

UNIVERSIDAD AUTÓNOMA DE CHIHUAHUA

FACULTAD DE INGENIERÍA

SECRETARÍA DE INVESTIGACIÓN Y POSGRADO



**MODELOS DE MACHINE LEARNING PARA LA PREDICCIÓN DE
RESULTADOS DE FÚTBOL**

POR:

ADALBERTO EMMANUEL ROJAS PEREA

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

DIRECTOR DE TESIS: DR. LUIS FERNANDO GAXÍOLA ORDUÑO

CHIHUAHUA, CHIH., MÉXICO

MAYO 2026



Modelos De Machine Learning Para La Predicción De Resultados De Fútbol. Tesis, presentada por Adalberto Emmanuel Perea Rojas 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 Fernando Gaxiola Orduño
Director de Tesis

Junio 2026

Fecha

COMITÉ

Dr. Luis Fernando Gaxiola Orduño
Dr. Luis Carlos González Gurrola
Dr. Jesús Roberto López Santillán
Dr. Alain Manzo Martínez



UNIVERSIDAD AUTÓNOMA DE
CHIHUAHUA

29 de junio de 2026.

ING. ADALBERTO EMMANUEL ROJAS PEREA
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 la directora **Dr. Luis Fernando Gaxiola Orduño** para que lo desarrolle como Tesis, con el título **“Modelos De Machine Learning para la predicción de resultados de fútbol”**.

Índice de Contenido

Agradecimientos

Resumen

Abstract

Índice de Contenido

Índice de tablas

Índice de figuras

Capítulo 1: Introducción

Capítulo 2. Trabajo relacionado

Capítulo 3: Marco Teórico

3.1. Machine Learning (ML)

3.2. Aprendizaje Supervisado

3.3. Algoritmos de Machine Learning

3.4. Validación Cruzada

Capítulo 4. Metodología

4.1 Recolección de Datos

4.2 Preprocesamiento de Datos

4.3 Ingeniería de Características (Feature Engineering)

4.4 Base de Datos Resultante

4.5 Estrategia de Validación Cruzada (Time Series Split)

4.6 Selección y Entrenamiento de Modelos

Capítulo 5. Análisis y Discusión de Resultados

5.1 Proceso de Evaluación de los Modelos



Por un
2026
con salud mental



UNIVERSIDAD AUTÓNOMA DE
CHIHUAHUA

5.2 Comparativa de Desempeño y efecto de la Optimización
5.3 Análisis de Importancia de Variables (Feature Importance).
5.4 Análisis por Clase y Matriz de Confusión
5.5 Discusión de resultados
Capítulo 6. Conclusiones y Recomendaciones
6.1 Conclusiones
6.2 Limitaciones
6.3 Recomendaciones
Referencias
Currículum Vitae

ATENTAMENTE

"naturam subiecit aliis"

EL DIRECTOR

**M.I. FABIÁN VINICIO HERNÁNDEZ
MARTÍNEZ**

**FACULTAD DE
INGENIERÍA**

**U.A.CH. SECRETARIO DE INVESTIGACIÓN
Y POSGRADO**



M.I. RODRIGO DE LA GARZA AGUILAR

DIRECCIÓN

Dedicatoria

*A mi esposa, Lucero,
compañera de vida y apoyo incondicional, por su paciencia, comprensión y fortaleza en cada momento
de este proceso.*

*A mis hijos, Emmanuel y Anneth,
presente y futuro de mi vida, razón de cada esfuerzo y cada meta alcanzada.
Que este logro represente no solo una meta académica cumplida, sino un testimonio de
constancia, responsabilidad y amor por la familia.*

Agradecimientos

Expreso mi más sincero agradecimiento al Dr. Luis Fernando Gaxiola Orduño, por su confianza, orientación y constante seguimiento durante el desarrollo de esta investigación. Su acompañamiento académico, observaciones críticas y disposición permanente para revisar los avances fueron fundamentales para la culminación de esta tesis. Su respaldo fortaleció tanto el rigor metodológico del trabajo como mi crecimiento profesional durante este proceso.

Asimismo, agradezco a la Facultad de Ingeniería de la Universidad Autónoma de Chihuahua y al programa de Maestría en Ingeniería en Computación por brindar el espacio académico, los recursos y el entorno formativo que hicieron posible la realización de este proyecto.

Resumen

La predicción de resultados deportivos representa un desafío complejo debido a la naturaleza estocástica e incierta de disciplinas como el fútbol profesional. En este contexto, la presente investigación tiene como objetivo desarrollar y evaluar modelos de *Machine Learning* para la predicción de resultados de partidos en la Liga Inglesa de Fútbol, bajo un enfoque metodológico que preserve la integridad temporal de los datos y evite la fuga de información.

La obtención de los datos representó uno de los principales retos del trabajo, ya que, al no estar disponibles de forma directa, fue necesario recurrir a técnicas de web scraping para extraerlos de internet. Una vez recopilados, se llevó a cabo un proceso de limpieza para descartar datos poco útiles, y a partir de este conjunto resultante se generaron nuevas variables, entre ellas promedios móviles de métricas ofensivas y defensivas. Para la fase de modelado se implementaron cuatro algoritmos de aprendizaje automático: Regresión Logística, Bosques Aleatorios (Random Forest), XGBoost y Máquinas de Vectores de Soporte (SVM). Dado que se trata de predecir resultados deportivos a partir de información histórica, mezclar los datos de forma aleatoria habría comprometido la validez del modelo, por lo que se utilizó una estrategia de validación por series temporales (Time Series Split) que respeta el orden cronológico, reservando los datos de las temporadas 2022 a 2024 para entrenamiento y la temporada 2025 para prueba.

Los resultados muestran que el modelo de Regresión Logística optimizado obtuvo la mejor exactitud, con un 49.60% en el conjunto de prueba. Este valor superó la probabilidad del azar, estimada en 33.33% para un problema de tres clases, así como el 44% que registraría un clasificador básico guiado por la tendencia de la localía. El análisis de importancia de variables (feature importance) permitió identificar que algunas métricas defensivas, especialmente los goles esperados en contra (xGA), tuvieron un papel relevante en la predicción, lo que sugiere que la solidez defensiva aporta información útil para predecir el resultado de un partido.

A partir de los resultados obtenidos, se concluye que los modelos de Machine Learning pueden extraer señales predictivas útiles en un problema altamente variable como los resultados de fútbol, siempre y cuando los datos se estructuren adecuadamente y se respete su secuencia temporal. Sin embargo, también se observó una limitación importante en la predicción de los empates, ya que esta clase presentó un comportamiento difícil de separar mediante las variables utilizadas.

Abstract

Predicting sports outcomes represents a complex challenge due to the stochastic and uncertain nature of professional football. In this context, this research aims to develop and evaluate Machine Learning models for predicting match results in the English Premier League, under a methodological framework that preserves the temporal integrity of the data and prevents data leakage.

Data acquisition was one of the main challenges of this work. Since the data were not directly available, it was necessary to employ web scraping techniques to extract them from the internet. Once collected, a cleaning process was performed to filter out low-utility data. From this resulting dataset, new features were generated, including rolling averages of offensive and defensive metrics. For the modeling phase, four machine learning algorithms were implemented: Logistic Regression, Random Forest, XGBoost, and Support Vector Machines (SVM). Because the objective is to predict sports outcomes using historical information, randomly shuffling the dataset would have compromised model validity. Therefore, a validation strategy based on a Time Series Split was used to respect chronological order, reserving data from the 2022 to 2024 seasons for training and the 2025 season for testing.

The results show that the optimized Logistic Regression model achieved the highest accuracy, reaching 49.60% on the test set. This value outperformed random probability—estimated at 33.33% for a three-class problem—as well as the 44% baseline that a naive classifier guided by home-field advantage trends would register. Feature importance analysis revealed that certain defensive metrics, especially Expected Goals Against (xGA), played a significant role in the prediction, suggesting that defensive solidity provides useful information for predicting match outcomes.

Based on the obtained results, it is concluded that Machine Learning models can extract useful predictive signals in highly volatile problems such as football match outcomes, provided the data is properly structured and its temporal sequence is respected. However, a major limitation was also observed in predicting draws, as this class presented a behavior that remains difficult to separate using the selected variables.



Índice de Contenido

Agradecimientos	iii
Resumen	iv
Abstract.....	v
Índice de Contenido	vi
Índice de Tablas	ix
Índice de Figuras.....	x
Capítulo 1: Introducción	1
Objetivos:.....	4
Objetivo General.....	4
Objetivos Específicos.	4
Capítulo 2: Trabajo Relacionado	6
Capítulo 3: Marco Teórico.....	7
3.1: Machine Learning (ML).	8
3.2: Aprendizaje Supervisado	8
3.3: Algoritmos de Machine Learning	10
3.3.1: Árboles de Decisión (Decision Trees).....	10
3.3.2: Regresión Logística	11
3.3.3 Máquinas de Vectores de Soporte (SVM)	12
3.3.4 eXtreme Gradient Boosting (XGBoost)	13
3.4: Validación Cruzada	15
3.4.1: Validación en Series Temporales (Time Series Split).....	16
3.5 Promedios Móviles (Rolling Averages)	16
Capítulo 4: Metodología	17
4.1: Recolección de Datos	19
4.1.1 Arquitectura del proceso de web scraping.....	20
Obtención de páginas web (Data Acquisition):	20
Análisis y extracción de información (Parsing):	20
Transformación y almacenamiento de datos (Structuring):.....	21
4.1.2 Implementación del scraper	21

4.2: Preprocesamiento de Datos	22
Limpieza y manejo de datos faltantes:.....	22
Codificación de variables categóricas:	22
Transformación y escalamiento de variables:.....	23
Análisis de la variable objetivo:	23
4.3: Ingeniería de Características (Feature Engineering):	23
4.3.1: Generación de variables ofensivas y defensivas.....	24
4.4: Base de Datos Resultante	25
4.4.1: Variables predictoras (Features)	25
4.4.2 Variable Objetivo (Target)	26
4.5: Validación temporal de los datos (Time Series Split)	27
4.6: Selección y Entrenamiento de Modelos	28
Configuración del entorno experimental	29
Regresión Logística	29
Tabla 4.1: Descripción de hiperparámetros Regresión Logística.	29
Tabla 4.2: Espacios de búsqueda Regresión Logística.	29
Random Forest.....	30
Tabla 4.3:Descripción de hiperparámetros Random Forest.....	30
Tabla 4.4: Espacios de búsqueda Random Forest.....	30
XGBoost	30
Tabla 4.5: Descripción de hiperparámetros XGBoost.....	30
Tabla 4.6: Espacios de búsqueda XGBoost.....	30
Support Vector Machines (SVM).....	31
Tabla 4.7: Descripción de hiperparámetros SVM.....	31
Tabla 4.8: Espacios de búsqueda SVM.....	31
Proceso de optimización	31
Selección del modelo final.....	31
Capítulo 5: Análisis y Discusión de Resultados	32
5.1 Proceso de Evaluación de los Modelos	32
5.2 Comparativa de Desempeño y efecto de la Optimización.....	32
Tabla 5.1: Comparativa de métricas por modelo (Datos 2025).....	33
Tabla 5.2: Resultados obtenidos para cada modelo	36
5.3 Análisis de Importancia de Variables (Feature Importance)	38
5.4 Análisis por Clase y Matriz de Confusión	39

5.5 Discusión de Resultados	41
Tabla 5.3: Comparación Accuracy con trabajos relacionados	41
Capítulo 6: Conclusiones y Recomendaciones	42
6.1 Conclusiones	42
Cumplimiento del Objetivo General:	43
Importancia de la Solidez Defensiva:	43
Eficiencia de Modelos Lineales en Entornos de Alta Incertidumbre:	43
Naturaleza Estocástica del Empate:	43
6.2 Limitaciones	44
6.3 Recomendaciones	45
Integración de Variables Contextuales:	45
Exploración de Modelos de Aprendizaje Profundo:	45
Uso de Datos Espaciales de Seguimiento:	45
Evaluación Multiliga:	45
Desarrollo de un Modelo Híbrido para la Predicción del Empate:	45
Referencias	47
Curriculum Vitae	50

Índice de Tablas

Tabla 4.1: Descripción de hiperparámetros Regresión Logística.....29

Tabla 4.2: Espacios de búsqueda Regresión Logística.....29

Tabla 4.3: Descripción de hiperparámetros Random Forest30

Tabla 4.4: Espacios de búsqueda Random Forest30

Tabla 4.5: Descripción de hiperparámetros XGBoost.....30

Tabla 4.6: Espacios de búsqueda XGBoost30

Tabla 4.7: Descripción de hiperparámetros SVM31

Tabla 4.8: Espacios de búsqueda SVM31

Tabla 5.1: Comparativa de métricas por modelo (Datos 2025).....33

Tabla 5.2: Resultados obtenidos para cada modelo36

Tabla 5.3: Comparación Accuracy con trabajos relacionados.....41

Índice de Figuras

Figura 1 Arquitectura integral del pipeline de datos. Fuente: Elaboración propia. Representación esquemática del flujo de trabajo metodológico.	17
Figura 2 Representación del proceso de construcción de datos utilizando una ventana deslizable (sliding window) de tamaño 3.	22
Figura 3 Comparación entre validación cruzada K-Fold y Time Series Split para datos con dependencia temporal	26
Figura 4 Comparación Final de Modelos Predictivos (Premier League 2025)	32
Figura 5 Comparativa de perfil de modelos	33
Figura 6 Mapa de calor: Rendimiento por Métrica	34
Figura 7 Comparación entre exactitud y el F1 - score para los modelos evaluados	35
Figura 8 Comparativa de importancia de variables entre modelos lineales y de ensamble	36
Figura 9 Matriz de confusión para el modelo de Regresión Logística (Valores reales vs. Predicciones).	37





Capítulo 1: Introducción

En la actualidad, el deporte desempeña un papel fundamental en nuestra sociedad. Su influencia va mucho más allá de ser una simple actividad física destinada para mejorar o mantener la salud individual. El alcance del deporte se extiende a diversos aspectos de la vida cotidiana, incluyendo su impacto económico y, en consecuencia, su relación con las apuestas deportivas.

Como lo señala Bardisa Serrano en su estudio del 2019 (Bardisa Serrano, 2019), el deporte, en particular el fútbol, ha adquirido un estatus excepcional en la cultura global. Más del 40% de la población se autodefine como aficionada al fútbol, consolidándolo como el deporte más reconocido y seguido en todo el mundo.

El impacto del deporte no se limita al entretenimiento ni a la práctica física, ya que también participa de manera importante en la economía. En el caso del fútbol, existen distintas actividades relacionadas con su desarrollo, como publicidad, mercadotecnia deportiva, venta de boletos, derechos de transmisión y la comercialización de productos oficiales de los equipos. Estos elementos generan ingresos importantes y muestran que el deporte también puede analizarse desde una perspectiva económica.

Dentro de este contexto, las apuestas deportivas han adquirido mayor presencia en los últimos años. Esta situación ha incrementado el interés por estudiar los resultados de los partidos no solo desde la afición, sino también desde el análisis de datos. Por ello, el fútbol puede considerarse un fenómeno deportivo, social y económico, en el que la disponibilidad de información abre la posibilidad de aplicar técnicas computacionales para identificar patrones y apoyar la predicción de resultados.

Especialmente en Europa, en términos de ingresos y el nivel de las competiciones, es el mercado más grande de fútbol en el mundo, llegando en muchas ocasiones a dejar de ser un deporte para pasar a ser una actividad que forma parte de sus vidas diarias, como ya lo menciona Acuña, “El fútbol es parte de nuestra cultura” (Acuña et al., 2016).

El fútbol es un deporte donde las predicciones de resultados son especialmente difíciles, ya que intervienen numerosos factores que influyen en el desarrollo de cada partido. Aunque en otros deportes la estadística se emplea con éxito para anticipar los desenlaces, en el fútbol estas técnicas suelen carecer de la precisión necesaria.



El objetivo de esta investigación no es recurrir a la estadística avanzada para pronosticar los encuentros de las ligas de fútbol, sino explorar el uso de nuevas tecnologías como computadoras, algoritmos, datos, lenguajes de programación y, particularmente, el *Machine Learning* para realizar dichas predicciones.

Existen actividades que para los seres humanos son muy simples y a menudo pasamos por alto, como reconocer de inmediato la voz de una persona o identificar rápidamente un patrón visual. Estas actividades se vuelven sencillas para nosotros debido a que, desde una edad temprana, nuestro cerebro se entrena para reconocer rápidamente sonidos o patrones visuales, y es esencialmente lo que hace el *Machine Learning* con las computadoras: se entrenan para identificar patrones.

Las computadoras pueden llevar a cabo múltiples tareas de manera mucho más eficiente que las diseñadas para realizar, especialmente aquellas tareas repetitivas que consideramos aburridas. Por ejemplo, realizar miles de cálculos en cuestión de segundos es una tarea sencilla para estas máquinas.

El *Machine Learning* tiene una amplia gama de aplicaciones, como motores de búsqueda, reconocimiento de voz, detección de rostros, filtros antispam en correos electrónicos y detección de software malicioso, entre muchas otras. En este contexto, el *Machine Learning* se apoya en el análisis de datos estadísticos para identificar patrones, relaciones y tendencias dentro de grandes volúmenes de información, lo cual resulta especialmente útil en dominios donde los datos históricos juegan un papel fundamental.

En esta investigación, se empleará el *Machine Learning* para predecir eventos, utilizando algoritmos de clasificación, en concreto los resultados de fútbol.

Se plantea como hipótesis que el uso de modelos de *Machine Learning* permitirá desarrollar una arquitectura para predecir de manera eficiente los resultados de juegos de la liga inglesa de fútbol.

Desde hace años las personas han tratado de predecir el resultado de eventos deportivos, utilizando varias técnicas como la estadística y el análisis de datos sin llegar realmente a un porcentaje efectivo y es precisamente ahora con la gran cantidad de información en tiempo real, que es posible utilizar las técnicas de *Machine Learning* para lograr predecir estos resultados tomando en cuenta tantas variables como sean necesarias para ello.



La primera muestra de la aplicación de esta disciplina en el deporte se remonta al año 2002, cuando los Athletics de Oakland, después de una etapa de crisis el año anterior, necesitaban formar un equipo de beisbol competitivo con un presupuesto mucho menor que el resto de sus rivales. Decidieron basarse en la estadística avanzada para crear nuevo equipo a partir de la incorporación de un grupo lleno de figuras infravaloradas por el mercado (Calderón Pérez-Lozao, 2017). A esta técnica se le llamo “Moneyball” y fue en aquel entonces realmente una sorpresa para todos porque llegaron a obtener 20 victorias consecutivas, gracias a esto los demás equipos empezaron a tomar en cuenta este tipo de técnicas para contratar deportistas, por lo que hoy en día los equipos de beisbol cuentan con un departamento de ciencias de datos la cual ayuda a la toma de decisiones. Esto para el fútbol hoy en día ha sido una tarea más complicada.

Como muestra de algunas investigaciones para la predicción de datos, Calderón Pérez-Lozao (Calderón Pérez-Lozao, 2017), señaló que su trabajo consistía en hacer una comparación entre los diferentes tipos de aprendizaje, y algunas de sus conclusiones fueron que con un modelo entrenado relativamente sencillo y con tan solo 5 características utilizando la regresión logística, consiguieron superar las casas de apuestas.

Por su parte, (Calderón Pérez-Lozao, 2017; Medrano Martínez, 2018) menciona que su objetivo fue realizar una investigación exhaustiva sobre la predicción de los resultados de un deporte con las técnicas de *Machine Learning*, y plantea responder algunas preguntas como: ¿Que deportes pueden ser predichos?, ¿Qué tipo de datos se necesitan para las predicciones?, y si ¿es posible conseguir un porcentaje de acierto mejor que la probabilidad aleatoria? Como resultado de su estudio concluyo: “Existen unas claras limitaciones inherentes a la predicción, ya que la aleatoriedad de los deportes supone que ningún resultado puede ser predicho con una certeza absoluta. Sin embargo, en este trabajo final de grado se ha demostrado que es posible, con una muestra de datos lo suficientemente grande, entrenar algoritmos de *Machine Learning* para la predicción de resultados deportivos”.

A su vez, (Soto Valero, 2018) en su artículo mencionó que “El uso de métodos de aprendizaje automático en el contexto deportivo brinda importantes ventajas para su análisis cuantitativo, en comparación con las técnicas estadísticas tradicionales.” Así pues, vemos que es posible la predicción de los resultados deportivos, este proyecto tiene como finalidad



aclarar hasta qué punto es capaz el *Machine Learning* de predecir algo tan complicado como son los resultados de fútbol y comparar trabajo con los modelos utilizados en la literatura.

Objetivos:

Objetivo General.

Desarrollar un modelo para predicción de resultados de los juegos de fútbol, utilizando técnicas de Machine Learning.

Objetivos Específicos.

1. Analizar los distintos modelos de Machine Learning a implementar en la predicción de eventos deportivos.
2. Determinar los modelos más adecuados para la predicción de resultados de fútbol.
3. Establecer la técnica “Scraping” para la recolección de datos históricos de los juegos de la liga de fútbol, incluyendo información sobre los equipos, los jugadores y los resultados.
4. Desarrollar las técnicas de Machine Learning para la recolección de datos históricos de los juegos de la liga de fútbol y predicción de resultados
5. Evaluar la precisión del modelo, comparando los resultados reales con los resultados arrojados por el modelo seleccionado.

H1:

El uso de modelos de Machine Learning basados en métricas ofensivas y defensivas históricas permite predecir los resultados de partidos de fútbol con una exactitud significativamente superior a la probabilidad aleatoria (33.33%) y a un clasificador ingenuo basado en la localía.

H0:

Los modelos de Machine Learning no presentan una mejora estadísticamente significativa respecto a la probabilidad aleatoria ni a un clasificador ingenuo.

La predicción de resultados en el mundo del deporte es una actividad relevante, ya que permite a los equipos planificar sus estrategias y tomar decisiones sobre la composición de los equipos y las tácticas de juego, asimismo los aficionados están interesados en conocer



las predicciones de los resultados, algunos como simple curiosidad y otros más para hacer apuestas.

Dentro del mundo del fútbol existen algunos torneos relevantes como, los juegos olímpicos, la Copa América, la Eurocopa y la más importante, la Copa Mundial, pero estos torneos tienen un formato muy rápido y no se juegan todos los años, por el contrario, las ligas de fútbol son torneos anuales y son de los eventos más importantes y continuos, por lo que la predicción de resultados para esta liga es de gran interés para la industria del deporte en general, así como para la comunidad Investigadora.

Como lo menciona Calderón Pérez-Lozao (Calderón Pérez-Lozao, 2017), “el análisis de los datos mediante técnicas de *Machine Learning*, puede ser algo muy interesante para ayudar a mejorar a los clubes pequeños y grandes, a tener en cuenta diferentes variables que antes no se utilizaban y poder mejorar su rendimiento.” Así entonces, se puede observar que el uso de *Machine Learning* para la predicción de resultados ha demostrado ser efectivo en otros torneos a lo largo del mundo. La disponibilidad de datos históricos sobre juegos de fútbol, así como datos importantes de los equipos, resultados y jugadores, hace posible el desarrollo de modelos para la predicción de resultados. Estos datos pueden ser recopilados y procesados usando técnicas de *Machine Learning*, para generar modelos de predicción precisos.

Esta investigación se distingue por su enfoque en el fútbol y la aplicación de técnicas de *Machine Learning*, la consideración de múltiples modelos y un enfoque riguroso en la evaluación del modelo.

Estas características la hacen única y aportan un valor diferencial en el campo de la predicción de resultados de fútbol.



Capítulo 2: Trabajo Relacionado

La predicción de resultados en el fútbol ha sido ampliamente abordada en la literatura reciente mediante el uso de técnicas de machine learning y modelos estadísticos, con el objetivo de mejorar la capacidad predictiva y, en algunos casos, desarrollar estrategias rentables frente a las casas de apuestas.

Steijns (2020) estudió la predicción de resultados de partidos en ligas europeas, considerando tres posibles salidas: victoria local, empate o victoria visitante. En su trabajo utilizó estadísticas de partido, así como información de equipos y jugadores tomada del videojuego FIFA. Para reducir y seleccionar variables aplicó correlación de Pearson y Análisis de Componentes Principales (PCA). Posteriormente evaluó modelos como Random Forest, XGBoost y Regresión Logística. Uno de los resultados más relevantes fue que los modelos contruidos únicamente con estadísticas de partido obtuvieron el mejor desempeño, con una precisión cercana al 56%. Esto muestra que agregar más información no siempre mejora el resultado del modelo, especialmente cuando las variables adicionales no aportan una señal clara para la predicción.

Viderström y Sjöberg (2024) analizaron el problema desde una perspectiva relacionada con las apuestas deportivas. Su estudio combinó modelos de Machine Learning con el criterio de Kelly, utilizado para la gestión del riesgo en apuestas. Para ello, trabajaron con datos históricos de la Premier League entre 1993 y 2023, y compararon distintos algoritmos, entre ellos Regresión Logística, Random Forest, XGBoost, SVM, Naive Bayes y redes neuronales. Aunque los modelos no superaron el 50% de precisión, los autores observaron que la utilidad económica de una estrategia no depende únicamente del porcentaje de acierto. Este punto es importante, ya que en el fútbol una predicción puede tener valor dependiendo del contexto, de la cuota disponible y del manejo del riesgo.

Kaunitz et al. (2017) propusieron un enfoque diferente, centrado en el comportamiento del mercado de apuestas más que en la predicción directa del marcador o del resultado. En su metodología utilizaron el promedio de cuotas de distintas casas de apuestas como una aproximación a la probabilidad real de cada evento. A partir de esta comparación, buscaron identificar situaciones donde existiera una diferencia entre la probabilidad estimada y la cuota ofrecida. Mediante simulaciones históricas y pruebas con



dinero real, reportaron una precisión de 47.2% y un retorno de inversión de 8.5%. Sin embargo, también señalaron limitaciones prácticas, como las restricciones que las casas de apuestas pueden imponer a usuarios con estrategias exitosas.

A partir de estos trabajos se observa que la predicción de resultados de fútbol sigue siendo un problema difícil, incluso cuando se utilizan modelos avanzados. Los porcentajes de acierto reportados suelen mantenerse por debajo del 60%, lo que confirma que el fútbol tiene un alto grado de incertidumbre y que existen factores difíciles de medir antes de cada partido. También se puede observar que incorporar más variables no garantiza necesariamente una mejora en el desempeño, por lo que resulta importante seleccionar información que realmente represente el estado reciente de los equipos.

Considerando estos antecedentes, en esta investigación se decidió trabajar con variables basadas en promedios móviles, ya que permiten representar el rendimiento reciente de los equipos antes de cada partido. También se optó por utilizar una validación temporal, con el fin de evitar que el modelo empleara información de partidos futuros durante el entrenamiento. Con esto se busca evaluar los modelos en una situación más cercana a la forma en que realmente se haría una predicción antes de que el partido ocurra.

Capítulo 3: Marco Teórico

Para el desarrollo de esta investigación fue necesario revisar conceptos relacionados con el fútbol como problema de predicción, así como con el uso de técnicas de Machine Learning aplicadas a datos históricos. La predicción de resultados deportivos no depende de un solo factor, por lo que se requiere comprender tanto la naturaleza del problema como las herramientas utilizadas para abordarlo.

En este capítulo se presentan los fundamentos teóricos que sustentan el trabajo. Primero se describe el concepto de Machine Learning y su relación con el aprendizaje supervisado. Posteriormente se explican los principales algoritmos utilizados en la investigación, entre ellos árboles de decisión, regresión logística, máquinas de vectores de soporte y XGBoost. Finalmente, se abordan conceptos necesarios para la metodología, como la validación cruzada, la validación en series temporales y el uso de promedios móviles.



3.1: Machine Learning (ML).

Actualmente, las interacciones entre las personas y los sistemas computacionales generan una cantidad masiva de información, llegando a construir lo que ahora conocemos como el big data. Como lo menciona Steven, A., Frank (Frank, 2019) , “Comprender los patrones es importante para los seres humanos, y el análisis de macrodatos puede ayudar a caracterizar y explicar estos patrones.” Así entonces la importancia de obtener patrones para poder predecir algún resultado, en este caso el marcador de un partido de fútbol.

Según Zhao, Yu-peng. (Zhao et al., 2022) El Machine Learning es una disciplina científica que se centra en el diseño de algoritmos para que las computadoras aprendan y desarrollen comportamientos basados en datos empíricos, es decir que a contrario de los paradigmas de programación anteriores el Machine Learning no sigue instrucciones detalladas para llegar a un resultado, si no que utilizan algoritmos para analizar patrones en los datos ingresados y generar modelos que pueden realizar predicciones o tomar decisiones basadas en nuevos datos que se generan sin la intervención del ser humano, esto es útil sobre todo cuando la cantidad de datos sobrepasa a las capacidades humanas y las relaciones son difíciles de percibir.

3.2: Aprendizaje Supervisado

El aprendizaje supervisado (*Supervised Learning*) es una de las subcategorías fundamentales del *Machine Learning*, cuyo objetivo central es enseñar a un algoritmo a través del análisis de un conjunto de datos de entrenamiento explícitamente etiquetados (Theobald, 2021.). En este enfoque, el modelo analiza combinaciones conocidas de datos de entrada (variables independientes o características, representadas como X) y resultados deseados (variables dependientes o etiquetas, representadas como y) para descifrar sus patrones subyacentes. Una vez entrenado, el algoritmo utiliza la función matemática aprendida para generar predicciones sobre datos nuevos o no vistos (Theobald, 2021).

Desde el punto de vista formal, el aprendizaje supervisado puede representarse como una función que relaciona un conjunto de variables de entrada con una salida esperada. En otras palabras, el modelo aprende una relación a partir de ejemplos previamente etiquetados, para después aplicar ese aprendizaje en nuevos datos.

$$y = f(X) \quad (1)$$



En esta expresión, X representa las variables de entrada y y corresponde a la salida que se desea predecir. El propósito no es únicamente que el modelo funcione bien con los datos usados durante el entrenamiento, sino que también pueda responder de manera adecuada ante datos que no ha visto antes. A esta capacidad se le conoce como generalización (Géron, 2022; Raschka et al., 2022).

Un aspecto fundamental en el aprendizaje supervisado es la capacidad de generalización, entendida como la habilidad del modelo para producir predicciones precisas sobre datos no vistos durante el entrenamiento. Un modelo que se ajusta en exceso a los datos de entrenamiento puede presentar sobreajuste (*overfitting*), mientras que un modelo demasiado simple puede incurrir en subajuste (*underfitting*), afectando negativamente su desempeño en escenarios reales (Géron, 2022; Murphy, 2012).

Los sistemas de aprendizaje supervisado se dividen en dos grandes categorías según la naturaleza del resultado a estimar: clasificación y regresión (Prosise, 2023).

La clasificación es una tarea del aprendizaje supervisado en la que el modelo busca asignar cada observación a una categoría definida previamente. En este tipo de problemas, la salida no es un valor numérico continuo, sino una clase. Por ejemplo, un modelo puede clasificar un correo como deseado o no deseado, una imagen dentro de una categoría específica o, en el caso de esta investigación, el resultado de un partido de fútbol.

Existen diferentes tipos de clasificación. La clasificación binaria se utiliza cuando solo hay dos posibles resultados; la clasificación multiclase se aplica cuando existen tres o más categorías excluyentes; y la clasificación multietiqueta permite que una misma observación pertenezca a más de una clase al mismo tiempo (Prosise, 2023; Géron, 2022; Murphy, 2012). Para este trabajo, el problema corresponde a una clasificación multiclase, ya que el resultado de un partido puede ser victoria local, empate o victoria visitante (Calderón Pérez-Lozao, 2017).

Entre los algoritmos que pueden aplicarse a este tipo de tareas se encuentran la Regresión Logística, los Árboles de Decisión, k-Vecinos Más Cercanos (k-NN), las Máquinas de Vectores de Soporte (SVM) y las Redes Neuronales (Theobald, 2021). En esta investigación se retoman algunos de estos enfoques con el propósito de comparar su desempeño en la predicción de resultados de fútbol.



Por otra parte, la regresión se utiliza cuando la variable que se desea predecir es numérica y continua, como el precio de una vivienda, la cantidad de visitas a un sitio web o alguna métrica deportiva cuantitativa, por ejemplo los goles esperados de un equipo (Prosise, 2023; Raschka et al., 2022; Ferreira et al., 2023). Aunque la regresión forma parte del aprendizaje supervisado, en esta investigación el enfoque principal es la clasificación, debido a que el objetivo consiste en estimar una de tres posibles categorías de resultado.

De esta manera, el aprendizaje supervisado permite construir modelos que aprenden a partir de datos históricos y posteriormente generan predicciones sobre nuevos casos. En el contexto de esta tesis, este enfoque se utiliza para analizar información previa de partidos y estimar el resultado más probable de encuentros posteriores.

3.3: Algoritmos de Machine Learning

Para el desarrollo de esta investigación se seleccionaron diferentes algoritmos de Machine Learning con el propósito de comparar su desempeño en la predicción de resultados de fútbol. Los modelos utilizados abordan el problema desde distintos enfoques: algunos trabajan con relaciones lineales, otros construyen reglas a partir de divisiones en los datos y otros combinan varios modelos para mejorar la predicción. En los siguientes apartados se describe de manera general el funcionamiento de cada algoritmo y su aplicación en tareas de clasificación.

3.3.1: Árboles de Decisión (Decision Trees)

Los árboles de decisión son algoritmos de aprendizaje supervisado que clasifican los datos mediante una serie de divisiones ordenadas. Su funcionamiento puede entenderse como una secuencia de preguntas aplicadas a las variables de entrada, donde cada respuesta conduce a una nueva división hasta llegar a una categoría final. Esta estructura facilita la interpretación del proceso de clasificación, ya que se organiza mediante un nodo inicial, nodos intermedios y hojas finales (Géron, 2022; Prosise, 2023; Theobald, 2021).

Durante el entrenamiento, el árbol busca separar los datos en grupos cada vez más homogéneos. Para ello, analiza las variables disponibles y selecciona el punto de corte que distingue mejor las clases. Esta separación se evalúa con criterios como el índice Gini o la entropía, los cuales indican qué tan mezcladas se encuentran las clases dentro de un nodo



(Géron, 2022; Theobald, 2021). Mientras menor sea la impureza, más clara será la separación obtenida.

En el análisis de resultados de fútbol, estos modelos pueden utilizarse para identificar patrones a partir de información histórica, por ejemplo, al relacionar el rendimiento reciente de un equipo con la probabilidad de obtener determinado resultado. Sin embargo, una limitación de los árboles individuales es que tienden a ajustarse demasiado a los datos de entrenamiento. Para reducir este problema se aplican restricciones como limitar la profundidad del árbol, establecer un número mínimo de muestras por nodo o trabajar con métodos de ensamble como Random Forest (Géron, 2022; Prosise, 2023).

$$G_i = 1 - \sum_{k=1}^n P_{i,k}^2 \quad (2)$$

El índice Gini se utiliza para medir la calidad de las divisiones realizadas por el árbol. Esta medida ayuda a identificar qué tan mezcladas se encuentran las clases dentro de un nodo. En la expresión anterior, $P_{i,k}$ representa la proporción de elementos de la clase k dentro del nodo i . Cuando el valor de Gini se acerca a cero, significa que el nodo contiene principalmente elementos de una sola clase, por lo que la separación obtenida es más clara.

3.3.2: Regresión Logística

Cuando realizamos una RL lo que pretendemos es estimar los parámetros de la ecuación ($\beta_0, \beta_1, \beta_2, \dots, \beta_k$) de la función que pretendemos evaluar:

$$Z = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_k X_k \quad (3)$$

Donde Z es el logaritmo neperiano ($\ln$) de la odds de padecer la enfermedad, el desenlace o el resultado que se está estudiando; β_0 es la ordenada en el origen de la función de regresión, $\beta_1, \beta_2, \dots, \beta_k$ representan los coeficientes de la pendiente de la recta y $X_1, X_2, \dots, X_k$ son las variables independientes o factores de riesgo. Si nuestros datos se ajustan de manera satisfactoria a este modelo, tendremos la suerte de poder explicar la relación entre las variables independientes y la respuesta de una manera muy sencilla. Los coeficientes β_i



expresan el logaritmo neperiano del odds ratio (OR) para cada factor de riesgo X_i . Por tanto el OR se estima a partir de la ecuación 4:

$$OR = \text{antilog}(\beta_i) = e^{\beta_i} \quad (4)$$

Una vez que hemos construido nuestro modelo de RL, debemos primero analizar los coeficientes de regresión (β_i) de cada variable independiente para obtener sus OR y luego confeccionar el valor predictivo de cada variable independiente o bien del modelo en su conjunto (Fiuza Pérez & Rodríguez Pérez, 2000).

3.3.3 Máquinas de Vectores de Soporte (SVM)

Las Máquinas de Vectores de Soporte (SVM, por sus siglas en inglés) constituyen un conjunto de algoritmos de aprendizaje automático altamente versátiles, capaces de abordar tareas de clasificación lineal y no lineal, regresión, así como detección de anomalías (Géron, 2022). Aunque su desarrollo se remonta a la década de 1990 dentro del campo de las ciencias de la computación, en la actualidad se emplean principalmente como técnicas de clasificación en diversos problemas del aprendizaje supervisado (Theobald, 2021).

El funcionamiento de las Máquinas de Vectores de Soporte se basa en encontrar una frontera que separe las clases dentro del conjunto de datos. Esta frontera recibe el nombre de hiperplano, y el modelo busca ubicarla de forma que exista la mayor distancia posible entre ella y los puntos más cercanos de cada clase. A este enfoque se le conoce como clasificación de margen máximo. Los datos que quedan más próximos al hiperplano son los que influyen directamente en su posición, por lo que se denominan vectores de soporte (Géron, 2022; Prosise, 2023).

En la práctica, los datos no siempre se pueden separar de manera perfecta. Por esta razón, las SVM incorporan el concepto de margen suave, el cual acepta ciertos errores de clasificación para evitar que el modelo se ajuste demasiado al conjunto de entrenamiento. Este comportamiento se regula mediante el parámetro C . Cuando C tiene valores bajos, el modelo tolera más errores y genera un margen más amplio; cuando C es alto, intenta clasificar correctamente más ejemplos del entrenamiento, aunque con mayor riesgo de sobreajuste (Theobald, 2021).



Una característica importante de las SVM es el uso de funciones kernel. Estas funciones se utilizan cuando las clases no pueden separarse mediante una frontera lineal simple. En lugar de transformar los datos de manera explícita, el kernel trick trabaja de forma implícita en un espacio de mayor dimensión, lo que ayuda a representar relaciones más complejas. Entre los kernels más comunes se encuentran el lineal, el polinómico, el sigmoide y el de función de base radial, conocido como RBF (Prosise, 2023).

En este tipo de modelos también es importante considerar el preprocesamiento. Debido a que las SVM dependen de distancias entre datos, las variables deben encontrarse en escalas comparables; por ello, normalmente se aplica normalización o estandarización antes del entrenamiento. Además, cuando el conjunto de datos crece demasiado, el entrenamiento puede volverse más costoso en términos computacionales.

En esta investigación, las SVM se utilizaron como una de las arquitecturas evaluadas para la predicción de resultados de fútbol. Debido a que este modelo es sensible a la escala de los datos, se aplicó estandarización previa a las variables de entrada. Además, sus hiperparámetros, como C, gamma y el tipo de kernel, fueron ajustados durante el proceso de búsqueda para evaluar su desempeño frente a los demás modelos considerados.

$$w \cdot x + b = 0 \quad (5)$$

Desde una perspectiva matemática, las SVM buscan encontrar un hiperplano óptimo que permita separar las clases en el espacio de características. La ecuación anterior representa dicho hiperplano, donde w es el vector normal y b el término de sesgo. En aplicaciones reales, se emplean variantes con margen suave que permiten cierta tolerancia a errores, controlada mediante el parámetro de regularización, lo cual favorece una mejor capacidad de generalización.

3.3.4 eXtreme Gradient Boosting (XGBoost)

El algoritmo XGBoost (eXtreme Gradient Boosting) es un método avanzado de aprendizaje automático supervisado que pertenece a la categoría de aprendizaje en conjunto (Ensemble Learning), específicamente basado en la técnica de potenciación del gradiente (Gradient Boosting) mediante árboles de decisión (Géron, 2022; Huyen, 2022). A diferencia



de otros enfoques como los bosques aleatorios, donde los árboles se generan de forma independiente, XGBoost construye los modelos de manera secuencial, de tal forma que cada nuevo árbol se entrena para corregir los errores residuales de los modelos anteriores, optimizando progresivamente una función de pérdida diferenciable (Huyen, 2022; Prosise, 2023; Raschka et al., 2022).

XGBoost incorpora mejoras sobre los métodos tradicionales de potenciación del gradiente, principalmente en la forma en que construye árboles de decisión de manera secuencial y corrige errores del modelo anterior. Esta característica lo hace útil en problemas donde existen relaciones no lineales entre las variables, como ocurre en muchos conjuntos de datos deportivos. Además, puede trabajar con una cantidad amplia de características, siempre que se realice una adecuada configuración de sus parámetros (Raschka et al., 2022).

En el análisis de resultados de fútbol y apuestas deportivas, distintos trabajos han utilizado modelos de Machine Learning para analizar datos históricos, estadísticas de juego e información contextual con el propósito de mejorar la predicción y apoyar la toma de decisiones. En una revisión sistemática reciente, Galekwa et al. (2025) señalan que técnicas como SVM, Random Forest y redes neuronales se han aplicado en el contexto del fútbol para analizar información histórica, datos del partido y variables en tiempo real. Aunque estos enfoques muestran potencial, los autores también destacan retos importantes relacionados con la calidad de los datos, la toma de decisiones en tiempo real y la incertidumbre propia de los eventos deportivos.

A pesar de sus ventajas, XGBoost también puede presentar problemas de sobreajuste cuando se entrena con datos ruidosos o cuando sus hiperparámetros no se ajustan correctamente. Por esta razón, es común controlar elementos como la profundidad de los árboles, la tasa de aprendizaje, el número de estimadores y el muestreo de observaciones o variables. En esta investigación, XGBoost se incluyó como uno de los modelos de ensamble evaluados, con el objetivo de comparar su desempeño frente a modelos lineales y otros métodos de clasificación.

$$F_m(x) = F_{m-1}(x) + \eta \gamma_m \quad (6)$$

XGBoost construye el modelo de forma aditiva, agregando en cada iteración un nuevo árbol que busca corregir los errores cometidos por el modelo anterior. En la expresión anterior, $F_m(x)$ representa el modelo actualizado, mientras que η es la tasa de aprendizaje



que regula la contribución del nuevo modelo. Este enfoque permite mejorar progresivamente la capacidad predictiva sin generar sobreajuste excesivo.

3.4: Validación Cruzada

La validación cruzada (*Cross validation*) es un método que consiste en evaluar y probar el rendimiento de un modelo de machine learning, con el fin de encontrar un mejor modelo rápidamente. Esta técnica ayuda a la comprensión y aplicación de este modelado predictivo, siendo fácil y sencilla de aplicar (Flores, s. f.)

La validación cruzada se convierte en un componente fundamental en el ciclo de entrenamiento y evaluación de modelos de Machine Learning. Su función primordial radica en estimar la capacidad de generalización de un modelo y disminuir cualquier sesgo que pueda surgir en la selección de datos. En el contexto de pronosticar resultados de partidos de fútbol, la validación cruzada se convierte en una herramienta esencial para medir la efectividad de los modelos y asegurar su solidez.

La validación cruzada permite evaluar la consistencia del rendimiento del modelo y detectar posibles problemas de sobreajuste. Es una técnica fundamental para garantizar la fiabilidad de los resultados y se utilizará en esta investigación para evaluar la capacidad de predicción de los modelos de Machine Learning.

En lugar de dividir los datos en solo dos fragmentos fijos (entrenamiento y prueba), maximiza la disponibilidad de datos para entrenamiento al dividir el conjunto de datos en varias combinaciones y probar cada combinación específica (Theobald, s. f.)

Fórmula para el Rendimiento Promedio (K – fold):

$$E = \frac{1}{K} \sum_{i=1}^K E_i \quad (7)$$

El rendimiento promedio estimado (E) se calcula como la media de las estimaciones de rendimiento individuales (E_i) obtenidas en cada pliegue: (Raschka, 2022)



Donde:

- E es el rendimiento promedio estimado del modelo.
- K es el número de pliegues utilizados.
- E_i es la métrica de rendimiento (ej. *accuracy* o *error*) obtenida en el i -ésimo pliegue de prueba.

3.4.1: Validación en Series Temporales (Time Series Split)

Aunque la validación cruzada tradicional, como K-Fold, se utiliza con frecuencia en problemas de Machine Learning, su aplicación directa no siempre es adecuada cuando los datos tienen un orden temporal. En el caso de los resultados deportivos, mezclar partidos de distintas fechas puede provocar que el modelo aprenda con información posterior para predecir eventos anteriores. Este problema se conoce como fuga de información temporal o look-ahead bias.

En la predicción de resultados de fútbol es importante respetar la secuencia cronológica de los datos. Esto implica entrenar el modelo con partidos anteriores y reservar los encuentros más recientes para la etapa de prueba. De esta forma, se evita utilizar información de un evento futuro para predecir uno pasado, y el proceso se acerca más a una situación real de predicción (Calderón Pérez-Lozao, 2017).

Para evaluar el comportamiento del modelo frente a datos futuros no vistos, puede utilizarse un esquema progresivo de entrenamiento y validación. Por ejemplo, en un escenario simplificado de cinco semanas, el modelo puede entrenarse con las primeras cuatro semanas y validarse con la quinta. Posteriormente, el bloque de entrenamiento puede ampliarse y la validación realizarse sobre un periodo posterior, manteniendo siempre el orden temporal de los datos (Huyen, 2022).

3.5 Promedios Móviles (Rolling Averages)

Los promedios móviles se utilizan en series temporales para resumir el comportamiento reciente de una variable. En lugar de tomar únicamente el valor de una observación aislada, se calcula el promedio de un conjunto de valores anteriores. Esto ayuda a disminuir variaciones puntuales y a observar mejor la tendencia reciente de los datos (Hyndman & Athanasopoulos, 2021; Box et al., 2015).



Formalmente, un promedio móvil de ventana k se define como:

$$MA_t = \frac{1}{k} \sum_{i=1}^k x_{t-i} \quad (8)$$

Donde MA_t representa el promedio móvil en el tiempo t , k es el tamaño de la ventana y x_{t-i} corresponde a los valores históricos considerados para el cálculo.

En el caso del fútbol, esta técnica ayuda a representar el rendimiento reciente de un equipo antes de un partido. Por ejemplo, pueden calcularse promedios de goles a favor, goles en contra, tiros, posesión u otras métricas relevantes, tomando como base los encuentros anteriores. De esta manera, el modelo no utiliza solo el dato de un partido, sino una medida resumida del comportamiento reciente del equipo.

En esta investigación, los promedios móviles se utilizaron para generar variables de entrada basadas en partidos anteriores. Esto fue importante porque las estadísticas del partido que se desea predecir no están disponibles antes de que ocurra. Por ello, las variables se construyeron únicamente con información previa, evitando el uso de datos futuros dentro del proceso de modelado.

Capítulo 4: Metodología

En este capítulo se explica el procedimiento seguido para desarrollar la investigación. Se describe cómo se obtuvieron los datos, cómo fueron preparados para el análisis y de qué manera se entrenaron y evaluaron los modelos de Machine Learning. El proceso incluye la recolección mediante web scraping, la limpieza de datos, la generación de nuevas variables, la definición del conjunto final, la validación temporal y la comparación de los modelos utilizados.

La Figura 1 muestra el flujo general de trabajo utilizado en la investigación. En ella se resumen las etapas principales, desde la extracción de información hasta la evaluación de los resultados obtenidos por los modelos.

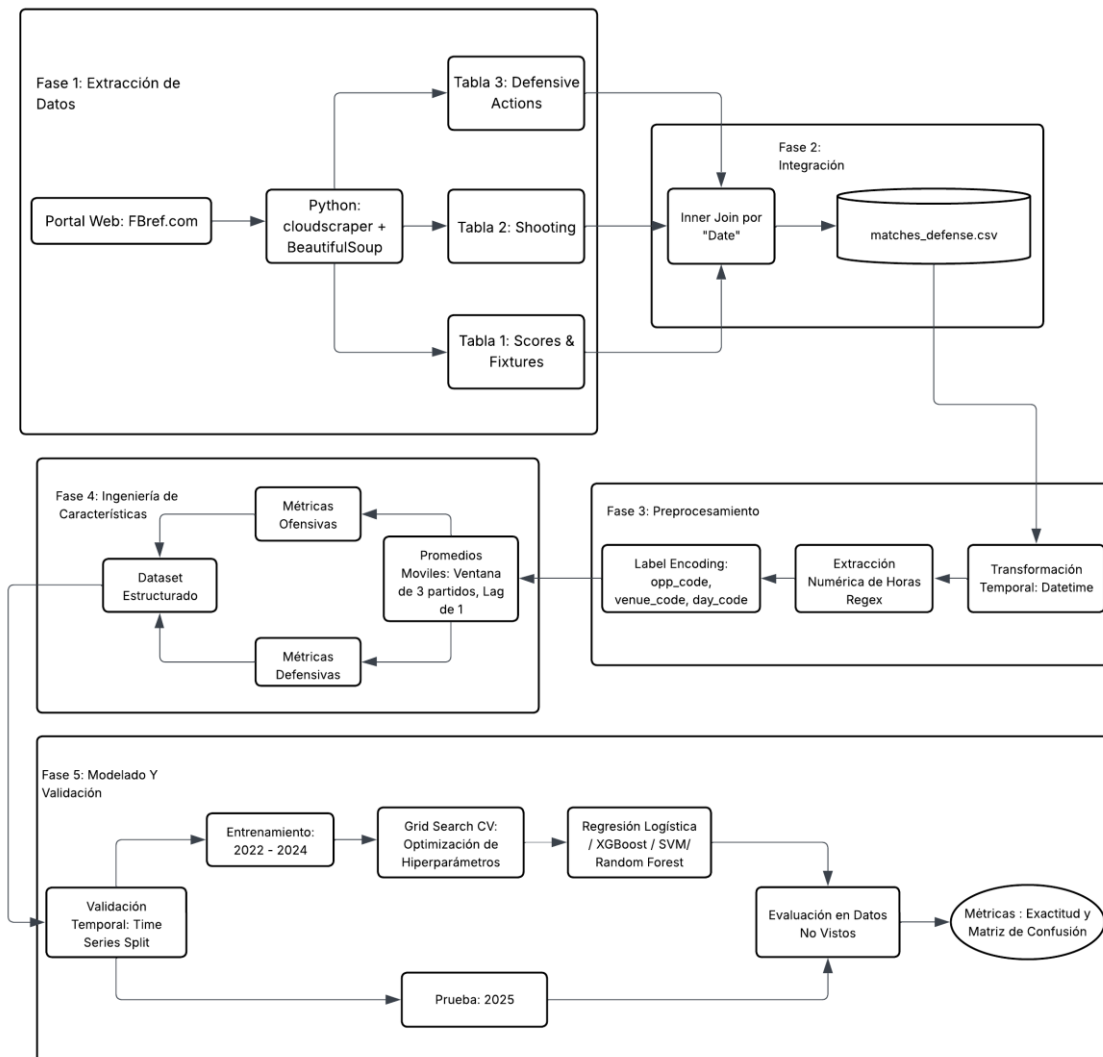


Figura 1 Flujo general del proceso de trabajo. Fuente: elaboración propia.

Como se observa en la Figura 1, el proceso se organizó en cinco fases. La primera fase correspondió a la extracción de datos. En esta etapa se obtuvo información del portal FBref.com mediante scripts desarrollados en Python, utilizando las librerías requests y BeautifulSoup. Con esto se reunieron estadísticas de partidos y variables relevantes para el análisis.

En la segunda fase, los datos extraídos se integraron en un archivo CSV, lo que facilitó su almacenamiento y posterior manejo. Después, en la fase de preprocesamiento, se realizó la limpieza del conjunto de datos, la codificación de variables categóricas y la transformación de algunos campos necesarios para el entrenamiento de los modelos.



La cuarta fase correspondió a la ingeniería de características. En esta etapa se generaron variables derivadas, principalmente promedios móviles de métricas ofensivas y defensivas. Estas variables se utilizaron para representar el rendimiento reciente de los equipos antes de cada partido.

Finalmente, en la quinta fase se llevó a cabo el modelado y la validación. Se entrenaron modelos de Regresión Logística, SVM, XGBoost y Random Forest, utilizando una estrategia basada en series temporales. También se realizó la optimización de hiperparámetros y la evaluación con datos no vistos de la temporada 2025, con el fin de comparar el desempeño de los modelos.

4.1: Recolección de Datos

Para esta investigación se recopilamos datos históricos de partidos de fútbol a partir del sitio FBref.com. La información utilizada incluyó resultados de partidos anteriores, estadísticas de rendimiento de los equipos, goles a favor, goles en contra, tiros, tiros a puerta y otras métricas necesarias para el análisis. Estos datos fueron la base para construir el conjunto utilizado posteriormente en el entrenamiento de los modelos.

La recolección no se realizó de forma manual, ya que el volumen de información requerido era grande y se encontraba distribuido en distintas tablas dentro del sitio. Por esta razón, se recurrió a técnicas de web scraping, utilizando scripts desarrollados en Python para extraer la información de manera automatizada.

Para este proceso se empleó la librería BeautifulSoup, junto con otras herramientas de Python, con el fin de leer el contenido HTML de las páginas y localizar las tablas de interés. Durante esta etapa fue necesario revisar la estructura del sitio, identificar encabezados, filas y columnas útiles, y eliminar elementos que no correspondían a datos de partidos.

Uno de los problemas más difíciles que se encontró, fue que FBref.com, presenta cierta resistencia al web scraping, lo que obligó a modificar el código de extracción en varias ocasiones, por ejemplo, el tiempo de espera para poder ejecutar el barrido de extracción de datos necesitaba cambiar de 5 a 10 segundos, en ocasiones hasta 15 segundos, algunas otras veces el acceso por https era imposible por lo que en ocasiones se cambiaba a http. Esto ocasiono que los tiempos de espera en algunos casos fuera muy largo. Por lo que se



contempló la posibilidad de adquirir una API comercial que facilitara el acceso a los datos, sin embargo, la combinación de recursos técnicos disponibles y la persistencia permitió resolver el problema sin costo adicional. El volumen total de datos recopilados resultó considerablemente mayor al esperado, por lo que se realizó un proceso de filtrado y selección para identificar las variables realmente útiles para el análisis.

En resumen, la etapa de web scraping, desempeñó un papel esencial en la investigación al proporcionar un conjunto de datos confiable y precisos para alimentar los algoritmos de aprendizaje automático y su importancia no debe subestimarse.

4.1.1 Arquitectura del proceso de web scraping

Con el objetivo de recolectar datos históricos de partidos de fútbol de manera automatizada, se diseñó una arquitectura de extracción basada en técnicas de web scraping. Este enfoque permitió obtener información estructurada a partir de fuentes web especializadas, específicamente del sitio FBref, reconocido por su amplia cobertura estadística del fútbol profesional.

El proceso fue implementado utilizando el lenguaje de programación Python, apoyado en librerías especializadas para la obtención y procesamiento de contenido web. En particular, se utilizó la librería *requests* para realizar solicitudes HTTP y recuperar el contenido de las páginas, y *BeautifulSoup* para el análisis y navegación del documento HTML.

La arquitectura del proceso de scraping se estructuró en tres etapas principales:

Obtención de páginas web (Data Acquisition):

Se realizó el acceso automatizado a las páginas correspondientes a las temporadas de la liga inglesa de fútbol. A partir de URLs específicas, el sistema recuperó el contenido HTML de cada página, simulando la navegación de un usuario.

Análisis y extracción de información (Parsing):

Una vez obtenido el contenido HTML, se procedió a identificar las estructuras relevantes dentro del documento, principalmente tablas que contienen estadísticas de partidos y equipos. Con BeautifulSoup se buscaron dentro del HTML las tablas y etiquetas necesarias para obtener la información de cada partido. Para ello se revisaron elementos como `<table>`,



`<tr>`, `<th>` y `<td>`, ya que en ellos se encontraban tanto los encabezados como los valores de los registros.

Transformación y almacenamiento de datos (Structuring):

Después de extraer la información, los datos se organizaron en DataFrames mediante la librería Pandas. Esto permitió revisar las tablas con mayor facilidad, corregir algunos detalles de estructura y preparar los archivos para las siguientes etapas del trabajo, como la limpieza, el preprocesamiento y la generación de variables.

Este procedimiento ayudó a automatizar la recolección de datos de distintas temporadas y redujo el trabajo manual. También permitió repetir el proceso cuando fue necesario, aunque siempre fue importante revisar la estructura del sitio, ya que algunos cambios en las tablas podían afectar la extracción de información.

4.1.2 Implementación del scraper

El proceso de web scraping se implementó mediante scripts en Python, trabajados en notebooks de Jupyter. Esta forma de trabajo facilitó probar el código por partes, revisar los resultados obtenidos y hacer ajustes durante la extracción de datos.

El scraper se diseñó para recorrer distintas temporadas de la liga inglesa de fútbol. Para ello, se utilizaron direcciones web específicas de FBref.com, desde las cuales se obtuvo la información estadística disponible por equipo y por partido. Este procedimiento evitó capturar los datos de forma manual y permitió reunir la información necesaria para construir el conjunto de datos.

Después de obtener cada página, se revisaron las tablas que contenían las métricas requeridas para el análisis. Entre los datos extraídos se consideraron goles a favor, goles en contra, tiros totales, tiros a puerta, intercepciones, entradas ganadas y métricas avanzadas como goles esperados (xG) y goles esperados en contra (xGA).

Durante la extracción se encontraron algunos problemas relacionados con la estructura de las tablas. En particular, algunas páginas incluían encabezados repetidos dentro del cuerpo de los datos, por lo que fue necesario agregar filtros para eliminar esas filas y conservar únicamente los registros correspondientes a partidos.

También se realizó una revisión inicial para identificar valores nulos o inconsistentes, los cuales fueron tratados en etapas posteriores del preprocesamiento. Este paso fue



fundamental para garantizar la calidad de los datos antes de su integración en el modelo de Machine Learning.

Finalmente, los datos extraídos fueron almacenados en estructuras tipo DataFrame utilizando la librería Pandas, y posteriormente exportados a archivos en formato CSV para su uso en las fases de limpieza, transformación e ingeniería de características.

Este enfoque permitió construir un dataset robusto, escalable y alineado con los objetivos de la investigación, asegurando que los modelos fueran entrenados con información relevante y correctamente estructurada.

4.2: Preprocesamiento de Datos

El preprocesamiento de los datos tuvo como objetivo garantizar la integridad, consistencia y compatibilidad del dataset para su uso en los modelos de Machine Learning. Este proceso se llevó a cabo mediante las siguientes etapas:

Limpieza y manejo de datos faltantes:

Se realizó una depuración de los datos para eliminar registros con información incompleta. Debido a la alta calidad de la fuente (FBref), la cantidad de valores faltantes fue mínima, por lo que se optó por la eliminación de filas incompletas para evitar introducir ruido en el cálculo de las métricas derivadas, particularmente en los promedios móviles (rolling averages).

Codificación de variables categóricas:

Las variables cualitativas fueron transformadas a representaciones numéricas mediante codificación por categorías.

venue_code: Representa la condición de local o visitante.

opp_code: Identificador numérico del equipo rival.

day_code: Día del partido codificado.

Adicionalmente, se incorporó la variable:

hour: Hora del encuentro, tratada como variable numérica.



Este proceso se realizó utilizando las capacidades de codificación categórica de la librería pandas, permitiendo mantener la identidad de las categorías sin incrementar significativamente la dimensionalidad del dataset.

Transformación y escalamiento de variables:

El escalamiento de características se aplicó únicamente a los modelos sensibles a la magnitud de los datos, específicamente Regresión Logística y Máquinas de Vectores de Soporte (SVM), utilizando técnicas de normalización estándar.

Para los modelos de ensamble (Random Forest y XGBoost), no se realizó escalamiento, ya que estos algoritmos no dependen de la escala de las variables.

Es importante destacar que el escalamiento se realizó exclusivamente sobre el conjunto de entrenamiento, aplicándose posteriormente al conjunto de prueba, con el fin de evitar fuga de información (data leakage).

Análisis de la variable objetivo:

Se analizó la distribución de la variable objetivo (derrota, empate y victoria), identificando un sesgo natural hacia la victoria del equipo local. Este análisis permitió interpretar adecuadamente las métricas de desempeño del modelo, sin necesidad de aplicar técnicas de balanceo artificial de clases.

4.3: Ingeniería de Características (Feature Engineering):

Uno de los desafíos principales en la predicción deportiva es que las estadísticas de un partido (goles, posesión, tiros, entre otros) solo se conocen una vez finalizado el evento. Por lo tanto, no pueden utilizarse como variables predictoras para ese mismo encuentro sin incurrir en un problema de causalidad o fuga de información (data leakage).

Para resolver este problema, se generaron variables de contexto histórico mediante el uso de promedios móviles (Rolling Averages). Se definió una ventana deslizante de 3 partidos, acompañada de un desfase temporal (lag) de una jornada, implementado mediante una operación de desplazamiento (shift).

La definición del tamaño de la ventana deslizante no fue arbitraria. Inicialmente se utilizó una ventana de 5 partidos, sin embargo, a sugerencia del director de tesis se redujo a 3, considerando que podría capturar mejor el rendimiento reciente. Con el fin de validar empíricamente esta decisión, se realizaron pruebas adicionales con ventanas de 2, 6, 10 y 15 partidos, modificando manualmente el parámetro en cada ejecución y comparando los resultados obtenidos. El valor de 3 partidos fue el que produjo el mejor desempeño en todos los modelos evaluados, por lo que se mantuvo como configuración final.

De esta forma, para predecir el rendimiento de un equipo en la jornada T, el modelo únicamente utiliza información correspondiente a las jornadas T-1, T-2 y T-3, garantizando que no se empleen datos futuros en el proceso de entrenamiento.

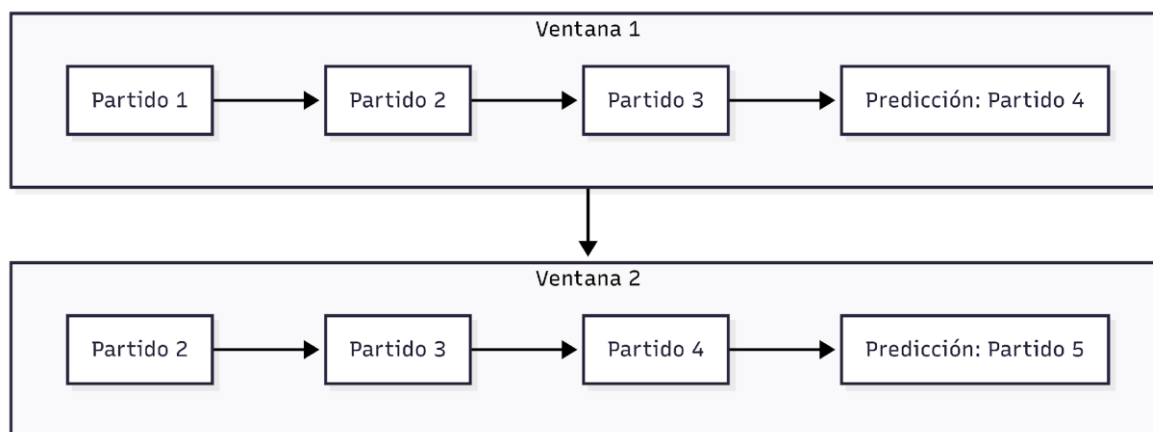


Figura 2 Representación del proceso de construcción de datos utilizando una ventana deslizante (sliding window) de tamaño 3.

4.3.1: Generación de variables ofensivas y defensivas

Las variables predictoras fueron construidas a partir de métricas de desempeño ofensivo y defensivo, calculadas como promedios móviles sobre la ventana definida.

En lugar de seleccionar manualmente un subconjunto específico de variables, se adoptó un enfoque general basado en la generación sistemática de características. Todas las métricas derivadas del dataset original fueron transformadas en promedios móviles, dando lugar a nuevas variables identificadas por el sufijo "_rolling".



Este procedimiento permitió incorporar de manera consistente múltiples indicadores de rendimiento reciente, incluyendo métricas ofensivas (como goles, tiros y precisión de disparo) y métricas defensivas (como goles en contra, intercepciones y entradas ganadas).

Algunas de las variables generadas fueron:

tklw_rolling (Tackles Won): promedio de entradas ganadas en los partidos anteriores.

int_rolling (Interceptions): promedio de intercepciones realizadas por el equipo.

xga_rolling (Expected Goals Against): promedio de goles esperados en contra, utilizado para representar la calidad de las oportunidades concedidas al rival.

Estas variables se mencionan como ejemplo, ya que el modelo no trabajó únicamente con ellas. En general, se utilizaron las características disponibles que fueron generadas mediante promedios móviles y que se identificaron con el sufijo "_rolling".

Con estas variables se buscó representar el rendimiento reciente de los equipos antes de cada partido. Esto ayudó a que el modelo no dependiera solo de un dato aislado, sino de un resumen del comportamiento mostrado en encuentros anteriores.

4.4: Base de Datos Resultante

Tras el proceso de limpieza y generación de características, la base de datos resultante se estructuró en un formato tabular listo para el entrenamiento de los modelos. El conjunto final consta de vectores de características que resumen el rendimiento reciente de cada equipo.

A diferencia de una selección manual de variables, el conjunto de características fue definido mediante un criterio programático basado en la estructura del dataset. En particular, se seleccionaron todas aquellas variables generadas mediante ingeniería de características que corresponden a promedios móviles, identificadas por el sufijo "_rolling". Este enfoque permite incluir de manera automática todas las métricas históricas relevantes sin introducir sesgos en la selección.

Adicionalmente, se incorporaron variables categóricas transformadas mediante codificación numérica, con el objetivo de representar información contextual del partido.

A continuación, se describen las variables utilizadas, categorizadas por su naturaleza:

4.4.1: Variables predictoras (Features)



A. Métricas de rendimiento (Promedios móviles):

Para representar el rendimiento reciente de los equipos se utilizaron variables generadas con una ventana móvil de 3 partidos. Estas variables se identifican en el conjunto de datos con el sufijo "_rolling". Algunas de las utilizadas fueron:

gf_rolling: Promedio de goles a favor en partidos anteriores.

sh_rolling: Promedio de tiros totales realizados.

sot_rolling: promedio de tiros al arco o tiros a puerta.

dist_rolling: Distancia promedio de los tiros. Indica la selección de tiro.

pk_rolling / fk_rolling: Penales y tiros libres ejecutados.

ga_rolling: Goles en contra. Mide la vulnerabilidad defensiva reciente.

tklw_rolling: Entradas ganadas. Representa la recuperación del balón.

int_rolling: Intercepciones. Indica la capacidad de cortar juego rival.

xga_rolling: Goles esperados en contra. Mide la calidad de las ocasiones concedidas.

Cabe destacar que, además de estas variables, el modelo considera automáticamente cualquier otra métrica generada bajo el esquema de promedios móviles disponible en el dataset.

B. Variables contextuales (codificadas):

venue_code: Indica si el equipo juega como local o visitante.

opp_code: Identificador numérico del equipo rival.

day_code: Día del partido codificado.

hour: Hora del encuentro.

Estas variables permiten incorporar información contextual que no está contenida directamente en las métricas de rendimiento.

4.4.2 Variable Objetivo (Target)

La variable objetivo utilizada en el modelo corresponde al resultado del partido. Para trabajar con los algoritmos de clasificación, esta variable fue codificada en tres clases numéricas:



Clase 0: Derrota

Clase 1: Empate

Clase 2: Victoria

Esta codificación permitió representar los tres posibles resultados del partido dentro del modelo de clasificación multiclase.

4.5: Validación temporal de los datos (Time Series Split)

En este trabajo no se utilizó una división aleatoria de los datos, ya que los partidos tienen un orden cronológico. Si se mezclaran partidos de distintas fechas, el modelo podría entrenarse con información posterior para predecir partidos anteriores, lo que generaría fuga de información o *data leakage*.

Para evitar este problema, los datos se dividieron respetando la fecha de los partidos. El conjunto de entrenamiento quedó formado por los registros históricos desde la temporada 2022 hasta el 31 de diciembre de 2024, mientras que el conjunto de prueba se integró con partidos disputados a partir del 1 de enero de 2025.

Conjunto de entrenamiento (Train): datos históricos desde la temporada 2022 hasta el 31 de diciembre de 2024.

Conjunto de prueba (Test): partidos disputados a partir del 1 de enero de 2025.

Esta separación permite evaluar el modelo de forma parecida a una predicción real, donde solo se dispone de información ocurrida antes del partido que se desea estimar.

Además de esta división principal, durante el entrenamiento y la búsqueda de hiperparámetros se utilizó `TimeSeriesSplit`. Este método genera particiones progresivas dentro del conjunto de entrenamiento: primero utiliza un bloque inicial de datos para entrenar y luego valida con un bloque posterior. En cada iteración, el bloque de entrenamiento crece y la validación se realiza con datos posteriores.

Con esta estrategia se buscó evaluar el comportamiento del modelo sin mezclar información futura dentro del proceso de entrenamiento. De esta manera, la validación temporal se aplicó tanto en la separación general de entrenamiento y prueba como en la optimización interna de los modelos.

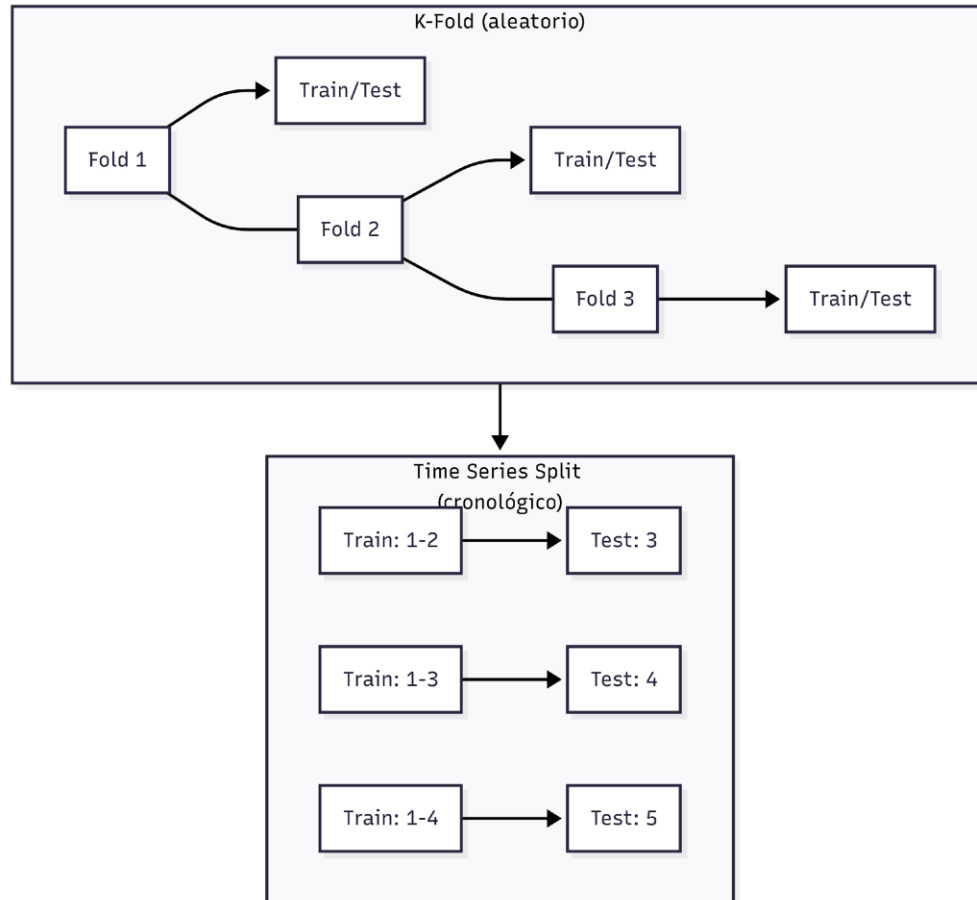


Figura 3 Figura 3. Comparación entre K-Fold y Time Series Split. Fuente: elaboración propia.

4.6: Selección y Entrenamiento de Modelos

Después de preparar la base de datos y definir la validación temporal, se entrenaron cuatro modelos de Machine Learning: Regresión Logística, Random Forest, XGBoost y Máquinas de Vectores de Soporte (SVM).

La selección de estos modelos responde a la necesidad de comparar diferentes enfoques de aprendizaje: modelos lineales, modelos basados en distancias y modelos de ensamble.

El conjunto de entrada para los modelos estuvo conformado por todas las variables generadas mediante promedios móviles (aquellas con sufijo "_rolling"), así como por variables categóricas codificadas (venue_code, opp_code, day_code y hour), mientras que la variable objetivo fue representada en formato multiclase {0,1,2}.



Configuración del entorno experimental

El desarrollo se realizó en Python 3.13.11, utilizando las bibliotecas Scikit-Learn y XGBoost.

El proceso de entrenamiento siguió las siguientes etapas:

- División temporal de los datos (train/test)
- Selección del conjunto de características
- Escalamiento de variables (cuando aplica)
- Entrenamiento de modelos base
- Optimización de hiperparámetros
- Evaluación en datos no vistos

Para la optimización de hiperparámetros se utilizó el método RandomizedSearchCV, en conjunto con una validación de series temporales (TimeSeriesSplit), garantizando que el modelo no tuviera acceso a información futura durante el entrenamiento.

Regresión Logística

La Regresión Logística es un modelo lineal que estima probabilidades mediante una función logística.

Dado que este modelo es sensible a la escala de los datos, se aplicó StandardScaler, ajustado sobre el conjunto de entrenamiento y posteriormente aplicado al conjunto de prueba.

Los hiperparámetros evaluados fueron:

Tabla 4.1: Descripción de hiperparámetros Regresión Logística.

C	Parámetro de regularización
solver	Algoritmo de optimización
penalty	Tipo de regularización

Espacio de búsqueda:

Tabla 4.2: Espacios de búsqueda Regresión Logística.

C:	[0.01, 0.1, 1, 10, 100]
solver:	['lbfgs', 'newton-cg', 'saga']
penalty:	['l2']



Random Forest

Random Forest es un modelo de ensamble basado en múltiples árboles de decisión, que reduce la varianza mediante agregación.

Los hiperparámetros evaluados fueron:

Tabla 4.3: Descripción de hiperparámetros Random Forest

n_estimators	número de árboles
max_depth	profundidad máxima del árbol
min_samples_split	mínimo de muestras para dividir
min_samples_leaf	mínimo de muestras en hojas
max_features	número de variables consideradas por división

Espacio de búsqueda:

Tabla 4.4: Espacios de búsqueda Random Forest

n_estimators	[100, 200, 300, 500]
max_depth	[None, 10, 20, 30]
min_samples_split	[2, 5, 10, 15]
min_samples_leaf	[1, 2, 4]
max_features	['sqrt', 'log2']

XGBoost

XGBoost es un algoritmo de gradient boosting que construye árboles de forma secuencial optimizando los errores residuales.

Los hiperparámetros evaluados fueron:

Tabla 4.5: Descripción de hiperparámetros XGBoost

n_estimators	número de árboles
learning_rate	tasa de aprendizaje
max_depth	profundidad del árbol
subsample	proporción de datos utilizada
colsample_bytree	proporción de variables utilizadas

Espacio de búsqueda:

Tabla 4.6: Espacios de búsqueda XGBoost

n_estimators	[100, 200, 300]
learning_rate	[0.01, 0.05, 0.1, 0.2]
max_depth	[3, 5, 7, 10]
subsample	[0.7, 0.8, 1.0]



colsample_bytree	[0.7, 0.8, 1.0]
------------------	-----------------

Support Vector Machines (SVM)

SVM es un modelo que busca encontrar el hiperplano óptimo que maximiza la separación entre clases.

Se utilizó escalamiento mediante StandardScaler, ajustado sobre el conjunto de entrenamiento, debido a la sensibilidad del modelo a la escala de los datos.

Se evaluaron diferentes kernels para analizar distintas formas de separación de clases.

Hiperparámetros evaluados:

Tabla 4.7: Descripción de hiperparámetros SVM

C	penalización del error
gamma	influencia de cada punto
kernel	tipo de función de transformación

Espacio de búsqueda:

Tabla 4.8: Espacios de búsqueda SVM

C	[0.1, 1, 10, 100]
gamma	['scale', 'auto', 0.01, 0.1, 1]
kernel	['rbf', 'poly', 'sigmoid']

Proceso de optimización

Para cada modelo se utilizó RandomizedSearchCV, el cual permite explorar de manera eficiente el espacio de hiperparámetros mediante muestreo aleatorio.

La búsqueda de hiperparámetros se realizó utilizando TimeSeriesSplit, de manera que las divisiones de entrenamiento y validación conservaran el orden temporal de los partidos. Con esto se evitó que el modelo utilizara información posterior dentro del proceso de ajuste.

Selección del modelo final

Para seleccionar el modelo final se tomó como referencia principal la exactitud (accuracy) obtenida en el conjunto de prueba de la temporada 2025. También se revisaron otras métricas, como F1-score, precision y recall, con el propósito de observar mejor el comportamiento de cada modelo.



Además, se analizaron las matrices de confusión y la precisión ponderada. Esto permitió identificar en qué clases se presentaban más errores, especialmente en el caso del empate, que fue la categoría más difícil de predecir.

Capítulo 5: Análisis y Discusión de Resultados

En este capítulo se muestran los resultados obtenidos después de entrenar y ajustar los modelos de Machine Learning utilizados en la investigación. También se comparan los algoritmos evaluados, se revisa el efecto de las variables generadas mediante ingeniería de características y se analiza el comportamiento de los modelos con datos de la temporada 2025 de la Liga Inglesa de Fútbol.

5.1 Proceso de Evaluación de los Modelos

Para evaluar los modelos se respetó el orden temporal de los partidos. A diferencia de una partición aleatoria, en este trabajo los datos de entrenamiento incluyeron registros históricos hasta finales de 2024, mientras que los partidos de 2025 se reservaron para la prueba final.

Para cada algoritmo se realizó una búsqueda de hiperparámetros mediante RandomizedSearchCV, utilizando TimeSeriesSplit durante el proceso de validación. De esta manera, las configuraciones evaluadas se probaron sin mezclar información futura dentro del entrenamiento.

El uso de RandomizedSearchCV permitió explorar de manera eficiente espacios de hiperparámetros amplios sin el alto costo computacional asociado a una búsqueda exhaustiva.

5.2 Comparativa de Desempeño y efecto de la Optimización

El rendimiento de los modelos se evaluó inicialmente en sus configuraciones por defecto (Base) y, posteriormente, tras un proceso de ajuste de hiperparámetros mediante RandomizedSearchCV. Esta comparativa es fundamental para identificar la capacidad de mejora de cada arquitectura ante datos de series temporales.

La Tabla a continuación resume los resultados obtenidos en el conjunto de prueba (Temporada 2025):



Tabla 5.1: Comparativa de métricas por modelo (Datos 2025)

Modelo	Exactitud (Base)	Exactitud (Optimizado)	Mejora Absoluta
Regresión Logística	0.4897	0.4960	+0.63%
XGBoost	0.4700	0.4929	+2.29%
SVM (Kernel RBF)	0.4500	0.4868	+3.68%
Random Forest	0.4865	0.4865	0.00%

Como se observa en la comparativa de la Figura 4, la Regresión Logística no solo presenta el desempeño más alto, sino que también muestra una gran estabilidad desde su configuración base. Por otro lado, modelos más complejos como SVM y XGBoost demostraron una mayor sensibilidad a los hiperparámetros, logrando incrementos significativos tras la optimización. El hecho de que el modelo de ensamble Random Forest no presentara cambios sugiere que sus parámetros por defecto ya capturaban la varianza máxima permitida por el ruido inherente a los datos de la temporada 2025.

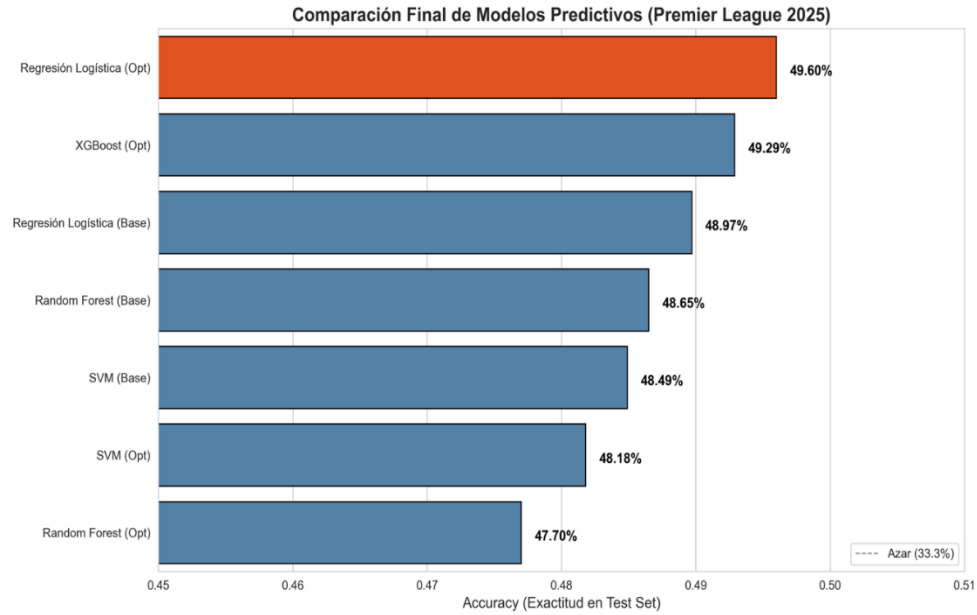


Figura 4 Comparación Final de Modelos Predictivos (Premier League 2025)

Como se observa en la **Figura 5**, el gráfico de radar revela una especialización distinta entre arquitecturas. Mientras la Regresión Logística maximiza el área de exactitud global, el modelo **SVM (RBF)** muestra un sesgo hacia la precisión en victorias. Esto sugiere que el kernel RBF crea fronteras de decisión más estrictas, lo que lo convierte en un modelo útil para escenarios donde el costo de un falso positivo (predecir una victoria que no ocurre) es elevado, a pesar de sacrificar la captura de algunos casos positivos (menor recall).

Comparativa de Perfil de Modelos

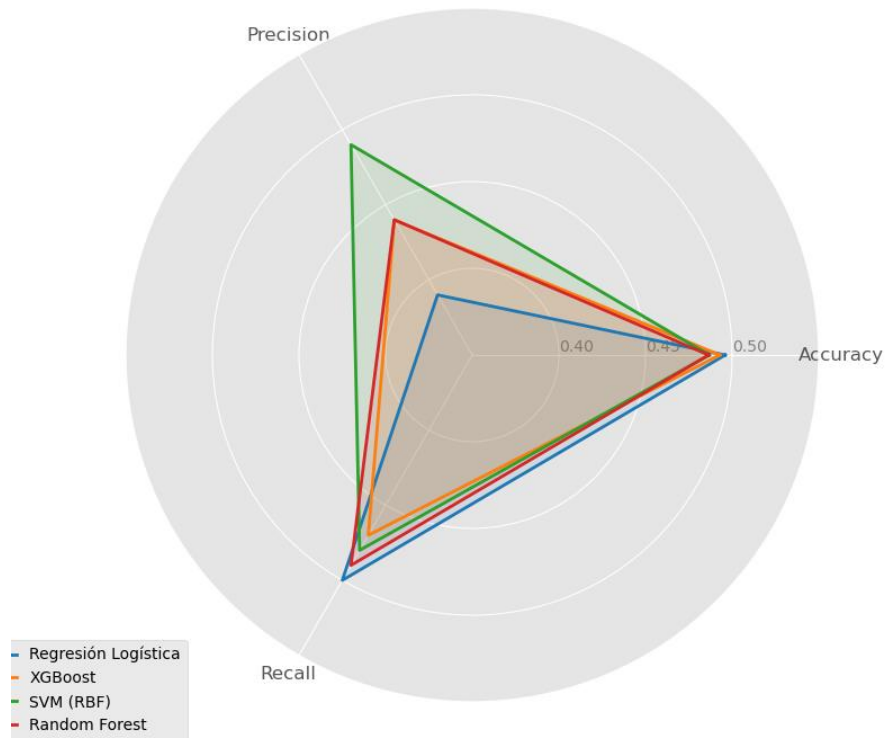


Figura 5 Comparativa de perfil de modelos

El mapa de calor de la Figura 6 permite identificar que la mayor confusión de los modelos de ensamble (RF y XGBoost) ocurre en la transición entre el empate y la derrota. La similitud en los perfiles de recall entre estos dos modelos confirma que, al compartir una lógica basada en árboles de decisión, ambos algoritmos encuentran las mismas limitaciones estructurales en la distribución de los datos, particularmente en partidos con bajo volumen de tiros al arco.

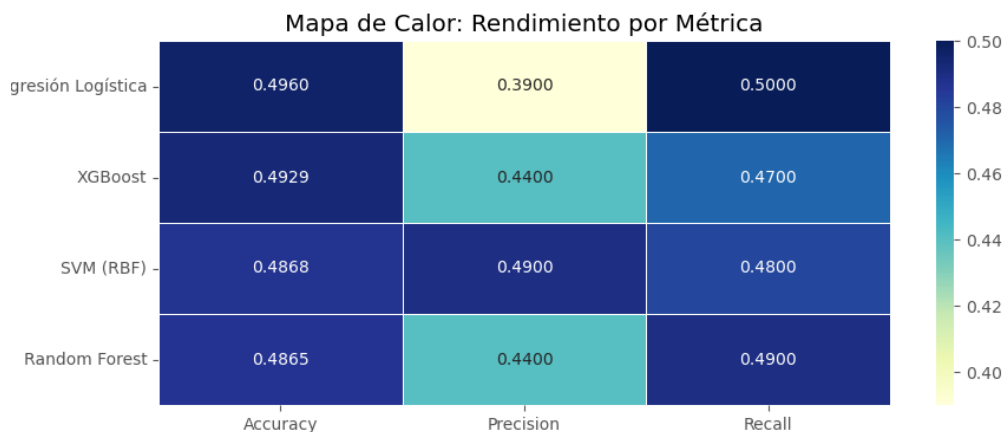


Figura 6 Mapa de calor: Rendimiento por Métrica

Si bien la exactitud (accuracy) permite obtener una visión general del desempeño de los modelos, esta métrica puede resultar insuficiente en problemas de clasificación multiclase, especialmente cuando existe una distribución compleja entre clases.

Con el objetivo de realizar una evaluación más robusta, se incorporaron métricas adicionales como precisión (precision), sensibilidad (recall) y F1-score en su versión macro, las cuales permiten analizar el comportamiento del modelo de manera equilibrada entre todas las clases.

Tabla 5.2: Resultados obtenidos para cada modelo

Modelo	Accuracy	Precision (macro)	Recall (macro)	F1-score (macro)
Regresión Logística	0.496	0.3721	0.345	0.3698
XGBoost	0.4929	0.385	0.36	0.3862
SVM (Kernel RBF)	0.4868	0.36	0.33	0.358
Random Forest	0.4865	0.365	0.335	0.3625

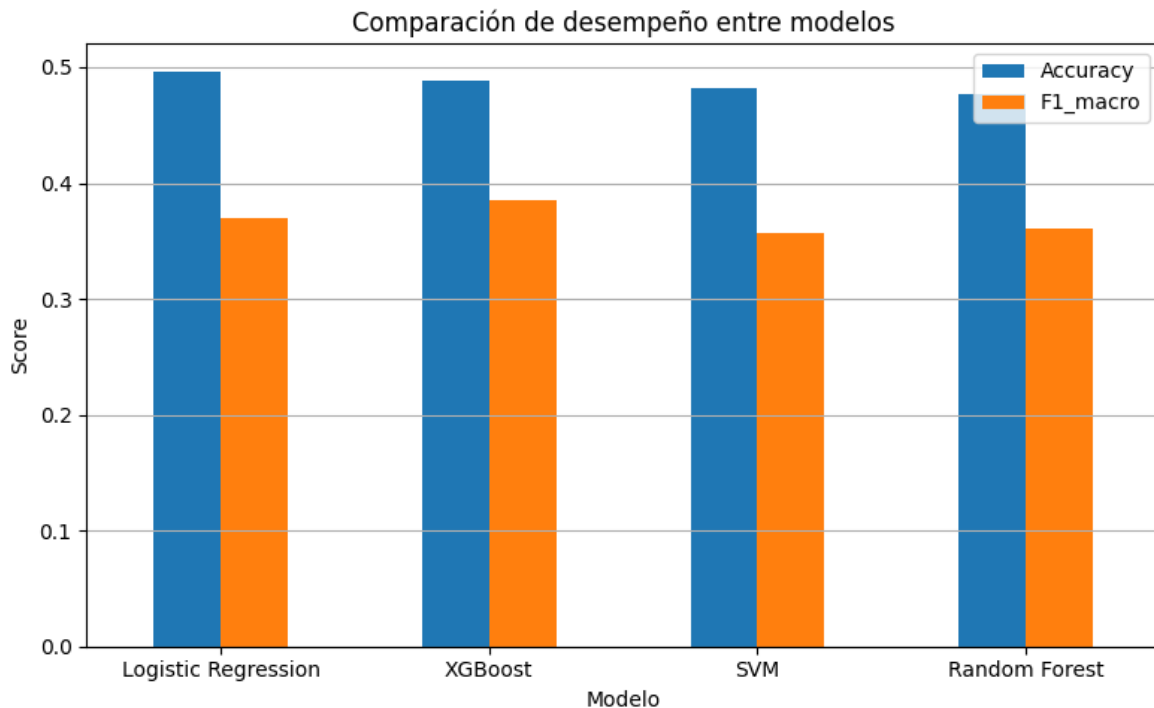


Figura 7 Comparación entre exactitud y F1-score de los modelos evaluados.

En la Figura 7 se observa que los modelos tienen valores de exactitud cercanos al 50%. Sin embargo, el F1-score es menor, con valores aproximados entre 0.35 y 0.39.

Esta diferencia muestra que los modelos no clasifican con la misma eficacia todas las clases. Aunque la exactitud general parece aceptable para el problema planteado, el F1-score permite ver que existen errores importantes en algunas categorías.

El caso más claro se presenta en la clase empate, donde los modelos tuvieron mayor dificultad para identificar correctamente los partidos. Esto indica que el empate no se separa con facilidad a partir de las variables utilizadas.

Por esta razón, la exactitud por sí sola no describe completamente el desempeño del modelo. Al revisar el F1-score y las métricas por clase, se obtiene una interpretación más completa de los resultados.

Este fenómeno es consistente con el análisis por clase presentado en la sección 4.4, donde se evidencia que el empate constituye una clase difícil de modelar.

En este sentido, el F1-score se posiciona como una métrica más adecuada para evaluar el desempeño real del sistema, al considerar tanto la precisión como la capacidad de recuperación de cada clase.

5.3 Análisis de Importancia de Variables (Feature Importance)

Para garantizar la robustez de los hallazgos, se analizó la importancia de las características en las tres arquitecturas principales (Figura 8). Este enfoque permite observar si diferentes algoritmos coinciden en los factores que determinan el resultado de un partido.

Como se muestra en la Figura, las variables generadas mediante ingeniería de características tuvieron un impacto determinante:

`xga_rolling` (Goles Esperados en Contra): Apareció consistentemente como un predictor crítico, validando que la capacidad de un equipo para limitar ocasiones de calidad al rival es tan importante como su capacidad goleadora.

`tklw_rolling` (Tackles ganados): Confirmó que la intensidad en la recuperación del balón es un indicador del éxito en el resultado final.

`dist_rolling`: La distancia promedio de tiro sugiere que los equipos que logran disparar desde posiciones más cercanas al arco tienen una mayor probabilidad de victoria.

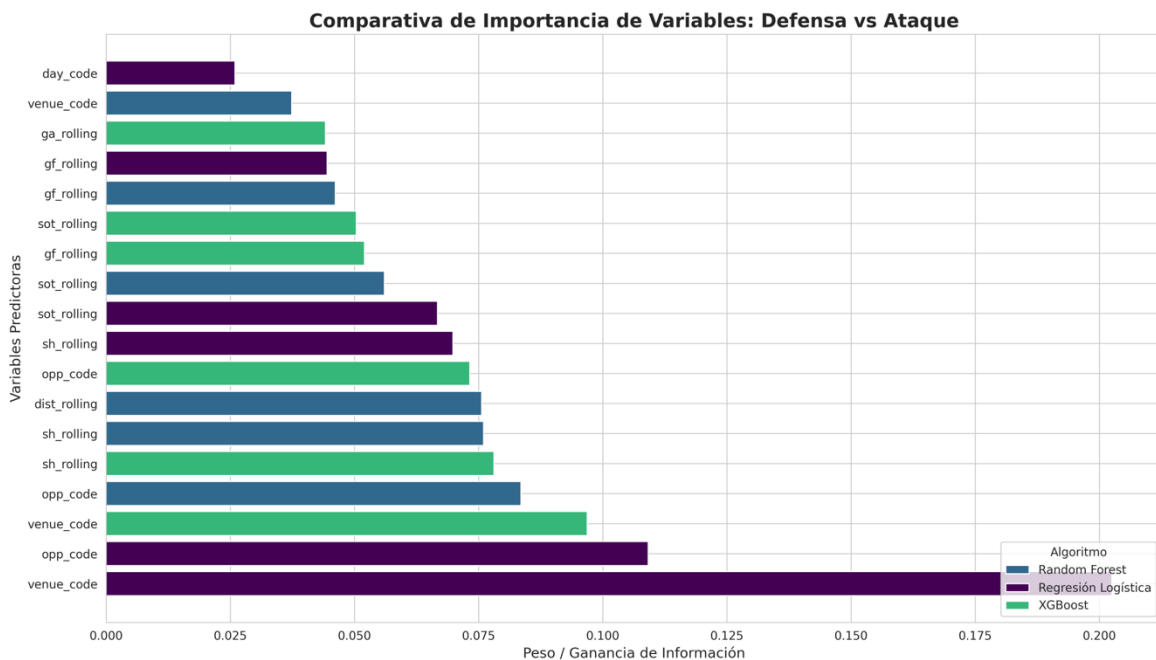


Figura 8 Comparativa de importancia de variables entre modelos lineales y de ensemble

El análisis de la Figura 8 revela un **Consenso Defensivo**: es notable que tanto en XGBoost como en la Regresión Logística, las métricas defensivas como `ga_rolling` (goles en contra) y `xga_rolling` se posicionan consistentemente en el top de importancia. Esto confirma que la capacidad de prevenir ocasiones del rival es un predictor crítico, validando la hipótesis central de esta investigación.



Cabe destacar que el modelo **SVM (Support Vector Machine)** no se incluyó en esta comparativa de relevancia de variables. Debido al uso de un **Kernel RBF (Radial Basis Function)**, el modelo opera en un espacio de características de dimensión infinita mediante el 'kernel trick', lo que impide la extracción directa de coef_ (coeficientes lineales) o pesos de importancia. Aunque existen métodos como la importancia por permutación, en este trabajo se decidió analizar las variables a partir de los modelos que ofrecen una interpretación más directa, como Regresión Logística, Random Forest y XGBoost. Esto permitió revisar si las métricas defensivas aparecían de forma consistente entre los modelos evaluados.

5.4 Análisis por Clase y Matriz de Confusión

El análisis por clase permite revisar con mayor detalle en qué resultados acierta o falla el modelo. Para ello se utilizó la matriz de confusión del modelo con mejor desempeño general, correspondiente a la Regresión Logística optimizada. Esta matriz muestra la relación entre los resultados reales y las predicciones generadas para las tres clases: derrota, empate y victoria.

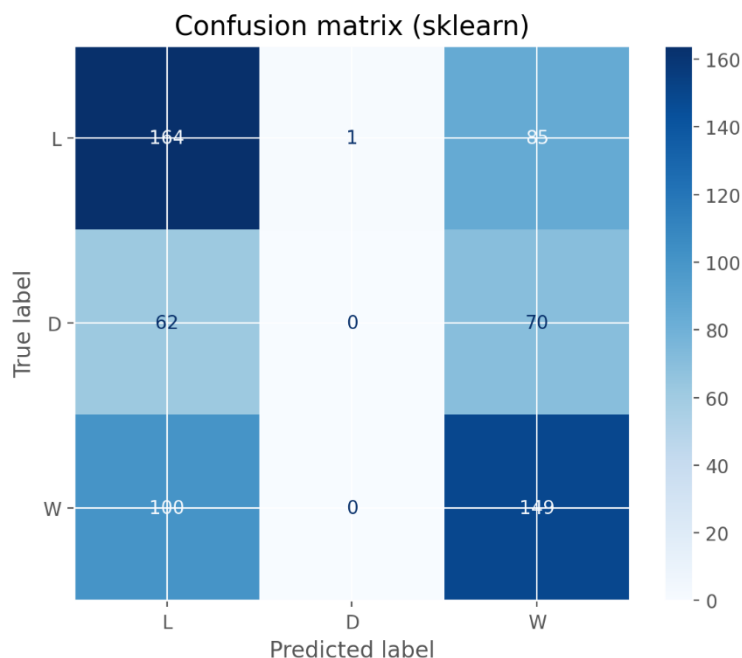


Figura 9. Matriz de confusión del modelo de Regresión Logística.

A partir de la Figura 9 se observa que el modelo tuvo mejor desempeño al identificar derrotas y victorias que al clasificar empates. Para la clase derrota se obtuvo un recall de 65.6%, mientras que para la clase victoria el recall fue de 59.8%. Estos resultados indican



que las variables construidas mediante promedios móviles ayudan a reconocer mejor los casos donde existe una diferencia más clara entre el rendimiento de los equipos.

En el caso del empate, el desempeño fue considerablemente menor. La matriz muestra que los empates reales fueron clasificados como derrotas o victorias, sin que el modelo lograra identificarlos correctamente como una clase independiente. Esto sugiere que, con las variables utilizadas, el empate no presenta un patrón suficientemente claro para separarlo de los otros dos resultados.

Una posible explicación es que el empate suele presentarse en partidos donde los equipos tienen comportamientos similares en sus métricas previas. Al utilizar promedios móviles ofensivos y defensivos, el modelo puede identificar con mayor facilidad diferencias marcadas entre equipos, pero tiene más dificultades cuando las estadísticas previas son parecidas. En estos casos, el resultado final puede depender de situaciones que no están incluidas en el conjunto de variables, como errores individuales, decisiones arbitrales, lesiones durante el partido o jugadas aisladas.

Este resultado no necesariamente indica un error en la metodología, sino una limitación del tipo de información utilizada. Para mejorar la identificación del empate sería necesario incorporar variables adicionales, como alineaciones, lesiones, contexto del partido, calendario, o métricas más detalladas de eventos dentro del juego.

También se observó confusión entre victorias y derrotas. Por ejemplo, una parte de las victorias reales fue clasificada como derrota, lo que refleja la dificultad de anticipar resultados en un deporte con alta variabilidad. Aunque los promedios móviles resumen el rendimiento reciente, no siempre capturan factores que pueden cambiar el desarrollo de un partido.

Finalmente, esta tendencia no fue exclusiva de la Regresión Logística. Los modelos Random Forest y XGBoost también presentaron dificultades para distinguir la clase empate. Esto refuerza la idea de que el problema no depende únicamente del algoritmo utilizado, sino también de las variables disponibles para representar cada partido.



5.5 Discusión de Resultados

Es importante contextualizar este resultado considerando la complejidad del problema abordado. En una primera etapa de experimentación, el modelo se planteó como un problema de clasificación binaria, considerando únicamente victoria y derrota, lo cual es común en la literatura de predicción deportiva, especialmente en deportes donde el empate es poco frecuente como el baloncesto o el fútbol americano. Bajo ese enfoque se alcanzaron valores de exactitud cercanos al 80%. Sin embargo, a sugerencia del director de tesis se incorporó el empate como tercera clase, lo que representa el problema real del fútbol. Este cambio redujo drásticamente el desempeño inicial hasta valores cercanos al 22%, lo que obligó a replantear la ingeniería de características y el proceso de optimización. El resultado final de 49.60% representa por tanto una recuperación significativa, y debe interpretarse en el contexto de un problema de tres clases con alta incertidumbre inherente.

El modelo de Regresión Logística optimizada alcanzó una exactitud de 49.60% en el conjunto de prueba. Este resultado se encuentra dentro de los valores reportados en trabajos relacionados sobre predicción de resultados de fútbol, especialmente cuando se consideran tres clases: victoria, empate y derrota. En general, la literatura muestra que superar niveles cercanos al 60% de exactitud sigue siendo complicado, debido a la incertidumbre propia del deporte y a la cantidad de factores que pueden influir en un partido.

En la Tabla 5.3 se muestra una comparación entre el modelo propuesto y algunos estudios previos reportados en la literatura.

Tabla 5.3: Comparación Accuracy con trabajos relacionados

Author(s)	Model / Method	Accuracy (%)
Rojas Perea (2026)	Logistic Regression (Optimized)	49.60
Obradović & Kečo (2024)	Decision Trees (LMT) / SVM	59.88
Båth & Sjöberg (2024)	Logistic Regression	47.00
Kaunitz et al. (2017)	Continuous Odds Simulation	47.60
Steijns (2020)	Combined Models (Players/Team)	47.00 – 53.00



Como se aprecia en la Tabla 5.3, el resultado obtenido en esta investigación supera los valores reportados por Báth y Sjöberg (2024), así como el resultado de Kaunitz et al. (2017), basado en simulaciones con cuotas de apuestas. También se encuentra dentro del rango reportado por Steijns (2020), cuyos modelos presentan valores entre 47% y 53%.

Por otra parte, el resultado de Obradović y Kečo (2024) es superior, con una exactitud de 59.88%. Sin embargo, esta comparación debe interpretarse con cautela, ya que cada estudio utiliza distintos modelos, conjuntos de datos, temporadas, variables y estrategias de validación. Por ello, los resultados no pueden compararse de manera directa en todos los casos.

En general, el desempeño alcanzado por el modelo propuesto se mantiene dentro de los márgenes observados en trabajos relacionados. Además, los resultados sugieren que las variables construidas mediante promedios móviles, junto con métricas como xG y xGA, aportan información útil para representar el rendimiento reciente de los equipos y apoyar la predicción de resultados.

Capítulo 6: Conclusiones y Recomendaciones

El presente capítulo sintetiza los principales hallazgos derivados del desarrollo e implementación de los modelos de Machine Learning propuestos en esta investigación. A partir del análisis de resultados presentado en el capítulo anterior, se realiza una interpretación global del desempeño de los modelos, considerando tanto su capacidad predictiva como sus limitaciones estructurales en un entorno de validación temporal realista dentro del fútbol profesional.

Asimismo, se exponen las conclusiones obtenidas en función de los objetivos planteados, se identifican las principales limitaciones metodológicas y de datos, y finalmente, se proponen líneas de trabajo futuro orientadas a mejorar la capacidad predictiva del sistema y ampliar el alcance de la investigación.

6.1 Conclusiones



El presente trabajo tuvo como objetivo desarrollar y evaluar modelos de Machine Learning orientados a la predicción de resultados de partidos de fútbol en la Liga Inglesa de Fútbol, bajo un enfoque metodológico que preservara la integridad temporal de los datos y evitara la fuga de información.

A partir del análisis experimental realizado, se derivan las siguientes conclusiones:

Cumplimiento del Objetivo General:

Se logró desarrollar un sistema predictivo capaz de superar tanto la probabilidad aleatoria (33.33%) como el rendimiento de un clasificador de mayoría basado en la frecuencia histórica de victoria local (44%). El modelo de Regresión Logística optimizado alcanzó una exactitud de 49.60% en los datos de prueba correspondientes a la temporada 2025. Este resultado muestra que las variables utilizadas contienen información útil para apoyar la predicción de resultados, aunque el problema continúa siendo complejo por la naturaleza del fútbol.

Importancia de la Solidez Defensiva:

El análisis de importancia de variables mostró que algunas métricas defensivas, como los goles esperados en contra (xGA) y los goles en contra (GA), tuvieron un papel relevante en los modelos evaluados. Esto indica que el comportamiento defensivo de los equipos aporta información importante para anticipar el resultado de un partido, especialmente cuando se analiza junto con el rendimiento reciente.

Eficiencia de Modelos Lineales en Entornos de Alta Incertidumbre:

Un resultado importante fue que la Regresión Logística obtuvo el mejor desempeño general frente a modelos más complejos como XGBoost y SVM. Esto sugiere que, para este conjunto de datos, un modelo lineal bien ajustado y acompañado de variables construidas mediante promedios móviles puede ofrecer resultados competitivos. En este caso, la mayor complejidad del modelo no garantizó una mejor predicción.

Naturaleza Estocástica del Empate:

El análisis por clase mostró que el empate fue el resultado más difícil de identificar. A diferencia de las victorias y derrotas, los empates no presentaron un patrón suficientemente



claro con las variables utilizadas. Esto puede deberse a que muchos partidos empatados ocurren en contextos de equilibrio entre equipos, donde pequeñas situaciones del juego pueden modificar el resultado final.

En conjunto, los resultados muestran que es posible obtener señales predictivas útiles a partir de datos históricos, siempre que se respete el orden temporal de los partidos y se construyan variables que representen el rendimiento reciente de los equipos. Sin embargo, también se observa que la predicción de fútbol tiene límites importantes, especialmente cuando se intenta distinguir partidos muy parejos.

Con base en los resultados obtenidos, se acepta la hipótesis general (H1) de esta investigación. El modelo desarrollado superó la probabilidad aleatoria de 33.33% y también el clasificador ingenuo basado en victoria local, con un valor de referencia de 44%. Esto indica que el uso de modelos de Machine Learning, aplicado a métricas ofensivas y defensivas históricas, puede aportar información útil para la predicción de resultados de fútbol.

6.2 Limitaciones

Si bien los resultados obtenidos apoyan la hipótesis planteada, también existen algunas limitaciones que deben considerarse:

- El modelo se entrenó y evaluó únicamente con datos de la Liga Inglesa de Fútbol, por lo que no se puede asegurar que el mismo comportamiento se mantenga en otras ligas.
- Se utilizó una ventana de tres partidos para calcular los promedios móviles. Esta decisión permitió representar el rendimiento reciente, aunque podría dejar fuera tendencias de más largo plazo.
- No se incluyeron variables externas como lesiones, carga de partidos, condiciones climáticas o valor de mercado de las plantillas.
- Tampoco se utilizaron datos de seguimiento espacial, los cuales podrían aportar información más detallada sobre presión, ocupación de espacios y comportamiento táctico.



6.3 Recomendaciones

Con el fin de ampliar los alcances de esta investigación, se proponen las siguientes líneas de trabajo:

Integración de Variables Contextuales:

Incorporar información sobre lesiones, días de descanso, calendario competitivo, condiciones climáticas y otros factores externos podría ayudar a mejorar la predicción, especialmente en partidos donde los equipos presentan métricas históricas similares.

Exploración de Modelos de Aprendizaje Profundo:

Dado que el fútbol constituye una serie temporal compleja, el uso de arquitecturas como LSTM o Transformers podría capturar dependencias dinámicas de mayor profundidad que los promedios móviles utilizados.

Uso de Datos Espaciales de Seguimiento:

La inclusión de coordenadas y métricas de ocupación de espacio permitiría modelar presión, control territorial y generación de superioridades numéricas.

Evaluación Multiliga:

Extender el análisis a otras competiciones europeas permitiría determinar si la relevancia de métricas defensivas constituye un patrón universal o dependiente del estilo táctico regional.

Desarrollo de un Modelo Híbrido para la Predicción del Empate:

Los resultados obtenidos evidenciaron que el empate representa la clase más difícil de predecir dentro de un modelo de clasificación multiclase. Como línea de trabajo futuro se propone desarrollar una arquitectura híbrida compuesta por dos modelos especializados: uno orientado a la clasificación binaria entre victoria y derrota, y otro entrenado específicamente para identificar partidos con alta probabilidad de empate. Ambos modelos generarían probabilidades independientes, y el resultado final se determinaría seleccionando la clase con mayor probabilidad entre los dos modelos. Este enfoque de



fusión podría mejorar significativamente la detección del empate sin sacrificar el desempeño en las otras dos clases.



Referencias

- Acuña Gómez, G., & Acuña Delgado, Á. (2016). El fútbol como producto cultural: Revisión y análisis bibliográfico. *Citius, Altius, Fortius*, 9(2), 31–58.
<https://doi.org/10.15366/citius2016.9.2>
- Amazon Web Services. (2025). *¿Qué es el machine learning? Explicación de la tecnología ML*. <https://aws.amazon.com/es/what-is/machine-learning/>
- Ameisen, E. (2020). *Building machine learning powered applications: Going from idea to product*. O'Reilly Media.
- Andrade, F. (2022, 18 de noviembre). *Prediciendo el Mundial de Qatar 2022 con Python* [Video]. YouTube. <https://www.youtube.com/>
- Bardisa Serrano, A. (2019). *Recopilación de información sobre fútbol y predicción de resultados* [Trabajo fin de grado, Universidad de Alicante]. Repositorio institucional.
- BetInAsia. (2025). *¿Qué son los goles esperados (xG) en el fútbol?*
<https://betinasia.com/es/>
- Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.
- Box, G. E. P., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time series analysis: Forecasting and control* (5.ª ed.). Wiley.
- Calderón Pérez-Lozao, S. (2017). *Predicción de resultados deportivos con técnicas de machine learning aplicado al fútbol* [Trabajo fin de grado, Universidad Carlos III de Madrid]. Repositorio institucional.
- Chavez Rivera, J. E. (2024). *Predicción de resultados en el fútbol a través de un producto software implementando técnicas de machine learning y análisis de datos* [Trabajo de grado, Universidad CESMAG]. Repositorio institucional.
- Deisenroth, M. P., Faisal, A. A., & Ong, C. S. (2019). *Mathematics for machine learning*. Cambridge University Press.
- Ferreira, P. H., Louzada, F., & do Nascimento, D. C. (2023). Partidos de fútbol: ¿Suerte o causalidad? Introducción a los modelos estadísticos de conteo y redes neuronales artificiales para la predicción de juegos. *Revista Científica*, 37, 9–21.
- Filatova, E. (2025, 20 de mayo). *Predicción de resultados de jugadores mediante TA-LSTM*. NeuronUP.
- Forrest, D., & Pérez, L. (2019). *Las apuestas: Beneficios y riesgos para el deporte*. Funcas.



Galekwa, R. M., Tshimula, J. M., Tajeuna, E. G., & Kyandoghere, K. (2025). *A systematic review of machine learning in soccer betting: Techniques, challenges, and future directions*. Preprints.org. <https://doi.org/10.20944/preprints202501.2060.v1>

Géron, A. (2023). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow* (3.^a ed.). O'Reilly Media.

Géron, A. (2025). *Hands-on machine learning with Scikit-Learn and PyTorch: Concepts, tools, and techniques to build intelligent systems*. O'Reilly Media.

Gift, N., & Deza, A. (2021). *Practical MLOps: Operationalizing machine learning models*. O'Reilly Media.

Google for Developers. (2025). *Clasificación: ROC y AUC | Machine learning*. <https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc>

Haythornthwaite, M. (s. f.). *Football_Prediction_Project* [Código fuente]. GitHub. Recuperado el 3 de junio de 2026, de https://github.com/mhaythornthwaite/Football_Prediction_Project

Huyen, C. (2022). *Designing machine learning systems: An iterative process for production-ready applications*. O'Reilly Media.

Hyndman, R. J., & Athanasopoulos, G. (2021). *Forecasting: Principles and practice* (3.^a ed.). OTexts. <https://otexts.com/fpp3/>

Kaunitz, L., Zhong, S., & Kreiner, J. (2017). *Beating the bookies with their own numbers—and how the online sports betting market is rigged*. arXiv. <https://arxiv.org/abs/1710.02824>

Kulesa, A., Krzywinski, M., Blainey, P., & Altman, N. (2015). Points of significance: Sampling distributions and the bootstrap. *Nature Methods*, 12(6). <https://doi.org/10.1038/nmeth.33>

Martín Castellanos, A. (s. f.). *Métricas en el fútbol moderno: Concepto, análisis y uso*. Objetivo Analista. <https://objetivoanalista.com/metricas-en-el-futbol-moderno/>

Medrano Martínez, J. M. (2018). *Aplicación de técnicas de machine learning para la predicción de resultados deportivos* [Trabajo fin de grado, Universidad de Alcalá]. Repositorio institucional.

Murphy, K. P. (2012). *Machine learning: A probabilistic perspective*. MIT Press.

OddsMatrix. (2025, 23 de junio). *Machine learning in sports betting: How will this new technology impact sports betting?* <https://oddsmatrix.com/>



Ovando Fuentealba, R. (2025). *Predicción de resultados de partidos de la liga profesional de fútbol chileno usando algoritmos de machine learning* [Tesis de maestría, Universidad del Desarrollo]. Repositorio institucional.

Prosise, J. (2021). *Applied machine learning and AI for engineers*. O'Reilly Media.

Raschka, S., Mirjalili, V., & Liu, H. (2022). *Hands-on machine learning with PyTorch*. Packt Publishing.

Rendón Arango, S. (2023). *Machine learning como herramienta para predecir resultados de Formula 1* [Proyecto de grado, Universidad de los Andes].

Steijns, B. A. G. M. (2020). *The characteristics of winning football matches: Determining a model that can predict the match outcome of football matches* [Tesis de maestría, Tilburg University]. Repositorio institucional.

TabularConf. (2021, 6 de febrero). *Aplicaciones del Machine Learning al mundo del fútbol - Javi Fernandez & Juan Calderón* [Video]. YouTube.
<https://www.youtube.com/watch?v=7J1cF6tg-KI>

Tempone, P. M. (2017). *Predicción de victorias en los equipos locales del fútbol argentino según las acciones de los jugadores* [Trabajo final integrador, Universidad de Buenos Aires]. Repositorio institucional.

Theobald, O. (2021). *Machine learning for absolute beginners: A plain English introduction* (3.^a ed.). Scatterplot Press.

Tiwari, E., Sardar, P., & Jain, S. (2020). *Football match result prediction using neural networks and deep learning*. National Institute of Technology Kurukshetra.

Universidad Rey Juan Carlos. (2024). *Modelos de machine learning para la predicción de partidos de fútbol* [Trabajo fin de grado]. Repositorio institucional.

Viderström, O. B., & Sjöberg, V. (2024). *Beating the bookies: Harnessing machine learning and the Kelly criterion for Premier League betting profits* [Tesis de maestría, Lund University]. Repositorio institucional.

Villar Ortiz, A. (2019). Machine learning para la predicción de eventos en la NBA. *International Research Journal of Engineering and Technology*, 6(2).

Wickwire, A. (2025, 3 de junio). *Calibración por encima de precisión: La clave para unas apuestas deportivas más inteligentes*. OpticOdds Blog.

.



Curriculum Vitae

Soy Ingeniero en Sistemas Computacionales por la Universidad Autónoma de Chihuahua y candidato al grado de Maestro en Ingeniería en Computación por la misma institución. Mi formación académica se ha orientado principalmente hacia el área de Inteligencia Artificial, Machine Learning y análisis de datos aplicados a la solución de problemas reales.

Cuento con más de veinte años de experiencia docente en nivel básico y más de diez años como profesor universitario en instituciones como la Universidad Autónoma de Chihuahua (UACH), la Universidad del Desarrollo Profesional (UNIDEP) y la Universidad Tecmilenio, además de experiencia en nivel medio superior. He impartido asignaturas relacionadas con programación, bases de datos, desarrollo de software, inteligencia artificial y análisis de datos, integrando tanto fundamentos teóricos como aplicaciones prácticas.

En el ámbito profesional, me he desempeñado como desarrollador de software y consultor en tecnologías de la información. Durante diez años ocupé el cargo de Jefe de Unidad en el Área de Sistemas de la Facultad de Ciencias de la Cultura Física de la Universidad Autónoma de Chihuahua, donde estuve a cargo de la administración tecnológica, soporte institucional y desarrollo de soluciones informáticas para la gestión académica y administrativa.

Actualmente soy fundador y director de ESISCOM (Especialistas en Sistemas Computacionales), empresa dedicada al desarrollo de software, soporte tecnológico y soluciones informáticas empresariales en el estado de Chihuahua.

Mis principales áreas de interés incluyen el aprendizaje automático, la minería de datos, la analítica aplicada al deporte y el desarrollo de sistemas inteligentes orientados a la toma de decisiones.

Correo electrónico: aperea@uach.mx

Esta tesis/disertación fue mecanografiada por Adalberto Emmanuel Rojas Perea.