UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO

FACULTAD DE INGENIERÍA ELÉCTRICA ELECTRÓNICA INFORMÁTICA Y MECÁNICA

ESCUELA PROFESIONAL DE INGENIERÍA INFORMÁTICA Y DE SISTEMAS

TESIS

INFLUENCIA METEOROLÓGICA EN LAS ESTRATEGIAS DE PREDICCIÓN DE PARTIDOS DE FÚTBOL MEDIANTE REDES NEURONALES RECURRENTES

PRESENTADO POR:

Br. ANTHONY MAYRON LOPEZ OQUENDO

Br. ERIK OMAR ALEGRIA VALENCIA

PARA OPTAR AL TÍTULO PROFESIONAL DE INGENIERO INFORMÁTICO Y DE SISTEMAS

ASESOR:

Dr. LUIS BELTRAN PALMA TTITO

CUSCO – PERU
2025

Universidad Nacional de San Antonio Abad del Cusco

INFORME DE SIMILITUD

(Aprobado por Resolución Nro.CU-321-2025-UNSAAC)

El que suscribe, el **Asesor** ..Luis Beltran Palma Tito.....
..... quien aplica el software de detección de similitud al trabajo de investigación/tesis titulada: ..INFLUENCIA METEOROLOGICA EN LAS ESTRATEGIAS DE PREDICCION DE PARTIDOS DE FUTBOL MEDIANTE REDES NEURONALES RECURRENTES
.....

Presentado por: ..ANTHONY MAYRON LOPEZ ORDENAO.. DNI N° ..72849150..;
presentado por: ..ERIK OMAR ALEGRIA VALENCIA.. DNI N°: ..23999373..
Para optar el título Profesional/Grado Académico de ..INGENIERO INFORMATICO Y DE SISTEMAS.....

Informo que el trabajo de investigación ha sido sometido a revisión por ..2.. veces, mediante el Software de Similitud, conforme al Art. 6° del ***Reglamento para Uso del Sistema Detección de Similitud en la UNSAAC*** y de la evaluación de originalidad se tiene un porcentaje de ..3..**%.**

Evaluación y acciones del reporte de coincidencia para trabajos de investigación conducentes a grado académico o título profesional, tesis

Porcentaje	Evaluación y Acciones	Marque con una (X)
Del 1 al 10%	No sobrepasa el porcentaje aceptado de similitud.	X
Del 11 al 30 %	Devolver al usuario para las subsanaciones.	
Mayor a 31%	El responsable de la revisión del documento emite un informe al inmediato jerárquico, conforme al reglamento, quien a su vez eleva el informe al Vicerrectorado de Investigación para que tome las acciones correspondientes; Sin perjuicio de las sanciones administrativas que correspondan de acuerdo a Ley.	

Por tanto, en mi condición de Asesor, firmo el presente informe en señal de conformidad y **adjunto** las primeras páginas del reporte del Sistema de Detección de Similitud.

Cusco, 15 de ENERO de 2026

.....
Firma

Post firma..Luis Beltran Palma Tito

Nro. de DNI..23999672

ORCID del Asesor..0000-0002-0950-5369

Se adjunta:

1. Reporte generado por el Sistema Antiplagio.
2. Enlace del Reporte Generado por el Sistema de Detección de Similitud: **oid:** ..27259:546515422..

ANTHONY MAYRON LOPEZ OQUENDO

INFLUENCIA METEOROLÓGICA EN LAS ESTRATEGIAS DE PREDICCIÓN DE PARTIDOS DE FÚTBOL MEDIANTE REDES N...

Universidad Nacional San Antonio Abad del Cusco

Detalles del documento

Identificador de la entrega
trn:oid:::27259:546515422

Fecha de entrega
15 ene 2026, 2:19 p.m. GMT-5

Fecha de descarga
15 ene 2026, 2:26 p.m. GMT-5

Nombre del archivo
anthony lopez oquendo tesis.pdf

Tamaño del archivo
2.5 MB

129 páginas

29.759 palabras

188.713 caracteres

3% Similitud general

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para ca...

Filtrado desde el informe

- Bibliografía
- Texto citado
- Texto mencionado
- Coincidencias menores (menos de 12 palabras)

Exclusiones

- N.º de coincidencias excluidas

Fuentes principales

1% Fuentes de Internet
0% Publicaciones
3% Trabajos entregados (trabajos del estudiante)

Marcas de integridad

N.º de alertas de integridad para revisión

No se han detectado manipulaciones de texto sospechosas.

Los algoritmos de nuestro sistema analizan un documento en profundidad para buscar inconsistencias que permitirían distinguirlo de una entrega normal. Si advertimos algo extraño, lo marcamos como una alerta para que pueda revisarlo.

Una marca de alerta no es necesariamente un indicador de problemas. Sin embargo, recomendamos que preste atención y la revise.

Dedicatoria

Dedico este trabajo académico a mi mamá, mi hermana, mi papá, mi enamorada y mi mejor amigo, por ser mi apoyo incondicional a lo largo de mi formación universitaria. Su aliento, paciencia, comprensión y amor han sido las condiciones antrópicas de mi crecimiento, gracias por su compañía que me permitió desarrollarme profesionalmente, aprender con libertad y explorar más allá del conocimiento, los logros se construyen junto a quienes nos acompañan (Anthony Lopez).

Primeramente, quiero dedicar esta tesis a Dios por guiar siempre mi camino y darme la fortaleza necesaria para lograr mis objetivos; a mis padres, mis hijos, mis hermanos, mi esposa, mis sobrinos y tíos, ya que, gracias a ellos, por su apoyo, amor, comprensión, paciencia, sacrificio, consejos y perseverancia, se logró el objetivo de concluir mi carrera y hacer de mí una mejor persona. Este logro marca mi primer hito personal y profesional, abriendo el camino a nuevas oportunidades y un futuro más claro, aprecio demasiado las lecciones de vida que me han compartido y por el cariño que siempre me han dado. Mi agradecimiento y gratitud hacia mi familia y amigos es imposible de expresar. Esta tesis es un agradecimiento, reconocimiento a ti papá y mamá a la eterna admiración y amor que siento por ustedes. Gracias por ser los mejores padres del mundo y gracias a mis hijos por ser el motor de mi vida (Erick Alegria).

Resumen

La predicción de resultados futbolísticos, inherentemente estocástica, encuentra su máxima complejidad en el contexto sudamericano por su variabilidad altitudinal extrema y térmica. Esta investigación aborda la subestimación de los factores ambientales, determinando la influencia predictiva de variables meteorológicas mediante el uso de Redes Neuronales Recurrentes (RNN) en cuatro ligas profesionales de la región andina: Colombia, Chile, Ecuador y Perú.

Los hallazgos revelaron que la incorporación de datos climáticos actúa como un catalizador de precisión no uniforme, dependiente del contexto nacional. En la liga peruana, esta adición no solo incrementó la potencia predictiva del modelo en un 6% respecto a su base puramente deportiva, sino que también optimizó su eficiencia computacional al permitir la simplificación de la arquitectura neuronal. El impacto es tangible, estableciendo referencias significativas en las ligas analizadas. Se concluye que, si bien la mejora es contextual, la evidencia global demuestra que, en geografías variables, el clima trasciende su rol secundario para erigirse como un determinante crucial del resultado final.

Palabras clave: Predicción de resultados futbolísticos, Redes neuronales recurrentes, Variables ambientales, Andes.

Abstract

The prediction of football outcomes, inherently stochastic, reaches its greatest complexity in the South American context due to extreme altitudinal and thermal variability. This research addresses the underestimation of environmental factors, determining the predictive influence of meteorological variables through the use of Recurrent Neural Networks (RNN) across four professional leagues in the Andean region: Colombia, Chile, Ecuador, and Peru.

The findings reveal that the incorporation of climate data functions as a non-uniform accuracy catalyst, contingent on national context. In the Peruvian league, this addition not only increased the model's predictive power by 6% compared to its purely sports-based baseline, but also enhanced computational efficiency by allowing the simplification of the neural architecture. The impact is tangible, establishing meaningful benchmarks within the analyzed leagues. The study concludes that, although improvement is context-dependent, the overall evidence demonstrates that in geographically variable settings, climate transcends its secondary role to become a crucial determinant of the final outcome.

Keywords: Football match prediction, Recurrent neural networks, Meteorological variables, Andean region / South America.

Introducción

La predicción de resultados en el fútbol constituye un desafío científico de considerable complejidad debido a la naturaleza estocástica inherente al deporte. A diferencia de sistemas deterministas, los encuentros futbolísticos están influenciados por múltiples factores que interactúan de manera no lineal: las capacidades técnicas y tácticas de los equipos, el rendimiento individual de los jugadores, decisiones arbitrales, factores psicológicos y, crucialmente, las condiciones ambientales en las que se desarrolla el encuentro. Esta multiplicidad de variables dificulta el establecimiento de patrones predictivos robustos mediante métodos tradicionales basados en análisis subjetivo o en el uso limitado de información histórica.

El territorio peruano presenta características geográficas excepcionales que lo convierten en un laboratorio natural para estudiar la influencia de factores ambientales en el rendimiento deportivo. La marcada zonificación altitudinal del país, consecuencia de la presencia de la Cordillera de los Andes, genera condiciones atmosféricas significativamente heterogéneas entre las distintas sedes donde se practican competiciones futbolísticas. Esta variabilidad incluye diferencias sustanciales en presión atmosférica, concentración de oxígeno, temperatura y humedad relativa. Sin embargo, la investigación sistemática sobre cómo estas variables meteorológicas influyen específicamente en los resultados de partidos de fútbol en contextos andinos permanece limitada.

La presente investigación aborda esta laguna de conocimiento mediante la aplicación de técnicas avanzadas de aprendizaje profundo, específicamente redes neuronales recurrentes tipo LSTM, para evaluar la capacidad predictiva de modelos que incorporan variables meteorológicas en comparación con aproximaciones que utilizan exclusivamente estadísticas deportivas tradicionales. El enfoque metodológico considera datos de ligas de fútbol en Perú, Chile, Ecuador y Colombia, países que comparten características geográficas similares al estar ubicados sobre la Cordillera de los Andes, permitiendo así un análisis de los efectos ambientales en contextos comparables.

Índice General

Índice de Figuras

CAPITULO I

Aspectos generales

1.1. Planteamiento del problema

1.1.1. Descripción del problema

La predicción de resultados en encuentros futbolísticos enfrenta desafíos metodológicos fundamentales derivados de la complejidad dinámica del deporte. Los modelos tradicionales de predicción han tendido a concentrarse en variables endógenas al juego, tales como estadísticas históricas de rendimiento, posesión de balón, goles esperados y otras métricas derivadas directamente de la acción deportiva. Si bien estos factores indudablemente influyen en el resultado final, su poder predictivo se ve limitado por la omisión de variables contextuales que pueden modular significativamente el rendimiento de los equipos.

Entre estas variables contextuales, los factores ambientales merecen particular atención en el contexto geográfico andino. La literatura científica en fisiología de ejercicio ha establecido que la altitud afecta la capacidad aeróbica, el metabolismo energético y la recuperación muscular. Similarmente estos factores pueden influir en aspectos técnicos del juego como el control del balón, la precisión en pases y la resistencia física de los jugadores. No obstante, la mayoría de los estudios predictivos en fútbol han sido desarrollados en contextos geográficos relativamente homogéneos, principalmente en Europa, donde las variaciones altitudinales y climáticas son menos pronunciadas.

1.1.2. Identificación del problema

El fútbol peruano y, por extensión, el fútbol andino, se disputa en condiciones ambientales marcadamente heterogéneas. Los estadios se ubican desde el nivel del mar hasta altitudes que superan los tres mil metros, con amplitudes térmicas diarias que pueden alcanzar veinticinco grados Celsius. Esta variabilidad genera condiciones de juego sustancialmente diferentes que podrían no ser capturadas adecuadamente por modelos predictivos desarrollados en contextos climáticamente uniformes.

Particularmente relevante resulta el caso peruano, donde las características geográficas y climáticas exhiben una heterogeneidad espacial pronunciada. Los equipos que compiten en la región norte del país enfrentan las temperaturas más elevadas del territorio nacional, con condiciones de humedad característica de zonas costero-desérticas. Posteriormente, estos mismos equipos pueden verse obligados a disputar encuentros en localidades serranas donde las condiciones atmosféricas contrastan radicalmente como temperaturas próximas o bajo cero grados Celsius, precipitaciones pluviales frecuentes y altitudes que superan ampliamente los tres mil quinientos metros sobre el nivel del mar. Esta transición abrupta entre microclimas geográficos impone demandas de aclimatación fisiológica que exceden las experimentadas en circuitos futbolísticos de regiones con menor diversidad orográfica. La adaptación metabólica requerida para el rendimiento óptimo en estas condiciones divergentes constituye un factor que raramente ha sido incorporado en arquitecturas predictivas convencionales.

En consecuencia, resulta necesario desarrollar aproximaciones metodológicas que integren explícitamente estas variables meteorológicas y evalúen su contribución marginal a la capacidad predictiva.

1.2. Formulación del problema

1.2.1. Problema general

¿En qué medida la incorporación de variables meteorológicas influye la capacidad de predicción de resultados de partidos de fútbol en contextos geográficos caracterizados por alta variabilidad altitudinal y climática?

1.2.2. Problemas específicos

- ¿Qué variables estadísticas del partido de futbol resultan más informativas para la predicción en el contexto estudiado?
- ¿Qué elementos meteorológicos ejercen mayor influencia en la capacidad predictiva de partidos de futbol?

- ¿En qué medida la heterogeneidad geográfica y climática de las distintas ligas estudiadas afecta la predictiva?
- ¿En qué medida mejora la eficiencia predictiva al incluir variables ambientales, al comparar sistemáticamente dos experimentos: (a) un modelo que utiliza exclusivamente variables deportivas que capturan tendencias recientes de los equipos, y (b) un modelo que incorpora variables meteorológicas agregadas mediante ventanas temporales que reflejan las tendencias recientes del clima asociado a los equipos?

1.3. Objetivos

1.3.1. Objetivo general

Determinar en qué medida la incorporación de variables meteorológicas en modelos de redes neuronales recurrentes mejora la capacidad de predicción de resultados de partidos de fútbol en contextos geográficos con alta variabilidad altitudinal y climática.

1.3.2. Objetivo específico

- Medir qué variables estadísticas del partido de futbol resultan más informativas para la predicción de resultados en el contexto estudiado, utilizando técnicas de aprendizaje supervisado basadas en ganancia de información.
- Determinar qué elementos meteorológicos ejercen mayor influencia en la capacidad predictiva de los modelos, mediante técnicas de aprendizaje supervisado basadas en ganancia de información.
- Evaluar la variabilidad el rendimiento predictivo del modelo desagregado por cada país para identificar posibles sesgos geográficos o climáticos en la predicción
- Comparar la eficiencia predictiva entre dos experimentos: (a) un modelo que emplea exclusivamente variables deportivas que capturan tendencias recientes de los equipos, y (b) un modelo que incorpora variables meteorológicas agregadas mediante ventanas temporales que representan las tendencias

recientes del clima asociado a los equipos, con el fin de establecer la mejora atribuible a la inclusión de variables ambientales.

1.4. Justificación.

Esta formulación del problema permite abordar tanto la relevancia teórica de los factores ambientales en el rendimiento deportivo como su utilidad práctica en sistemas de predicción, contribuyendo al desarrollo de modelos contextualmente adaptados a la realidad geográfica andina.

Esta investigación representa un reto debido a la complejidad dinámica de un partido de fútbol y su relación con el comportamiento ambiental. En este estudio, se aborda y contribuye a llenar un vacío existente, analizando cómo las variables meteorológicas afectan la dinámica del fútbol en Perú y en otros países ubicados a lo largo de la Cordillera de los Andes. Este análisis no solo amplía el conocimiento científico, sino que también tendrá un impacto en el contexto cultural del país, influyendo en estrategias deportivas, decisiones económicas y en la conexión de las comunidades con su deporte más querido.

Este trabajo se justifica por varias razones clave:

Brecha de Conocimiento: Se observa una evidente falta de comprensión detallada sobre cómo la zonificación altitudinal influye en el rendimiento deportivo, las estrategias de juego y las decisiones relacionadas con las apuestas, específicamente en el contexto de Perú. La carencia de estudios profundos en esta área destaca la necesidad urgente de abordar y llenar este vacío de conocimiento.

Desarrollo del Fútbol: Para federaciones deportivas y organizadores de torneos, los resultados pueden informar decisiones sobre programación de partidos, especialmente en contextos donde condiciones meteorológicas extremas podrían comprometer la calidad del espectáculo o la seguridad de los participantes. Las implicaciones prácticas de esta investigación se extienden a múltiples ámbitos del ecosistema futbolístico. Para cuerpos técnicos y preparadores físicos, los hallazgos pueden informar estrategias de preparación específicas para partidos en condiciones ambientales desafiantes. El conocimiento sobre

cómo variables meteorológicas específicas afecta el rendimiento puede guiar decisiones sobre aclimatación, hidratación, estrategias tácticas y rotación de jugadores.

Desafíos Únicos en Perú: Perú ofrece un entorno ideal para estudiar la relación entre los factores ambientales y el rendimiento en el fútbol. Analizar estas dinámicas en un país con tan rica diversidad geográfica tiene el potencial de generar conocimientos aplicables a otras regiones con características similares.

Aplicación en Otros Deportes: Los resultados de este estudio podrían tener aplicaciones más allá del fútbol. Otros deportes que se juegan en condiciones similares podrían beneficiarse de este conocimiento, y los modelos desarrollados podrían adaptarse para hacer predicciones en diferentes contextos geográficos.

Aplicación de Tecnologías Emergentes: La aplicación de técnicas avanzadas de aprendizaje profundo en el análisis y predicción de eventos deportivos es un campo en constante crecimiento. Este estudio contribuye a la literatura existente al explorar cómo estas tecnologías pueden adaptarse a las particularidades de un país con una marcada zonificación altitudinal, como Perú.

Motivación

A medida que el fútbol se convierte en un elemento clave de la identidad peruana, surge la necesidad de comprender las complejidades que introduce la zonificación altitudinal propia del país en este deporte. Desde las tácticas de los equipos hasta las apuestas que generan un flujo económico significativo, este estudio se posiciona como un referente en un campo de investigación poco explorado, impulsado por la implementación de tecnologías avanzadas de aprendizaje profundo. Su contribución no solo enriquecerá la comprensión del fútbol en Perú, sino que también ofrecerá conocimientos valiosos que podrán ser aplicados en otras regiones con características geográficas similares.

Este trabajo aporta:

Cultural Deportivo: El fútbol es un deporte que tiene un profundo impacto cultural en numerosos países. Comprender cómo la zonificación altitudinal influye en las predicciones de los resultados podría modificar la manera en que las comunidades se

vinculan con el deporte, especialmente en regiones donde existen variaciones significativas de altitud y temperatura.

Estrategia Deportiva: Los resultados de esta investigación podrían influir directamente en las estrategias de los equipos y entrenadores. La adaptación a las condiciones de altitud podría convertirse en un factor clave en la planificación estratégica, lo cual impactaría en la competitividad de los equipos y, por ende, en el interés de los aficionados, jugadores y entrenadores. Además, este conocimiento mejoraría la toma de decisiones tácticas y la apreciación del juego.

Aporte Económico: En sociedades donde las apuestas deportivas son comunes, esta investigación podría influir en la manera en que se realizan las apuestas y en las expectativas de los apostadores. Este impacto económico y social sería significativo, ya que el interés en las apuestas deportivas está estrechamente relacionado con el interés en los eventos deportivos. En Perú, el sector de las apuestas genero un movimiento económico cercano a los mil millones de dólares, siendo el 90 % de las apuestas relacionadas con el fútbol. Según estimaciones del Ministerio de Comercio Exterior y Turismo (Mincetur), el movimiento económico ascendería a 4,500 millones de soles.

Contribución al Conocimiento Científico: La investigación en la intersección de deportes, geografía y tecnologías emergentes es aún limitada. Este estudio tiene como objetivo llenar este vacío, proporcionando una comprensión más profunda de cómo la altitud y las variables asociadas impactan en la capacidad predictiva de los modelos de aprendizaje profundo en un contexto deportivo.

1.5. Alcances

La investigación cubre partidos de fútbol profesional disputados en las principales ligas de Perú, Chile, Ecuador y Colombia durante el período comprendido entre los años 2017 y parte del 2025, determinados por la disponibilidad de datos. La inclusión de múltiples países permite evaluar la robustez de los hallazgos a través de diferentes contextos dentro del marco geográfico andino, aumentando la generalización de las conclusiones.

El análisis se restringe a partidos donde está disponible información completa tanto de estadísticas deportivas como de condiciones meteorológicas, garantizando así la integridad del análisis multivariado. Las variables meteorológicas son obtenidas de fuentes satelitales, proporcionando mediciones objetivas y consistentes a través de las diferentes ubicaciones geográficas.

1.6. Limitaciones

Falta de información detallada sobre el estado físico de los jugadores, lesiones, suspensiones, tácticas propuestas, el director técnico y otros factores de alineación, los cuales pueden influir considerablemente en el resultado de los partidos, pero no son reflejados por las estadísticas agregadas del equipo. Asimismo, no se considera aspectos tácticos, motivacionales y psicológicos que podrían ser clave en determinados encuentros. Los datos utilizados provienen de fuentes que combinan recolección manual y visión por computadora, lo que puede generar errores e imprecisiones en las estadísticas y cronología de los eventos, afectando la calidad de los datos. Además, se limita a partidos con transmisión o cobertura oficial, excluyendo aquellos sin registros suficientes en localidades remotas o con infraestructura limitada.

Las variables meteorológicas, aunque obtenidas de fuentes confiables, representan mediciones para la ubicación general del estadio en ventanas temporales específicas, y pueden no capturar perfectamente las condiciones micro climáticas exactas en el terreno de juego durante las dos horas de duración del partido. Variaciones localizadas en condiciones dentro del estadio o cambios rápidos durante el transcurso del encuentro no son capturados completamente por las mediciones utilizadas.

Finalmente, la naturaleza observacional del estudio impide el establecimiento de relaciones causales definitivas entre variables meteorológicas y resultados. Las asociaciones identificadas, aunque informativas, son interpretadas con la cautela apropiada para estudios no experimentales donde no es posible controlar completamente todas las variables confundentes.

CAPITULO II

Marco teórico

2.1. Marco Teórico

2.1.1. El Fútbol y su Influencia Cultural

El fútbol constituye un fenómeno sociocultural de primera magnitud en el Perú, trascendiendo su dimensión deportiva para convertirse en un catalizador de identidad colectiva y cohesión social. A diferencia de otras manifestaciones culturales, este deporte posee la capacidad única de articular diversos estratos sociales, regiones geográficas y tradiciones locales en torno a una experiencia compartida que refuerza el sentido de pertenencia nacional. Los encuentros deportivos de las selecciones nacionales no se reducen a meros eventos competitivos, sino que se configuran como rituales colectivos donde las victorias o derrotas adquieren significados que trascienden el resultado deportivo, consolidándose como experiencias que fortalecen los lazos de solidaridad ciudadana y la construcción de una narrativa nacional compartida (Escalona, 2021). Este fenómeno se observa particularmente en países sudamericanos donde el fútbol representa uno de los principales vehículos de expresión identitaria y movilización social. Desde una perspectiva económica, el fútbol ha evolucionado hacia una industria globalizada que genera valor económico sustancial mediante múltiples canales: derechos de transmisión, patrocinios corporativos, comercialización de productos oficiales y turismo deportivo. La profesionalización creciente del deporte ha estimulado el desarrollo de infraestructura especializada, creación de empleos directos e indirectos, y transferencias económicas significativas entre clubes y federaciones nacionales (Amaya Gómez; Luis Ángel, 2022).

El análisis del rendimiento futbolístico ha experimentado una transformación radical durante las últimas dos décadas, transitando desde observaciones cualitativas basadas en la experiencia de entrenadores hacia sistemas cuantitativos sofisticados que integran tecnologías de captura de datos en tiempo real. Esta evolución responde a la creciente competitividad del fútbol profesional y la necesidad de optimizar el rendimiento mediante decisiones fundamentadas en evidencia empírica. Los sistemas contemporáneos de

análisis de rendimiento emplean tecnologías de rastreo óptico y dispositivos inerciales portátiles que registran métricas físicas, tácticas y técnicas con precisión milimétrica (René Manassé Galekwa; Jean Marie Tshimula; Etienne Gael Tajeuna; Kyamakya Kyandoghere, 2024). Estas herramientas permiten cuantificar variables como distancia recorrida, velocidad de desplazamiento, aceleraciones, zonas de actividad en el campo, precisión en pases y eficiencia en duelos individuales. La disponibilidad de estos datos ha democratizado parcialmente el acceso a información previamente exclusiva de organizaciones con recursos sustanciales. La incorporación de paradigmas analíticos provenientes de la ciencia de datos ha introducido metodologías estadísticas avanzadas y algoritmos de aprendizaje automático en el análisis futbolístico (René Manassé Galekwa; Jean Marie Tshimula; Etienne Gael Tajeuna; Kyamakya Kyandoghere, 2024). Estos enfoques permiten identificar patrones tácticos complejos, evaluar probabilidades de eventos específicos durante el juego y desarrollar modelos predictivos que informan decisiones estratégicas tanto en preparación previa como durante la competición (Daniel Memmert ; Dominik Raabe, 2023).

2.1.2 Predicción de Resultados en Fútbol

2.1.2.1 Modelos estadísticos clásicos

El deporte ha evolucionado mucho más allá de las estadísticas tradicionales como goles anotados, tiros o porcentajes de posesión. En el juego actual, los conocimientos basados en datos son indispensables para equipos, entrenadores y analistas que buscan optimizar el rendimiento, mejorar la toma de decisiones tácticas y obtener una ventaja competitiva. Los modelos basados en distribuciones de Poisson representan otro enfoque clásico ampliamente utilizado para estimar probabilidades de diferentes marcadores (Amadu, 2024) (Rory Bunker; Calvin Yeung; Teo Susnjak; Chester Espie; Keisuke Fujii, 2023). Estos modelos asumen que el número de goles anotados por cada equipo sigue una distribución de Poisson independiente, cuyos parámetros de tasa se estiman mediante regresión considerando variables como fuerza ofensiva, capacidad defensiva, valor en el mercado, goles a favor, goles en contra y ventaja de localía. A pesar de su simplicidad

conceptual, estos modelos han demostrado capacidad predictiva razonable en diversas competiciones (Gómez & Reyes, 2024). Con el auge de las tecnologías de seguimiento y la disponibilidad de grandes conjuntos de datos que registran cada movimiento de los jugadores y del balón, el análisis del fútbol ha pasado de métricas meramente descriptivas a modelos complejos que evalúan las acciones de los jugadores, predicen resultados y simulan estrategias de partido (Daniel Carrilho ; Micael Santos Couceiro; João Brito ; Pedro Figueiredo ; Rui J. Lopes ; Duarte Araújo). El aprovechamiento del aprendizaje automático (Machine Learning) y la inteligencia artificial (IA) ha permitido obtener una comprensión más profunda del comportamiento de los jugadores, las formaciones tácticas y la dinámica de los equipos, revolucionando la forma en que se analizan los encuentros y se desarrollan las estrategias (Amadu, 2024).

2.1.2.2 Evolución hacia enfoques basados en datos y aprendizaje profundo

La disponibilidad creciente de datos granulares sobre eventos de juego ha catalizado el desarrollo de metodologías predictivas más sofisticadas fundamentadas en técnicas de aprendizaje automático. Los algoritmos de clasificación supervisada, incluyendo bosques aleatorios, máquinas de vectores de soporte y métodos de ensamble, han demostrado capacidad para capturar relaciones no lineales complejas entre múltiples variables predictoras y los resultados de partidos (Rory Bunker, Calvin Yeung, Keisuke Fujii, 2024). Las redes neuronales profundas representan la frontera actual en modelización predictiva deportiva. Estas arquitecturas multicapa pueden aprender representaciones jerárquicas de características, identificando automáticamente patrones relevantes sin requerir ingeniería manual exhaustiva de variables. Las redes neuronales recurrentes, específicamente diseñadas para procesar secuencias temporales, resultan particularmente apropiadas para capturar dinámicas evolutivas del rendimiento de equipos a lo largo de temporadas competitivas (Nallapa, 2022).

2.1.2.3 Limitaciones de los Modelos Tradicionales

Los enfoques predictivos tradicionales presentan limitaciones metodológicas significativas derivadas de sus supuestos fundamentales que fueron construidos. Los

modelos basados en estadísticas agregadas asumen implícitamente estacionariedad en el rendimiento de equipos, ignorando fluctuaciones debidas a cambios en plantillas, lesiones, variaciones en forma física o modificaciones tácticas implementadas por cuerpos técnicos. Esta asunción resulta particularmente problemática en competiciones extensas donde la composición y estrategia de equipos evoluciona sustancialmente (Spyridon Plakias ; Themistoklis Tsatalas ; Xenofon Betsios; Giannis Giakas, 2025). La mayoría de modelos tradicionales desestiman factores contextuales que la evidencia empírica sugiere que ejercen influencia significativa sobre resultados. Variables como condiciones meteorológicas adversas, altitud del estadio, fatiga acumulada por calendarios congestionados, importancia relativa del encuentro dentro de la competición y presión psicológica asociada a derbi locales o partidos decisivos raramente se incorporan en formulaciones clásicas. La naturaleza inherentemente estocástica del fútbol impone un límite fundamental a la precisión alcanzable mediante cualquier sistema predictivo. Eventos de baja probabilidad, pero alto impacto, como errores arbitrales controvertidos, expulsiones tempranas o lesiones inesperadas durante el partido, pueden alterar radicalmente el desarrollo y resultado de encuentros de manera difícilmente predecible a priori (Mazi Essoloani Aleza; D. Vetrithangam, 2023). Esta irreductibilidad estocástica sugiere que incluso los modelos más sofisticados alcanzarán precisiones modestas en términos absolutos.

2.1.3 Factores Ambientales y su Influencia en el Rendimiento Futbolístico

2.1.3.1 Variables meteorológicas relevantes

- Temperatura y humedad relativa: Las condiciones ambientales de temperatura y humedad constituyen determinantes primarios del estrés térmico experimentado por deportistas durante actividad física intensa. Las temperaturas ambientales elevadas intensifican los procesos de deshidratación y aceleran la aparición de fatiga prematura, reduciendo la capacidad aeróbica y aumentando el riesgo de lesiones musculares y golpes de calor. Paralelamente, los niveles elevados de humedad relativa comprometen la eficiencia de los mecanismos termorreguladores

del organismo, obstaculizando la disipación de calor corporal mediante evaporación cutánea (Jhonny Francisco Segovia Romero; Joseph Taro, 2025) (Philo U Saunders ; David B Pyne ; Christopher J Gore, 2009).

- Precipitación: Las condiciones pluviométricas modifican sustancialmente las características de la superficie de juego, afectando tanto el comportamiento del balón como la biomecánica de los desplazamientos de jugadores. La presencia de agua en el césped reduce el coeficiente de fricción, incrementando la velocidad de rodamiento del balón y dificultando el control técnico en recepciones y conducciones. Simultáneamente, la saturación del terreno aumenta el riesgo de resbalones y caídas, modificando patrones de movimiento y potencialmente incrementando la incidencia lesional.
- Velocidad y dirección del viento: Las condiciones anemométricas ejercen influencia directa sobre la trayectoria y velocidad del balón, particularmente en pases largos, centros y disparos a distancia. Vientos con velocidades superiores pueden desviar significativamente las trayectorias balísticas, introduciendo incertidumbre adicional en la ejecución técnica. Este efecto resulta especialmente relevante en estadios descubiertos sin protección perimetral que mitigue la exposición al viento (Sungchan Hong ; Ryosuke Nobori, 2016).
- Presión atmosférica y altitud: La altitud sobre el nivel del mar constituye una variable ambiental particularmente crítica en el contexto geográfico andino. La disminución progresiva de la presión barométrica en función de la altura genera una reducción proporcional en la presión parcial de oxígeno atmosférico, fenómeno que compromete la difusión alveolar de oxígeno y, consecuentemente, la saturación de oxígeno en hemoglobina. Esta disminución de oxígeno en el organismo debido a la reducida presión desencadena adaptaciones fisiológicas agudas que incluyen incremento de frecuencia respiratoria y cardíaca, reducción del volumen sistólico y

disminución de la capacidad aeróbica máxima (Sarah Illmer; Frank Daumann, 2022) (Ronaldo Kobal ; Irineu Loturco, 2022).

2.1.3.2 Efectos fisiológicos en el jugador

La exposición a condiciones ambientales adversas activas respuestas fisiológicas compensatorias que pueden comprometer el rendimiento deportivo cuando superan la capacidad adaptativa del organismo. En ambientes calurosos, el incremento del flujo sanguíneo cutáneo para facilitar la disipación térmica compite con las demandas metabólicas de la musculatura activa, resultando en una reducción de la capacidad de trabajo físico. La deshidratación progresiva, evidenciada mediante pérdidas hídricas del peso corporal, deteriora tanto el rendimiento físico como las funciones cognitivas relevantes para la toma de decisiones tácticas (Jhonny Francisco Segovia Romero; Joseph Taro, 2025).

La hipoxia de altitud induce adaptaciones hematológicas agudas, incluyendo incremento en la síntesis de eritropoyetina y consecuente estimulación de la eritropoyesis. Sin embargo, estas adaptaciones requieren períodos de aclimatación de varias semanas para desarrollarse completamente. En ausencia de aclimatación adecuada, la exposición aguda a altitudes superiores a 2500 metros resulta en deterioro significativo de la capacidad aeróbica, manifestándose en reducción de la velocidad de carrera, menor distancia total recorrida y tiempos de recuperación prolongados entre esfuerzos de alta intensidad (Ronaldo Kobal ; Irineu Loturco, 2022).

2.1.3.3 Impacto en la táctica, precisión y ritmo del partido

Las condiciones ambientales no solamente afectan las capacidades físicas individuales, sino que también modulan aspectos tácticos y estratégicos del juego colectivo. En condiciones de temperatura elevada, se observa típicamente una reducción en el ritmo general del partido, menor densidad de acciones de alta intensidad y modificaciones en patrones de posesión tendientes a economizar gasto energético. Los equipos tienden a adoptar estrategias más conservadoras, priorizando control de posesión sobre presión intensiva constante. La precisión técnica en pases y disparos puede verse

comprometida por condiciones meteorológicas adversas. Terrenos de juego saturados por lluvia incrementan la variabilidad en los rebotes del balón, dificultando la anticipación y control. El viento introduce incertidumbre adicional en trayectorias aéreas, reduciendo la efectividad de centros laterales y disparos lejanos. Estas perturbaciones pueden favorecer estrategias basadas en juego directo y transiciones rápidas sobre elaboración prolongada mediante pases cortos (Sarah Illmer; Frank Daumann, 2022). La altitud modifica sustancialmente las propiedades físicas del balón y su comportamiento dinámico. La menor densidad del aire en altura reduce la resistencia aerodinámica, incrementando la velocidad de desplazamiento y modificando trayectorias de manera menos predecible. Este fenómeno genera parábolas más extendidas en pases largos y disparos a distancia, demandando ajustes técnicos y tácticos específicos por parte de jugadores y cuerpos técnicos (Sungchan Hong ; Ryosuke Nobori, 2016).

2.1.4 Fuentes de Datos y Herramientas

2.1.4.1 Plataformas de Estadísticas de Fútbol

Los datos de rastreo posicional, obtenidos mediante sistemas ópticos multicámara o dispositivos GPS portátiles, capturan las coordenadas espaciales de todos los jugadores y el balón con frecuencias de muestreo. Esta información permite cuantificar métricas físicas como distancias recorridas, velocidades máximas, aceleraciones, desaceleraciones y mapas de calor que visualizan zonas de mayor actividad. Adicionalmente, facilita análisis tácticos sofisticados como formaciones dinámicas, amplitud del equipo, profundidad ofensiva y coordinación de líneas (Wilton W Fok; Louis C Chan; Carol Chen, 2018). Los datos de desempeño físico, obtenidos mediante sensores inerciales integrados en indumentaria especializada, registran variables fisiológicas y biomecánicas como frecuencia cardíaca, carga metabólica, asimetrías en patrones de carrera y distribución de impactos. Esta información resulta valiosa para monitorear estados de fatiga, prevenir lesiones y personalizar programas de entrenamiento (Daniel Carrilho ; Micael Santos Couceiro; João Brito ; Pedro Figueiredo ; Rui J. Lopes ; Duarte Araújo).

Los principales proveedores de datos deportivos se diferencian por su enfoque y alcance algunos garantizan alta precisión mediante codificación manual de eventos por analistas especializados, aunque su cobertura se limita a competiciones de élite. Otros combinan datos de eventos con video sincronizado, facilitando análisis contextualizados y comparaciones internacionales, siendo útil para scouting de jugadores. También ofrecen datos bajo licencias académicas y ofreciendo métricas avanzadas como el valor esperado de gol (xG) basado en modelos estadísticos contextuales. Por su parte, nuestra fuente de datos actúa como agregador global de información, proporcionando acceso sistemático a estadísticas a través de su API, aunque con limitaciones en granularidad temporal y en detalle de eventos individuales

2.1.4.2 NASA POWER

Fundamentos técnicos del sistema:

El proyecto POWER (Prediction Of Worldwide Energy Resources) de la NASA constituye una fuente particularmente adecuada para investigación que requiere datos climáticos con cobertura global y consistencia temporal. El sistema integra información de múltiples fuentes satelitales y modelos atmosféricos para generar estimaciones de variables meteorológicas con resolución temporal horaria y resolución espacial de aproximadamente 0.5 grados de latitud-longitud (equivalente a aproximadamente 55 km en el ecuador) (Center, 2025). Los productos de datos POWER se fundamentan principalmente en MERRA-2 (Modern-Era Retrospective analysis for Research and Applications, Version 2), un sistema de reanálisis atmosférico desarrollado por el Global Modeling and Assimilation Office de la NASA. MERRA-2 asimila observaciones satelitales y de estaciones terrestres en un modelo numérico de predicción meteorológica, generando campos meteorológicos espacialmente completos y físicamente consistentes que cubren el período desde 1980 hasta el presente. Complementariamente, POWER incorpora procesamiento específico de datos satelitales para variables relacionadas con radiación solar y propiedades de nubes, derivados del proyecto CERES (Clouds and the Earth’s Radiant Energy System) (Center, 2025). Esta combinación de fuentes permite ofrecer un

conjunto comprehensivo de variables relevantes para aplicaciones que requieren caracterización detallada de condiciones atmosféricas.

Variables climáticas disponibles y su pertinencia:

El sistema POWER provee acceso a más de 200 parámetros meteorológicos y solares, incluyendo variables directamente relevantes para análisis de rendimiento deportivo. Entre las variables de mayor pertinencia se encuentran la temperatura del aire a 2 metros de altura (T2M), que representa condiciones térmicas experimentadas por individuos en superficie; la humedad relativa a 2 metros (RH2M), indicativa del contenido de vapor de agua atmosférico; la velocidad y dirección del viento a 2 y 10 metros (WS2M, WD2M, WS10M, WD10M), la precipitación total corregida (PRECTOT-CORR), que incluye ajustes por subestimación sistemática en productos satelitales; y la cobertura nubosa (CLOUD_AMT). Adicionalmente, POWER ofrece variables radiactivas como la radiación solar incidente en superficie (ALLSKY_SFC_SW_DWN), relevante para evaluar exposición a radiación ultravioleta y cargas térmicas radiactivas (Center, 2025). Estas variables se distribuyen con resolución temporal horaria, permitiendo sincronización precisa con horarios de eventos deportivos y captura de variaciones diurnas en condiciones meteorológicas.

Ventajas metodológicas para investigación:

La utilización de datos POWER ofrece ventajas metodológicas significativas para investigación científica. La cobertura global permite estudios comparativos entre regiones geográficas diversas sin restricciones de disponibilidad de estaciones meteorológicas locales. Las series históricas extensas, facilitan análisis retrospectivos comprehensivos y permiten controlar variabilidad climática interanual. La consistencia metodológica derivada del uso de sistemas de reanálisis que asimilan múltiples fuentes observacionales en un marco físico consistente minimiza discontinuidades temporales y valores anómalos artificiales. Esta propiedad resulta particularmente valiosa para análisis de series temporales donde discontinuidades metodológicas pueden introducir artefactos que confunden señales reales. La accesibilidad mediante API pública permite automatización

completa de procesos de descarga, facilitando la replicabilidad de estudios y la actualización continua de bases de datos. La ausencia de costos de licenciamiento democratiza el acceso a información climática de calidad para investigación académica, contrastando con fuentes comerciales que imponen barreras económicas significativas.

Limitaciones y consideraciones:

La resolución espacial de aproximadamente 50 km implica que las estimaciones corresponden a promedios sobre áreas considerables, pudiendo no capturar microclimas locales o efectos topográficos de pequeña escala. En contextos de topografía compleja, como las zonas andinas del Perú, esta limitación puede resultar en divergencias entre condiciones estimadas y las efectivamente experimentadas en el estadio específico. Los datos POWER, al derivarse de productos modelados que combinan observaciones directas con ecuaciones físicas, incorporan incertidumbre inherente. Variables como la precipitación, particularmente difíciles de observar desde satélite sobre superficies continentales, pueden presentar mayores errores que variables directamente medidas como la temperatura. El sistema utiliza el valor -999 como indicador de dato faltante, requiriendo manejo apropiado en análisis estadístico para evitar interpretaciones erróneas.

La latencia en la disponibilidad de datos representa otra consideración relevante. Aunque los productos POWER se actualizan regularmente, puede existir un desfase temporal de varios días entre la ocurrencia de condiciones meteorológicas y su disponibilidad en el sistema. Esta limitación resulta menos relevante para estudios retrospectivos, pero puede restringir aplicaciones que requieren información en tiempo casi real.

2.1.5 Aprendizaje Automático y Aprendizaje Profundo

Los algoritmos de aprendizaje automático operan mediante la identificación de patrones estadísticos y estructuras latentes en conjuntos de datos, utilizando estos patrones para realizar inferencias o predicciones sobre observaciones no contempladas durante el proceso de entrenamiento. El aprendizaje supervisado representa el paradigma más directamente aplicable a problemas de predicción deportiva. Este enfoque utiliza

conjuntos de datos etiquetados donde cada observación comprende tanto las variables predictoras como la variable objetivo conocida. El algoritmo aprende una función de mapeo que relaciona entradas con salidas, optimizando sus parámetros para minimizar el error entre predicciones y valores reales observados. Los problemas de clasificación, donde se predice una etiqueta categórica y los problemas de regresión, donde se estima un valor continuo, constituyen las dos vertientes principales del aprendizaje supervisado (Huyen, 2022).

2.1.5.1 Selección de características

La selección de características constituye un proceso fundamental en el desarrollo de modelos de aprendizaje automático que busca identificar el subconjunto óptimo de variables predictoras que maximizan el desempeño del modelo mientras minimizan la complejidad computacional y el riesgo de sobreajuste. En dominios con alta dimensionalidad, como el análisis deportivo donde múltiples estadísticas y variables contextuales pueden registrarse, la inclusión indiscriminada de todas las características disponibles introduce riesgos sustanciales: incremento de requisitos computacionales, mayor propensión al sobreajuste debido a la captura de correlaciones espurias, y degradación de la interpretabilidad del modelo. La selección hacia adelante (forward selection) comienza con un conjunto vacío, agregando secuencialmente la característica que más mejora el desempeño. La eliminación hacia atrás (backward elimination) inicia con todas las características, removiendo iterativamente aquella cuya ausencia menos degrada el desempeño. Aunque estos métodos consideran la relevancia específica para el modelo empleado, resultan computacionalmente costosos al requerir múltiples ciclos de entrenamiento (Alice Zheng; Amanda Casari, 2018).

2.1.5.2 Proceso de modelado

El desarrollo de modelos de aprendizaje automático sigue un flujo estructurado que comienza con el preprocesamiento de datos. Esta etapa comprende limpieza de valores faltantes o anómalos, normalización o estandarización de variables para homogenizar escalas, codificación de variables categóricas en representaciones numéricas, y

potencialmente ingeniería de características para crear variables derivadas que capturen relaciones relevantes. La fase de entrenamiento emplea algoritmos de optimización que ajustan iterativamente los parámetros del modelo para minimizar una función de pérdida que cuantifica la discrepancia entre predicciones y valores reales. La elección del algoritmo de optimización (gradiente descendente estocástico, Adam, RMSprop) y la configuración de hiper parámetros (tasa de aprendizaje, tamaño de lote, número de épocas) influyen significativamente en la convergencia y desempeño final del modelo (Huyen, 2022). La validación mediante conjuntos de datos independientes permite evaluar la capacidad de generalización del modelo a datos no vistos durante el entrenamiento. La detección temprana de sobreajuste, donde el modelo memoriza patrones específicos del conjunto de entrenamiento sin capturar relaciones generalizables, constituye un aspecto crítico de esta fase. La evaluación final emplea métricas cuantitativas apropiadas al tipo de problema. Para clasificación multiclase, métricas como precisión, recall, F1-score y matrices de confusión permiten caracterizar el desempeño diferenciado en cada categoría. Para regresión, métricas como error cuadrático medio, error absoluto medio y coeficiente de determinación R2 cuantifican la precisión de las estimaciones numéricas (Huyen, 2022).

2.1.5.3 Redes Neuronales Recurrentes (RNN)

Las redes neuronales recurrentes representan una clase especializada de arquitecturas diseñadas específicamente para procesar datos secuenciales donde el orden temporal o espacial contiene información relevante. A diferencia de las redes feed forward que asumen independencia entre observaciones, las RNN incorporan conexiones cíclicas que permiten mantener un estado interno o memoria que captura información de pasos temporales anteriores.

2.1.5.3.1 Concepto de dependencia temporal

La dependencia temporal surge cuando el valor apropiado de una predicción en un momento determinado depende no solamente de las características observadas en ese instante, sino también del contexto proporcionado por observaciones previas (Wilton W Fok; Louis C Chan; Carol Chen, 2018). En el análisis deportivo, esta propiedad resulta

fundamental, el resultado esperado de un partido depende no únicamente de las estadísticas actuales de los equipos, sino también de su trayectoria reciente, secuencias de victorias o derrotas, y evolución de forma física a lo largo de la temporada. Las RNN abordan esta dependencia temporal mediante un mecanismo de estado oculto que se actualiza en cada paso temporal, integrando información de la entrada actual con el estado heredado del paso anterior. Esta recursión permite, en principio, que la red capture dependencias temporales de longitud arbitraria, aunque en la práctica las RNN tradicionales presentan dificultades para aprender dependencias muy extensas (Roger Grosse ; Jimmy Ba's, 2017).

2.1.5.3.2 Arquitecturas de Redes Neuronales

Las redes neuronales artificiales constituyen sistemas computacionales inspirados en la arquitectura del sistema nervioso biológico, diseñadas para reconocer patrones complejos mediante el procesamiento jerárquico de información. La unidad fundamental, la neurona artificial, implementa una transformación no lineal de una combinación ponderada de sus entradas, emulando conceptualmente el comportamiento de neuronas biológicas que integran señales sinápticas y generan potenciales de acción.

Una red neuronal típica organiza neuronas en capas diferenciadas funcionalmente. La capa de entrada recibe las características o variables predictoras, codificándolas en representaciones numéricas apropiadas. Las capas ocultas ejecutan transformaciones sucesivas de la información mediante operaciones matriciales ponderadas seguidas de funciones de activación no lineales. La capa de salida genera las predicciones finales, adaptándose a la naturaleza del problema (activación softmax para clasificación multiclase, activación lineal para regresión). Cada neurona calcula una suma ponderada de sus entradas más un término de sesgo, aplicando posteriormente una función de activación que introduce no linealidad. Funciones de activación comúnmente empleadas incluyen la tangente hiperbólica (tanh), la unidad lineal rectificada (ReLU) y sus variantes (Leaky ReLU, ELU). La elección de la función de activación influye tanto en la capacidad expresiva

de la red como en la eficiencia del entrenamiento (Ian Goodfellow ; Yoshua Bengio ; Aaron Courville, 2016).

2.1.5.3.3 Retro propagación y optimización

El entrenamiento de redes neuronales emplea el algoritmo de retro propagación, que calcula eficientemente los gradientes de la función de pérdida respecto a todos los parámetros mediante aplicación repetida de la regla de la cadena del cálculo diferencial. Estos gradientes dirigen la actualización de pesos mediante algoritmos de optimización basados en descenso de gradiente, que ajustan iterativamente los parámetros en la dirección que reduce la función de pérdida. Los optimizadores adaptativos modernos, como Adam (Adaptive Moment Estimation) y RMSprop, mantienen estimaciones de primer y segundo momento de los gradientes, ajustando dinámicamente las tasas de aprendizaje por parámetro. Esta adaptabilidad acelera la convergencia y mejora la robustez frente a hiper parámetros mal configurados, aunque requiere memoria adicional para almacenar los momentos (Ian Goodfellow ; Yoshua Bengio ; Aaron Courville, 2016).

2.1.5.3.4 Regularización y prevención de sobreajuste

El sobreajuste constituye un desafío fundamental en redes neuronales profundas, manifestándose cuando el modelo desarrolla representaciones excesivamente específicas a los datos de entrenamiento que no generalizan a datos nuevos. Múltiples estrategias de regularización mitigan este problema. El dropout desactiva aleatoriamente una fracción de neuronas durante el entrenamiento, forzando la red a aprender representaciones redundantes más robustas. La parada temprana interrumpe el entrenamiento cuando el desempeño en el conjunto de validación comienza a deteriorarse, previniendo ajuste excesivo a ruido en los datos de entrenamiento (Roger Grosse ; Jimmy Ba's, 2017). La augmentación de datos, cuando resulta aplicable al dominio específico, incrementa artificialmente el tamaño efectivo del conjunto de entrenamiento mediante transformaciones que preservan la etiqueta correcta. En contextos deportivos, esto podría incluir reflexiones de posiciones en el campo o agregación de ruido calibrado a estadísticas numéricas.

2.1.5.3.5 Problema del desvanecimiento y explosión del gradiente:

El entrenamiento de RNN mediante retro propagación temporal enfrenta desafíos significativos de la propagación de gradientes a través de muchos pasos temporales. Durante la retro propagación, los gradientes se multiplican repetidamente por las mismas matrices de pesos, resultando en dos fenómenos patológicos. El desvanecimiento del gradiente ocurre cuando productos repetidos de valores menores que uno conducen a gradientes exponencialmente decrecientes que efectivamente anulan la señal de aprendizaje para dependencias largas. Conversamente, la explosión del gradiente surge cuando productos de valores mayores que uno generan gradientes exponencialmente crecientes que desestabilizan el proceso de optimización (Roger Grosse ; Jimmy Ba's, 2017).

2.1.5.3.6 Long Short-Term Memory (LSTM)

Las redes LSTM, fueron diseñadas específicamente para resolver el problema del desvanecimiento del gradiente mediante la introducción de una arquitectura de celda de memoria con mecanismos de control de flujo de información. La innovación fundamental radica en la separación entre el estado de celda (memoria a largo plazo) y el estado oculto (salida a corto plazo), junto con un sistema de puertas que regulan selectivamente qué información se retiene, actualiza o descarta.

Esta arquitectura modular permite que las LSTM mantengan información relevante durante períodos temporales extensos, evitando el desvanecimiento del gradiente mediante el flujo constante de información a través del estado de celda. La capacidad resultante para capturar dependencias a largo plazo ha consolidado las LSTM como arquitectura preferida para múltiples aplicaciones de procesamiento secuencial (Wilton W Fok; Louis C Chan; Carol Chen, 2018).

2.1.5.3.6 Gated Recurrent Unit (GRU)

Las unidades GRU, representan una simplificación arquitectónica de las LSTM que mantiene su capacidad de modelar dependencias a largo plazo mientras reduce la complejidad computacional. La innovación principal consiste en la consolidación de las

puertas de entrada y olvido en una única puerta de actualización, eliminando además la distinción entre estado de celda y estado oculto (Ariana Yunita ; MHD Iqbal Pratama, 2025). La arquitectura GRU implementa dos mecanismos de control.

Gracias a esta estructura compacta, la GRU logra un desempeño comparable al de la LSTM con menor complejidad computacional y menos parámetros a entrenar (Ariana Yunita ; MHD Iqbal Pratama, 2025).

2.1.5.4 Árboles de Decisión

Los algoritmos basados en árboles de decisión constituyen una familia de métodos de aprendizaje supervisado que modelan relaciones entre variables mediante estructuras jerárquicas de decisiones sucesivas. Estos algoritmos poseen propiedades particularmente valiosas para análisis de características: capacidad innata para manejar no linealidades e interacciones complejas sin transformaciones explícitas, robustez ante variables en diferentes escalas sin requerir normalización, y generación natural de métricas de importancia de características.

Un árbol de decisión particiona recursivamente el espacio de características mediante reglas de decisión binarias, construyendo una estructura jerárquica donde cada nodo interno representa una prueba sobre una característica específica, cada rama corresponde al resultado de esa prueba, y cada nodo hoja asigna una etiqueta de clase o valor de regresión. El proceso de construcción emplea algoritmos greedy que seleccionan en cada paso la partición que maximiza la pureza de los subconjuntos resultantes. Para clasificación, criterios como el índice de Gini o la entropía de Shannon cuantifican la homogeneidad de clases en cada nodo (Alice Zheng; Amanda Casari, 2018).

Los árboles de decisión individuales tienden a desarrollar estructuras profundas que memorizan el conjunto de entrenamiento, manifestando alto sobreajuste. Técnicas de poda limitan este problema removiendo ramas que proporcionan mejoras marginales insuficientes o estableciendo restricciones sobre profundidad máxima, número mínimo de instancias por nodo, o ganancia mínima requerida para particionar.

2.1.5.5 Métricas y Evaluación de Modelos Multiclase

Métricas fundamentales:

La evaluación de modelos de clasificación multiclase requiere métricas que capturen diferentes aspectos del desempeño predictivo. La exactitud (accuracy) representa la proporción de predicciones correctas sobre el total de observaciones, constituyendo la métrica más intuitiva pero potencialmente engañosa en presencia de desbalance de clases (Huyen, 2022).

La precisión (precisión) cuantifica la proporción de predicciones positivas que resultan correctas para cada clase, respondiendo a la pregunta: "De todos los casos que el modelo predijo como clase *k*, ¿cuántos realmente pertenecían a esa clase?" Formalmente, para la clase *k*:

$$\text{Precisión}_k = \frac{VP_k}{VP_k + FP_k}$$

Donde VP_k denota los verdaderos positivos y FP_k los falsos positivos para la clase k

El **recall** (sensibilidad o exhaustividad) mide la proporción de instancias reales de cada clase que el modelo identifica correctamente, respondiendo: “¿De todos los casos reales de la clase *k*, ¿cuántos identificó correctamente el modelo?”

$$Recall_k = \frac{VP_k}{VP_k + FN_k}$$

Donde FN_k denota los falsos negativos.

El **F1-score** combina precisión y recall mediante su media armónica, proporcionando una métrica balanceada particularmente útil cuando ambos aspectos resultan igualmente importantes:

$$F1_k = 2\,\frac{\text{Precisión}_k . Recall_k}{\text{Precisión}_k + Recall_k}$$

Para obtener métricas globales en problemas multiclase, se emplean esquemas de agregación. El promedio macro calcula la métrica independientemente para cada clase y promedia sin ponderación, tratando todas las clases equitativamente. El promedio

ponderado (weighted) pondera las métricas por clase según su frecuencia en el conjunto de datos, resultando más representativo del desempeño global cuando las clases presentan tamaños desiguales. La matriz de confusión proporciona una visualización comprehensiva del desempeño, mostrando en cada celda (i, j) el número de instancias de la clase verdadera i que fueron predichas como clase j. El análisis de esta matriz revela patrones de confusión específicos, como si el modelo confunde sistemáticamente empates con victorias del equipo visitante, información valiosa para interpretar limitaciones del modelo (Huyen, 2022).

Manejo del desbalance de clases:

El desbalance de clases, donde ciertas categorías (típicamente empates en fútbol) ocurren con frecuencia sustancialmente menor que otras, introduce sesgos que degradan el desempeño de modelos entrenados con funciones de pérdida estándar. Múltiples estrategias abordan este problema. La asignación de pesos de clase (class weights) modifica la función de pérdida para penalizar más severamente errores en clases minoritarias. Durante el entrenamiento, el error asociado a cada ejemplo se multiplica por un factor inversamente proporcional a la frecuencia de su clase. En frameworks como TensorFlow y PyTorch.

El oversampling (sobre muestreo) de clases minoritarias replica instancias de estas categorías para balancear artificialmente la distribución. Técnicas sofisticadas como SMOTE (Synthetic Minority Over-sampling Technique) generan ejemplos sintéticos mediante interpolación entre instancias existentes de la clase minoritaria, aumentando la diversidad del conjunto aumentado.

El submuestreo (under sampling) reduce el número de instancias de clases mayoritarias para equilibrar la distribución. Aunque simple, esta técnica descarta información potencialmente valiosa, pudiendo degradar el desempeño cuando los datos resultan limitados. La focal loss, introducida constituye una modificación de la función de pérdida de entropía cruzada que reduce automáticamente el peso de ejemplos bien clasificados, concentrando el aprendizaje en casos difíciles.

Esta formulación resulta particularmente efectiva en escenarios de desbalance extremo sin requerir ponderaciones manuales. La evaluación estratificada asegura que los conjuntos de entrenamiento, validación y prueba mantengan proporciones similares de cada clase, previniendo que conjuntos de evaluación pequeños resulten dominados por clases específicas y proporcionando estimaciones más estables del desempeño.

2.2. Antecedentes de estudio

(Stevens, 2024)**. Predicting the outcome of Women's World Cup matches taking weather conditions into account, using K-Nearest Neighbors, Random Forest and Support Vector Machines. Tilburg University, Paises Bajos**

Conclusiones:

Esta tesis aborda la predicción de resultados en partidos de la Copa Mundial Femenina de la FIFA mediante la integración de variables meteorológicas, utilizando tres técnicas de Aprendizaje Automático (ML): K-Nearest Neighbors (KNN), Random Forest y Support Vector Machines (SVM). La investigación utilizó un conjunto de datos que abarca resultados de partidos, clasificaciones FIFA y condiciones meteorológicas (sensación térmica, viento, humedad) de los Mundiales Femeninos desde 2011 (Alemania), 2015 (Canadá), 2019 (Francia) y 2023 (Australia/Nueva Zelanda). El objetivo principal fue determinar el impacto de incluir las condiciones climáticas como una característica adicional en la capacidad predictiva de los modelos. Respecto a la variable meteorológica, el hallazgo central fue que la inclusión de las condiciones climáticas no mejoró la precisión predictiva del modelo de Aprendizaje Automático KNN. La precisión en el conjunto de prueba se mantuvo en 0.65 cuando se incluyó la sensación térmica. En un análisis secundario para evaluar si el impacto era diferente para los equipos europeos en comparación con los no europeos, la mayor precisión se obtuvo en el modelo para equipos no europeos sin la inclusión de la variable meteorológica (0.615), y los resultados para los equipos europeos con o sin datos meteorológicos fueron inferiores o comparables, lo que no sustenta la hipótesis de que el clima influya significativamente en el resultado del partido en este contexto. El autor atribuye la falta de impacto del clima a dos limitaciones

principales: el tamaño reducido del conjunto de datos (solo cuatro Mundiales) y la imprecisión de los datos meteorológicos, que fueron medidos a nivel de país o ciudad y no en la ubicación específica del estadio.

Comentario: Este documento es metodológicamente relevante, ya que aplica técnicas de ML similares directamente al problema de la predicción de resultados de fútbol internacional, incluyendo específicamente el factor climático. El resultado central, la ausencia de una mejora significativa en la precisión al incorporar la sensación térmica, es un hallazgo empírico crucial para la investigación propuesta. Este resultado plantea la hipótesis nula para el uso de variables meteorológicas en contextos de ML para predicción de partidos internacionales, lo que obliga a la investigación con RNN a justificar y explorar por qué su metodología o conjunto de datos podría arrojar resultados diferentes. La crítica metodológica del autor sobre la granularidad de los datos climáticos (medición a nivel de país en lugar de estadio) es vital. Esto sugiere que para que las RNN detecten una señal climática, es imperativo utilizar datos meteorológicos más precisos y localizados, un desafío que debe abordarse si la investigación actual busca superar las limitaciones observadas por este autor.

(Ditsuhi Iskandaryan , Francisco Ramos, 2020),The effect of weather in soccer results: an approach using machine learning techniques. Universitat Jaume I, España

Conclusiones:

Esta investigación determina el efecto de las condiciones climáticas en los resultados de partidos de fútbol mediante la implementación de técnicas de Aprendizaje Automático (ML), analizando datos de LaLiga y la Segunda División española de las temporadas 2013-2014 a 2017-2018. El estudio se estructuró en dos tareas de clasificación: Multivariante (Predecir victoria local, victoria visitante o empate) y Bivariante (Predecir empate o no empate). Utilizo algoritmos de ML como Random Forest (RF), Support Vector Machines (SVM), K-Nearest Neighbors (KNN) y Extra-Trees. Los datos de

fútbol se complementaron con información meteorológica altamente localizada, obtenida de 775 estaciones cercanas a los 25 estadios utilizados. Las variables climáticas consideradas incluyen temperatura máxima, mínima y media, ráfagas de viento, velocidad máxima del viento y precipitación, además de características derivadas como las diferencias entre las condiciones del equipo local y el visitante. Los resultados demostraron una precisión significativamente mayor en la predicción de resultados en comparación con los modelos que excluían estas características. Para el Caso de Estudio 1, el clasificador $Extra-Trees$ fue superior, alcanzando una precisión del 65.9 % con datos meteorológicos, frente a solo 53.3 % sin ellos (para RF, que fue el mejor sin datos climáticos). Para el Caso de Estudio 2, SVM fue el más eficiente con una precisión del 79.3 % con datos meteorológicos. Se concluye que la inclusión de datos meteorológicos es útil para predecir el resultado de un partido de fútbol, siendo la diferencia de temperatura promedio ($Tmed\_Diff$) y la diferencia máxima de velocidad del viento ($Vmax\_Diff$) algunas de las características finales más relevantes.

Comentario: Este trabajo es fundamental porque proporciona una evidencia empírica directa y cuantitativa de que las condiciones meteorológicas sí pueden mejorar significativamente la precisión de la predicción de resultados de fútbol cuando se utilizan técnicas de ML. Metodológicamente, se diferencia del estudio de Stevens por su contexto (liga doméstica vs. Copa Mundial) y, crucialmente, por la granularidad de sus datos. El uso de 775 estaciones meteorológicas en España para cubrir 25 estadios minimiza el problema de la imprecisión de la ubicación del partido. Esta diferencia en la recolección de datos sugiere que la señal climática es detectable solo cuando se mide con precisión, lo que tiene implicaciones directas para la investigación con RNN, la cual deberá priorizar la calidad espacial de los datos meteorológicos. Este documento apoya la premisa de trabajo de la investigación propuesta (que las variables meteoro-lógicas influyen en el resultado) y justifica la exploración de modelos avanzados (como las RNN) para capturar estas correlaciones de manera más efectiva.

(Niek Tax; Niek Tax, 2015)**. Predicting The Dutch Football Competition Using Public Data: A Machine Learning Approach. Universidad de Twente, Paises bajos**

Conclusiones:

El estudio identificó una amplia gama de factores con valor predictivo (rendimiento histórico, rachas, cambio de DT, ventaja de localía, fatiga y distancia de viaje). El modelo con datos públicos alcanzó una precisión máxima de 54.702 % (Naive Bayes o Multilayer Perceptron combinado con PCA). Al igual que en otros estudios, los modelos tuvieron serias dificultades para predecir el empate, ya que las características utilizadas no ofrecían valor predictivo para esta clase minoritaria. La combinación de datos públicos con probabilidades de apuestas (modelo híbrido) mejoró ligeramente la precisión al 56.054 %, sugiriendo que las probabilidades incluyen factores no capturados por los datos públicos.

Comentario: La metodología empleada, que incluye una revisión sistemática de factores y un enfoque retrodictivo que parte por el objetivo final construyendo un plan hacia atrás para el entrenamiento, es directamente aplicable a la investigación con RNN, dado el carácter temporal de los datos climáticos. Aunque el estudio no incluye variables meteorológicas, sí considera el factor "Fatiga", que se modela a partir de la dureza del partido anterior y el tiempo transcurrido desde el último encuentro. Las condiciones meteorológicas extremas (calor, humedad) son conocidas por inducir fatiga y estrés fisiológico, lo que sugiere que la meteorología es una variable cuantificable que complementa el análisis de rendimiento físico y la fatiga, justificando su inclusión para refinar la capacidad predictiva. Además, los autores excluyeron datos difíciles de recuperar automáticamente, lo que implica que las variables meteorológicas deben ser rigurosamente cuantificables para ser útiles.

(Walker J. Ross; Madeleine Orr, 2022)**. Predicting climate impacts to the Olympic Games and FIFA Men’s World Cups from 2022 to 2032. Sport in Society. Universidad de Edimburgo, Reino Unido.**

Conclusiones:

Este estudio se centra en la proyección de las condiciones climáticas y de calidad del aire para los mega eventos deportivos programados entre 2022 y 2032, incluyendo los Juegos Olímpicos y la Copa Mundial Masculina de la FIFA. El trabajo establece condiciones límite ambientales críticas para garantizar la seguridad y la integridad competitiva, el trabajo establece criterios para evaluar los riesgos ambientales en eventos deportivos mediante umbrales. Fútbol (FIFA World Cups): La Copa Mundial de Qatar 2022 fue reprogramada a noviembre/diciembre debido al calor extremo, aunque las temperaturas históricas sugieren que casi todos los días aún superan el umbral de riesgo. La región enfrentará probablemente un aumento de días calurosos y olas de calor en los años posteriores, comprometiendo el uso de las instalaciones si no se mantienen las adaptaciones (como el aire acondicionado en estadios). Para la Copa Mundial UNITED 2026 (Canadá, EE. UU., México), el calor extremo será la principal preocupación en casi todas las ciudades anfitrionas. Juegos Olímpicos: Beijing 2022 enfrentó mala calidad del aire e insuficientes temperaturas frías para la nieve natural. El estudio concluye enfatizando la necesidad de que los organizadores creen eventos, infraestructura y legados resilientes al clima, implementando planes de contingencia para proteger a los atletas y mantener la integridad competitiva.

Comentario: Este documento establece un marco conceptual robusto, la ecología del deporte, al postular una relación bidireccional entre el deporte y el medio ambiente: el impacto del deporte en el clima y el impacto del clima en la operación deportiva. Su relevancia para la investigación propuesta reside en dos aspectos: 1. Cuantificación de Variables de Riesgo: Proporciona umbrales de riesgo cuantitativos específicos. Estos límites definidos por especialistas pueden ser utilizados para categorizar los datos meteorológicos de entrada en el modelo RNN, transformando variables continuas en variables categóricas o binarias de riesgo. 2. Contexto de Mega eventos: Documenta que el calor extremo es un factor disruptivo recurrente en el fútbol a nivel de Copa Mundial. Si las condiciones de calor extremo obligan a los jugadores a adoptar estrategias de ritmo o aumentan el riesgo de enfermedades relacionadas con el calor, esta variable debe ser

inherentemente predictiva de cambios en el rendimiento y, potencialmente, en el resultado final.

(Jimenez, 2023) **Sistema para pronosticar resultados de partidos de futbol en opciones dobles. Universidad de Lima, Perú.**

Conclusiones:

El proyecto desarrolló un sistema de pronóstico para la liga peruana, enfocado en maximizar la probabilidad de acierto utilizando la modalidad de opciones dobles como Local Gana o Empata. El sistema logró un 82 % de acierto para las recomendaciones de mayor peso (PRO-PESO = 7). El modelo busca una estrategia de ganancia incremental a largo plazo (multiplicador semanal de 1.20 a 1.25). En cuanto a las variables utilizadas, el sistema se basa en resultados históricos, tablas de posiciones y estadísticas agregadas.

Comentario: La relevancia principal radica en la exclusión explícita de variables contextuales. El diseño del sistema no contempló "Factores climatológicos de las ciudades donde se efectúa el partido de fútbol", ni el "Factor emocional de los jugadores", ni las lesiones. Esta exclusión, frecuente en los sistemas de predicción que se basan en datos deportivos estándar, justifica plenamente la necesidad de integrar y cuantificar la influencia de estas variables exógenas. Se cuenta con detalles de la implementación del sistema, mas no sobre el modelo de predicciones por lo que no se tiene detalles de la arquitectura ni experimentos que realizo. Además, la recomendación para futuros trabajos incluye la incorporación del aprendizaje de máquina para refinar las sugerencias de combinaciones, lo que valida la exploración de modelos avanzados como las RNN para la clasificación y predicción de resultados.

(Bustos, 2023) **Sistema de Predicción de Resultados para los Partidos de Futbol de la Liga Profesional Colombiana. Universitaria de Bogotá Jorge Tadeo Lozano, Colombia.**

Conclusiones:

La investigación evaluó la capacidad predictiva de Redes Neuronales Artificiales (ANN), Máquinas de Soporte Vectorial (SVM), Árboles de Decisión (DT) y el Sistema de

Clasificación ELO para la liga colombiana. Los resultados de precisión obtenidos fueron limitados para la clasificación de tres clases (gana local, empate, gana visitante), con el mejor rendimiento logrado por el modelo ELO (43 % de accuracy) y DT (42 % de accuracy). La RNN mostró el rendimiento más bajo (34 % de accuracy), lo que sugiere que, con las variables históricas utilizadas, este modelo no logró capturar patrones efectivos. La conclusión central es que la precisión no superó el 45 %, lo que demanda la necesidad de incorporar un conjunto más amplio de variables para robustecer la predicción.

Comentario: Este documento es de capital importancia, ya que aborda directamente la implementación de un algoritmo de Redes Neuronales Recurrentes en la predicción de partidos de fútbol. La baja precisión del modelo (34 % de accuracy) sugiere que las variables de entrada tradicionales (resultados históricos, goles, posición) son insuficientes en este contexto. La principal relevancia radica en la justificación de la introducción de nuevas variables, ya que los autores recomiendan explícitamente explorar y evaluar la inclusión de más variables en el análisis". El bajo rendimiento de la RNN con datos estadísticos estándar justifica la hipótesis de que la inclusión de factores externos no correlacionados, como las variables meteorológicas, podría ser la clave para que la RNN desarrolle un poder discriminatorio superior.

(FUENTEALBA, 2025) **Predicción de Resultados de Partidos de la Liga Profesional de Futbol Chileno usando Algoritmos de Machine Learning. Universidad del Desarrollo, Chile.**

Conclusiones:

El proyecto reafirma la complejidad inherente a la predicción de resultados de fútbol, siendo la clase empate la más difícil de identificar, una limitación que persiste incluso tras aplicar técnicas avanzadas de balanceo de clases (SMOTE, ADASYN) y optimización de hiper parámetros. Los modelos utilizados y métrica de precisión son: Random Forest (0.3693), XGBoost (0.4156), CatBoost (0.4140) y Regresión Logística (0.3740) mostraron un rendimiento modesto, manteniéndose cerca del nivel de un clasificador aleatorio. Este patrón se replicó en la Premier League, confirmando que el desafío es intrínseco al

dominio, no a una liga específica. Se concluye que se requiere la incorporación de "nuevas fuentes de información más ricas (variables contextuales, tácticas, calidad individual de jugadores, condiciones externas)"para lograr una mejora significativa.

Comentario: Este estudio provee la justificación más explícita y directa para la investigación propuesta. La mención de condiciones externas como un requisito para mejorar el desempeño predictivo valida directamente la inclusión de variables meteorológicas. El fracaso consistente de los modelos en predecir el empate es crucial. Dado que los métodos basados en árboles y regresión logística no lograron superar este obstáculo, se justifica la exploración de modelos RNN, los cuales son más adecuados para manejar datos secuenciales y relaciones no lineales sutiles, buscando precisamente la señal climática que podría diferenciar los resultados en condiciones extremas.

CAPITULO III

Metodología

3.1. Enfoque metodológico

La metodología empleada en este estudio se inscribe en un enfoque cuantitativo, utiliza la estadística como herramienta.

3.1.1. Diseño de investigación

La investigación tiene un nivel descriptivo según (Hernández, Fernández, & Baptista, 2014) Estudios descriptivos Busca especificar propiedades y características importantes de cualquier fenómeno que se analice. Describe tendencias de un grupo o población. (p.92)

3.1.2 Nivel de investigación

Los estudios descriptivos pretenden especificar las propiedades, características y perfiles de personas, grupos, comunidades, procesos, objetos o cualquier otro fenómeno que se someta a un análisis. (Hernandez-Sampieri & Mendoza, 2019) Es decir, miden o recolectan datos y reportan información sobre diversos conceptos, variables, aspectos, dimensiones o componentes del fenómeno o problema a investigar (p.108)

3.1.3. Población

Nuestra población está constituida por datos estadísticos de partidos de futbol de las ligas profesionales de Colombia Chile Ecuador y Peru “Es el conjunto de todos los elementos (unidades de analsis9 que pertenecen al ámbito espacial donde se desarrolla el trabajo de investigación” (Carrasco, 2019)

3.1.4. Muestra

Se ha considerado como muestra a los datos de la stemporadas del 2017 al 2025 por su parte (Supo, 2024) El tamaño de la muestra “está determinado por el nivel de precisión que deseamos para los resultados y las conclusiones, mientras mayor sea el tamaño de la muestra tendremos mayor precisión y mientras menos precisión se exija, menos tamaño tendrá la muestra” (p161).

3.1.5. Muestreo por conveniencia

El muestreo es por conveniencia en vista que la disponibilidad de datos estadísticos de partidos de futbol en un nivel detallado se encuentra en las fechas disponibles (Supo, 2024) Recibe también el nombre muestreo deliberado, porque no cuenta con ningún procedimiento estandarizado, ninguna acción especifica que realzar, ni razón más que la comodidad o única oportunidad de muestrear; en suma, no hay ninguna forma de seleccionar la muestra, es simplemente deliberado (p. 190)

3.1.6. Diseño

El diseño corresponde al experimental puesto que se ha realizado mediciones a los resultados del entrenamiento de aprendizaje automático. Según (Bernal, 2010). En la investigación experimental existen diversos tipos de diseño, que se clasifican de diferentes formas. Sin embargo, la clasificación más usada, según Salkind (1998) e investigadores como Briones (1985), es la de Campbell y Stanley, quienes identifican tres categorías generales de diseños de investigación: preexperimentales, cuasi experimentales y experimentales verdaderos.

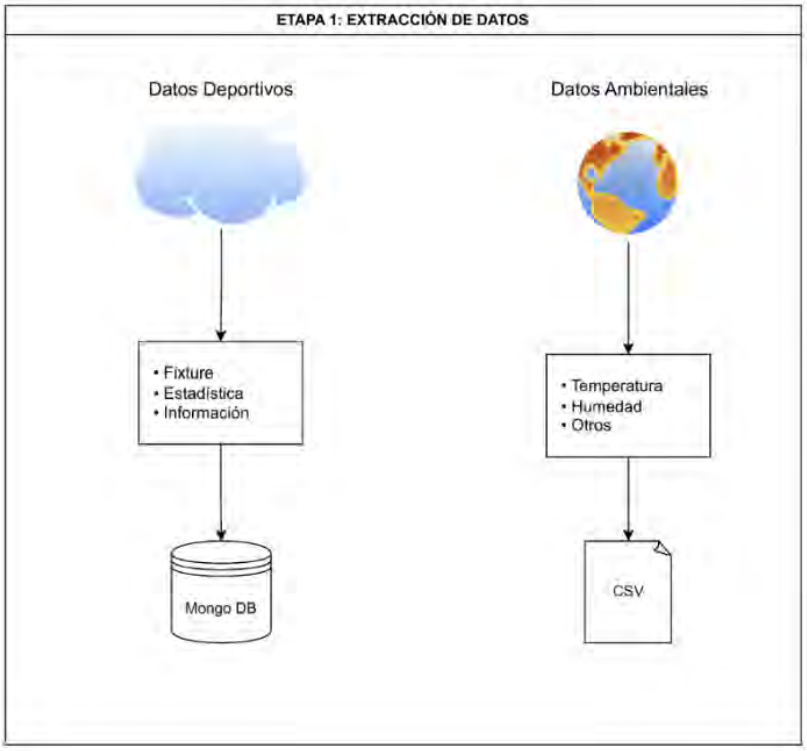
3.2. Etapas del proceso metodológico

3.2.1. Etapa 1 Extracción de Datos

Esta etapa es el punto de partida para la recolección de un conjunto diverso de datos, que incluye estadísticas detalladas de fútbol como información adicional, calendarios de partidos y alineaciones correspondientes a las ligas profesionales de futbol de Chile, Colombia, Ecuador y Perú. La recopilación abarca desde el 3 de febrero de 2017, fecha del primer partido de esas ligas en dicho año, hasta el 19 de mayo de 2025, cuando concluyó la extracción tanto de los datos futbolísticos. Estas fechas de calendario de partidos se cruza con los datos de condiciones ambientales proporcionadas por la NASA POWER API. La información deportiva se almacenó en una base de datos no relacional MongoDB, mientras que los datos ambientales se guardaron en archivos CSV, tal como se aprecia en la figura 1.

Figura 1

Etapa 1 Extracción de datos

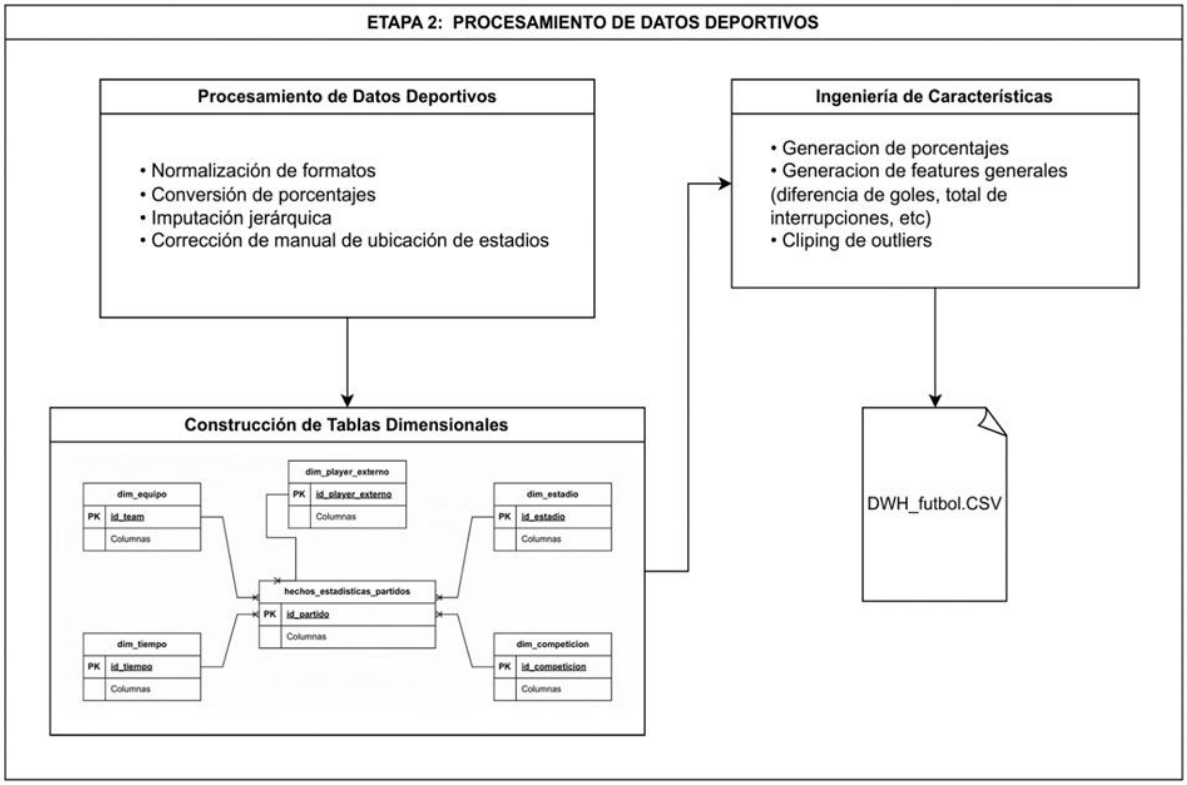


3.3.2. Etapa 2 Procesamiento de Datos de Futbol

Ejecuta un proceso de transformación multietapa que incluye la normalización de formatos inconsistentes, la implementación de un esquema de imputación jerárquica que aprovecha similitudes contextuales entre equipos, competiciones y el conjunto global, así como la aplicación de ingeniería de características, donde se obtiene un total de 44 variables, esta etapa podemos resumir en la figura 2

Figura 2

Etapa 2: Procesamiento de Datos Deportivos

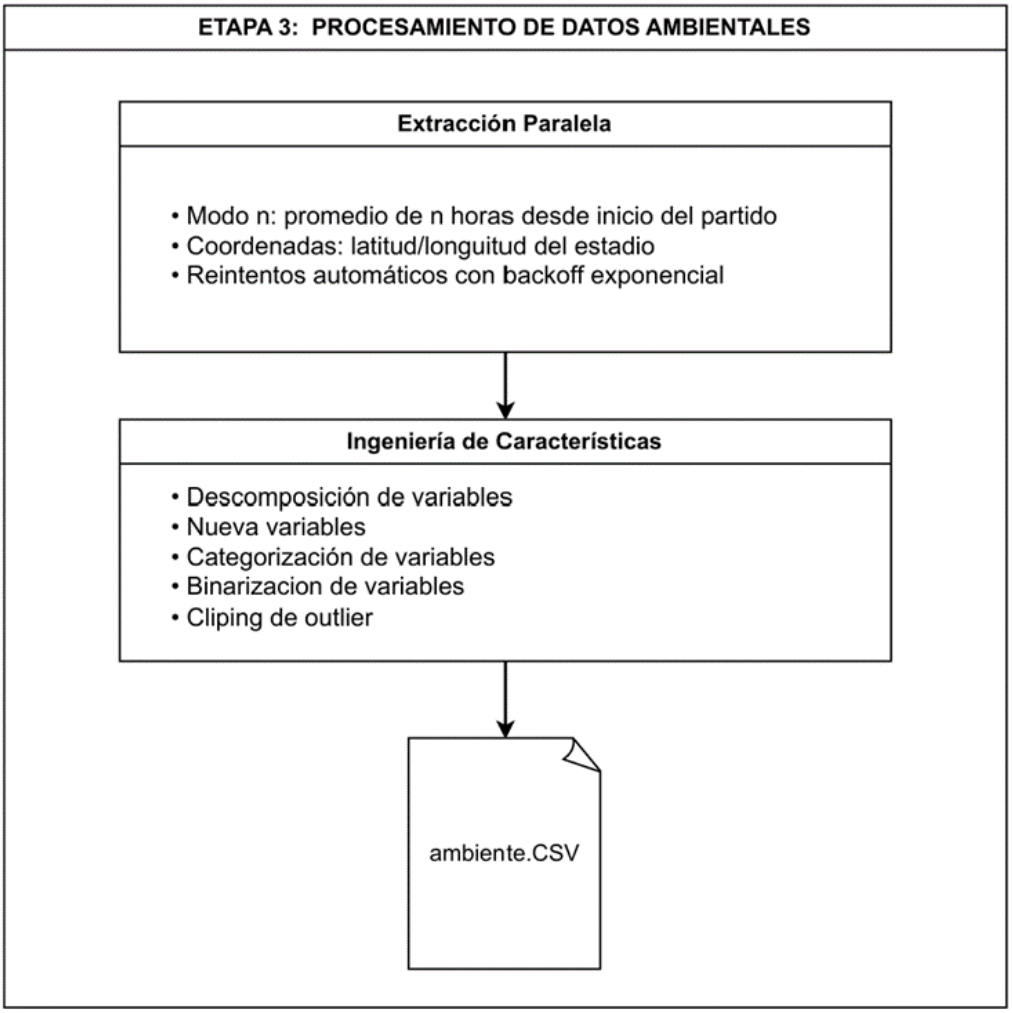


3.3.3. Etapa 3 Procesamiento de Datos de Ambiente

Procesa específicamente las variables meteorológicas mediante extracción paralela con gestión de reintentos, calculando promedios temporales de tres horas durante cada encuentro, descomponiendo vectorialmente el viento en componentes cartesianas y generando variables categóricas que identifican condiciones climáticas adversas según umbrales físicamente fundamentados, como se muestra en la figura 3.

Figura 3

Etapa 3: Procesamiento de Datos Ambientales



3.3.4. Etapa 4 Preparación para Modelado

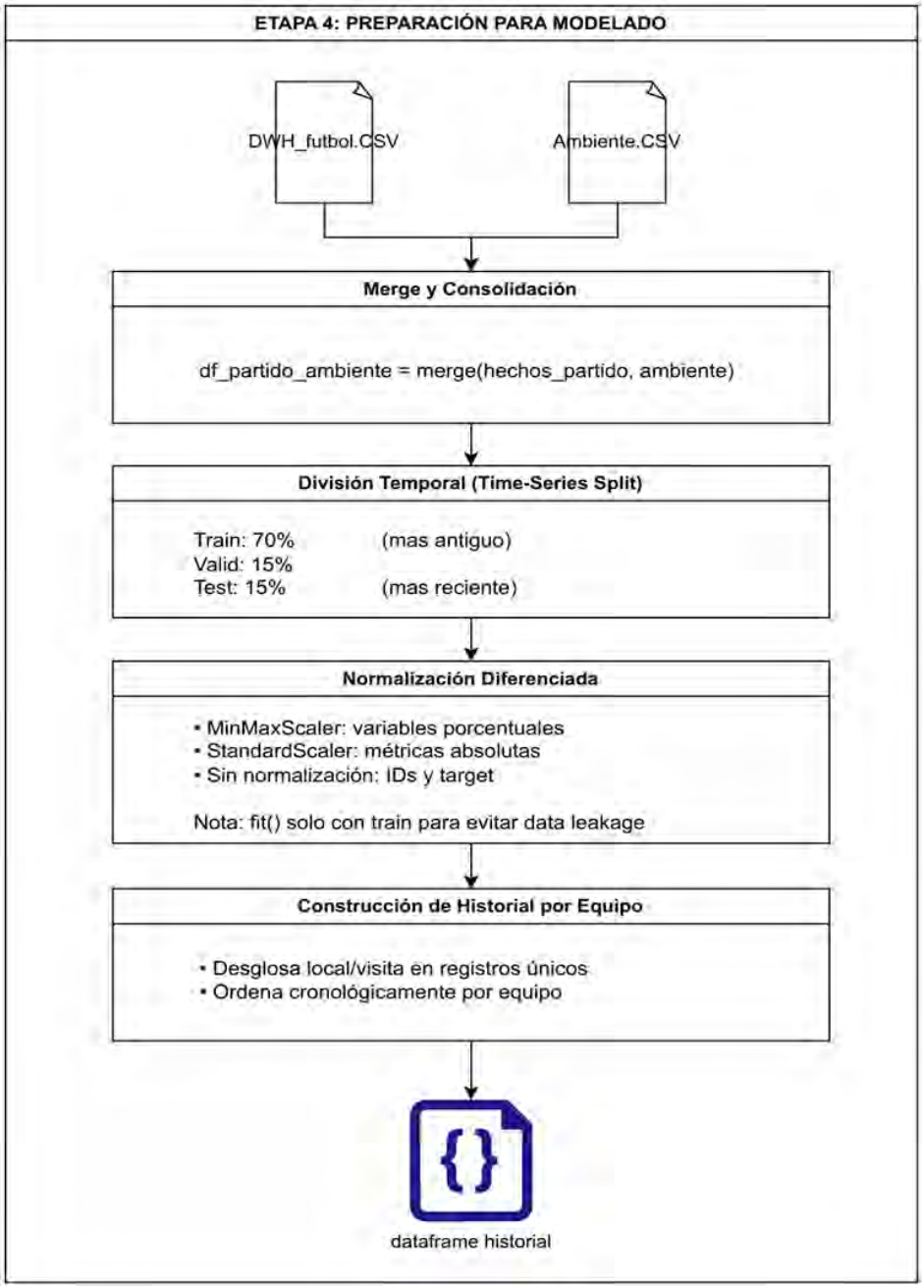
Consolida ambos conjuntos de datos mediante operaciones de cruce de datos mediante espacio temporal basadas en coordenadas de estadio y marcas temporales, implementa una división cronológica estricta que preserva la estructura temporal (70 % entrenamiento, 15 % validación, 15 % prueba) y aplica normalización diferenciada según la naturaleza estadística de cada variable, ajustando exclusivamente sobre el conjunto de entrenamiento para evitar filtración de información, como se muestra en la figura 4.

El conjunto de entrenamiento está compuesto por 6 563 registros, desde el 2017-02-03 22:45:00 hasta el 2023-03-04 23:15:00; el conjunto de validación incluye 1 406 registros, desde el 2023-03-05 00:00:00 hasta el 2024-03-30 17:15:00; y el conjunto de prueba contiene 1 408 registros, con un intervalo que va desde el 2024-03-30 18:15:00

hasta el 2025-05-19 01:15:00. Todos en formato de horas UTC conformando un total de 9377 partidos.

Figura 4

Etapa 4: Preparación para Modelado



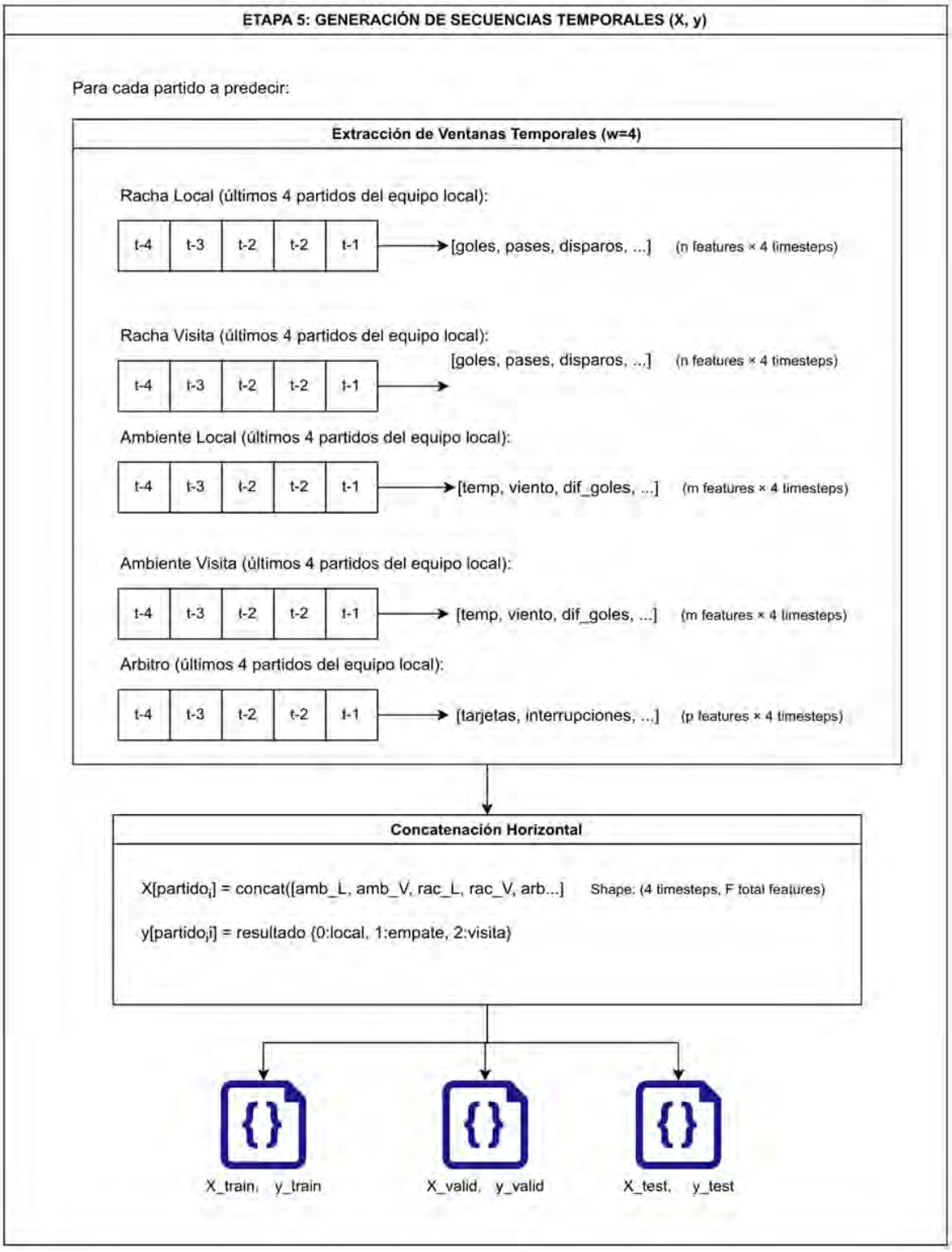
3.3.5. Etapa 5 Generación de Secuencias Temporales (X, y)

Transforma los datos tabulares en secuencias temporales tridimensionales mediante la extracción de ventanas deslizantes de cuatro partidos previos, concatenando horizontalmente las rachas de rendimiento de ambos equipos, las condiciones ambientales

históricas asociadas a cada equipo en su condición local/visitante y el comportamiento histórico del árbitro asignado, generando tensores con dimensiones *(n_muestras, 4 timesteps, F características),* como se muestra en la figura 5.

Figura 5

Etapa 5 Generación de Secuencias Temporales (X, y)



3.3.6. Etapa 6 Importancia Relativa de Características

Tenemos construido los dataframes que contiene estadísticas deportivas y ambientales, se procede a implementar Random Forest, este modelo nos ayuda con la interpretabilidad ya que mediante la métrica Gini obtenemos la importancia de cada una de las variables del dataframe. Ya que estamos en un problema de series temporales, para simular o replicar una ventana de tiempo procedemos a ingresar al modelo el promedio de las estadísticas de los últimos 4 partidos, para esta etapa nos basamos en la etapa 5 que obtiene las ventanas de tiempo, pero para este caso ingresa promediado.

3.3.7. Etapa 7 Optimización de hiper parámetros (Grid Search)

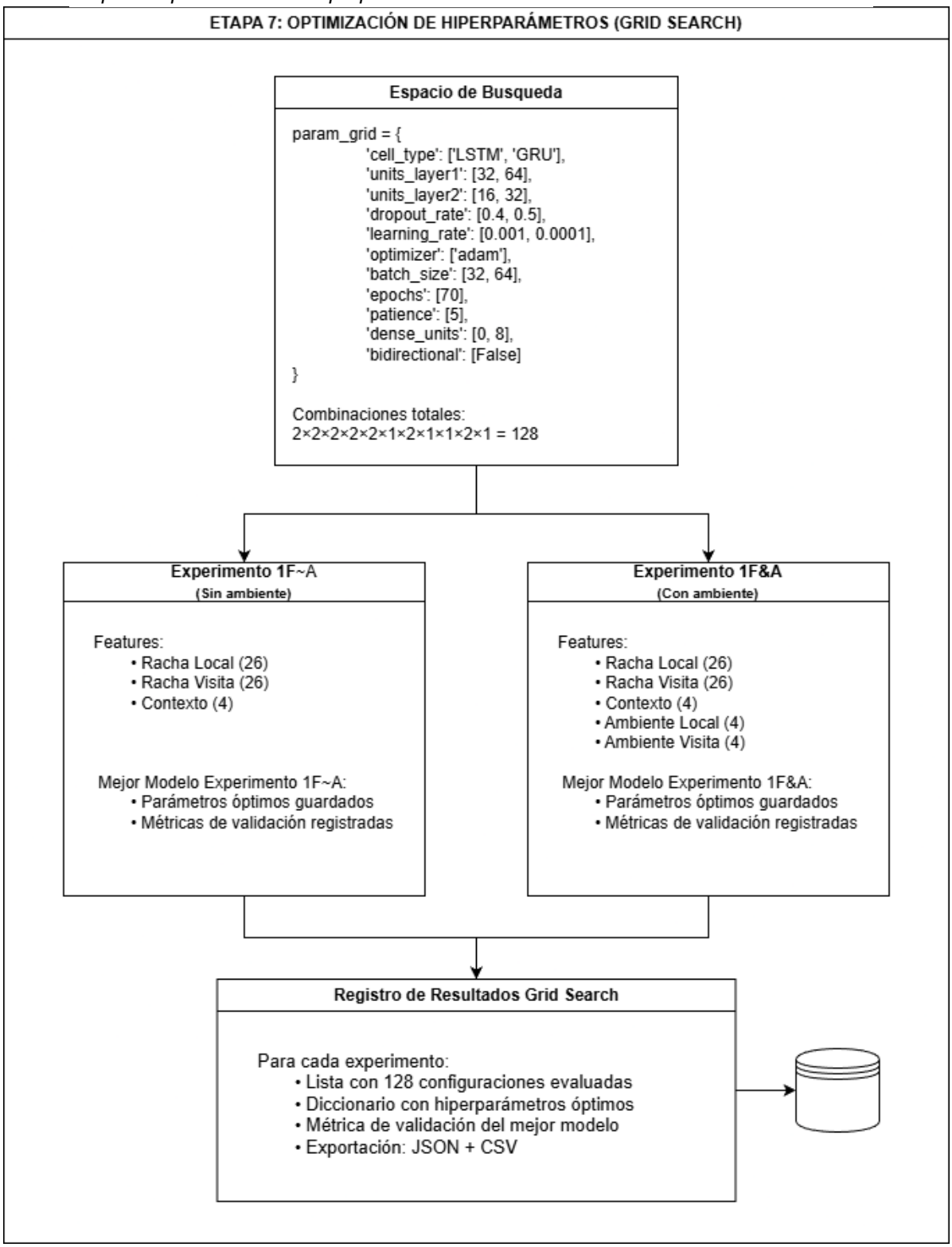
Para el desarrollo del presente estudio, se seleccionaron las Redes Neuronales Recurrentes (RNN), específicamente las arquitecturas LSTM y GRU. Se descartaron los modelos de Machine Learning convencionales debido a su incapacidad para modelar de forma nativa las dependencias temporales largas, tales como las rachas de victorias o derrotas de los equipos. Como señala la literatura especializada, estas arquitecturas con mecanismos de memoria son fundamentales para capitalizar la información histórica, permitiendo que el modelo aprenda patrones evolutivos que los modelos estáticos ignorarían.

Implementa una estrategia de optimización exhaustiva mediante Grid Search sobre un espacio de 128 configuraciones de hiper parámetros, evaluando arquitecturas LSTM y GRU con diferentes profundidades, tasas de dropout, optimizadores y tamaños de lote.

Este proceso se ejecuta de forma independiente para dos experimentos: Experimento "F-1” utilizando únicamente 28 características deportivas y de árbitro, y Experimento "F-A1” incorporando 16 características ambientales adicionales (8 por equipo), identificando para cada uno la configuración óptima según métricas de validación, como se muestra en la figura 6.

Figura 6

Etapa 6: Optimización de hiperparametros



3.3.8. Etapa 8 Evaluación de Resultados

Se realiza una evaluación estratificada geográficamente, desagregando el conjunto de prueba según el país de origen de la competición y calculando métricas de rendimiento individuales para cada estrato. También se realiza un análisis comparativo sistemático entre ambos experimentos, contrastando rendimientos globales y estratificados, documentando exhaustivamente las condiciones bajo las cuales la incorporación de información meteorológica resulta beneficiosa, neutral o contraproducente para la capacidad predictiva del modelo.

CAPITULO IV

Desarrollo

Este capítulo describe el proceso de desarrollo e implementación empleado para la recopilación, procesamiento y análisis de datos deportivos y meteorológicos en el contexto de esta investigación. El desarrollo experimental siguió una estructura basada en la metodología CRISP-DM, adaptada a las particularidades de los datos deportivos y ambientales. El proceso comprende diferentes etapas de procesamiento de datos, organizadas en capas de madurez (bronce, plata y oro) para el procesamiento de datos deportivos y para los datos meteorológicos, que transforman datos crudos en información estructurada lista para el modelado predictivo.

4.1. Enfoque General

La estrategia metodológica de esta investigación se fundamenta en la integración de dos líneas de datos complementarias: estadísticas deportivas de partidos de fútbol y variables meteorológicas. El objetivo principal consiste en construir un modelo predictivo capaz de anticipar los resultados de encuentros deportivos, considerando tanto el rendimiento histórico de los equipos como las condiciones ambientales durante los partidos.

El proceso de desarrollo e implementación se estructura en tres capas de procesamiento, siguiendo una arquitectura de lakehouse basada en el modelo medallón:

- **Capa Bronce**: Almacenamiento de datos crudos extraídos directamente de las fuentes
- **Capa Plata**: Transformación y normalización de datos en estructuras relacionales
- **Capa Oro**: Preparación final de datos con ingeniería de características para análisis

Esta arquitectura permite mantener la trazabilidad completa de los datos desde su origen hasta su uso en modelos predictivos, facilitando la reproducibilidad del proceso y la identificación de inconsistencias en etapas tempranas.

4.2. Extracción de Datos Deportivos (Capa Bronce)

4.2.1. Selección de Fuentes de Datos

La identificación de fuentes confiables de datos deportivos representó el primer desafío metodológico. Aunque existen múltiples plataformas que ofrecen estadísticas de fútbol, como Transfermarkt, WhoScored, BFREF y Understat, se seleccionó Sofascore como fuente principal por tres razones fundamentales:

1. Cobertura temporal extensa: Proporciona datos históricos de la liga peruana desde 2008, período significativamente más amplio que el ofrecido por competidores.
2. Granularidad de estadísticas: A partir de 2017, incorporó métricas avanzadas como posiciones promedio de jugadores, mapas de calor y estadísticas detalladas de pases.
3. Disponibilidad de API no documentada: Aunque no cuenta con documentación oficial, su arquitectura REST permite el acceso programático mediante ingeniería inversa.

La selección de Sofascore implicó aceptar ciertas limitaciones en términos de disponibilidad de datos para jugadores menos conocidos, equipos recién llegados al futbol profesional y partidos no televisados, particularmente en temporadas anteriores a 2017.Se recolecto un conjunto diverso de datos, incluyendo estadísticas detalladas de fútbol, calendarios de partidos y alineaciones de las ligas profesionales de Chile, Colombia, Ecuador y Perú, abarcando desde el 3 de febrero de 2017 hasta el 19 de mayo de 2025.

4.2.2. Diseño del Sistema de Extracción

El sistema de extracción se construyó sobre la librería Botasaurus Driver, que proporciona capacidades de navegación automatizada resistentes a mecanismos anti

robot. La implementación se estructuró en torno a cinco funciones principales de extracción:

1. Extracción de temporadas válidas: Esta función identifica las temporadas disponibles para cada liga mediante consulta al endpoint de temporadas. Retorna un diccionario que mapea años a identificadores internos de temporada, permitiendo la iteración sistemática sobre períodos históricos.

2. Extracción de fixtures: Recupera la lista completa de partidos para una temporada específica mediante paginación incremental. El proceso continúa solicitando páginas hasta recibir una respuesta de error para aplicar técnicas de reintentos, cuando finaliza el proceso finaliza se determina que se han recuperado todos los partidos disponibles.

3. Extracción de información adicional del partido: API que obtiene detalles complementarios de cada encuentro, incluyendo información del estadio, árbitro, directores técnicos y condiciones específicas del partido.

4. Extracción de estadísticas detalladas: API que recupera métricas de rendimiento agrupadas en categorías: visión general del partido, pases, duelos, tiros, defensa y portería. Estas estadísticas se estructuran en formato JSON anidado que posteriormente requiere transformación.

5. Extracción Completa: Integra las 4 funciones anteriores para realizar la extracción de cada uno de los partidos de una determinada temporada de una liga, tal como se aprecia en el algoritmo 1.

4.2.3. Implementación de la Estrategia de Extracción

La función principal extract_ingestion_All orquesta el proceso de extracción completo mediante un diseño de tres bucles anidados:

Algoritmo 1 Extracción de Datos de Ligas y Temporadas

Algorithm 1 Extracción de Datos de Ligas y Temporadas.

```
procedure ExtraccionDatos(L, T, F, P)
    Entrada: L = lista de ligas, T = lista de temporadas válidas, F = fixture
    Salida: Datos ingeridos en MongoDB
    for cada liga l ∈ L do
        for cada temporada válida t ∈ T do
            Extraer fixture completo F(l, t)
            for cada partido p ∈ F(l, t) do
                Extraer información adicional de p
                Extraer estadísticas detalladas de p
            end for
            Ingerir datos en MongoDB
        end for
    end for
    return Datos ingeridos
end procedure
```

Se implementó un retardo de 2 segundos entre solicitudes consecutivas para evitar la activación de mecanismos de limitación de tasa del servidor y se ejecutó el pipeline cada 2 temporadas. Esta decisión, aunque incrementa el tiempo total de extracción, resultó fundamental para mantener la estabilidad del proceso.

4.2.4. Gestión de Restricciones del API

El acceso a la API de Sofascore presentó varios desafíos técnicos que requirieron soluciones específicas:

Limitación de tasa: Los servidores implementan restricciones que bloquean direcciones IP tras detectar patrones de solicitudes automatizadas. Para mitigar este problema, se incorporó la simulación de tráfico legítimo desde el navegador.

Cambios en estructura de datos: La estructura JSON de las respuestas varía según la temporada y liga, particularmente para datos históricos. Se implementaron bloques $try - except$ que capturan excepciones de indexación o claves faltantes, registrando advertencias sin interrumpir el proceso global.

Datos incompletos: No todos los partidos cuentan con información completa, especialmente en competiciones menores o temporadas antiguas. La estrategia adoptada

consistió en almacenar valores nulos cuando los datos no están disponibles, permitiendo su posterior imputación en fases de transformación

4.2.5. Almacenamiento en MongoDB

Los datos extraídos se almacenaron en MongoDB, sistema de base de datos NoSQL orientado a documentos. La elección de MongoDB sobre bases de datos relacionales se fundamentó en:

- Flexibilidad de esquema: Permite almacenar documentos JSON con estructura variable sin necesidad de definir esquemas rígidos previamente
- Consultas eficientes: Soporta índices compuestos que aceleran búsquedas por múltiples campos simultáneamente.
- Escalabilidad horizontal: Facilita la distribución de datos entre múltiples nodos si el volumen de información crece significativamente. Logrando así un aislamiento por país que permite realizar ingestas para nuevos países, esta capacidad permite realizar la re ingesta de datos de un solo país y no realizar el pipeline completo.

  Se crearon tres colecciones principales por liga:

  1. $\{nombre_liga\}$_match: Información básica de partidos
  2. $\{nombre_liga\}$_event: Detalles ampliados de encuentros
  3. $\{nombre_liga\}$_statistics: Estadísticas de rendimiento

Esta estructura de colecciones separadas, aunque introduce cierta redundancia, facilita la verificación de integridad de datos y permite la reconstrucción incremental en caso de fallos parciales del proceso de extracción.

4.2.6. Evaluación de Calidad en Datos Deportivos

La evaluación de calidad de los datos deportivos se realizó siguiendo los principios del DAMA-DMBOK (International, 2017) , adaptados al contexto de datos no estructurados y semiestructurados extraídos de APIs web.

Exactitud: La verificación de exactitud se realizó mediante muestreo aleatorio, comparando los datos extraídos con fuentes alternativas (Transfermarkt, BFREF, ESPN).

Se observó concordancia en resultados finales (goles anotados) y en estadísticas básicas (tiros, córneres, tarjetas). Las discrepancias identificadas se concentraron en métricas derivadas como precisión de pases, posesión de balón donde la definición puede variar entre proveedores. Las coordenadas de estadios presentaron una tasa de error, identificándose tres patrones principales:

-Coordenadas nulas para estadios nuevos

-Coordenadas invertidas (latitud/longitudes intercambiadas)

-Coordenadas incorrectas que situaban estadios fuera de su ubicación real

Completitud: El análisis de valores faltantes reveló patrones heterogéneos según la temporada y variable:

-La completitud mejoró dramáticamente a partir de 2017 coincidiendo con la adopción de sistemas de tracking automático por parte de Sofascore. Los valores nulos en estadísticas básicas correspondieron exclusivamente a partidos de ligas menores o fases eliminatorias no cubiertas.

Consistencia: Se identificaron inconsistencias en el formato de datos que requirieron normalización:

-Variables porcentuales expresadas como texto ("67 %") en lugar de decimales

-Métricas fraccionarias en formato mixto ("12/28 (43 %)")

-Timestamps en múltiples zonas horarias que requirieron estandarización a UTC

Unicidad: No se detectaron duplicados en partidos únicos. Sin embargo, la estructura de documentos anidados en MongoDB introducía redundancia intencional (información de equipos replicada en cada partido) que fue normalizada durante la transformación a modelo dimensional.

Validez: Los valores numéricos se encontraron dentro de rangos esperados. Los casos atípicos identificados correspondieron a partidos con circunstancias excepcionales (expulsiones tempranas, condiciones climáticas extremas) que fueron preservados tras verificación manual.

4.3. Transformación de Datos Deportivos (Capa Plata)

4.3.1. Esquena del Modelo Estrella del Data Warehouse

La transformación de datos crudos almacenados en MongoDB hacia estructuras relacionales siguió un modelo dimensional tipo estrella, compuesto por tablas de dimensiones y una tabla de hechos como se muestra en la figura 7. Este diseño responde a la necesidad de mantener integridad referencial mientras se optimiza el rendimiento de consultas analíticas.

4.3.1.1. Dimensiones

Las dimensiones identificadas fueron:

Dimensión Equipo: Almacena información estática de los equipos participantes, incluyendo identificador único, nombre oficial, código abreviado, país de origen, fecha de fundación y colores representativos. La verificación de existencia.

Dimensión Personal Externo: Agrupa árbitros y directores técnicos bajo una estructura común que diferencia tipos mediante un campo categórico. Esta decisión de diseño reconoce las similitudes estructurales entre ambos roles desde una perspectiva de modelado de datos.

Dimensión Estadio: Contiene información geográfica y de capacidad de los recintos deportivos. Incluye coordenadas geográficas que posteriormente permitirán la integración con datos meteorológicos.

Dimensión Competición: Identifica las ligas y torneos, vinculándolos con el país organizador.

Dimensión Tiempo: Registra información temporal del partido, incluyendo temporada, jornada, timestamp de inicio y tiempo añadido.

4.3.2. Construcción de la Tabla de Hechos

La tabla de hechos constituye el núcleo analítico del modelo, integrando claves foráneas a todas las dimensiones y métricas de rendimiento. Su construcción involucró tres procesos principales:

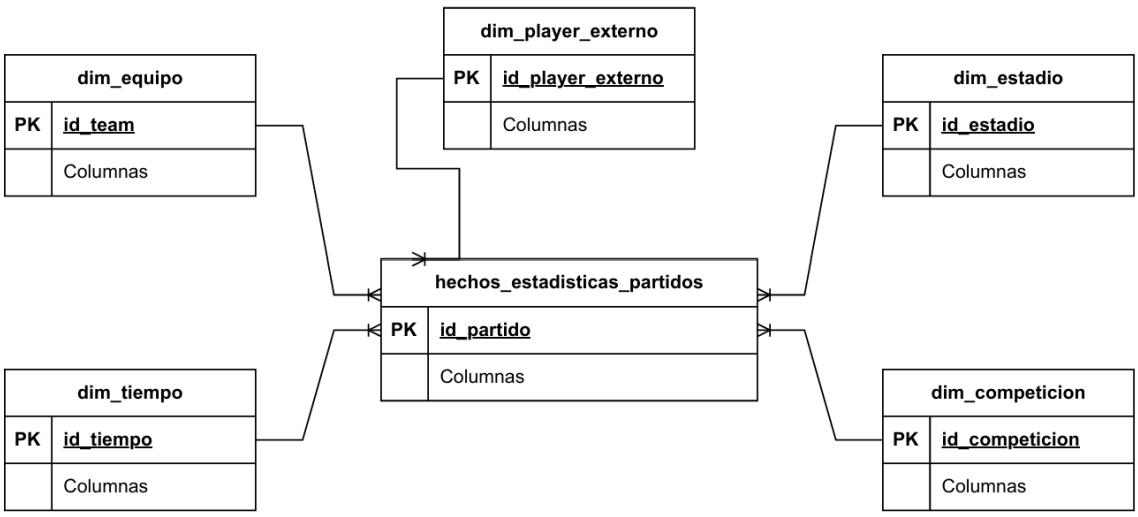
Extracción de valores anidados: Las estadísticas en MongoDB se almacenan en estructuras JSON anidadas donde cada métrica se identifica por una clave textual. Se implementó la función obtener_valor_por_clave que navega estos diccionarios anidados para extraer valores específicos tanto del equipo local como visitante. Esta función maneja gracefully la ausencia de claves, retornando None cuando una estadística no está disponible.

Normalización de métricas: Algunas estadísticas presentan formatos heterogéneos que requieren transformación. Por ejemplo, los duelos aéreos se expresan como "ganados/totales (porcentaje %)", mientras que otros porcentajes aparecen solo como "67 %". La estrategia de normalización se pospuso para la capa oro, almacenando en esta fase los valores tal como aparecen en la fuente.

Gestión de referencias nulas: No todos los partidos cuentan con información completa de árbitros o estadios. El modelo permite valores nulos en claves foráneas no críticas, priorizando la preservación de datos disponibles sobre la integridad referencial estricta.

Figura 7

Modelo Estrella



Transformación

Se realizo la corrección de los estadios quedando los puntos dentro de la cordillera de los Andes como se muestra en la figura 8. Por otra parte, la implementación de

validaciones automáticas detectó 23 registros con inconsistencias lógicas (por ejemplo, disparos al arco superiores a disparos totales), que fueron corregidos asignando el valor de disparos totales a disparos al arco.

Validación y Exportación

Antes de exportar los datos a CSV, se realizaron validaciones de integridad, incluyendo verificación de unicidad de identificadores, conteo de valores nulos, validación de rangos en métricas numéricas y comprobación de claves foráneas. Los datos validados se exportaron a archivos CSV individuales por tabla, formato que facilita la inspección manual y la carga en herramientas de análisis. Esta decisión de diseño, aunque incrementa el número de archivos, mejora la trazabilidad y permite la recarga selectiva de tablas específicas sin afectar el conjunto completo.

Figura 8

Estadios en Colombia, Chile, Ecuador y Perú

4.4. Preparación Final de Datos Deportivos (Capa Oro)

4.4.1. Integración de Dimensiones

La preparación final inició con la integración de las tablas de dimensiones con la tabla de hechos mediante operaciones de combinación secuenciales. Este proceso enriqueció el dataset con información contextual necesaria para el análisis

Se cruza los registros para garantizar que solo se mantengan registros con información completa en todas las dimensiones relevantes.

4.4.2. Transformación de Formatos Heterogéneos

El análisis exploratorio reveló que múltiples columnas contenían valores en formatos no numéricos que impedían su uso directo en modelos predictivos. Se identificaron dos patrones principales:

Formato fracción con porcentaje: Variables como duelos aéreos se expresaban como "12/28 (43 %)", conteniendo simultáneamente valores absolutos y proporción. La transformación aplicada extrajo numerador y denominador mediante expresiones regulares, calculó la proporción manualmente y descartó los valores absolutos para evitar multicolinealidad.

Formato porcentaje simple: Variables como posesión del balón aparecían como "67 %". La transformación consistió en remover el símbolo porcentual y convertir a representación decimal.

Este proceso se aplicó sistemáticamente a variables de duelos aéreos, duelos ganados, balones largos, posesión, centros acertados y regates acertados, tanto para equipos locales como visitantes.

4.4.3. Estrategia de Imputación Jerárquica

La presencia de valores nulos en estadísticas numéricas requirió el diseño de una estrategia de imputación que respetara la estructura jerárquica de los datos deportivos.

El método implementado opera en tres niveles de especificidad decreciente:

- Nivel 1 - Distribución por equipo: Para cada valor faltante, el algoritmo primero intenta construir una distribución empírica basada en los partidos históricos del

equipo específico (local o visitante según corresponda). Si esta distribución contiene al menos 5 observaciones válidas, se realiza un muestreo aleatorio de ella para imputar el valor faltante. Este enfoque preserva las características particulares de rendimiento de cada equipo.

• Nivel 2 - Distribución por competición: Si el equipo carece de suficientes observaciones históricas, el algoritmo recurre a la distribución de la métrica en la competición completa. Esto es particularmente útil para equipos recién ascendidos o con datos históricos limitados.

• Nivel 3 – Distribución global: Como último recurso se utiliza la distribución observada en el dataset completo, independientemente de equipo o competición.

La implementación pre calcula las distribuciones antes de iniciar la imputación para mejorar la eficiencia computacional: Esta estrategia reconoce que las características de juego varían entre equipos y competiciones, evitando la homogenización artificial que produciría una imputación global directa. Un ejemplo es que la liga de Ecuador no posee las mismas distribuciones estadísticas que la liga peruana en una temporada determinada.

4.4.4. Corrección de Datos Geográficos

La validación de las coordenadas de estadios reveló inconsistencias significativas que requerían corrección manual. Los problemas identificados incluyeron:

- Coordenadas invertidas: Algunos estadios presentaban latitud y longitud intercambiadas
- Coordenadas nulas: Estadios nuevos o menos conocidos carecían completamente de información geográfica
- Coordenadas erróneas: Ubicaciones que situaban estadios en océanos o países incorrectos

La corrección se realizó mediante verificación manual en Google Maps y posterior actualización directa en el DataFrame:

Este proceso, aunque laborioso, resultó fundamental para la posterior integración con datos meteorológicos, que dependen críticamente de coordenadas geográficas precisas. Los puntos de los estadios se encuentran situados sobre los países a estudiar que a su vez se encuentra sobre la Codillera de los Andes como se mostró en la figura 8.

4.4.5. Ingeniería de Características

La transformación de variables estadísticas brutas en características predictivas significativas constituyó una fase crucial del preprocesamiento. Se identificaron relaciones funcionales entre variables que, al ser explicitadas, podrían mejorar la capacidad predictiva del modelo. Por ejemplo, se calculó la proporción de pases acertados dividiendo los pases completados exitosamente entre el total de pases intentados, aplicando un término de suavizado pequeño en el denominador para evitar divisiones por cero. Esta misma lógica se extendió a otras métricas como los disparos al arco, donde se calcularon proporciones específicas para disparos dentro del área, fuera del área, bloqueados y dirigidos al marco. Un aspecto relevante surgió durante el análisis de estas proporciones calculadas: algunas excedían el valor unitario, lo cual resulta matemáticamente inconsistente para una proporción genuina. Esta anomalía sugiere posibles errores en los datos fuente o diferencias en los criterios de conteo entre variables relacionadas. Para mantener la coherencia del modelo, se aplicó una función de recorte que limitó estos valores al rango válido entre cero y uno, documentando la frecuencia de estas correcciones para cada variable. La caracterización del estilo arbitral representó otra dimensión importante de la ingeniería de características. Se construyeron variables agregadas que capturan la severidad y el patrón de intervención de cada árbitro. La variable "tarjetas por falta cuantifica la propensión del árbitro a sancionar disciplinariamente, mientras que "total de interrupciones “suma todas las detenciones del juego, incluyendo faltas, tiros libres y saques de banda. Estas métricas permiten al modelo capturar indirectamente el ritmo e intensidad del partido, aspectos que pueden influir en el resultado final.

La construcción de variables derivadas se fundamentó en el conocimiento del dominio futbolístico y la estructura de correlación observada en los datos:

4.4.5.1. *Pases*

Estas características son fundamentales para modelar el ritmo de juego de los equipos en el partido. Los pases representan estrategias de juego como también la asociación de los equipos.

- Porcentaje de pases acertados: El porcentaje de pases acertados mide la precisión de los pases realizados por un equipo en relación con el total de pases intentados. Este indicador es crucial para evaluar la calidad del juego en equipo y la capacidad de los jugadores para mantener la posesión del balón y avanzar en el campo. Un alto porcentaje de pases acertados generalmente refleja un equipo bien organizado y con buena coordinación, mientras que un bajo porcentaje puede indicar problemas en la comunicación, la precisión o el ritmo del juego, a entender cómo un equipo gestiona la posesión del balón, su capacidad para organizar jugadas y su efectividad en la creación de oportunidades.

$$porc\_pases\_acertados\_equipo = \frac{pases\_acertados\_equipo}{pases\_totales\_equipo + \epsilon}$$

4.4.5.2. Disparos

Estas características son fundamentales para modelar la efectividad ofensiva de los equipos en el partido. Los disparos representan oportunidades para marcar goles, y las redes neuronales recurrentes son muy buenas para capturar patrones temporales y secuenciales en estos eventos. Al incluir estas características, el modelo puede aprender cómo el rendimiento ofensivo de un equipo medido a través de la cantidad y calidad de los disparos, evoluciona a lo largo del tiempo y cómo está relacionado con el rendimiento general del equipo en el partido, identifique cuándo un equipo tiene un rendimiento ofensivo fuerte o débil, y cómo las tácticas ofensivas de ambos equipos y sus defensas pueden influir en el resultado del partido.

- **Porcentaje de disparos al arco:** El porcentaje de disparos al arco refleja la efectividad de un equipo al realizar disparos dirigidos al área del portero

contrario. Un valor alto indica que el equipo está disparando con precisión hacia el arco, lo que generalmente es un indicador positivo de su capacidad ofensiva. Esta característica es útil para modelar el rendimiento ofensivo de los equipos, ya que los disparos al arco son uno de los eventos más importantes para generar oportunidades de gol. La RNN puede aprender patrones sobre cómo los disparos al arco afectan la probabilidad de anotar, y cómo el estilo de juego de un equipo se ve reflejado en su habilidad para generar oportunidades claras.

$$porc\_disparos\_arco\_equipo = \frac{disparos\_arco\_equipo}{total\_disparos\_equipo + \epsilon}$$

- Porcentaje de disparos dentro del área: El porcentaje de disparos dentro del área es crucial, ya que los disparos desde el área suelen ser más efectivos debido a la mayor proximidad al arco. Un alto porcentaje de disparos dentro del área indica que el equipo está realizando jugadas más directas y está generando oportunidades más claras de gol. Esta característica ayuda a la RNN a identificar cómo las tácticas ofensivas de un equipo afectan la calidad de sus disparos y su capacidad para crear oportunidades de gol desde zonas más peligrosas.

$$porc\_disparos\_area\_equipo = \frac{disparos\_area\_equipo}{total\_disparos\_equipo + \epsilon}$$

- Porcentaje de disparos fuera del área: El porcentaje de disparos fuera del área mide la tendencia de un equipo a realizar disparos desde larga distancia, los cuales, aunque pueden ser espectaculares, generalmente tienen menos probabilidad de ser efectivos. Esta variable ayuda a la RNN a modelar el comportamiento ofensivo del equipo, especialmente cuando se enfrentan a defensas sólidas o cuando no pueden penetrar en el área contraria. La RNN puede aprender si un equipo tiene éxito o no con disparos de larga distancia y cómo eso afecta sus posibilidades de ganar el partido.

$$porc\_disparos\_fuera\_area\_equipo = \frac{disparos\_fuera\_area\_equipo}{total\_disparos\_equipo + \epsilon}$$

- Porcentaje de disparos bloqueados: El porcentaje de disparos bloqueados refleja cuántos de los disparos realizados por el equipo fueron interceptados o bloqueados por los defensores del equipo contrario. Un alto porcentaje de disparos bloqueados puede indicar que el equipo está teniendo dificultades para superar la defensa contraria, lo que limita su capacidad para crear oportunidades de gol. Esta característica permite que la RNN aprenda cómo la defensa adversaria afecta la capacidad de disparo del equipo y cómo esto influye en el resultado del partido.

$$porc\_disparos\_bloqueados\_equipo = \frac{disparos\_bloqueados\_equipo}{total\_disparos\_equipo + \epsilon}$$

4.4.5.3. Arbitro

Para estas nuevas características no se realiza el cálculo para local y visita ya que representan características globales que surgieron del juego y no corresponde a ningún equipo, pero es producto del enfrentamiento. Las características creadas por el árbitro reflejan decisiones clave durante el partido que afectan el ritmo, la dinámica y la estrategia del juego. Las faltas, tarjetas, saques y tiros libres son eventos que modifican constantemente el flujo del partido, y comprender su influencia es esencial para la predicción de los resultados. La RNN puede aprender cómo estos eventos influyen en la secuencia temporal del partido, ayudando a modelar patrones complejos relacionados con las sanciones y las interrupciones. En conjunto, estas características permiten a la RNN captar la influencia del árbitro en la dinámica global del juego, proporcionando una mejor comprensión de cómo los factores relacionados con las sanciones y las interrupciones afectan el comportamiento de los equipos y, en última instancia, el resultado del partido.

- Total de tarjetas: El total de tarjetas es una métrica combinada que da una visión completa de las sanciones aplicadas a los jugadores durante el partido. Tanto las tarjetas rojas como las amarillas afectan al desarrollo del juego, ya que las amonestaciones y expulsiones pueden influir en la moral de los jugadores y en la estrategia del equipo. Esta variable es importante para que la RNN aprenda

cómo las sanciones afectan la dinámica del juego a lo largo del tiempo y cómo estas interacciones pueden influir en la predicción del resultado del partido.

$$total\_tarjetas = total\_tarjetas\_rojas + total\_tarjetas\_amarillas$$

- Tarjetas por falta: Esta característica mide la relación entre el número de tarjetas tanto rojas como amarillas y el número total de faltas cometidas. Un alto valor de $tarjetas\ por\ falta$ sugiere que el árbitro está sancionando agresivamente las faltas, mientras que un valor bajo puede indicar un arbitraje más permisivo. Esta métrica es importante para comprender la tendencia del árbitro en cuanto a las sanciones por faltas, lo cual puede influir en el ritmo del juego y en las decisiones tácticas de los equipos. Ayuda a la RNN a identificar cómo las decisiones del árbitro afectan la dinámica general del juego.

$$tarjetas\_por\_falta = \frac{total\_tarjetas}{total\_faltas + \epsilon}$$

- Total de interrupciones: El total de interrupciones es una métrica que agrega faltas, tiros libres y saques de banda, lo que da una idea general del número de veces que el flujo del juego se ve interrumpido. Un alto número de interrupciones puede ser un indicador de un juego más fragmentado y de mayor tensión, lo que puede afectar la fluidez del partido. Esta característica ayuda a la RNN a identificar cómo los cambios frecuentes en el flujo del juego pueden afectar la dinámica y el resultado del partido.

$$total\_interrupciones = total\_faltas + total\_tiros\_libres + total\_saques\_banda$$

Donde $\varepsilon = 10^{-5}$ una constante pequeña añadida para evitar la división por cero y equipo representa “local” y "visita", según corresponda.

A través de estos pasos, los datos fueron finalmente preparados y estructurados para ser alimentados en los modelos de aprendizaje supervisado

4.5. Extracción de Datos Meteorológicos (Capa Bronce)

4.5.1. Selección de Fuente de Datos Ambientales

Para la integración de variables climáticas se seleccionó la plataforma NASA POWER (Prediction Of Worldwide Energy Resources), que proporciona datos meteorológicos globales derivados de observaciones satelitales. Esta elección se fundamentó en tres ventajas principales sobre estaciones meteorológicas terrestres:

1. Cobertura espacial completa: Los satélites proporcionan datos en cualquier coordenada geográfica, eliminando el problema de estadios ubicados lejos de estaciones meteorológicas
2. Consistencia temporal: Los datos satelitales están disponibles desde 1981 con metodología consistente, mientras que estaciones terrestres presentan períodos de operación variables
3. Acceso programático: La API REST de NASA POWER permite la extracción automatizada sin restricciones severas de tasa de solicitud

La principal limitación de esta fuente radica en la resolución espacial de aproximadamente 0.5° × 0.5° (aproximadamente 50 km en el ecuador), que puede introducir imprecisiones en áreas con microclimas marcados

4.5.2. Diseño del Sistema de Extracción Paralela

La extracción de datos meteorológicos presenta un desafío computacional significativo debido al volumen de solicitudes requeridas. Para cada partido, se necesita consultar múltiples horas de datos meteorológicos, lo que para un dataset de varios miles de partidos puede resultar en decenas de miles de llamadas al API.

Para abordar este desafío, se diseñó un sistema de extracción basado en concurrencia mediante $ThreadPoolExecutor$ de Python. La arquitectura se estructura en tres componentes principales:

Clase contenedora: Encapsula toda la lógica de extracción y manejo de errores. Se inicializa con la lista de variables meteorológicas deseadas, número máximo de $workers$ paralelos y límite de reintentos por solicitud. Motor de solicitudes con reintentos:

Implementa una estrategia de $backoff$ exponencial para manejar fallos transitorios de red. Si una solicitud falla, el sistema espera 2 x n segundos antes de reintentar, donde n es el número de intento. Esta estrategia reduce la probabilidad de exceder límites de tasa del servidor el algoritmo utilizado es el 2.

Algorithm 2 Reintentos de Petición a la API.

procedure REALIZARPETICIONAPI(url, $max\_retries$)
    Entrada: URL de la API url, número máximo de reintentos $max\_retries$
    Salida: Respuesta de la API en formato JSON o None en caso de error
    for $attempt = 0$ to $max\_retries - 1$ **do**
        Intentar realizar la petición HTTP GET a url con tiempo de espera de 30 segundos
        Si la respuesta tiene un código de estado exitoso:
            Retornar la respuesta en formato JSON
        Si ocurre una excepción RequestException:
        if $attempt = max\_retries - 1$ **then**
            Loguear error definitivo con el mensaje de la excepción
                return None
        else
            Calcular el tiempo de espera $wait\_time = 2^{attempt}$
            Esperar $wait\_time$ segundos antes de reintentar
        end if
    end for
    return None
end procedure

Sistema de procesamiento paralelo: Utiliza $ThreadPoolExecutor$ para procesar múltiples registros simultáneamente, manteniendo un pool de $workers$ que procesan solicitudes de manera concurrente. La implementación incluye una barra de progreso que proporciona retroalimentación visual del avance, el procesamiento implementado se muestra en el algoritmo 3.

Algorithm 3 Procesamiento Concurrente de Registros.

procedure PROCESARREGISTROSCONCURRENTEMENTE(*records*, *max_workers*, *mode*)
    Entrada: Lista de registros *records*, número máximo de trabajadores *max_workers*, modo de procesamiento *mode*
    Salida: Lista de resultados procesados
    Inicializar una lista vacía *results*
    Crear un `ThreadPoolExecutor` con *max_workers* trabajadores
    Crear un diccionario vacío *future_to_record*
    for cada registro *record* en *records* **do**
        Enviar la tarea de procesamiento del registro *record* al `ThreadPoolExecutor`, asociando el resultado con el registro en el diccionario *future_to_record*
    end for
    for cada tarea completada *future* en `as_completed(future_to_record)` **do**
        Obtener el resultado de la tarea *result* ← *future.result*()
        Agregar *result* a la lista *results*
    end for
    return *results*
end procedure

4.5.3. Estrategias de Agregación Temporal

La API de NASA POWER proporciona datos horarios, mientras que los partidos de fútbol tienen duración aproximada de 2 horas. Se implementaron dos modos de agregación temporal para capturar las condiciones meteorológicas relevantes:

Modo de 3 horas ($mode = '3h'$): Calcula el promedio de 3 horas consecutivas antes de la hora de inicio del partido. Este enfoque proporciona una representación más precisa de las condiciones durante el encuentro, particularmente relevante para variables como velocidad del viento que pueden cambiar significativamente en períodos cortos.

4.5.4. Evaluación de Calidad en Datos Meteorológicos

Después de la ejecución del algoritmo de ingesta paralela se analiza estas series temporales al igual que se realizó con los datos estadísticos de futbol para analizar la calidad de datos obtenidos, siguiendo los principios del DAMA-DMBOK (International, 2017), adaptados al contexto de datos de series temporales extraídos de APIs web.

Exactitud: La validación de datos meteorológicos presentó desafíos únicos debido a la ausencia de verdad fundamental absoluta. Se implementó una estrategia de validación cruzada comparando los datos satelitales de NASA POWER con:

- Estaciones meteorológicas cercanas.
- Rangos históricos esperados para cada ubicación geográfica.
- Coherencia física entre variables relacionadas (temperatura-humedad, precipitación-nubosidad).
- Las series temporales muestran correlaciones.

Completitud: Se cuenta con la data completa, la API de NASA POWER utiliza el valor -999 para indicar datos faltantes o inválidos. El sistema de extracción filtra estos valores automáticamente antes de calcular promedios, evitando que contaminen las estadísticas agregadas.

Consistencia: Se implementaron las siguientes validaciones de coherencia física. Precipitación no nula implica nubosidad, radiación solar inversamente proporcional a nubosidad y humedad relativa entre 0-100 %. Los casos inconsistentes fueron tratados mediante imputación basada en patrones temporales como interpolación de horas adyacentes.

Temporalidad: Un desafío particular fue la sincronización horaria. Los datos deportivos utilizan hora local del estadio, NASA POWER emplea UTC que es igual al mismo formato horario los partidos. Se considerando zona horaria del país y diferencias entre hora programada y hora real de inicio.

4.6. Transformación de Datos Meteorológicos (Capa Plata)

4.6.1. Variables Meteorológicas Seleccionadas

Se seleccionaron siete variables meteorológicas en base a su relevancia documentada en la literatura deportiva y su disponibilidad en la plataforma NASA POWER:

- Temperatura a 2 metros: Afecta el rendimiento físico de los jugadores y la dinámica del balón

- Humedad relativa a 2 metros: Influye en la sensación térmica y la fatiga
- Velocidad del viento a 2 metros: Impacta la trayectoria del balón, especialmente en pases largos y tiros
- Dirección del viento: Permite identificar ventajas direccionales
- Cobertura nubosa: Afecta la visibilidad y condiciones de juego
- Precipitación corregida: Determina si el campo está húmedo o seco
- Radiación solar descendente: Relacionada con temperatura percibida y visibilidad

4.6.2. Gestión de Errores y Trazabilidad

El sistema implementa múltiples niveles de manejo de errores para garantizar robustez ante fallos parciales:

1. Validación de coordenadas: Verifica que las coordenadas geográficas sean vá lidas antes de intentar la extracción
2. Manejo de timeouts: Cada solicitud tiene un timeout de 30 segundos para evitar bloqueos indefinidos
3. Registro de fallos: Los errores se registran con nivel de detalle suficiente para diagnóstico posterior
4. Campo de éxito: Cada registro resultante incluye un campo $api\_success$ que indica si la extracción fue exitosa

Al finalizar el proceso, se genera una columna $api\_success$ estadística que indica la tasa de éxito global. Este diseño permite identificar patrones sistemáticos de fallo, por ejemplo, ciertos rangos de coordenadas o períodos temporales, que puedan requerir atención especial.

4.6.3. Limpieza y Normalización

El proceso de preparación de datos meteorológicos inició con la eliminación de la columna $api\_success$, utilizada únicamente para control de calidad durante la extracción.

Posteriormente, se reconstruyó un índice temporal unificado combinando las columnas fecha y hora.

Esta transformación facilita la sincronización temporal con los datos deportivos y permite análisis de series temporales si se requieren.

La renombrado de columnas se realizó siguiendo convenciones descriptivas en español para mantener consistencia con el dataset deportivo.

4.7. Preparación Final de Datos Meteorológicos (Capa Oro)

4.7.1. Ingeniería de Características Meteorológicas

Estas nuevas características es incorporar datos ambientales que puedan tener un impacto significativo en el juego de fútbol. Las redes neuronales recurrentes son especialmente adecuadas para este tipo de tareas, ya que permiten modelar secuencias temporales y patrones dependientes del tiempo. En este caso, las características ambientales como el viento, la lluvia y las condiciones climáticas adversas pueden afectar la dinámica del partido de manera no lineal y de forma continua a lo largo del tiempo.

Componente *x* del viento: El viento tiene una dirección y una velocidad. Para entender mejor su efecto en el juego de fútbol, es útil descomponer la velocidad del viento en sus componentes en los ejes *x* y *y* La componente *x* (u horizontal) es crucial para entender cómo el viento afecta a los movimientos horizontales del balón (como en los tiros o el pase). Esta característica ayudará a la red neuronal recurrente a captar los patrones temporales relacionados con la influencia del viento en el desplazamiento del balón durante el partido.

$$viento_x = velocidad\_viento \,.\cos(\text{dirección\_viento})$$

Componente *y* del viento: La componente *y* del viento representa la dirección vertical del viento. Al igual que la componente *x* esto se descompone para estudiar cómo el viento afecta los movimientos verticales del balón, como cuando el viento puede hacer que el balón se desplace en el aire de arriba hacia abajo. Al incorporar esta característica,

podemos modelar la relación entre la dirección del viento y los patrones de juego, lo cual es crucial para predicciones precisas en condiciones adversas.

$$viento_y = velocidad\_viento \,.\, \text{sin}(\text{dirección\_viento})$$

Lluvia (1 si hay lluvia, 0 si no): La lluvia tiene un impacto directo en el terreno de juego, haciendo que el balón se desplace de manera diferente y afectando la capacidad de los jugadores para controlar el balón. Esta característica es binaria (1 si hay lluvia, 0 si no) y ayuda a la RNN a reconocer patrones de juego en condiciones de lluvia, lo que puede afectar el rendimiento de ambos equipos, especialmente en cuanto a precisión y control del balón.

$$lluvia = \begin{cases} 1, & precipitacion > UMBRAL\_LLUVIA \\ 0, & precipitacion \leq 0 \end{cases}$$

Clima adverso (1 si se cumple alguna condición adversa, 0 si no): El clima adverso se define como una combinación de condiciones que pueden afectar el desarrollo del partido, como lluvia, viento fuerte o temperaturas extremas. Esta variable binaria captura la presencia de condiciones climáticas adversas. La RNN puede aprender cómo estos factores combinados influyen en el rendimiento y la estrategia de los equipos. Por ejemplo, el viento fuerte o la lluvia pueden hacer que el juego sea más impredecible, afectando tanto el control del balón como la toma de decisiones de los jugadores.

Viento severo (cuadrado de la velocidad del viento): El viento severo es un indicador del impacto potencial del viento en el juego. Cuanto mayor sea la velocidad del viento, más difícil será controlar el balón y predecir su trayectoria. Al usar el cuadrado de la velocidad del viento, esta característica amplifica el efecto de vientos fuertes, lo que puede ser crucial para la RNN al modelar situaciones de viento extremo, donde la influencia es mucho más significativa. Esto también ayuda a identificar patrones de juego donde los equipos tienen que adaptarse a un viento más fuerte, como en el caso de disparos o pases largos.

$$viento\_severo = velocidad\_viento^2$$

Esta transformación amplifica la diferencia entre condiciones ventosas moderadas y severas, reflejando que el impacto del viento sobre la trayectoria del balón escala aproximadamente con el cuadrado de la velocidad según principios aerodinámicos.

4.7.2. Tratamiento de Valores Atípicos

El análisis exploratorio mediante gráficos de caja reveló la presencia de valores extremos en varias variables meteorológicas. Para mantener la distribución natural de los datos mientras se limitan observaciones potencialmente erróneas, se implementó un clipping basado en el rango intercuartílico.

Este método, basado en el criterio de Tukey, preserva los datos bajo distribución normal mientras restringe valores extremos que podrían representar errores de medición o condiciones excepcionales no representativas.

La decisión de utilizar clipping en lugar de eliminación de outliers se fundamentó en dos consideraciones:

- Preservación del tamaño muestral: La eliminación de registros completos por outliers en una sola variable reduciría significativamente el dataset
- Naturaleza de datos satelitales: Los valores extremos pueden representar fenómenos meteorológicos reales como tormentas u olas de calor, son raros pero relevantes para el análisis.

4.7.3. Integración Datos Deportivos con Datos Ambientales

La incorporación de datos meteorológicos requirió un flujo de procesamiento independiente pero complementario. Complementario por que se requiere la finalización del procesamiento de los datos de futbol mediante la transformación, ya se imputan datos faltantes de manera jerárquica como se aprecia en el algoritmo 5 para poder cruzar la información y sea consistente. Los datos ambientales se obtuvieron mediante consultas a servicios de información climática histórica, sincronizados temporalmente con cada partido en ventanas de tres horas alrededor del horario de inicio. Esta sincronización temporal precisa resultó crítica para garantizar que las condiciones registradas correspondieran efectivamente a las experimentadas durante el evento deportivo. Las variables climáticas

crudas incluían temperatura, humedad relativa, velocidad del viento, dirección del viento, nubosidad, precipitación y radiación solar. Sin embargo, estas mediciones directas no necesariamente capturan los efectos físicos relevantes para el rendimiento deportivo.

La fusión de los datos meteorológicos y deportivos se realizó mediante combinación por identificador de partido, garantizando la correspondencia temporal exacta.

Se cruzan los datos para asegurar que solo se mantengan partidos con información meteorológica completa.

El dataset integrado resultante contiene los siguientes *features*:

- Variables identificadoras: $id\_partido$, $id\_estadio$, $id\_arbitro$, id_team_local, id_team_visita.
- Estadísticas deportivas: 28 variables que describen el rendimiento de los equipos.
- Variables meteorológicas: 16 variables que capturan las condiciones ambientales.
- Variables derivadas**:** Diferencias de rendimiento e indicadores categóricos.

El análisis se estructura en tres segmentos principales: en el ámbito del Equipo, se evalúan métricas de rendimiento como despejes, tiros de esquina, atajadas y disparos fuera del arco, junto con los porcentajes de balones largos, centros, regates, pases, disparos al arco, disparos en el área, fuera de ella y bloqueados. Respecto al Árbitro, se consideran las tarjetas por falta y el total de interrupciones, mientras que en la dimensión del Ambiente se analiza la diferencia en duelos aéreos, temperatura, porcentaje de humedad, dirección de viento, porcentaje de nubosidad, radiación, viento de sur a norte, viento de oeste a este y viento severo.

4.7.4. Validación Final de Calidad

Previo a la exportación del dataset integrado, se ejecutó una batería de verificaciones de calidad:

- Verificación de completitud: Se confirmó la completitud de variables críticas para el modelo, aceptando nulos únicamente en campos opcionales.
- Validación de rangos: Se verificaron rangos esperados para variables numéricas:
    - Temperatura entre -10°C y 45°C
    - Humedad relativa entre 0 % y 100 %
    - Velocidad del viento entre 0 y 30 m/s
    - Precipitación entre 0 y 50 mm/hora
- Consistencia temporal: Se validó que todos los registros tuvieran timestamps válidos dentro del período de estudio (2017-2025).
- Integridad referencial: Se confirmó que todos los identificadores de equipos, estadios y competiciones existieran en sus respectivas tablas dimensionales.

El dataset final se exportó en formato CSV con codificación UTF-8, preservando caracteres especiales en nombres de equipos y estadios

Este archivo constituye el punto de entrada para las fases posteriores de modelado predictivo, conteniendo 9294 registros de partidos con 44 variables explicativas y una variable objetivo que es el resultado del partido.

CAPITULO V

Pruebas y resultados

En este capítulo, nos centramos en las pruebas y resultados, cuyo conjunto de entradas está compuesto por estadísticas del partido de fútbol, así como datos relacionados con el equipo local, el equipo visitante y las condiciones generales del partido, como la caracterización del árbitro, los cuales fueron definidos en el capítulo de ingeniería de características.

Por otro lado, se incluye el modelo $Enchantress$ elaborado en el experimento denominado "$F-A1$", que utiliza las mismas entradas que el modelo base denominado "$F-1$", pero con la adición de características ambientales. Los experimentos presentados en este capítulo abarcan diferentes variaciones en la arquitectura de los modelos. Cada uno de estos experimentos se lleva a cabo tanto para el modelo base como para el modelo $Enchantress$, con el fin de evaluar su desempeño y comparar los resultados obtenidos.

Además, los experimentos incluyen datos provenientes de diversos países, tales como Colombia, Ecuador, Chile y Perú, lo que permite analizar el comportamiento del modelo en contextos nacionales específicos. Parte de este análisis se enfoca en examinar cómo varían las métricas de desempeño entre los diferentes países, lo que proporciona una visión más detallada sobre la influencia de las características contextuales en los resultados de las predicciones.

5.1. Preparación para pruebas

5.1.1. División de Dataset

El flujo general de los modelos de aprendizaje supervisado se inicia con un proceso fundamental: la división de los datos en tres subconjuntos, requirió una estructuración cuidadosa para evitar fugas de información del futuro hacia el pasado, destinados al entrenamiento, validación y prueba del modelo. En primer lugar, se lleva a cabo una división de los datos de acuerdo con una proporción estándar de 75 % para entrenamiento, 10 % para validación y 15 % para prueba. Este enfoque de partición asegura que el modelo

sea entrenado sobre un conjunto de datos suficientemente grande, al tiempo que se conserva una porción representativa de los datos para validación y evaluación.

En este proceso, se ha dado especial atención a la temporalidad de los partidos. En lugar de realizar una partición arbitraria de los datos, se ha respetado el orden temporal de los partidos, de manera que el conjunto de entrenamiento incluye los partidos más antiguos, mientras que los conjuntos de validación y prueba contienen los partidos más recientes. Este enfoque garantiza que el modelo sea evaluado en datos futuros a partir de los datos históricos, un aspecto crucial para problemas de predicción de eventos deportivos, donde la naturaleza secuencial de los datos no debe ser ignorada.

5.1.2. Normalización

A continuación, se realiza el proceso de normalización de los datos, que se lleva a cabo de acuerdo con el tipo de columna y su distribución específica. La normalización es esencial para garantizar que las variables con diferentes escalas no dominen el proceso de entrenamiento del modelo, ya que muchas técnicas de aprendizaje automático, como los algoritmos basados en distancias o redes neuronales, son sensibles a la magnitud de las características. Este proceso se realiza para cada tipo de dato, asegurando que los valores sean ajustados de manera que todas las características tengan una distribución similar, lo que facilita la convergencia del modelo durante el entrenamiento.

La normalización de las características numéricas se realizó exclusivamente basándose en las estadísticas del conjunto de entrenamiento. Se aplicó estandarización robusta a la mayoría de las variables para soportar valores atípicos. Las variables que representan proporciones naturales recibieron un tratamiento diferente mediante escalamiento $min - max$, que preserva su interpretación como valores entre cero y uno. Los escaladores ajustados en el entrenamiento se aplicaron posteriormente de manera idéntica a los conjuntos de validación y prueba, manteniendo la consistencia en la representación de las características.

5.1.3. Arquitectura de series temporales

Además, se lleva a cabo la construcción de la ventana de temporalidad, que se calcula específicamente por grupo. La transformación del dataset tabular en secuencias temporales adecuadas para redes neuronales recurrentes representó uno de los aspectos más complejos del diseño experimental. El objetivo consistía en capturar la dinámica temporal del rendimiento de los equipos mediante ventanas históricas de cuatro partidos previos. Esta ventana se refiere al historial de los partidos, es decir, los datos previos a un partido específico, lo cual es crucial para la predicción de eventos futuros basados en el comportamiento histórico. Para los partidos de equipo local y visitante, se busca en el historial de partidos previos de esos mismos equipos, de modo que las características utilizadas para predecir los resultados futuros provienen de datos pasados de las mismas entidades deportivas. Esta estrategia permite aprovechar el contexto histórico de cada equipo, lo que mejora la capacidad predictiva del modelo.

5.1.3.1. Historial por Equipo.

Se desarrolló un mecanismo de extracción de rachas que recupera los últimos N partidos de cada equipo anteriores a la fecha del encuentro a predecir. Este proceso requirió primero desagregar el dataset original, donde cada fila representa un partido completo con estadísticas de ambos equipos, en un formato de equipo-partido donde cada fila corresponde a la participación de un equipo específico en un partido dado. Esta transformación duplica el número de observaciones, pero permite rastrear la trayectoria temporal de cada equipo de manera independiente. Para cada equipo involucrado en un partido, se extrajeron sus cuatro partidos inmediatamente anteriores a la fecha del encuentro. Esta ventana de cuatro partidos se determinó balanceando dos consideraciones opuestas: ventanas más largas proporcionan más contexto histórico, pero pueden incluir información obsoleta debido a cambios en la plantilla o el esquema táctico; ventanas más cortas capturan el estado reciente, pero son más susceptibles a la variabilidad aleatoria de encuentros individuales. La elección de cuatro partidos representa

un compromiso razonable que captura aproximadamente un mes de actividad para equipos que juegan semanalmente.

5.1.3.2. Incorporación de Contexto Ambiental y Arbitral

En cuanto a los datos climáticos, la construcción de la ventana de temporalidad se realiza de manera similar. Los datos climáticos se cruzan exclusivamente según las características específicas de cada partido, es decir, el contexto climático del día del partido es asignado a la fecha del evento, considerando las variables climáticas de partidos pasados.

La caracterización del árbitro siguió una lógica similar, pero con un enfoque diferente. Se recuperaron sus últimos cuatro partidos arbitrados, independientemente de los equipos involucrados, calculando el promedio de variables. Este promedio móvil captura el estilo arbitral reciente del oficial, que puede variar en el tiempo debido a directrices cambiantes de las asociaciones de árbitros o la evolución personal del criterio arbitral.

Un desafío metodológico surgió con equipos que carecían de suficiente historial en el dataset. Particularmente, equipos recién ascendidos o aquellos con registros incompletos no podían proporcionar cuatro partidos previos. La decisión tomada fue excluir estas observaciones del entrenamiento, aceptando una reducción en el tamaño muestral a cambio de mantener la consistencia estructural de las secuencias. Esta exclusión afecta principalmente a las primeras temporadas del dataset y a competiciones con cobertura parcial.

5.2. Evaluación para la predicción

En base al objetivo general Ambos experimentos ("$F-1$": base deportivo, "$F-A1$": enriquecido con clima) siguieron protocolos idénticos en términos de espacio de búsqueda de hiper parámetros, procedimientos de evaluación y estrategias de regularización, variando únicamente en la dimensionalidad de entrada. El experimento base procesó secuencias de 28 características derivadas exclusivamente de estadísticas deportivas y contexto arbitral. El experimento enriquecido incorporó 16 variables meteorológicas

adicionales, resultando en secuencias de 44 características. Esta estrategia de comparación controlada permite aislar el efecto atribuible a la inclusión de información ambiental, manteniendo constantes el resto de factores metodológicos.

5.2.1 Búsqueda en rejilla de hiperparametros (Grid Search)

El espacio de hiper parámetros explorado abarcó 12 combinaciones distintas, definidas por las siguientes configuraciones:

5.2.1.1 Arquitectura Recurrente

La base de la experimentación se centró en la evaluación de la arquitectura, comenzando con el Tipo de Célula recurrente, donde se utilizó la LSTM y la GRU ambas esenciales para la gestión efectiva de dependencias de largo alcance en las secuencias. La capacidad de representación del modelo fue modulada mediante la variación de las Unidades en la Primera Capa (24, 32 o 64), mientras que la profundidad y complejidad se ajustaron con la Segunda Capa, probando 0 (configuración simple), 16 o 32 unidades para la captura de patrones de orden superior. Por razones de causalidad temporal, inherente a los datos secuenciales, la Direccionalidad se mantuvo estrictamente Unidireccional (forward).

5.2.1.2 Regularización

Para mitigar el riesgo de sobreajuste (overfitting), se implementaron dos mecanismos clave de regularización. Primero, se aplicó el método Dropout con tasas de 0.4 y 0.5 inmediatamente después de cada capa recurrente. Segundo, se evaluó la inclusión de una Capa Densa Intermedia con 8 unidades para realizar una transformación no lineal antes de la capa de clasificación final; alternativamente, se probó el valor 0, estableciendo una conexión directa entre las capas recurrentes y la salida.

5.2.1.3 Optimización y Entrenamiento

En cuanto al proceso de optimización, se seleccionó el algoritmo Adam debido a su probada eficiencia computacional y su robusta convergencia en una amplia variedad de tareas de Deep Learning. La Tasa de Aprendizaje se exploró con 0.001 buscando un balance adecuado entre la velocidad de convergencia y la estabilidad del proceso. El

Tamaño del Lote (Batch Size) se fijó en 128, impactando directamente en la estabilidad del gradiente y el tiempo total de entrenamiento. Finalmente, se estableció un límite de 100 Épocas Máximas, complementado con una estrategia de Detención Temprana (Early Stopping) con una Paciencia de épocas sin observar mejora en el conjunto de validación, garantizando así una convergencia eficiente y la prevención del sobre entrenamiento.

5.2.1.4 Función Objetivo

Se seleccionó la precisión global como métrica de optimización, definida como la proporción de predicciones correctas sobre el total de instancias. Aunque esta métrica puede ser sensible al desbalance de clases, su uso se justifica al incorporar pesos de clase durante el entrenamiento, lo que compensa la distribución desigual de resultados deportivos

Para cada configuración, se realizó el siguiente procedimiento:

1. Inicialización del modelo con la arquitectura especificada
2. Entrenamiento sobre el conjunto de entrenamiento con monitorización en validación
3. Evaluación de la métrica objetivo (accuracy) sobre el conjunto de validación
4. Registro de métricas complementarias (precisión por clase, recall, F1-score)
5. Almacenamiento de resultados y configuración para análisis posterior

El proceso completo se instrumentó para medir tiempos de ejecución, permitiendo evaluar el costo computacional asociado a cada configuración. Los resultados se serializaron en formato JSON, facilitando su análisis comparativo y reproducibilidad experimental.

5.2.2 Métricas de evaluación

Para evaluar el desempeño del modelo LSTM en un escenario de clasificación multiclase con fuerte desbalance, es fundamental priorizar métricas que reflejen la calidad del aprendizaje en cada categoría y no solo el rendimiento global. Aunque la precisión general (accuracy) ofrece una referencia inicial, su utilidad es limitada porque favorece a la clase mayoritaria. En contraste, las métricas por clase precisión, recall y F1 permiten

identificar si el modelo reconoce adecuadamente las clases minoritarias, que suelen ser las más difíciles de predecir.

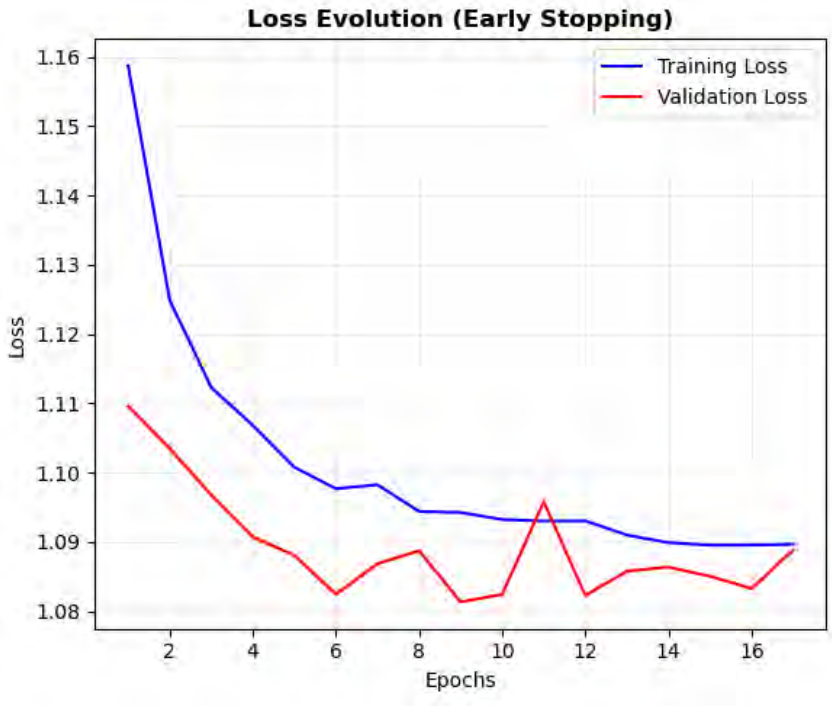
Finalmente, la matriz de confusión complementa el análisis al mostrar explícitamente los patrones de error y las confusiones entre categorías, ofreciendo una comprensión más profunda del comportamiento del modelo.

5.3. Experimento ‘F - 1’ (únicamente variables deportivas).

El proceso de optimización mediante búsqueda en rejilla de hiperparámetros identificó una arquitectura óptima basada en unidades recurrentes cerradas (GRU, por sus siglas en inglés). La configuración resultante emplea una única capa recurrente con 24 unidades, tasa de abandono del 50%, tasa de aprendizaje de 0.001 mediante el optimizador Adam, y tamaño de lote de 128 instancias. Esta arquitectura fue seleccionada tras evaluar sistemáticamente 36 combinaciones distintas de hiperparámetros durante un período de entrenamiento que abarcó 17 épocas antes de la activación del mecanismo de detención temprana.

Figura 9

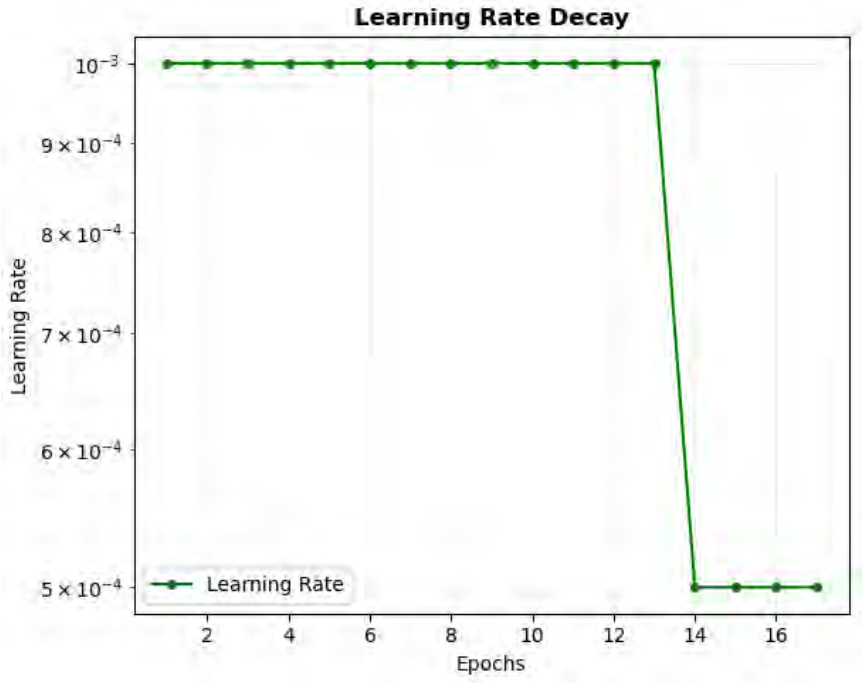
Función de perdida experimento 'F-1'



La evolución del proceso de entrenamiento, ilustrada en la Figura 9, revela una convergencia progresiva de las funciones de pérdida tanto en el conjunto de entrenamiento como en el de validación. La pérdida de entrenamiento exhibe una reducción pronunciada durante las primeras cinco épocas, decreciendo desde 1.16 hasta aproximadamente 1.09, para posteriormente estabilizarse en valores cercanos a 1.09. De manera análoga, la pérdida de validación desciende desde 1.11 hasta 1.08 durante el mismo intervalo temporal, manifestando posteriormente oscilaciones controladas que sugieren un equilibrio adecuado entre capacidad de ajuste y generalización.

Figura 10

Disminución de la tasa de aprendizaje experimento 'F-1'



La implementación del mecanismo de reducción adaptativa de la tasa de aprendizaje Figura 10 demuestra su activación en la época 13, disminuyendo el valor desde 0.001 hasta 0.0005, estrategia que contribuyó a la estabilización del proceso optimización sin comprometer la capacidad predictiva del sistema.

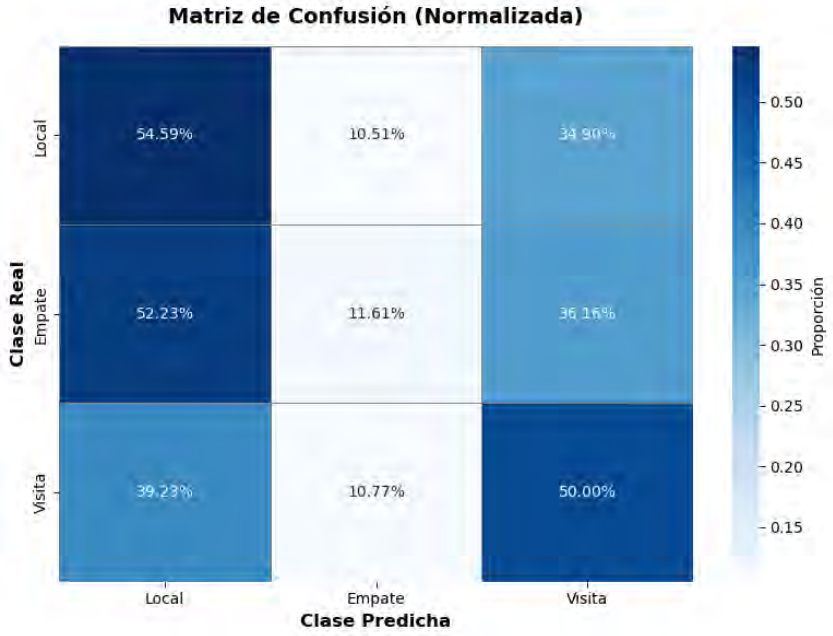
5.3.1 Rendimiento Global

El modelo optimizado alcanzó una exactitud del 43.7% en el conjunto de validación, métrica que constituyó el criterio de selección durante la búsqueda de hiperparámetros.

Esta cifra representa una mejora sustancial respecto a la línea base aleatoria (33.3% para un problema de tres clases equilibrado), evidenciando la capacidad del sistema para capturar patrones relevantes en los datos históricos. El puntaje F1-macro, que pondera equitativamente el rendimiento en las tres categorías de resultado (victoria local, empate, victoria visitante), se situó en 0.337, mientras que el F1-ponderado alcanzó 0.383, reflejando un desempeño diferenciado según la clase predictiva.

Figura 11

Matriz de confusión experimento 'F - 1'



La evaluación del rendimiento del modelo mediante la matriz de confusión normalizada como se muestra en la figura 11 reveló que, si bien el clasificador exhibe una capacidad aceptable para identificar las victorias (alcanzando una precisión del 54.59% para la clase Local y un 50.00% para la clase Visita), existe una dificultad metodológica crítica centrada en la predicción de la clase Empate. Esta clase es clasificada correctamente en solo el 11.61% de las ocasiones, lo que se debe principalmente a un marcado sesgo del modelo que confunde los Empates reales con los resultados de Local 52.23% y Visita 36.16%

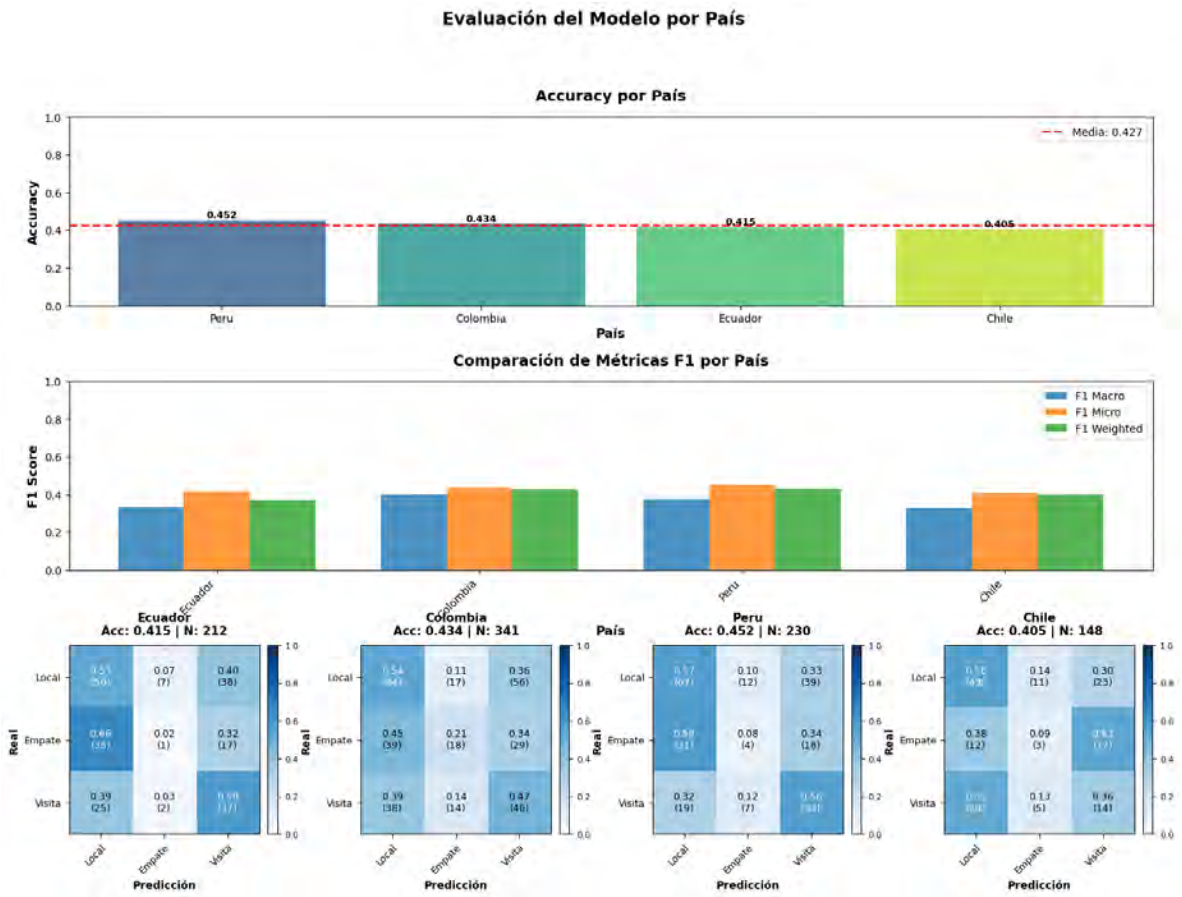
5.3.2 Rendimiento por País

Con el modelo ya entrenado, volvemos a evaluar con el conjunto de datos de test, para esta solución se cruza con los países de cada partido. La evaluación desagregada por contexto geográfico revela variaciones significativas en la capacidad predictiva del modelo (Figura 12). Perú emerge como el territorio con mayor exactitud (45.2%), seguido por Colombia (43.4%), Ecuador (41.5%) y Chile (40.5%). Esta disparidad, aunque moderada (rango de 4.7 puntos porcentuales), resulta estadísticamente relevante dado el volumen diferencial de instancias evaluadas: 230 partidos para Perú, 341 para Colombia, 212 para Ecuador y 148 para Chile.

El análisis de las métricas F1 por país Figura 12 muestra una consistencia notable entre los diferentes indicadores. Los valores de F1-macro oscilan entre 0.31 (Chile) y 0.37 (Ecuador y Colombia), mientras que los puntajes F1-ponderados exhiben un rango ligeramente superior (0.36-0.44). Esta convergencia relativa entre métricas sugiere que el modelo no presenta sesgos pronunciados hacia clases mayoritarias en ninguno de los contextos evaluados.

Figura 12

Evaluación por país experimento 'F - 1'



Las matrices de confusión normalizadas (panel inferior de la Figura 12) revelan patrones sistemáticos en el comportamiento predictivo del sistema. En todos los países analizados, la clase correspondiente a victoria del equipo local presenta las tasas de acierto más elevadas, oscilando entre 0.53 (Ecuador) y 0.57 (Perú). Este fenómeno resulta consistente con la ventaja estadística documentada del factor localía en el fútbol sudamericano. Contrariamente, la categoría de empate exhibe las mayores dificultades de discriminación, con tasas de recuperación que fluctúan entre 0.38 (Chile) y 0.66 (Ecuador), evidenciando la naturaleza inherentemente impredecible de este resultado. La clase de victoria visitante muestra un rendimiento intermedio, con valores de sensibilidad entre 0.32 (Perú) y 0.56 (ambos en Ecuador).

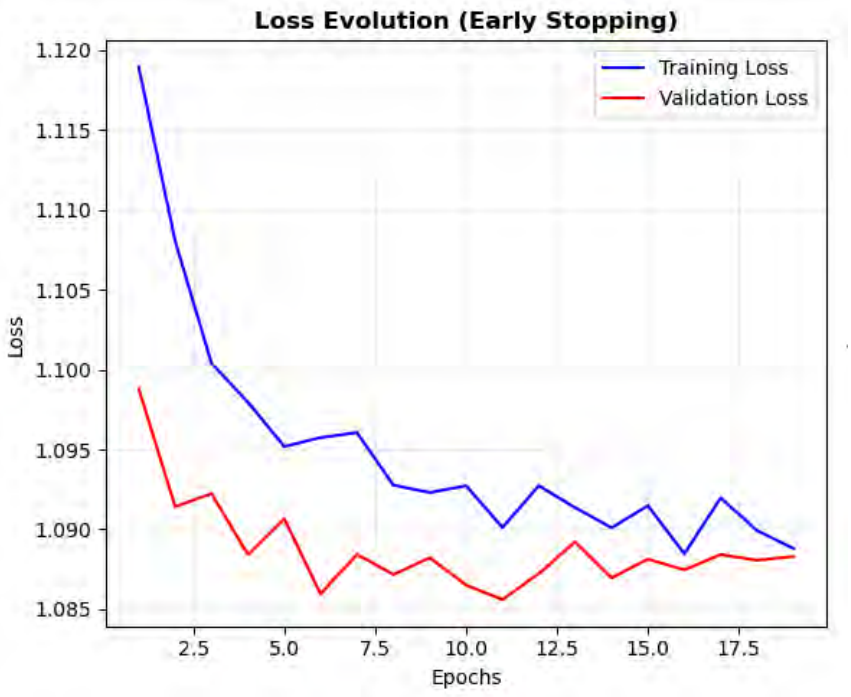
5.4. Experimento 'F – A1' (incorporación de variables ambientales).

El proceso de optimización mediante búsqueda en rejilla de hiperparámetros identificó una arquitectura óptima basada en unidades recurrentes cerradas (GRU, por sus siglas en inglés). La incorporación de variables meteorológicas al conjunto de predictores

condujo a la selección de una arquitectura alternativa durante el proceso de optimización. El modelo resultante emplea celdas LSTM (memoria a largo-corto plazo) con una configuración minimalista de 8 unidades en una única capa recurrente, manteniendo una tasa de abandono del 50% y parámetros de optimización idénticos al modelo base (tasa de aprendizaje de 0.001, optimizador Adam, tamaño de lote de 128). Esta simplificación arquitectural contrasta notablemente con las 24 unidades GRU del modelo exclusivamente futbolístico, sugiriendo que la información meteorológica permite alcanzar capacidad representacional comparable con menor complejidad estructural.

Figura 13

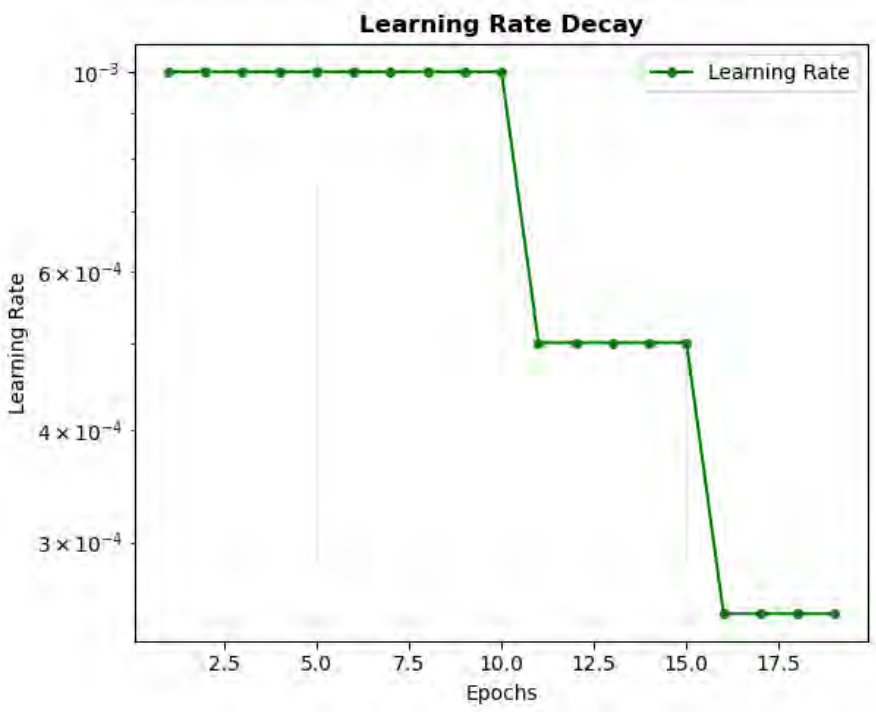
Función de perdida experimento 'F - A1'



El análisis de las curvas de aprendizaje Figura 13 evidencia un comportamiento de convergencia diferenciado respecto al modelo base. La pérdida de entrenamiento desciende desde 1.119 hasta estabilizarse en 1.089 tras 18 épocas, exhibiendo oscilaciones más pronunciadas durante el proceso. La pérdida de validación muestra un patrón paralelo, disminuyendo desde 1.099 hasta 1.085, con fluctuaciones que sugieren mayor sensibilidad a la composición específica de lotes durante el entrenamiento.

Figura 14

Disminución de la tasa de aprendizaje experimento 'F - A1'



El mecanismo de reducción adaptativa de la tasa de aprendizaje, Figura 14 se activó en dos momentos críticos: época 10 (reducción a 5×10^{-4}) y época 15 (reducción adicional a 2.5×10^{-4}), estrategia que facilitó la exploración refinada del espacio de parámetros sin comprometer la estabilidad del proceso

El modelo sugiere que el modelo captura adecuadamente señales asociadas con superioridad deportiva, pero enfrenta limitaciones al discernir situaciones de paridad competitiva.

5.4.1 Rendimiento Global

El sistema integrado alcanzó una exactitud promedio del 43.5% a través de los cuatro contextos geográficos evaluados, representando una disminución marginal de 0.2 puntos porcentuales respecto al modelo base (43.7%). Esta variación, si bien contraintuitiva considerando la información adicional incorporada, resulta estadísticamente insignificante y se encuentra dentro del margen de variabilidad inherente a procesos

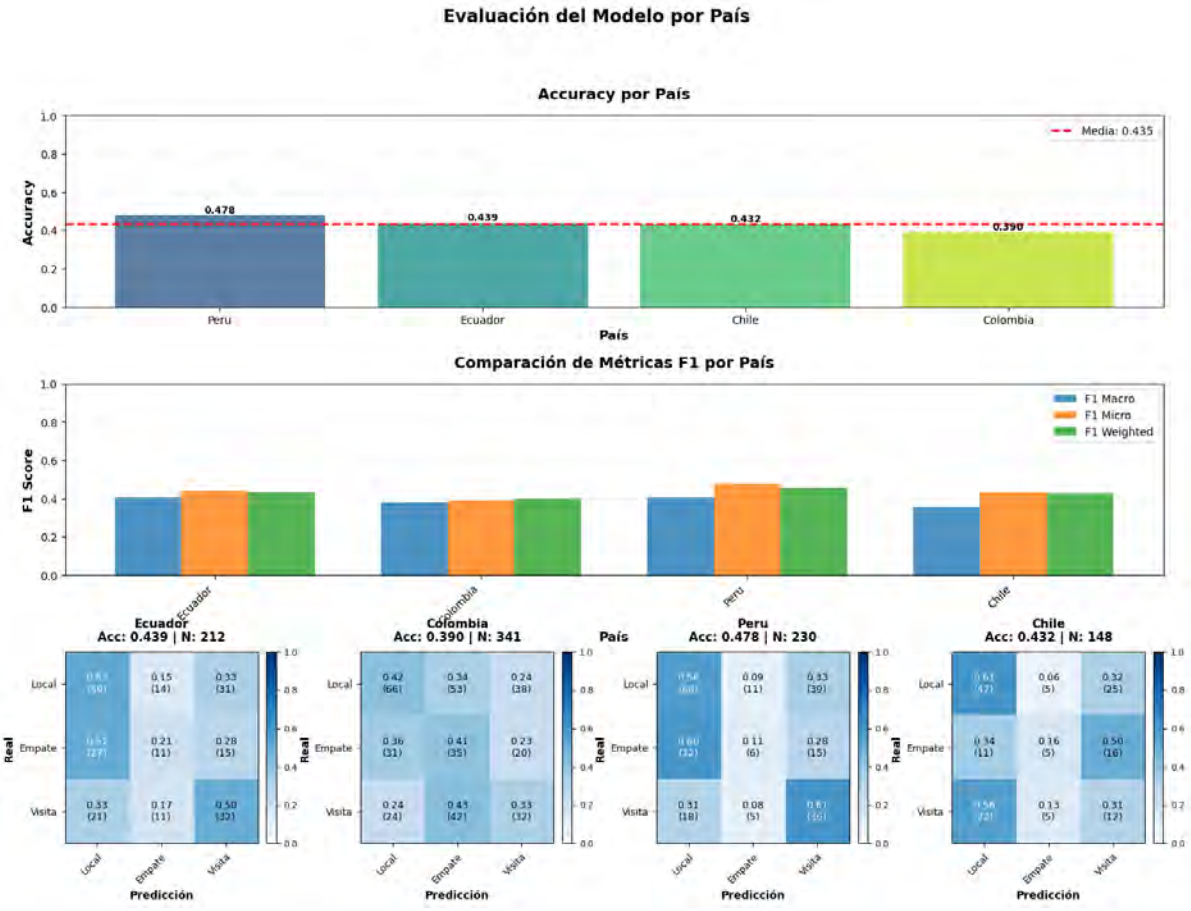
estocásticos de entrenamiento. Las métricas F1 reflejan un patrón análogo: el F1-macro promedio se mantuvo en valores comparables (diferencia inferior a 0.01 puntos), mientras que el F1-ponderado exhibió fluctuaciones similares entre ambas configuraciones.

5.4.2 Análisis por País

La evaluación desagregada por territorio revela patrones heterogéneos en la respuesta a la incorporación de variables meteorológicas (Figura 15).

Figura 15

Evaluación por país experimento 'F - A1'



Perú experimentó la mejora más sustancial, incrementando su exactitud de 45.2% a 47.8% (+2.6 puntos porcentuales, +5.8% relativo). Este resultado adquiere particular relevancia considerando que constituye el contexto focal de la investigación, donde la diversidad climática inherente al territorio peruano podría proporcionar señales discriminativas más informativas. Ecuador y Chile manifestaron mejoras modestas de 1.9

y 2.7 puntos porcentuales respectivamente (43.9% y 43.2%), mientras que Colombia experimentó una reducción de 4.4 puntos (39.0%), constituyendo el único caso de deterioro notable en el desempeño predictivo.

El análisis de métricas F1 por país (panel central, Figura 15) corrobora esta heterogeneidad. Perú exhibe los indicadores más robustos del conjunto evaluado (F1-macro: 0.41, F1-ponderado: 0.46), representando incrementos del 11% y 10% respectivamente frente al modelo base. Ecuador mantiene valores estables (F1-macro: 0.37), mientras que Chile muestra mejoras moderadas. Colombia, consistentemente con su reducción en exactitud, presenta los indicadores F1 más deteriorados del conjunto (F1-macro: 0.33, F1-ponderado: 0.41), con decrementos del 11% en F1-macro.

Las matrices de confusión normalizadas (panel inferior, Figura 15) revelan transformaciones específicas en los patrones de clasificación. En Perú, la sensibilidad para victorias locales se incrementó marginalmente (0.57 a 0.58), mientras que la capacidad de discriminación de empates mejoró sustancialmente (0.58 a 0.60), reduciendo simultáneamente confusiones con victorias locales (de 31 a 32 instancias, pero con mejor distribución proporcional). La clase de victoria visitante exhibió el cambio más significativo, incrementando su tasa de recuperación de 0.32 a 0.61 (+90% relativo), aunque con mayor confusión con empates (de 7 a 5 instancias correctamente clasificadas del total disponible).

Colombia presenta el patrón más preocupante: la sensibilidad para victorias locales decreció de 0.54 a 0.42 (-22% relativo), con incremento concomitante de confusiones hacia empates (de 17 a 53 instancias). La categoría de empate mejoró marginalmente su discriminación (0.45 a 0.41, considerando normalización), pero a expensas de mayor confusión con victorias visitantes. Ecuador y Chile muestran comportamientos intermedios, con mejoras en ciertas clases compensadas por deterioros en otras, resultando en incrementos netos modestos de exactitud global.

5.3. Importancia de Variables

Para abordar los objetivos específicos planteados de identificar variables estadísticas deportivas con mayor relevancia predictiva y determinar el poder predictivo de

variables meteorológicas se implementó un análisis de importancia de características mediante el modelo de Bosque Aleatorio. Esta técnica permite cuantificar la contribución de cada variable al proceso de clasificación, proporcionando interpretabilidad al sistema predictivo sin comprometer su capacidad de generalización.

5.3.1 Preparación de datos para análisis de importancia

El conjunto de datos utilizado integra estadísticas deportivas históricas y variables ambientales correspondientes a múltiples partidos de fútbol. Dado que el problema se enmarca en el contexto de series temporales, se aplicó una estrategia de ventana móvil para capturar patrones de rendimiento reciente: para cada partido, se calculó el promedio de las estadísticas de los últimos cuatro encuentros de cada equipo. Este enfoque permite condensar información temporal en una representación tabular sin pérdida significativa de contexto, generando así una matriz de características adecuada para algoritmos que operan sobre datos estructurados.

5.3.2 Configuración del modelo

El modelo corresponde a un clasificador de bosque aleatorio entrenado sobre una representación bidimensional de las secuencias, lo cual elimina la dependencia temporal y permite tratar cada muestra como un estado estático del partido. Bajo este enfoque, los hiperparámetros cumplen un rol central en controlar el equilibrio entre complejidad y generalización: se emplea un número elevado de árboles ($n\_estimators = 400$) para asegurar estabilidad estadística; una profundidad máxima restringida ($max\_depth = 12$) y límites mínimos para dividir y formar hojas ($min\_samples\_split = 20, min\_samples\_leaf = 10$) para evitar memorizar patrones espurios y reducir la varianza; una selección aleatoria acotada de variables en cada división ($max\_features = "sqrt"$) para promover independencia entre árboles; y un ajuste automático del peso de clases ($class\_weight = "balanced"$) para mitigar el sesgo hacia resultados mayoritarios. En conjunto, esta configuración impone una estructura robusta que favorece la interpretabilidad y estabiliza

la predicción sin depender de la memoria secuencial, apoyándose en la riqueza informativa ya contenida en los atributos preprocesados.

5.3.3 Resultados

El modelo Árbol Aleatorio empleando las 44 variables disponibles permitió establecer una primera aproximación al peso relativo que ejercen tanto los indicadores de rendimiento deportivo como las variables ambientales en la predicción del resultado de los partidos logrando una precisión de 47.8% con el conjunto de validación.

En relación con las variables de naturaleza deportiva respecto al primer objetivo específico, el modelo otorgó mayor relevancia a indicadores asociados con la creación y finalización de oportunidades ofensivas. Entre los predictores más destacados como se aprecia en la figura 16 se encuentran el porcentaje de balones largos ejecutados por el equipo local, la proporción de disparos realizados desde dentro del área y la frecuencia de remates desviados. Estas métricas, al reflejar tanto la orientación táctica como la capacidad de generar peligro de forma sostenida, aparecen consistentemente como elementos con capacidad discriminativa dentro del árbol de decisión. Su posición en el ranking sugiere que las dinámicas ofensivas inmediatas del encuentro constituyen señales particularmente informativas para anticipar el desenlace del partido.

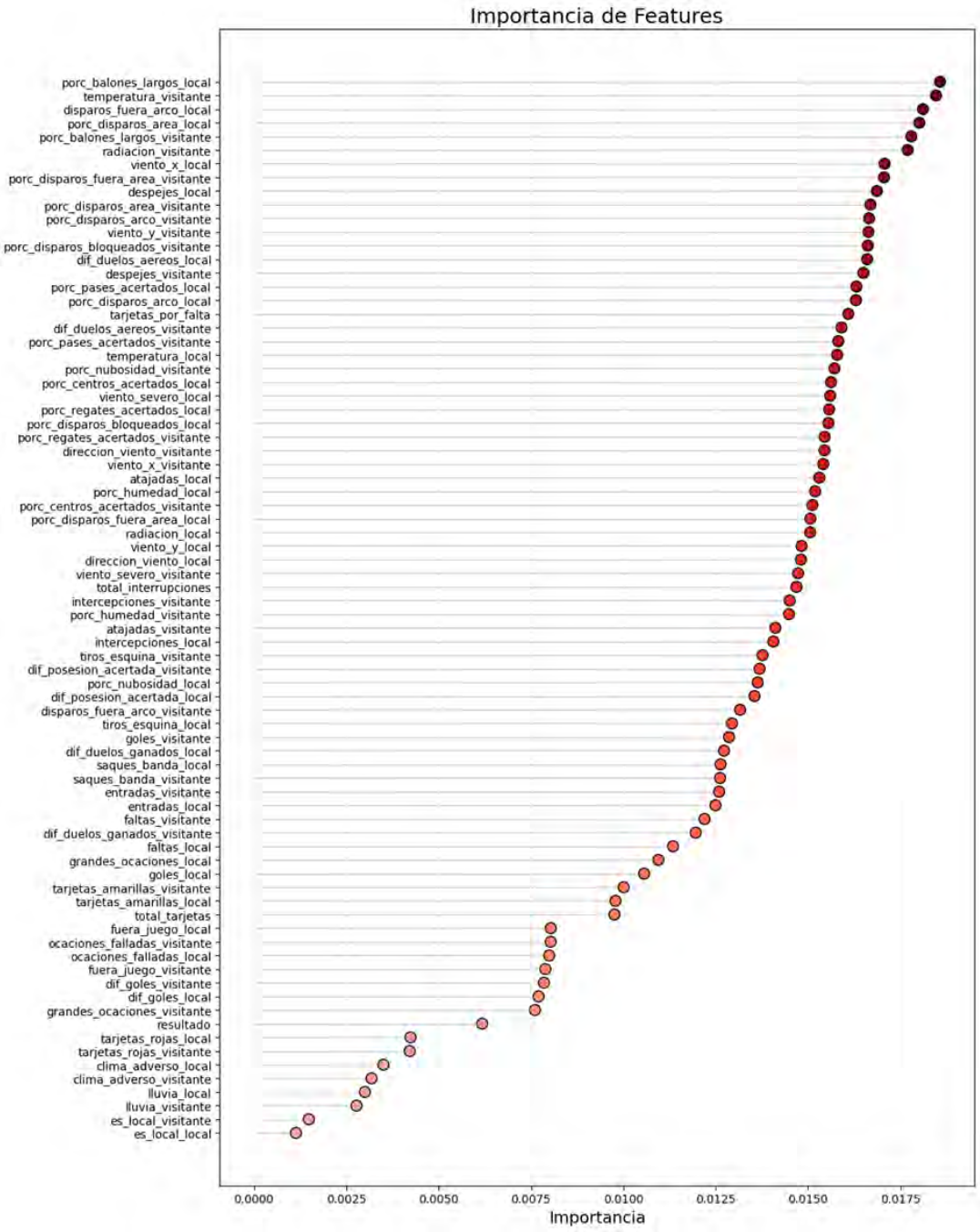
Respecto al segundo objetivo específico, vinculado al papel de las condiciones meteorológicas, el modelo identificó ciertos factores ambientales con contribución apreciable, aunque en menor magnitud en comparación con las variables estrictamente deportivas. La temperatura asociada al equipo visitante aparece entre las variables con mayor aporte dentro de esta categoría, lo que podría reflejar diferencias en la aclimatación o en la respuesta fisiológica frente a determinados contextos térmicos. Otras variables climáticas como aquellas relacionadas con humedad, viento o precipitación muestran un peso inferior y menos consistente, lo que sugiere que su influencia sobre el resultado es más tenue o indirecta dentro del marco del modelo empleado.

En conjunto, estos resultados permiten concluir que, dentro de la aproximación basada en árboles de decisión, las características derivadas del rendimiento futbolístico

inmediato concentran la mayor capacidad explicativa, mientras que las condiciones ambientales aportan información adicional, pero de forma más moderada.

Figura 16

Ranking de importa de variables de futbol y ambientales



5.5. Comparación Directa Entre Experimentos

5.5.1 Arquitectura y Complejidad del Modelo

Existe una diferencia sustancial en la topología de red seleccionada por el proceso de optimización para cada caso.

• Experimento F-1 (Solo Fútbol): La búsqueda de hiperparámetros seleccionó una arquitectura basada en GRU (Gated Recurrent Units) con 24 unidades en una capa oculta.

• Experimento F-A1 (Con Variables Ambientales): La incorporación de datos meteorológicos derivó en una arquitectura más minimalista basada en celdas LSTM (Long Short-Term Memory) con solo 8 unidades.

La inclusión de variables climáticas permitió una reducción de la complejidad estructural del modelo (de 24 a 8 unidades), sugiriendo que la información meteorológica aporta una riqueza representacional que permite al modelo aprender con menos neuronas, manteniendo parámetros de optimización idénticos (tasa de aprendizaje 0.001 y Dropout 50%).

5.5.2 Dinámica de Entrenamiento

El comportamiento durante el aprendizaje mostró divergencias en estabilidad.

• Convergencia: El modelo F-1 mostró una estabilización suave de la pérdida de validación alrededor de 1.08 tras 17 épocas. Por el contrario, el modelo F-A1, aunque alcanzó niveles similares de pérdida (1.085), exhibió oscilaciones más pronunciadas y una mayor sensibilidad a la composición de los lotes, requiriendo reducciones más agresivas en la tasa de aprendizaje (hasta $5x10^{-4}$) para estabilizarse.

5.2.3. Desempeño Predictivo Global vs. Local (Perú)

A nivel global, la comparación arroja resultados contraintuitivos, pero al desagregar por el objetivo de la investigación (Perú), la diferencia es marcada.

• Nivel Global: El modelo F-1 alcanzó una exactitud del 43.7%, superando marginalmente al modelo F-A1 que obtuvo un 43.5%. Esta diferencia de 0.2% es estadísticamente insignificante, indicando que, en promedio para toda Sudamérica, el clima no garantiza una mejora universal.

• Caso Perú (Objetivo central): Aquí radica la diferencia crítica. Mientras que el modelo F-1 logró una exactitud del 45.2% en Perú, el modelo F-A1 elevó esta métrica al 47.8%. Esto representa una mejora relativa del 5.8% en la capacidad predictiva específica para el contexto peruano al incluir datos meteorológicos.

5.2.4. Análisis de Clases (Matriz de Confusión) en Perú

La calidad de la predicción cambió drásticamente en la identificación de resultados específicos en Perú:

• Victorias Visitantes: El modelo F-1 tenía una sensibilidad baja de 0.32 para detectar victorias visitantes. El modelo F-A1 casi duplicó esta capacidad, subiendo a 0.61.

• Empates: Ambos modelos luchan con esta clase, pero el F-A1 mejoró la discriminación de empates en Perú de 0.58 a 0.60.

5.2.5. Análisis de métricas

La comparación entre ambos experimentos revela una mejora consistente en el desempeño predictivo cuando se incorporan variables meteorológicas al modelo. El Experimento ’F - 1’, basado exclusivamente en variables deportivas, alcanzó una exactitud promedio de 0.427 con variabilidad notable entre países: Perú (0.452), Ecuador (0.415), Colombia (0.434) y Chile (0.405), este último representando el desempeño más bajo del conjunto. En contraste, el Experimento ‘F – A1’, que integra información ambiental, elevó la exactitud promedio a 0.435 y, significativamente, invirtió el ordenamiento de desempeño por país, situando a Perú como líder con 0.478 y a Colombia como el caso más desafiante con 0.390.

CAPITULO VI

Discusión de resultados

Esta investigación tuvo como objetivo evaluar el impacto de variables meteorológicas en la predicción de resultados de fútbol mediante redes neuronales recurrentes (GRU y LSTM). A continuación, se contrastan los hallazgos de los experimentos "F-1" (solo fútbol) y "F-A1" (fútbol + clima) con la evidencia científica reciente en la región y el mundo.

4.1 Influencia de las Variables Meteorológicas en el Perú (Objetivo 1)

Los resultados confirman que la influencia de las variables meteorológicas es positiva y significativa específicamente para el caso peruano, a diferencia de otros contextos geográficos evaluados.

La mejora de la exactitud en Perú (del 45.2% al 47.8%) y el incremento sustancial en el puntaje F1-macro (de aprox. 0.37 a 0.41) sugieren que la diversidad climática del territorio peruano introduce "señales" discriminativas valiosas que no están presentes en las variables puramente deportivas. Es notable cómo la inclusión del clima permitió al modelo predecir mejor las victorias visitantes (incremento de sensibilidad del 90% relativo). Esto podría interpretarse como la capacidad de la red neuronal para identificar condiciones climáticas adversas o específicas que rompen la tradicional "localía" o ventaja de casa, permitiendo al visitante ganar.

En contraste, el deterioro del modelo en Colombia (caída del 39.0% en exactitud) indica que la influencia meteorológica no es universalmente beneficiosa y puede introducir ruido en contextos donde el clima es más homogéneo o menos determinante para el juego.

4.3. Heterogeneidad Geográfica: El Clima como Predictor Contextual

Uno de los hallazgos más notables de esta investigación es que la inclusión de variables meteorológicas no generó una mejora universal, sino altamente dependiente del contexto geográfico. Mientras que a nivel global la exactitud se mantuvo estancada (43.7% vs 43.5%), en el caso específico de Perú se observó un incremento significativo del desempeño (de 45.2% a 47.8%).

Este comportamiento dual encuentra respaldo en la literatura contradictoria sobre el tema. Por un lado, (Stevens, 2024), al analizar la Copa Mundial Femenina, concluyó que agregar variables climáticas no mejoraba la precisión del modelo (manteniéndose en 0.65), sugiriendo que en torneos globales la señal climática se diluye. Sin embargo, (Ditsuhi Iskandaryan , Francisco Ramos, 2020) demostraron que, en ligas nacionales específicas como la española, la integración de datos meteorológicos sí mejora significativamente la predicción.

Nuestros resultados sugieren que el Perú se comporta de manera similar al caso español documentado por Iskandaryan, donde el clima es un factor determinante. Esto se alinea con la revisión de (Sarah Illmer; Frank Daumann, 2022), quienes indican que factores ambientales extremos, especialmente la altitud y el calor, afectan la distancia total recorrida y la intensidad de las carreras de los jugadores. Dado que el Perú posee una geografía diversa con ciudades de altura, es coherente que el modelo $F - A1$ logre capturar patrones latentes asociados a la fatiga física descrita en la literatura, mejorando la predicción de victorias visitantes en un 90% relativo.

4.4. Comparativa de Desempeño Predictivo en la Región (Colombia y Chile)

Al situar nuestros resultados en el contexto sudamericano, las métricas obtenidas son competitivas y, en algunos casos, superiores a los benchmarks locales.

Caso Chile: Contrastamos nuestros hallazgos con el Experimento 2 de Ovando Fuentealba (2025), el cual arrojó el mejor desempeño de su investigación utilizando un modelo Random Forest con características seleccionadas por importancia y ventanas móviles. Mientras que el mejor modelo de experimento de Ovando alcanzó una exactitud del **41.5%** en el conjunto de prueba, nuestro modelo base ($F - 1$) logró un **40.5%**, y la incorporación de variables meteorológicas ($F - A1$) elevó el rendimiento al 43.2% para chile. Esto demuestra que la arquitectura recurrente (LSTM/GRU) alimentada con datos climáticos supera a los métodos de ensamble (Random Forest/XGBoost) utilizados en Chile, incluso cuando estos últimos intentaron optimizaciones complejas de balanceo de datos (Experimentos 3 y 4) que resultaron en una caída del rendimiento (hacia el 34-35%).

Caso Colombia: El desempeño en Colombia fue el punto crítico de nuestro experimento, donde la inclusión del clima deterioró la exactitud del 43.4% al 39.0%. Esto resuena con la investigación de (Bustos, 2023) en la liga colombiana, quien reportó dificultades para superar el 43% de exactitud incluso con modelos SVM y ELO, citando limitaciones en la calidad de los datos. La caída en el rendimiento de nuestro modelo $F-A1$ en Colombia podría explicarse por la homogeneidad climática relativa de ciertas regiones colombianas o la falta de precisión en los datos meteorológicos locales, un limitante también señalado por Stevens (2024) como causa de la falta de mejora en sus modelos.

4.5. Eficiencia Arquitectónica y Adaptación Fisiológica

Desde una perspectiva técnica, el experimento $F-A1$ demostró que la inclusión de datos ambientales permitió reducir la complejidad de la red de 24 unidades GRU a solo 8 unidades LSTM. Esto sugiere que las variables climáticas actúan como "atajos" informativos que reducen la carga computacional necesaria para encontrar patrones.

Esta simplificación arquitectónica tiene un correlato biológico. (Walker J. Ross; Madeleine Orr, 2022) y (Bustos, 2023) establecen que existen condiciones límite (ej. temperaturas sobre 28°C o alta humedad) que obligan a los jugadores a modular su actividad física para preservar el rendimiento. Al alimentar al modelo con datos explícitos de temperatura o precipitación, la red neuronal no necesita "inferir" estas condiciones adversas a partir de las estadísticas de juego, sino que puede asociar directamente condiciones extremas (como las descritas por Ross y Orr para eventos futuros) con una mayor probabilidad de errores defensivos o baja intensidad, facilitando la predicción de resultados sorpresivos (como victorias visitantes).

4.6. Limitaciones y El Problema de la Universalidad

A pesar del éxito en Perú, la falta de mejora global concuerda con lo observado por Stevens (2024). Esto indica que la influencia del clima no es una regla universal en el fútbol, sino una variable de interacción local. Como señalan (Sarah Illmer; Frank Daumann, 2022), aunque el clima afecta el rendimiento físico, los equipos profesionales adoptan "estrategias de ritmo" (pacing strategies) para mantener el rendimiento técnico (pases, posesión) a pesar del estrés ambiental. Es posible que, en ligas donde los equipos están mejor adaptados o las condiciones son menos extremas (como podría ser el caso de los datos de Colombia en nuestra muestra), esta adaptación técnica anule la ventaja predictiva de las variables meteorológicas.

4.7. Conclusión de la Discusión

La investigación valida que, tal como sugieren Iskandaryan et al. (2020), el clima contiene información valiosa para la predicción deportiva, pero este valor está condicionado geográficamente. Nuestros modelos superan los umbrales de exactitud reportados recientemente para Chile y Colombia, demostrando que una arquitectura LSTM ligera alimentada con datos ambientales es una estrategia efectiva para contextos de alta variabilidad climática y geográfica como el Perú, aunque su eficacia disminuye en entornos donde las variables físicas no son tan determinantes para el resultado final.

Conclusiones

1. La incorporación de variables meteorológicas en modelos de redes neuronales recurrentes no garantiza una mejora universal, sino que su eficacia es altamente dependiente del contexto geográfico. A nivel global, el desempeño predictivo se mantuvo estable (variación marginal de 43.7% a 43.5%). Sin embargo, en el caso específico del Perú (territorio caracterizado por su alta variabilidad altitudinal y climática) la inclusión de estos datos generó un incremento significativo en la exactitud del 45.2% al 47.8% (una mejora relativa del 5.8%). Esto permite concluir que la información ambiental actúa como un discriminador eficaz únicamente en regiones donde las condiciones climáticas son lo suficientemente heterogéneas como para influir en el desarrollo del juego
2. Sobre las variables estadísticas deportivas más informativas, mediante técnicas de aprendizaje supervisado basadas en Bosques Aleatorios (con una precisión de validación del 47.8%), se determinó que las dinámicas ofensivas inmediatas poseen la mayor carga de información (ganancia). Las variables más determinantes en primer lugar fueron el porcentaje de balones largos del equipo local, segundo lugar la proporción de disparos dentro del área y como tercer lugar la frecuencia de remates desviados. Esto indica que la capacidad del modelo para predecir el resultado depende primariamente de métricas que reflejan la orientación táctica y la finalización de jugadas, superando en relevancia a las variables de posesión o defensivas.
3. Sobre la influencia de elementos meteorológicos, el análisis de importancia de características reveló que la influencia de las variables meteorológicas es moderada en comparación con las deportivas, siendo la ventana de temperatura asociada al equipo visitante el predictor ambiental más relevante. Variables ventanas como la humedad, el viento o la precipitación mostraron un peso inferior. Esto sugiere que el modelo captura patrones relacionados con la aclimatación física o el estrés térmico que sufren los equipos visitantes, más que efectos directos sobre

la física del balón (viento), lo cual es coherente con la mejora en la detección de victorias visitantes observada en los experimentos.

4. Existe una heterogeneidad marcada en el desempeño del modelo según el país, revelando sesgos geográficos operativos. Perú demostró ser el contexto más beneficiado, alcanzando los indicadores más robustos (F1-macro: 0.41) y mejorando sustancialmente la discriminación de la clase "Victoria Visitante" (+90% relativo). Por otra parte, Chile y Ecuador mostraron mejoras modestas en exactitud (+2.7 y +1.9 puntos porcentuales respectivamente). Sin embargo, Colombia presentó un deterioro notable (-4.4 puntos en exactitud), sugiriendo que en este contexto las variables meteorológicas introdujeron ruido en lugar de señal. Esta variabilidad confirma que los modelos de predicción deportiva no son "talla única" y requieren calibración específica según la diversidad climática del territorio objetivo.
5. El análisis comparativo entre el Modelo A, basado exclusivamente en variables deportivas y optimizado mediante 24 unidades GRU, y el Modelo B, que incorpora variables meteorológicas con arquitectura simplificada de 8 unidades LSTM, revela que la inclusión de información ambiental permite alcanzar capacidad predictiva comparable. Esta simplificación arquitectural, manteniendo idénticos parámetros de regularización y optimización (dropout, Adam, lotes de 128 instancias), sugiere que las variables meteorológicas aportan estructura informativa complementaria que facilita la discriminación de patrones de desempeño deportivo. La divergencia en el tipo de celda recurrente (GRU versus LSTM) y la dramática reducción dimensional constituyen evidencia empírica de que las condiciones ambientales capturan variabilidad ortogonal respecto a las estadísticas futbolísticas, reduciendo la complejidad necesaria mientras se mejora la generalización.

Recomendaciones

1. Se recomienda establecer colaboraciones con federaciones deportivas nacionales para instalar estaciones meteorológicas permanentes en estadios representativos de diferentes zonas climáticas. Estas estaciones deben registrar temperatura ambiente y del césped, humedad relativa, velocidad y dirección del viento a nivel de superficie, radiación solar y precipitación con resolución de un minuto. La disponibilidad de datos in situ permitiría validar las estimaciones satelitales, calibrar factores de corrección específicos por estadio y capturar fenómenos micro climáticos no detectables remotamente.
2. Se sugiere implementar procesos de cruce de estadísticas mediante comparación sistemática entre múltiples proveedores de datos. Las discrepancias significativas deberían activar procesos de revisión manual por analistas deportivos. Para variables subjetivas, se recomienda establecer protocolos estandarizados de registro y programas de entrenamiento para operadores, siguiendo estándares establecidos por ligas profesionales europeas.
3. Para el contexto peruano específico, se recomienda priorizar la ingeniería de características sobre variables tácticas y físicas observables (cambios estratégicos durante el partido, patrones de presión defensiva, transiciones rápidas) sobre refinamientos meteorológicos. La inversión en análisis de video asistido por visión por computadora para extraer métricas avanzadas de posicionamiento y movimiento colectivo probablemente genere mayor retorno predictivo que datos climáticos adicionales.
4. Los sistemas predictivos destinados a uso profesional por casas de apuestas o cuerpos técnicos deberían implementar lógica condicional que active o desactive módulos meteorológicos según características del contexto: activación en ligas con alta heterogeneidad climática (Chile, Argentina, Colombia) y desactivación en ligas homogéneas (Perú, Uruguay). Esta estrategia optimiza el balance sesgo-varianza del modelo según las características informativas del dominio.

5. Otra dirección prometedora consiste en explorar arquitecturas de modelado temporal más complejas, como Transformers o modelos híbridos que integren señales meteorológicas, dinámicas tácticas y métricas fisiológicas. Estas alternativas permitirían contrastar si el buen desempeño de las redes LSTM en territorios de alta heterogeneidad climática se debe a su capacidad para capturar dependencias no lineales o si arquitecturas más recientes pueden mejorar esa eficiencia.
6. Asimismo, resultaría pertinente ampliar el estudio a ventanas temporales más extensas e incluir secuencias longitudinales de carga física, congestión de calendario o viajes interregionales, con el fin de analizar si estos factores moderan el impacto del clima en el rendimiento deportivo. La integración de datos de tracking o métricas de esfuerzo podría permitir evaluar la relación entre aclimatación, fatiga acumulada y probabilidad de resultados atípicos.
7. La replicación del enfoque en otras ligas sudamericanas y europeas con diversidad geográfica contrastante permitiría delimitar con mayor precisión los umbrales ambientales necesarios para que las variables meteorológicas aporten valor predictivo. Este análisis comparado ayudaría a establecer criterios reproducibles para decidir cuándo conviene incorporar información ambiental en modelos de predicción deportiva y cuándo su efecto tiende a diluirse por adaptación táctica o homogeneidad climática.
8. Finalmente integrar sistémicamente variables cinemáticas, como la velocidad de explosión y la capacidad aeróbica, orientadas a refinar la resolución de las predicciones de desempeño. Asimismo, la inclusión de indicadores biométricos de composición corporal y respuesta fisiológica a la altitud, junto con determinantes psicométricos como el estado anímico y los niveles de motivación, permitiría una caracterización integral y multidimensional del deportista.

Referencias

Roger Grosse ; Jimmy Ba's. (2017). *Neural Networks and Deep Learning.* Class at the University of Toronto.

Alice Zheng; Amanda Casari. (2018). *Feature Engineering for Machine Learning.* O'Reilly Media, Inc.

Amadu, P. (2024, Octubre 4). Short Survey in Machine Learning for Soccer Analytics. *Preprints*. doi:10.20944/preprints202410.0178.v1

Amaya Gómez; Luis Ángel. (2022). El discurso del periodismo deportivo televisivo en el Perú y la incorporación de elementos de la identidad nacional. *Universidad Peruana de Ciencias Aplicadas (UPC)*.

Ariana Yunita ; MHD Iqbal Pratama. (2025). Performance analysis of neural network architectures for time series forecasting: A comparative study of RNN, LSTM, GRU, and hybrid models. *MethodsX, 15*. doi:https://doi.org/10.1016/j.mex.2025.103462

Bernal, T. C. (2010). *Metodología de la investigación.* Bogotá : PEARSON EDUCACIÓN,.

Bustos, E. S. (2023). *Sistema de predicción de resultados para los partidos de fútbol de la Liga Profesional Colombiana.* Universidad Jorge Tadeo Lozano, Maestría en Ingeniería y Analítica de Datos. Bogota: Facultad De Ciencias Naturales E Ingeniería. doi:20.500.12010/34043

Carrasco, D. S. (2019). *Metodología de la Investigación Científica.* Lima: San Marcos.

Center, N. L. (2025, 10 7). *NASA Prediction of Worldwide Energy Resources (POWER) Project*. Retrieved from NASA Prediction of Worldwide Energy Resources (POWER) Project: ttps://power.larc.nasa.gov/

Daniel Carrilho ; Micael Santos Couceiro; João Brito ; Pedro Figueiredo ; Rui J. Lopes ; Duarte Araújo. (n.d.). Using Optical Tracking System Data to Measure Team Synergic Behavior: Synchronization of Player-Ball-Goal Angles in a Football Match. *Sensor, 20*(17). doi:https://doi.org/10.3390/s20174990

Daniel Memmert ; Dominik Raabe. (2023). *Data Analytics in Football: Positional Data Collection, Modelling and Analysis* (Vol. 2). London: Routledge. doi:0815381549

Ditsuhi Iskandaryan , Francisco Ramos. (2020). The Effect of Weather in Soccer Results: An Approach. (A. Sciences, Ed.) *Computational Intelligence and Data Mining in Sports, 10*(19). doi:10.3390/app10196750

Escalona, G. Á. (2021). *Del barrio al estadio: identidad y espectáculo en el fútbol peruano.* Lima: Fondo Editorial PUCP.

FUENTEALBA, R. O. (2025). *PREDICCIÓN DE RESULTADOS DE PARTIDOS DE LA LIGA PROFESIONAL DEFUTBOL CHILENO USANDO ALGORITMOS DE MACHINE LEARNING.* Universidad del desarrollo. Santiago: Universidad del Desarrollo. Facultad de Ingeniería. Retrieved from https://hdl.handle.net/11447/9941

Gómez, P. G., & Reyes, A. M. (2024). *Métodos estadísticos aplicados a la predicción de.* Sevilla: Universidad de Sevilla.

Hernández, S. R., Fernández, C. C., & Baptista, L. M. (2014). *Metodologia de la Investigacion.* Mexico: McGRAW-HILL.

Hernandez-Sampieri, R., & Mendoza, T. C. (2019). *Metodología de la Investigación.* mexico: Mc Graw Hill.

Huyen, C. (2022). *Designing Machine Learning Systems.* O'Reilly Media, Inc. doi:9781098107956

Ian Goodfellow ; Yoshua Bengio ; Aaron Courville. (2016). *Deep Learning.* MIT Press. Retrieved from http://www.deeplearningbook.org

International, D. (2017). *DAMA-DMBOK: Data Management Body of Knowledge* (2 ed.). Denville, NJ, USA: Technics Publications, LLC. doi:1634622340

Jhonny Francisco Segovia Romero; Joseph Taro. (2025). Influencia ambiental en las respuestas fisiológicas durante el entrenamiento de deportistas élite de orientación. *6*, 39–48. doi:https://doi.org/10.5281/zenodo.15867184

Jimenez, I. V. (2023). *Sistema para pronosticar resultados de partidos de fútbol en opciones dobles.* Universidad de Lima. Lima: Repositorio Institucional ULima.

Mazi Essoloani Aleza; D. Vetrithangam. (2023). Use of Artificial Intelligence to Avoid Errors in Referring a Football Match. *Artificial Intelligence and Applications (ICAIA), International Conference on, Technology Conference (ATCON-1), Alliance*, 1 - 6. doi:10.1109/ICAIA57370.2023.10169463

Nallapa, V. S. (2022). Predicting Soccer Match Outcomes Using Deep Learning (LSTM). *Intemational Journal of Science and Research (IJSR), 11*, 1454 - 1458. doi:10.21275/SR22108120231

Niek Tax; Niek Tax. (2015). Predicting The Dutch Football Competition Using Public Data: A Machine Learning Approach. *TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, X*(X).

Philo U Saunders ; David B Pyne ; Christopher J Gore. (2009). Endurance training at altitude. *National Center for Biotechno/ogy Information, 10*, 135-148. doi:10.1089/ham.2008.1092

René Manassé Galekwa; Jean Marie Tshimula; Etienne Gael Tajeuna; Kyamakya Kyandoghere. (2024). A Systematic Review of Machine Learning in Sports Betting: Techniques, Challenges, and Future Directions. *arXiv preprint arXiv:2410.21484*. doi:https://doi.org/10.48550/arXiv.2410.21484

Roberto Hernández Sampieri ; Christian Paulina Mendoza Torres. (2023). *Metodología de la investigación: Las rutas cuantitativa, cualitativa y mixta.* Ciudad de México: McGraw-Hill Interamericana S.A.

Ronaldo Kobal ; Irineu Loturco. (2022). Effects of moderate altitude on the physical performance of elite female soccer players during an official soccer tournament. *International Journal of Sports Science & Coaching*, 1558–1566. doi:https://doi.org/10.1177/1747954122111714

Rory Bunker, Calvin Yeung, Keisuke Fujii. (2024). Machine Learning for Soccer Match Result Prediction. *Cornell University*. doi:10.48550/arXiv.2403.07669

Rory Bunker; Calvin Yeung; Teo Susnjak; Chester Espie; Keisuke Fujii. (2023). A comparative evaluation of Elo ratings- and machine learning-based methods for

tennis match result prediction. *Proceedings of the Institution of Mechanical Engineers, Part P: Journal of Sports Engineering and Technology*, 305–316. doi:10.1177/17543371231212235

Sarah Illmer; Frank Daumann. (2022). The effects of weather factors and altitude on physical and technical performance in professional soccer: A systematic review. *JSAMS Plus, 1*(100002). doi:https://doi.org/10.1016/j.jsampl.2022.100002

Spyridon Plakias ; Themistoklis Tsatalas ; Xenofon Betsios; Giannis Giakas. (2025). A new era in soccer performance analysis research? *Insight -Sports Science, 7*(741). doi:https://doi.org/10.18282/iss741

Stevens, H. (2024). *redicting the outcome of Women's World Cupmatches taking weather conditions into account,using K-Nearest Neighbors, Random Forest andSupport Vector Machines.* Tilburg University, Department of Cognitive Science & Artificial Intelligence. Tilburg, The Netherlands: School of Humanities and Digital Sciences.

Sungchan Hong ; Ryosuke Nobori. (2016). Experiment of Aerodynamic Force on a Rotating Soccer Ball. *Procedia Engineering, 147*, 56 - 61. doi:https://doi.org/10.1016/j.proeng.2016.06.189

Supo, C. J. (2024). *Metodología de laInvestigación Científica.* Arequipa: BIOESTADISTICO EEDU.EIRL.

Walker J. Ross; Madeleine Orr. (2022). Predicting climate impacts to the Olympic Games and FIFA Men's World Cups from 2022 to 2032. *Sport in Society, 25*(4). doi:10.1080/17430437.2021.1984426

Wilton W Fok; Louis C Chan; Carol Chen. (2018). Artificial Intelligence for Sport Actions and Performance Analysis using Recurrent Neural Network (RNN) with Long Short-Term Memory (LSTM). *ICRAI '18: Proceedings of the 4th International Conference on Robotics and Artificial Intelligence*, 40-44. doi:https://doi.org/10.1145/3297097.3297115

Anexos

- Obtener Ventana de Equipo

```
def get_racha_equipo(historial: pd.DataFrame, equipo_id: int,
fecha_actual: datetime, ventana=5):
    """
    Extrae la racha (últimos N partidos) de un equipo antes de una fecha
dada.
    El DataFrame debe tener el índice como datetime.
    """

    # Filtrar partidos del equipo antes de la fecha actual
    historial_equipo = historial[
        (historial['id_team'] == equipo_id) &
        (historial.index < fecha_actual)
    ].sort_index(ascending=False)

    if len(historial_equipo) < ventana:
        return None

    racha = historial_equipo.iloc[:ventana].copy()

    # Calcular diferencia en días respecto al partido actual
    #racha['delta_dias'] = (fecha_actual - racha.index).days

    # Excluir columnas innecesarias
    columnas_usar = FEATURES_RACHA_EQUIPO

    # Retornar los valores en orden cronológico (más reciente primero)
    return racha[columnas_usar].values
```

- Obtener Ventana de Arbitro

```
def get_ventana_arbitro(df_partidos, id_arbitro, fecha_actual,
ventana=5):
    df_enviroment = df_partidos[
        (df_partidos['id_arbitro'] == id_arbitro) &
        (df_partidos.index < fecha_actual)
    ].sort_index(ascending=False)


    if len(df_enviroment) < ventana:
        return None

    enviroment = df_enviroment.iloc[:ventana].copy()

    columnas_usar = FEATURES_ARBITRO

    return enviroment[columnas_usar].values
```

- Obtener Ventana de Ambiente

```
def get_ventana_ambiente(df_partidos, team, fecha_actual, ventana=5):
```

```
    df_enviroment = df_partidos[
        ((df_partidos['id_team_local'] == team) |
         (df_partidos['id_team_visita'] == team))  &
        (df_partidos.index < fecha_actual)
    ].sort_index(ascending=False)

    if len(df_enviroment) < ventana:
        return None

    enviroment = df_enviroment.iloc[:ventana].copy()

    columnas_usar = FEATURES_AMBIENTE

    # el más reciente esta primero
    return enviroment[columnas_usar].values
```

- Construye valores X como ventana y el target

```
def construir_X_y(df_partidos, historial, ventana):
    """
    Construye los arrays X (inputs) e y (targets) para entrenamiento o
evaluación.

    Parámetros:
    - df_partidos: DataFrame de partidos a usar (ej: df_train o df_test)
    - historial: DataFrame con partidos históricos de todos los equipos
    - ventana: cuántos partidos anteriores usar por equipo

    Retorna:
    - X: array de shape (n_partidos, ventana, n_features_totales)
    - y: array de shape (n_partidos,)
    """
    X = []
    y = []

    # Ordenar
    df_partidos = df_partidos.sort_index()

    # Extraer ventana de cada partido
    for idx, row in df_partidos.iterrows():
        # Enviroment
        id_partido = row['id_partido']
        id_arbitro = row['id_arbitro']
        eq_local = row['id_team_local']
        eq_visita = row['id_team_visita']
        fecha_actual = idx
        resultado = row['resultado']            #RESULTADO es respecto al
local

        # Obtenemos las rachas (últimos n partidos previos)
        racha_local = get_racha_equipo(historial, eq_local, fecha_actual,
ventana)
```

```
        racha_visita = get_racha_equipo(historial, eq_visita,
fecha_actual, ventana)
        enviroment_local = get_ventana_ambiente(df_partidos, eq_local,
fecha_actual, ventana)   #Estamos extrando la ventana de ambiente en base
al equipo local
        enviroment_visita = get_ventana_ambiente(df_partidos, eq_visita,
fecha_actual, ventana)
        arbitro = get_ventana_arbitro(df_partidos, id_arbitro,
fecha_actual, ventana)

        # Si alguno de los equipos no tiene suficientes partidos previos,
se omite
        if racha_local is None or racha_visita is None or
enviroment_local is None or enviroment_visita is None or arbitro is None:
            continue

        #print(enviroment_local.shape, racha_local.shape, arbitro.shape)
        # Concatenamos la racha local y la visitante horizontalmente (por
timestep)
        racha_completa = np.concatenate([racha_local, enviroment_local,
racha_visita, enviroment_visita, arbitro], axis=1)  # shape: (ventana,
n_features_local + n_features_visita)

        X.append(racha_completa)
        y.append(resultado)

    X = np.array(X)
    y = np.array(y)

    #print(X.shape)

    print(f"Construidos {X.shape[0]} muestras con shape {X.shape[1:]}
(ventana={ventana}, features={X.shape[2]})")

    return X, y
```

- Entrenamiento y resultados

```
class RNNGridSearchOptimizer:
    """
    Optimizador de Grid Search para modelos RNN con gestión eficiente de
GPU.

    Características:
    - Limpieza automática de sesión entre experimentos
    - Re-establecimiento de semilla para reproducibilidad
    - Soporte para LSTM, GRU, Bidirectional
    - Class weights automáticos
    - Mixed Precision Training
    """

    def __init__(self, n_classes: int = 3, seed: int = SEED):
        """
        Inicializa el optimizador.

        Args:
            n_classes: Número de clases de salida
            seed: Semilla para reproducibilidad
        """
        self.n_classes = n_classes
        self.seed = seed
        self.param_grid = {}
        self.results = []
        self.best_score = -np.inf
        self.best_params = None
        self.best_model = None
        self.best_model_history = None  # Nuevo: Para guardar historial
del mejor modelo
        self.class_weights = None

        print(f"  RNNGridSearchOptimizer inicializado")
        print(f"  Clases: {n_classes}")
        print(f"  Semilla: {seed}")

    def set_param_grid(self, param_grid: Dict[str, List]) -> None:
        """Define el espacio de búsqueda de hiperparámetros."""
        self.param_grid = param_grid
        total_combinations = np.prod([len(v) for v in
param_grid.values()])
        print(f"✓ Grid de parámetros configurado: {total_combinations}
combinaciones")

    def prepare_target_data(
        self,
        y_train: np.ndarray,
        y_val: np.ndarray,
        y_test: np.ndarray
    ) -> Tuple[np.ndarray, np.ndarray, np.ndarray]:
        """
```

```
        Prepara datos de salida (one-hot encoding) y calcula class
weights.

        Args:
            y_train, y_val, y_test: Arrays de etiquetas (enteros)

        Returns:
            Tupla de arrays one-hot encoded
        """
        # Calcular class weights solo una vez
        if self.class_weights is None:
            self.class_weights = calcular_class_weights(y_train)

        # One-hot encoding
        y_train_cat = to_categorical(y_train, num_classes=self.n_classes)
        y_val_cat = to_categorical(y_val, num_classes=self.n_classes)
        y_test_cat = to_categorical(y_test, num_classes=self.n_classes)

        return y_train_cat, y_val_cat, y_test_cat

    def _build_model(self, params: Dict, input_shape: Tuple) ->
Sequential:
        """Construye modelo RNN según parámetros."""
        model = Sequential(name=f"{params['cell_type']}_model")

        # Seleccionar tipo de celda recurrente
        RecurrentLayer = LSTM if params['cell_type'] == 'LSTM' else GRU

        # Primera capa recurrente
        first_layer = RecurrentLayer(
            units=params['units_layer1'],
            return_sequences=params.get('units_layer2', 0) > 0,
            name=f"{params['cell_type']}_1"
        )

        if params.get('bidirectional', False):
            first_layer = Bidirectional(first_layer,
name='bidirectional_1')

        model.add(first_layer)
        model.add(Dropout(params['dropout_rate'], name='dropout_1'))

        # Segunda capa recurrente (opcional)
        if params.get('units_layer2', 0) > 0:
            second_layer = RecurrentLayer(
                units=params['units_layer2'],
                return_sequences=False,
                name=f"{params['cell_type']}_2"
            )

            if params.get('bidirectional', False):
                second_layer = Bidirectional(second_layer,
name='bidirectional_2')

            model.add(second_layer)
            model.add(Dropout(params['dropout_rate'], name='dropout_2'))
```

```
        # Capa densa oculta (opcional)
        if params.get('dense_units', 0) > 0:
            model.add(Dense(params['dense_units'], activation='relu',
name='dense_hidden'))
            model.add(Dropout(params['dropout_rate'],
name='dropout_dense'))

        # Capa de salida
        model.add(Dense(self.n_classes, activation='softmax',
name='output', dtype='float32'))

        # Optimizador
        optimizer_name = params.get('optimizer', 'adam').lower()
        if optimizer_name == 'adam':
            optimizer = Adam(learning_rate=params['learning_rate'])
        elif optimizer_name == 'rmsprop':
            optimizer = RMSprop(learning_rate=params['learning_rate'])
        elif optimizer_name == 'sgd':
            optimizer = SGD(learning_rate=params['learning_rate'],
momentum=0.9)
        else:
            optimizer = Adam(learning_rate=params['learning_rate'])

        # Compilar modelo
        model.compile(
            loss='categorical_crossentropy',
            optimizer=optimizer,
            metrics=['accuracy']
        )

        return model

    def grid_search(
        self,
        X_train: np.ndarray,
        y_train: np.ndarray,
        X_val: np.ndarray,
        y_val: np.ndarray,
        X_test: np.ndarray,
        y_test: np.ndarray,
        metric: str = 'f1_macro',
        verbose: int = 1
    ) -> Dict:
        """
        Args:
            X_train, y_train: Datos de entrenamiento
            X_val, y_val: Datos de validación
            X_test, y_test: Datos de prueba
            metric: Métrica para seleccionar mejor modelo
            verbose: Nivel de verbosidad

        Returns:
            Diccionario con mejores parámetros, modelo y resultados
        """
        if not self.param_grid:
            raise ValueError("Debe definir param_grid usando
set_param_grid()")
```

```
        # Preparar datos
        y_train_cat, y_val_cat, y_test_cat = self.prepare_target_data(
            y_train, y_val, y_test
        )

        # Generar combinaciones de parámetros
        keys = list(self.param_grid.keys())
        values = [self.param_grid[k] for k in keys]
        param_combinations = [dict(zip(keys, v)) for v in
itertools.product(*values)]

        total_combinations = len(param_combinations)

        if verbose > 0:
            print(f"\n{'='*70}")
            print(f"Iniciando Grid Search con {total_combinations}
combinaciones")
            print(f"Métrica de optimización: {metric}")
            print(f"{'='*70}\n")

        # Archivo temporal para guardar el mejor modelo
        temp_best_model_path = 'temp_best_model_grid_search.h5'

        for idx, params in enumerate(param_combinations, 1):
            if verbose > 0:
                print(f"\n[{idx}/{total_combinations}] Evaluando
configuración:")
                print(f"  {params}")

            try:
                limpiar_sesion()
                set_global_seed(self.seed)

                # Construir y entrenar modelo
                model = self._build_model(params,
input_shape=X_train.shape[1:])

                history = entrenar_modelo(
                    model=model,
                    X_train=X_train,
                    y_train=y_train_cat,
                    X_valid=X_val,
                    y_valid=y_val_cat,
                    batch_size=params.get('batch_size',
BATCH_SIZE_OPTIMO),
                    epochs=params.get('epochs', EPOCHS_DEFAULT),
                    patience=params.get('patience', PATIENCE_DEFAULT),
                    class_weights=self.class_weights,
                    verbose=0 if verbose == 0 else 1
                )

                # Evaluar en validación y prueba
                val_metrics = evaluar_y_reportar(
                    model, X_val, y_val_cat, plot_confusion_matrix=False
                )
                test_metrics = evaluar_y_reportar(
```

```
                    model, X_test, y_test_cat,
plot_confusion_matrix=False
                )

                # Guardar resultados
                result = {
                    'params': params.copy(),
                    'val_metrics': val_metrics,
                    'test_metrics': test_metrics,
                    'final_train_loss': float(history.history['loss'][-
1]),
                    'final_val_loss': float(history.history['val_loss'][-
1]),
                    'epochs_trained': len(history.history['loss']),
                    'timestamp': datetime.now().isoformat()
                }

                self.results.append(result)

                # Actualizar mejor modelo
                current_score = val_metrics[metric]
                if current_score > self.best_score:
                    self.best_score = current_score
                    self.best_params = params.copy()

                    # Guardar modelo en disco inmediatamente para
sobrevivir a clear_session()
                    model.save(temp_best_model_path)
                    self.best_model_history = history.history

                    if verbose > 0:
                        print(f"  ✓ Nuevo mejor modelo encontrado!")
                        print(f"    {metric} validación:
{current_score:.4f}")
                        print(f"    {metric} prueba:
{test_metrics[metric]:.4f}")

                elif verbose > 1:
                    print(f"    {metric} validación:
{current_score:.4f}")
                    print(f"    {metric} prueba:
{test_metrics[metric]:.4f}")

            except Exception as e:
                if verbose > 0:
                    print(f"  ✗ Error durante entrenamiento: {str(e)}")
                continue

        # Cargar el mejor modelo desde el archivo temporal
        if os.path.exists(temp_best_model_path):
            print(f"\nCargando mejor modelo desde
{temp_best_model_path}...")
            # Limpiar sesión una última vez antes de cargar el mejor
modelo
            limpiar_sesion()
```

```
            self.best_model =
keras.models.load_model(temp_best_model_path)

        if verbose > 0:
            print(f"\n{'='*70}")
            print("Grid Search completado")
            print(f"{'='*70}")
            print(f"\nMejor configuración encontrada:")
            if self.best_params:
                for key, value in self.best_params.items():
                    print(f"  {key}: {value}")
                print(f"\nMejor {metric} en validación:
{self.best_score:.4f}")

                best_result = next(
                    r for r in self.results
                    if r['params'] == self.best_params
                )
                print(f"\nMétricas en prueba del mejor modelo:")
                for key, value in best_result['test_metrics'].items():
                    if key not in ['classification_report',
'confusion_matrix']:
                        print(f"  {key}: {value:.4f}")

    def plot_best_model_history(self, figsize: Tuple[int, int] = (12, 5),
save_path: Optional[str] = None) -> None:
        """
        Grafica la evolución de Loss y Learning Rate del mejor modelo.

        Args:
            figsize: Tamaño de la figura
            save_path: Ruta donde guardar la imagen. Si es None, no
guarda.
        """
        if self.best_model_history is None:
            print("No hay historial disponible para graficar.")
            return

        history = self.best_model_history
        epochs = range(1, len(history['loss']) + 1)

        plt.figure(figsize=figsize)

        # Gráfica 1: Loss vs Epochs
        plt.subplot(1, 2, 1)
        plt.plot(epochs, history['loss'], 'b-', label='Training Loss')
        plt.plot(epochs, history['val_loss'], 'r-', label='Validation
Loss')
        plt.title('Loss Evolution (Early Stopping)', fontsize=12,
fontweight='bold')
        plt.xlabel('Epochs')
        plt.ylabel('Loss')
        plt.legend()
        plt.grid(True, alpha=0.3)

        # Gráfica 2: Learning Rate vs Epochs
        if 'lr' in history:
```

```
            plt.subplot(1, 2, 2)
            plt.plot(epochs, history['lr'], 'g-o', markersize=4,
label='Learning Rate')
            plt.title('Learning Rate Decay', fontsize=12,
fontweight='bold')
            plt.xlabel('Epochs')
            plt.ylabel('Learning Rate')
            plt.yscale('log')  # Escala logarítmica para ver mejor los
cambios
            plt.legend()
            plt.grid(True, alpha=0.3)

        plt.tight_layout()

        # Guardar imagen si se proporciona ruta
        if save_path:
            plt.savefig(save_path, dpi=300, bbox_inches='tight')
            print(f"Gráfica guardada en: {save_path}")

            plt.show()
            plt.yscale('log')  # Escala logarítmica para ver mejor los
cambios
            plt.legend()
            plt.grid(True, alpha=0.3)

        plt.tight_layout()
        plt.show()

    def save_results(self, filepath: str, save_model: bool = True) ->
None:
        """Guarda resultados en JSON y modelo en H5."""
        results_to_save = {
            'best_params': self.best_params,
            'best_score': float(self.best_score),
            'all_results': self.results,
            'n_classes': self.n_classes,
            'param_grid': self.param_grid
        }

        with open(filepath, 'w') as f:
            json.dump(results_to_save, f, indent=2)

        print(f"\nResultados guardados en: {filepath}")

        if save_model and self.best_model is not None:
            model_path = filepath.replace('.json', '.h5')
            self.best_model.save(model_path)
            print(f"Mejor modelo guardado en: {model_path}")

    def evaluate_by_country(self, df_test_paises: pd.DataFrame,
                            df_historial: pd.DataFrame,
                            construir_X_y_func,
                            ventana: int = 4,
                            verbose: int = 1) -> Dict:
        """
        Evalúa el mejor modelo por país.
```

```
        Args:
            df_test_paises: DataFrame normalizado de test que incluye
columna 'pais'
            df_historial: DataFrame de historial de partidos
            construir_X_y_func: Función para construir X e y
            ventana: Tamaño de ventana temporal
            verbose: Nivel de verbosidad

        Returns:
            Diccionario con métricas por país
        """
        if self.best_model is None:
            raise ValueError("Debe ejecutar grid_search() antes de
evaluar por país")

        if 'pais' not in df_test_paises.columns:
            raise ValueError("El DataFrame debe contener la columna
'pais'")

        metricas_por_pais = {}

        if verbose > 0:
            print(f"\n{'='*70}")
            print(f"Evaluando modelo por país")
            print(f"{'='*70}\n")

        for pais in df_test_paises['pais'].unique():
            df_test_pais = df_test_paises[df_test_paises['pais'] == pais]

            if verbose > 0:
                print(f"País: {pais} - Shape: {df_test_pais.shape}")

            # Eliminar la columna 'pais' antes de construir X e y
            X_test_pais, y_test_pais = construir_X_y_func(
                df_test_pais.drop('pais', axis=1),
                df_historial,
                ventana=ventana
            )

            if len(X_test_pais) == 0:
                if verbose > 0:
                    print(f"  ⚠ Sin datos suficientes para {pais}\n")
                continue

            # Predecir
            y_test_pais_cat = to_categorical(y_test_pais,
num_classes=self.n_classes)
            y_pred_probs = self.best_model.predict(X_test_pais,
verbose=0)
            y_pred = np.argmax(y_pred_probs, axis=1)

            # Calcular métricas
            accuracy = accuracy_score(y_test_pais, y_pred)
            f1_macro = f1_score(y_test_pais, y_pred, average='macro',
zero_division=0)
            f1_micro = f1_score(y_test_pais, y_pred, average='micro',
zero_division=0)
```

```
            f1_weighted = f1_score(y_test_pais, y_pred,
average='weighted', zero_division=0)

            # Matriz de confusión
            cm = confusion_matrix(y_test_pais, y_pred)

            # Guardar en diccionario
            metricas_por_pais[pais] = {
                'n_samples': len(X_test_pais),
                'accuracy': accuracy,
                'f1_macro': f1_macro,
                'f1_micro': f1_micro,
                'f1_weighted': f1_weighted,
                'confusion_matrix': cm,
                'y_true': y_test_pais,
                'y_pred': y_pred
            }

            if verbose > 0:
                print(f"  ✓ Muestras: {len(X_test_pais)}")
                print(f"    Accuracy: {accuracy:.4f}")
                print(f"    F1 Macro: {f1_macro:.4f}")
                print(f"    F1 Weighted: {f1_weighted:.4f}\n")

        self.metricas_por_pais = metricas_por_pais

        if verbose > 0:
            print(f"{'='*70}")
            print("Evaluación por país completada")
            print(f"{'='*70}\n")

        return metricas_por_pais

    def plot_metrics_by_country(self, save_path: str = None, figsize:
Tuple[int, int] = (18, 12)):
        """
        Genera visualizaciones de métricas por país usando matplotlib.

        Args:
            save_path: Ruta para guardar la figura (opcional)
            figsize: Tamaño de la figura
        """
        if not self.metricas_por_pais:
            raise ValueError("Debe ejecutar evaluate_by_country()
primero")

        paises = list(self.metricas_por_pais.keys())
        n_paises = len(paises)

        # Crear DataFrame para facilitar el graficado
        df_metricas = pd.DataFrame([
            {
                'pais': pais,
                'n_samples': self.metricas_por_pais[pais]['n_samples'],
                'accuracy': self.metricas_por_pais[pais]['accuracy'],
                'f1_macro': self.metricas_por_pais[pais]['f1_macro'],
                'f1_micro': self.metricas_por_pais[pais]['f1_micro'],
```

```
                'f1_weighted':
self.metricas_por_pais[pais]['f1_weighted']
            }
            for pais in paises
        ])

        # Crear figura con subplots
        fig = plt.figure(figsize=figsize)
        gs = fig.add_gridspec(3, n_paises, hspace=0.4, wspace=0.4)

        # Fila 1: Gráfico de barras - Accuracy por país
        ax1 = fig.add_subplot(gs[0, :])
        df_sorted = df_metricas.sort_values('accuracy', ascending=False)
        colors = plt.cm.viridis(np.linspace(0.3, 0.9, len(df_sorted)))
        bars = ax1.bar(df_sorted['pais'], df_sorted['accuracy'],
color=colors, alpha=0.8)
        ax1.set_ylabel('Accuracy', fontsize=12, fontweight='bold')
        ax1.set_xlabel('País', fontsize=12, fontweight='bold')
        ax1.set_title('Accuracy por País', fontsize=14,
fontweight='bold', pad=15)
        ax1.axhline(df_metricas['accuracy'].mean(), color='red',
linestyle='--',
                    linewidth=2, label=f'Media:
{df_metricas["accuracy"].mean():.3f}')
        ax1.legend(loc='upper right')
        ax1.grid(axis='y', alpha=0.3, linestyle='--')
        ax1.set_ylim([0, 1.0])

        # Añadir valores sobre las barras
        for bar in bars:
            height = bar.get_height()
            ax1.text(bar.get_x() + bar.get_width()/2., height,
                    f'{height:.3f}',
                    ha='center', va='bottom', fontsize=9,
fontweight='bold')

        # Fila 2: Comparación de métricas F1 por país
        ax2 = fig.add_subplot(gs[1, :])
        x = np.arange(len(paises))
        width = 0.25

        bars1 = ax2.bar(x - width, df_metricas['f1_macro'], width,
                        label='F1 Macro', alpha=0.8, color='#1f77b4')
        bars2 = ax2.bar(x, df_metricas['f1_micro'], width,
                        label='F1 Micro', alpha=0.8, color='#ff7f0e')
        bars3 = ax2.bar(x + width, df_metricas['f1_weighted'], width,
                        label='F1 Weighted', alpha=0.8, color='#2ca02c')

        ax2.set_xlabel('País', fontsize=12, fontweight='bold')
        ax2.set_ylabel('F1 Score', fontsize=12, fontweight='bold')
        ax2.set_title('Comparación de Métricas F1 por País', fontsize=14,
fontweight='bold', pad=15)
        ax2.set_xticks(x)
        ax2.set_xticklabels(df_metricas['pais'], rotation=45, ha='right')
        ax2.legend(loc='upper right')
        ax2.grid(axis='y', alpha=0.3, linestyle='--')
        ax2.set_ylim([0, 1.0])
```

```
        # Fila 3: Matrices de confusión por país
        for idx, pais in enumerate(paises):
            ax = fig.add_subplot(gs[2, idx])

            cm = self.metricas_por_pais[pais]['confusion_matrix']
            accuracy = self.metricas_por_pais[pais]['accuracy']
            n_samples = self.metricas_por_pais[pais]['n_samples']

            # Normalizar matriz de confusión
            cm_normalized = cm.astype('float') / cm.sum(axis=1)[:,
np.newaxis]

            # Crear heatmap
            im = ax.imshow(cm_normalized, cmap='Blues', aspect='auto',
vmin=0, vmax=1)

            # Añadir colorbar
            cbar = plt.colorbar(im, ax=ax, fraction=0.046, pad=0.04)
            cbar.ax.tick_params(labelsize=8)

            # Añadir anotaciones
            for i in range(cm.shape[0]):
                for j in range(cm.shape[1]):
                    text = ax.text(j, i, f'{cm_normalized[i,
j]:.2f}\n({cm[i, j]})',
                                   ha="center", va="center",
                                   color="white" if cm_normalized[i, j] >
0.5 else "black",
                                   fontsize=9)

            ax.set_xticks(np.arange(self.n_classes))
            ax.set_yticks(np.arange(self.n_classes))
            ax.set_xticklabels(['Local', 'Empate', 'Visita'], fontsize=9)
            ax.set_yticklabels(['Local', 'Empate', 'Visita'], fontsize=9)

            ax.set_title(f'{pais}\nAcc: {accuracy:.3f} | N: {n_samples}',
                         fontsize=11, fontweight='bold', pad=10)
            ax.set_ylabel('Real', fontsize=10, fontweight='bold')
            ax.set_xlabel('Predicción', fontsize=10, fontweight='bold')

            # Rotar labels
            plt.setp(ax.get_xticklabels(), rotation=45, ha="right",
rotation_mode="anchor")

        plt.suptitle('Evaluación del Modelo por País',
                     fontsize=16, fontweight='bold', y=0.995)

        if save_path:
            plt.savefig(save_path, dpi=300, bbox_inches='tight')
            print(f"✓ Gráfico guardado en: {save_path}")

        plt.tight_layout()
        plt.show()

    def get_country_metrics_dataframe(self) -> pd.DataFrame:
        """
```

```
        Retorna un DataFrame con las métricas por país.

        Returns:
            DataFrame con métricas por país
        """
        if not self.metricas_por_pais:
            raise ValueError("Debe ejecutar evaluate_by_country()
primero")

        records = []
        for pais, metricas in self.metricas_por_pais.items():
            records.append({
                'pais': pais,
                'n_samples': metricas['n_samples'],
                'accuracy': metricas['accuracy'],
                'f1_macro': metricas['f1_macro'],
                'f1_micro': metricas['f1_micro'],
                'f1_weighted': metricas['f1_weighted']
            })

        return pd.DataFrame(records).sort_values('accuracy',
ascending=False)

    def save_results_country(self, model_directory_path: str) -> None:
        """
        Guarda los resultados de la búsqueda en formato JSON y el modelo
de Keras/TF.

        :param model_directory_path: La ruta donde se guardará el modelo
(SavedModel, es un directorio).
        """

        # 1. Definir la ruta del archivo JSON
        json_filepath = f"{model_directory_path}_metrics.json"

        results_to_save = {
            'best_params': self.best_params,
            'best_score': float(self.best_score),
            'all_results': self.results,
            'n_classes': self.n_classes,
            'param_grid': self.param_grid
        }

        if self.metricas_por_pais:
            # Convertir métricas por país (sin arrays numpy)
            country_metrics = {}
            for pais, metricas in self.metricas_por_pais.items():
                country_metrics[pais] = {
                    'n_samples': int(metricas['n_samples']),
                    'accuracy': float(metricas['accuracy']),
                    'f1_macro': float(metricas['f1_macro']),
                    'f1_micro': float(metricas['f1_micro']),
                    'f1_weighted': float(metricas['f1_weighted']),
                    'confusion_matrix':
metricas['confusion_matrix'].tolist()
                }
            results_to_save['country_metrics'] = country_metrics
```

```
        # 2. Guardar el archivo JSON usando su propia ruta
(json_filepath)
        with open(json_filepath, 'w') as f:
            json.dump(results_to_save, f, indent=2)

        print(f"\nResultados guardados en: {json_filepath}")

        # 3. Guardar el modelo Keras usando su propia ruta
(model_directory_path)
        self.best_model.save(model_directory_path)
        print(f"Modelo exportado como SavedModel en:
{model_directory_path}")

    def get_results_dataframe(self) -> pd.DataFrame:
            """Convierte los resultados en un DataFrame de pandas para
análisis."""
            if not self.results:
                return pd.DataFrame()

            records = []
            for result in self.results:
                record = {}
                for key, value in result['params'].items():
                    record[f'param_{key}'] = value

                for key, value in result['val_metrics'].items():
                    record[f'val_{key}'] = value

                for key, value in result['test_metrics'].items():
                    record[f'test_{key}'] = value

                record['final_train_loss'] = result['final_train_loss']
                record['final_val_loss'] = result['final_val_loss']
                record['epochs_trained'] = result['epochs_trained']

                records.append(record)

            return pd.DataFrame(records)
```