a las ventajas, [40] enuncia que la inferencia causal proporciona un marco teórico sólido para comprender y modelar las relaciones de causa y efecto entre las variables en los datos. Esta comprensión profunda de las relaciones causales subyacentes permite una mejor interpretación de los modelos de machine learning y facilita la toma de decisiones informadas en entornos complejos. Además, la inferencia causal puede ayudar a mitigar el sesgo y mejorar la generalización de los modelos al controlar variables confusoras y eliminar la contaminación causada por relaciones espurias. Sin embargo, también existen desventajas en la aplicación de la inferencia causal en el machine learning. Por ejemplo, [75] sostienen que la identificación y especificación adecuada de las relaciones causales puede ser un desafío en muchos casos, especialmente cuando las relaciones son complejas o no lineales. Además, la inferencia causal puede requerir datos adicionales o información sobre el mecanismo causal subyacente, lo que puede ser difícil o costoso de obtener en la práctica. A pesar de estas limitaciones, como se verá en la siguiente sección, la integración de la inferencia causal en el machine learning sigue siendo un área de investigación activa y prometedora que tiene el potencial de mejorar significativamente la calidad y la ética de los modelos de aprendizaje automático.

3.3. Trabajos relacionados

3.3.1. Estimación de la iluminancia

Estimar la iluminancia representa un desafío significativo en este campo, ya que es una medida fotométrica crucial para evaluar la calidad de la iluminación [76]. Hasta donde se sabe, investigaciones anteriores no han explorado la aplicación del aprendizaje profundo en la estimación de la iluminancia y su integración en el proceso de configuración de la iluminación. Hay diversos enfoques para este problema. Por ejemplo, en [77] los autores proponen una red neuronal convolucional que estima la iluminación en una escena mediante aprendizaje supervisado, considerando la limitación de requerir una gran cantidad de datos de entrenamiento. Gardner et al. [78] presentan un método automático que utiliza una red neuronal convolucional para inferir la iluminación a partir de una única fotografía y un campo de visión limitado en una escena interior. En [79] los autores abordan la estimación de la luz que llega desde todas las direcciones a un punto 3D en una imagen descomponiendo la predicción de la iluminación en estimación

de geometría, completado de escena y estimación de rango dinámico de baja a alta. Zhang et al. [80] proponen un método de aprendizaje basado en el canal de luminancia de la región del cielo y una red neuronal convolucional con mejoras en el rendimiento. En cuanto a la luminancia aplicada a la industria de los videojuegos, en [81] el autor desarrolló un sistema de iluminación dinámica en tiempo real llamado ELE (Expressive Lighting Engine) para establecer una configuración de iluminación óptima. Otros estudios, como [82, 83, 84, 85, 86, 87, 88, 89, 90], también han explorado diferentes técnicas para la estimación de la iluminancia utilizando aprendizaje profundo o métodos relacionados.

3.3.2. Inferencia causal para la explicación de eventos captados en imágenes

Al considerar cualquier imagen, más allá de verla simplemente como un conjunto de objetos, es natural para un ser humano atribuirle significado o inferir la explicación de algún evento de interés capturado en ella. Sin embargo, ¿cómo puede alcanzar tal inferencia la inteligencia artificial? La inferencia causal puede aplicarse en diversas áreas de la ciencia y la tecnología, como la economía, la epidemiología, el procesamiento de imágenes y la conducción autónoma, donde es crucial tomar decisiones precisas.

Actualmente, existen métodos ampliamente estudiados que, a través de la correlación, permiten reconocer y clasificar objetos utilizando conjuntos de datos robustos, como se presenta en [3], que proporcionan información suficiente para garantizar una alta precisión en estas tareas [4]. Sin embargo, en la última década, como señala [5], la inteligencia artificial explicativa (XAI) ha surgido como respuesta a la complejidad creciente de los algoritmos y la falta de transparencia de los modelos, lo que ha llevado a la necesidad de hacer que los sistemas de IA sean más interpretables y comprensibles, especialmente en dominios críticos.

Entonces, ¿cómo pueden obtenerse explicaciones sobre eventos capturados en imágenes mediante inteligencia artificial? Para lograrlo, se deben considerar relaciones causales que pueden inferirse a través del conocimiento experto [6] o mediante intervenciones en los conjuntos de datos, como se describe en [7]. Sin embargo, es importante tener en cuenta que, en el lenguaje probabilístico, la incapacidad para distinguir entre dar valor a una variable y observarla puede dificultar la modelización de las relaciones causa-efecto [8].

Por lo tanto, el modelado juega un papel crucial en la inferencia causal. [91] discute cómo la inferencia causal puede mejorar la interpretabilidad y la robustez de los métodos de aprendizaje automático, destacando oportunidades en el desarrollo de modelos de aprendizaje automático con capacidad causal adaptada para el análisis de la movilidad a partir de imágenes o datos secuenciales.

En el ámbito específico de la inferencia causal aplicada a imágenes, existen diversas aproximaciones. Por ejemplo, [10] proponen el uso de coeficientes de causalidad neural (NCC) calculados mediante redes neuronales convolucionales (CNN) aplicadas a los píxeles de una imagen. Otros, como [11], han desarrollado un enfoque estadístico basado en la transferencia de entropía para descubrir y cuantificar la relación entre el movimiento de la cámara y el de un objeto rastreado, mientras que [12] han presentado un modelo gramatical bayesiano (C-AOG) para relaciones

causales percibidas por humanos en videos. Asimismo, [13] utilizan el método de causalidad, junto con técnicas de visión por computadora y aprendizaje automático, para determinar si un video se reproduce hacia adelante o hacia atrás mediante la observación de la flecha del tiempo en una secuencia temporal. Tomando en consideración las causas subyacentes de la formación de sombras, la inferencia causal puede brindar predicciones más precisas y mejorar el realismo general de los entornos virtuales. Por lo cual, dada la relevancia de este tema y la necesidad de experimentación en casos específicos que podrían ser contribuyentes a campos en evolución como los gráficos 3D, donde la detección de sombras representa un área en la que la inferencia causal puede aplicarse para mejorar la precisión y eficiencia en este proceso, en contraposición a técnicas tradicionales como el trazado de rayos, que resulta computacionalmente costoso en términos de manejo de escenas complejas con múltiples objetos [92]. Este desafío es reconocido igualmente en soluciones innovadoras como la propuesta por [93], quien calcula derivadas de funciones escalares en una imagen renderizada con respecto a parámetros arbitrarios de la escena, como la ubicación de la cámara, la geometría de la escena, los materiales y los parámetros de iluminación, utilizando un algoritmo de muestreo de bordes. Nuestro estudio proporciona un punto de referencia para abordar un aspecto de este problema mediante el uso de la inferencia causal para detectar y deducir la causa de una sombra proyectada sobre la superficie de una escena 3D.

3.3.3. Inferencia causal para determinar el comportamiento de un agente inteligente

Los agentes inteligentes, impulsados por la inteligencia artificial y el aprendizaje automático, han adquirido una creciente presencia en diversas tareas de toma de decisiones en campos como los videojuegos [14], la conducción autónoma [15], y los sistemas IoT [16], entre otros. Un agente inteligente percibe su entorno, toma acciones autónomas para alcanzar objetivos y puede mejorar su desempeño mediante el aprendizaje y la adquisición de conocimientos [17]. Están diseñados para aprender de los datos recopilados, adaptarse a su entorno y tomar decisiones informadas para realizar tareas.

En el contexto de la cuarta revolución industrial, la falta de transparencia en los sistemas basados en inteligencia artificial representa un obstáculo crucial para su adopción, lo que ha llevado a la aplicación de la inteligencia artificial explicativa (XAI) [18] a estos sistemas. Por ejemplo, los agentes inteligentes convencionales están entrenados para realizar tareas sin considerar las relaciones causales subyacentes que fundamentan el problema que deben resolver. Este es un problema que debe abordarse. Sin embargo, como señaló [94], incorporar la inferencia causal en los agentes inteligentes es un desafío debido a las numerosas variables ocultas dentro del modelo. No obstante, ha habido un reciente aumento del interés en el desarrollo de algoritmos que generan comportamiento de agente interpretable con respecto a objetivos, planes o recompensas, como se discute en [95]. Esto ha allanado el camino para investigaciones, como las de [96], [97], y [98], que demuestran la viabilidad de integrar el razonamiento probabilístico y los mapas causales en la lógica de los agentes inteligentes. Al adoptar el razonamiento causal, los agentes inteligentes pueden ir más allá de simplemente identificar correlaciones y discernir las causas fundamentales de los eventos, facilitando procesos de toma de decisiones más sólidos, fiables e interpretables. Por ejemplo, [99] han utilizado el “aprendizaje meta-reforzado” para generar un agente capaz de ejecutar tareas mediante inferencias causales, incluso sin conocimiento explícito de la causalidad.

De manera similar, en relación con el diseño de un sistema de energía de red inteligente, [100] propone un modelo de comunicación de inferencia causal (CICM) para garantizar un funcionamiento eficiente del sistema y reducir el ancho de banda de comunicación. La eficacia de su algoritmo se demostró en experimentos utilizando tareas de navegación en el mundo virtual de StarCraft II, un videojuego. [101] ha presentado un agente de inferencia dinámica que utiliza representaciones numéricas en lugar de representaciones simbólicas para modelar, inferir y tomar decisiones.

Además, [102] han sugerido un método de diseño y una arquitectura de agente que se basan en el marco de Creencias, Deseos e Intenciones (BDI) que [103] describieron para crear agentes inteligentes. Resaltando la dificultad de modelar con precisión la estructura causal de un agente, [104] ha enfatizado este desafío utilizando la Competición Angry Birds AI como ejemplo, donde los agentes deben analizar niveles y predecir las consecuencias físicas de sus acciones para obtener altas puntuaciones, como se describe en [105]. En respuesta a esta competencia, [106] han desarrollado una arquitectura de agente que emplea inferencia bayesiana para mejorar las habilidades de toma de decisiones.

Aunque algunos estudios han abordado el desarrollo de agentes inteligentes que incorporan inferencia causal en su aprendizaje, se necesita más experimentación en este tema, lo que indica la necesidad de más estudios empíricos para obtener una comprensión más profunda de cómo la inferencia causal puede mejorar las capacidades de toma de decisiones de los agentes inteligentes.

Esta investigación examinará a fondo la intersección entre agentes inteligentes e inferencia causal para explorar cómo la incorporación del razonamiento causal puede mejorar significativamente las habilidades de toma de decisiones y la ejecución de tareas, proporcionando a estos agentes una ventaja distintiva sobre otros sistemas. El propósito principal de esta investigación es contribuir al creciente cuerpo de conocimiento en los campos de sistemas de agentes inteligentes e inferencia causal, arrojando luz sobre el potencial prometedor de fusionar estas áreas para crear sistemas de toma de decisiones más informados y transparentes.

3.3.4. Inferencia causal para el muestreo priorizado en modelos de aprendizaje por refuerzo

El aprendizaje profundo por refuerzo (DRL) se ha convertido en una disciplina poderosa dentro de los campos de la inteligencia artificial y el aprendizaje automático. Su prominencia radica en su capacidad para entrenar máquinas para aprender y tomar decisiones autónomas en entornos complejos y dinámicos. Los avances en DRL han encontrado aplicaciones en diversos campos, incluyendo la robótica [19], la automatización industrial [20], los videojuegos [21], y la atención médica [22]. En consecuencia, el aprendizaje por refuerzo implica que los agentes adquieran representaciones eficientes del entorno a partir de entradas sensoriales de alta dimensión y las utilicen para generalizar experiencias pasadas a nuevas situaciones [23],[24]. En este contexto, la Deep Q-Network introducida por [25] se ha convertido en un punto de referencia fundamental para gran parte de la investigación en aprendizaje profundo por refuerzo.

El enfoque de memoria de repetición (Replay Buffer) es una parte importante para mejorar

el aprendizaje por refuerzo. Este método, que implica almacenar y luego utilizar experiencias previas, se ha demostrado que es fundamental para aumentar la velocidad y la estabilidad del proceso de aprendizaje en múltiples situaciones y algoritmos. Por ejemplo, en el caso de algoritmos basados en Deep Q-Network, como se exploró en [25], el uso de la memoria de repetición ha permitido un entrenamiento más efectivo en entornos altamente complejos. De manera similar, la memoria de repetición ha surgido como una herramienta significativa para abordar problemas en situaciones dinámicas y no estacionarias en sistemas actor-critic [26]. Además, la memoria de repetición es crítica para limitar la sobreestimación de los valores de acción en el contexto del doble Q-Learning [27]. Esto demuestra la relevancia y la adaptabilidad de la técnica de memoria de repetición en una amplia gama de algoritmos y aplicaciones de aprendizaje por refuerzo.

En el campo del aprendizaje por refuerzo (RL), la priorización de la experiencia ha sido un tema de investigación crucial destinado a mejorar la eficiencia y la capacidad de aprendizaje de los algoritmos de RL. Uno de los trabajos pioneros en este ámbito priorizó los métodos de muestreo para algoritmos de RL [28]. El principio subyacente de estos métodos es proporcionar experiencia con una mayor probabilidad de selección cuando el error de diferencia temporal (TD) indica un mayor cambio en el valor. Como se muestra en [29], una serie de experimentos han demostrado que este método funciona, mostrando que la priorización puede acelerar el proceso de aprendizaje en entornos de RL en gran medida.

También se han investigado métodos alternativos de ponderación de experiencias en algoritmos de RL en la literatura. Por ejemplo, ciertos estudios han examinado la posibilidad de priorizar experiencias según las recompensas adquiridas en lugar de los errores TD, como se demuestra en [30]. Estos estudios han demostrado que la elección meticulosa de medidas de priorización puede influir en la eficacia de los algoritmos de aprendizaje por refuerzo para diversas tareas.

Por otro lado, uno de los desafíos centrales en el aprendizaje por refuerzo radica en la toma de decisiones autónoma dentro de entornos complejos y dinámicos. En este contexto, el concepto de inferencia causal ha surgido como un componente esencial para mejorar la eficiencia y efectividad de los algoritmos de RL. Según [31], la inferencia causal puede abordar el desafío fundamental de la generalización en RL. Esta capacidad permite que los agentes de RL comprendan las relaciones de causa y efecto en su entorno, lo cual es crucial para tomar decisiones informadas. Un agente debe ser capaz de discernir qué acciones pasadas contribuyeron a los resultados observados. Al incorporar la inferencia causal en los algoritmos de RL, los agentes pueden generalizar más efectivamente su conocimiento a nuevas situaciones, lo que resulta en una toma de decisiones más precisa. En [107], se propone un algoritmo basado en DQN arraigado en la inferencia causal, denominado Causal Inference Q-Network (CIQ)". La evaluación del rendimiento de CIQ en diversos entornos de referencia de Redes Q Profundas con diferentes tipos de interferencias como etiquetas auxiliares demuestra que el método propuesto podría lograr un mejor rendimiento y una mayor resistencia contra interferencias observacionales. Además, [108] profundiza en cómo la inferencia causal puede aprovecharse para entender el comportamiento de sistemas de aprendizaje altamente complejos que interactúan con sus entornos. Además, enfatiza cómo las predicciones basadas en la inferencia causal pueden ser beneficiosas tanto para los humanos como para los algoritmos al seleccionar cambios que podrían mejorar potencialmente el rendimiento del sistema. Este enfoque proporciona un marco general para aplicar el razonamien-

to contrafáctico y la inferencia causal a sistemas de aprendizaje, allanando el camino para un enfoque más sólido y preciso para optimizar sistemas complejos en el contexto del aprendizaje por refuerzo.

Según [109], los autores exploraron el uso de tareas auxiliares no supervisadas en el aprendizaje por refuerzo. Los investigadores crearon un agente de aprendizaje por refuerzo que maximiza múltiples funciones de pseudo-recompensa y dirige la recompensa acumulativa. La representación compartida de las tareas auxiliares evoluciona sin recompensas externas, como el aprendizaje no supervisado. También introdujeron una nueva forma de centrarse en las recompensas externas, lo que ayuda a los agentes a comprender rápidamente los puntos clave de la tarea. El agente utilizado superó al anterior mejor rendimiento en Atari en un 880 % en comparación con el rendimiento humano experto. Además, superó a los humanos expertos en tareas de laberinto en primera persona y tridimensionales difíciles, con un aumento promedio de velocidad de aprendizaje de $10\times$ y un rendimiento promedio del 87 %. El estudio de [110] aborda el desafío de entrenar agentes generalistas para una adaptación eficiente a nuevas tareas en el aprendizaje profundo por refuerzo (DRL). Además investigaron el pre-entrenamiento no supervisado con recompensas intrínsecas autosupervisadas para la adaptación. El Banco de Pruebas de Aprendizaje por Refuerzo No Supervisado (URLB) facilita comparaciones justas y un desarrollo posterior. URLB tiene pre-entrenamiento sin recompensas y adaptación de tareas basadas en recompensas extrínsecas. Prueban sus habilidades con doce tareas de control continuo en tres dominios utilizando la Suite de Control DeepMind. Ofrecen versiones de código abierto de ocho de los mejores métodos de RL no supervisados. Aunque se han logrado avances, las líneas de base no resuelven URLB, lo que resalta la necesidad de más investigación.

A pesar de que ha habido mejoras en cómo el aprendizaje por refuerzo prioriza la experiencia, todavía existe una brecha en la literatura sobre cómo agregar inferencia causal al proceso de muestreo priorizado de una Deep Q-Network. Esta investigación aborda esta brecha explorando cómo la inferencia causal puede complementar y mejorar las técnicas de priorización existentes en el contexto del aprendizaje profundo por refuerzo.

3.4. Conclusiones

En esta sección, se exploraron diversos aspectos relacionados con la aplicación de la inferencia causal en el campo del machine learning. Se discutieron diferentes enfoques, técnicas y aplicaciones de la inferencia causal, así como su relevancia para mejorar la interpretabilidad, la precisión y la capacidad de generalización de los modelos de machine learning.

La inferencia causal desempeña un papel crucial en el fortalecimiento de los modelos de machine learning al proporcionar un marco teórico sólido para comprender las relaciones subyacentes de causa y efecto en los datos. Se destacó que los modelos de machine learning que integran la inferencia causal pueden identificar variables confusoras, mitigar sesgos y mejorar la interpretación de los resultados. Además, se resaltó la importancia de la inferencia causal en la generación de datos sintéticos realistas y controlados, así como en la toma de decisiones bajo incertidumbre en contextos como la salud pública.

La integración de la inferencia causal en el machine learning sigue siendo un área de investigación activa y prometedora. Se sugiere que futuras investigaciones se centren en desarrollar métodos y técnicas más avanzados para la inferencia causal en machine learning, especialmente en entornos donde los datos son escasos o costosos de obtener. Además, se anima a explorar aplicaciones específicas de la inferencia causal en áreas como la interpretación de modelos de deep learning, el diseño de experimentos y la optimización de políticas en sistemas de toma de decisiones autónomas.

En síntesis, la inferencia causal tiene el potencial de impulsar el avance del machine learning al proporcionar una comprensión más profunda y sólida de las relaciones causa-efecto en los datos, lo que puede conducir a modelos más robustos, interpretables y éticos. Su aplicación adecuada y continuada en el ámbito del machine learning promete abrir nuevas oportunidades y desafíos emocionantes en la investigación y la práctica.

Parte II

Experimentación

Simulación y visualización del fenómeno interacción luz-materia

4.1. Objetivo

El objetivo de este experimento fue simular la distribución del valor de iluminancia (E), una magnitud asociada a un fenómeno físico de interacción luz-materia, en un dominio espacial 3D vacío con distintas configuraciones de iluminación, para generar los conjuntos de datos utilizados en los experimentos posteriores y establecer una línea base para la visualización de resultados.

4.2. Método

Diseño y configuración del escenario de pruebas

La forma y tamaño definidos para el escenario fue un cubo de $11m$ por lado, este ocupa un volumen de $1331m^3$; el cual se consideró como un volumen razonable para la experimentación. Para definir puntos de interés dentro del escenario, este se dividió en dos mallas regulares de $11 \times 11 \times 11$ y $121 \times 121 \times 121$ para obtener un cálculo más denso de la iluminancia por m^3 del escenario. Cada punto de las mallas se caracterizó por su posición (X, Y, Z) en el espacio tridimensional y el valor de su iluminancia (E).

La iluminación del escenario se configuró posicionando fuentes lumínicas en lugares específicos del espacio. Cada fuente lumínica estaba determinada por su posición (X, Y, Z) en el espacio tridimensional y su flujo luminoso en lux. Con el fin de inspeccionar la variación de iluminancia dentro del escenario, definimos nueve cortes, como lo ilustra la figura 4.1: uno por cada cara del cubo que representa al escenario y tres que cortan por la mitad cada eje coordenado.

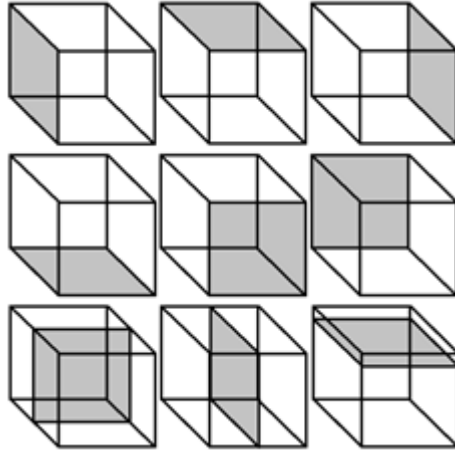


Figura 4.1. Cortes definidos en un plano

Cálculo de la Iluminancia (E)

La iluminancia (E) es una unidad de medida expresada como $E = \frac{\phi}{S}$, lo que corresponde al flujo luminoso por unidad de área [2]. Sin embargo, dadas las características del escenario definido para la experimentación, se estimó necesario tener en cuenta el ángulo que forma la dirección de la luz con la superficie sobre la cual incide. La figura 4.2 muestra las dos componentes -Iluminancia horizontal (E_H) e Iluminancia vertical (E_V)- que incorporó el cálculo de la iluminancia en este experimento aplicando el razonamiento que se muestra en las ecuaciones 4.1, 4.2 y 4.3, siendo (I) la intensidad de cada fuente lumínica, (d) la distancia entre el punto y las fuentes lumínicas, (h) la altura del punto, (\overline{hp}) el segmento de recta entre el punto de corte de (h) con el plano horizontal y el punto (P) y (α) el ángulo.

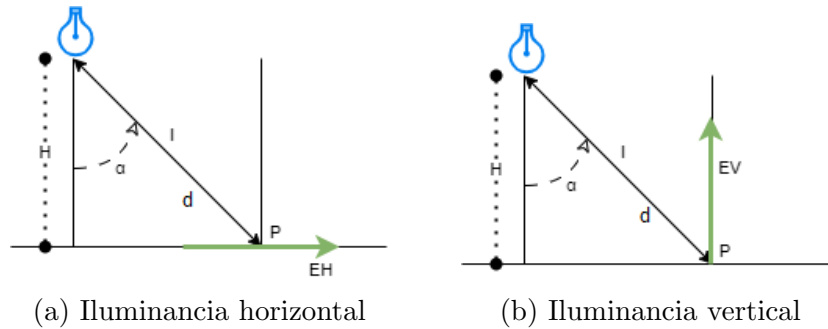


Figura 4.2. Iluminancia horizontal (E_H) y vertical (E_V) para el cálculo de E.

$$\begin{aligned}
 E_H &= \frac{I * \cos \alpha}{d^2} \text{ como } \cos \alpha = \frac{h}{d} \text{ tenemos} \\
 &= \frac{I * \frac{h}{d}}{d^2} \therefore \\
 &= \frac{I * h}{d^3}
 \end{aligned} \tag{4.1}$$

$$\begin{aligned}
E_V &= \frac{I * \sin \alpha}{d^2} \text{ como } \sin \alpha = \frac{\overline{hp}}{d} \text{ tenemos} \\
&= \frac{I * \frac{\overline{hp}}{d}}{d^2} \text{ sabemos que } \overline{hp} = \sqrt{h^2 - d^2} \therefore \\
&= \frac{I * \sqrt{d^2 - h^2}}{d^3}
\end{aligned} \tag{4.2}$$

$$E = \sqrt{(E_H)^2 + (E_V)^2} \tag{4.3}$$

Generación de la data

Para la generación de la data se definió una muestra uniforme de 1728 puntos del escenario, a cada uno de los cuales se le calculó el valor de iluminancia correspondiente aplicando la ecuación 4.3. Se generaron dos (2) conjuntos de datos cuya estructura se compone de la posición (x, y, z) de cada fuente, la intensidad (I) de cada fuente, la posición (x, y, z) de cada punto, la distancia (d) desde entre cada una de las fuentes lumínicas y el punto y la altura (h) de cada punto.

Visualización

La visualización fue el instrumento de validación del cálculo de iluminancia para los puntos definidos en la malla. Para este proceso se consideraron dos configuraciones del escenario. En la primera, el escenario estaba iluminado con una sola fuente lumínica de 6522 lx de intensidad, ubicada de manera centrada sobre la cara superior del cubo que representa el escenario para garantizar simetría en la iluminación. En la segunda, el escenario se iluminó con tres fuentes lumínicas, la primera ubicada en el centro de la cara superior, la segunda ubicada en la cara lateral derecha y la tercera ubicada en la cara posterior; la intensidad lumínica de las tres fuentes fue 6522 lx, 3261 lx y 3261 lx respectivamente. Con cada configuración se hicieron tres ejercicios de visualización, en el primer ejercicio se consideraron el cálculo de la iluminancia (CI) y su visualización en escala real (VER), en el segundo se anuló la fuente lumínica (AF) pero se consideró su visualización en escala real (VAF) y, en el tercero, se tomaron en cuenta los cálculos del primer ejercicio (CI) pero la visualización del segundo (VAF). La tabla 4.1 muestra la casuística de los ejercicios de visualización.

Tabla 4.1. Casuística de los ejercicios de visualización

Escenario	No Fuentes	Posición de la fuente			Intensidad lumínica (lx)	Visualización y resultados			
		X	Y	Z		VER	CI	VAF	AF
1	1	5.5	5.5	10.5	6522	X	X		
							X	X	
									X
2	3	5.5	10.5	5.5	3261	X	X		
								X	X
							X	X	

4.3. Resultados

Primera configuración:

En el primer ejercicio de visualización era de esperarse la visualización de una distribución uniforme de la luz como lo muestra la figura 4.3; sin embargo, la escala consideraba saltos muy grandes entre los rangos, lo que significa que quedaban detalles por apreciarse a pesar de usar curvas de nivel para intentar detectarlos. Esto sucedió porque al considerar la ubicación de la fuente lumínica para el cálculo de la iluminancia, se presentaba una singularidad.

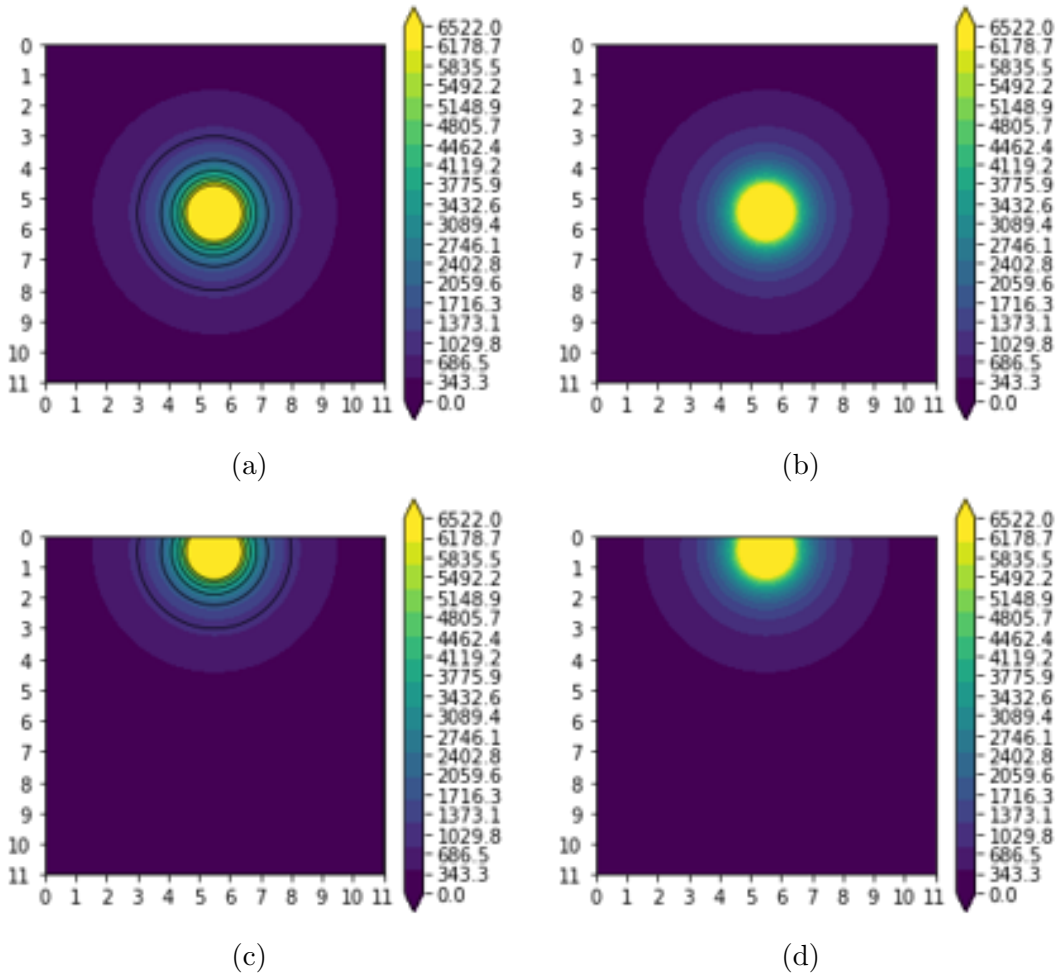


Figura 4.3. Distribución uniforme a escala real:(a) y (b) corresponden a un corte transversal por la cara superior del escenario debajo de donde se encuentra la fuente de luz. (c) y (d) corresponden a un corte perpendicular por el centro del escenario.

En el segundo ejercicio de visualización, para controlar la singularidad detectada, en el cálculo de la iluminancia no se incluyeron los puntos cuya distancia a la ubicación de la fuente lumínica estuviera en el rango $[0, 1]$. Así, al haberse evitado la singularidad, se pudo observar con mejor detalle la propagación de la luz en zonas más alejadas de la fuente lumínica como lo muestra la figura 4.4

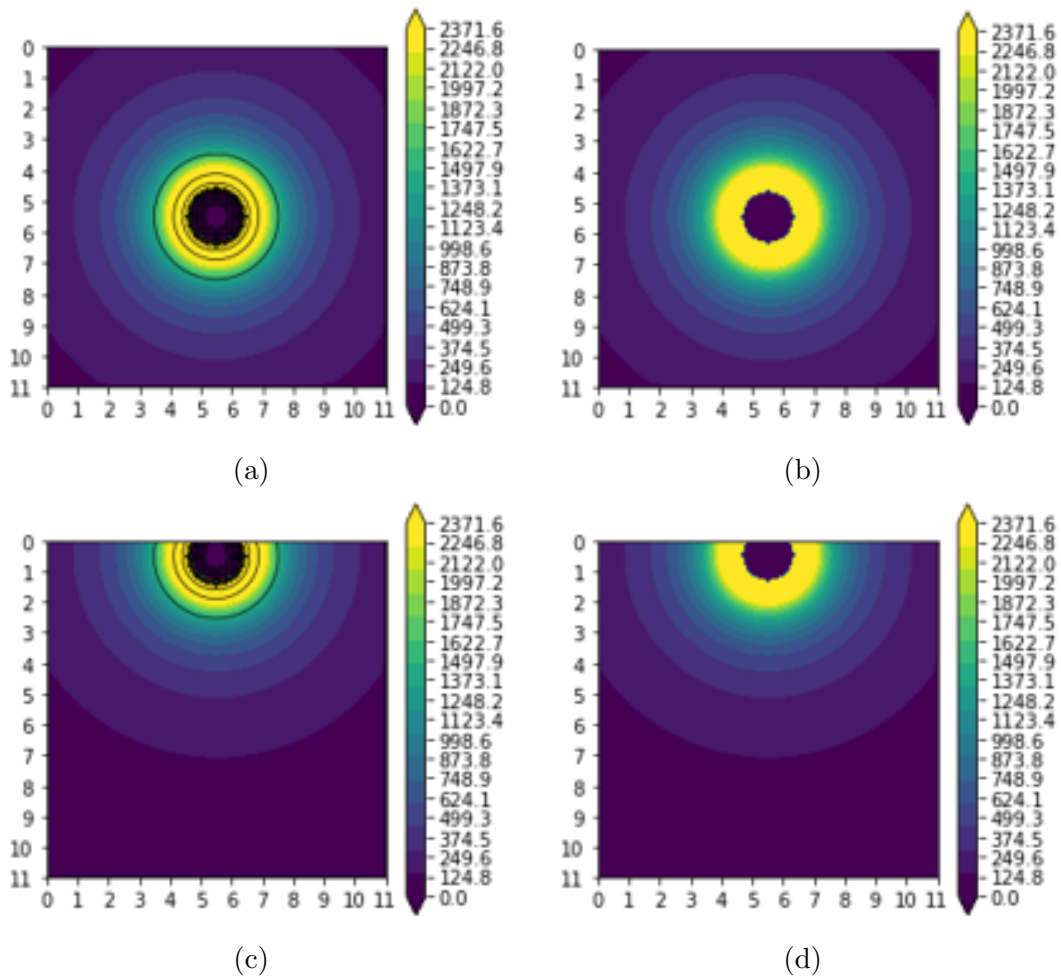


Figura 4.4. Distribución uniforme a escala real evitando la singularidad: (a) y (b) corresponden a un corte transversal por la cara superior del escenario debajo de donde se encuentra la fuente de luz. (c) y (d) corresponden a un corte perpendicular por el centro del escenario.

Finalmente, con el fin de evitar la singularidad en la visualización pero sí conservar los cálculos reales de la iluminancia, en el tercer ejercicio de visualización, se consideró el cálculo de la iluminancia y la visualización del primer y segundo ejercicio de visualización respectivamente. Sin embargo, aunque en este ejercicio se visualizaron mejores detalles y los cálculos reales, No fue suficiente para mostrar la cobertura total del área de corte, por lo cual se utilizó la técnica del falso color [111] para la visualización y así poder apreciar con mayor claridad las transiciones que hay entre los rangos de la escala, como se aprecia en la figura 4.5.

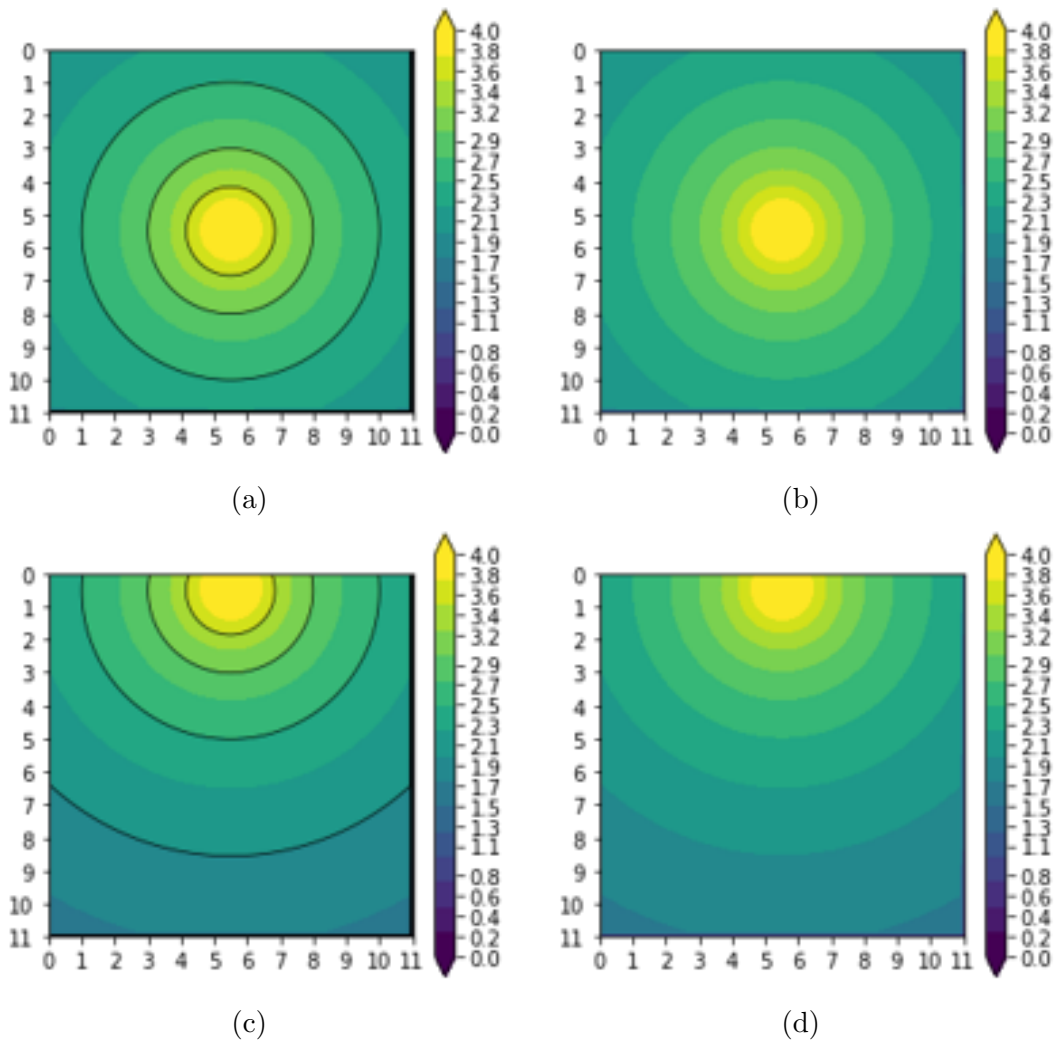


Figura 4.5. Distribución uniforme a escala logarítmica evitando singularidad: (a) y (b) corresponden a un corte transversal por la cara superior del escenario debajo de donde se encuentra la fuente de luz. (c) y (d) corresponden a un corte perpendicular por el centro del escenario.

Segunda configuración:

El propósito de definir más de una fuente lumínica fue romper con la simetría observada en los ejercicios de visualización para un escenario iluminado con una sola fuente lumínica (primera configuración) y así validar que la distribución de los valores de iluminancia fuera la esperada. En la figura 4.6 puede verse la distribución asimétrica de la iluminancia en escala logarítmica.

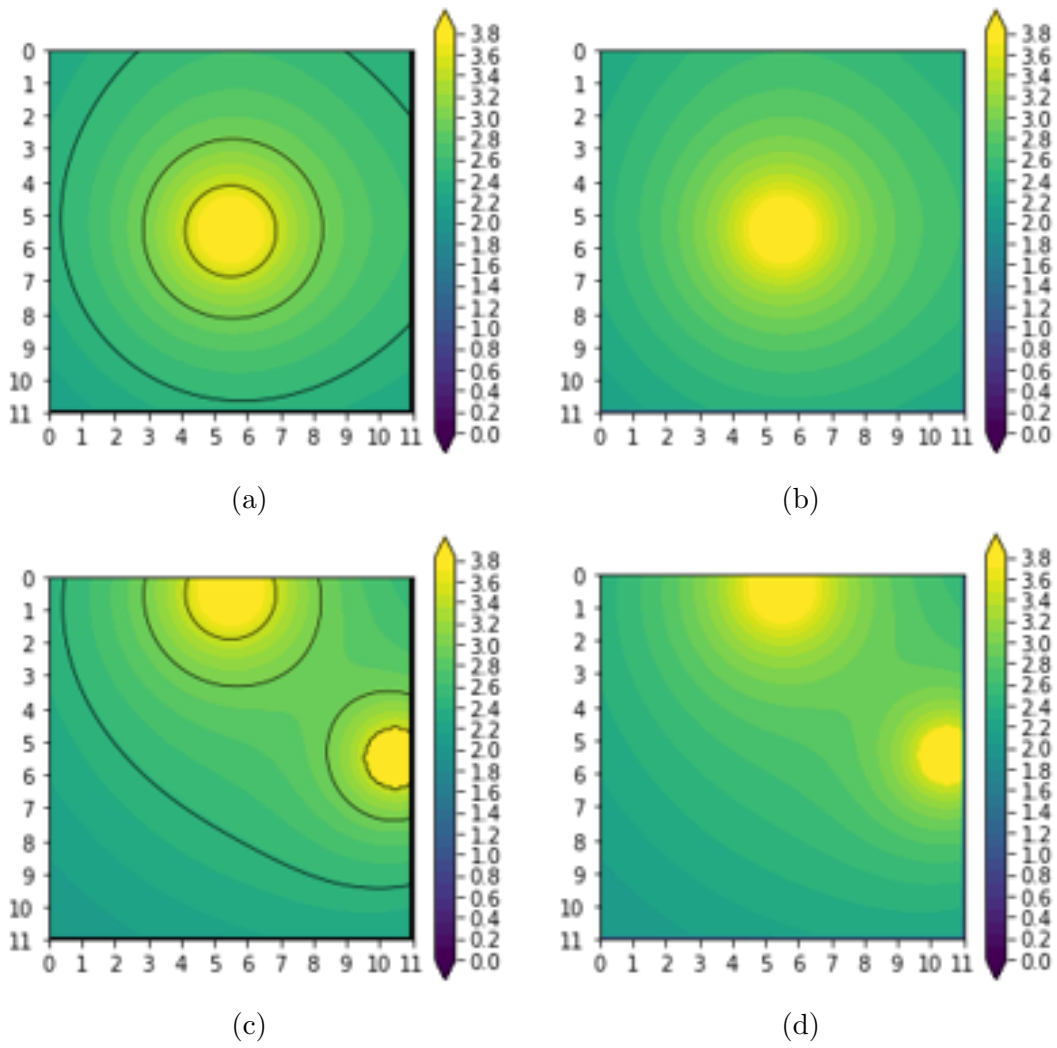


Figura 4.6. Distribución no uniforme a escala logarítmica evitando singularidad: (a) y (b) corresponden a un corte transversal por la cara superior del escenario debajo de donde se encuentra la fuente de luz. (c) y (d) corresponden a un corte perpendicular por el centro del escenario.

4.4. Conclusiones

Se logró un mejor detalle de la visualización cuando las subdivisiones son de once por cada metro cúbico, aunque el tiempo de cálculo aumentó casi nueve veces más que el de los escenarios subdivididos en uno por cada metro cúbico.

La escala logarítmica fue determinante para visualizar con mejor detalle los cambios de iluminancia en el escenario, esto permitió comprender la exactitud del cálculo de iluminancia y validarlo. La anulación de la fuente lumínica, en las visualizaciones, hizo posible enfocarse en los cambios de iluminancia tras descartar la singularidad que se presentaba en el cálculo de iluminancia en los puntos cuya distancia a la fuente lumínica es menor o igual a uno.

El espaciado entre curvas de nivel fue útil para validar que, en la escala real, la visualización debía considerar ajustes para no excluir de la visualización algunos rangos de iluminancia.

Estimación de valores de iluminancia mediante Deep Learning

5.1. Objetivo

El objetivo de este experimento fue desarrollar modelos de redes neuronales capaces de estimar el valor de iluminancia (E) en un punto dentro de un espacio tridimensional vacío, teniendo en cuenta una configuración precisa de fuentes de luz. Esta estimación se integraría en el proceso estándar utilizado para calcular la luminancia en una imagen, ofreciendo un método complementario para mejorar la precisión y la fidelidad visual de las representaciones generadas. Este enfoque busca aprovechar el poder de las redes neuronales para proporcionar una estimación más precisa y detallada de la iluminancia.

5.2. Método

Para obtener los modelos, el enfoque de este experimento implicó dos pasos principales: en primer lugar, se creó un entorno virtual utilizando Unreal Engine [112, 113], lo que hizo posible manipular la configuración de iluminación para la generación de datos. Posteriormente, utilizando Keras [114] y TensorFlow [115], se diseñaron dos redes neuronales (una feed-forward y una convolucional). La Figura 5.1 proporciona una representación visual del proceso involucrado en la obtención de cada modelo. Cada red neuronal pasó por un proceso iterativo de refinamiento, donde -aplicando la ecuación 5.1- se compararon los errores obtenidos en su fase de aprendizaje hasta lograr dos modelos de precisión satisfactoria para la estimación de la iluminación en cada tipo de red.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2} \tag{5.1}$$

Generación de la data

Para la generación de la data, se siguió el mismo método que se aplicó para la simulación del fenómeno interacción luz-materia. Se simuló un escenario en 3D para configurar la iluminación especificando la posición, intensidad y número de fuentes de luz. Por ejemplo, la Figura 5.2

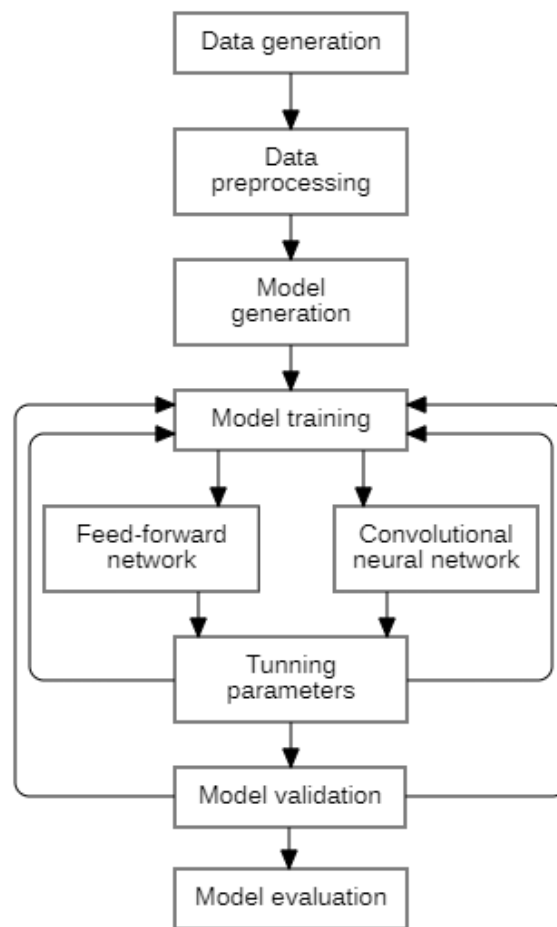
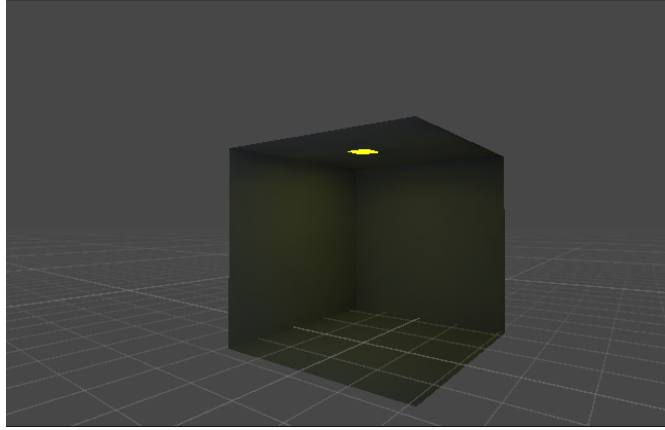
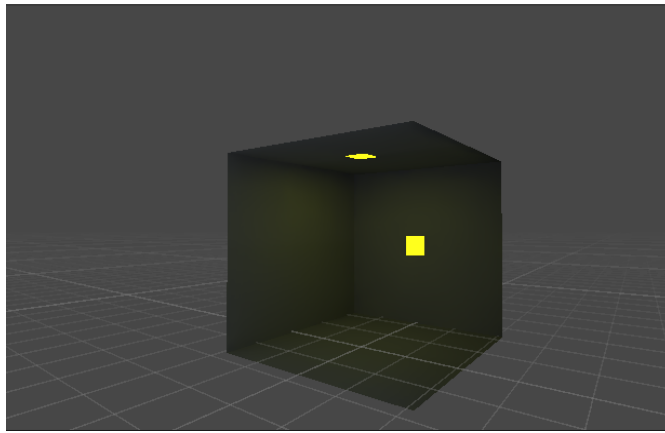


Figura 5.1. Proceso de adquisición del modelo.

ilustra dos escenarios potenciales: el primero con una sola fuente de luz para garantizar simetría en la iluminación, y el segundo con dos fuentes de luz para introducir asimetría.



(a) Escenario simétricamente iluminado



(b) Escenario asimétricamente iluminado

Figura 5.2. Escenarios con distintas configuraciones de iluminación.

Luego, se generó una muestra uniforme de 1728 puntos dentro del escenario. A cada uno de estos puntos, se aplicó la ley del coseno cuadrado de Lambert (ec. 5.2) [2], considerando la intensidad (I) de cada fuente de luz, la distancia (d) de cada fuente de luz al punto y la altura (h) del punto.

$$E = \sqrt{\left(\frac{I * h^2}{d^3}\right) + \left(\frac{I\sqrt{d^2 - h^2}}{d^3}\right)} \quad (5.2)$$

El Algoritmo 1 describe el método para crear el conjunto de datos. Antes de su ejecución, el algoritmo requiere una colección de puntos que definen la escena y una configuración que contiene información sobre las fuentes de luz que iluminan la escena. Para cada fuente de luz, se recopiló la información sobre su posición e intensidad luminosa (líneas 3 a 5). Luego, se determinaron la altura y la distancia de cada punto en el escenario dado con respecto a la fuente de luz (líneas 6 a 9). Si la distancia cae dentro del rango $[0, 1]$, la intensidad del punto se establece igual a la de la fuente de luz para evitar una singularidad (líneas 10 y 11). De lo contrario, se calcula la iluminación horizontal y vertical y estas se suman al valor de intensidad general del punto (líneas 13 a 18). Luego, se retornó el conjunto de datos, habiendo agregado previamente la información de la fuente de luz, la información del punto y los valores de iluminación horizontal

y vertical (líneas 20 a 23).

Algorithm 1: Generación de Datos

Input: *lst_lights*, *lst_points*

Output: *dataset*

```

1 dataset  $\leftarrow []$ ;
2 foreach light in lst_lights do
3   light  $\leftarrow$  lst_lights[f].p;
4   I  $\leftarrow$  lst_lights[f].I;
5   foreach p in lst_points do
6     h  $\leftarrow$  | light.z - p.z |;
7     p.h  $\leftarrow$  h;
8     dist  $\leftarrow \sqrt{(\text{light.x} - p.x)^2 + (\text{light.y} - p.y)^2 + (\text{light.z} - p.z)^2}$ ;
9     if dist  $\geq 0$  and dist  $\leq 1$  then
10      | p.I  $\leftarrow$  I;
11    end
12    else
13      EH  $\leftarrow$  (I  $\times$  h) / (dist3);
14      EV  $\leftarrow$  (I  $\times \sqrt{(\text{dist}^2) - (h^2)}$ ) / (dist3);
15      p.I  $\leftarrow$  p.I +  $\sqrt{(EH^2) + (EV^2)}$ ;
16      if p = light then
17        | p.I  $\leftarrow$  I;
18      end
19    end
20    dataset.append([light.x, light.y, light.z, I, p.x, p.y, p.z, dist, h, EH, EV, p.I]);
21  end
22 end
23 return dataset;

```

Las Figuras 5.3 y 5.4 muestran tres secciones transversales a lo largo de los ejes x, y, y z para evaluar la distribución de luz en el escenario, lo que proporciona información valiosa sobre las diversas condiciones de iluminación que surgen de diferentes configuraciones, facilitando la comprensión de los impactos sutiles de los arreglos de iluminación en la distribución general de brillo dentro del escenario. La Figura 5.3 representa una escena con iluminación simétrica, con una única fuente de luz ubicada en el centro del lado superior de la escena. La fuente de luz tiene una intensidad de 6522 lx y, como se puede observar, proporciona una distribución de luz consistente en todo el escenario.

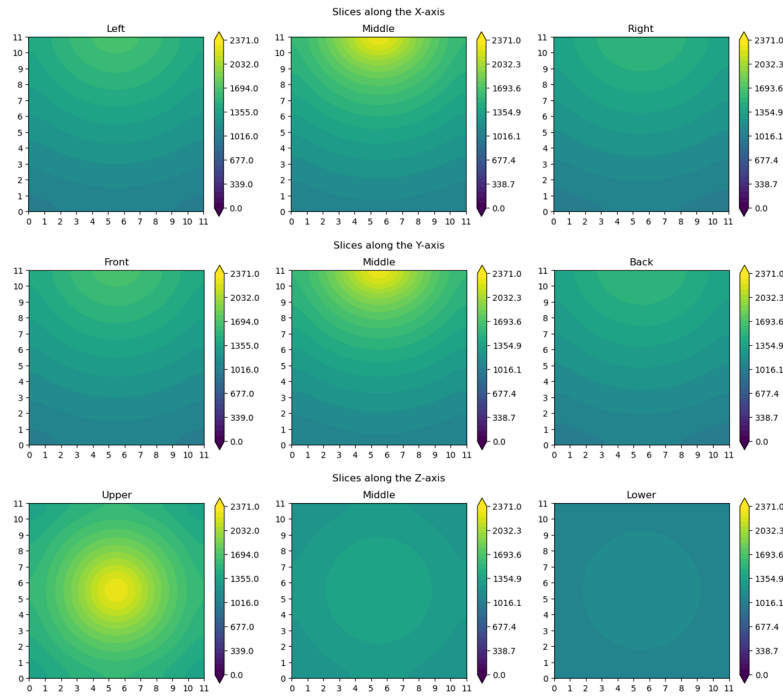


Figura 5.3. Representación detallada de la distribución de la luz en un escenario con una única fuente de luz

La Figura 5.4 representa un arreglo de iluminación asimétrico con dos fuentes de luz independientes. La fuente inicial, que tiene una intensidad luminosa de 4000 lx, se encuentra en el centro superior de la escena. En contraste, la segunda fuente, con una intensidad de 2522 lx, está ubicada en la esquina superior derecha de la escena. Esta configuración produce una dispersión de luz variada, resaltando las variaciones en el brillo a lo largo del rango espacial del escenario.

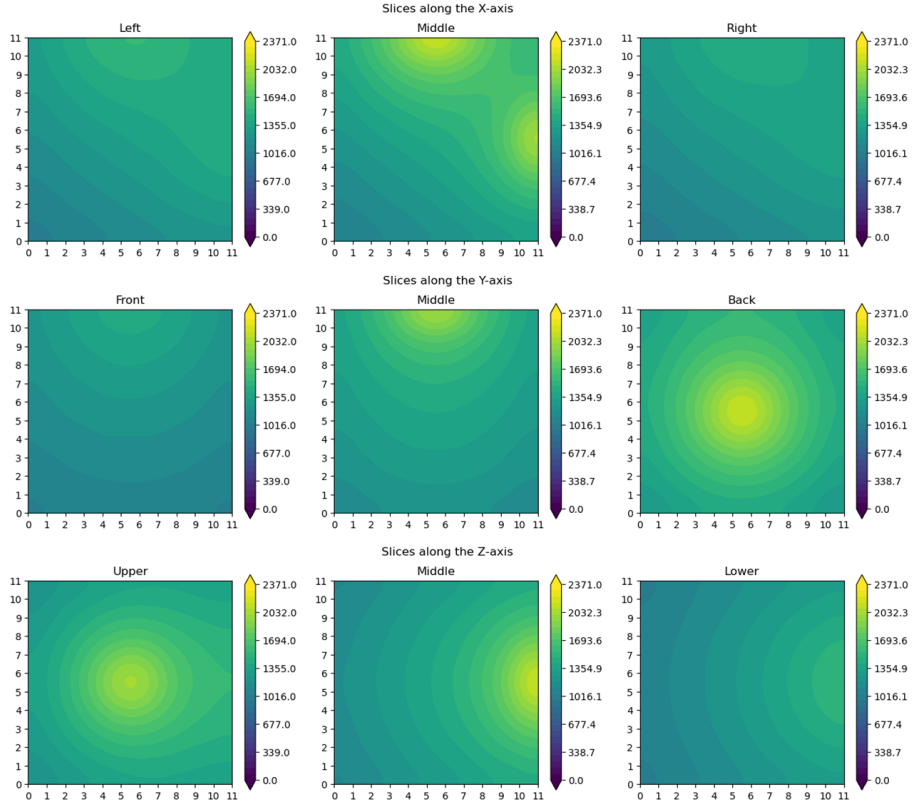


Figura 5.4. Representación detallada de la distribución de la luz en un escenario con dos fuentes de luz

Preprocesamiento de la data

Para garantizar que las características están en la misma escala, lo que se sabe que mejora el rendimiento de los algoritmos de optimización y aprendizaje automático [116], se normalizaron los datos del conjunto de datos. Esto se hizo mediante la aplicación de la ecuación 5.3, teniendo en cuenta la amplia gama de valores cubiertos por el conjunto de datos.

$$X_{norm} = \frac{X_i - X_{max}}{X_{max} - X_{min}} \quad (5.3)$$

A continuación, el conjunto de datos se dividió en tres partes: el 80 % de los datos se asignó aleatoriamente a la fase de entrenamiento, mientras que el 20 % restante se repartió equitativamente entre las fases de validación y prueba.

Enfoque de redes neuronales Feed-Forward

Mediante experimentación, se ajustaron iterativamente los valores paramétricos de dos redes, como la tasa de aprendizaje y el número de neuronas en la capa oculta. Así, la tabla 5.1 muestra diferentes modelos experimentales que cambiaron tanto el número de neuronas como la tasa de aprendizaje para estimar los valores de iluminación en un entorno con iluminación simétrica. Como puede observarse, el Modelo 6 (FFNN-1) demostró una precisión superior en la estimación.

Tabla 5.1. Ajuste óptimo de parámetros para una red feed-forward en una configuración de iluminación simétrica.

Modelo	Tasa de aprendizaje	Neuronas	Mean Squared Error
1	0.01	70	0.000076
2	0.01	80	0.000083
3	0.01	90	0.000007
4	0.001	70	0.000057
5	0.001	80	0.000032
6 (FFNN-1)	0.001	90	0.000002
7	0.0001	70	0.000015
8	0.0001	80	0.000023
9	0.0001	90	0.000006

La Tabla 5.2 presenta varios modelos experimentales que varían en el número de neuronas y la tasa de aprendizaje. Estos modelos se utilizaron para estimar los valores de iluminación en un entorno con iluminación asimétrica. En esta instancia, el Modelo 8 (FFNN-2) exhibió una precisión superior en sus estimaciones.

Tabla 5.2. ajuste óptimo de parámetros para una red feed-forward en una configuración de iluminación asimétrica.

Modelo	Tasa de aprendizaje	Neuronas	Mean Squared Error
1	0.01	70	0.009691
2	0.01	80	0.006986
3	0.01	90	0.000108
4	0.001	70	0.00012
5	0.001	80	0.000158
6	0.001	90	0.000182
7	0.0001	70	0.000098
8 (FFNN-2)	0.0001	80	0.000094
9	0.0001	90	0.000141

Este proceso condujo a la identificación de una arquitectura de red óptima, como se muestra en la Figura 5.5, mientras que la Figura 5.6 representa las curvas de aprendizaje asociadas con cada uno de esos modelos.

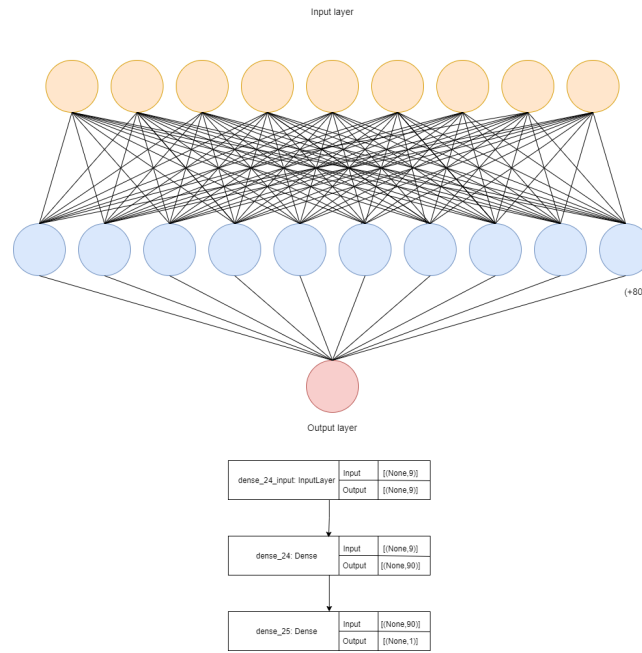


Figura 5.5. Arquitectura de las FFNNs.

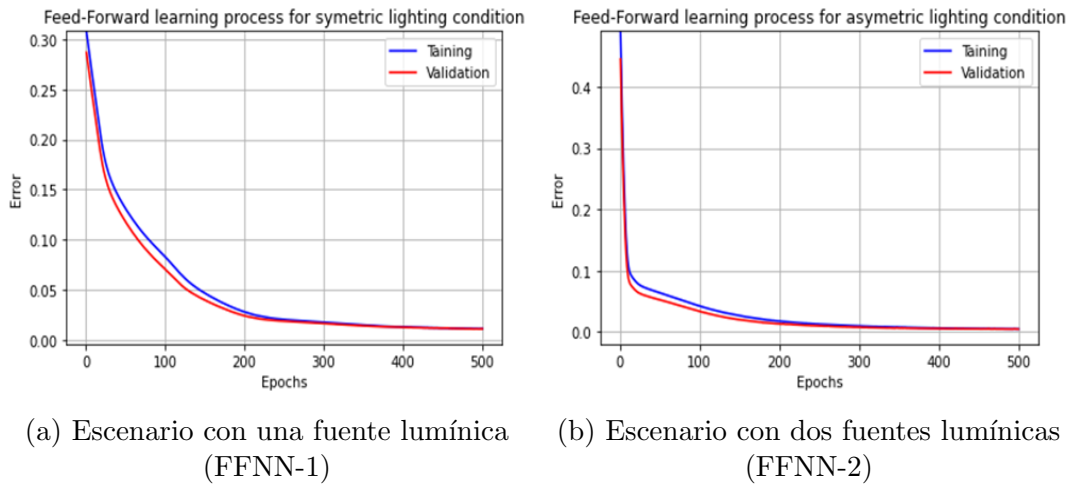


Figura 5.6. Gráficos de aprendizaje de FFNN para diferentes configuraciones de iluminación.

Enfoque de redes neuronales convolucionales

Se diseñaron dos redes convolucionales (CNN-1 y CNN-2), y sus valores de parámetros se obtuvieron mediante un proceso iterativo similar al utilizado para las redes feed-forward. En estos modelos experimentales, el único parámetro que experimentó variación fue la tasa de aprendizaje. El Modelo 9 (CNN-1) se muestra como el más preciso en la tabla 5.3 para estimar los valores de iluminación en un espacio con iluminación uniforme.

Tabla 5.3. Ajuste óptimo de parámetros para una red neuronal convolucional en una configuración de iluminación simétrica.

Modelo	Tasa de aprendizaje	Mean Squared Error
1	10	15748180.0
2	1	28.7261524
3	0.5	0.8346124
4	0.1	0.0336059
5	0.05	0.0320606
6	0.01	0.0000985
7	0.005	0.0000695
8	0.001	0.0001140
9 (CNN-1)	0.0005	0.0000921
10	0.0001	0.0004632
11	0.00005	0.0009414
12	0.00001	0.0047176
13	0.000005	0.0049788
14	0.000001	0.0277012

De igual manera, el Modelo 7 (CNN-2) en la tabla 5.4 fue el más preciso para predecir la cantidad de luz en una situación con simetría en la configuración de las fuentes lumínica.

Tabla 5.4. Ajuste óptimo de parámetros para una red neuronal convolucional en una configuración de iluminación asimétrica.

Modelo	Tasa de aprendizaje	Mean Squared Error
1	10	76042736.0
2	1	145.2624359
3	0.5	0.791321397
4	0.1	0.030259281
5	0.05	0.005472712
6	0.01	0.000259001
7 (CNN-2)	0.005	0.000128485
8	0.001	0.000164776
9	0.0005	0.000246788
10	0.0001	0.00067971
11	0.00005	0.0016336
12	0.00001	0.005003111
13	0.000005	0.010371411
14	0.000001	0.022327378

La arquitectura de estas redes convolucionales se ilustra en la figura 5.7, y la figura 5.8 muestra las curvas de aprendizaje para cada modelo.

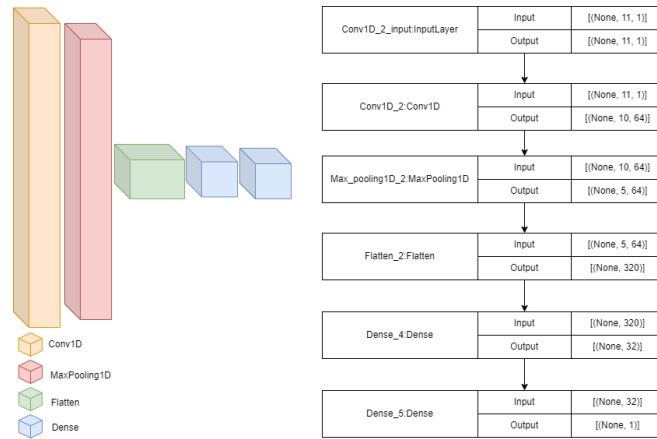


Figura 5.7. Arquitectura de las CNN.

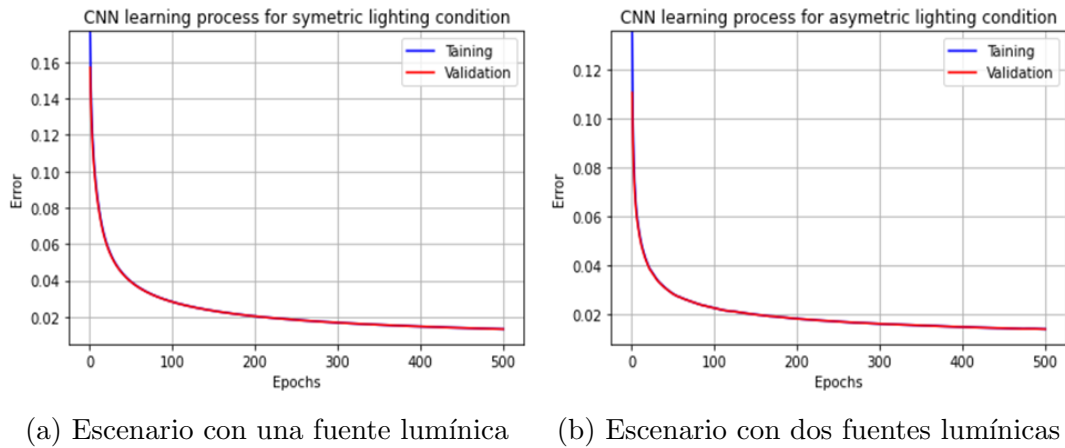


Figura 5.8. Gráficos de aprendizaje de redes neuronales convolucionales para diferentes configuraciones de iluminación.

Estimación de la luminancia (L) de una imagen mediante los modelos

Para este ejercicio, fue crucial comprender la distinción entre dos conceptos fundamentales: la luminancia (L) y la iluminancia (E). Según [117], la luminancia (L) se refiere al brillo percibido de una superficie, mientras que la iluminancia (E) cuantifica la cantidad de luz incidente sobre dicha superficie. Así entonces se generó un total de dieciocho escenarios de prueba, que comprendían seis escenas iluminadas con una única fuente de luz, seis escenas iluminadas con dos fuentes de luz y seis escenas iluminadas con tres fuentes de luz. Posteriormente, empleando el método descrito en [118], se usó la ecuación 5.4 para calcular la luminancia de cada escena mediante un cálculo de suma ponderada. Finalmente, se compararon estos resultados con las estimaciones proporcionadas por las redes convolucionales para cada escena respectiva.

$$I(x, y) = [0.2989 \quad 0.5870 \quad 0.1140] \begin{bmatrix} R(x, y) \\ G(x, y) \\ B(x, y) \end{bmatrix} \quad (5.4)$$

$$L = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} I(x, y)$$

Donde:

- $I(x, y)$: Representa la función que hace la conversión de un píxel a escala de grises basada en sus coordenadas x e y .
- M, N : Representan el ancho y el alto de la imagen, respectivamente.

El Algoritmo 2 muestra el cálculo del brillo de la imagen, requiriendo la recuperación de todos los datos de píxeles y las dimensiones de la imagen (líneas 2-4). La imagen se convierte a escala de grises y se rota de manera lineal multiplicando el componente RGB de cada píxel por su valor correspondiente en la secuencia $[0.2989, 0.5870, 0.1140]$. Finalmente, el valor total se divide por el tamaño de la imagen (líneas 5-10).

Algorithm 2: Algoritmo para el cálculo de luminancia

Input : Image img

Output: Luminance L

```

1 pixels ← img.load();
2 L ← 0;
3 S ← img.size();
4 for x in 1 to S[0] do
5   | for y in 1 to S[1] do
6   |   L ← L + (img[x][y] × [0.2989, 0.5870, 0.1140]);
7   | end
8 end
9 return((L × 1/(S[0] × S[1])));
```

5.3. Resultados

Enfoque de redes neuronales Feed-Forward

Cuando solo había una fuente de luz, el modelo logró determinar con gran precisión el brillo de los objetos, con una tasa de éxito del 98.8 %. De manera más general, al considerar situaciones con dos fuentes de luz, el modelo mostró un alto nivel de precisión, con una tasa de precisión del 96.0 %. Los valores de iluminancia obtenidos a partir de los cálculos para ambos escenarios se representan visualmente en la figura 5.9. La imagen muestra los valores esperados para la iluminación en una configuración con una sola fuente de luz (figura 5.9A), las proyecciones realizadas por la red neuronal feed-forward (figura 5.9B), y la diferencia entre los valores esperados y predichos (figura 5.9C). Además, el diagrama presenta los niveles anticipados de luminosidad para

una configuración que incluye dos fuentes de luz (figura 5.9D), los valores calculados derivados de la red neuronal feed-forward (figura 5.9E), y la disparidad entre ambos (figura 5.9F).

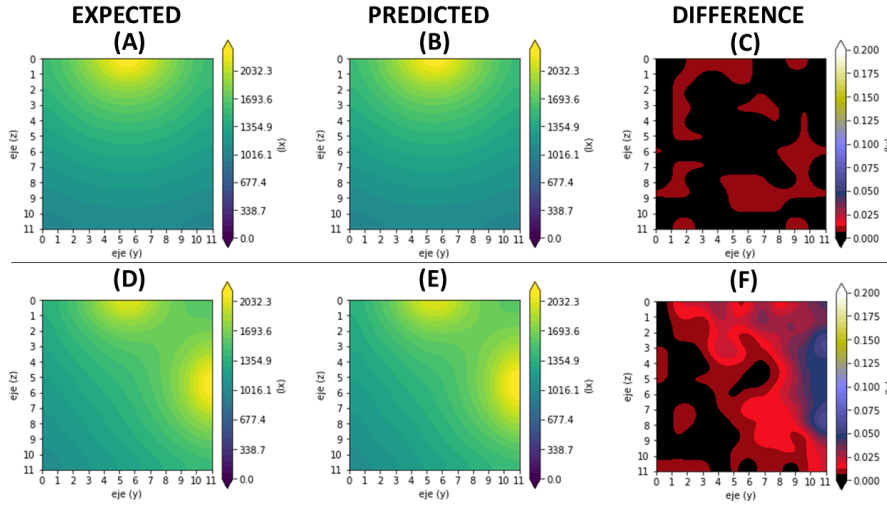


Figura 5.9. Estimación de la red Feed-Forward (FFNN). Los valores esperados para dos escenarios iluminados por una y dos fuentes de luz se muestran en las subfiguras A y D. Las predicciones de iluminancia de FFNN para ambos escenarios se encuentran en las subfiguras B y E. Las subfiguras C y F muestran las discrepancias entre los valores esperados de cada escenario y las proyecciones realizadas.

Enfoque de las redes Convolucionales

En situaciones donde solo había una fuente de luz, el modelo demostró un alto nivel de precisión en sus estimaciones, logrando una tasa de precisión del 99.2%. En casos donde el escenario estaba iluminado por dos fuentes de luz, el nivel de precisión observado se redujo ligeramente al 96.4%. Los valores de iluminancia estimados para ambos tipos de escenarios se presentan en la figura 5.10. La figura 5.10A ilustra los valores de iluminancia predichos para una configuración con una única fuente de luz. Por otro lado, la figura 5.10B muestra la estimación comparable derivada del modelo de Red Neuronal Convolutiva (CNN). Finalmente, la figura 5.10C exhibe la disparidad entre estas dos representaciones. La figura 5.10D muestra los valores de iluminancia predichos cuando se consideraban dos fuentes de luz. La estimación de la CNN se ilustra en la figura 5.10E, mientras que la figura 5.10F presenta la disparidad entre ambas.

Redes Neuronales Feed-Forward Vs Redes Neuronales Convolucionales

La superioridad de uno u otro tipo de modelo, FFNN o CNN, dependió del contexto específico y de las características del problema en cuestión. Los distintos escenarios mostraron diferentes niveles de precisión para cada tipo de modelo. En este caso, los modelos CNN superaron a los modelos FFNN con un ligero margen. En la Tabla 5.5 se comparan las estimaciones obtenidas con los modelos.

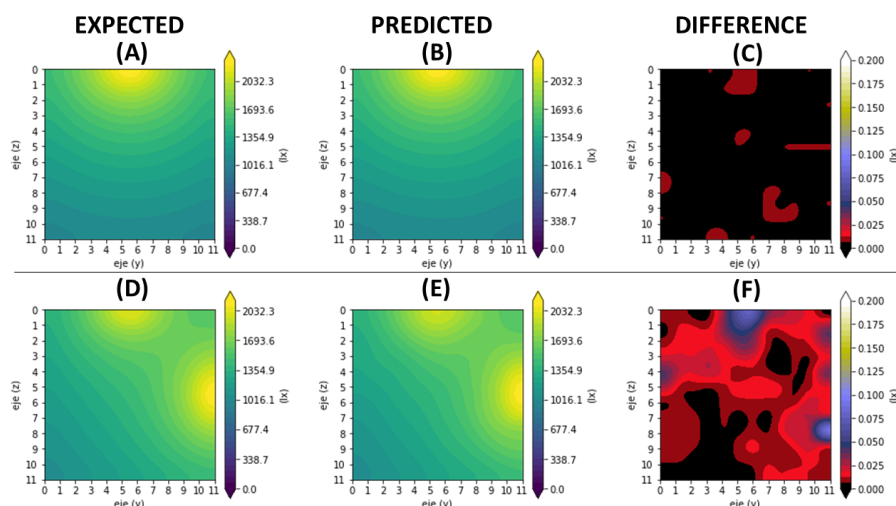


Figura 5.10. Estimaciones de la red neuronal convolucional (CNN). Se muestran los valores esperados para dos escenarios iluminados por una y dos fuentes de luz en las subfiguras A y D.

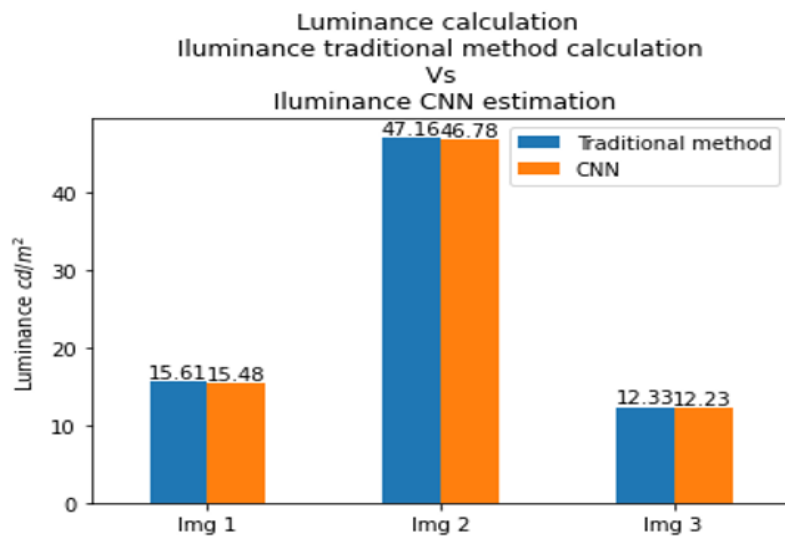
Las predicciones de iluminancia de la CNN para ambos escenarios se presentan en las subfiguras B y E. Las subfiguras C y F ilustran las discrepancias entre los valores esperados de cada escenario y las predicciones.

Tabla 5.5. Comparación entre los modelos FFNN y CNN

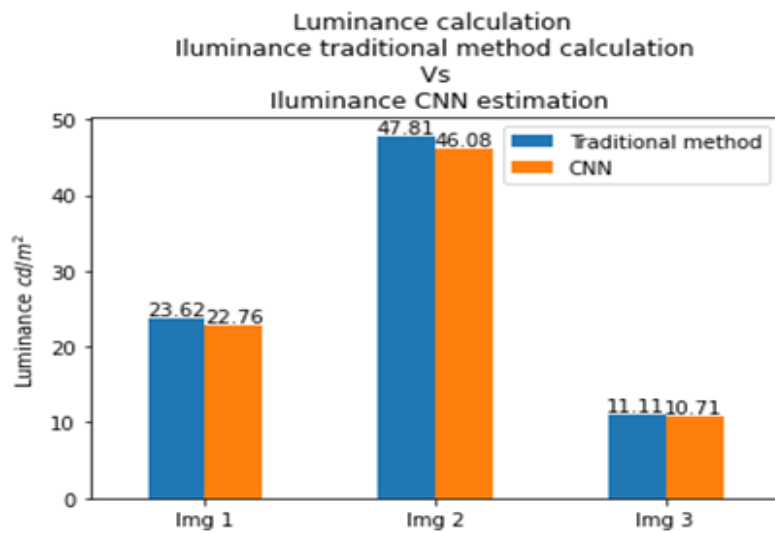
Configuración lumínica	Modelo	Precisión
Una fuente lumínica	FFNN-1	98.8 %
	CNN-1	99.2 %
Dos fuentes lumínicas	FFNN-2	96.0 %
	CNN-2	96.4 %

Comparación de las estimaciones basadas en CNN con el método tradicional en distintas condiciones de iluminación

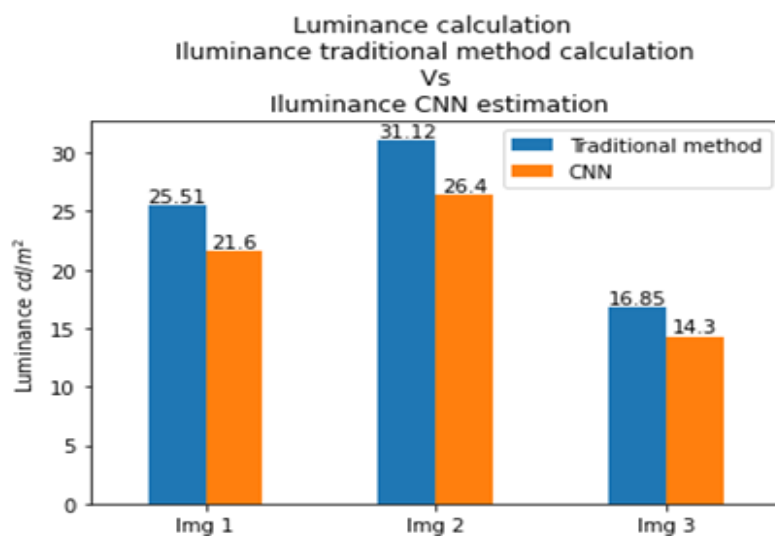
En las escenas en las que la iluminación se ajustó a los datos de entrenamiento de las CNN, concretamente los escenarios con una y dos fuentes de luz, las estimaciones proporcionadas por estas coincidieron estrechamente con las calculadas por el método tradicional. Sin embargo, en el caso de la tercera escena, en la que intervinieron tres fuentes de luz, la discrepancia entre los cálculos y las estimaciones se hizo más pronunciada, lo que indica que hubo una disminución de la precisión del modelo. La figura 5.11 presenta los resultados obtenidos, contrastando las estimaciones basadas en CNN con los resultados derivados de aplicar el método tradicional en escenarios con una, dos y tres fuentes de luz.



(a) Configuración con una fuente lumínica



(b) Configuración con dos fuentes lumínicas



(c) Configuración con tres fuentes lumínicas

Figura 5.11. Comparación de resultados: CNN vs. método tradicional.

Inferencia causal en el cálculo de la iluminancia

La inferencia causal se refiere a la comprensión de las relaciones de causa y efecto entre diferentes variables en un sistema [119]. En el contexto de la ecuación para calcular la iluminancia $E = \sqrt{(E_H)^2 + (E_V)^2}$, es posible aplicar la inferencia causal para entender cómo varios factores afectan la iluminancia en un entorno determinado. Como fue indicado anteriormente, E representa la iluminancia total, E_H y E_V son las componentes horizontales y verticales de la iluminancia, respectivamente; de manera que es posible considerar que estas componentes representan diferentes fuentes de luz o diferentes características de la iluminación en un entorno determinado. Aplicando la inferencia causal, fue posible examinar cómo varios factores afectan las componentes horizontales y verticales de la iluminancia, lo que a su vez influye en la iluminancia total E . Algunos factores que podrían influir en estas componentes podrían ser:

- **Orientación de las fuentes de luz:** La orientación de las fuentes de luz podría afectar tanto a la componente horizontal como a la vertical de la iluminancia. Por ejemplo, una fuente de luz situada en un ángulo particular podría tener un impacto más significativo en la iluminancia horizontal que en la vertical, o viceversa. Así, es posible modelar la influencia de la orientación de las fuentes de luz (θ) en las componentes horizontales y verticales de la iluminancia por medio de las siguientes ecuaciones estructurales: $E_H = f(\theta, Z_1)$, $E_V = g(\theta, Z_2)$, donde Z_1 y Z_2 son variables latentes que representan otros factores no observados que pueden influir en la iluminancia.
- **Distribución espacial de las fuentes de luz:** La manera en que están distribuidas las fuentes de luz en el entorno también podría influir en las componentes horizontales y verticales de la iluminancia. Por ejemplo, una distribución más concentrada de fuentes de luz podría generar una mayor iluminancia en ciertas direcciones, afectando así las componentes horizontal y vertical. Así, es factible modelar la influencia de la distribución espacial de las fuentes de luz (D) en las componentes horizontales y verticales de la iluminancia por medio de las siguientes ecuaciones estructurales: $E_H = h(D, Z_3)$, $E_V = i(D, Z_4)$, donde Z_3 y Z_4 son variables latentes adicionales.
- **Reflectividad de las superficies:** La reflectividad de las superficies en el entorno puede influir en cómo se distribuye la luz, afectando así las componentes horizontal y vertical de la iluminancia. Por ejemplo, superficies altamente reflectantes pueden aumentar la iluminancia horizontal en comparación con la vertical. Así, la reflectividad (R) puede influir en la iluminancia, lo cual puede ser modelado a través de las siguientes ecuaciones estructurales: $E_H = j(R, Z_5)$, $E_V = k(R, Z_6)$, donde Z_5 y Z_6 son variables latentes relacionadas con la reflectividad.
- **Obstrucciones y sombras:** La presencia de obstrucciones o sombras en el entorno puede afectar la distribución de la luz y, por lo tanto, las componentes horizontal y vertical de la iluminancia. Así, la presencia de obstrucciones o sombras (O) puede influir en las componentes de iluminancia de la siguiente manera: $E_H = l(O, Z_7)$, $E_V = m(O, Z_8)$, donde Z_7 y Z_8 son variables latentes asociadas con las obstrucciones y sombras.

Utilizando la regla de la cadena de Pearl [120], es posible calcular cómo cada uno de estos factores afecta la iluminancia total E , de manera que para cada uno de estos factores, es necesario calcular las derivadas parciales correspondientes de las componentes horizontales y verticales de

la iluminancia con respecto a los factores causales relevantes. No obstante, para realizar un análisis causal más concreto, se haría necesario contar con datos específicos sobre cómo cada factor afecta la iluminancia en un entorno real. Dado que el conjunto de datos generado no cuenta con esos datos específicos, se usaron valores hipotéticos para ilustrar el proceso de análisis causal así:

Orientación de las fuentes de luz (θ)

Incrementar la orientación de las fuentes de luz en 10 grados (θ) aumenta la iluminancia horizontal (E_H) en 100 lux y la iluminancia vertical (E_V) en 50 lux.

$$\frac{\partial E}{\partial \theta} = \frac{\partial E}{\partial E_H} \frac{\partial E_H}{\partial \theta} + \frac{\partial E}{\partial E_V} \frac{\partial E_V}{\partial \theta}$$

Dado que: $\frac{\partial E}{\partial E_H} = 1$, $\frac{\partial E}{\partial E_V} = 1$, $\frac{\partial E_H}{\partial \theta} = 100 \text{ lux}/^\circ$, $\frac{\partial E_V}{\partial \theta} = 50 \text{ lux}/^\circ$

Entonces: $\frac{\partial E}{\partial \theta} = (1 \times 100) + (1 \times 50) = 150 \text{ lux}/^\circ$

Distribución espacial de las fuentes de luz (D)

Agregar una luz adicional en el techo (D) aumenta la iluminancia horizontal (E_H) en 150 lux y la iluminancia vertical (E_V) en 100 lux.

$$\frac{\partial E}{\partial D} = \frac{\partial E}{\partial E_H} \frac{\partial E_H}{\partial D} + \frac{\partial E}{\partial E_V} \frac{\partial E_V}{\partial D}$$

Dado que: $\frac{\partial E_H}{\partial D} = 150 \text{ lux/luz}$, $\frac{\partial E_V}{\partial D} = 50 \text{ lux/luz}$

Entonces: $\frac{\partial E}{\partial D} = (1 \times 150) + (1 \times 50) = 200 \text{ lux/luz}$

Reflectividad de las superficies (R)

Pintar las paredes con pintura reflectante (R) aumenta la iluminancia horizontal (E_H) en 80 lux y la iluminancia vertical (E_V) en 40 lux.

$$\frac{\partial E}{\partial R} = \frac{\partial E}{\partial E_H} \frac{\partial E_H}{\partial R} + \frac{\partial E}{\partial E_V} \frac{\partial E_V}{\partial R}$$

Dado que: $\frac{\partial E_H}{\partial R} = 80 \text{ lux/reflectividad}$, $\frac{\partial E_V}{\partial R} = 40 \text{ lux/reflectividad}$

Entonces: $\frac{\partial E}{\partial R} = (1 \times 80) + (1 \times 40) = 120 \text{ lux/reflectividad}$

Obstrucciones y sombras (O)

Eliminar una partición en la oficina (O) aumenta la iluminancia horizontal (E_H) en 120 lux y la iluminancia vertical (E_V) en 60 lux

$$\frac{\partial E}{\partial O} = \frac{\partial E}{\partial E_H} \frac{\partial E_H}{\partial O} + \frac{\partial E}{\partial E_V} \frac{\partial E_V}{\partial O}$$

Dado que: $\frac{\partial E_H}{\partial O} = 120 \text{ lux/partición}$, $\frac{\partial E_V}{\partial O} = 60 \text{ lux/partición}$

Entonces: $\frac{\partial E}{\partial O} = (1 \times 120) + (1 \times 60) = 180 \text{ lux/partición}$

Al entender cómo estos factores afectan las componentes horizontales y verticales de la iluminancia, es posible tener una comprensión más completa de cómo se determina la iluminancia total en un entorno dado. Esto permite tomar decisiones informadas sobre cómo diseñar o ajustar la iluminación para cumplir con ciertos requisitos o necesidades específicas. Por ejemplo, en cuanto a la orientación de las fuentes de luz, es posible observar que un aumento en la orientación de las fuentes de luz puede aumentar significativamente la iluminancia total en el escenario.

En cuanto a la distribución espacial de las fuentes de luz, la instalación de luces adicionales podría aumentar aún más la iluminancia total en el escenario. Por lo tanto, una posible decisión podría ser agregar más luces en áreas específicas. En cuanto a la reflectividad de las superficies, pintar las paredes de un color reflectante podría ser una decisión efectiva para aumentar la iluminancia total en la oficina, pues al aumentar la reflectividad de las superficies, se incrementaría la cantidad de luz reflejada en el espacio, lo que resultaría en una mayor iluminancia horizontal y vertical en todo el escenario.

En cuanto a las obstrucciones y sombras, eliminar o reubicar elementos que creen sombras podría mejorar la iluminancia total en el escenario, pues al eliminar obstáculos que bloqueen la luz, se permitiría una distribución más uniforme de la iluminancia en todo el escenario, lo que resultaría en una mejora en la iluminación general.

5.4. Conclusiones

Este experimento demostró el éxito de la aplicación de técnicas de Deep Learning para estimar valores de iluminancia en diversas configuraciones de iluminación. Aprovechando un volumen considerable de datos y ajustando cuidadosamente los parámetros de las redes neuronales, se lograron estimaciones con niveles aceptables de precisión.

Aunque la precisión de las estimaciones de las redes neuronales no alcanzó el 100 %, es considerable aducir que la precisión conseguida es suficientemente fiable para incorporar el aprendizaje profundo a la tarea de estimación de la iluminancia. Sin embargo, fue evidente que la precisión de la estimación disminuyó significativamente cuando el entrenamiento de los modelos no tuvo en cuenta el número de fuentes de luz presentes en la escena. Esta observación pone de manifiesto la necesidad de diseñar modelos adaptables a las variaciones en las configuraciones de iluminación.

Los hallazgos de este experimento subrayan el potencial del aprendizaje profundo como una herramienta valiosa para la estimación de la iluminancia, al tiempo que destacan la importancia de abordar los desafíos relacionados con la variación del número de fuentes de luz. El desarrollo de modelos adaptativos capaces de estimar con precisión la iluminancia bajo diversas configuraciones de iluminación presenta un área interesante para la investigación futura.

El uso de la inferencia causal en el cálculo de la iluminancia ofrece una visión más detallada sobre cómo diferentes factores inciden en la distribución de la luz en un entorno específico. Se identificaron diversos elementos influyentes, como la orientación y distribución de las fuentes de luz, la reflectividad de las superficies y la presencia de obstáculos o sombras. Estos factores tienen un impacto notable en la iluminancia total del área, afectando tanto su componente horizontal

como vertical y, por ende, la calidad general de la iluminación.

En general, este experimento aportó evidencias sólidas de la aplicabilidad del aprendizaje profundo en la estimación de la iluminancia, sentando las bases para nuevos avances en este campo y allanando el camino para modelos más precisos y adaptables en escenarios de iluminación del mundo real.

Inferencia causal para determinar la causa de un fenómeno de sombra en una imagen

6.1. Objetivo

El objetivo de este experimento fue desarrollar un modelo artificial que utilizara la inferencia causal para explicar la formación de una sombra proyectada sobre la superficie inferior de un escenario tridimensional. En este entorno, además de la iluminación, se incluyó la presencia de un objeto esférico. Al aplicar la explicabilidad en inteligencia artificial (XAI) a este contexto, se buscó comprender cómo el modelo interpreta y analiza los datos de entrada, especialmente en relación con la formación de la sombra captada en una imagen. Esto permitió una comprensión más profunda de cómo el modelo utiliza la información causal para generar predicciones y explicar fenómenos visuales complejos.

6.2. Método

Generación de la data

Para la generación de la data produjimos sintéticamente cuatro posibles escenas en las que, a pesar de haber muchas más como el color y la posición de los objetos, entre otros, definimos cuatro características observables. En la primera escena se observan la luz, la esfera, la superficie y la sombra, en la segunda escena solo se observan la luz y la esfera, en la tercera escena se perciben la luz y la superficie, y en la cuarta escena ninguna característica se percibe. La figura 6.1 muestra las cuatro posibles escenas.

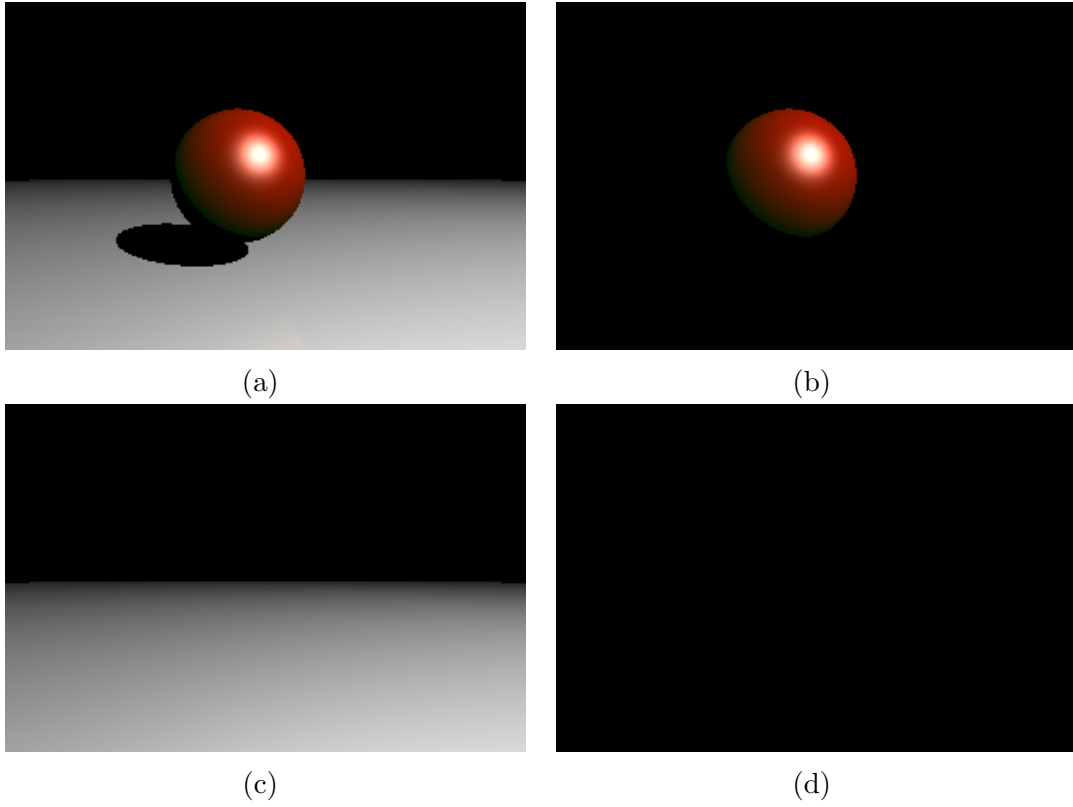


Figura 6.1. Posibles escenas para generación de datos.

Así, se creó un conjunto de datos sintético de 1×10^6 registros de observaciones basados en un escenario hipotético similar al usado por [120] para demostrar la importancia de las probabilidades. El 99 % de los registros todas las características se observaron, mientras que del 1 % restante, en el 70 % de los registros se observaron la iluminación y la esfera, pero no superficie ni sombra, del 30 % restante, en la mitad de las imágenes ninguna característica fue observada, mientras que, en la segunda mitad, solo se percibieron la iluminación y la superficie, mas no la esfera ni la sombra. El algoritmo 3 ilustra el proceso para la construcción del conjunto de datos y la tabla 6.1 muestra las proporciones de cada muestra y da una idea clara de la representación de cada observación.

Algorithm 3: Proceso para la generación del conjunto de datos.

```
1  $DS \leftarrow createDataset()$  ;
2  $DS.columns \leftarrow [A', B', C', Y']$  ;
3  $DS.size \leftarrow 1000000$  ;
4  $idxAllColumnsSample \leftarrow DS.sample(990000)$  ;
5 foreach  $idx$  in  $idxAllColumnsSample$  do
6   foreach  $col$  in  $DS.columns$  do
7      $DS[idx][col] \leftarrow 1$ ;
8  $Filter \leftarrow DS[A] = 0 \ \& \ DS[B] = 0 \ \& \ DS[C] = 0 \ \& \ DS[Y] = 0$ ;
9  $sublistLightSphere \leftarrow DS.sublist(Filter)$ ;
10  $idxLightSphereColumnsSample \leftarrow sublistLightSphere.sample(7000)$ ;
11 foreach  $idx$  in  $idxLightSphereColumnsSample$  do
12    $Ds[idx][A] \leftarrow 1$ ;
13    $Ds[idx][B] \leftarrow 1$ ;
14  $Filter \leftarrow DS[A] = 0 \ \& \ ds[B] = 0 \ \& \ ds[c] = 0 \ \& \ ds[y] = 0$ ;
15  $sublistLightSurface \leftarrow DS.sublist(Filter)$ ;
16  $idxLightSurfaceColumnsSample \leftarrow sublistLightSurface.sample(1500)$ ;
17 foreach  $idx$  in  $idxLightSurfaceColumnsSample$  do
18    $DS[idx][A] \leftarrow 1$ ;
19    $DS[idx][C] \leftarrow 1$ ;
```

Tabla 6.1. Proporciones de cada escena en el conjunto de datos

A (Luz)	B (Esfera)	C (Superficie)	D (Sombra)	Cantidad
1	1	1	1	990000
1	1	0	0	7000
1	0	1	0	1500
0	0	0	0	1500

Modelo Causal Estructural (SCM)

Según [119], un modelo causal estructural es una forma de describir las características relevantes del mundo y cómo interactúan entre sí. Específicamente, este modelo describe cómo la naturaleza asigna valores a las variables de interés. La inferencia causal generalmente requiere conocimientos especializados e hipótesis no comprobables sobre la red causal que vincula el tratamiento, el resultado y otras variables.

Al resumir el conocimiento y las hipótesis de manera intuitiva, los grafos ayudan a aclarar problemas conceptuales y mejorar la comunicación entre investigadores [75]. En consecuencia, se diseñó un modelo causal estructural en el que cada nodo corresponde a una característica observable y todas las aristas apuntan a un único nodo colisionador, "Y". La Figura 2 muestra el modelo causal estructural y la independencia de las variables que lo componen.

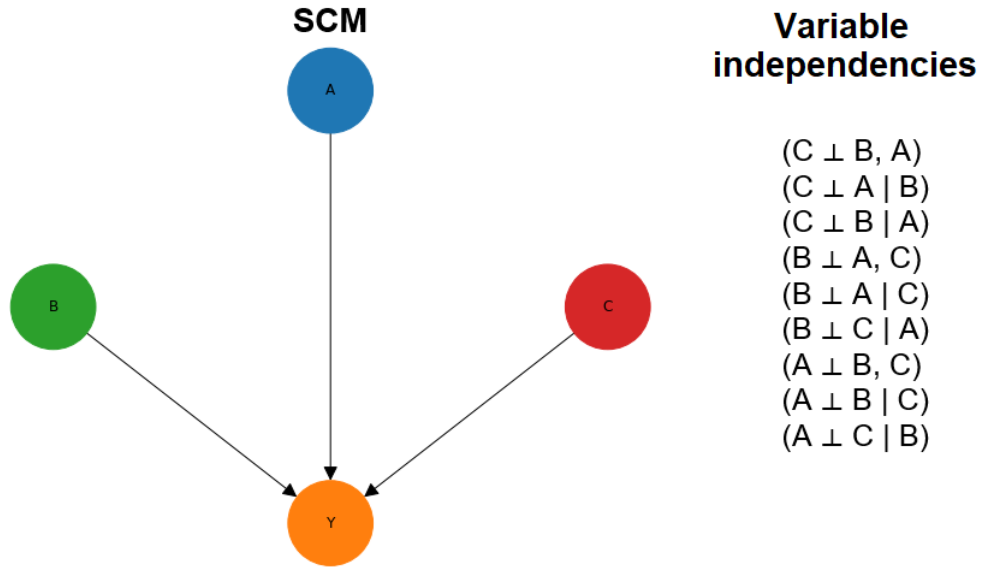


Figura 6.2. Modelo causal estructural e independencia de variables.

Inferencia causal

Una vez que se construye el modelo, se calcularon las distribuciones de probabilidad condicional, las cuales estaban definidas para un conjunto de variables aleatorias discretas y mutuamente dependientes para mostrar las probabilidades condicionales de una variable única con respecto a las demás [50]. Estas se calcularon aplicando la regla de la cadena, como se ilustra en la ecuación (1), donde X_i es un evento y N es el número de eventos considerados en el modelo, por lo tanto, $0 < i < N$.

$$P(x_n) = P(x_1)P(x_2|x_1)P(x_3|x_1, x_2) \dots P(x_n|x_1, x_2, x_3 \dots x_{(n-1)}) \quad (6.1)$$

Como se muestra en la Figura 6.3, se debe calcular la probabilidad de cada valor posible de cada variable conociendo los valores tomados por las otras variables.

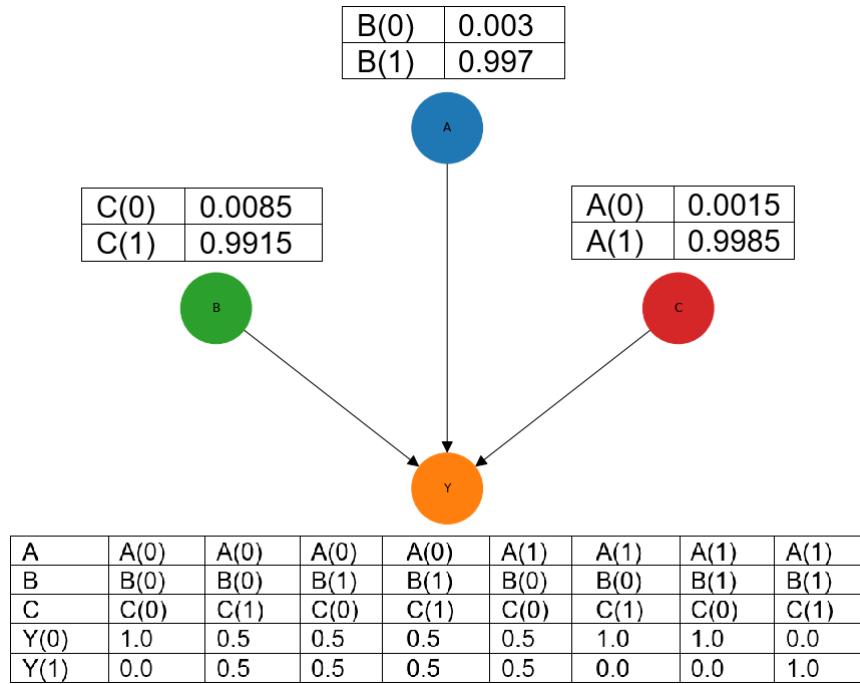


Figura 6.3. Distribución de probabilidad condicional para cada variable del modelo.

Luego, para fortalecer la hipótesis formulada, se consultó al modelo sobre qué sucedería si no se hubiera detectado ninguna esfera, es decir, se hizo una intervención al modelo debido a la no detección de la esfera para obtener la probabilidad de detectar la sombra. Para aclarar qué papel desempeñan las variables del modelo causal estructural en el proceso de inferencia causal que se siguió, entre otros, [121], [122] y [123], quienes explican que un proceso de inferencia causal está determinado por un tratamiento, un conjunto de factores de confusión y un resultado, la tabla 6.2 muestra el papel de cada variable del modelo en el proceso de inferencia causal.

Tabla 6.2. Papel de las variables del modelo estructural en el proceso de inferencia causal.

Etiqueta	Variable
Tratamiento	B
Factores de confusión	[A,C]
Resultado	Y

Posteriormente, considerando esta intervención, para todo el conjunto de casos (N) se calculó el efecto del tratamiento para medir la diferencia promedio entre los casos en los que se aplicó el tratamiento (Y1) y los casos de control (Y2) mediante la aplicación de 6.2.

$$ATE = \frac{1}{N} \sum_{i=1}^N (Y_1(i) - Y_0(i)) \quad (6.2)$$

Finalmente, para contrastar los resultados del tratamiento y así obtener la estimación de qué tan lejos estaba la hipótesis de ser nula, es decir, que no hubiera relación entre la esfera y la sombra, para un intervalo de confianza del 95 %, se generó una tabla de puntuaciones z aplicando

(3), donde X_i es cada resultado, μ es la media y σ es la desviación estándar, y luego se calculó el valor p de cola bilateral y un nivel de significancia $\alpha = 0,05$.

$$Z = \frac{X_i - \mu}{\sigma} \quad (6.3)$$

Así, se tomó una muestra de control de 3000 escenas y una muestra de tratamiento de 997000 escenas para intervenir el modelo con dos consultas de tipo "what if":

1. ¿Qué pasaría si no hubiera habido sombra?
2. ¿Qué pasaría si no hubiera habido luz?

Detección de la sombra

Dentro del ámbito de la detección de sombras en imágenes, se destacan métodos prominentes como el umbral adaptativo, la segmentación por umbral [124], y la segmentación basada en agrupamiento (Felzenszwalb) [125]. Este último método se complementó integrándole un módulo de inferencia causal, lo cual no solo facilitó la detección de sombras, sino que también permitió la identificación de sus causas fundamentales dentro de escenarios específicos. Este enfoque demandó parámetros que abarcaran un registro completo de elementos perceptibles dentro de la imagen, como la luz, la esfera, la superficie y la sombra, junto con el modelo de inferencia causal.

Para ilustrar este proceso, el Algoritmo 4 detalla la intrincada interacción entre la detección de sombras y la inferencia causal dentro del fenómeno de las sombras.

Algorithm 4: Algoritmo para la detección de sombra y su explicación mediante inferencia causal

Data: Image, scale, sigma, min_size, light, sphere, shadow, surface, CI_model

Result: Segmented regions

```

1 Initialize an empty image segment map segments;
2 Initialize an empty priority queue pq for merging;
3 for each pixel p in Image do
4   | Create a new segment for p and add it to pq;
5 end
6 while pq is not empty do
7   | Merge the two smallest segments from pq;
8   | if merge does not violate min_size constraint then
9     |   Add the merged segment to segments;
10    |   Add the merged segment to pq;
11  | end
12 end
13 return CI_model.estimate(light, sphere, shadow, surface)
```

6.3. Resultados

Modelo causal

El modelo causal estructural fue diseñado basado en el conocimiento experto como señaló [75], pero validado en dos intentos mediante el uso del método de descubrimiento causal NOTEARS. En el primer intento, el algoritmo tomó casi 5 minutos para generar el modelo mostrado en la figura 6.4a, lo cual se consideró bastante largo para el tamaño del conjunto de datos, resultando en un modelo que no fue muy coherente según el conocimiento experto. Por otro lado, en el segundo intento, se agregó una restricción al algoritmo para considerar que A, B y C son independientes. El algoritmo, como se puede observar en la figura 6.4b, generó el modelo con la coherencia esperada y en muy corto tiempo (menos de un minuto).

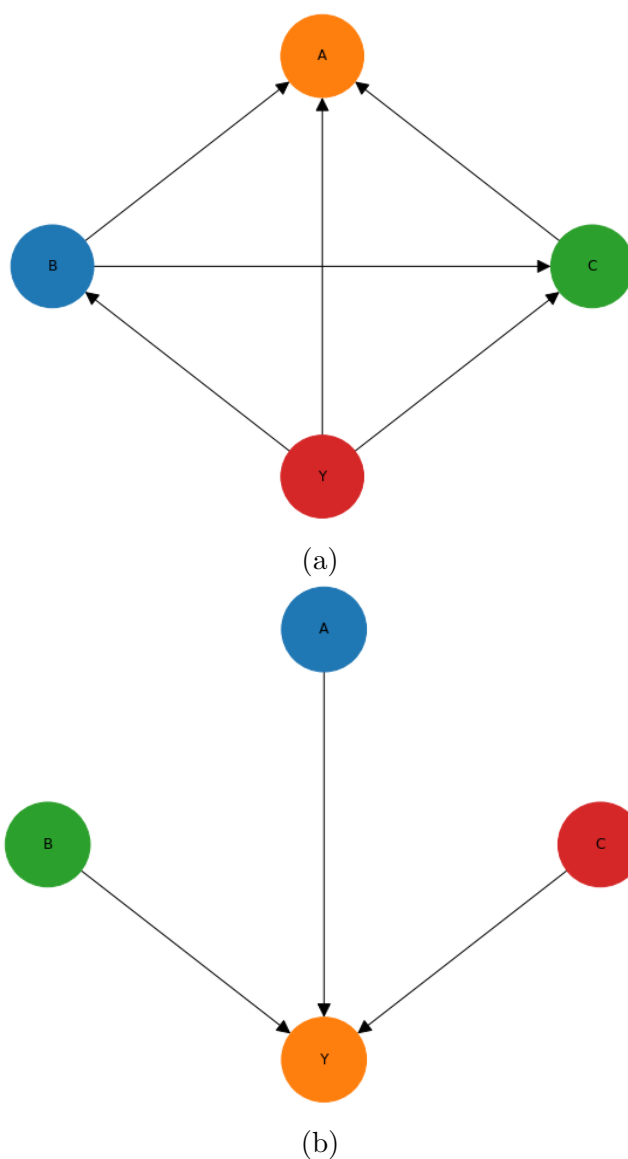


Figura 6.4. Modelos descubiertos por el algoritmo NOTEARS.

Inferencia causal

A partir de la distribución de probabilidad condicional, fue posible consultar el modelo bajo la hipótesis formulada. En la Tabla 6.3 se puede observar que, al eliminar la esfera, habría una probabilidad del 99.5 % de que no se proyectara ninguna sombra; además, se puede ver que la hipótesis adquiere fuerza al obtener un valor de p menor que 0.05 (el valor de umbral predeterminado), lo que indica que la hipótesis nula es falsa, y un valor de efecto promedio de tratamiento positivo (ATE) sugiere que $P(Y|B) > P(Y)$, lo que indica que la presencia de la esfera aumenta la probabilidad de detectar una sombra proyectada en la superficie o, en otras palabras, fue posible inferir que la esfera es la causa más probable de la sombra proyectada en la superficie.

Tabla 6.3. Inferencia causal de la intervención $P(Y \text{ — } \text{do}(B = 0))$

Resultado	Probabilidad	ATE	z	$P > z $	Intervalo de confianza
$Y(0)$	0.995	0.993	11874	0.00001	95 %
$Y(1)$	0.005				

Inferencia causal y detección de sombras

Para establecer un contraste, se utilizó una imagen idéntica y se introdujo un elemento de confusión al alinear el color de fondo con la tonalidad de la sombra proyectada sobre la superficie. La ejecución posterior abarcó el método de Felzenszwalb integrado con el módulo de inferencia causal, así como las técnicas de umbral adaptativo y segmentación de umbral. Los resultados de esta aproximación se visualizan en la Figura 6.5.

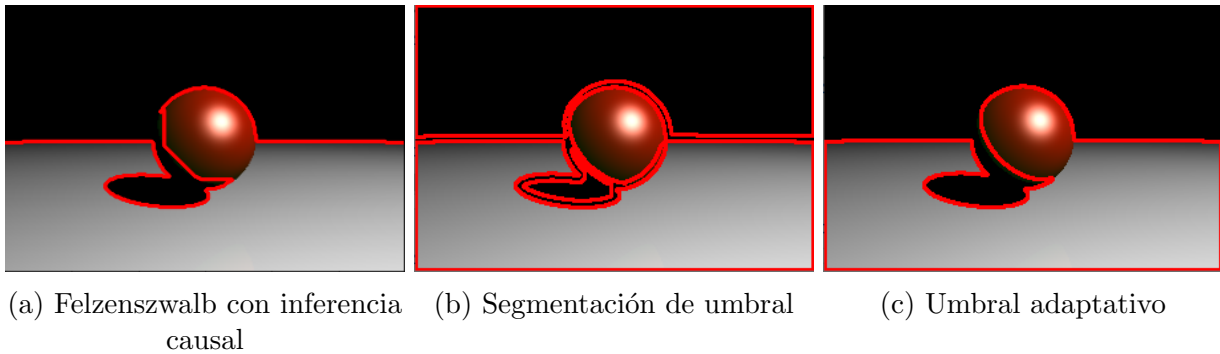


Figura 6.5. Resultados de la detección de sombras.

En el contexto de la detección de sombras, los resultados son evidentes. Entre los enfoques, la combinación del método de Felzenszwalb con inferencia causal (Figura 6.5a) mostró los resultados más prometedores. Logró una precisión aceptable en la detección de sombras del 51,5 %. Siguiendo de cerca, el método de umbral adaptativo alcanzó una precisión del 51,7 % (Figura 6.5c), mientras que el método de segmentación de umbrales alcanzó una precisión del 55,9 % (Figura 6.5b).

Es importante destacar que la presencia de factores de confusión influyó significativamente en la precisión de los resultados de detección. Sin embargo, al considerar la determinación de

la causalidad de la sombra, el impacto de los factores de confusión se volvió insignificante. Especialmente, solo el método de Felzenszwalb (Figura 6.5a) arrojó un resultado sustancial en este sentido.

6.4. Conclusiones

Se demostró cómo emplear la inferencia causal para fortalecer una hipótesis y, como resultado, deducir la causa de un fenómeno de sombra con alta certeza. Esto se logró utilizando intervenciones e investigaciones dentro del modelo causal.

Se comenzó con un conjunto de imágenes de un escenario 3D en el que se examinaron cuatro eventos como parte de un modelo causal estructural validado con el algoritmo NOTEARS para detección causal. Al contrastar su desempeño, también se demostró que agregar un módulo de inferencia causal a un enfoque de detección de sombras es factible y ventajoso. Esto abre la puerta a conexiones similares en otras formas diversas y complejas.

La representación visual de un modelo causal mejora la comprensión del problema y los roles que desempeñan los eventos en su resolución. A pesar de probar el modelo causal con el algoritmo NOTEARS, se encontró cierta preocupación sobre la necesidad de establecer límites basados en el conocimiento experto.

Se requiere un conjunto de datos con una estructura más intrincada para la aplicación de la inferencia causal, en comparación con los conjuntos de datos típicos utilizados para aplicaciones de aprendizaje automático, en los que predomina la correlación entre las características.

Los factores de confusión tuvieron un impacto considerable en la precisión del método de detección, pero no en el modelo de inferencia causal.

Inferencia causal para el modelo de comportamiento de un agente inteligente

7.1. Objetivo

El objetivo de este experimento fue desarrollar un agente inteligente que empleara la inferencia causal para comprender las causas detrás de los fallos de otros agentes al completar una tarea. Esta comprensión permitiría al agente identificar los factores clave que llevan al fracaso y, en consecuencia, mejorar su capacidad para abordar la tarea de manera más efectiva. Luego, este agente fue diseñado para ejecutar la tarea tomando decisiones informadas basadas en la inferencia causal, utilizando esta comprensión para anticipar posibles obstáculos y tomar medidas correctivas apropiadas. El objetivo final era aumentar la probabilidad de completar la tarea con éxito al minimizar los errores y maximizar la eficiencia del agente durante la ejecución de la tarea.

7.2. Método

Diseño de la tarea

La tarea implicaba que un agente inteligente navegara a través de un entorno virtual, con el objetivo de alcanzar la posición ocupada por otro agente dentro del mismo entorno sin ser detectado. Las etapas del experimento incluyeron el diseño y configuración del entorno virtual, el desarrollo de los agentes explorador (EBOT) y guardián (GBOT), la recolección de datos y la implementación del agente causal (CBOT).

Diseño del escenario y configuración del entorno virtual

Para los experimentos, se utilizó Unreal Tournament, un videojuego de tipo First-Person Shooter que permite a los desarrolladores configurar entornos y programar la lógica de los bots antes de desplegarlos en el mundo virtual. En Unreal Editor [126] se diseñó el escenario de pruebas, que consiste en una serie de corredores interconectados, como se muestra en la figura 7.1, para garantizar el acceso a cualquier punto de interés en el mapa a través de siete rutas distintas:

- Path: S-A-C-E-H-K-T
- Path: S-B-D-E-H-K-T
- Path: S-A-C-F-H-K-T
- Path: S-B-D-G-H-K-T
- Path: S-A-C-F-I-K-T
- Path: S-B-D-G-J-K-T
- Path: S-B-D-G-J-M-N-T

Estas rutas fueron cuidadosamente diseñadas para garantizar el acceso a varias áreas de interés en el entorno virtual.

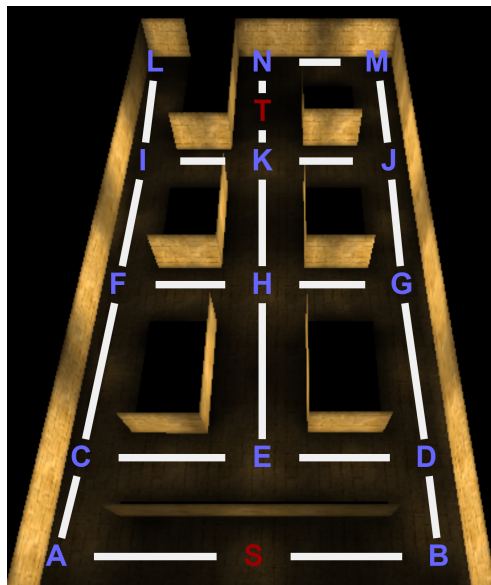


Figura 7.1. Rutas configuradas en el entorno virtual.

Pogamut es un componente crucial que garantiza una interacción fluida entre los agentes dentro del entorno virtual. Sirve como una interfaz intermedia que permite controlar a los agentes virtuales en diferentes motores de juego y entornos. Este proporciona una API de Java para generar y gestionar agentes virtuales, y una interfaz gráfica fácil de usar para facilitar la depuración [127]. El modelo arquitectónico que soporta la integración del CBOT con el entorno virtual a través de Pogamut se ilustra en la Figura 7.2.

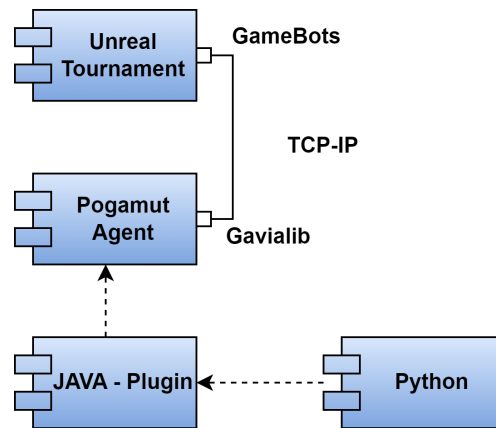


Figura 7.2. Modelo Arquitectónico para la ejecución del CBOT.

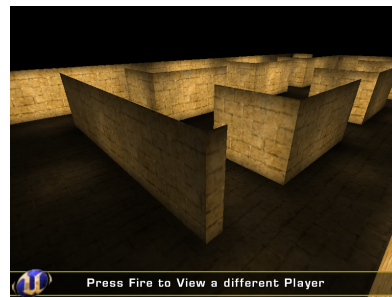
Después de instalar el middleware, se configuró el script para lanzar el servidor DeathMatch y cargar el escenario de prueba diseñado específicamente para la tarea. Luego, para desarrollar los modelos de comportamiento del agente, se utilizó la herramienta Eclipse IDE y la plantilla EmptyBot predefinida gestionada a través de un repositorio Maven. El proceso de ejecución de una partida se muestra en la Figura 7.3.

```

C:\WINDOWS\system32\cmd.exe
C:\UT2004\System>ucc server Test_Environment?game=GameBots2004.BotScen
arioMutator=GameBots2004.GBScenarioMutator,GameBots2004.GBHJD, GameB
ots2004.PathMarkerTimeLimit=999999
Executing Class Engine.ServerCommandlet
Browse: Test_Environment?Name=El_mago_vago?Class=Engine.Pawn?Character
?ClassName=255?PawnID=GameBots2004.BotScenario?Mutator=GameBots2
004.GBScenarioMutator,GameBots2004.GBHJD,
Collecting garbage
Purging garbage
Garbage: objects: 16387->16386; refs: 148397
Game class is 'BotScenario'
Bringing Level Test_Environment.myLevel up for play (20) appSeconds: 2
.513800...
(Karma): Autodetecting CPU for SSE
(Karma): Using SSE Optimizations
Mutators: GameBots2004.GBScenarioMutator,GameBots2004.GBHJD,
GameInfo::InitGame : bEnableStatLogging False
UpServerQuery(crc): Port 7787 successfully bound.
Resolving master0.gamespy.com...

```

(a)



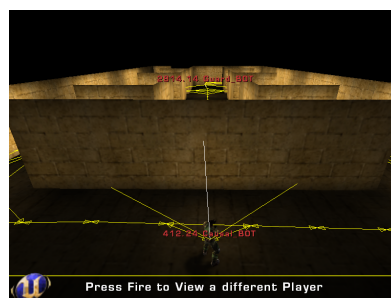
(c)



(b)



(d)



(e)

Figura 7.3. Proceso de ejecución de una partida. (a) Consola. (b) Servidor. (c) Escenario vacío. (d) Vista GBOT. (e) Vista CBOT.

La secuencia de acciones durante la ejecución de la partida se resumió en los siguientes pasos: primero, se ejecutó el servidor responsable de cargar el entorno de prueba designado (Figura 7.3a). Luego, UT2004 se conectó a este servidor (Figura 7.3b), lo que resultó en la aparición del escenario de prueba vacío (Figura 7.3c). Posteriormente, desde Eclipse, se lanzaron ambos agentes (GBOT y EBOT) en el entorno virtual. Las figuras 7.3d y 7.3e muestran el punto de vista de cada agente, respectivamente.

Utilizando Pogamut versión 3.7.0 y Eclipse[®] versión 4.29.0 (Build 20230907-1323), se desarrolló una plataforma para diseñar, probar e implementar los comportamientos de cada uno de los agentes y su interacción con el entorno. Estas herramientas fueron fundamentales para la ejecución y prueba de agentes inteligentes dentro de un entorno experimental controlado y reproducible.

Desarrollo de los agentes explorador (EBOT) y guardián (GBOT)

Las características de los agentes fueron heredadas de los avatares que interactúan en el videojuego Unreal Tournament [128]. Sin embargo, se programaron completamente sus modelos de comportamiento utilizando la API de Pogamut.

Agente guardian (GBOT)

La tarea principal del agente Guardián era eliminar cualquier otro agente que identificara acercándose a su posición, siguiendo una alerta del sistema de mensajería global del entorno virtual.

Modelo de comportamiento

El agente Guardian fue diseñado para mantener una posición fija en el entorno, con la posibilidad de activar o desactivar sus sensores visual y auditivo. Cuando el sensor visual estaba activo, y un agente enemigo entraba en su campo de visión, el agente guardian lo identificaba con éxito. Por otro lado, si el sensor visual estaba inactivo, pero el sensor auditivo estaba activo, el agente detectaba los pasos que se acercaban y giraba en dirección a la fuente del sonido, pero no identificaba al agente que producía el sonido. Es decir, detectaba pero no identificaba. Sin embargo, cuando ambos sensores estaban activos, el agente Guardian detectaba e identificaba eficazmente al agente que se acercaba, sin importar la dirección. La lógica del GBOT se ilustra claramente en el Algoritmo 5.

Algorithm 5: Modelo de comportamiento del GBOT

Input : agent's configuration parameters: canSee, canListen

Output: None

```
1 while true do
2   if canSee then
3     player ← game.players.getNearestPlayer();
4     if player is not null then
5       sendMessage(player, "you've been identified.");
6       game.kick(player);
7   if canListen then
8     if isHearingNoise then
9       turnTo(getNoiseSource());
```

Agente Explorador (EBOT)

El objetivo principal del agente explorador consistió en navegar por una ruta que lo llevara a tomar la posición ocupada por el agente GBOT. El entorno ofrecía diversas rutas, lo que permitía al agente Explorador acercarse al agente GBOT desde diferentes direcciones, ya sea desde el frente, por ambos costados o desde la parte trasera.

Modelo de comportamiento

Se crearon dos versiones de este agente. La primera versión, que se utilizó para generar el conjunto de datos, empleó un algoritmo de búsqueda no informada. Este agente siempre comenzaba en un punto fijo y elegía aleatoriamente una de las rutas disponibles que lo llevaría a la ubicación del agente guardian. La segunda versión, que se utilizó para evaluar el rendimiento del agente causal, utilizaba el algoritmo de búsqueda informada A* por su utilidad inherente para solucionar problemas de búsqueda de rutas. La ejecución del agente terminaba de dos maneras: si el agente GBOT identificaba al agente explorador o si el agente explorador tomaba con éxito la ubicación del agente GBOT. Antes de concluir su ejecución, el agente explorador registraba sus datos de rendimiento, que se muestran en la tabla 7.1. Tanto el EBOT de búsqueda no informada como el EBOT de búsqueda informada siguen la lógica descrita en los algoritmos 6 y 7, respectivamente.

Tabla 7.1. Estructura del conjunto de datos.

Etiqueta	Valor	Descripción
BOT	[0. . . 1199]	Número consecutivo que identifica a cada EBOT
C	[1. . . 7]	Camino elegido por el EBOT
T_Nodes	[1. . . path_length]	Longitud del camino
T_Visited	[1. . . path_length]	Cantidad de nodos visitados por EBOT
Can_see	[0,1]	¿El GBOT puede ver?
Can_listen	[0,1]	¿El GBOT puede escuchar?
Outcome	[0,1]	¿El EBOT completó la tarea?

Algorithm 6: EBOT- Búsqueda no informada

Input : Map of the game: *game.map*

Output: None

```
1  $g \leftarrow \text{generateGraph}(\text{game.map});$ 
2  $\text{lstPaths} \leftarrow \text{getAllPaths}(g, S, T);$ 
3  $(\text{path}, \text{pathId}) \leftarrow \text{chooseRandomPath}(\text{lstPaths});$ 
4 while true do
5      $\text{visited} \leftarrow 0;$ 
6     foreach node in path do
7          $\text{agent.moveTo}(\text{node});$ 
8          $\text{visited} \leftarrow \text{visited} + 1;$ 
9         if getMessage() is "you've been identified." then
10              $\text{id} = \text{game.agentId};$ 
11              $\text{pathId} = \text{pathId};$ 
12              $\text{length} = \text{path.length};$ 
13              $\text{canListen} = \text{game.GBOT.canListen}();$ 
14              $\text{canSee} = \text{game.GBOT.canSee}();$ 
15              $\text{saveData}(\text{id}, \text{pathId}, \text{length}, \text{visited}, \text{canListen}, \text{canSee}, 1);$ 
16             return;
17         if node is T then
18              $\text{id} = \text{game.agentId};$ 
19              $\text{pathId} = \text{pathId};$ 
20              $\text{length} = \text{path.length};$ 
21              $\text{canListen} = \text{game.GBOT.canListen}();$ 
22              $\text{canSee} = \text{game.GBOT.canSee}();$ 
23              $\text{saveData}(\text{id}, \text{pathId}, \text{length}, \text{visited}, \text{canListen}, \text{canSee}, 0);$ 
24             return;
```

Algorithm 7: EBOT - Búsqueda informada

Data: start, goal

Result: Camino más corto desde start hasta goal

```
1 Function Astar(start, goal):
2   openSet  $\leftarrow$  {start};
3   cameFrom  $\leftarrow$  un mapa vacío;
4   gScore[start]  $\leftarrow$  0;
5   fScore[start]  $\leftarrow$  Heuristic(start, goal);
6   while openSet no está vacío do
7     current  $\leftarrow$  el nodo en openSet con el valor de fScore más bajo;
8     if current es goal then
9       | return ReconstructPath(cameFrom, current);
10    end
11    openSet.remove(current);
12    foreach neighbor de current do
13      tentativeGScore  $\leftarrow$  gScore[current] + Distance(current, neighbor);
14      if tentativeGScore < gScore[neighbor] then
15        | cameFrom[neighbor]  $\leftarrow$  current;
16        | gScore[neighbor]  $\leftarrow$  tentativeGScore;
17        | fScore[neighbor]  $\leftarrow$  gScore[neighbor] + Heuristic(neighbor, goal);
18        | if neighbor not in openSet then
19          | openSet.add(neighbor);
20        | end
21      end
22    end
23  end
24  return emptySet
25 Function Heuristic(node, goal):
26 | return | node.x - goal.x | + | node.y - goal.y | + penalty;
```

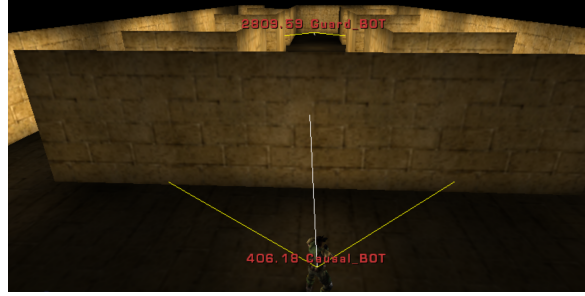
La complejidad temporal del algoritmo A* depende tanto de la heurística elegida como de la estructura subyacente del grafo. El mejor escenario, definido por una heurística consistente, tiene una complejidad computacional de $O(d)$, donde d es la longitud del camino más corto, mientras que la naturaleza exponencial del peor escenario puede atribuirse a la presencia de heurísticas ineficaces o grafos complejos. La complejidad espacial de un grafo se determina por su tamaño, así como por el número de nodos abiertos y cerrados. En circunstancias desfavorables, el costo computacional también puede ser sustancial debido al espacio de almacenamiento necesario para retener datos sobre los nodos recorridos.

Recolección de los datos

Para recopilar los datos necesarios, Se realizaron 1200 partidas en el entorno de prueba, teniendo en cuenta la complejidad de las tareas asignadas a los agentes GBOT y EBOT. Cada partida siguió la secuencia de lanzar primero el agente GBOT, seguido por el EBOT. Al final de

cada partida, el agente EBOT registró y almacenó los datos relevantes.

La figura 7.4a muestra las posiciones iniciales de los agentes en el entorno virtual al comienzo de una partida, y la figura 7.4b proporciona una instantánea del conjunto de datos generado durante una de estas partidas.



(a)

BOT	C	T-Nodes	T-Visited	Can_See	Can_Listen	Outcome
0	5	25	12	1	0	0
1	5	25	25	0	1	1
2	2	27	13	1	0	0
3	2	27	13	1	0	0
4	4	25	16	1	0	0
...
1195	3	24	24	0	1	1
1196	7	30	30	0	1	1
1197	2	27	13	1	0	0
1198	3	24	21	1	1	0
1199	7	30	26	1	1	0

(b)

Figura 7.4. Ejemplo de una partida y los datos generados. (a) Ubicación inicial de los agentes. (b) Dataset generado con las variables de una partida.

Implementación del agente causal (CBOT)

El papel del agente CBOT es idéntico al del agente EBOT, pero su proceso está guiado por inferencia causal antes de entrar al entorno virtual.

Inferencia Causal

El proceso de inferencia causal comenzó cargando los datos de las 1200 partidas. Se utilizó un modelo causal estructural para estimar la distribución conjunta del conjunto de datos [129]. Posteriormente, diseñamos el modelo causal, ilustrado en la figura 7.5, para proporcionar información adicional en la toma de decisiones del agente CBOT durante su ejecución.

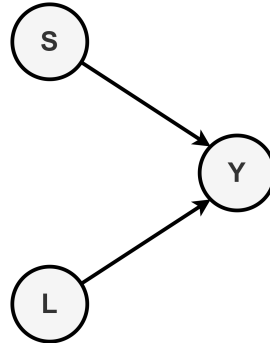


Figura 7.5. Modelo causal estructural.

En este modelo, se consideraron dos variables explicativas: la del sensor visual (S) y la del sensor auditivo (L). El resultado (Y) se determinó en función del estado de estos dos sensores, sin descartar que inicialmente el CBOT tenía formulada la hipótesis de que la activación de uno de estos sentidos influía en el éxito de la tarea. Para probar la validez de esta creencia, al consultar el modelo se predijo el efecto de intervenciones específicas en las relaciones causales, siguiendo el enfoque propuesto por [120].

Distribución de Probabilidad Condicional

Una distribución de probabilidad condicional es una tabla que muestra las probabilidades de una variable con respecto a otras [50]. Según [130], estas estructuras permiten realizar consultas al modelo y razonar sobre los contrafácticos utilizando la ecuación 7.1.

$$P(A \mid B) = P(A \cap B) / P(B) \quad (7.1)$$

Donde:

- $P(A \mid B)$: Representa la probabilidad de que ocurra el evento A dado que el evento B ya ha ocurrido.
- $P(A \cap B)$: Es la probabilidad de que ocurran simultáneamente los eventos A y B.
- $P(B)$: Es la probabilidad de que ocurra el evento B, sin condiciones adicionales.

Se utilizó el paquete pgmpy [131] para generar la distribución de probabilidad condicional correspondiente y se realizaron las siguientes consultas:

- ¿Qué sucedería si el GBOT pudiera ver y escuchar al mismo tiempo?
- ¿Qué sucedería si el GBOT pudiera ver pero no escuchar?
- ¿Qué sucedería si el GBOT pudiera escuchar pero no ver?

Para validar los resultados de estas consultas "qué pasaría si" [75], se dividió la muestra en un grupo de control y un grupo de tratamiento utilizando el paquete causalnex [132]. Esto permitió medir el impacto de cada intervención, como activar o desactivar un sensor, a través del efecto promedio del tratamiento (Average Treatment Effect, ecuación 7.2). Al analizar esta

medida, pudimos determinar la precisión de la creencia del CBOT en su proceso de toma de decisiones.

$$ATE = \frac{1}{N} \underbrace{\sum_{i=1}^N (Y_1(i))}_{\text{Resultado total con tratamiento}} - \underbrace{\sum_{i=1}^N (Y_0(i))}_{\text{Resultado total sin tratamiento}} \quad (7.2)$$

Donde:

- N : Número de muestras.
- $Y_0(i)$: Resultado sin tratamiento para la muestra i .
- $Y_1(i)$: Resultado con tratamiento para la muestra i .
- i : Índice que representa muestras individuales.

Modelo de comportamiento

Después de llevar a cabo el proceso de inferencia causal, el CBOT obtuvo una comprensión de la causa subyacente del fracaso basada en su creencia sobre el logro de la tarea. Siguiendo un proceso lógico similar al del EBOT, donde el camino se elige al azar, el CBOT ejecuta la ruta con la mayor probabilidad de éxito.

7.3. Resultados

En el contexto de este experimento, se examinó la evaluación del rendimiento de agentes en un entorno virtual donde el objetivo principal era que los agentes EBOT se desplazaran desde un punto inicial hasta la posición ocupada por un agente guardián GBOT sin ser detectados por este, cuyo sentido de visión y audición pueden estar activados o desactivados a demanda.

Dado que esta configuración única no ha sido ampliamente explorada en la literatura académica consultada, las comparaciones directas con estudios anteriores han resultado difíciles debido a la falta de estudios completamente análogos. Como resultado, para situar nuestros hallazgos en el contexto más amplio de los métodos de toma de decisiones, llevamos a cabo una evaluación considerando métricas clave como la tasa de éxito de la tarea, el consumo de energía de los agentes y la relación entre el consumo de energía de estos y sus tasas de éxito y fracaso. Si bien estudios anteriores se han centrado en diferentes enfoques y contextos, el análisis comparativo que se realizó hizo posible discutir las implicaciones de estos hallazgos en el contexto de la toma de decisiones mediada por la inteligencia artificial.

Inferencia causal

Al calcular la distribución de probabilidad condicional, como se muestra en la tabla 7.2, fue posible consultar el modelo causal para responder preguntas contrafácticas. Esto permitió determinar la probabilidad de éxito ($Y = 1$) o fracaso ($Y = 0$) para cada combinación de estados o posibles valores de las variables explicativas (L, S).

Tabla 7.2. Distribucion de probabilidad condicional del modelo causal estructural.

L S	L(0) S(0)	L(0) S(1)	L(1) S(0)	L(1) S(1)
Y(0)	0.5	0.8204	0.0	1.0
Y(1)	0.5	0.1795	1.0	0.0

El CBOT creía que el éxito en la ejecución de la tarea estaba determinado por uno de los sensores del GBOT, lo que requería intervenciones en ambos sensores para reforzar esta creencia. La Figura 7.6 ilustra los efectos de las intervenciones resultantes de las consultas al modelo, con el objetivo de deducir la causa del fracaso en la tarea.

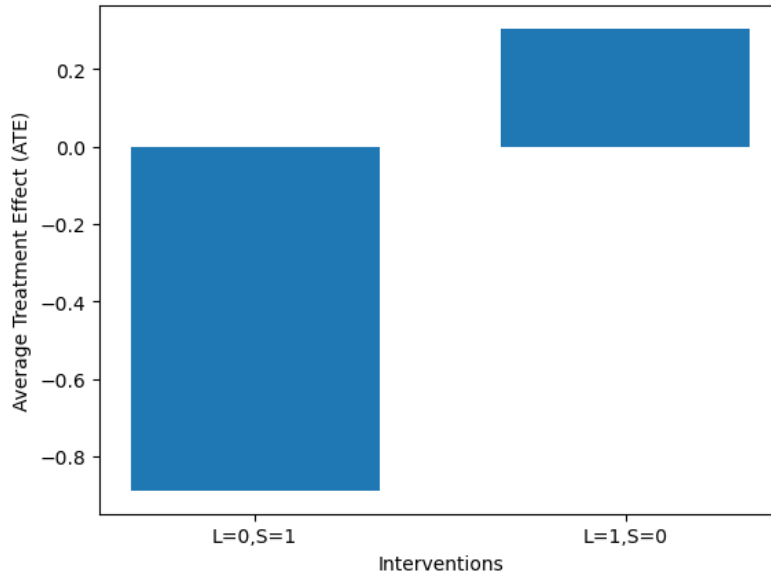


Figura 7.6. Efecto de las intervenciones.

Si el GBOT podía ver y escuchar ($S = 1, L = 1$), la probabilidad de fracaso ($Y = 0$) expresada como $P(Y = 0|S = 1, L = 1) = 1,0$ fue absoluta, lo que implicaba que no había posibilidad de lograr la tarea. A pesar de reforzar la creencia del CBOT, no fue posible identificar qué sensor determinó el éxito de la tarea.

Por otro lado, si el GBOT podía ver pero no escuchar ($S = 1, L = 0$), la probabilidad de fracaso ($Y = 0$) expresada como $P(Y = 1|S = 1, L = 0) = 0,8204$ fue del 82 % independientemente de la ruta tomada, de manera que activar el sensor de visión ($S = 1$) tuvo un impacto desfavorable en la variable de interés (Y) con un efecto de $ATE = -0,888$. La creencia del CBOT se reforzó, lo que permitió hacer una primera inferencia con considerable certeza sobre qué sensor determinó el éxito de la tarea.

Finalmente, si el GBOT no podía ver pero podía escuchar ($S = 0, L = 1$), la probabilidad de fracaso ($Y = 0$) expresada como $P(Y = 1|S = 0, L = 1) = 0,0$ fue nula, independientemente de la ruta tomada, de manera que, en este caso, activar el sensor auditivo ($L = 1$) tuvo un

impacto favorable en la variable de interés (Y) con un efecto de $ATE = 0,304$, siempre y cuando el sensor de visión permaneciera inactivo ($S = 0$). Esto reforzó la creencia de que el sensor de visión determinó el éxito de la tarea.

Evaluación del rendimiento de CBOT

El rendimiento de CBOT se comparó con dos EBOTs: uno que utiliza un algoritmo de búsqueda a ciegas (sin información) y otro un algoritmo de búsqueda informada (A^*). La Tabla 7.3 presenta los resultados de 250 partidas en el mismo escenario de prueba.

Tabla 7.3. Resultado de evaluación del desempeño.

Agente	Lógica	Aciertos	Fallos	Tasa de acierto	Tasa de fallos	Energía
EBOT-US ^a	Random search	75	175	30	70	76.39
EBOT-IS ^b	A^*	198	52	79.2	20.8	37.7
CBOT	Causal inference	205	45	82	18	35.4

^a EBOT con búsqueda no informada. ^b EBOT con búsqueda informada.

Como se esperaba, el EBOT con el algoritmo de búsqueda ciega tuvo un rendimiento deficiente en comparación con los demás en la Figura 7.7. El CBOT superó por poco al EBOT con A^* , principalmente porque el cálculo heurístico de este último incluía una penalización por ser detectado mientras calculaba la ruta, considerándolo un fracaso. Sin embargo, el comportamiento de este EBOT se basó en un cálculo heurístico que no tenía en cuenta elementos causales.

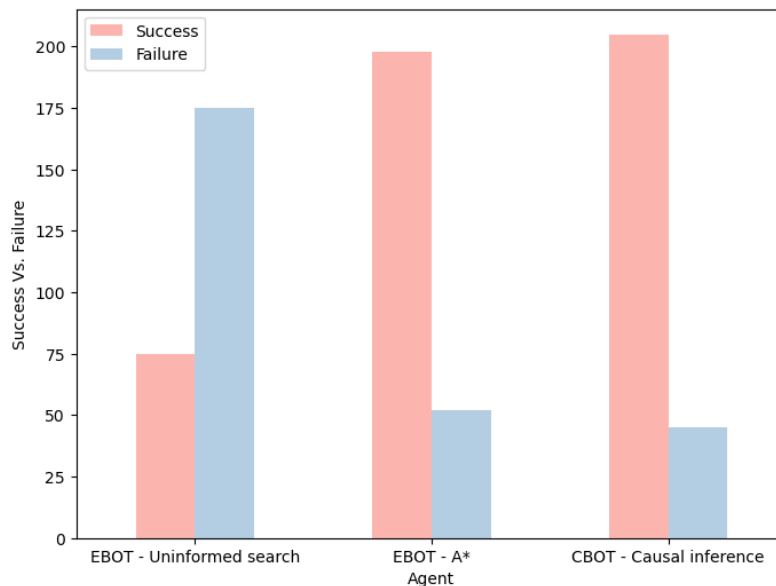


Figura 7.7. Resultados de validación.

La Figura 7.8 muestra un análisis de la tasa de éxito de los agentes en relación con el estado del sensor de visión del GBOT. Se observa una diferencia significativa entre los casos en los que el sensor de visión está activado y los casos en los que está desactivado. La capacidad de evadir

la detección es considerablemente más efectiva cuando el agente enemigo no puede utilizar su sentido de la vista para rastrearlos, lo cual sucede cuando el sensor de visión está desactivado.

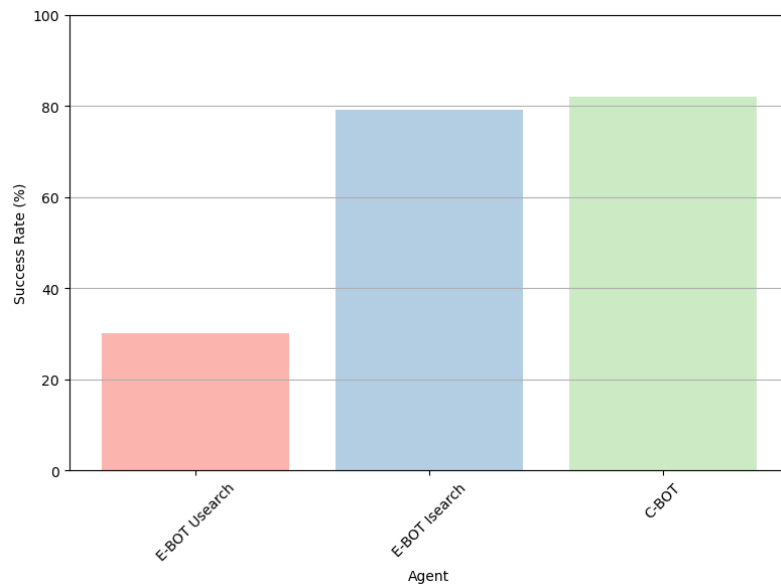


Figura 7.8. Tasa de éxito según el estado del sensor de vista.

La figura 7.9 presenta un análisis de la relación entre el consumo de energía de los agentes y el estado del sensor de visión del agente enemigo. Es evidente que los agentes que operan en un entorno donde el sensor de visión está desactivado consumen menos energía que aquellos que operan en entornos donde el sensor de visión está activado. Este hallazgo ilustra cómo la capacidad de evadir la detección visual puede afectar la eficiencia con la que los agentes utilizan sus recursos energéticos.

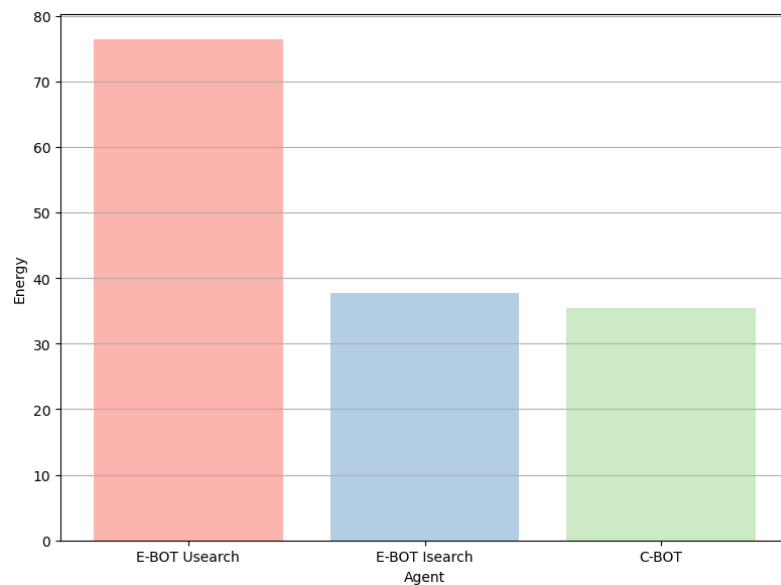


Figura 7.9. Consumo de energía según el estado del sensor de vista.

La figura 7.10 ilustra la relación entre el consumo de energía y la detección por parte de GBOT. Como se puede observar, el consumo de energía generalmente es menor en los casos en los que los agentes no son detectados. Sin embargo, esta relación presenta algunos valores atípicos que sugieren cierto grado de variación. Por lo tanto, aunque existe una correlación entre la detección y el consumo de energía, otros factores, como la presencia de un sensor de escucha, también podrían afectar este factor.

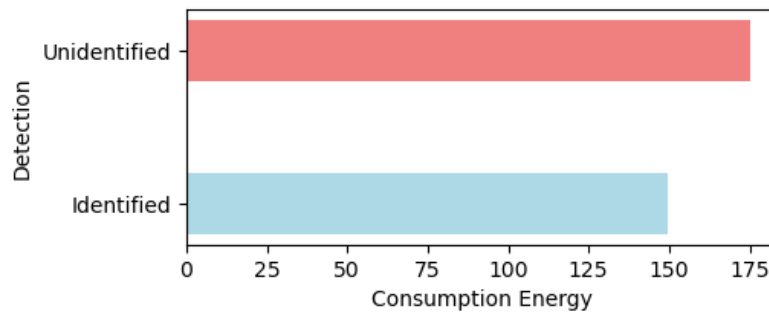


Figura 7.10. Consumo de energía Vs. Identificación de los EBOT.

7.4. Conclusiones

En este experimento, se exploró la convergencia entre agentes inteligentes e inferencia causal. Los hallazgos proporcionaron una nueva perspectiva sobre los efectos transformadores de incorporar el razonamiento causal en la ejecución de tareas y la toma de decisiones, otorgando así una ventaja competitiva a los agentes.

La aplicación de la inferencia causal en el CBOT mostró una ventaja significativa sobre las metodologías alternativas. Proporcionó una comprensión profunda de las causas detrás de los eventos de detección, permitiendo al CBOT tomar decisiones inteligentes y diseñar planes que optimizaran la finalización de tareas. En contraste, los agentes EBOT se basaban únicamente en la selección aleatoria de rutas, lo que requería una comprensión exhaustiva de los factores influyentes. Del mismo modo, el GBOT se enfocaba en la eliminación de agentes sin considerar explícitamente las relaciones causales.

Este experimento resaltó la importancia de las distribuciones de probabilidad condicional para comprender las posibles eventualidades y validar hipótesis. Sin embargo, también puso de manifiesto la naturaleza contextual de estos métodos, como lo destacado por [133], subrayando la importancia de las técnicas de razonamiento para los agentes inteligentes que operan en entornos de simulación física diversos. Además, evidenció el potencial de agregar la inferencia causal a agentes inteligentes, especialmente en situaciones donde los sensores pueden indicar qué comportamientos son causados, lo que facilitaría la toma de decisiones más acertadas.

Incorporación de la inferencia causal en el procedimiento de muestreo de una Deep Q-Network para resolver problemas de control clásico

8.1. Objetivo

El objetivo de este experimento fue desarrollar una Deep Q-Network que integrara un método de muestreo basado en la inferencia causal. Esta red neuronal tenía como propósito aprender a equilibrar de manera efectiva un poste vertical sobre un carrito móvil mediante un conjunto limitado de operaciones, específicamente movimientos hacia la izquierda o derecha. A través de un diseño experimental, se evaluó cómo el uso del muestreo basado en la inferencia causal afecta el rendimiento de la Deep Q-Network en comparación con el método tradicional de muestreo aleatorio que se hace sobre el replay buffer del agente.

8.2. Método

Diseño de la tarea

El entorno CartPole provisto por OpenAI Gym, conocido como un ejemplo clásico en el campo del aprendizaje por refuerzo, fue seleccionado como el escenario experimental para este experimento [134]. La tarea del CartPole consiste en mantener el equilibrio vertical de un poste mediante dos acciones diferentes: moviendo el carro hacia la izquierda o hacia la derecha.

La idea fue implementar una Deep Q-Network que permitiera a un agente el aprendizaje de una política que solucionara dicha tarea; así, esta implementación se llevó a cabo utilizando el framework PyTorch [135], mientras que el procedimiento de muestreo basado en la inferencia causal fue desarrollado utilizando los módulos scikit-learn [136] y SciPy [137].

Deep Q-Network (DQN)

La red neuronal profunda utilizada en este estudio, como se ilustra en la figura 8.1, siguió una arquitectura convencional para el aprendizaje profundo por refuerzo. La capa de entrada recibió observaciones del entorno. Se estableció una capa oculta compuesta por 128 neuronas, cada una de las cuales utilizó la función de activación Rectified Linear Unit (ReLU) que por sus características resulta ser útil en neuronas de capas intermedias. Además, se creó una capa de salida que contenía dos neuronas para cada acción en la tarea de CartPole.

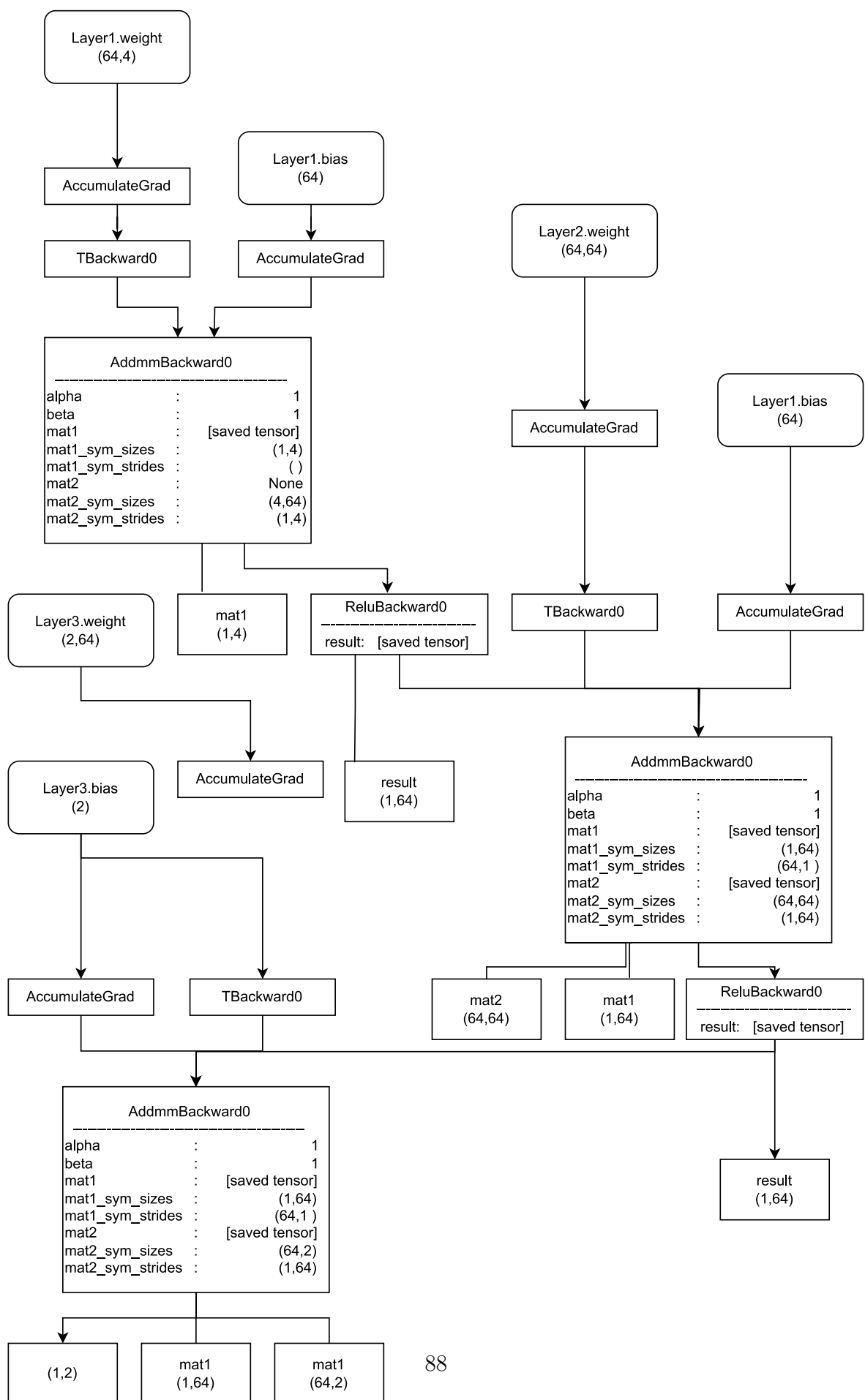


Figura 8.1. Arquitectura de la DQN

Entrenamiento de la Deep Q-Network

El procedimiento de entrenamiento de la Deep Q-Network siguió metodologías bien establecidas en el campo del aprendizaje por refuerzo, tomando como referencia el algoritmo de Q-learning como marco fundamental [138]. A lo largo del procedimiento de entrenamiento, la Deep Q-Network se integró en un patrón recurrente de episodios en los cuales interactuaba con el entorno. Según la información proporcionada en la publicación de [139], el agente en cuestión elegiría una acción de un conjunto predefinido de acciones durante cada episodio. Esta acción seleccionada luego se aplicaría al entorno de prueba, lo que resultaría en alteraciones tanto en la recompensa que obtenía el agente, como en el conjunto de acciones (transiciones) que había ejecutado durante su fase de entrenamiento.

Para asegurar una exploración efectiva y un equilibrio entre la exploración y la explotación, se implementó un mecanismo de exploración-explotación basado en la política epsilon-greedy según lo recomendado por [140]. Esto implicó que para cada iteración, el agente seleccionaría una acción mediante uno de dos enfoques potenciales: exploración o explotación. El proceso de exploración implica seleccionar acciones al azar con una probabilidad denotada como ϵ . Por el contrario, la explotación implica seleccionar la acción con el valor "Q" más alto basado en la política actual, con una probabilidad de $1 - \epsilon$.

Lo novedoso de este enfoque de entrenamiento radicó en cómo se manejó el replay buffer. El replay buffer es una parte clave del aprendizaje profundo por refuerzo, que ayuda a aprender mediante el uso de transiciones de muestra [141]. La utilización de un replay buffer facilitó el almacenamiento y la recuperación de transiciones históricas del entorno, lo que permitió al agente adquirir conocimientos a partir de experiencias previas. No obstante, el experimento implementó una metodología novedosa al modificar dinámicamente las prioridades de las transiciones almacenadas en el replay buffer.

Para aplicar técnicas de inferencia causal para examinar una transición en el replay buffer derivó en analizar la acción que tomó el agente en cada episodio. Según [142], si el valor p derivado del análisis estadístico resultaba ser menor que el nivel de significancia predeterminado (α), indicando una disparidad estadísticamente significativa en la longitud media de equilibrio resultante de la intervención, se asignaba una prioridad correspondiente a la transición respectiva. Al enfatizar las transiciones que tuvieron un impacto significativo en su rendimiento, esta técnica facilitó la capacidad del agente para aprender de manera más eficiente.

En síntesis, el procedimiento de entrenamiento de la Deep Q-Network se diseñó a enfoques convencionales en el campo del aprendizaje por refuerzo. Sin embargo, introdujo un aspecto novedoso al integrar la inferencia causal en el mecanismo de priorización del manejo del replay buffer. Esto permitió al agente mejorar su eficiencia de aprendizaje y, en última instancia, mejorar su rendimiento en el problema de control clásico.

Muestreo Basado en Inferencia Causal

La metodología implicó la integración de un procedimiento de inferencia causal en tiempo real en el enfoque de muestreo del replay buffer. Para cada transición de experiencia almacenada en el replay buffer, se llevó a cabo un análisis de inferencia causal específicamente sobre la

acción del agente. El objetivo de este enfoque consistió en evaluar el impacto de priorizar las transiciones en el proceso de muestreo, basándose en la presencia o ausencia de una diferencia causal significativa, en la recompensa obtenida. La hipótesis nula se confirmó cuando no se observó una diferencia causal estadísticamente significativa, lo que indicó que la acción no tuvo un impacto significativo en la recompensa obtenida. El resultado se logró ajustando la prioridad de la transición en el replay buffer, considerando la magnitud de la diferencia causal identificada.

El algoritmo 8, muestra los pasos en los que se incorporó la prioridad determinada por la inferencia causal.

Algorithm 8: Proceso para agregar transiciones al replay buffer con inferencia causal

```

Require: state, action, nextState not Null
Require: rdw, episode_dur  $\geq 0$ 
1 if current_episode < MIN_EPISODES then
2   | T  $\leftarrow$  Transition(state, action, nextState, rdw);
3   | replay_memory.push(T, episode_dur);
4   | priorities.push(0);
5 end
6 else
7   | ps1  $\leftarrow$  calc_prop_score(action = left);
8   | ps2  $\leftarrow$  calc_prop_score(action = right);
9   | match  $\leftarrow$  KNN(ps1, ps2);
10  | T  $\leftarrow$  Transition(state, action, nextState, rdw);
11  | replay_memory.push(T, episode_dur);
12  | has_significant_diff, diff  $\leftarrow$  calc_diff_dur_avg(match, ps1, ps2);
13  | if has_significant_diff then
14  |   | priorities.push(diff);
15  |   end
16  |   else
17  |   | priorities.push(0);
18  |   end
19  |   adjust_priorities(has_significant_diff, diff);
20 end

```

- La variable *state* corresponde a la representación actual del entorno durante la simulación. Para el problema del CartPole, el estado típicamente comprende cuatro variables separadas:

Posición de la rueda: denota la ubicación actual de la rueda en un eje horizontal.

Velocidad del carrito: se refiere a la velocidad a la que el carrito se desplaza en dirección horizontal.

Ángulo del péndulo: se refiere a la inclinación actual del poste en relación con su posición vertical.

Velocidad angular del péndulo: corresponde a la tasa instantánea a la que el poste se mueve en dirección hacia adelante o hacia atrás.

- La variable *action* denota el desplazamiento lateral del carrito, que es una acción fundamental para mantener el equilibrio en el péndulo.
- La variable *nextState* dicta la nueva configuración del sistema después de que el agente ejecuta una acción particular.
- La variable *reward* representa las recompensas acumulativas obtenidas durante cada episodio.
- La variable *episode_dur* especifica la duración o longitud de cada episodio en el proceso de simulación.

Priorización de Transiciones

Para priorizar las transiciones en el replay buffer, basándose en la inferencia causal, se utilizó el propensity score matching [143], una técnica de la teoría causal que se basa en calcular puntajes (8.1), que representan la probabilidad condicional de tomar una acción específica dado un conjunto de características.

$$PS(T | X) = P(T | X) \quad (8.1)$$

Donde $P(T | X)$ se estima a partir de los datos de entrenamiento. Estos puntajes de Propensión se usaron para emparejar transiciones con acciones ligeramente distintas, calculando la diferencia promedio en las duraciones de los episodios entre transiciones con $T = 1$ y $T = 0$.

Así entonces, se determinó T cómo la acción tomada por el agente (por ejemplo, mover hacia la izquierda o hacia la derecha), Y como la duración del episodio, y X como un conjunto de características observadas (por ejemplo, el estado del entorno). Luego, se aplicó el propensity score matching para estimar el efecto causal de T en Y controlando las diferencias en X .

Las prioridades de las transiciones se ajustaron según la diferencia causal estimada, como se muestra en la ecuación 8.2 para una transición i .

$$Priority(i) = |EstimatedCausalDifference(i)| \quad (8.2)$$

La diferencia causal estimada se obtuvo mediante el emparejamiento de estos puntajes, por lo tanto, si la diferencia causal es significativa ($valor - p > \alpha$), se asigna una alta prioridad; de lo contrario, la prioridad es cero. Donde α corresponde al nivel de significancia en la prueba de hipótesis para determinar si la diferencia causal es estadísticamente significativa o no [144].

Muestreo Basado en Prioridades

Durante el proceso de muestreo del replay buffer, la elección de las transiciones se llevó a cabo considerando su prioridad. Las probabilidades de selección $P(i)$ para cada transición i se definieron siguiendo la ecuación (8.3).

$$P(i) = \frac{Priority(i)}{\sum_j Priority(j)} \quad (8.3)$$

Esto garantizó que las transiciones con prioridades más altas tuvieran una probabilidad mayor de ser seleccionadas durante el proceso de muestreo, mientras que aquellas con prioridades cercanas a cero tuvieron una probabilidad muy baja.

Validación Experimental

Para evaluar el rendimiento del enfoque de muestreo basado en la inferencia causal, se realizó una comparación rigurosa con un enfoque tradicional de muestreo aleatorio, por lo cual se implementaron dos versiones de la Deep Q-Network con configuraciones idénticas, excepto por cómo se gestionaba el replay buffer como se describe a continuación:

Deep Q-Network con muestreo basado en inferencia causal Esta iteración de la Deep Q-Network implementó el proceso de muestreo basado en inferencia causal descrito previamente. Las transiciones almacenadas en el Replay Buffer fueron priorizadas en función de la magnitud de la diferencia causal estimada, lo que resultó en un muestreo no uniforme. El agente de aprendizaje utilizó estas muestras para actualizar su política y mejorar su rendimiento en CartPole.

Deep Q-Network con Muestreo Basado en Inferencia Causal La versión de la Deep Q-Network con muestreo aleatorio seguía la práctica estándar de seleccionar transiciones de manera aleatoria y uniforme del replay buffer. Estas transiciones se elegían sin tener en cuenta ninguna inferencia causal, lo que refleja el enfoque tradicional del aprendizaje por refuerzo.

Métricas de Evaluación

Durante el proceso de entrenamiento de la Deep Q-Network con muestreo basado en inferencia causal, se utilizaron diversas métricas para evaluar el rendimiento y el comportamiento del agente en el entorno del problema de control clásico. Estas métricas se calcularon para cada episodio y proporcionaron la información relativa a la capacidad de aprendizaje y exploración del agente. Las métricas clave incluyen:

Recompensa acumulativa o Duración Esta métrica representa la recompensa total obtenida por el agente en un episodio. La recompensa acumulativa, también denominada duración en ciertos escenarios, indica la suma total de las recompensas obtenidas por el agente dentro de un episodio particular en un entorno de aprendizaje por refuerzo. Esta métrica es fundamental para evaluar la eficiencia del agente en lograr sus objetivos dentro del entorno dado.

En un entorno de control clásico, como el entorno de CartPole, la duración está intrínsecamente relacionada con la capacidad del agente para mantener el sistema en un estado equilibrado durante un período de tiempo dado. La duración se define como el número de pasos de tiempo, o iteraciones, durante los cuales el agente logra mantener el sistema estable antes de que se alcance un criterio de finalización predefinido.

El cálculo de la recompensa acumulativa, como se muestra en (8.4), se obtiene sumando las recompensas individuales R_t adquiridas en cada paso de tiempo t a lo largo de un episodio completo.

$$CR = \sum_{t=1}^T R_t \quad (8.4)$$

Pérdida acumulativa La pérdida acumulativa representa la pérdida total experimentada por la red neuronal durante un episodio. Esta métrica se calcula sumando las pérdidas en cada paso de tiempo y puede utilizarse para evaluar la capacidad de adaptación del modelo Deep Q-Network.

Porcentaje de Exploración Este valor representa la proporción de acciones que fueron seleccionadas mediante exploración en lugar de explotación durante el episodio. La exploración se realiza seleccionando acciones al azar, mientras que la explotación elige la acción con el mayor valor esperado según la política actual. Un alto porcentaje de exploración indica un agente que continúa explorando nuevas acciones.

Porcentaje de Explotación Este valor es el complemento del porcentaje de exploración y representa la proporción de acciones seleccionadas a través de la explotación en lugar de la exploración. Un alto porcentaje de explotación sugiere que el agente depende más de su conocimiento adquirido para seleccionar acciones.

Entropía de la Política de la Red La entropía es una medida de la incertidumbre en la política del agente. Se calcula a partir de las probabilidades de selección de acciones. Una entropía alta indica una política incierta y exploratoria, mientras que una entropía baja indica una política más determinista.

Al calcular y registrar métricas para cada episodio durante el proceso de entrenamiento de la Deep Q-Network, el estudio evaluó qué tan bien funcionaba agregar inferencia causal al aprendizaje por refuerzo. Específicamente, se realizó una comparación entre el rendimiento del algoritmo de Deep Q-Network con muestreo basado en inferencia causal y el algoritmo de Deep Q-Network con muestreo aleatorio.

Es esencial destacar que las métricas utilizadas en este experimento han demostrado ser altamente informativas para evaluar tanto el rendimiento como la exploración del agente. Estas métricas han establecido la base para el análisis adicional y las conclusiones derivadas de este experimento, pues ambas variantes de la Deep Q-Network fueron evaluadas a lo largo de múltiples episodios de entrenamiento, registrando métricas relevantes en cada episodio. Se compararon sus resultados de rendimiento en términos de duración del episodio, recompensa acumulativa, pérdida acumulativa, porcentaje de exploración, porcentaje de explotación y entropía de las políticas aprendidas.

8.3. Resultados

Se presentan los resultados obtenidos utilizando la metodología de muestreo basada en inferencia causal dentro del marco de la Deep Q-Network. Durante las distintas sesiones de entre-

namiento y evaluación, se recopiló una gran cantidad de datos que ofrecen una visión detallada sobre la eficacia y las implicaciones de integrar la inferencia causal en el proceso de muestreo de la Deep Q-Network.

Comparación de rendimiento entre modelos con muestreo basado en inferencia causal y muestreo aleatorio en diferentes configuraciones de redes neuronales

Se realizó una evaluación de diez modelos de Deep Q-Network en el entorno de aprendizaje por refuerzo CartPole. Estos modelos se pueden dividir en dos grupos distintos: cinco de ellos aplican técnicas de muestreo basadas en inferencia causal en el replay buffer, mientras que los otros cinco utilizan métodos convencionales de muestreo aleatorio. La diferencia principal entre estos dos grupos reside en su enfoque para manejar la selección de transiciones durante el entrenamiento de la Deep Q-Network.

El experimento involucró la configuración de diversas arquitecturas de redes neuronales para cada grupo de modelos. Esto implicó la implementación de dos metodologías de muestreo: una basada en inferencia causal y otra utilizando muestreo aleatorio. Para estas configuraciones, fue necesario ajustar el número de neuronas en la capa oculta. Se estableció un nivel de significancia de $\alpha = 0,05$ para el experimento. El tamaño de la muestra se determinó en función del tamaño de la memoria de repetición del agente, mientras que el tamaño del lote se mantuvo constante en 64.

Se observó cómo se comportó el rendimiento de la Deep Q-Network al emplear muestreo basado en inferencia causal en diversas situaciones de modelado de la función Q , utilizando diferentes arquitecturas de redes. La tabla 8.1 proporciona una visión general exhaustiva de los resultados recopilados, mostrando diversas métricas y sus abreviaturas correspondientes.

Las métricas incluyen el modelo utilizado, el número de capas ocultas (NHL) y las técnicas de muestreo utilizadas, como muestreo basado en inferencia causal (CIS) y muestreo aleatorio (RS). Además, también presenta métricas relacionadas con la recompensa promedio (AR), la exploración promedio (AER) y la explotación promedio (AET).

Tabla 8.1. Métricas de rendimiento de modelos con muestreo basado en inferencia causal y muestreo aleatorio.

Model	NHL	CIS	RS	AR	AER	AET
1	8	X		119.5	44.30	75.20
2	8		X	116.95	44.20	72.75
3	16	X		120.75	44.85	75.90
4	16		X	148.15	45.80	102.35
5	32	X		125.05	44.60	80.45
6	32		X	120.2	44.45	75.75
7	64	X		132.9	45.45	87.45
8	64		X	117.9	45.75	72.15
9	128	X		131.9	45.35	86.55
10	128		X	345.95	62.35	283.60

Es evidente que, en tres de los cinco grupos, la Deep Q-Network con muestreo basado en inferencia causal exhibió un rendimiento superior en comparación con la Deep Q-Network con muestreo aleatorio. Sin embargo, es importante tener en cuenta que la arquitectura y la configuración del modelo tienen un impacto significativo en su rendimiento. Entre los tres modelos considerados, el de la Deep Q-Network con muestreo basado en inferencia causal destacó por su rendimiento notable. Para propósitos de evaluación, se optó por el modelo 7, el cual demostró un rendimiento superior, al igual que su contraparte, el modelo 8.

Comparación de modelos de proceso de Markov y la Deep Q-Network con muestreo basado en inferencia causal

Se compararon dos modelos de proceso de Markov en el escenario de control CartPole de OpenAI Gym [134] (figura 8.2), y se observaron disparidades significativas en términos de rendimiento y efectividad. El modelo inicial, desarrollado internamente y presentado en el algoritmo 9, alcanza una recompensa promedio de 29.806. Aunque este modelo produce resultados aceptables, el otro modelo evaluado exhibe un rendimiento superior.

El segundo modelo, [145], muestra una recompensa media notablemente elevada, alcanzando un valor de 96.15. La mejora sustancial en el rendimiento implica la utilización de estrategias más refinadas en contraste con el modelo inicial. Por otro lado, el modelo Deep Q-Network, que se basa en inferencia causal para su procedimiento de muestreo, alcanza una recompensa promedio de 20.17. Aunque este modelo demuestra un rendimiento ligeramente inferior en comparación con los modelos de proceso de Markov, vale la pena mencionar que los tres modelos muestran diferentes niveles de éxito al abordar el problema de CartPole. La variación en los resultados entre estos modelos resalta la importancia de la metodología de diseño y la incorporación de técnicas particulares en algoritmos de aprendizaje por refuerzo.



Figura 8.2. Entornos de prueba CartPole y MountainCar.

Algorithm 9: Cálculo de recompensas en el entorno CartPole usando el proceso de Markov

Input: *env, policy, render*

```
1 Function DiscretizeAngle(angle):  
2   if angle  $\geq$  0 then  
3     return 1  
4   end  
5   else  
6     return 0  
7   end  
8 observation  $\leftarrow$  env.reset();  
9 total_reward  $\leftarrow$  0;  
10 for t = 1 to 1000 do  
11   if render then  
12     env.render();  
13   end  
14   angle  $\leftarrow$  observation[2] state  $\leftarrow$  DiscretizeAngle(angle) action  $\leftarrow$  policy(state)  
    observation, reward, done, _, _  $\leftarrow$  env.step(action);  
15   total_reward  $\leftarrow$  total_reward + reward;  
16   if done then  
17     break;  
18   end  
19 end  
20 return total_reward;
```

Rendimiento de la Deep Q-Network en los entornos de MountainCar y CartPole

Se evaluó el rendimiento de la Deep Q-Network en el entorno de OpenAI Gym, específicamente empleando muestreo basado en inferencia causal en el escenario de MountainCar. El modelo mostró una recompensa promedio de -175, lo cual es significativamente menor que la recompensa promedio original de 20.17 en el escenario de CartPole. En el escenario de MountainCar, el proceso de toma de decisiones abarcó tres variables distintas para cada estado, mientras que en CartPole solo están involucradas dos variables. En el entorno de MountainCar, el objetivo principal es subir la montaña, lo que generalmente conduce a una disminución en la recompensa total. La presencia de una recompensa promedio negativa indica que existe la necesidad de mejorar y optimizar el modelo. La influencia de la complejidad de las variables y el número de episodios de entrenamiento en el proceso de aprendizaje es evidente, lo que resalta la importancia de tener una cantidad suficiente de datos en el replay buffer para mejorar la precisión del proceso de inferencia causal.

Métricas de Evaluación

Para evaluar la efectividad de integrar inferencia causal en el proceso de muestreo de la Deep Q-Network, se recolectaron y analizaron diversas métricas de evaluación durante la fase de entrenamiento y las interacciones entre el agente y el entorno en un escenario de control estándar. Estas métricas brindan una visión integral del rendimiento de los modelos en términos

de aprendizaje, exploración, explotación y estabilidad durante la resolución de tareas de control.

En la evaluación del rendimiento de los modelos 7 y 8, se observaron diferencias notables en su comportamiento y rendimiento en el problema de control clásico. Como se muestra en la figura 8.3, el modelo 7, que utilizó el muestreo basado en inferencia causal como parte de su proceso de aprendizaje, tiende a obtener más recompensas en cada episodio de entrenamiento que el modelo 8. Este hallazgo sugiere que la integración de la inferencia causal en el muestreo puede conducir a una mejora en la capacidad del agente para alcanzar un rendimiento más alto en términos de recompensas acumuladas. Además, se observa que los episodios en el modelo 7 tienden a durar ligeramente más en comparación con el modelo 8. La duración de los episodios es un indicador de la estabilidad del controlador, y estos resultados sugieren que el modelo 7 puede ser más efectivo en mantener el sistema en un estado equilibrado durante más tiempo.

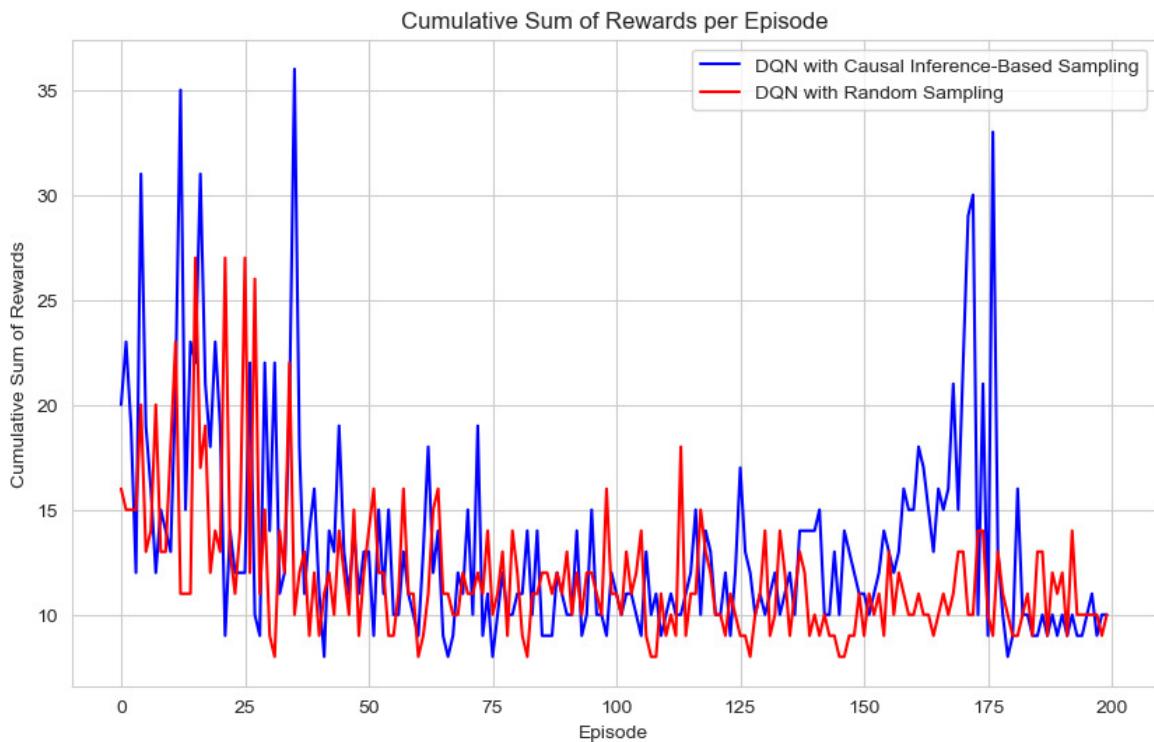


Figura 8.3. Recompensa Acumulativa: Muestreo Basado en Inferencia Causal vs. Muestreo Aleatorio en Deep Q-Network

Sin embargo, como se muestra en la figura 8.4, un aspecto interesante es la divergencia en las estrategias de exploración entre los dos modelos. A pesar de alcanzar recompensas más altas y episodios más largos, el modelo 7 exhibió un aumento en el porcentaje de exploración en comparación con el modelo 8. Este aumento en la exploración puede indicar que el Modelo 7 continuó explorando un rango más amplio de acciones, incluso cuando su rendimiento fue superior.

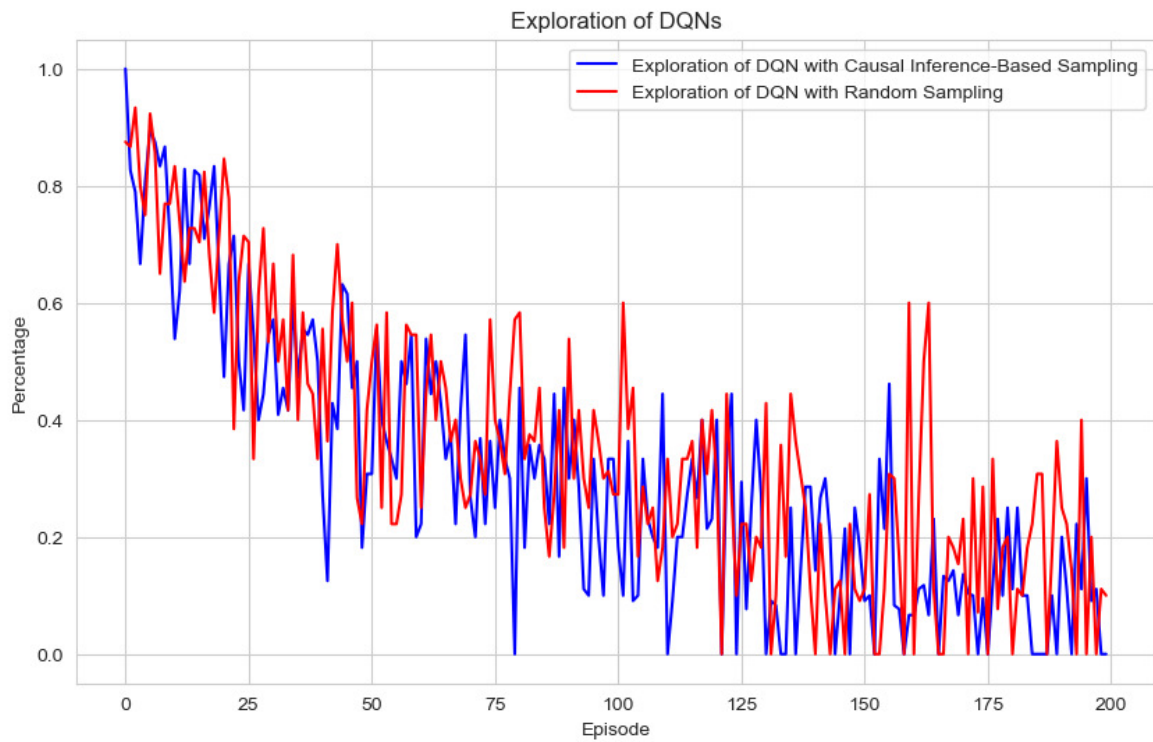


Figura 8.4. Porcentaje de Exploración: Muestreo Basado en Inferencia Causal vs. Muestreo Aleatorio en Deep Q-Network

Por otro lado, como se ilustra en la figura 8.5, el modelo 7, aunque tuvo un porcentaje de exploración más bajo, muestra un porcentaje de explotación más alto en sus acciones. Este equilibrio entre exploración y explotación es una característica importante del aprendizaje por refuerzo y puede influir en la efectividad general del agente.

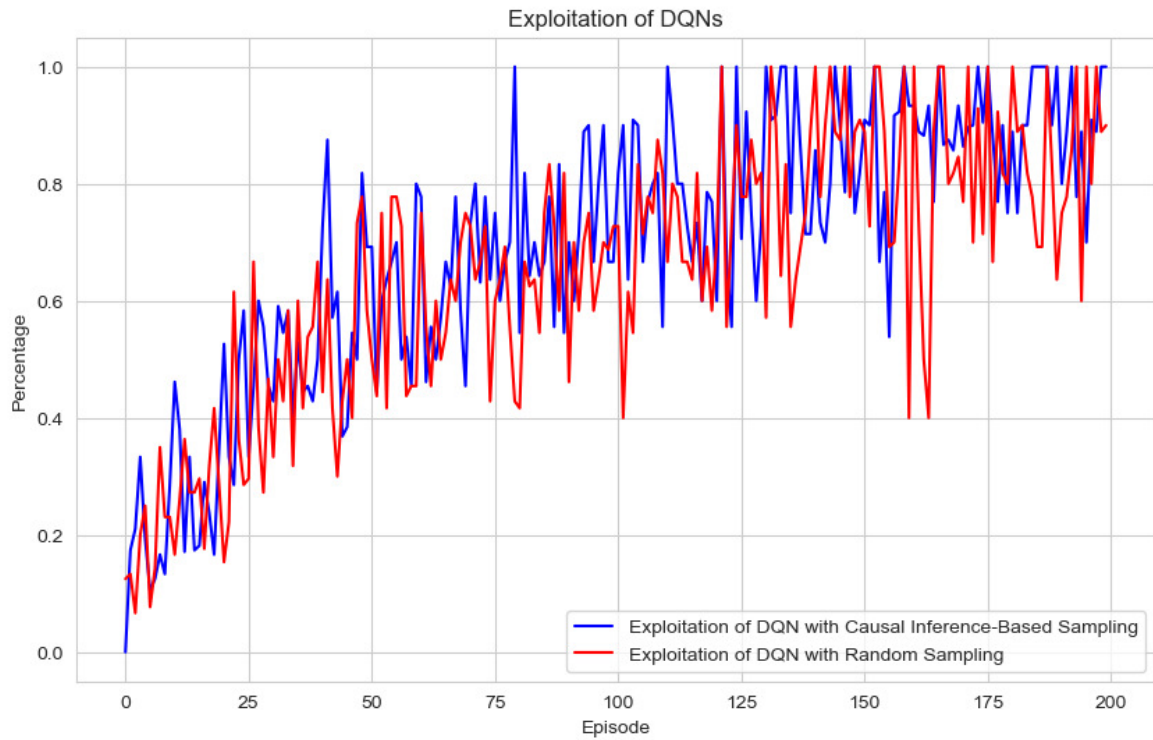


Figura 8.5. Porcentaje de Explotación: Muestreo Basado en Inferencia Causal vs. Muestreo Aleatorio en Deep Q-Network

En cuanto a las métricas de pérdida en la figura 8.6, el modelo 8 muestra valores de pérdida acumulada más bajos en comparación con el modelo 7. Esto puede indicar una convergencia más rápida o una adaptación más efectiva del Modelo 8 a las recompensas y transiciones del entorno. Esto también confirma lo mencionado por [146], *"la priorización en el replay buffer podría ralentizar el aprendizaje"*. No obstante, es importante considerar que la pérdida no necesariamente es el único indicador de rendimiento, y otras métricas como las recompensas acumuladas también son cruciales para evaluar el rendimiento del agente.

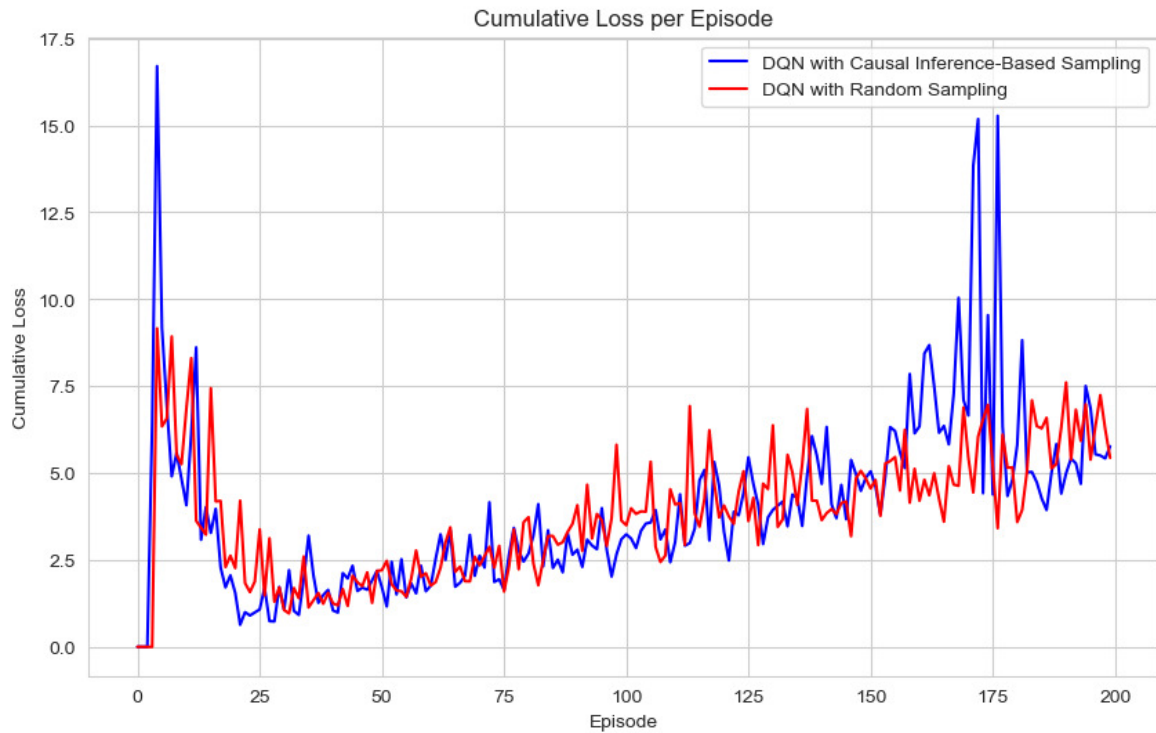


Figura 8.6. Pérdida Acumulativa: Muestreo Basado en Inferencia Causal vs. Muestreo Aleatorio en Deep Q-Network

Un aspecto intrigante se relaciona con la entropía de las políticas aprendidas en ambos modelos. En la figura 8.7 se observa que la entropía de la política del modelo 7 se mantuvo consistentemente por encima de la del modelo 8, y ambos mostraron un aumento considerable en cada episodio. Este fenómeno puede interpretarse como una señal de que una política con alta entropía es más exploratoria, ya que tiende a considerar un factor de ajuste (γ) más amplio de acciones de manera equitativa. Por otro lado, una política con baja entropía se vuelve más determinista y predecible en sus acciones. La presencia de alta entropía en el modelo 7 pudo estar relacionada con el grado de exploración y la capacidad para descubrir estrategias más diversas.

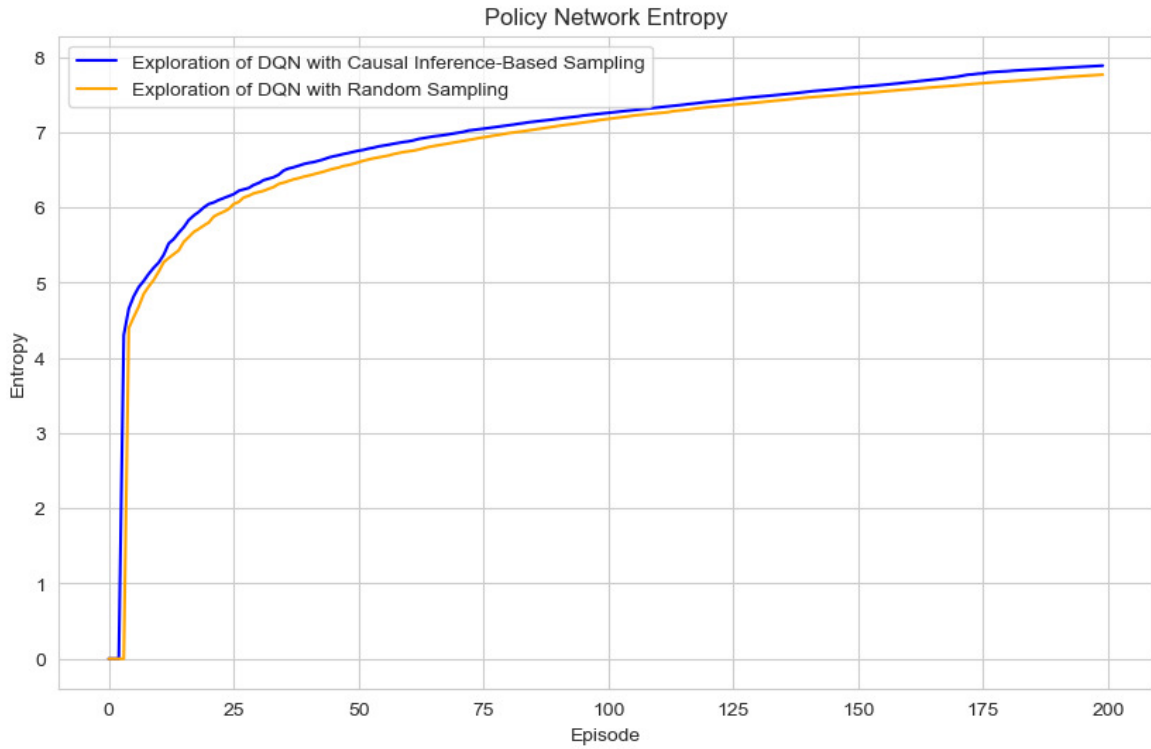


Figura 8.7. Entropía de la Política aprendida: Muestreo Basado en Inferencia Causal vs. Muestreo Aleatorio en Deep Q-Network

En general, estos resultados resaltan la influencia de la inferencia causal en el proceso de aprendizaje por refuerzo. Si bien el modelo 7 mostró una tendencia hacia la obtención de recompensas más altas y episodios más largos, su estrategia de exploración y la alta entropía de la política despertaron interés en comprender cómo se equilibran la exploración y la explotación en la toma de decisiones del agente. Estos hallazgos enfatizan la complejidad de integrar la inferencia causal en el aprendizaje por refuerzo y sugieren que su impacto puede variar según el contexto del problema y los objetivos del agente.

8.4. Conclusiones

En este experimento, se evaluó el rendimiento de dos modelos de Deep Q-Network con arquitecturas de redes neuronales idénticas pero empleando dos enfoques de muestreo diferentes: muestreo basado en inferencia causal y muestreo aleatorio. Los hallazgos revelan diferencias significativas en el comportamiento y rendimiento de estos modelos en el contexto de un problema de control clásico.

Los resultados indican que la integración de la inferencia causal en el muestreo de una Deep Q-Network puede mejorar su rendimiento en términos de duración del episodio y recompensas acumuladas en problemas de control clásicos. Sin embargo, también se observaron diferencias en la exploración y la entropía de la política, subrayando la importancia de considerar cuidadosamente el equilibrio entre exploración y explotación en el diseño de algoritmos de aprendizaje por refuerzo.

A través de la implementación de esta estrategia, el objetivo no fue solo evaluar la eficacia numérica de los modelos, sino también adquirir una comprensión más profunda de su capacidad para ajustarse y aplicarse a diversas circunstancias.

El muestreo basado en inferencia causal generó conocimientos que pueden eventualmente mejorar el progreso de los algoritmos de aprendizaje por refuerzo mediante Deep Learning, permitiendo soluciones más flexibles y resilientes a problemas complejos del mundo real.

Parte III

Resultados, conclusiones y trabajo futuro

Resultados

En esta estudio, se realizaron cinco experimentos para investigar la integración de la inferencia causal en procesos de Machine Learning. Cada experimento abordó aspectos específicos de esta integración y proporcionó información relevante sobre su efectividad y aplicabilidad en diferentes contextos.

9.1. Experimento 1: Simulación y visualización del fenómeno interacción luz-materia

El primer experimento tuvo como objetivo simular la distribución del valor de iluminancia en un dominio espacial 3D con diversas configuraciones de iluminación. Los resultados de este experimento proporcionaron una línea base para la visualización de resultados en experimentos posteriores.

En el primer ejercicio de visualización, se observó una distribución uniforme de la luz, pero con detalles perdidos debido a una singularidad en los cálculos.

En el segundo ejercicio, al evitar esta singularidad, se mejoró la visualización de la propagación de la luz en zonas más alejadas de la fuente lumínica. Sin embargo, aún no se logró una cobertura total del área de interés.

En el tercer ejercicio, se combinaron los cálculos reales de iluminancia con la visualización mejorada del segundo ejercicio, aunque se utilizó la técnica del falso color para una representación más clara de los datos. Además, se demostró la asimetría en la distribución de la iluminancia al introducir múltiples fuentes lumínicas.

Patrones emergentes: Los resultados de este experimento destacaron la importancia de considerar singularidades en los cálculos de iluminancia para una visualización precisa. Además, se demostró la capacidad de las técnicas de visualización para mejorar la comprensión de fenómenos físicos complejos.

9.2. Experimento 2: Estimación de valores de iluminancia mediante Deep Learning

Este experimento se centró en el desarrollo de modelos de redes neuronales para estimar el valor de iluminancia en un espacio tridimensional, teniendo en cuenta configuraciones precisas de fuentes de luz -determinadas en el primer experimento-. Se evaluaron dos enfoques: uno basado en Feed-Forward Neural Networks (FFNN) y otro en Convolutional Neural Networks (CNN).

En el enfoque Feed-Forward Neural Networks, se observó que el modelo FFNN logró una alta precisión en la estimación de la iluminancia, alcanzando tasas de éxito del 98.8 % y 96.0 % en situaciones con una y dos fuentes de luz, respectivamente. La visualización de los valores de iluminancia estimados mostró una correspondencia estrecha con los valores esperados, con diferencias mínimas entre ellos. En el enfoque Convolutional Neural Networks, el modelo CNN también demostró un alto nivel de precisión en sus estimaciones, con tasas de éxito del 99.2 % y 96.4 % en situaciones con una y dos fuentes de luz, respectivamente. La comparación entre los valores de iluminancia estimados y los valores esperados mostró una excelente correspondencia, destacando la efectividad de este enfoque. Al comparar los modelos FFNN y CNN, se mostró que ambos modelos tuvieron un desempeño similar en términos de precisión, aunque los modelos CNN mostraron una ligera ventaja sobre los modelos FFNN en algunos escenarios.

En este experimento se exploró la aplicación de la inferencia causal en el cálculo de la iluminancia, considerando diversos factores que podrían influir en la distribución de la luz. Estos factores incluyeron, entre otras, la orientación de las fuentes de luz, la distribución espacial de las fuentes de luz, la reflectividad de las superficies y la presencia de obstrucciones y sombras. A través de modelos estructurales, se pudo analizar cómo cada factor afecta las componentes horizontal y vertical de la iluminancia, lo que proporcionó una comprensión más completa de la iluminación en un entorno dado.

9.3. Experimento 3: Inferencia causal para determinar la causa de un fenómeno de sombra en una imagen

En este experimento, se desarrolló un modelo artificial basado en inferencia causal para explicar la formación de sombras proyectadas en una imagen, específicamente sobre la superficie inferior de un escenario tridimensional. Se utilizó el método de descubrimiento causal NOTEARS para validar el modelo causal estructural diseñado.

Se observó que la generación del modelo causal estructural fue exitosa tras aplicar una restricción adicional al algoritmo de descubrimiento causal. Este modelo proporcionó coherencia a la representación de las relaciones causales entre las variables involucradas, incluida la presencia de la esfera y la proyección de la sombra. A partir de la distribución de probabilidad condicional obtenida del modelo causal, se realizaron inferencias sobre la causa de la sombra proyectada. Se encontró que la presencia de la esfera aumenta significativamente la probabilidad de detectar una sombra en la superficie, lo que sugirió que la esfera fue la causa más probable de la sombra observada.

Se compararon diferentes métodos de detección de sombras, incluida una combinación del método de Felzenszwalb con inferencia causal. Se encontró que este enfoque mostró los resultados más prometedores, con una precisión del 51.5 % en la detección de sombras. A pesar de la presencia de factores de confusión, este enfoque logró determinar la causalidad de la sombra con éxito, lo que resalta la utilidad de la inferencia causal en la comprensión de fenómenos visuales complejos a costa de un conjunto de datos que no solo se caracterice por lo correlativo sino que incluya esencialmente información sobre intervenciones, factores de confusión, y resultados.

9.4. Experimento 4: Inferencia causal para el modelo de comportamiento de un agente inteligente

En este experimento, se desarrolló un agente inteligente que empleó la inferencia causal para comprender las causas detrás de los fallos de otros agentes al completar una tarea. Este agente fue diseñado para ejecutar la tarea tomando decisiones informadas basadas en la inferencia causal, lo que le permitió anticipar posibles obstáculos y tomar medidas correctivas apropiadas.

Se aplicó inferencia causal para determinar la causa de los fallos en la ejecución de la tarea por parte de otros agentes. Se encontró que la presencia de sensores específicos en el agente oponente tenía un impacto significativo en la probabilidad de éxito del agente. La inferencia causal permitió identificar qué sensor del agente oponente influía más en el resultado de la tarea. Se comparó el rendimiento del agente inteligente basado en inferencia causal con otros dos agentes: uno que utilizaba un algoritmo de búsqueda no informada y otro que utilizaba el algoritmo de búsqueda informada A^* . Se encontró que el agente basado en inferencia causal superaba al agente con búsqueda no informada y que también tenía un rendimiento similar al agente con búsqueda informada, pero con una comprensión más profunda de las causas subyacentes de los fallos.

Además, se analizó la relación entre el estado de los sensores del agente oponente y el éxito en la tarea, así como el consumo de energía de los agentes en diferentes escenarios. Se observó una correlación entre la detección por parte del agente oponente y el consumo de energía del agente, lo que sugiere que la capacidad de evadir la detección visual puede afectar la eficiencia con la que se utilizan los recursos energéticos.

9.5. Experimento 5: Incorporación de la inferencia causal en el procedimiento de muestreo de una Deep Q-Network

El objetivo fue desarrollar una Deep Q-Network que integrara un método de muestreo basado en la inferencia causal para equilibrar un poste vertical sobre un carrito móvil (entorno CartPole de OpenAI GYM).

Se evaluaron diez modelos de tipo Deep Q-Network, cinco con muestreo basado en inferencia causal y cinco con muestreo aleatorio. Se observó un rendimiento superior en tres de los cinco

modelos con muestreo basado en inferencia causal. La elección del modelo dependió de la arquitectura y configuración de la red. Se compararon dos modelos de proceso de Markov con el modelo Deep Q-Network que utiliza muestreo basado en inferencia causal. Aunque este último mostró un rendimiento ligeramente inferior, se destacó la importancia de las estrategias de diseño en el aprendizaje por refuerzo. Así mismo, se contrastó el rendimiento del modelo Deep Q-Network con muestreo basado en inferencia causal en los entornos de control clásico CartPole y MountainCar -también de OpenAI GYM-. En esta evaluación se observó una recompensa promedio más baja en MountainCar debido a la complejidad de las variables y la necesidad de más datos en la memoria de repetición o replay buffer.

Se determinó que la integración de la inferencia causal en el muestreo de una Deep Q-Network puede mejorar su capacidad para alcanzar mayores recompensas y prolongar los episodios. También se evidenció que la estrategia de exploración y la entropía de la política aprendida son aspectos cruciales al evaluar el impacto de la inferencia causal en el aprendizaje por refuerzo. Además, la complejidad del problema y la cantidad de datos disponibles pueden influir en la efectividad de la inferencia causal en este contexto.

9.6. Generalización de resultados

Aunque los hallazgos de cada experimento tienen el potencial de aplicarse a contextos más amplios, es importante tener en cuenta las limitaciones específicas de cada estudio. La generalización efectiva requiere considerar la variabilidad y la complejidad de los entornos y sistemas en los que se aplicarán los hallazgos [62], [42], [147] y [148].

Experimento 1

Aplicabilidad: Los resultados de este experimento, que resaltan la importancia de considerar singularidades en los cálculos de iluminancia para una visualización precisa, son relevantes no solo en simulaciones de iluminación, sino también en cualquier contexto donde la visualización de datos espaciales sea crucial.

Limitaciones: La simulación se centró en un fenómeno específico (distribución de iluminancia en un espacio 3D) y puede que no sea directamente aplicable a otros fenómenos físicos o sistemas de datos.

Experimento 2

Aplicabilidad: Los modelos de redes neuronales desarrollados para estimar la iluminancia pueden ser aplicables en una variedad de contextos donde la estimación de variables espaciales es necesaria, como la monitorización ambiental o la planificación urbana.

Limitaciones: La precisión y eficacia de los modelos pueden verse afectadas por la complejidad y la variabilidad de los entornos de iluminación, así como por la disponibilidad y calidad de los datos de entrenamiento.

Experimento 3

Aplicabilidad: El uso de inferencia causal para entender la formación de sombras en imágenes puede ser relevante en campos como la visión por computadora, la robótica y la fotografía computacional.

Limitaciones: La efectividad del modelo causal estructural puede depender de la complejidad de las escenas y la disponibilidad de datos para entrenar el modelo. Además, el método de detección de sombras utilizado puede no ser óptimo en todas las situaciones.

Experimento 4

Aplicabilidad: El enfoque de utilizar inferencia causal para comprender el comportamiento de agentes inteligentes puede ser útil en sistemas de inteligencia artificial y robótica, así como en la toma de decisiones en entornos dinámicos.

Limitaciones: La generalización del rendimiento del agente basado en inferencia causal puede depender de la complejidad y la variabilidad de los entornos, así como de la disponibilidad de datos para entrenar el modelo causal.

Experimento 5

Aplicabilidad: La integración de la inferencia causal en el aprendizaje por refuerzo puede ser relevante en una amplia gama de aplicaciones, desde control de robots hasta optimización de procesos.

Limitaciones: La efectividad del enfoque puede verse afectada por la complejidad del entorno y la tarea, así como por la capacidad del modelo de inferencia causal para capturar relaciones causales complejas.

9.7. Análisis comparativo

El análisis entre los resultados obtenidos en los diferentes experimentos realizados se centra en identificar similitudes y diferencias significativas en los hallazgos de cada experimento, así como en discutir cómo estas variaciones podrían relacionarse con las diferencias en los diseños experimentales. Para facilitar la comprensión y visualización de las comparaciones, se presenta la Tabla 9.1, que sintetiza los aspectos clave de cada experimento, incluyendo la precisión de los modelos utilizados, la utilidad de la inferencia causal, la aplicabilidad de los modelos en diferentes contextos, el rendimiento de los modelos y la complejidad de los fenómenos estudiados. Esta tabla servirá como referencia para el análisis detallado que se llevará a cabo a continuación.

Tabla 9.1. Análisis Comparativo de Experimentos

Aspecto	EXPERIMENTOS				
	E1	E2	E3	E4	E5
Precisión de Modelos	-	Alta	-	-	Alta
Utilidad de la Inferencia Causal	-	-	Alta	Alta	-
Aplicabilidad de Modelos	-	-	Visual	Comportamental	Refuerzo
Rendimiento de Modelos	-	Ligeramente inferior	-	-	Mejoras
Complejidad del Fenómeno Estudiado	Media	Media	Baja	Alta	Alta
Diversidad de Datos y Entornos	Media	Media	Baja	Alta	Media

Similitudes en los Hallazgos:

Precisión de Modelos de Redes Neuronales: Tanto en el Experimento 2, en el que se trabajó con redes feed-forward y redes convolucionales, como en el Experimento 5, en el que el modelo intervenido fue la Deep Q-Network, se observó una alta precisión en la estimación de variables (iluminancia en el Experimento 2 y equilibrio del poste en el Experimento 5) utilizando modelos de redes neuronales.

Utilidad de la Inferencia Causal: En los Experimentos 3 y 4, la aplicación de inferencia causal demostró ser útil para comprender fenómenos complejos, ya sea la formación de sombras en imágenes o el comportamiento de agentes inteligentes.

Diferencias en los Hallazgos:

Aplicabilidad de Modelos de Inferencia Causal: Mientras que en el Experimento 3 la inferencia causal se aplicó con éxito para comprender un fenómeno visual, en el Experimento 4 se utilizó para comprender el comportamiento de agentes inteligentes en un entorno dinámico.

Rendimiento de Modelos de Aprendizaje por Refuerzo: Aunque en el Experimento 5 se observaron mejoras en el rendimiento de la Deep Q-Network con muestreo basado en inferencia causal, este enfoque mostró un rendimiento ligeramente inferior en comparación con el muestreo aleatorio en algunos modelos.

Relación con los Diseños Experimentales:

Complejidad del Fenómeno Estudiado: Las diferencias en la complejidad de los fenómenos estudiados pueden haber influido en la efectividad de los enfoques utilizados. Por ejemplo, la inferencia causal puede ser más efectiva para comprender fenómenos visuales simples que para modelar comportamientos complejos de agentes inteligentes.

Diversidad de Datos y Entornos: Las variaciones en los diseños experimentales pueden reflejar diferencias en la diversidad de datos utilizados y en la complejidad de los entornos simulados, lo que a su vez puede afectar la aplicabilidad y la generalización de los resultados.

9.8. Relación entre los Resultados de los Experimentos y los Objetivos de la Investigación

En esta sección, se analiza cómo los resultados obtenidos en cada experimento contribuyen al logro de los objetivos específicos de la investigación.

Objetivo 1: Desarrollar un modelo de aprendizaje de máquina para estimar el valor de iluminancia en un escenario 3D

El Experimento 2 se centró en el desarrollo de modelos de redes neuronales de tipo feed-forward y convolucionales para estimar la iluminancia en un entorno tridimensional. Los resultados de este experimento demostraron una alta precisión en la estimación de la iluminancia, lo que cumple con el objetivo de desarrollar un modelo de aprendizaje de máquina para este propósito específico.

Objetivo 2: Desarrollar un modelo que aplique la inferencia causal para determinar la causa de un fenómeno de sombra en una imagen

El Experimento 3 se enfocó en el desarrollo de un modelo artificial basado en inferencia causal para explicar la formación de sombras proyectadas en una imagen. Los resultados de este experimento proporcionaron una representación coherente de las relaciones causales entre las variables involucradas, lo que respalda el objetivo de desarrollar un modelo que aplique la inferencia causal para entender/explicar los fenómenos de sombras en imágenes.

Objetivo 3: Desarrollar un modelo cognitivo basado en inferencia causal para apoyar la toma de decisiones de un agente en un entorno virtual

El Experimento 4 implicó el desarrollo de un agente inteligente que utilizó la inferencia causal para comprender las causas detrás de los fallos en la ejecución de una tarea y tomar decisiones informadas en un entorno virtual. Estos resultados están alineados con el objetivo de desarrollar un modelo cognitivo basado en inferencia causal para apoyar la toma de decisiones de un agente en un entorno virtual.

Objetivo 4: Desarrollar un modelo de aprendizaje de máquina con una Deep Q-Network que utilice inferencia causal para el proceso de muestreo del replay buffer

El Experimento 5 se centró en el desarrollo de una Deep Q-Network que integró la inferencia causal en el proceso de muestreo del replay buffer para solucionar un problema de control clásico. Los resultados de este experimento mostraron mejoras significativas en la capacidad de la Deep Q-Network para solucionar problemas de control clásico, lo que respalda el objetivo de desarrollar un modelo de aprendizaje de máquina que utilice inferencia causal para este propósito.

Capítulo 10

Conclusiones

Durante el desarrollo de esta investigación, se ha explorado la integración de la inferencia causal en procesos de Machine Learning, con el objetivo de mejorar la resolución de problemas donde la explicabilidad y la comprensión de las relaciones causales son fundamentales. A través de una serie de experimentos meticulosamente diseñados y ejecutados, se han obtenido resultados significativos que confirman el valor y las implicaciones de esta integración en diversas aplicaciones del aprendizaje automático.

El primer experimento proporcionó una comprensión detallada de los desafíos asociados con la visualización de la distribución de iluminancia en un escenario tridimensional. Los hallazgos destacaron la importancia de considerar factores como la subdivisión del escenario, el uso de escalas logarítmicas y la anulación de fuentes lumínicas para obtener visualizaciones precisas y significativas. Estas lecciones son fundamentales en el contexto más amplio de esta investigación, ya que subrayan la necesidad de abordar la complejidad de los datos en el aprendizaje automático y la importancia de la explicabilidad en la comprensión de los fenómenos estudiados.

El segundo experimento se centró en el desarrollo de modelos de aprendizaje de máquina para estimar la iluminancia en escenarios 3D. Aunque la precisión de las estimaciones no alcanzó el 100 %, los resultados demostraron el éxito de la aplicación de técnicas de Deep Learning en esta tarea, cumpliendo así con el objetivo de explorar cómo la integración y complementación del Machine Learning y la inferencia causal pueden mejorar la resolución de problemas donde la explicabilidad es fundamental. Por ejemplo, se observó que tanto los modelos de redes neuronales Feed-Forward (FFNN) como Convolutional Neural Networks (CNN) lograron una alta precisión en la estimación de la iluminancia en un entorno 3D, con tasas de éxito superiores al 95 %, respaldando la eficacia de estos modelos para estimar variables ambientales.

En el tercer experimento, se introdujo el uso de la inferencia causal para determinar la causa de fenómenos de sombra en imágenes. Este enfoque permitió fortalecer hipótesis y mejorar la comprensión de eventos complejos, cumpliendo con el objetivo de analizar cómo la inferencia causal puede potenciar diversas aplicaciones de Machine Learning relacionadas con la explicabilidad de imágenes. Por ejemplo, se encontró que el modelo basado en inferencia causal pudo explicar de manera coherente la formación de sombras en una imagen, lo que indica que la inferencia causal puede ser efectiva para entender fenómenos visuales complejos.

El cuarto experimento exploró la convergencia entre agentes inteligentes e inferencia causal, revelando el potencial transformador de incorporar el razonamiento causal en la toma de decisiones de los agentes. Los resultados obtenidos resaltaron la ventaja significativa de la aplicación de la inferencia causal en entornos de simulación física diversos, cumpliendo así con el objetivo de analizar cómo la integración de la inferencia causal puede mejorar la resolución de problemas donde la explicabilidad es fundamental. Por ejemplo, se demostró que el agente inteligente basado en inferencia causal superó a otros agentes en la ejecución de una tarea compleja en un entorno virtual, demostrando una comprensión más profunda de las causas subyacentes de los fallos en la ejecución de la tarea y tomando decisiones informadas para evitarlos.

Finalmente, el quinto experimento evaluó el rendimiento de modelos de Deep Q-Network con y sin integración de inferencia causal en problemas de control clásico. Los hallazgos demostraron mejoras significativas en el rendimiento al incorporar la inferencia causal en el proceso de muestreo, destacando la importancia de considerar cuidadosamente el equilibrio entre exploración y explotación en el diseño de algoritmos de aprendizaje por refuerzo. Este cumplimiento del objetivo de analizar cómo la integración y complementación del Machine Learning y la inferencia causal pueden mejorar la resolución de problemas se refleja en los resultados obtenidos. Por ejemplo, se encontró que la DQN que integró la inferencia causal en el proceso de muestreo del replay buffer mostró mejoras significativas en la capacidad de alcanzar recompensas más altas y completar episodios más largos en comparación con una DQN que utilizaba muestreo aleatorio, lo que respalda la efectividad de este enfoque para mejorar el aprendizaje por refuerzo.

En conjunto, los resultados obtenidos en esta investigación subrayan el potencial y la relevancia de la integración de la inferencia causal en procesos de Machine Learning. Además de contribuir al avance teórico en este campo, los hallazgos tienen importantes implicaciones prácticas para diversas aplicaciones del aprendizaje automático, desde la estimación de iluminancia hasta el aprendizaje por refuerzo en entornos complejos.

Se identifican áreas prometedoras para futuras investigaciones, como la mejora de la inferencia causal en entornos gráficos complejos, la exploración de nuevas estrategias de aprendizaje por refuerzo basadas en principios de inferencia causal y el desarrollo de modelos adaptativos capaces de manejar variaciones en las configuraciones de iluminación y otras condiciones ambientales.

En síntesis, este trabajo de tesis proporciona una contribución significativa al entendimiento y la aplicación de la inferencia causal en el contexto del Machine Learning, cumpliendo con el objetivo de explorar cómo esta integración puede mejorar la resolución de problemas donde la explicabilidad es fundamental, y ofreciendo una base sólida para futuras investigaciones y el desarrollo de aplicaciones prácticas en una variedad de campos.

Trabajo futuro

Basado en los resultados y observaciones de los experimentos actuales, se identifican varias vías para futuras investigaciones y desarrollos. Estas incluyen la mejora de los algoritmos, la exploración de nuevos entornos y la mejora de la robustez y generalización de las políticas aprendidas. Específicamente, se proponen las siguientes áreas de enfoque:

Mejoras algorítmicas

Estrategias avanzadas de exploración

Implementar técnicas avanzadas de exploración, como Upper Confidence Bound y Muestreo de Thompson, podría mejorar el equilibrio entre exploración y explotación, potencialmente llevando a un aprendizaje más eficiente.

Aprendizaje profundo por refuerzo

Extender el trabajo actual para incorporar técnicas de aprendizaje profundo, como Double Deep Q-Network o Deep Deterministic Policy Gradient, podría permitir manejar entornos más complejos con espacios de estado de alta dimensión.

Aprendizaje por refuerzo multi-agente

Explorar escenarios con múltiples agentes interactuando podría proporcionar información sobre dinámicas colaborativas y competitivas, permitiendo el desarrollo de políticas más sofisticadas y robustas.

Diversidad de Entornos

Espacios de estado y acción continuos

Investigar entornos con espacios de estado y acción continuos, como Pendulum-v1 o LunarLanderContinuous-v2, podría ayudar a entender la escalabilidad y adaptabilidad de los algoritmos.

Aplicaciones del mundo real

Aplicar los algoritmos a problemas del mundo real, como el control de robots, la conducción autónoma o el comercio financiero, podría validar su utilidad práctica y efectividad fuera de entornos simulados.

Entornos estocásticos y parcialmente observables

Examinar el rendimiento en entornos estocásticos y parcialmente observables, como los Procesos de Decisión de Markov Parcialmente Observables, podría llevar al desarrollo de políticas más resilientes.

Optimización de hiperparámetros

Ajuste Automático de Hiperparámetros

Implementar técnicas de optimización automática de hiperparámetros, como la Optimización Bayesiana o Algoritmos Genéticos, podría identificar configuraciones de parámetros óptimas más eficientemente y mejorar el rendimiento del aprendizaje.

Tasas de aprendizaje adaptativas

Explorar esquemas de tasas de aprendizaje adaptativas podría mejorar la velocidad de convergencia y la estabilidad de los algoritmos, particularmente en entornos dinámicos y no estacionarios.

Robustez y generalización de políticas

Aprendizaje por transferencia

Investigar enfoques de aprendizaje por transferencia para aplicar el conocimiento adquirido en un entorno a otro podría mejorar la eficiencia del aprendizaje y la generalización de políticas a través de diferentes tareas.

Meta-Aprendizaje

Implementar técnicas de meta-aprendizaje para permitir que los algoritmos se adapten rápidamente a nuevos entornos o cambios dentro del mismo entorno podría mejorar su flexibilidad y aplicabilidad.

Robustez ante la incertidumbre

Desarrollar métodos para manejar la incertidumbre y la variabilidad en el entorno, como el aprendizaje por refuerzo robusto o enfoques sensibles al riesgo, podría mejorar la fiabilidad y seguridad de las políticas aprendidas.

Análisis de Desempeño y Comparativas

Análisis comparativos detallados

Realizar comparativas detalladas a través de una gama más amplia de entornos y algoritmos podría proporcionar una comprensión más detallada de las fortalezas y limitaciones de cada enfoque.

Evaluación a largo plazo

Realizar evaluaciones a largo plazo para evaluar la estabilidad y persistencia de las políticas aprendidas durante períodos extendidos podría asegurar su robustez y efectividad en aplicaciones del mundo real.

Enfoques interdisciplinarios

Incorporar conocimientos específicos del dominio

Incorporar conocimiento específico del dominio y heurísticas en el proceso de aprendizaje podría acelerar la convergencia y mejorar el rendimiento en aplicaciones especializadas.

Aprendizaje con Human-in-the-Loop

Desarrollar marcos para el aprendizaje por refuerzo con humanos en el bucle, donde la experiencia humana guía el proceso de aprendizaje, podría mejorar la calidad e interpretabilidad de las políticas aprendidas.

Al seguir estas líneas de acción futuras, se puede mejorar significativamente la robustez, eficiencia y aplicabilidad de los algoritmos de aprendizaje por refuerzo, allanando el camino para su adopción e impacto más amplio en diversos dominios.

Contribuciones de la investigación

Esta investigación ha demostrado la aplicabilidad exitosa de la inferencia causal en una variedad de contextos de Machine Learning, incluyendo la estimación de iluminancia, la detección de sombras, la toma de decisiones de agentes inteligentes y el aprendizaje por refuerzo. Esta amplia gama de aplicaciones destaca la versatilidad y utilidad de la inferencia causal en diferentes áreas de estudio.

12.1. Contribuciones metodológicas y empíricas

12.1.1. Avance en la Estimación de Iluminancia mediante Aprendizaje Profundo

Esta investigación ha demostrado el éxito de la aplicación de técnicas de Deep Learning para la estimación de iluminancia en escenarios 3D. Esta contribución sugiere que el aprendizaje profundo puede ser una herramienta valiosa para resolver problemas de estimación de iluminancia en diversas configuraciones de iluminación.

12.1.2. Fortalecimiento de la Detección de Sombras a través de la Inferencia Causal

La incorporación de la inferencia causal ha mejorado la capacidad de determinar la causa de fenómenos de sombra en imágenes. Esta contribución subraya cómo la inferencia causal puede fortalecer hipótesis y mejorar la comprensión de eventos complejos en el análisis de imágenes.

12.1.3. Mejora en la Toma de Decisiones de Agentes Inteligentes mediante Inferencia Causal

La integración de la inferencia causal ha proporcionado una ventaja significativa en la toma de decisiones de agentes inteligentes en entornos virtuales. Este hallazgo destaca el potencial transformador de incorporar el razonamiento causal en la ejecución de tareas complejas por parte de agentes inteligentes.

12.1.4. La integración de la inferencia causal en el muestreo de una Deep Q-Network ha mostrado resultados prometedores relacionados con su rendimiento en problemas de control clásico

Esta contribución resalta la importancia de considerar cuidadosamente el equilibrio entre exploración y explotación en el diseño de algoritmos de aprendizaje por refuerzo, apoyándose en un muestreo prioritario, de las experiencias aprendidas por un agente, basándose en la inferencia causal.

12.2. Contribuciones a la literatura

Las publicaciones derivadas de esta investigación han contribuido significativamente al cuerpo existente de literatura en el campo del Machine Learning y la inferencia causal, proporcionando nuevos conocimientos, métodos y perspectivas que avanzan en la comprensión y aplicación de estas áreas. Estas contribuciones se resumen a continuación:

Exploring Deep Learning Techniques for Illuminance Estimation

En esta publicación, exploramos en detalle el impacto de la integración de la inferencia causal en la estimación de iluminancia en escenarios 3D. Nuestros hallazgos han demostrado la viabilidad y eficacia de esta aproximación en la resolución de problemas de estimación de iluminancia en entornos complejos, lo que amplía el conjunto de herramientas disponibles para los investigadores y profesionales en el campo de la iluminación arquitectónica y el diseño de escenarios virtuales.

Journal: Expert Systems

EISSN: 1468-0394

JCR: Q2

DOI: <http://doi.org/10.1111/exsy.13559>

URL: <https://onlinelibrary.wiley.com/doi/10.1111/exsy.13559>

Causal Inference Applied to Explaining the Appearance of Shadow Phenomena in an Image

En esta publicación, presentamos un enfoque novedoso para la detección de sombras en imágenes mediante el uso de la inferencia causal. Nuestros resultados destacan la importancia de considerar la causalidad en la comprensión de fenómenos visuales complejos, y proporcionan nuevas perspectivas sobre cómo abordar este problema fundamental en el procesamiento de imágenes y la visión por computadora.

Journal: INFORMATICA

EISSN: 1822-8844

JCR: Q2

DOI <https://doi.org/10.15388/23-INFOR526>

URL: <https://informatica.vu.lt/journal/INFORMATICA/article/1302/info>

Approach from Artificial Intelligence to poorly predictive behaviors derived from artificial cognitive models

En esta publicación, propusimos generar modelos conductuales artificiales para determinar las condiciones en que estos demuestran comportamientos poco predictivos. El entorno virtual de pruebas, la arquitectura de los modelos y el desarrollo de software fueron posibles gracias a la interacción de Pogamut, Unreal Tournament, SOAR y Java. Los modelos conductuales propuestos en el experimento consistieron en máquinas de estados finitos y producciones derivadas del uso de la arquitectura cognitiva SOAR.

Journal: Tesis Psicológica

EISSN: 2422-0450

PUBLINDEX: Categoría B Equivalente a Q3 según el índice bibliográfico nacional del Ministerio de ciencia, tecnología e innovación de la Republica de Colombia.

URL: <https://revistas.libertadores.edu.co/index.php/TesisPsicologica/article/view/1101>

Intelligent Agents and Causal Inference: Enhancing Decision-Making through Causal Reasoning

En esta publicación, exploramos el papel de la inferencia causal en la toma de decisiones de agentes inteligentes en entornos virtuales. Nuestros hallazgos han demostrado que la integración de la causalidad en los modelos cognitivos de los agentes puede mejorar significativamente su capacidad para realizar tareas complejas y adaptarse a entornos cambiantes, lo que tiene importantes implicaciones para el diseño de sistemas de inteligencia artificial en una variedad de aplicaciones prácticas.

Journal: Applied Sciences

EISSN: 2076-3417

JCR: Q2

DOI: <https://doi.org/10.3390/app14093818>

URL: <https://www.mdpi.com/2076-3417/14/9/3818>

Integration of Causal Inference in the DQN Sampling Process for Classical Control Problems

En esta publicación, investigamos el impacto de la integración de la inferencia causal en el aprendizaje por refuerzo mediante el uso de Deep Q-Networks. Nuestros resultados sugieren que la incorporación de la causalidad en el proceso de muestreo puede mejorar el rendimiento y la eficacia de los algoritmos de aprendizaje por refuerzo en la resolución de problemas de control clásico, lo que abre nuevas oportunidades para aplicaciones en robótica, juegos y otras áreas relacionadas.

Journal: NEURAL COMPUTING & APPLICATIONS

EISSN: 1433-3058

JCR: Q2

DOI: Pendiente

URL: Pendiente

En conjunto, estas publicaciones representan contribuciones importantes a la literatura académica en el campo del Machine Learning y la inferencia causal, proporcionando nuevas ideas, métodos y resultados que avanzan en el conocimiento y la práctica en estas áreas.

12.2.1. Otras publicaciones

S-COGIT: A Natural Language Processing Tool for Linguistic Analysis of the Social Interaction Between Individuals with Attention-Deficit Disorder

Este artículo describe el diseño e implementación de una plataforma informática destinada a monitorear la interacción social en sujetos con trastorno por déficit de atención como una condición cognitiva especial. Aplicando algoritmos de Procesamiento de Lenguaje Natural (NLP), se pretende apoyar el monitoreo e intervención de individuos con necesidades especiales mediante el análisis del lenguaje, como en el caso del trastorno por déficit de atención.

Journal: Advances in Intelligent Systems and Computing

EISSN: 978-3-030-53036-5

SJR: Q4

DOI: https://doi.org/10.1007/978-3-030-53036-5_32

URL: https://link.springer.com/chapter/10.1007/978-3-030-53036-5_32

Bibliografía

- [1] J. Watchtel, “Report on digital sign brightness,” *The Veridian Group Final Report*, 2014.
- [2] H. A. E. Keitz, *Light Calculations and Measurements*. London: MACMILLAN AND CO LTD London, 2 ed., 1971.
- [3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.
- [4] M. D. Zeiler and R. Fergus, “Visualizing and understanding convolutional networks,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8689 LNCS, no. PART 1, pp. 818–833, 2014.
- [5] W. Saeed and C. Omlin, “Explainable AI (XAI): A Systematic Meta-Survey of Current Challenges and Future Opportunities,” *Knowledge-Based Systems*, vol. 263, p. 110273, 2021.
- [6] O. Martin, *Bayesian Analysis with Python*. Packt Publishing, 2018.
- [7] Y. B. He and Z. Geng, “Active learning of causal networks with intervention experiments and optimal designs,” *Journal of Machine Learning Research*, vol. 9, pp. 2523–2547, 2008.
- [8] M. J. Perry, “CAUSALITY: MODELS, REASONING, AND INFERENCE,” in *The Encyclopedia of Ancient History*, vol. 19, pp. 675–685, Cambridge University Press, 2003.
- [9] Y. Xin, N. Tagasovska, F. Perez-Cruz, and M. Raubal, “Vision Paper: Causal Inference for Interpretable and Robust Machine Learning in Mobility Analysis,” in *Proceedings of the 30th International Conference on Advances in Geographic Information Systems*, SIGSPATIAL ’22, (New York, NY, USA), Association for Computing Machinery, 2022.
- [10] D. Lopez-Paz, R. Nishihara, S. Chintala, B. Schölkopf, and L. Bottou, “Discovering causal signals in images,” *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 58–66, 2017.
- [11] K. Lebeda, S. Hadfield, and R. Bowden, “Exploring causal relationships in visual object tracking,” *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2015 Inter, no. November, pp. 3065–3073, 2015.

- [12] A. Fire and S.-C. Zhu, “Using Causal Induction in Humans to Learn and Infer Causality from Video,” *Proceedings of the Annual Meeting of the Cognitive Science Society*, vol. 35, pp. 2297–2302, 2013.
- [13] L. C. Pickup, Z. Pan, D. Wei, Y. Shih, C. Zhang, A. Zisserman, B. Scholkopf, and W. T. Freeman, “Seeing the arrow of time,” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 3, pp. 2043–2050, 2014.
- [14] Y. Zhao, I. Borovikov, F. de Mesentier Silva, A. Beirami, J. Rupert, C. Somers, J. Har-der, J. Kolen, J. Pinto, R. Pourabolghasem, J. Pestrak, H. Chaput, M. Sardari, L. Lin, S. Narravula, N. Aghdaie, and K. Zaman, “Winning is not everything: Enhancing game development with intelligent agents,” *IEEE Transactions on Games*, vol. 12, no. 2, pp. 199–212, 2020.
- [15] P. Palanisamy, “Multi-agent connected autonomous driving using deep reinforcement learning,” in *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–7, IEEE, 2020.
- [16] L. Lei, Y. Tan, K. Zheng, S. Liu, K. Zhang, and X. Shen, “Deep reinforcement learning for autonomous internet of things: Model, applications and challenges,” *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1722–1760, 2020.
- [17] S. J. Russell, *Artificial intelligence a modern approach*. Pearson Education, Inc., 2010.
- [18] A. Adadi and M. Berrada, “Peeking inside the black-box: A survey on explainable artificial intelligence (xai),” *IEEE Access*, vol. 6, pp. 52138–52160, 2018.
- [19] Tai, Lei and Liu, Ming, “A robot exploration strategy based on q-learning network,” in *2016 IEEE International Conference on Real-time Computing and Robotics (RCAR)*, pp. 57–62, IEEE, 2016.
- [20] W. Chen, X. Qiu, T. Cai, H.-N. Dai, Z. Zheng, and Y. Zhang, “Deep reinforcement learning for Internet of Things: A comprehensive survey,” *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 1659–1692, 2021.
- [21] K. Shao, Z. Tang, Y. Zhu, N. Li, and D. Zhao, “A survey of deep reinforcement learning in video games,” *arXiv preprint arXiv:1912.10944*, 2019.
- [22] S. K. Zhou, H. N. Le, K. Luu, H. V. Nguyen, and N. Ayache, “Deep reinforcement learning in medical imaging: A literature review,” *Medical image analysis*, vol. 73, p. 102193, 2021.
- [23] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.
- [24] Y. Huang, *Deep Q-Networks*, pp. 135–160. Singapore: Springer Singapore, 2020.
- [25] K. Mnih, Volodymyr Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, “Playing atari with deep reinforcement learning,” *arXiv preprint arXiv:1312.5602*, pp. 1–9, 2013.

- [26] P. Wawrzyński, “Real-time reinforcement learning by sequential actor–critics and experience replay,” *Neural networks*, vol. 22, no. 10, pp. 1484–1497, 2009.
- [27] H. Van Hasselt, A. Guez, and D. Silver, “Deep reinforcement learning with double q-learning,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 30, 2016.
- [28] A. W. Moore and C. G. Atkeson, “Prioritized sweeping: Reinforcement learning with less data and less time,” *Machine learning*, vol. 13, pp. 103–130, 1993.
- [29] H. Van Seijen and R. S. Sutton, “Efficient planning in MDPs by small backups,” in *Proc. 30th Int. Conf. Mach. Learn.*, vol. 28, pp. 1–9, 2013.
- [30] C. Tessler, S. Givony, T. Zahavy, D. Mankowitz, and S. Mannor, “A deep hierarchical approach to lifelong learning in minecraft,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 31, 2017.
- [31] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.
- [32] G. Fernández-Vizcaíno and F. J. Gallego-Durán, “Ajustando q-learning para generar jugadores automáticos: un ejemplo basado en atari breakout,” in *CoSECivi*, pp. 77–88, 2016.
- [33] C. Janiesch, P. Zschech, and K. Heinrich, “Machine learning and deep learning,” *Electronic Markets*, vol. 31, no. 3, pp. 685–695, 2021.
- [34] P. Bertens, A. Guitart, P. P. Chen, and A. Perianez, “A machine-learning item recommendation system for video games,” in *2018 IEEE Conference on Computational Intelligence and Games (CIG)*, pp. 1–4, IEEE, 2018.
- [35] B. J. Geisler, *An Empirical Study of Machine Learning Algorithms Applied to Modeling Player Behavior in a “First Person Shooter” Video Game*. PhD thesis, Citeseer, 2002.
- [36] J.-A. Hitar-Garcia, L. Moran-Fernandez, and V. Bolon-Canedo, “Machine learning methods for predicting league of legends game outcome,” *IEEE Transactions on Games*, 2022.
- [37] F. Doshi-Velez and B. Kim, “Towards a rigorous science of interpretable machine learning,” Feb. 2017.
- [38] Z. C. Lipton, D. C. Kale, C. Elkan, and R. Wetzal, “Learning to diagnose with LSTM recurrent neural networks,” Nov. 2015.
- [39] M.-I. Mahraz, L. Benabbou, and A. Berrado, “Machine learning in supply chain management: A systematic literature review,” *International Journal of Supply and Operations Management*, vol. 9, no. 4, pp. 398–416, 2022.
- [40] J. Pearl, *Causality*. Cambridge university press, 2009.
- [41] J. Peters, D. Janzing, and B. Schölkopf, *Elements of Causal Inference*. 2017.
- [42] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

- [43] R. Xu, D. Wunsch II, and R. Frank, “Inference of genetic regulatory networks with recurrent neural network models using particle swarm optimization,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 4, no. 4, pp. 681–692, 2007.
- [44] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, *et al.*, “Mastering the game of go without human knowledge,” *nature*, vol. 550, no. 7676, pp. 354–359, 2017.
- [45] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to linear regression analysis*. John Wiley & Sons, 2021.
- [46] T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman, *The elements of statistical learning: data mining, inference, and prediction*, vol. 2. Springer, 2009.
- [47] M. Praveena and V. Jaiganesh, “A literature review on supervised machine learning algorithms and boosting process,” *International Journal of Computer Applications*, vol. 169, no. 8, pp. 32–35, 2017.
- [48] P. Domingos, “A few useful things to know about machine learning,” *Communications of the ACM*, vol. 55, no. 10, pp. 78–87, 2012.
- [49] S. Shalev-Shwartz and S. Ben-David, *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.
- [50] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*, vol. 16. Massachusetts: The MIT Press, 2012.
- [51] K. Hsu, S. Levine, and C. Finn, “Unsupervised learning via meta-learning,” *arXiv preprint arXiv:1810.02334*, 2018.
- [52] C. C. Aggarwal, “Outlier analysis second edition,” 2016.
- [53] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: A survey,” *ACM computing surveys (CSUR)*, vol. 41, no. 3, pp. 1–58, 2009.
- [54] A. Likas, N. Vlassis, and J. J. Verbeek, “The global k-means clustering algorithm,” *Pattern recognition*, vol. 36, no. 2, pp. 451–461, 2003.
- [55] K. Khan, S. U. Rehman, K. Aziz, S. Fong, and S. Sarasvady, “Dbscan: Past, present and future,” in *The fifth international conference on the applications of digital information and web technologies (ICADIWT 2014)*, pp. 232–238, IEEE, 2014.
- [56] F. Murtagh and P. Contreras, “Algorithms for hierarchical clustering: an overview,” *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 2, no. 1, pp. 86–97, 2012.
- [57] T. Kurita, “Principal component analysis (pca),” *Computer Vision: A Reference Guide*, pp. 1–4, 2019.
- [58] G. C. Linderman and S. Steinerberger, “Clustering with t-sne, provably,” *SIAM journal on mathematics of data science*, vol. 1, no. 2, pp. 313–332, 2019.

- [59] I. Syarif, A. Prugel-Bennett, and G. Wills, “Unsupervised clustering approach for network anomaly detection,” in *Networked Digital Technologies: 4th International Conference, NDT 2012, Dubai, UAE, April 24-26, 2012. Proceedings, Part I 4*, pp. 135–145, Springer, 2012.
- [60] F. M. Nafie Ali and A. A. Mohamed Hamed, “Usage apriori and clustering algorithms in weka tools to mining dataset of traffic accidents,” *Journal of Information and Telecommunication*, vol. 2, no. 3, pp. 231–245, 2018.
- [61] I. Syukra, A. Hidayat, and M. Z. Fauzi, “Implementation of k-medoids and fp-growth algorithms for grouping and product offering recommendations,” *Indonesian Journal of Artificial Intelligence and Data Mining*, vol. 2, no. 2, pp. 107–115, 2019.
- [62] C. M. Bishop, “Pattern recognition and machine learning,” *Springer google schola*, vol. 2, pp. 645–678, 2006.
- [63] J. Kober, J. A. Bagnell, and J. Peters, “Reinforcement learning in robotics: A survey,” *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1238–1274, 2013.
- [64] S. Levine, C. Finn, T. Darrell, and P. Abbeel, “End-to-end training of deep visuomotor policies,” *Journal of Machine Learning Research*, vol. 17, no. 39, pp. 1–40, 2016.
- [65] B. Zhang, W. Hu, D. Cao, Q. Huang, Z. Chen, and F. Blaabjerg, “Deep reinforcement learning-based approach for optimizing energy conversion in integrated electrical and heating system with renewable energy,” *Energy conversion and management*, vol. 202, p. 112199, 2019.
- [66] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, “Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor,” in *International conference on machine learning*, pp. 1861–1870, PMLR, 2018.
- [67] C. J. Watkins and P. Dayan, “Q-learning,” *Machine learning*, vol. 8, pp. 279–292, 1992.
- [68] G. A. Rummery and M. Niranjan, *On-line Q-learning using connectionist systems*, vol. 37. University of Cambridge, Department of Engineering Cambridge, UK, 1994.
- [69] R. J. Williams, “Simple statistical gradient-following algorithms for connectionist reinforcement learning,” *Machine learning*, vol. 8, pp. 229–256, 1992.
- [70] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, “Trust region policy optimization,” in *International conference on machine learning*, pp. 1889–1897, PMLR, 2015.
- [71] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, “Asynchronous methods for deep reinforcement learning,” in *International conference on machine learning*, pp. 1928–1937, PMLR, 2016.
- [72] L. P. Kaelbling, M. L. Littman, and A. W. Moore, “Reinforcement learning: A survey,” *Journal of artificial intelligence research*, vol. 4, pp. 237–285, 1996.
- [73] T. Heskes, E. Sijben, I. G. Bucur, and T. Claassen, “Causal shapley values: Exploiting causal knowledge to explain individual predictions of complex models,” *Advances in neural information processing systems*, vol. 33, pp. 4778–4789, 2020.

- [74] B. Van Breugel, T. Kyono, J. Berrevoets, and M. Van der Schaar, “Decaf: Generating fair synthetic data using causally-aware generative networks,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 22221–22233, 2021.
- [75] M. A. Hernán and J. M. Robins, *Causal Inference: What If*. Taylor & Francis Group, i ed., 2020.
- [76] L. Bellia, M. Musto, and G. Spada, “Illuminance measurements through hdr imaging photometry in scholastic environment,” *Energy and buildings*, vol. 43, no. 10, pp. 2843–2849, 2011.
- [77] M. Wang and Z. Shang, “Deep separable convolution neural network for illumination estimation,” in *International Conference on Agents and Artificial Intelligence*, vol. 11, pp. 879–886, 2019.
- [78] M. A. Gardner, K. Sunkavalli, E. Yumer, X. Shen, E. Gambaretto, C. Gagne, and J. F. Lalonde, “Learning to predict indoor illumination from a single image,” *ACM Transactions on Graphics*, vol. 36, no. 6, 2017.
- [79] S. Song and T. Funkhouser, “Neural illumination: Lighting prediction for indoor environments,” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2019-June, pp. 6911–6919, 2019.
- [80] K. Zhang, X. Li, X. Jin, B. Liu, X. Li, and H. Sun, “Outdoor illumination estimation via all convolutional neural networks,” *Computers and Electrical Engineering*, vol. 90, no. 7, 2021.
- [81] M. S. El-nasr, “Intellegent lighting for game environments,” *Journal of Game Development*, vol. 1, no. 2, pp. 17–50, 2005.
- [82] K. Koscevic, M. Subasic, and S. Loncaric, “Deep learning-based illumination estimation using light source classification,” *IEEE Access*, vol. 8, no. 1, pp. 84239–84247, 2020.
- [83] P. Kan and H. Kafumann, “Deeplight: light source estimation for augmented reality using deep learning,” *The Visual Computer*, vol. 35, no. 6-8, pp. 873–883, 2019.
- [84] M. Yang, X. Nie, and R. W. Liu, “Coarse-to-fine luminance estimation for low-light image enhancement in maritime video surveillance,” in *2019 IEEE Intelligent Transportation Systems Conference, ITSC 2019*, pp. 299–304, IEEE, 2019.
- [85] S. U.C., S. Aaqib, B. Sachin, K. Thijs W., and O. Tanir, “Lumnet : Learning to estimate vertical visual field luminance for adaptive lighting control,” *ACM Interact. Mob. Wearable Ubiquitous Technol*, vol. 5, no. 2, 2021.
- [86] H. Liu, S. Chen, N. Zheng, Y. Wang, J. Ge, K. Ding, Z. Guo, W. Li, and J. Lan, “Ground pedestrian and vehicle detections using imaging environment perception mechanisms and deep learning networks,” *Electronics (Switzerland)*, vol. 11, no. 12, 2022.

- [87] M. Kayakus and K. K. Cevik, “Estimating luminance measurements in road lighting by deep learning method,” in *Artificial Intelligence and Applied Mathematics in Engineering Problems* (D. J. Hemanth and U. Kose, eds.), (Cham), pp. 940–948, Springer International Publishing, 2020.
- [88] H. A. Sial, R. Baldrich, M. Vanrell, and D. Samaras, “Light direction and color estimation from single image with deep regression,” *London Imaging Meeting*, vol. 1, no. 1, pp. 139–143, 2020.
- [89] M. Li, J. Guo, X. Cui, R. Pan, Y. Guo, C. Wang, P. Yu, and F. Pan, “Deep spherical gaussian illumination estimation for indoor scene,” in *Proceedings of the ACM Multimedia Asia*, pp. 1–6, 2019.
- [90] M. Miki, T. Hiroyasu, K. Imazato, and M. Yonezawa, “Intelligent lighting control using correlation coefficient between luminance and illuminance,” *Proc. IASTED Intelligent Systems and Control*, vol. 497, no. 078, pp. 31–36, 2005.
- [91] Y. Xin, N. Tagasovska, F. Perez-Cruz, and M. Raubal, “Vision paper: causal inference for interpretable and robust machine learning in mobility analysis,” in *GIS: Proceedings of the ACM International Symposium on Advances in Geographic Information Systems*, SIGSPATIAL ’22, (New York, NY, USA), Association for Computing Machinery, 2022.
- [92] M. Levoy, “Efficient Ray Tracing of Volume Data,” *ACM Transactions on Graphics (TOG)*, vol. 9, no. 3, pp. 245–261, 1990.
- [93] T.-M. Li, M. Aittala, F. Durand, and J. Lehtinen, “Differentiable monte carlo ray tracing through edge sampling,” *ACM Trans. Graph.*, vol. 37, dec 2018.
- [94] S. Maes, S. Meganck, and B. Manderick, “Inference in multi-agent causal models,” *International Journal of Approximate Reasoning*, vol. 46, no. 2, pp. 274–299, 2007.
- [95] T. Chakraborti, A. Kulkarni, S. Sreedharan, D. E. Smith, and S. Kambhampati, “Explicability? legibility? predictability? transparency? privacy? security? The emerging landscape of interpretable agent behavior,” *Proceedings International Conference on Automated Planning and Scheduling, ICAPS*, no. Icaps, pp. 86–96, 2019.
- [96] E. Neufeld and S. Kristtorn, “Does non-correlation imply non-causation?,” *International Journal of Approximate Reasoning*, vol. 46, no. 2, pp. 257–273, 2007.
- [97] G. Jezic, M. Kusek, I. Lovrek, R. J. Howlett, and L. C. Jain, “Preface,” *Advances in Intelligent Systems and Computing*, vol. 296, no. ii, pp. 5–6, 2014.
- [98] S. Meganck, S. Maes, B. Manderick, and P. Leray, “Distributed learning of multi-agent causal models,” *Proceedings - 2005 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IAT’05*, vol. 2005, no. 1, pp. 285–288, 2005.
- [99] I. Dasgupta, J. Wang, S. Chiappa, J. Mitrovic, P. Ortega, D. Raposo, E. Hughes, P. Battaglia, M. Botvinick, and Z. Kurth-Nelson, “Causal Reasoning from Meta-reinforcement Learning,” *arXiv preprint*, 2019.

- [100] X. Zhang, Y. Liu, W. Li, and C. Gong, “Pruning the communication bandwidth between reinforcement learning agents through causal inference: An innovative approach to designing a smart grid power system,” *Sensors*, vol. 22, p. 7785, 2022.
- [101] C. Miao, A. Goh, Y. Miao, and Z. Yang, “A dynamic inference model for intelligent agents,” *International Journal of Software Engineering and Knowledge Engineering*, vol. 11, no. 5, pp. 509–528, 2001.
- [102] H. G. Ceballos and F. J. Cantu, “Modelling intelligent agents through causality theory,” *Proceedings - 2007 6th Mexican International Conference on Artificial Intelligence, Special Session, MICAI 2007*, pp. 201–210, 2007.
- [103] V. Mascardi, D. Demergasso, and D. Ancona, “Languages for programming BDI-style agents: An overview,” *WOA 2005 - 6th AI*IA/TABOO Joint Workshop "From Objects to Agents": Simulation and Formal Analysis of Complex Systems*, no. L, pp. 9–15, 2005.
- [104] D. D. Jensen, “Improving Causal Inference by Increasing Model Expressiveness,” *35th AAAI Conference on Artificial Intelligence, AAAI 2021*, vol. 17A, no. Pearl, pp. 15053–15057, 2021.
- [105] J. Renz, X. Y. Ge, R. Verma, and P. Zhang, “Angry birds as a challenge for artificial intelligence,” *30th AAAI Conference on Artificial Intelligence, AAAI 2016*, pp. 4338–4339, 2016.
- [106] N. Tziortziotis, G. Papagiannis, and K. Blekas, “A Bayesian Ensemble Regression Framework on the Angry Birds Game,” *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 8, no. 2, pp. 104–115, 2016.
- [107] C. H. H. Yang, I. T. D. Hung, Y. Ouyang, and P. Y. Chen, “Training a Resilient Q-network against Observational Interference,” *Proceedings of the 36th AAAI Conference on Artificial Intelligence, AAAI 2022*, vol. 36, pp. 8814–8822, 2022.
- [108] L. Bottou, J. Peters, J. Quiñonero-Candela, D. X. Charles, D. M. Chickering, E. Portugaly, D. Ray, P. Simard, and E. Snelson, “Counterfactual Reasoning and Learning Systems: The Example of Computational Advertising,” *Journal of Machine Learning Research*, vol. 14, no. 11, 2013.
- [109] M. Jaderberg, V. Mnih, W. M. Czarnecki, T. Schaul, J. Z. Leibo, D. Silver, and K. Kavukcuoglu, “Reinforcement learning with unsupervised auxiliary tasks,” 2016.
- [110] M. Laskin, D. Yarats, H. Liu, K. Lee, A. Zhan, K. Lu, C. Cang, L. Pinto, and P. Abbeel, “Urlb: Unsupervised reinforcement learning benchmark,” 2021.
- [111] “Comparative study of semantic mapping images,”
- [112] A. Sanders, *An introduction to Unreal engine 4*. CRC Press, 2016.
- [113] J. Lee, *Learning unreal engine game development*. Packt Publishing Ltd, 2016.
- [114] N. Ketkar and N. Ketkar, “Introduction to keras,” *Deep learning with python: a hands-on introduction*, pp. 97–111, 2017.

- [115] B. Pang, E. Nijkamp, and Y. N. Wu, “Deep learning with tensorflow: A review,” *Journal of Educational and Behavioral Statistics*, vol. 45, no. 2, pp. 227–248, 2020.
- [116] S. Raschka and V. Mirjalili, *Python machine learning: Machine learning and deep learning with Python, scikit-learn, and TensorFlow 2*. Packt Publishing Ltd, 2019.
- [117] J. Wachtel, “Report on digital sign brightness,” tech. rep., The Veridian Group, Inc., 2014.
- [118] A. H. Ismail, M. S. M. Azmi, M. A. Hashim, M. N. Ayob, M. S. M. Hashim, and H. B. Hassrizal, “Development of a webcam based lux meter,” in *2013 IEEE Symposium on Computers & Informatics (ISCI)*, pp. 70–74, IEEE, 2013.
- [119] J. Pearl, “The seven tools of causal inference, with reflections on machine learning,” *Communications of the ACM*, vol. 62, no. 3, pp. 54–60, 2019.
- [120] J. Pearl and D. Mackenzie, *The Book of Why*. New York: Basic Books, i ed., 2018.
- [121] S. Chiappa and W. S. Isaac, “A Causal Bayesian Networks Viewpoint on Fairness,” *Springer, Cham*, pp. 1–19, 2019.
- [122] R. Guo, L. Cheng, J. Li, P. R. Hahn, and H. Liu, “A Survey of Learning Causality with Data: Problems and Methods,” *ACM Computing Surveys*, vol. 53, no. 4, pp. 1–37, 2020.
- [123] L. Dague and J. N. Lahey, “Causal inference methods: Lessons from applied microeconomics,” *Journal of Public Administration Research and Theory*, vol. 29, no. 3, pp. 511–529, 2019.
- [124] G. Bradski, “The OpenCV Library,” *Dr. Dobb’s Journal of Software Tools*, 2000.
- [125] S. Van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, and T. Yu, “scikit-image: image processing in python,” *PeerJ*, vol. 2, p. e453, 2014.
- [126] Epic Games, “Unreal Engine,” 2019.
- [127] J. Gemrot, R. Kadlec, M. Bida, O. Burkert, R. Pibil, J. Havlicek, L. Zemcak, J. Simlovic, R. Vansa, M. Stolba, T. Plch, and C. Brom, “Pogamut 3 Can Assist Developers in Building AI (Not Only) for Their Videogame Agents,” *Springer*, vol. 7522, no. Agents for Games and Simulations, pp. 1 – 15, 2012.
- [128] D. Adams, “Unreal Tournament 2004 Review Does Epic’s latest do the series proud?,” 2018.
- [129] J. Pearl, M. Glymour, and N. P. Jewell, *CAUSAL INFERENCE IN STATISTICS A PRIMER*, vol. 61. John Wiley & Sons, 1986.
- [130] N. Pawlowski, D. C. Castro, and B. Glocker, “Deep structural causal models for tractable counterfactual inference,” *Advances in Neural Information Processing Systems*, vol. 2020-Decem, no. NeurIPS, 2020.
- [131] A. Ankan and A. Panda, “pgmpy: Probabilistic graphical models using python,” in *Proceedings of the 14th Python in Science Conference (SCIPY 2015)*, Citeseer, 2015.

- [132] Beaumont, Paul and Horsburgh, Ben and Pilgerstorfer, Philip and Droth, Angel and Oentaryo, Richard and Ler, Steven and Nguyen, Hiep and Ferreira, Gabriel Azevedo and Patel, Zain and Leong, Wesley, “Causalnex,” 2021.
- [133] M. Stephenson and J. Renz, “Creating a hyper-agent for solving angry birds levels,” *Proceedings of the 13th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, AIIDE 2017*, pp. 234–240, 2017.
- [134] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, “OpenAI Gym,” *arXiv preprint arXiv:1606.01540*, pp. 1–4, 2016.
- [135] PyTorch Contributors, “Pytorch.” <https://github.com/pytorch/pytorch>, 2016. Versión 1.x.
- [136] scikit-learn Contributors, “scikit-learn.” <https://github.com/scikit-learn/scikit-learn>, 2011. Versión 0.24.2.
- [137] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. Carey, Í. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, “SciPy 1.0: Fundamental algorithms for scientific computing in python,” *Nature Methods*, vol. 17, no. 3, pp. 261–272, 2020.
- [138] J. Fan, Z. Wang, Y. Xie, and Z. Yang, “A Theoretical Analysis of Deep Q-Learning,” *Learning for dynamics and control*, vol. 120, no. 1995, pp. 1–4, 2020.
- [139] S. Thrun and M. L. Littman, “Reinforcement learning: an introduction,” *AI Magazine*, vol. 21, no. 1, pp. 103–103, 2000.
- [140] M. Wunder, M. Littman, and M. Babes, “Classes of multiagent q-learning dynamics with epsilon-greedy exploration,” in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pp. 1167–1174, 2010.
- [141] S. Zhang and R. S. Sutton, “A deeper look at experience replay,” *arXiv preprint arXiv:1712.01275*, 2017.
- [142] J. M. Bland and D. G. Altman, “Multiple significance tests: the Bonferroni method,” *Bmj*, vol. 310, no. 6973, p. 170, 1995.
- [143] M. Caliendo and S. Kopeinig, “Some practical guidance for the implementation of propensity score matching,” *Journal of economic surveys*, vol. 22, no. 1, pp. 31–72, 2008.
- [144] A. Maydeu-Olivares, D. L. Coffman, C. García-Forero, and D. Gallardo-Pujol, “Hypothesis testing for coefficient alpha: An sem approach,” *Behavior research methods*, vol. 42, no. 2, pp. 618–625, 2010.
- [145] Y. Guts, “cartpole-q-learning.” <https://github.com/YuriyGuts/cartpole-q-learning>, June 2023.

- [146] R. Liu and J. Zou, “The effects of memory replay in reinforcement learning,” in *2018 56th annual allerton conference on communication, control, and computing (Allerton)*, pp. 478–485, IEEE, 2018.
- [147] Y. Yao, L. Rosasco, and A. Caponnetto, “On early stopping in gradient descent learning,” *Constructive Approximation*, vol. 26, no. 2, pp. 289–315, 2007.
- [148] V. N. Vapnik, V. Vapnik, *et al.*, “Statistical learning theory,” 1998.
- [149] Y. Wang, N. Wagner, and J. M. Rondinelli, “Symbolic regression in materials science,” *MRS Communications*, vol. 9, no. 3, pp. 793–805, 2019.
- [150] G. Lample and F. Charton, “Deep learning for symbolic mathematics,” *arXiv preprint arXiv:1912.01412*, 2019.

Código Fuente del Software Desarrollado

A.1. Introducción

En este apéndice se proporciona información sobre los códigos fuente del software desarrollado para esta tesis doctoral. Cada uno de los softwares ha sido registrado en Zenodo y está disponible a través de un DOI (Digital Object Identifier) para garantizar su accesibilidad y permanencia a largo plazo.

A.2. Software 1: ML-Atlas

Descripción

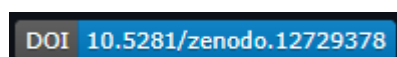
El desarrollo de técnicas avanzadas de machine learning ha revolucionado diversas áreas del conocimiento, incluyendo la física experimental. Este software ha sido diseñado para experimentar con una amplia gama de técnicas de machine learning con el objetivo de descubrir y formular leyes físicas a partir de datos experimentales. Al aprovechar la capacidad de procesamiento y análisis de estas técnicas, es posible identificar patrones complejos y relaciones subyacentes en los datos que de otro modo podrían pasar desapercibidos.

El software implementa una variedad de algoritmos de machine learning, desde métodos tradicionales como regresión lineal y árboles de decisión, hasta enfoques más sofisticados como redes neuronales profundas y técnicas de aprendizaje no supervisado. La flexibilidad y el poder de estas herramientas permiten abordar problemas de diferentes naturalezas y escalas, ofreciendo una visión integral del proceso experimental.

Repositorio GitHub

<https://github.com/jvelez-dev/ML-Atlas.git>

DOI en Zenodo



A.3. Software 2: DL-Illuminance

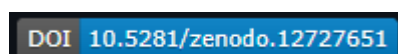
Descripción

Este software de redes neuronales está diseñado para predecir los valores de iluminancia en un escenario 3D configurable con distintas distribuciones de fuentes lumínicas, tanto simétricas como asimétricas. Utilizando modelos avanzados de redes neuronales convolucionales (CNN) y perceptrones multicapa (MLP), el software analiza y procesa datos complejos para generar predicciones precisas sobre la distribución de la luz en el espacio. Este enfoque permite a los usuarios explorar cómo diferentes configuraciones de iluminación afectan la iluminancia del entorno, facilitando la planificación y optimización de sistemas de iluminación en diversas aplicaciones.

Repositorio GitHub

<https://github.com/jvelez-dev/DL-Illuminance.git>

DOI en Zenodo



A.4. Software 3: SR-Illuminance

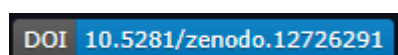
Descripción

Este software de regresión simbólica utiliza la potencia de cálculo de un clúster de máquinas para descubrir las ecuaciones que gobiernan conjuntos de datos de iluminancia en un escenario 3D con diversas configuraciones de iluminación, tanto simétricas como asimétricas. A través de técnicas avanzadas de regresión simbólica, el software analiza y modela la distribución de la luz en el espacio, identificando patrones y relaciones subyacentes en los datos. Este enfoque permite una comprensión más profunda y precisa de cómo diferentes configuraciones de iluminación afectan la iluminancia en el entorno, facilitando así la optimización y el diseño de sistemas de iluminación eficientes y efectivos.

Repositorio GitHub

https://github.com/jvelez-dev/SR_Illuminance.git

DOI en Zenodo



A.5. Software 4: CI-Shadow

Descripción

Este software permite la experimentación con técnicas de inferencia causal, con el objetivo de identificar y analizar la causa de la aparición de una sombra en una imagen. A través de este enfoque, se busca aplicar métodos avanzados de inferencia causal para determinar las variables y factores que contribuyen a la presencia de sombras, lo cual permite una comprensión más profunda de los procesos subyacentes en la formación de imágenes. El experimento no solo se centra en la detección de sombras, sino también en la identificación de las relaciones causales entre distintos elementos de la imagen, proporcionando así un marco sólido para futuros estudios y aplicaciones en el campo del procesamiento de imágenes y la visión por computadora. Este trabajo demuestra cómo la inferencia causal puede ser una herramienta poderosa para mejorar el análisis y la interpretación de datos visuales, abriendo nuevas posibilidades para la investigación y el desarrollo en esta área.

Repositorio GitHub

<https://github.com/jvelez-dev/CI-Shadow.git>

DOI en Zenodo



A.6. Software 5: CI-Based Behaviour

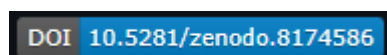
Descripción

En este repositorio se presenta un experimento en el que un agente inteligente utiliza la inferencia causal para guiar su comportamiento. El escenario del experimento es un entorno de videojuego FPS (Unreal Tournament 2004), en el cual se ha diseñado un escenario específico denominado "TestEnvironment". En este entorno, un agente llamado "GuardBOT" se posiciona en una ubicación determinada, mientras que otro agente, "CausalBOT", tiene la tarea de tomar la posición del GuardBOT sin ser detectado. Este experimento demuestra cómo la inferencia causal puede ser aplicada en entornos complejos para mejorar la toma de decisiones de agentes inteligentes.

Repositorio GitHub

https://github.com/jvelez-dev/Causal_Inference.git

DOI en Zenodo



A.7. Software 6: CI-Based Sampling

Descripción

Al incorporar el muestreo basado en inferencia causal en el buffer de repetición de un DQN, se buscó mejorar la eficiencia y efectividad del aprendizaje en la resolución del problema clásico de control. Este enfoque permite priorizar experiencias con mayor probabilidad de tener un impacto causal en el proceso de toma de decisiones del agente, conduciendo a un aprendizaje más enfocado e informado. En contraste, el muestreo aleatorio tradicional en el buffer de repetición de una Deep Q-Network carece de esta priorización, lo que puede resultar en una convergencia más lenta y un rendimiento subóptimo.

Repositorio GitHUB

https://github.com/jvelez-dev/CI-based_Sampling.git

DOI en Zenodo

DOI 10.5281/zenodo.8421523

Regresión simbólica para determinar las ecuaciones que gobiernan los conjuntos de datos de iluminancia

B.1. Introducción

Durante el desarrollo de modelos predictivos de Deep Learning, específicamente redes neuronales convolucionales y feed-forward, surgió un interés por experimentar con la regresión simbólica para descubrir las ecuaciones subyacentes en los conjuntos de datos utilizados para entrenar estos modelos. La regresión simbólica, a diferencia de otros métodos de aprendizaje automático, busca identificar expresiones matemáticas explícitas que describan las relaciones presentes en los datos, proporcionando así una comprensión más intuitiva y directa de los procesos estudiados.

Aprovechando la capacidad de cómputo disponible en el Instituto Universitario de Investigación en Ingeniería de Aragón (I3A), fue posible procesar grandes volúmenes de datos y realizar cálculos complejos en paralelo, lo que permitió la generación de dos modelos de regresión simbólica. Estos modelos lograron identificar con un alto nivel de precisión las ecuaciones que gobiernan los datos.

El éxito de esta integración no solo demostró la eficacia de la regresión simbólica en la identificación de patrones y relaciones en datos complejos, sino que también abrió nuevas posibilidades para el análisis y la interpretación de datos en diversos campos de estudio. Al descubrir las ecuaciones subyacentes, se obtiene una visión más profunda de los mecanismos que generan los datos, lo cual es fundamental para avanzar en la comprensión científica y el desarrollo tecnológico. Este experimento ha establecido un precedente para futuras investigaciones que busquen combinar técnicas de Deep Learning y regresión simbólica, maximizando así el valor de los recursos computacionales y el conocimiento derivado de los datos.

B.2. Método

B.2.1. Obtención y Preprocesamiento de Datos

Los conjuntos de datos utilizados en este estudio fueron seleccionados de diversas fuentes relevantes para el análisis de iluminancia en escenarios 3D con diferentes configuraciones de

iluminación. Estos datos incluyeron medidas de intensidad lumínica en diversas posiciones y condiciones, tanto simétricas como asimétricas. Los datos fueron preprocesados para eliminar valores atípicos y normalizados para asegurar una coherencia en el entrenamiento de los modelos.

B.2.2. Desarrollo de Modelos Predictivos de Deep Learning

Inicialmente, se desarrollaron modelos predictivos utilizando redes neuronales convolucionales (CNN) y redes feed-forward. Estos modelos fueron entrenados con los conjuntos de datos preprocesados para identificar patrones y relaciones complejas dentro de los datos. Las arquitecturas de las redes neuronales fueron diseñadas y ajustadas mediante técnicas de validación cruzada y optimización de hiperparámetros para maximizar la precisión de las predicciones. El detalle de esta experimentación puede verse en la sección 5 de este documento.

B.2.3. Implementación de la Regresión Simbólica

Una vez que los modelos de Deep Learning demostraron ser efectivos en la predicción de iluminancia, se procedió a implementar técnicas de regresión simbólica para descubrir las ecuaciones subyacentes en los datos. La regresión simbólica se llevó a cabo utilizando herramientas especializadas que permiten la búsqueda de expresiones matemáticas explícitas que mejor describen las relaciones en los datos.

Parafraseando la explicación proporcionada por [149], el procedimiento comienza definiendo una población con un conjunto de nodos generados aleatoriamente, formando árboles de diferentes tamaños y estructuras. Luego, se inicia el proceso de selección, en el cual se evalúa la aptitud de cada solución en la población comparando su valor de aptitud con el valor esperado. A continuación, la generación actual evoluciona aplicando operaciones genéticas (cruces, mutaciones y reproducción) a los individuos. Dependiendo del valor de aptitud de un individuo, será seleccionado como progenitor; generalmente, cuanto mayor sea el valor de aptitud, mayor será la probabilidad de ser seleccionado para la reproducción (supervivencia del más apto). Las operaciones genéticas que se pueden realizar en la evolución son el cruce, la mutación y la reproducción; la primera toma dos ganadores del proceso de selección como progenitores para reproducir su descendencia; la segunda toma solo una estructura progenitora y reemplaza aleatoriamente un subárbol con otra estructura generada aleatoriamente; y la tercera duplica el programa seleccionado e inserta directamente su descendencia en la siguiente generación. Los recién nacidos se añaden a la siguiente generación, y la siguiente generación pasa nuevamente por el proceso de evaluación de aptitud y selección natural hasta que el valor de aptitud alcance un cierto criterio o se alcance un número máximo de generaciones; así, el resultado será el árbol de expresión matemática correspondiente a la ecuación [150].

La asignación de los valores paramétricos, al igual que en las redes neuronales, fue experimental, realizando variaciones en los valores de los parámetros del algoritmo genético para seleccionar el modelo que presentó la mejor medida de error. La Tabla B.1 muestra el valor de cada parámetro sujeto a variación en el proceso de ajuste, destacando en este caso, que el número de trabajos se configuró de manera que el proceso de evolución se ejecutara en paralelo como una contramedida al tiempo de ejecución implicado por la naturaleza iterativa del algoritmo.

Tabla B.1. Parametrización del regresor simbólico

Parametro	Valor
Tamaño población	1000
Generaciones	500
Criterio de parada	0.01
Conjunto de funciones	'add', 'mul', 'sub', 'div', 'sqrt'
Número de trabajos	-1
Métrica	RMSE

Integración con la Capacidad de Cómputo del I3A

Aprovechando la capacidad de cómputo avanzada del Instituto Universitario de Investigación en Ingeniería de Aragón (I3A), se implementó un entorno de procesamiento paralelo. Esto permitió manejar grandes volúmenes de datos y realizar cálculos complejos eficientemente. Los recursos computacionales del I3A fueron esenciales para la generación y evaluación de múltiples modelos de regresión simbólica en paralelo. El código fuente reposa en el repositorio https://github.com/jvelez-dev/SR_Illuminance.git.

Se generaron dos modelos de regresión simbólica que fueron evaluados en términos de precisión y capacidad para identificar las ecuaciones subyacentes en los datos. La evaluación se realizó mediante métricas estándar de precisión y validación cruzada para asegurar la robustez y generalización de los modelos. Se compararon los resultados obtenidos con los modelos de regresión simbólica contra las predicciones realizadas por los modelos de Deep Learning para verificar la consistencia y exactitud de las ecuaciones descubiertas.

B.3. Resultados

Los modelos de regresión simbólica lograron identificar las ecuaciones con un alto nivel de precisión. Se realizó un análisis detallado de las ecuaciones descubiertas para interpretar las relaciones y patrones presentes en los datos. Este análisis no solo confirmó la eficacia de la regresión simbólica en la identificación de patrones complejos, sino que también proporcionó una comprensión más profunda de los mecanismos subyacentes en los datos de iluminancia. La Tabla B.2 muestra la precisión de cada modelo, en la cual el coeficiente de correlación R^2 es superior al 90 % cuando se trabaja con la configuración asimétrica en la iluminación y, en los casos donde la configuración fue simétrica, sorprendentemente la precisión es casi del 100 %

Tabla B.2. Precisión de los modelos para cada conjunto de datos

Conjunto de datos	R^2
1 Simétrico	0.9998
3 Asimétrico	0.9147

B.3.1. Ecuaciones obtenidas

Para la interpretación de cada ecuación, es necesario, primero, conocer cuáles son los componentes de las ecuaciones. La Tabla B.3 muestra la descripción de cada uno de ellos.

Tabla B.3. Descripción de las componentes de las ecuaciones generadas por el proceso de Regresión Simbólica.

Etiquetas de las características	
Variable	Descripción
Fx	coordenada x de la fuente lumínica
Fy	coordenada y de la fuente lumínica
Fz	coordenada z de la fuente lumínica
FI	Intensidad de la fuente lumínica
Px	coordenada x de la fuente lumínica del punto iluminado
Py	coordenada y de la fuente lumínica del punto iluminado
Pz	coordenada z de la fuente lumínica del punto iluminado
D	Distancia de la fuente lumínica al punto iluminado
E_H	Valor de iluminancia horizontal
E_V	Valor de iluminancia vertical
F1x	coordenada x de la primera fuente lumínica
F1y	coordenada y de la primera fuente lumínica
F1z	coordenada z de la primera fuente lumínica
F1I	Intensidad de la primera fuente lumínica
F2x	coordenada x de la segunda fuente lumínica
F2y	coordenada y de la segunda fuente lumínica
F2z	coordenada z de la segunda fuente lumínica
F2I	Intensidad de la segunda fuente lumínica
EP	Iluminancia en el punto P

Ecuación que gobierna los datos del escenario con iluminación simétrica y todos los componentes

$$\begin{aligned}
 EP = & -E_H^{0,5} + E_H + \left(\frac{E_V F x}{E_H + \frac{D P z - E_H}{F z^{0,5}}} \right) + \frac{2E_V}{F y} + \frac{E_V}{F x} + P z - \left(-3E_H + E_V \right)^{0,5} \\
 & - \left[-E_V + \left(-E_H + 2E_V \right)^{0,5} \right]^{0,5} + \left(-2E_H + 3E_V - F I^{0,5} \right)^{0,5} \\
 & + \left(\frac{1}{F I} \right) \left\{ \left[\frac{E_V}{E_H^{0,25}} - \frac{E_V F x^2}{-E_H + \frac{E_H}{F x} + \frac{E_H - F I}{F z^{0,25}}} + P z \right] \left[D - 5E_H + 3E_V + F z \left(-E_H + 2E_V \right)^{0,5} \right] \right\}
 \end{aligned}
 \tag{B.1}$$

Ecuación que gobierna los datos del escenario con iluminación asimétrica y todos los componentes

[illegible]

B.4. Conclusiones

Con este experimento se demostró cómo, utilizando regresión simbólica, es posible enfrentar el desafío de encontrar la relación matemática que puede describir un conjunto de datos experimentales, lo que llevó a identificar dos ecuaciones que gobiernan los conjuntos de datos de iluminancia. Sorprendentemente, aunque están lejos de las ecuaciones utilizadas para generar los datos, las ecuaciones para los conjuntos de datos con iluminación simétrica (completa y parsimoniosa) muestran una precisión de casi el 100 %.

Sin duda, contar con un volumen considerable de datos en el conjunto de datos de aprendizaje y ajustar los parámetros del algoritmo de regresión simbólica, fueron condiciones que garantizaron la precisión en las ecuaciones que gobiernan los conjuntos de datos de aprendizaje.

Implicaciones y Futuras Investigaciones

El éxito de esta integración de Deep Learning y regresión simbólica abre nuevas posibilidades para el análisis de datos en diversos campos. Las técnicas desarrolladas en este experimento establecen un precedente para futuras investigaciones que busquen combinar estas metodologías, maximizando así el valor de los recursos computacionales y el conocimiento derivado de los datos. Este enfoque promete avances en la comprensión científica a través de la identificación precisa de relaciones matemáticas subyacentes en conjuntos de datos complejos.