UNIVERSIDAD AUTÓNOMA DE CHIHUAHUA

FACULTAD DE INGENIERÍA

SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

PREDICCIÓN DE PARTIDOS DE HOCKEY SOBRE HIELO DE LA NHL HACIENDO USO DEL APRENDIZAJE AUTOMÁTICO

POR:

MIGUEL ÁNGEL GUTIÉRREZ LÓPEZ

TESIS PRESENTADA COMO REQUISITO PARA OBTENER
EL GRADO DE
MAESTRÍA EN INGENIERÍA EN COMPUTACIÓN

DIRECTOR DE TESIS: DR. LUIS CARLOS GONZÁLEZ GURROLA

CHIHUAHUA, CHIH., MÉXICO **4 DE DICIEMBRE DE 2025**

Predicción de Partidos de Hockey Sobre Hielo de la NHL Haciendo Uso del Aprendizaje Automático. Tesis presentada por Miguel Ángel Gutiérrez López como requisito parcial para obtener el grado de Maestro en Ingeniería en Computación, ha sido aprobado y aceptado por:

M.I. Fabián Vinicio Hernández Martínez
Director de la Facultad de Ingeniería

M.I. Rodrigo de la Garza Aguilar
Secretario de Investigación y Posgrado

M.S.I. Karina Rocío Requena Yáñez
Coordinadora Académica

Dr. Luis Carlos González Gurrola
Director de Tesis

Noviembre 2025
Fecha

COMITÉ
Dr. Luis Carlos González Gurrola
Dr. Raymundo Cornejo García
Dr. Jesús Roberto López Santillán
Dra. Vania Carolina Álvarez Olivas

UNIVERSIDAD AUTÓNOMA DE
CHIHUAHUA

28 de noviembre de 2025.

ING. MIGUEL ANGEL GUTIERREZ LOPEZ
Presente. -

En atención a su solicitud relativa al trabajo de tesis para obtener el grado de Maestría en Ingeniería en Computación, nos es grato transcribirle el tema aprobado por esta Dirección, propuesto y dirigido por el director **Dr. Luis Carlos González Gurrola** para que lo desarrolle como Tesis, con el título **"Predicción de partidos de hockey sobre hielo de la NHL haciendo uso del aprendizaje automático"**.

Índice de Contenido

DEDICATORIA
AGRADECIMIENTOS
RESUMEN
Índice de Contenido
Índice de tablas
Índice de figuras
Índice de ecuaciones
1. Introducción
    1.1 Estado del Arte
    1.2 Hipótesis
    1.3 Problema de Investigación
    1.4 Objetivos
    1.5 Justificación
2. Marco Teórico
    2.1 Extracción de contenido web (Web Scrapping)
    2.2 Aprendizaje automático o machine learning (ML)
    2.3 Aprendizaje no supervisado
    2.4 Aprendizaje de refuerzo
    2.5 Aprendizaje supervisado
    2.6 Algoritmos de aprendizaje supervisado
    2.7 Optimización de hiperparámetros
    2.8 Hockey en la NHL

FACULTAD DE INGENIERÍA
Circuito No. 1, Campus Universitario 2
Chihuahua, Chih., México C.P. 31125
Tel. (614) 442-95-00
www.fing.uach.mx

UNIVERSIDAD AUTÓNOMA DE
CHIHUAHUA

3. Metodología
    3.1 Creación de la base de datos
    3.2 Pre procesamiento de la base datos
    3.3 Selección de modelos de aprendizaje automático para clasificación
    3.4 Optimización de los modelos mediante Optuna
    3.5 Comparación de los resultados de los modelos seleccionados
    3.6 Conclusión y discusiones
4. Resultados
    4.1 Comportamiento de los modelos sin optimización ni PCA
    4.2 Impacto de la reducción de la dimensionalidad (PCA) sin optimización (Optuna)
    4.3 Impacto de la optimización con Optuna en el rendimiento de los modelos
    4.4 Impacto de la optimización con Optuna y haciendo uso de PCA
    4.5 Enfoque más efectivo para predecir el ganador
    4.6 Comparación general de los modelos y enfoques
5. Conclusiones
6. Recomendaciones
7. Discusión
8. Limitaciones
9. Trabajo futuro

Referencias

Curriculum Vitae

A T E N T A M E N T E

"naturam subiecit aliis"

FACULTAD DE INGENIERÍA U.A.CH.

EL DIRECTOR

M.I. FABIÁN VINICIO HERNÁNDEZ MARTÍNEZ

SECRETARIO DE INVESTIGACIÓN Y POSGRADO

M.I. RODRIGO DE LA GARZA AGUILAR

DIRECCIÓN

DEDICATORIA

A mi familia, por su apoyo incondicional desde el momento en que tuve que dejar mi ciudad natal en busca de oportunidades profesionales, y por ser la base que me permitió, años después, iniciar y completar esta maestría.

A mis amigos, quienes comprendieron mis largos periodos de ausencia y, aun así, permanecieron presentes con su amistad sincera.

Y, finalmente, a mi novia, Samantha Meraz, por acompañarme en mis mejores y peores días, ofreciéndome siempre su paciencia, cariño y apoyo inquebrantable.

AGRADECIMIENTOS

Deseo expresar mi profundo agradecimiento a la M.S.I. Karina Rocío Requena Yáñez, coordinadora de la maestría, por haberme abierto las puertas a este programa y por el acompañamiento que brindó a lo largo de toda la formación. Su disposición y compromiso hicieron posible transitar este proceso con claridad y confianza.

De igual manera, agradezco al Dr. Luis Carlos Gurrola González, quien asumió la tutoría de este trabajo. Su orientación académica, sus observaciones puntuales y el tiempo dedicado al seguimiento de mi avance fueron esenciales para dar solidez y dirección a esta tesis.

Extiendo también mi agradecimiento a Javier Robles y Ethel Chavira, quienes depositaron su confianza en mí y me dieron la oportunidad de continuar mis estudios mientras seguía desarrollándome profesionalmente a su lado.

A mis compañeros de maestría, gracias por su apoyo constante, por las discusiones que enriquecieron nuestro aprendizaje y por los momentos compartidos —incluyendo aquellos en los que solo quedaba reírnos de la carga de trabajo y seguir adelante.

Finalmente, un agradecimiento especial para Hugo Santiesteban, con quien este recorrido académico se transformó en una amistad sincera y un vínculo de hermandad que trasciende la vida universitaria.

RESUMEN

El presente estudio se centra en la predicción de los resultados de partidos de la National Hockey League (NHL) mediante el uso de modelos de clasificación de aprendizaje automático. La motivación principal es desarrollar una metodología basada en el estado del arte que permita la predicción en tiempo real de los partidos futuros, utilizando datos históricos como base para la estimación de los posibles ganadores.

Para lograrlo, se han empleado diversos modelos de clasificación, incluyendo Logistic Regression, Random Forest, XGBoost y CatBoost. Dichos modelos fueron evaluados a través de la exactitud (accuracy), con el objetivo de comprobar el rendimiento de estos modelos utilizando diferentes conjuntos de datos con variaciones en las características consideradas.

El estudio se dividió en fases de experimentación separadas para examinar los cambios debido a la optimización y la reducción de dimensionalidad en los modelos. Los modelos fueron probados sin optimización ni reducción de dimensionalidad usando PCA y se encontró que la Regresión Logística obtiene la mejor precisión (57.75% en el conjunto de datos OZDZ). Posteriormente, se implementó la reducción de dimensionalidad mediante PCA y se observó una pérdida general de precisión, particularmente para el modelo XGBoost.

Luego se realizó la optimización de hiperparámetros con Optuna en otra parte, lo que llevó a una mejora en el rendimiento de varios modelos. Específicamente, la Regresión Logística alcanzó una precisión del 59.06% en el conjunto de datos WP2M. Finalmente, la optimización se combinó con la reducción de dimensionalidad utilizando PCA, pero los resultados no superaron a los obtenidos con la optimización de hiperparámetros sin PCA.

El análisis muestra que la optimización de hiperparámetros es uno de los aspectos críticos para mejorar la precisión del modelo, mientras que la reducción de dimensionalidad sin una adecuada sintonización puede llevar a una degradación del rendimiento. Además, se identificó que la precisión del modelo varía a lo largo de la temporada, con un rendimiento más bajo al principio debido a la falta de datos recientes, y un mejor rendimiento a medida

que avanza la temporada regular. Estos hallazgos constituyen una base sólida para la investigación y la predicción de eventos deportivos en tiempo real en el futuro.

ÍNDICE DE CONTENIDO

DEDICATORIA II
AGRADECIMIENTOS III
RESUMEN IV
Índice de Contenido VI
Índice de tablas VIII
Índice de figuras IX
Índice de ecuaciones X
1. Introducción 1
1.1 Estado del Arte 2
1.2 Hipótesis 4
1.3 Problema de Investigación 4
1.4 Objetivos 6
1.4.1 Objetivo General 6
1.4.1 Objetivos Específicos 6
1.5 Justificación 6
2. Marco Teórico 7
2.1 Extracción de contenido web (Web Scrapping) 7
2.2 Aprendizaje automático o machine learning (ML) 8
2.3 Aprendizaje no supervisado 10
2.4 Aprendizaje de refuerzo 10
2.5 Aprendizaje supervisado 11
2.6 Algoritmos de aprendizaje supervisado 11
2.6.1 Árboles de decisión (Decision Trees) 12
2.6.2 Bosque aleatorio (Random Forest) 13
2.6.3 XGBoost (Extreme Gradient Boosting) 14
2.6.4 CatBoost 15
2.6.5 Regresión logística (Logistic Regression) 16
2.7 Optimización de hiperparámetros 18
2.7.1 Optuna 20

2.8 Hockey en la NHL 21
3. Metodología 23
3.1 Creación de la base de datos 23
3.2 Preprocesamiento de la base datos 23
3.3 Selección de modelos de aprendizaje automático para clasificación 25
3.4 Optimización de los modelos mediante Optuna 26
3.5 Comparación de los resultados de los modelos seleccionados 27
3.6 Conclusión y discusiones 27
4. Resultados 28
4.1 Comportamiento de los modelos sin optimización ni PCA 28
4.2 Impacto de la reducción de la dimensionalidad (PCA) sin optimización (Optuna) 29
4.3 Impacto de la optimización con Optuna en el rendimiento de los modelos 31
4.4 Impacto de la optimización con Optuna y haciendo uso de PCA 32
4.5 Enfoque más efectivo para predecir el ganador 34
4.6 Comparación general de los modelos y enfoques 34
5. Conclusiones 36
6. Recomendaciones 37
7. Discusión 38
8. Limitaciones 40
9. Trabajo futuro 42
Referencias 43
Curriculum Vitae 45

ÍNDICE DE TABLAS

Tabla 1. Comparación de precisión de Modelos del Estado del Arte. 4
Tabla 2. Desempeño de los modelos sin Optimización ni PCA. 28
Tabla 3. Desempeño de los modelos con PCA sin Optimización. 29
Tabla 4. Desempeño de los modelos con Optimización sin PCA. 31
Tabla 5. Desempeño de los modelos con Optimización y haciendo uso de PCA. 33

ÍNDICE DE FIGURAS

Figura 1.Pasos en la extracción de datos web (web scrapping)..... 8
Figura 2. Las tres principales tareas de los algoritmos de aprendizaje automático (machine learning)..... 9
Figura 3. Representación de un árbol de decisión para definir si una persona aprueba un examen o no..... 13
Figura 4. Representación de un árbol de un bosque aleatorio donde se llega a una predicción final a partir del conjunto de predicciones obtenido por cada árbol de decisión..... 14
Figura 5.Cómo separa las probabilidades el modelo de regresión logística..... 18
Figura 6. Diferencia entre grid search y random search..... 19
Figura 7. Comparativa de la reducción del error promedio entre Optuna y otras metodologías de búsqueda de hiperparámetros(Sandha et al., 2021). 21
Figura 8. Comparativa del desempeño de los modelos a través de los diferentes datasets sin optimización ni uso de PCA. 29
Figura 9. Comparativa del desempeño de los modelos a través de los diferentes datasets haciendo uso de PCA sin optimización. 30
Figura 10. Comparativa del desempeño de los modelos a través de los diferentes datasets haciendo uso de optimización..... 32
Figura 11. Comparativa del desempeño de los modelos a través de los diferentes datasets haciendo uso de optimización y de la metodología PCA 33

ÍNDICE DE ECUACIONES

Ecuación 1. Función logistica. 17
Ecuación 2. Cálculo de simple promedio móvil (SMA). 24
Ecuación 3.Cálculo de métrica de rendimiento defensivo y ofensivo. 25

1. INTRODUCCIÓN

El aprendizaje automático, también conocido como machine learning, se encuentra dentro del ámbito de las ciencias de la computación. Su objetivo principal es desarrollar algoritmos que se basen en ejemplos de fenómenos para ser funcionales [1]. De esta manera, mediante el análisis estadístico, el ordenador es capaz de reconocer patrones y aprender de manera autónoma.

El aprendizaje automático se especializa en dos tareas en particular: predicción y clasificación. La clasificación asigna la clase a la que pertenece un conjunto de datos. Dicho conjunto de datos debe ser particionados de tal manera que se tenga un conjunto de entrenamiento y uno de pruebas, esto con la finalidad de construir un modelo de aprendizaje automático a través de los datos de entrenamiento para, posteriormente, probar la eficiencia del modelo creado a través de clasificar el conjunto de prueba. De esta manera se logra obtener qué tan bien se adapta el modelo a la hora de clasificar los datos. A este tipo de aprendizaje automático se le denomina aprendizaje supervisado dado que el conjunto de datos incluye la clasificación (o etiqueta) a la que pertenece dicho conjunto, de esta manera el algoritmo aprende a clasificar los nuevos datos que le sean dados (Raschka & Mirjalili, n.d.). En el aprendizaje automático existen diversos algoritmos que están más adaptados o se desempeñan en el ámbito de la clasificación, por lo que la selección de un algoritmo de aprendizaje automático también es una tarea para tener en cuenta.

La predicción deportiva se ha convertido en un ámbito cada vez más importante en el que se han aplicado técnicas de aprendizaje automático. La precisión de los modelos de predicción deportiva es esencial, ya que no se trata solo de una cuestión de interés personal, sino que también implica grandes cantidades monetarias en forma de apuestas. Debido a esto, el interés en la predicción deportiva ha ganado popularidad aunada a la creciente disponibilidad de estadísticas, datos en línea y casas de apuesta en línea (R. Bunker & Susnjak, 2022).

Dada la naturaleza de la predicción deportiva, esta se toma como un problema de clasificación con una clase a predecir (ganó, perdió y empató) (R. P. Bunker & Thabtah, 2019). Los deportes tienen una gran ventaja para la predicción a través de aprendizaje

automático debido a que se registran diversas estadísticas de cada deporte que en conjunto con datos históricos pueden crear modelos de clasificación para la predicción deportiva.

Una tarea importante en la predicción de resultados en el deporte es seleccionar el mejor conjunto de características para el modelo predictivo (R. Bunker & Susnjak, 2022). Esto se logra a través de ingeniería de datos con la finalidad de obtener un conjunto de datos significativo para entrenar un modelo de aprendizaje automático, dado que si se eligen datos que no tengan una relación importante pueden llevar al modelo de aprendizaje automático a no generalizar de forma adecuada dando como resultado una predicción del ganador de un partido deportivo pobre. Por esta razón es importante una selección y procesamiento adecuado de los datos a utilizar para entrenar al modelo de aprendizaje automático.

1.1 Estado del Arte

En la actualidad la predicción deportiva es llevada a cabo por diversos servicios en línea como puede ser el de AWS (Amazon) hasta diversos portales web de suscripción. Pero también es posible entrenar modelos de aprendizaje automático para dicha tarea.

Un deporte poco explorado en la predicción deportiva por medio de modelos de aprendizaje automático es el hockey, en específico la NHL (National Hockey League). En consecuencia, en comparación con el baloncesto (NBA) y el fútbol americano (NFL), los resultados de los equipos y jugadores de hockey son más difíciles de predecir, debido a las frecuentes sustituciones de jugadores, la alta velocidad de los partidos, las colisiones y las peleas (Gu et al., 2019).

En el año 2013, Weissbock, et al., estudiaron el efecto de las características en la precisión predictiva utilizando distintos algoritmos de aprendizaje automático en el contexto de la NHL. Se consideraron tanto las estadísticas tradicionales como las métricas de rendimiento utilizadas por los escritores de blogs y los estadísticos empleados por los equipos. Para entrenar los clasificadores, se utilizaron las implementaciones de WEKA de ANN, Naive Bayes, SVM y C4.5 en conjuntos de datos que describían los resultados de los partidos de la NHL en la temporada 2012/2013. Como resultado, se encontró que las estadísticas tradicionales fueron más efectivas en la predicción de los resultados de partidos

individuales mediante una validación cruzada de 10 veces, mientras que la ANN obtuvo la mejor precisión (59.38%) (Weissbock et al., 2013).

En el año 2014, Weissbock e Inkpen hicieron una nueva investigación tomando como base la realizada en el año 2013. Analizaron la relación entre los informes previos al partido y las características estadísticas para predecir los resultados de los partidos de hockey sobre hielo de la NHL. Utilizaron datos de 708 partidos de las temporadas 2012/2013, y aplicaron tanto el procesamiento del lenguaje natural como el análisis de sentimientos a los informes previos al partido. Se compararon tres modelos: solo características estadísticas, solo texto del informe previo al partido y características de análisis de sentimientos. Se encontró que los modelos que utilizaban solo características estadísticas superaban a los modelos que utilizaban informes previos al partido. Un metaclasificador con voto mayoritario proporcionó la mejor precisión (60.25\%). Sin embargo, los autores señalaron que predecir partidos con datos de una o dos temporadas anteriores era complicado debido a los cambios de jugadores, entrenadores, etc.(Weissbock & Inkpen, 2014).

La investigación más reciente encontrada es del año 2019, donde Gu y Foster. usaron métodos de conjunto (ensamble) para predecir los resultados de partidos de hockey de la NHL durante varias temporadas. Extrajeron 1,230 partidos de varias páginas web de la temporada 2007/2008 a 2016/2017 y fusionaron datos de múltiples fuentes, incluyendo resultados históricos, información de la oposición, indicadores de rendimiento a nivel de jugador y clasificaciones de jugadores utilizando PCA (Principal Component Analysis). El estudio incluyó 26 variables de rendimiento de equipos y comparó diferentes algoritmos, con los métodos basados en conjuntos (ensamble) obteniendo la mayor precisión en el conjunto de pruebas (91.8%). Los autores concluyeron que las características adicionales tales como las métricas de clasificación de los jugadores y los juicios estratégicos o tácticos de los entrenadores podrían mejorar las predicciones, y que el problema de ML podría reformularse para predecir diferentes resultados, entrenando un modelo con datos de la temporada regular (Gu et al., 2019).

La comparación completa entre los modelos del estado del arte la podemos observar en la Tabla 1.

Autor	Mejor Modelo	Datos	Características	Temporada	Precisión
Weissbock (2013)	ANN	517	12	2012-2013	59.38%
Weissbock (2014)	Cascading Ensemble	720	13	2012-2013	60.25%
Wei Gu (2016)	ANP	1230	17	2014-2015	77.50%
Wei Gu (2019)	Ensemble ML: SVM, DT, KNN	1230	19	2014-2015	91.80%
Čermák (2021)	SVM	9120	100	2009-2020	62.30%
Sivakumar (2023)	Logistic Regression	5717	No Disponible	2015-2021	77.82%

Tabla 1. Comparación de precisión de Modelos del Estado del Arte.

En la actualidad, el campo de las predicciones deportivas continúa siendo explorado mediante la combinación de diversas técnicas de aprendizaje automático. Sin embargo, uno de los deportes menos estudiados en este sentido, en comparación con la NFL y la NBA, es el hockey. Por lo tanto, lograr predicciones competitivas en los partidos de hockey representa un desafío importante a superar.

1.2 Hipótesis

En el contexto de la predicción de ganadores en la NHL utilizando datos tabulares, los modelos tradicionales de aprendizaje automático, como la regresión logística, pueden igualar o incluso superar el rendimiento de modelos más complejos como XGBoost y CatBoost, desafiando así la tendencia predominante en el estado del arte.

1.3 Problema de Investigación

En la predicción deportiva, parece haber una notable falta de investigación sobre el hockey sobre hielo, particularmente en la Liga Nacional de Hockey (NHL), en comparación con disciplinas ampliamente investigadas como la Asociación Nacional de Baloncesto (NBA), la Liga Nacional de Fútbol Americano (NFL) y el fútbol. Aunque existe un creciente número de estudios sobre aplicaciones de modelos de aprendizaje automático en deportes, la predicción de resultados en partidos de hockey ha recibido relativamente poca atención académica. Este déficit no solo impide el establecimiento de metodologías predictivas que puedan mejorar el rendimiento del equipo, sino que también restringe la planificación estratégica en los deportes profesionales.

Al mismo tiempo, también hay una tendencia creciente en el aprendizaje automático donde los modelos basados en árboles de decisión se han establecido como el estado del arte para resolver problemas de datos tabulares, particularmente enfoques de potenciación como XGBoost y CatBoost. Esta afirmación se refleja en artículos críticos, por ejemplo, los de Grinsztajn et al. (2022), que afirman la superioridad constante de tales modelos en comparación con métodos más sofisticados, como las redes neuronales profundas, en dichos datos. Además, investigaciones como las realizadas por Prokhorenkova et al. (2018) se centran específicamente en los beneficios de CatBoost para manejar variables categóricas sin sesgo, mientras que aplicaciones prácticas en entornos del mundo real, por ejemplo, la detección de robo de electricidad (Energy Reports, 2021), corroboran su efectividad.

Pero esta preferencia generalizada por modelos complejos plantea la pregunta sobre el verdadero valor de los modelos clásicos, como la regresión logística, en tareas específicas. La regresión logística, un método simple pero capaz, ofrece interpretabilidad, bajo costo computacional y se basa en una sólida base teórica. Las preguntas anteriores son directamente relevantes para predecir el resultado de la NHL, ya que la estructura de los datos y el problema en la NHL pueden ser sustancialmente diferentes a los utilizados en otros dominios, y por lo tanto es apropiado preguntar si métodos más complejos funcionarán mejor que tales modelos clásicos. Este enfoque tiene relevancia ya que estudios recientes, como los de Yan y Xu (2024), introducen soluciones híbridas basadas en la naturaleza limitada de varias formas de datos heterogéneos.

Por tanto, este estudio se propone abordar dos vacíos relevantes: por un lado, contribuir al desarrollo de sistemas predictivos aplicados específicamente al hockey profesional; y por otro, evaluar empíricamente el desempeño comparado de modelos clásicos y modernos en datos tabulares, desafiando la suposición predominante de que los modelos de boosting son siempre superiores en este tipo de tareas.

1.4 Objetivos

1.4.1 Objetivo General

Diseñar y evaluar un modelo de predicción deportiva para partidos de hockey de la NHL utilizando técnicas de machine learning, comparando el desempeño de modelos simples y complejos.

1.4.1 Objetivos Específicos

- Recopilar información acerca de las estadísticas más relevantes en la NHL.
- Hacer uso de minería de datos para extraer las estadísticas necesarias para la predicción.
- Analizar las estadísticas extraídas para determinar la contribución a la predicción.
- Seleccionar y entrenar modelos de machine learning adecuados, considerando tanto modelos simples como complejos.
- Evaluar y comparar el desempeño de los modelos seleccionados para identificar el más competitivo.

1.5 Justificación

La predicción deportiva se ha convertido en un tema de interés popular gracias al continuo avance en los algoritmos de machine learning y al amplio acceso a datos y estadísticas en línea. Tal interés ha impulsado la búsqueda de algoritmos eficientes, para impactar la estrategia en el campo del deporte. Sin embargo, el hockey está relativamente poco explorado, posiblemente debido a su naturaleza rápida y dinámica. Esta investigación tiene como objetivo llenar este vacío, utilizando técnicas de minería de datos y algoritmos de machine learning menos utilizados en este contexto. Se espera que los hallazgos de este estudio contribuyan al desarrollo de nuevas técnicas y herramientas para la predicción deportiva, así como a la mejora del rendimiento de los jugadores y equipos de hockey.

2. MARCO TEÓRICO

2.1 Extracción de contenido web (Web Scrapping)

La extracción de datos web, también conocida como web scraping, es una técnica que utiliza software para extraer datos de forma automática de sitios web. Aplicada para construir bases de datos, la extracción de datos web es útil cuando necesitas los metadatos que actualmente no se pueden obtener como archivos CSV o Excel, en tablas HTML (Lenguaje de Marcado de Hipertexto).

HTML es el lenguaje base de las páginas web. Define la estructura y el contenido de una página web, incluyendo elementos como texto, imágenes, encabezados, párrafos y, por supuesto, tablas.

El web scraping funciona analizando el código HTML de una página web para identificar los elementos específicos que contienen la información que deseas extraer. Luego, utiliza herramientas y técnicas de programación para aislar y extraer esos datos.

En la práctica, la extracción de contenido web, conlleva una gran variedad de técnicas de programación y tecnologías, como análisis de datos, procesamiento de lenguaje natural e incluso seguridad de la información (Mitchell, n.d.). Por lo que para llevar a cabo un proceso de extracción de información web, los siguientes pasos son los que se llevan a cabo generalmente:

1) Identificación del sitio web y páginas de las que se desea extraer la información.
    - Identificar el sitio que contiene la información necesaria.
    - Determinar las aginas especificas donde se encuentran las tablas a consumir.

2) Análisis del HTML.
    - Evaluar la metodología usada en el código fuente de la página (HMTL).
    - Encontrar las tablas que contienen la información de interés.

3) creación del programa para extraer la información.
    - Seleccionar la librería que mejor se adapte a nuestras necesidades.

- Escribir el código que extraerá la información de los elementos HTML previamente seleccionados.

4) Procesamiento y limpieza de los datos.
   - Evaluar si los datos extraídos contienen información que no es de interés.
   - Limpiar y transformar los datos a los formatos requeridos.

5) Guardar los datos procesados en una base de datos.
   - Seleccionar el tipo de base de datos optima.
   - Guardar los datos de acuerdo con la metodología usada para la base de datos seleccionada.

A continuación, la Figura 1 muestra de forma esquemática los pasos previamente detallados para llevar a cabo el proceso de extracción de información web.

Figura 1.Pasos en la extracción de datos web (web scrapping).

2.2 Aprendizaje automático o machine learning (ML)

El aprendizaje automático es la ciencia de programar computadoras, permitiendo a las mismas aprender de los datos (Geron, n.d.). El término aprendizaje automático fue acuñado por Arthur Samuel (IBM, n.d.). Esta disciplina, una rama de la inteligencia artificial (IA), se enfoca en utilizar datos y algoritmos para emular el proceso de aprendizaje humano. A través del machine learning, las computadoras pueden identificar patrones en los datos de entrada y generar predicciones o clasificaciones para nuevos conjuntos de datos, utilizando algoritmos estadísticos.

Dentro del aprendizaje automático, existen tres tareas principales en las que se desempeñan los algoritmos. La regresión implica predecir datos de salida en función de los

datos de entrada, como pronosticar el rendimiento futuro de una empresa basándose en sus registros históricos (Harrington, n.d.).

La clasificación es el proceso de etiquetar estos datos para decidir a qué clase pertenecen; por ejemplo, podemos categorizar la raza de un perro a partir de imágenes clasificadas por razas. Por último, el agrupamiento implica que el algoritmo aprende de los datos para identificar patrones y agruparlos en conjuntos, revelando relaciones entre ellos que pueden no ser evidentes inicialmente. Una forma gráfica de visualizar estas tareas se muestra en la Figura 2. Los algoritmos de aprendizaje automático realizan 3 tareas principales. En la regresión, el algoritmo de aprendizaje automático busca predecir dentro de una línea de tendencia, permitiendo predicciones más precisas. En la clasificación, se buscan diferentes tipos de colores que representan clases; donde intenta clasificar los datos en una clase u otra. Por último, en el agrupamiento, los conjuntos de datos se caracterizan por la presencia de centros (x), donde emergen los patrones y la información se separa en grupos.

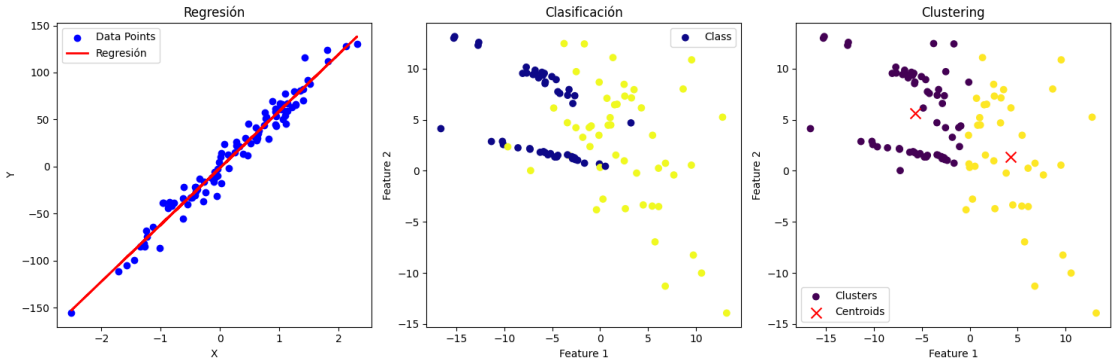

Figura 2. Las tres principales tareas de los algoritmos de aprendizaje automático (machine learning).

El machine learning se emplea en una amplia gama de aplicaciones actuales. Por ejemplo, se utiliza para detectar tumores en imágenes cerebrales, clasificar automáticamente nuevos artículos mediante procesamiento de lenguaje natural y predecir el rendimiento futuro de una empresa en función de su historial de producción. Además, los algoritmos de ML son capaces de reconocer patrones, permitiendo a las computadoras imitar acciones específicas, como la conducción autónoma de vehículos eléctricos. Estos algoritmos aprenden a realizar estas tareas principales a través de tres tipos de aprendizaje: supervisado, no supervisado y de refuerzo (Harrington, n.d.).

2.3 Aprendizaje no supervisado

En este tipo de aprendizaje los datos no están etiquetados, por lo que la computadora trata de aprender las relaciones existentes entre los datos, creando de este modo agrupaciones. Un ejemplo de aprendizaje no supervisado podría ser agrupar los datos de visitantes de una página de internet, de esta manera el algoritmo podría encontrar que ciertos visitantes son hombres con intereses en ciencia de la computación, otro grupo de visitantes podría estar formado por personas con intereses en redes sociales que lee la página de internet a ciertas horas del día, y así podríamos seguir agrupando visitantes por medio de las relaciones que encuentre el algoritmo. Los algoritmos de visualización son buenos ejemplos de aprendizaje no supervisado, donde se alimenta el algoritmo de ML con un conjunto de datos no etiquetados y el algoritmo entrega como salida una representación 2D o 3D de los datos. Una de las principales tareas en las que se utiliza este tipo de aprendizaje es en la reducción de dimensionalidad, donde se busca simplificar un conjunto de datos muy grande sin perder mucha información. También se puede usar para detectar anomalías en los datos, como por ejemplo detectar transacciones en tarjetas de crédito inusuales que podría ser fraude.

2.4 Aprendizaje de refuerzo

El aprendizaje por refuerzo implica que el algoritmo de aprendizaje automático interactúe con su entorno, tome decisiones y reciba recompensas o penalizaciones basadas en esas decisiones. Esta interacción continua permite que el algoritmo aprenda de manera autónoma, determinando la estrategia más efectiva para maximizar las recompensas a lo largo del tiempo. Por ejemplo, en el caso de la conducción autónoma de vehículos eléctricos, el algoritmo aprende la mejor manera de conducir basándose en las recompensas obtenidas de sus acciones. Esto significa que el algoritmo desarrolla reglas para la toma de decisiones, de modo que cada vez que sigue la regla correcta y recibe una recompensa, aumenta la probabilidad de tomar la misma decisión en situaciones similares en el futuro. En otras palabras, el algoritmo ajusta su comportamiento con el tiempo basándose en experiencias pasadas, lo que le permite mejorar su rendimiento y adaptarse a diferentes circunstancias.

2.5 Aprendizaje supervisado

En el aprendizaje supervisado, el algoritmo de aprendizaje automático ha sido entrenado en un conjunto de datos que contiene la salida (en otras palabras, soluciones conocidas) para cada entrada. Estos datos a menudo se denominan datos etiquetados, ya que cada muestra en nuestros datos tiene una etiqueta que representa el resultado. Un enfoque de aprendizaje supervisado típicamente se centra en una de sus tareas más comunes: la clasificación. Aquí es donde el algoritmo aprende la relación entre las características de los datos y las clases a las que pertenecen; lo hace utilizando los patrones y relaciones entre los datos. Para un conjunto de datos de este tipo de imágenes (imágenes de animales como 'perro', 'gato' o 'pájaro'), el algoritmo de clasificación puede aprender a identificar una clase basada en características como la forma, el color, la textura, etc.

Aparte de la clasificación, la regresión es otro problema importante en el aprendizaje supervisado. En lugar de predecir una clase discreta, el algoritmo predice un valor numérico continuo. Por ejemplo, con características como la ubicación geográfica, el tamaño de la casa y el número de habitaciones, el algoritmo puede proyectar el precio de una casa a partir de datos históricos de ventas.

El aprendizaje supervisado permite al algoritmo aprender de datos etiquetados para realizar tareas como la clasificación y la regresión, lo que facilita la toma de decisiones y la predicción en una variedad de aplicaciones.

2.6 Algoritmos de aprendizaje supervisado

Debido a los tipos de datos que se van a emplear para lograr la predicción deportiva, los algoritmos de aprendizaje automático (machine learning) empleados a considerar serán aquellos que sean de aprendizaje supervisado. La tarea para desempeñar es clasificación, debido a que los datos estarán etiquetados al contener el ganador o perdedor de los equipos de la NHL. A continuación, se describe el funcionamiento de algunos algoritmos de aprendizaje automático para la tarea de clasificación:

2.6.1 Árboles de decisión (Decision Trees)

Los árboles de decisión son modelos muy utilizados para tareas de clasificación y regresión. Esencialmente, aprenden una jerarquía de preguntas if/else, que conducen a una decisión (Müller & Guido, 2016).

La idea básica de los árboles de decisión es crear una estructura en forma de árbol en la que cada nodo interno representa una prueba sobre una característica específica, y cada nodo representa una etiqueta de clase o un valor numérico. El objetivo es crear un árbol de decisión capaz de predecir con exactitud la clase o el valor de nuevos puntos de datos basándose en sus características de entrada.

Al crear el árbol de decisión, el algoritmo comienza con una selección de qué característica clasifica mejor los datos en diferentes clases o grupos. Al obtener números como la ganancia de información o el índice de Gini para cada característica, el algoritmo calcula qué tan bien la característica separa los datos. Una vez que se selecciona la característica más adecuada, el algoritmo también producirá un nodo en el árbol y dividirá los datos en grupos según sus valores. Esto se repite para cada subconjunto, creando recursivamente nuevos nodos y divisiones hasta que se alcanza un punto de parada, por ejemplo, con una profundidad máxima o un número mínimo de muestras en un nodo hoja.

El algoritmo recorre el árbol de decisión para predecir nuevos datos y va desde el nodo raíz hasta el camino correcto basado en los valores de las características de entrada. En el nodo hoja, el sistema predice la clase o el valor de un nuevo punto de datos. Los árboles de decisión tienen la ventaja de ser fácilmente interpretables junto con una visualización útil para entender el proceso de decisión de un modelo. La Figura 3 es una ilustración de tal estructura en la que también se formula un árbol de decisión basado en las horas de estudio para decidir si un estudiante aprueba o reprueba.

Los árboles de decisión también pueden manejar datos categóricos y numéricos, y pueden utilizarse para tareas de clasificación tanto binarias como multiclase.

El sobreajuste puede crear problemas con los árboles de decisión, ya que se vuelven demasiado complicados; cuando el modelo se ajusta al ruido en lugar de a los patrones subyacentes en los datos. Como solución a este problema, se puede podar el árbol y establecer un número mínimo de muestras en un nodo hoja para ayudar a reducir el sobreajuste. Los árboles de decisión son una herramienta poderosa de aprendizaje automático utilizada para

muchas tareas diferentes. Debido a que los datos también pueden dividirse recursivamente en función de los valores de las características de entrada, los árboles de decisión hacen predicciones precisas y ofrecen información sobre el proceso de toma de decisiones del modelo (Harrington, s.f.).

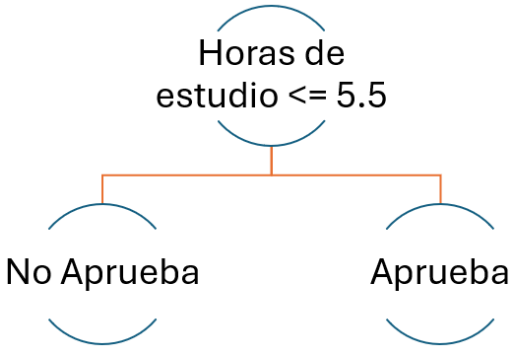

Figura 3. Representación de un árbol de decisión para definir si una persona aprueba un examen o no.

2.6.2 Bosque aleatorio (Random Forest)

Random forest es un algoritmo popular de aprendizaje por conjuntos que se utiliza para tareas de clasificación y regresión. Funciona mediante la creación de múltiples árboles de decisión en diferentes subconjuntos de los datos de entrenamiento y luego combinando sus predicciones para hacer una predicción final.

La idea que subyace en el algoritmo de random forest es reducir el sobreajuste y aumentar la generalización del modelo mediante la creación de múltiples árboles de decisión que se entrenan en diferentes subconjuntos de los datos de entrenamiento. Cada árbol de decisión se construye seleccionando aleatoriamente un subconjunto de las características y un subconjunto de los datos de entrenamiento y, a continuación, utilizando un criterio de división para dividir los datos en grupos cada vez más pequeños en función de los valores de las características seleccionadas. A continuación, se asigna a cada grupo una clase o valor de predicción basado en la clase mayoritaria o el valor medio de los ejemplos de entrenamiento de ese grupo.

Para hacer una predicción de un nuevo punto de datos, el algoritmo de random forest pasa el punto de datos por cada uno de los árboles de decisión del random forest y recoge las

predicciones de cada árbol. La predicción final se realiza combinando las predicciones individuales mediante un voto mayoritario (para clasificación).

En la Figura 4 se ilustra este proceso, mostrando cómo un conjunto de árboles de decisión independientes produce predicciones parciales que posteriormente se combinan para generar la predicción final del modelo.

Una de las ventajas del bosque aleatorio es que puede manejar espacios de características de alta dimensión y datos ruidosos, y es relativamente insensible al sobreajuste. También puede proporcionar medidas de la importancia de las características, que pueden ser útiles para la selección de características y el análisis de datos.

Sin embargo, el algoritmo random forest puede ser costoso desde el punto de vista informático y consumir mucha memoria, especialmente para grandes conjuntos de datos con muchas características. También requiere el ajuste de hiperparámetros como el número de árboles, la profundidad máxima de los árboles y el número de características a seleccionar aleatoriamente en cada división (Raschka & Mirjalili, n.d.).

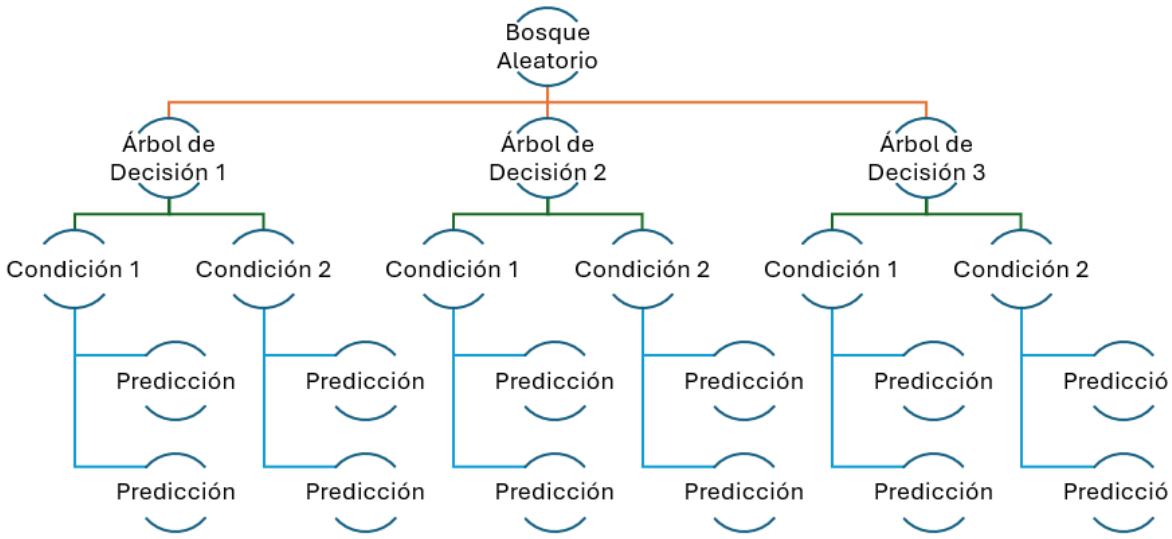

Figura 4. Representación de un árbol de un bosque aleatorio donde se llega a una predicción final a partir del conjunto de predicciones obtenido por cada árbol de decisión.

2.6.3 XGBoost (Extreme Gradient Boosting)

XGBoost, que significa Extreme Gradient Boosting, es una técnica de aprendizaje automático que ha ganado popularidad por su efectividad en la clasificación y regresión en

conjuntos de datos estructurados. Es una implementación eficiente y escalable del algoritmo de Gradient Boosting.

XGBoost se utiliza en clasificación para crear una serie de árboles de decisión, donde cada árbol intenta corregir los errores de los árboles anteriores. El concepto detrás del Gradient Boosting es fusionar varios modelos débiles en un modelo fuerte. La función de pérdida para el modelo XGBoost es aquella para la cual se aborda el problema de clasificación, por ejemplo, logística para clasificación binaria o softmax para clasificación multiclase. En el proceso de entrenamiento, XGBoost ajusta iterativamente los árboles de decisión para minimizar la función de pérdida a través del descenso de gradiente durante el proceso. Además, XGBoost implementa regularización para evitar el sobreajuste. Esto se logra mediante la inclusión de términos de penalización en la función de pérdida que penalizan la complejidad del modelo, como el número de nodos en los árboles o la magnitud de los coeficientes.

XGBoost tiene una propiedad novedosa para elegir automáticamente las variables importantes. Durante el entrenamiento, se evalúa la importancia de cada variable predictora y ayuda en la construcción de los árboles de decisión. Además, la eficiencia de XGBoost en términos de tiempo de entrenamiento y predicción es extremadamente alta porque tiene una implementación optimizada que aprovecha al máximo el paralelismo del sistema y la memoria.

XGBoost es un modelo de aprendizaje automático potente y versátil que se utiliza ampliamente en problemas de clasificación debido a su capacidad para construir modelos precisos y eficientes en conjuntos de datos estructurados. Su capacidad para manejar la regularización, la selección de variables importantes y su eficiencia computacional lo convierten en una herramienta valiosa en el campo del aprendizaje automático.(Chen & Guestrin, 2016)

2.6.4 CatBoost

CatBoost es una biblioteca utilizada para el aprendizaje automático en trabajos de clasificación y regresión, desarrollada por Yandex. Está basada en el Boosting de Gradiente y tiene como objetivo mejorar la precisión de las predicciones y la eficiencia computacional. Una de las características clave de CatBoost es su capacidad para manejar datos categóricos sin ningún preprocesamiento. Esto le permite abordar conjuntos de datos con variables

categóricas directamente y no requiere preprocesarlos a valores numéricos utilizando técnicas como one-hot encoding.

CatBoost emplea un enfoque de aprendizaje por gradiente para ajustar una secuencia de modelos débiles, generalmente árboles de decisión, de modo que cada árbol se centra en corregir los errores de los árboles anteriores. Durante el entrenamiento, se minimiza una función de pérdida específica para el tipo de problema que se está abordando; ejemplos incluyen la pérdida logística para clasificación binaria y la pérdida de desviación para regresión. Además, CatBoost se basa en múltiples métodos de regularización para limitar el sobreajuste, como la poda de árboles, la profundidad máxima del árbol y la reducción de la tasa de aprendizaje. Esto mejora la generalización del modelo y previene un ajuste excesivo al conjunto de datos de entrenamiento dado.

Otra característica importante de CatBoost es su capacidad para manejar automáticamente la selección de características importantes y la interacción entre características. Durante el entrenamiento, el algoritmo evalúa la importancia de cada característica y utiliza esta información para guiar la construcción de los árboles de decisión.

CatBoost es famoso por su eficiencia computacional y su capacidad para manejar grandes conjuntos de datos. Utiliza técnicas de paralelización, así como optimización de memoria para acelerar el entrenamiento y la predicción, haciéndolo apropiado para aplicaciones en tiempo real y entornos altamente centrados en datos. CatBoost es una biblioteca de aprendizaje automático poderosa y eficiente, que sirve para problemas de clasificación y regresión. Su capacidad para manejar eficientemente datos categóricos, su enfoque en la regularización y selección de características, y su eficiencia computacional lo convierten en una herramienta valiosa en el campo del aprendizaje automático. (Prokhorenkova et al., 2017)

2.6.5 Regresión logística (Logistic Regression)

La regresión logística es un modelo estadístico utilizado para predecir la probabilidad de que una variable categórica tenga uno de dos posibles resultados. Es ampliamente utilizado en problemas de clasificación binaria, donde la variable de respuesta es dicotómica, es decir, tiene solo dos categorías posibles.

El modelo de regresión logística se basa en la función logística, que transforma la salida de un modelo lineal a una escala de 0 a 1, representando así la probabilidad de pertenencia a una de las dos clases. Matemáticamente, la función logística se expresa como:

$$P(Y = 1 \mid X) = \frac{1}{1 + e - (\beta 0 + \beta 1X1 + \beta 2X2 + \ldots + \beta nXn)}$$

Ecuación 1. Función logistica.

Donde:

- $P(Y = 1 \mid X)$ es la probabilidad condicional de que la variable de respuesta sea igual a 1 dado el vector de variables predictoras X.
- e es la base del algoritmo natural.
- $\beta 0, \beta 1, \ldots, \beta$n son los coeficientes del modelo, que representan el efecto de cada variable predictora en la probabilidad de pertenencia a la clase 1.
- $X1, X1, .., X1$ son las variables predictoras.

El modelo de regresión logística se entrena mediante el método de máxima verosimilitud, que busca encontrar los valores óptimos de los coeficientes que maximizan la probabilidad de observar los datos de entrenamiento. Una vez que el modelo está entrenado, se puede utilizar para predecir la probabilidad de pertenencia a una clase dada una nueva observación.

El umbral de decisión se utiliza para convertir las probabilidades predichas en clases discretas. Por ejemplo, si la probabilidad predicha es mayor que 0.5, se clasificará como perteneciente a la clase 1, de lo contrario, se clasificará como perteneciente a la clase 0 (Albon, 2018).

En la Figura 5 se muestra una representación visual de este proceso, ilustrando cómo la función sigmoide separa las probabilidades y define la frontera que determina la pertenencia a cada clase.

La regresión logística es un modelo flexible y poderoso que se utiliza ampliamente en la clasificación binaria debido a su capacidad para modelar la relación entre variables predictoras y la probabilidad de pertenencia a una clase específica.

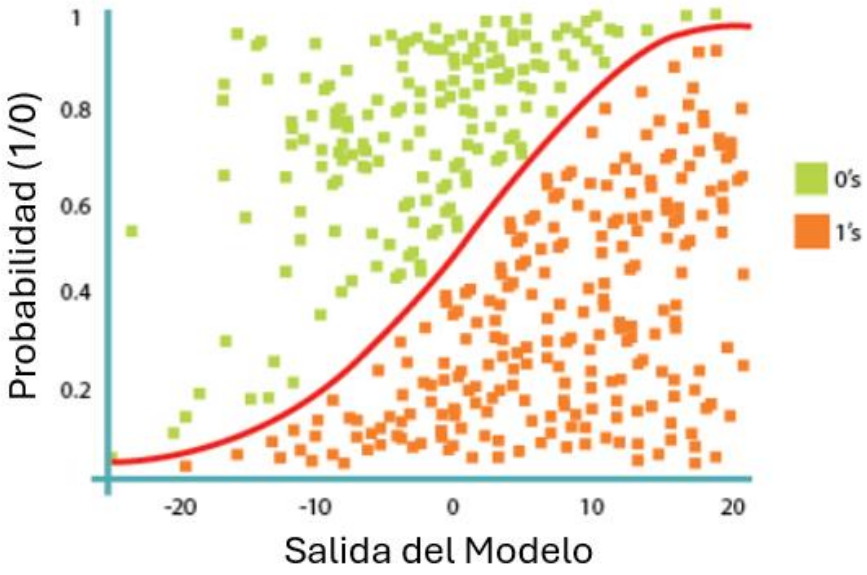

Figura 5.Cómo separa las probabilidades el modelo de regresión logística.

2.7 Optimización de hiperparámetros

La optimización de hiperparámetros en el aprendizaje automático se refiere a encontrar los valores correctos para los hiperparámetros de un algoritmo de aprendizaje automático. Los hiperparámetros son configuraciones que el modelo no aprende directamente de los datos durante el entrenamiento, a diferencia de los parámetros del modelo. Más bien, los hiperparámetros deben ajustarse antes del entrenamiento para modificar la capacidad del modelo de desempeñarse bien, así como su capacidad general. Estos incluyen la tasa de aprendizaje, el número de árboles en un bosque aleatorio, el número de capas ocultas en una red neuronal, y así sucesivamente. Por lo tanto, el conjunto de hiperparámetros para cada modelo será diferente.

La optimización de hiperparámetros es el proceso de encontrar la combinación óptima de valores de hiperparámetros para lograr la máxima precisión, el mínimo error o para optimizar alguna otra métrica de rendimiento aplicable al problema. Generalmente se realiza con algún método como la búsqueda exhaustiva a través de una cuadrícula de valores posibles, búsqueda aleatoria, optimización bayesiana, o técnicas más avanzadas como la optimización evolutiva o algoritmos genéticos. La optimización de hiperparámetros tiene como objetivo mejorar el rendimiento del modelo y lograr resultados más precisos y

generalizables. Si los hiperparámetros no se optimizan adecuadamente, es poco probable que un modelo de aprendizaje automático alcance su máximo potencial predictivo.

Para lograr esto hay distintas metodologías disponibles que impactan en el costo computacional en la búsqueda de los hiperparámetros. Entre estas metodologías las más comunes son busqueda aleatoria y búsqueda por cuadricula (grid search). La diferencia principal entre estas dos metodologias es el cómo se usan las combinaciones de hiperparámetros a evaluar. Con Grid Search se tomarán en cuenta todas las combinaciones de hiperparámetros definidas, mientras que para random search, las combinaciones de hiperparámetros serán aleatorias de acuerdo con el espacio de hiperparámetros que se haya definido para el modelo.

En la Figura 6 se comparan visualmente ambas metodologías mediante nubes de puntos que ilustran cómo Grid Search explora sistemáticamente el espacio de hiperparámetros, mientras que Random Search lo hace de manera aleatoria.

La búsqueda de hiperparámetros puede ser ejecutado a través del uso de distintas librerías disponibles para modelos de aprendizaje automático, entre ellas Sklearn y Optuna.

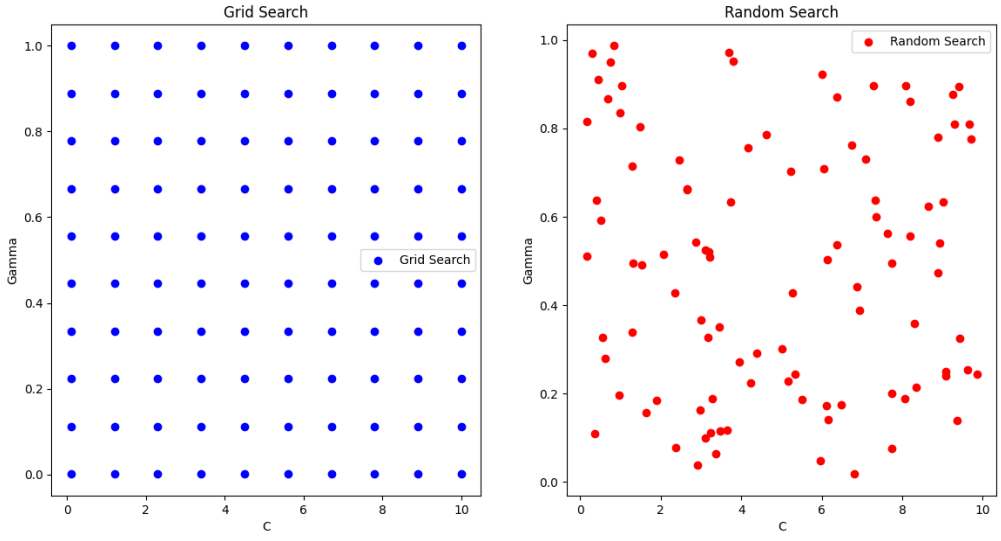

Figura 6. Diferencia entre grid search y random search.

2.7.1 Optuna

Optuna es un marco de trabajo de código abierto para la optimización de hiperparámetros en el ámbito del aprendizaje automático. Se posiciona como una alternativa a enfoques tradicionales como la búsqueda por cuadrícula (Grid Search) y la búsqueda aleatoria (Random Search), ofreciendo ventajas en cuanto a eficiencia, flexibilidad y facilidad de uso.

Optuna trabaja bajo una arquitectura "define-by-run" (Akiba et al., 2019). A diferencia de Grid Search y Random Search, donde el espacio de búsqueda de parámetros se define de forma estática y rígida, Optuna permite a los usuarios definirlo dinámicamente durante cada ejecución del entrenamiento del modelo. Esta flexibilidad permite adaptar la búsqueda a las necesidades específicas de cada modelo y a la información disponible en cada iteración. Esto se logra a través de una función objetivo, en la cual se define el espacio de parámetros mientras que el modelo se entrena, de este modo se buscan los hiperparámetros, y se va seleccionando el conjunto de hiperparámetros que tuvo mejor desempeño.

La búsqueda de hiperparámetros se logra de manera inteligente al usar algoritmos como Bayesian Optimization y Successive Halving Algorithm, lo que permite una convergencia hacia los hiperparámetros óptimos con un numero significativamente menor de iteraciones en comparación con Grid Search o Random Search. En la Figura 7 podemos observar una comparación de la reducción del error promedio en cada iteración entre Optuna y algunas metodologías de optimización.

Optuna también permite al usuario decidir bajo qué métrica se evaluarán los hiperparámetros, dando la flexibilidad de maximizar o minimizar la métrica con mayor impacto para la aplicación final del modelo.

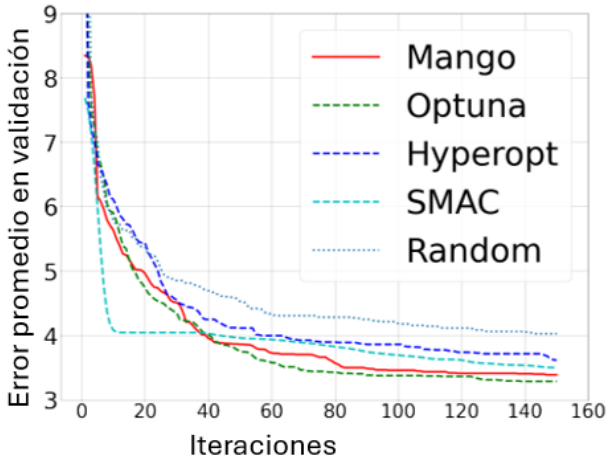

Figura 7. Comparativa de la reducción del error promedio entre Optuna y otras metodologías de búsqueda de hiperparámetros(Sandha et al., 2021).

2.8 Hockey en la NHL

Los partidos de la NHL (Liga Nacional de Hockey) se disputan en una pista de hielo entre dos equipos, cada uno de los cuales cuenta con seis jugadores sobre el hielo en todo momento. El partido suele dividirse en tres periodos de 20 minutos cada uno, con un descanso de 15 minutos tras el segundo periodo (Gu et al., 2019).

El objetivo del juego es marcar más goles que el equipo contrario. Se marca un gol cuando el disco, un pequeño disco de goma, se lanza a la portería del equipo contrario y cruza completamente la línea de gol. El equipo con más goles al final del partido es declarado vencedor.

Al principio de cada partido, los equipos se alinean en sus respectivos extremos de la pista y el árbitro lanza el disco entre dos jugadores contrarios. Esto se conoce como "cara a cara" y se utiliza para empezar el partido, para reanudar el juego tras una interrupción o para resolver infracciones menores de las reglas.

Durante el juego, cada equipo intenta mantener el control del disco y moverlo hacia la portería contraria. Los jugadores pueden utilizar sus palos para pasar el disco a sus compañeros de equipo, disparar a la portería contraria o defender su propia portería bloqueando los tiros y marcando a los jugadores contrarios. Los jugadores también pueden utilizar su cuerpo para proteger el disco de los contrarios o para bloquear a los contrarios y alejarlos del disco.

Si un jugador comete una infracción, como poner una zancadilla o sujetar a un contrario, es enviado al área de castigo durante un tiempo determinado, y su equipo debe jugar con un jugador menos sobre el hielo durante ese tiempo. Esto se conoce como "juego de poder", y puede ser una ventaja para el equipo contrario, que puede intentar marcar mientras tiene un jugador más sobre el hielo.

Los partidos de la NHL son rápidos y físicos, con mucha acción y emoción. El juego requiere una combinación de habilidad, estrategia y trabajo en equipo, ya que los jugadores deben trabajar juntos para mover el disco por el hielo y defender su propia portería.

3. METODOLOGÍA

Esta investigación pretende diseñar un modelo de aprendizaje automático que sea capaz de predecir el ganador o perdedor de un partido de hockey de la NHL, por lo que, para lograr dicho cometido, es necesario encontrar los datos/estadísticas de entrada para el modelo que tengan mayor relevancia con la finalidad de diseñar dicho modelo de predicción haciendo uso de técnicas de aprendizaje automático. Partiendo de esta premisa y describiendo de manera general, se seguirá la metodología descrita a continuación:

3.1 Creación de la base de datos

Para lograr obtener un conjunto de datos completo se siguieron los siguientes lo siguientes pasos:

1. **Obtención de los horarios de los partidos.**

A través del uso de web-scrapping se obtuvieron las fechas en las cuales se dieorn lugar partidos de la temporada regular 2022-2023 y 2023-2024. Haciendo uso de estas fecahs se obtuvieron las estadísticas de la NHL para las dos temporadas regulares.

2. **Recopilación de datos estadísticos.**

Con la intención de obtener los datos estadísticos que se usaran para las predicciones de los modelos de machine learning se utilizaran las fechas obtenidas en las cuales se celebraron partidos durante las dos temporadas regulares, debido a que para obtener los datos de manera satisfactoria usando web-scrapping. Los datos fueron obtenidos a través de la página oficial de la NHL.

3.2 Preprocesamiento de la base datos

Transformar los datos obtenidos con la finalidad de darle limpieza y estandarización a los mismos. Los datos obtenidos anteriormente están estructurados de tal modo que se tienen las estadísticas de cada equipo por separado, de tal modo que, si quisiéramos analizar el comportamiento de los equipos que se están enfrentando en un partido, tendríamos que ver ambos conjuntos de datos (el del equipo jugando en casa, y el del equipo al que se enfrenta).

Debido a que los modelos de aprendizaje automático hacen uso de las estadísticas de ambos equipos que se están enfrentando para obtener una predicción efectiva, se debe conseguir tener los conjuntos de datos de ambos equipos en un solo conjunto de datos.

También hay que tener en cuenta que se busca predecir el ganador de un partido que esta por ocurrir y dado esto se requiere un conjunto de datos que contenga la eficiencia de ambos equipos a través del tiempo. Esto puede lograrse a través de calcular la media móvil de las estadísticas de cada equipo. Para lograr esto se sigue la siguiente serie de pasos:

Paso1: Cálculo de media móvil para cada estadística.

Para cada partido que se celebró de cada equipo se calcularan los promedios o medias móviles, en función del número de partidos jugados hasta ese momento de la temporada. Por ejemplo, si un equipo jugaba su tercer partido, se calculaba la media de las estadísticas de sus dos primeros partidos. Esto permite evaluar el rendimiento del equipo a lo largo del tiempo. Esto se puede calcular haciendo uso de la siguiente formula.

$$SMA = \frac{E_1 + E_2 + E_3 + \cdots + E_n}{n}$$

Ecuación 2. Cálculo de simple promedio móvil (SMA).

Donde:

- $E_1, E_2, E_3, \ldots, E_n$ valor de la estadística durante un partido.
- n: numero de partidos celebrados para el equipo.

Paso 2: Integración de estadísticas de ambos equipos.

Los datos de los equipos locales y visitantes se fusionaron en un único conjunto de características por partido, lo que permite una visión global de cada enfrentamiento. Esta integración es crucial para que el modelo pueda evaluar a ambos equipos simultáneamente.

Paso 3: Creación de métrica de rendimiento defensivo y ofensivo.

Se creará una nueva estadística para medir el dominio de un equipo en los faceoffs ofensivos y defensivos. Se calcula para ambos equipos como la diferencia entre el porcentaje

de victorias en los faceoffs de la zona ofensiva y el porcentaje de victorias en los faceoffs de la zona defensiva. Se calcula a través de la siguiente formula:

$$OZDZ.FO_{diff} = \mathrm{OZ.FO} - \mathrm{DZ.FO}$$

Ecuación 3.Cálculo de métrica de rendimiento defensivo y ofensivo.

Donde:

- $\mathrm{OZ.FO}$ numero de enfrentamientos cara a cara entre dos jugadores en la zona de juego ofensiva para el equipo al cual se mide la estadística.
- $\mathrm{DZ.FO}$ numero de enfrentamientos cara a cara entre dos jugadores en la zona de juego defensiva para el equipo al cual se mide la estadística.

Paso 4: Creación de métrica WPM.

Se creará una nueva métrica que medirá si anteriormente hubo un partido donde se enfrentaran ambos equipos y quien fue el ganador de este.

Paso 5: Creación de métrica WP2M.

Se creará una nueva métrica con la cual se medirá si ambos equipos se han enfrentado uno al otro dos veces y quien ha sido el ganador. De este modose beneficiara al equipo que ha tenido mejor desempeño frente al otro, debido a que, si un equipo ha ganado dos veces frente al otro, tendrá mayor probabilidad de volver a ganar frente al mismo.

Paso 6: Aplicación del método Análisis de Componentes Principales (PCA).

Se creará una nueva base de datos utilizando la técnica de PCA buscando reducir la complejidad de esta, buscando evaluar si contribuye a la predicción usar el método de PCA.

3.3 Selección de modelos de aprendizaje automático para clasificación

A través de la lectura de la bibliografía y de la exploración de los modelos de clasificación existentes, se seleccionarán modelo para el problema de predecir el ganador de partidos de hockey de la NHL. Para la selección se usaron los siguientes dos enfoques:

1.Revisión bibliográfica

Se analizarán los trabajos de investigación existentes de acuerdo con la bibliografía sobre la predicción de ganadores de partidos de la NHL mediante aprendizaje automático. De este modo, se identificarán los modelos de clasificación más utilizados y con mejores resultados para este problema específico.

2.Algoritmos de refuerzo (boosting)

De acuerdo con la bibliografía existente, los modelos de refuerzo no han sido ampliamente explorados, por lo que serán incluidos modelos de refuerzo en la selección de modelo. Dentro del campo de aprendizaje automático los modelos de refuerzo proporcionan una gran precisión en tareas de clasificación.

3.4 Optimización de los modelos mediante Optuna

Para mejorar el rendimiento de los modelos, se llevará a cabo la optimización de hiperparámetros utilizando Optuna, un marco de optimización de hiperparámetros:

3.Selección de métrica a optimizar.

Se utilizará la métrica de exactitud (accuracy) con la intención de incrementarla a través del uso de Optuna.

4.**Definición de función objetivo.**

Se definirá una malla de hiperparámetros a utilizar dependiendo del modelo y sus hiperparámetros disponibles.

5.Creación de callback para early stopping.

Debido a que la malla de hiperparámetros contiene un numero de combinaciones alto, se define una función de early stopping, la cual nos ayudara a detener Optuna en la búsqueda de hiperparámetros, si después de 25 iteraciones el modelo no mejora la exactitud o esta decrece. Esto con la finalidad de reducir el tiempo de entrenamiento en dado escenario que nuestra función objetivo no mejore la precisión.

3.5 Comparación de los resultados de los modelos seleccionados

Se compararon los resultados de cada modelo para identificar el algoritmo más eficaz para predecir los resultados de los partidos.

3.6 Conclusión y discusiones

En base a los resultados obtenidos en la comparación de resultados se dará una conclusión sobre la investigación y se comentaran los hallazgos. Para cerrar se conversará sobre los posibles trabajos o líneas de investigación a explorar.

4. RESULTADOS

A continuación, se presentan los resultados obtenidos tras el desarrollo del modelo de aprendizaje automático para predecir los ganadores de partidos de la NHL en la temporada regular 2022-2023. En los resultados presentados se utilizan cuatro diferentes conjuntos de datos, donde la diferencia entre ellos es una característica añadida que en los demás no existe.

4.1 Comportamiento de los modelos sin optimización ni PCA

El desempeño de los modelos de clasificación sin aplicar técnicas de optimización (Optuna) ni de reducción de dimensionalidad (PCA), se obtuvieron los resultados que se muestran en la siguiente tabla en términos de exactitud (accuracy). Los modelos evaluados fueron Logistic Regression, Random Forest Classifier, XGBoost y CatBoost, como se muestra en la Tabla 2.

DataFrame	LogisticRegression	RandomForest	XGBoost	CatBoost
RawData	57.24 %	56.89 %	55.71 %	55.52%
OZDZ	57.75 %	55.90 %	57.17 %	56.41 %
NHL_DF_WPM	56.85 %	57.60 %	56.34 %	56.05 %
NHL_DF_WP2M	56.80 %	56.33 %	56.92 %	55.68 %

Tabla 2. Desempeño de los modelos sin Optimización ni PCA.

Como se puede observar en la tabla 2, el modelo con mejor rendimiento fue Logistic Regression con una exactitud de 57.75% en el conjunto de datos OZDZ. Sin embargo, al observar el desempeño del modelo a través de las predicciones hechas por día, podemos ver que al inicio de los días de la temporada, el modelo comienza con una exactitud pobre, comenzando desde el 0% de exactitud. Pero conforme avanza la temporada regular, el modelo aprende de los datos y consigue un mejor desempeño.

Esto puede ser debido a que al inicio de la temporada se cuenta con menos datos de entrenamiento, por lo tanto, el modelo tiene una baja cantidad de muestras de las cuales extraer la información necesaria para clasificar con éxito el ganador del partido.

En la siguiente imagen se muestra el desempeño de los cuatro modelos a través de la temporada sin hacer uso de optimización ni PCA. La Figura 8 presenta esta comparativa, donde se observa la evolución del rendimiento de cada modelo a través del tiempo.

Podemos observar que, aunque LogisticRegression tiene un mejor desempeño a largo plazo, el modelo XGBoost tiene un mejor desempeño al inicio de la temporada, por lo que una combinación de ambos modelos podría ser una opción a la hora de querer mejorar el desempeño al inicio de la temporada.

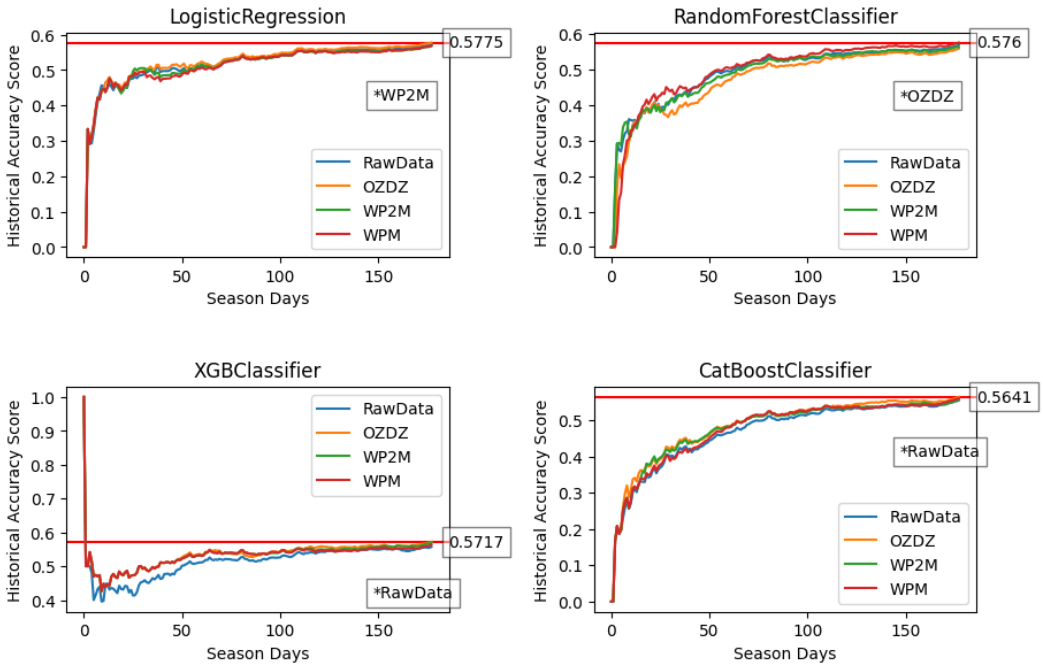

Figura 8. Comparativa del desempeño de los modelos a través de los diferentes datasets sin optimización ni uso de PCA.

4.2 Impacto de la reducción de la dimensionalidad (PCA) sin optimización (Optuna)

A continuación, se evalúa el impacto de la metodología PCA sin el uso de optimización de los modelos. En la siguiente tabla podemos observar el desempeño de los modelos al usar este enfoque, como se muestra en la Tabla 3.

DataFrame	LogisticRegression	RandomForest	XGBoost	CatBoost
RawData	55.84 %	54.03 %	52.35 %	52.76%
OZDZ	55.25 %	54.79 %	53.97 %	54.74 %
NHL_DF_WPM	55.34 %	55.57 %	52.92 %	55.71 %
NHL_DF_WP2M	56.34 %	54.64 %	52.10 %	56.53 %

Tabla 3. Desempeño de los modelos con PCA sin Optimización.

En general, el uso de PCA sin optimización redujo la exactitud de los modelos. El mayor impacto negativo se observa en el modelo XGBoost, donde el modelo paso de un desempeño del 56.92 % sin optimización a 52.10% usando PCA y bajo el mismo conjunto de datos. Esto indica que la reducción de dimensionalidad mediante PCA, sin un ajuste adecuado de hiperparámetros, no es beneficioso para el modelo e incluso podría descartar información importante y relevante para la predicción.

En la Figura 9 se presenta la evolución del desempeño de los modelos a lo largo de la temporada utilizando PCA sin optimización.

También se puede observar a través de la siguiente imagen que el desempeño de las predicciones de los modelos a través de la temporada regular no obtuvo una mejora significante, e incluso podemos observar que en caso contrario al caso anterior donde no se optimizaron los modelos ni se usó PCA, el modelo XGBoost obtuvo un mejor desempeñp al inicio de la temporada. Sin embargo, en esta metodología, podemos observar que el modelo tuvo un peor desempeño al inicio, obteniendo predicciones con una exactitud del 0%.

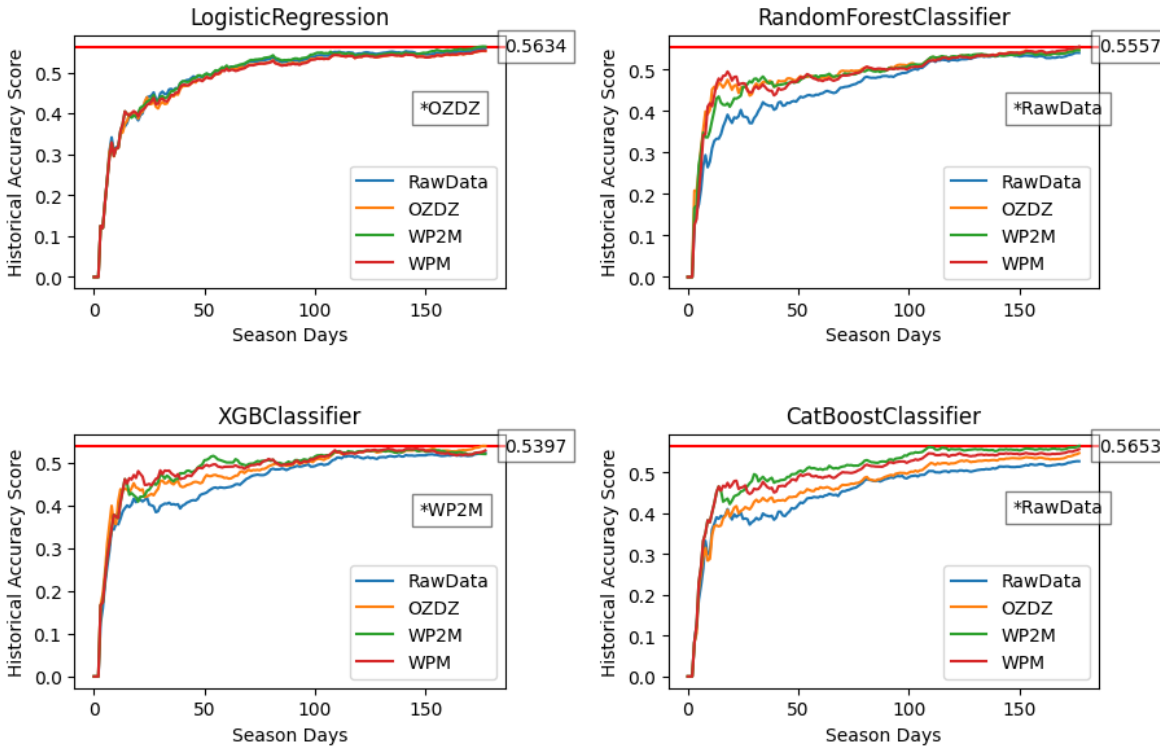

Figura 9. Comparativa del desempeño de los modelos a través de los diferentes datasets haciendo uso de PCA sin optimización.

4.3 Impacto de la optimización con Optuna en el rendimiento de los modelos

El uso de Optuna se hizo con el objetivo de optimizar los hiperparámetros de los modelos seleccionados: logistic Regression, Random Forest, XGBoost y CatBoost. La métrica utiliza para evaluar el rendimiento fue la exactitud, donde se buscaba incrementar la misma a través de buscar la combinación de hiperparámetros con mayor beneficio.

Se observo que la optimización a través de Optuna mejora el rendimiento de los modelos en términos de exactitud en comparación con los modelos donde no se utilizó optimización de hiperparámetros, como se muestra en la Tabla 4.

DataFrame	LogisticRegression	RandomForest	XGBoost	CatBoost
RawData	56.46 %	57.44 %	55.14 %	57.44%
OZDZ	57.89 %	58.14 %	56.03 %	56.02 %
NHL_DF_WPM	56.72 %	56.82 %	55.93 %	57.29 %
NHL_DF_WP2M	59.06 %	55.96%	58.20 %	55.62 %

Tabla 4. Desempeño de los modelos con Optimización sin PCA.

La optimización con Optuna mejora significativamente el desempeño de los modelos en la mayoría de los casos. En particular, se puede observar que el modelo LogisticRegression con el conjunto de datos WP2M obtuvo un desempeño del 59.06% en comparación con el 57.75% de exactitud obtenido al entrenar el modelo en el conjunto de datos OZDZ sin hacer uso de Optimización ni PCA.

En la Figura 10 se muestra la evolución del desempeño de los modelos optimizados a lo largo de la temporada, permitiendo observar cómo la optimización afecta el comportamiento de las predicciones día a día.

Sin embargo, en algunos casos como por ejemplo el modelo XGBoost con el conjunto de datos RawData sin optimización ni PCA se obtuvo una exactitud del 55.71%, la cual es mayor a la obtenida con la optimización y el mismo conjunto de datos, alcanzando un 55.14%. Donde la exactitud en caso contrario a los demás casos cae un 0.57%.

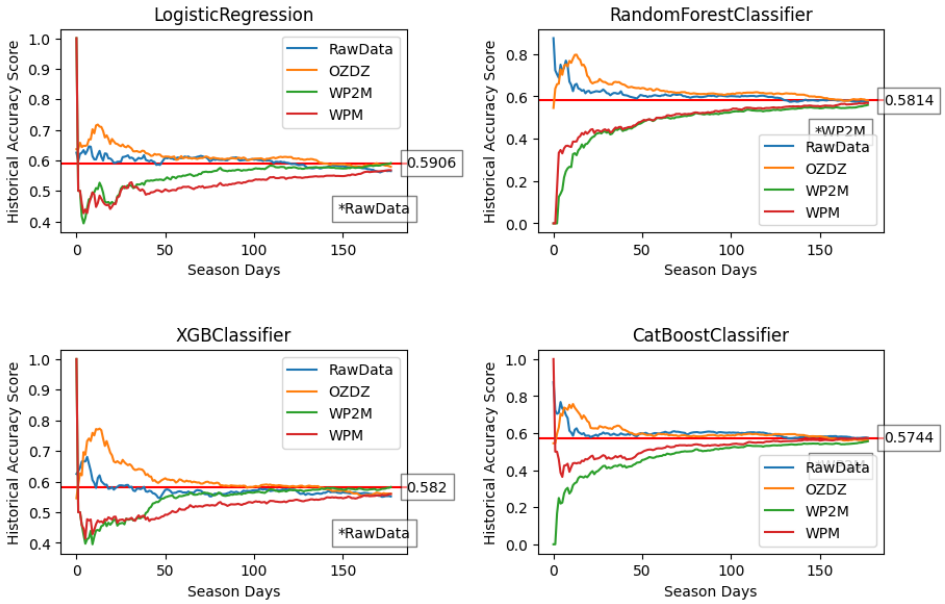

Figura 10. Comparativa del desempeño de los modelos a través de los diferentes datasets haciendo uso de optimización.

También podemos observar a partir de la imagen anterior, que el desempeño de los modelos a través de la temporada mejoro notablemente en comparación a los modelos sin optimización ni PCA en los cuales se obtuvieron los mejores resultados descartando los modelos optimizados con Optuna.

Podemos observar que los conjuntos de datos RawData y OZDZ, mantienen un mejor desempeño a través de la temporada, incluso obteniendo un desempeño notable al inicio de la temporada en los cuatro modelos.

4.4 Impacto de la optimización con Optuna y haciendo uso de PCA

En este apartado se evaluó el impacto de la combinación de la optimización de hiperparámetros mediante Optuna y la reducción de dimensionalidad utilizando PCA. Aunque el objetivo era observar una mejora significativa tanto en la exactitud de los modelos como en su desempeño general, los resultados no superaron a los obtenidos con la optimización por Optuna sin el uso de PCA, como se muestra en la Tabla 5.

DataFrame	LogisticRegression	RandomForest	XGBoost	CatBoost
RawData	56.83 %	53.80 %	51.86 %	53.01%
OZDZ	57.14 %	53.97 %	55.28 %	54.83 %
NHL_DF_WPM	55.53 %	55.59%	58.34 %	55.76 %
NHL_DF_WP2M	56.90 %	55.14%	55.58 %	55.23 %

Tabla 5. Desempeño de los modelos con Optimización y haciendo uso de PCA.

Por los resultados obtenidos podemos observar que el uso de la metodología PCA en conjunto con la optimización de los modelos a travs de Optuna no obtuvo una mejor precisión, e incluso, bajo la precisión del mejor modelo en optimización sin PCA por 2 puntos porcentuales, por lo que paso de 59.06% a 56.90%. Esto nos deja ver que la reducción de dimensionalidad no otorga un mejor desempeño a Logistic Regression, e incluso con los modelos de boosting no se logró una mejora significativa.

En la Figura 11 se muestra la evolución del desempeño de los modelos optimizados con PCA a lo largo de la temporada, permitiendo visualizar cómo esta combinación afecta las predicciones día a día en comparación con los otros enfoques evaluados.

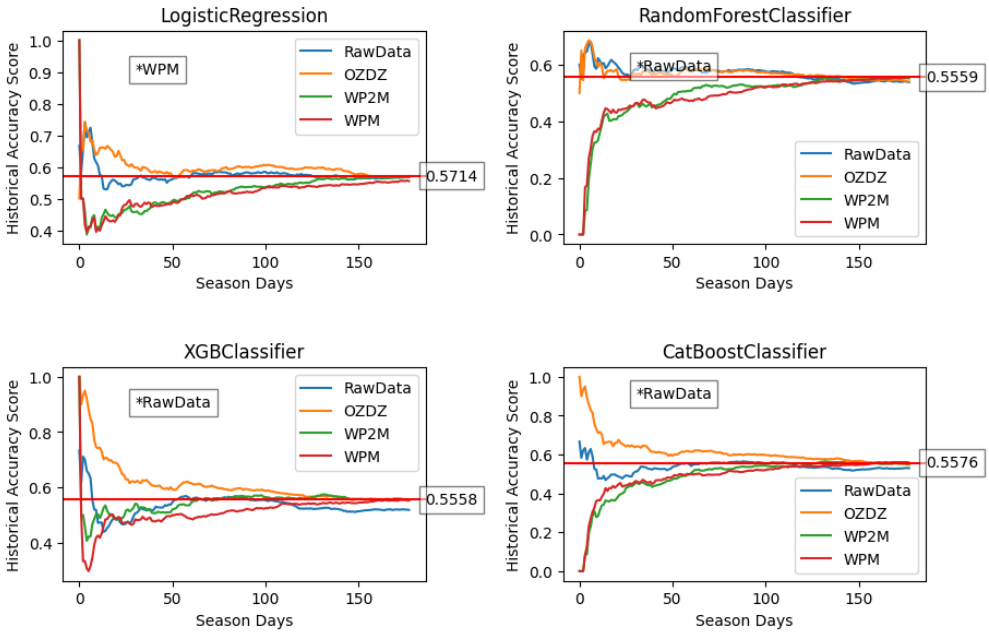

Figura 11. Comparativa del desempeño de los modelos a través de los diferentes datasets haciendo uso de optimización y de la metodología PCA

A través del uso de PCA con Optimización podemos observar también que los modelos de boosting (XGBoost y CatBoost) tienen un mejor inicio de temporada en comparación con los modelos haciendo uso solo de optimización, sin embargo, a medida que se acerca al final de temporada podemos observar que el desempeño de los modelos empeora. Es destacable el conjunto de datos OZDZ es el que tiene un mejor rendimiento para ambos modelos de boosting, aunque, sin superar al método de optimización sin PCA.

4.5 Enfoque más efectivo para predecir el ganador

Con base en los resultados obtenidos, el enfoque más efectivo para predecir al ganador de los partidos de la NHL en la temporada regular 2022-2023 fue la optimización de hiperparámetros mediante Optuna sin aplicar reducción de dimensionalidad (PCA).

Particularmente, el modelo de Logistic Regression optimizado con el conjunto de datos NHL_DF_WP2M logró la mayor exactitud, alcanzando un 59.06%, superando los resultados de todos los demás métodos evaluados.

Aunque se observaron mejoras en varios modelos con la optimización de Optuna, la combinación de Optuna y PCA no logró mejorar la exactitud; en la mayoría de los casos, incluso redujo el desempeño de los modelos.

Esto confirma que, para este problema específico de predicción en la NHL, optimizar directamente los modelos sobre el conjunto de datos completo es más efectivo que aplicar reducción de características, ya que eliminar dimensiones importantes puede llevar a la pérdida de información crítica para la predicción.

4.6 Comparación general de los modelos y enfoques

De manera general, se pueden extraer las siguientes observaciones de la comparación de los enfoques:

- Sin Optimización ni PCA: Logistic Regression fue el mejor modelo, con un 57.75% de exactitud usando el dataset OZDZ. Sin embargo, los modelos mostraban bajo desempeño al inicio de la temporada debido a la escasez de datos de entrenamiento.
- Uso de PCA sin Optimización: El desempeño de todos los modelos disminuyó. La reducción de dimensionalidad eliminó información útil, afectando negativamente especialmente a XGBoost, que cayó más de 4 puntos porcentuales.

- Optimización con Optuna (sin PCA): Mejoró el rendimiento de casi todos los modelos. Logistic Regression y Random Forest mostraron incrementos importantes en exactitud. Además, la precisión de las predicciones al inicio de la temporada fue notablemente superior a la de los modelos sin optimización.
- Optimización con Optuna + PCA: Esta combinación no logró mejorar los resultados respecto a la optimización sin PCA. En algunos casos, incluso degradó el desempeño, reduciendo hasta 2 puntos porcentuales en exactitud. Aunque los modelos de boosting (XGBoost y CatBoost) presentaron mejor arranque de temporada con esta metodología, su rendimiento disminuyó en las etapas finales.

En resumen, optimizar hiperparámetros directamente sobre los datos sin reducción de dimensionalidad fue el enfoque más robusto y consistente.

5. CONCLUSIONES

A partir del desarrollo y evaluación de los diferentes enfoques para predecir el ganador de partidos de la NHL, se pueden establecer las siguientes conclusiones:

- La optimización de hiperparámetros es crucial para mejorar el desempeño predictivo de los modelos. Aplicar Optuna permitió alcanzar una mayor exactitud en comparación con el uso de modelos base.
- La reducción de dimensionalidad mediante PCA, sin un ajuste cuidadoso, resultó ser contraproducente en este caso, ya que eliminó información relevante que afectó negativamente las predicciones.
- El modelo de Logistic Regression, optimizado mediante Optuna sobre el dataset NHL_DF_WP2M, resultó ser el mejor enfoque, alcanzando una exactitud de 59.06%.
- Los modelos presentan una evolución en el desempeño a lo largo de la temporada: el desempeño inicial es menor debido a la falta de datos, pero mejora a medida que avanza la temporada y se acumulan más datos de entrenamiento.
- Una combinación de enfoques podría ser considerada: dado que modelos como XGBoost mostraron mejor desempeño al inicio de la temporada, mientras Logistic Regression fue superior a largo plazo, diseñar un sistema híbrido podría llevar a mejores resultados generales.

Estos hallazgos destacan la importancia de adaptar la estrategia de modelado al contexto específico del problema, y de realizar una evaluación cuidadosa de cada etapa del proceso de machine learning, desde el preprocesamiento de datos hasta la selección y optimización de modelos.

6. RECOMENDACIONES

Con base en los resultados obtenidos y el análisis realizado, se proponen las siguientes recomendaciones para futuros trabajos o mejoras en la predicción de ganadores de partidos en la NHL:

- Aplicar técnicas de optimización de hiperparámetros como parte fundamental del proceso de modelado. Herramientas como Optuna demostraron ser efectivas para mejorar el rendimiento, por lo que su uso es altamente recomendable.
- Evitar el uso de reducción de dimensionalidad (PCA) de manera automática. Si se desea aplicar PCA u otras técnicas similares, se recomienda hacerlo de forma controlada, evaluando su impacto en el rendimiento en cada modelo particular, y considerando métodos de selección de características basados en importancia de variables.
- Explorar estrategias de modelos híbridos: dado que distintos modelos tienen mejores desempeños en diferentes etapas de la temporada, sería interesante combinar modelos (por ejemplo, usar XGBoost al inicio de temporada y Logistic Regression en etapas avanzadas).
- Incrementar el volumen y calidad de los datos de entrenamiento: el bajo desempeño observado al inicio de la temporada podría mitigarse si se utilizan datos históricos de temporadas anteriores o simulaciones que permitan construir un set de datos inicial más robusto.
- Considerar la implementación de técnicas de aprendizaje incremental (online learning), especialmente para situaciones en las que los datos van llegando de manera secuencial durante la temporada.
- Realizar un análisis más profundo de las características utilizadas: incorporar nuevas variables, como estadísticas avanzadas de jugadores o condiciones externas (lesiones, localía, rachas de victorias/derrotas) podría mejorar significativamente la capacidad predictiva de los modelos.
- Evaluar otros algoritmos de boosting como LightGBM o métodos de ensamblado (ensemble learning) que combinen las predicciones de varios modelos para intentar superar la exactitud lograda hasta ahora.

7. DISCUSIÓN

La discusión de los resultados obtenidos permite contextualizar el rendimiento de los distintos modelos evaluados de acuerdo con el estado del arte y analizar cómo influyen factores como la optimización de hiperparámetros, la reducción de dimensionalidad (PCA) y la disponibilidad temporal de datos (datos por temporada).

En primer lugar, el desempeño máximo alcanzado en este estudio fue 59.06% utilizando Logistic Regression optimizado con Optuna sobre el conjunto de datos NHL_DF_WP2M. Lo cual muestra una notable similitud con los resultados reportados en trabajos previos. Estudios como los de (Weissbock et al., 2013) y (Weissbock, 2014) obtuvieron precisiones entre el 59% y 60% utilizando modelos más complejos basados en redes neuronales y ensambles. El hecho de que un modelo lineal, con una complejidad computacional sustancialmente menor, alcance un rendimiento comparable evidencia que, para datos tabulares y de escala moderada, los algoritmos tradicionales pueden ser igual de competitivos que técnicas avanzadas. Este hallazgo respalda la hipótesis planteada en este trabajo: la complejidad del modelo no garantiza un mejor desempeño, especialmente en contextos donde la información disponible es limitada o presenta estructuras estadísticas simples.

Al comparar estos resultados con trabajos más recientes como (Gu et al., 2019) que reportan precisiones superiores al 90%, es importante considerar las diferencias metodológicas. Dichos estudios incorporan un conjunto más amplio de variables, métricas avanzadas y estrategias de ensamble más elaboradas, así como información de múltiples temporadas. En contraste, el presente estudio se basa exclusivamente en datos de una sola temporada y en un conjunto reducido de características numéricas. Bajo este contexto, el rendimiento obtenido resulta consistente y razonable respecto a la literatura.

En relación con los modelos de boosting (XGBoost y CatBoost), los resultados revelan un comportamiento distinto al reportado en trabajos donde estos modelos han mostrado capacidades superiores para la clasificación de datos tabulares. En este estudio, XGBoost y CatBoost no superaron a Logistic Regression en ninguna de las configuraciones evaluadas. Incluso tras la optimización de hiperparámetros, su rendimiento se mantuvo por debajo del modelo lineal. Esta discrepancia puede explicarse mediante dos factores principales:

1. La limitada cantidad de muestras, que reduce la ventaja de modelos complejos sensibles al sobreajuste, y
2. La estructura probabilística de las características, que parece estar mejor capturada por un modelo lineal.

Estos resultados sugieren que, en escenarios con datos moderados y comportamiento estadístico estable, los modelos de boosting pueden no ofrecer un beneficio significativo sin un proceso extensivo de ajuste y sin un volumen mayor de información.

Por otro lado, la evolución temporal del rendimiento de los modelos a lo largo de la temporada muestra tendencias coherentes con lo reportado por(Weissbock, 2014). Al inicio de la temporada, cuando la cantidad de datos disponibles es menor, el nivel de acierto de los modelos es considerablemente más bajo. A medida que se acumulan partidos y las distribuciones estadísticas se estabilizan, los modelos mejoran su desempeño. Un hallazgo relevante es que XGBoost presenta un mejor arranque en algunas configuraciones, pero su rendimiento decrece hacia el final de la temporada, lo que sugiere un sobreajuste inicial. En contraste, Logistic Regression muestra una curva de aprendizaje más estable y progresiva, minimizando la variabilidad entre las etapas iniciales y finales de la temporada.

En cuanto a la reducción de dimensionalidad mediante PCA, se puede observar que este método no resulta en una mejora del nivel de predicción en este escenario. PCA sin optimización y PCA con Optuna tuvieron una reducción de precisión en la gran mayoría de los modelos, aún más en el caso de XGBoost. Esto está en línea con el estudio previo, que indicaba que las características originales tenían información estructural significativa que se diluyó o se perdió durante su conversión en PCA, ya que se sabe en esta literatura que PCA no siempre mejora el rendimiento cuando se encuentra con una dimensionalidad inicial pequeña, o cuando las características ya tienen una interpretación clara. De hecho, la disminución en la precisión cuando se usa PCA junto con Optuna confirma esta afirmación e indica que la reducción de dimensionalidad no es adecuada para este tipo de datos para la predicción de partidos de NHL por sí sola.

En conjunto, los resultados obtenidos permiten afirmar que:

- La optimización de hiperparámetros es un factor determinante en el rendimiento de los modelos.
- Los modelos simples y lineales pueden ser más robustos y generalizables en escenarios con datos limitados.
- Los modelos complejos no necesariamente superan a los tradicionales sin un volumen de datos suficientemente grande.
- La reducción de dimensionalidad mediante PCA no beneficia a este tipo de modelos ni a este conjunto de características.
- El comportamiento temporal del rendimiento confirma observaciones previas en la literatura acerca de la dificultad inherente del inicio de temporada.

Finalmente, los hallazgos refutan la noción frecuentemente argumentada de que los modelos complejos son siempre mejores para las predicciones deportivas. Por el contrario, este estudio muestra que cuando los datos son tabulares, relativamente simples y de volumen moderado, los modelos tradicionales (complementados con una adecuada optimización) pueden igualar o superar a alternativas más sofisticadas. Esto nos ayuda a saber, además, qué tipos de modelos son adecuados para la predicción de partidos de la NHL, lo que sienta las bases para futuras investigaciones como se describe.

8. LIMITACIONES

Aunque surgieron estrategias prometedoras para predecir los partidos de la NHL y el presente estudio encuentra resultados efectivos, estos últimos tienen varias limitaciones que

restringen la generalización y el alcance de este estudio. En primer lugar, el tamaño y la cobertura temporal del conjunto de datos constituyen una limitación importante. El análisis se realizó exclusivamente en la temporada regular 2022-2023, lo que reduce la capacidad de los modelos para abordar la variabilidad interanual, los efectos contextuales o los cambios estructurales en el rendimiento de los equipos. Dicha limitación es particularmente relevante en los juegos deportivos, como los partidos, donde circunstancias como transferencias, lesiones y modificaciones tácticas tienen el potencial de cambiar la dinámica competitiva de juego de una temporada a otra.

En segundo lugar, los modelos indican una fuerte dependencia de la disponibilidad de información en las primeras semanas de la temporada. La precisión obtenida en estas etapas es algo menor debido a la escasez de datos históricos, lo que limita la aplicabilidad de los modelos en escenarios de predicción temprana o apuestas a corto plazo. Y, aunque se realizaron algunas metodologías de optimización como PCA y ajuste de hiperparámetros con Optuna, no se emplearon todos los enfoques de preprocesamiento y modelado. Métodos que incluyen, pero no se limitan a, la selección automática de características, normalizaciones más avanzadas o combinaciones de varios modelos para aumentar la estabilidad y precisión fueron excluidos.

La otra limitación es la calidad de las características utilizadas. Aunque se utiliza una amplia variedad de estadísticas, no se realizó un análisis exhaustivo de la importancia de las variables y no se descartaron aquellas potencialmente redundantes o generadoras de ruido. Esto puede afectar la capacidad de generalización de los modelos.

Por último, pero no menos importante, hemos evaluado el rendimiento con respecto a la métrica de precisión. Aunque es una métrica intuitiva y comúnmente empleada, puede no representar completamente el rendimiento del modelo cuando un sistema está desequilibrado o la predicción tiene una incertidumbre relativamente grande. La inclusión de otras métricas como F1-score, ROC-AUC o LogLoss puede proporcionar una interpretación más completa de las capacidades predictivas.

9. TRABAJO FUTURO

A partir de las restricciones mencionadas anteriormente y las observaciones que surgieron, se presentan un conjunto de recomendaciones para futuros desarrollos con el fin de aumentar el poder predictivo, la generalización y el rigor metodológico de los modelos obtenidos.

En primer lugar, sería aconsejable enriquecer el conjunto de datos incluyendo múltiples temporadas, información de playoffs y estadísticas históricas adicionales. Esto permitirá entrenar modelos que aprendan patrones más estables y generales.

En segundo lugar, se deben investigar modelos de aprendizaje profundo (por ejemplo, redes neuronales LSTM o modelos tipo transformer) que puedan representar mejor la evolución temporal del rendimiento de equipos y jugadores. Otra propuesta consiste en implementar técnicas de aprendizaje en línea, donde el modelo se actualiza a medida que avanza la temporada o incluye variables que introducen eventos inesperados como lesiones, cambios climáticos, etc. Este enfoque podría mejorar la adaptación del sistema a cambios repentinos en el rendimiento deportivo.

También sugerimos evaluar otras métricas más allá de la precisión para obtener una evaluación más completa del rendimiento, especialmente cuando los datos no están equilibrados.

Por último, se podrían desarrollar sistemas basados en aprendizaje en conjunto, que consolidarían las predicciones de diferentes algoritmos. Este enfoque generalmente produce resultados más estables y precisos y podría mejorar la capacidad predictiva en el contexto de la NHL.

REFERENCIAS

[1] T. Akiba, S. Sano, T. Yanase, T. Ohta, y M. Koyama, “Optuna: A next-generation hyperparameter optimization framework,” arXiv preprint arXiv:1907.10902, 2019. [En línea]. Disponible en: http://arxiv.org/abs/1907.10902

[2] C. Albon, Machine Learning with Python Cookbook: Practical Solutions from Preprocessing to Deep Learning. O’Reilly Media, 2018. ISBN: 978-1-491-98938-8.

[3] R. P. Bunker y F. Thabtah, “A machine learning framework for sport result prediction,” Applied Computing and Informatics, vol. 15, n.º 1, pp. 27–33, 2019, doi: 10.1016/j.aci.2017.09.005.

[4] R. Bunker y T. Susnjak, “The application of machine learning techniques for predicting match results in team sport: A review,” Journal of Artificial Intelligence Research, vol. 73, pp. 1285–1322, Apr. 2022. doi: 10.1613/jair.1.13509.

[5] T. Chen y C. Guestrin, “XGBoost: A scalable tree boosting system,” en Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, doi: 10.1145/2939672.2939785.

[6] A. Géron, Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. O’Reilly Media, 2019. ISBN: 978-1492032632.

[7] W. Gu, K. Foster, J. Shang, y L. Wei, “A game-predicting expert system using big data and machine learning,” Expert Systems with Applications, vol. 130, pp. 293–305, 2019, doi: 10.1016/j.eswa.2019.04.025.

[8] P. Harrington, Machine Learning in Action. Manning Publications, 2012. ISBN: 978-1617290183.

[9] IBM, “What is machine learning?” [En línea]. Disponible en: https://www.ibm.com/topics/machine-learning

[10] R. Mitchell, Web Scraping with Python: Collecting More Data from the Modern Web. O’Reilly Media, 2015. ISBN: 978-1-491-98557-1.

[11] A. C. Müller y S. Guido, Introduction to Machine Learning with Python: A Guide for Data Scientists. O’Reilly Media, 2016. ISBN: 978-1-449-36941-5.

[12] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, y A. Gulin, “CatBoost: Unbiased boosting with categorical features,” arXiv preprint arXiv:1706.09516, 2017. [En línea]. Disponible en: http://arxiv.org/abs/1706.09516

[13] S. Raschka y V. Mirjalili, Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow 2. Packt Publishing, 2019. ISBN: 978-1-78995-575-0.

[14] S. S. Sandha, M. Aggarwal, S. S. Saha, y M. Srivastava, “Enabling hyperparameter tuning of machine learning classifiers in production,” en 2021 IEEE 3rd International Conference on Cognitive Machine Intelligence (CogMI), 2021, pp. 262–271, doi: 10.1109/CogMI52975.2021.00041.

[15] J. Weissbock, “Forecasting success in the National Hockey League using in-game statistics and textual data,” 2014. doi: 10.20381/ruor-6351.

[16] J. Weissbock y D. Inkpen, “Combining textual pre-game reports and statistical data for predicting success in the National Hockey League,” en Advances in Artificial Intelligence, 2014, doi: 10.1007/978-3-319-06483-3_22.

[17] J. Weissbock, H. L. Viktor y D. Inkpen, “Use of performance metrics to forecast success in the National Hockey League,” en Proceedings of MLSA@PKDD/ECML, 2013. [En línea]. Disponible en: https://api.semanticscholar.org/CorpusID:42267309

[18] National Hockey League, “Official website of the NHL,” NHL.com. [En línea]. Disponible en: https://www.nhl.com.

CURRICULUM VITAE

Miguel Gutiérrez es Ingeniero en Mecatrónica por el Instituto Tecnológico de Parral (2018) y Maestro en Ingeniería en Computación con especialidad en Inteligencia Artificial por la Universidad Autónoma de Chihuahua (2024). Su formación académica se ha complementado con certificaciones en automatización de procesos, programación full stack (JavaScript y React), ciencia de datos con Python y desarrollo de soluciones mediante RPA.

Su experiencia profesional incluye más de seis años en áreas de tecnología, automatización y gestión técnica. Entre 2019 y 2021 se desempeñó como Programming Engineer en Autronic S.A. de C.V., donde participó en proyectos de automatización industrial, programación de PLCs, robótica, integración de sistemas de control y diseño de arquitecturas de comunicación industrial para los sectores minero y manufacturero. Posteriormente, en Jabil, ocupó cargos relacionados con la gestión de procesos, automatización y análisis de datos. Como Sr. Enterprise Resources Planning SME trabajó en la automatización de procesos administrativos, optimización de flujos de información y desarrollo de herramientas computacionales para la mejora operativa. A partir de 2025 asumió el rol de SME Assistant Manager, donde coordina un equipo especializado en procesos de materiales, automatización computacional y estandarización de procedimientos mediante tecnologías como Power Automate, Blue Prism, VBA, Python y Power BI, además de liderar iniciativas de transformación digital y mejora continua.

Durante su trayectoria ha desarrollado soluciones de software para automatización operativa, herramientas basadas en machine learning para análisis de información, así como sistemas de control para entornos industriales. Aunque aún no cuenta con publicaciones arbitradas, su trabajo de investigación aplicado al modelado predictivo y optimización de modelos de aprendizaje automático contribuye al desarrollo de tecnologías para la predicción deportiva y el análisis basado en datos.

Puede ser contactado al correo electrónico lopez.mgu@gmail.com
Domicilio Permanente: Chihuahua, Chihuahua, Mexico.

Esta tesis/disertación fue mecanografiada por Miguel Angel Gutierrez Lopez.