

# **UNIVERSIDAD NACIONAL DE SAN LUIS**

Facultad de Cs. Físico Matemáticas y Naturales –Facultad de Cs. Económicas, Jurídicas y Sociales



## **Curso “Modelos y simulación” de la Licenciatura en Análisis y Gestión de Datos**

*“Trabajo final Integrador Análisis de datos de Fútbol”*

Alumnos : Matías Leonel Pérez & Angel José Villareal Barroso

Docentes : Cristian Tissera & Enrique Vallejo

## **2. Índice**

<b>1. UNIVERSIDAD NACIONAL DE SAN LUIS</b>	<b>1</b>
2. Índice	2
3. Resumen	3
4. Introducción	3
4.1 Contexto general del problema	3
4.2 Justificación del trabajo	3
4.3 Objetivos generales y específicos	4
4.4 Estructura del informe	4
5. Marco teórico	5
5.1. Modelos matemáticos y computacionales	5
5.2. Técnicas de simulación utilizadas	6
Justificación de la técnica elegida	6
5.3. Fundamentos estadísticos y de análisis de datos	7
6. Caso de Estudio	8
6.1. Descripción del trabajo de simulación de temporada	8
6.2. Supuestos y restricciones para el modelo	8
6.3. Variables involucradas	9
6.4. Representación del sistema	10
7. Metodología	12
7.1. Descripción general del modelo aplicado	12
7.2. Tipo de modelo	12
7.3. Ecuaciones y algoritmos utilizados	12
7.4. Herramientas y lenguajes utilizados	13
7.5. Procedimiento paso a paso	13
7.6. Justificación de decisiones metodológicas	18
8. Implementación y Simulación	19
8.1. Descripción de la implementación	19
8.2. Parámetros de entrada	20
8.3. Escenarios simulados	21
8.4. Configuración de la simulación	21
8.5. Número de repeticiones o iteraciones	22
9. Resultados	22
9.1. Análisis descriptivo de los datos históricos	22
9.2. Evaluación básica del ajuste del modelo	25
9.3. Comportamiento de los puntos en las temporadas simuladas	26
9.4. Comparación con la temporada real	28
9.5. Visualizaciones y tablas resumen	30
10. Discusión	31
10.1. Interpretación de los resultados obtenidos	31
10.2. Fortalezas del enfoque utilizado	31
10.3. Limitaciones del modelo	32
10.4. Posibles extensiones y mejoras	32
11. Conclusiones	33
11.1. Conclusiones generales del trabajo	33
11.2. Aprendizajes desde la perspectiva de Modelos y Simulación	33
11.3. Reflexión final	34

12. Referencias bibliográficas	34
13. Anexo	35

## 3. Resumen

En este trabajo se aplica la metodología de **modelos y simulación** al análisis de resultados de la Premier League inglesa en el período 2020–2023.

A partir de una base de datos de partidos reales se modela la cantidad de goles anotados por los equipos considerando la condición de local y visitante, utilizando **distribuciones de Poisson** ajustadas a los datos históricos. Sobre esta base se construye un modelo estocástico que permite simular múltiples temporadas completas mediante el método de **Monte Carlo**. Se analiza la distribución de puntos obtenidos por los equipos, la variabilidad posible en la tabla de posiciones y la probabilidad de que determinados equipos alcancen ciertos umbrales de rendimiento.

Los resultados muestran que, aun manteniendo constantes las **tasas medias de gol** estimadas a partir de los datos reales, existe una alta variabilidad en los desenlaces de la temporada, lo que resalta la importancia del componente aleatorio en el fútbol.

El estudio ilustra cómo las técnicas de simulación pueden utilizarse para comprender mejor la incertidumbre inherente a las competiciones deportivas.

## 4. Introducción

### 4.1 Contexto general del problema

El fútbol profesional es un sistema complejo en el que intervienen múltiples factores: calidad de los planteles, decisiones tácticas, estado físico de los jugadores, arbitraje, localía y, en general, una fuerte componente de azar.

En este contexto, la Premier League inglesa se destaca por su competitividad y por la abundancia de datos estadísticos disponibles sobre sus partidos. Estas características la convierten en un escenario atractivo para aplicar herramientas de modelado y simulación, con el objetivo de estudiar cómo la variabilidad aleatoria puede afectar el desarrollo de una temporada.

### 4.2 Justificación del trabajo

Desde la perspectiva de la asignatura Modelos y Simulación, trabajar con datos de fútbol permite traducir conceptos teóricos , como variables aleatorias, distribuciones de probabilidad, modelos estocásticos y simulación de Monte Carlo, a un caso de estudio cercano y fácil de interpretar.

En particular, modelar la cantidad de goles y los resultados de los partidos brinda una forma concreta de analizar cómo, a partir de supuestos probabilísticos relativamente simples, es posible recrear escenarios alternativos de una misma competición y cuantificar la incertidumbre asociada a los desenlaces observados. Esto no solo tiene interés académico, sino también aplicaciones potenciales en análisis deportivo, predicción y toma de decisiones.

#### **4.3 Objetivos generales y específicos**

##### Objetivo general

Construir y analizar un modelo de simulación estocástico basado en datos históricos de la Premier League, que permita generar temporadas ficticias y estudiar la variabilidad de los resultados de la competición.

##### Objetivos específicos

- Modelar la cantidad de goles de los equipos considerando la condición de local y visitante a partir de los datos históricos.
- Definir un mecanismo probabilístico para simular el resultado de cada partido del fixture.
- Implementar una simulación de Monte Carlo con un número elevado de réplicas de la temporada (por ejemplo, 1000 temporadas simuladas).
- Analizar la distribución de puntos y posiciones finales de los equipos en las simulaciones y compararlas con la temporada real.
- Evaluar las limitaciones del modelo y proponer posibles mejoras o extensiones.

#### **4.4 Estructura del informe**

El informe se organiza de la siguiente manera:

Como **Marco Teórico**, se presentan los conceptos principales de modelos estocásticos, distribuciones de probabilidad para conteos de eventos y simulación de Monte Carlo, junto con los fundamentos estadísticos utilizados.

Luego, se detalla el **caso de estudio** de la Premier League, las fuentes de datos y las variables consideradas.

En la sección de **Metodología** se describe el modelo planteado, los supuestos adoptados y el procedimiento seguido para la simulación de temporadas.

A continuación, se procede a la **Implementación y Simulación**, de las herramientas empleadas (principalmente Python) y la configuración de los experimentos computacionales.

Tras lograr haber alcanzado los resultados, presentamos los principales **hallazgos** cuantitativos y visualizaciones obtenidas.

Finalmente, en **Discusión y Conclusiones** se interpretan los resultados, se analizan las limitaciones del modelo y se proponen posibles líneas de trabajo futuras.

## 5. Marco teórico

### 5.1. Modelos matemáticos y computacionales

En el contexto de Modelos y Simulación, un **modelo** es una representación simplificada de un sistema real que permite estudiar su comportamiento sin tener que experimentar directamente sobre la realidad. En este trabajo, el **sistema real** es una **temporada de la Premier League**: un conjunto de equipos que se enfrentan según un fixture preestablecido, generando resultados en forma de goles y puntos.

Los modelos pueden ser **deterministas** o **estocásticos**.

- En un modelo determinista, dadas las mismas condiciones iniciales, el resultado siempre es el mismo.
- En un modelo estocástico, en cambio, se incorpora explícitamente la **aleatoriedad** mediante variables aleatorias, de modo que el mismo experimento puede producir resultados diferentes en cada ejecución.

Dado que el rendimiento deportivo tiene un componente fuertemente azaroso (rebotes, errores, decisiones arbitrales, etc.), en este trabajo se utiliza un **modelo estocástico**. El objetivo no es predecir exactamente el resultado de un partido, sino describir la **distribución de posibles resultados** y la variabilidad de una temporada completa bajo ciertos supuestos probabilísticos.

El modelo computacional implementado en Python puede verse como una “caja negra” que recibe:

- parámetros que resumen las **tasas medias de goles** de los equipos según la condición de juego (local o visitante),

- un fixture de partidos,
- y números pseudoaleatorios generados por computadora,

y devuelve como salida un conjunto de temporadas simuladas con sus tablas de posiciones y estadísticas asociadas.

## 5.2. Técnicas de simulación utilizadas

La técnica central empleada es la **Simulación de Monte Carlo**. Este enfoque consiste en:

1. Definir un modelo probabilístico del sistema (en este caso, cómo se generan los goles de cada partido).
2. Repetir muchas veces el experimento de interés (simular la temporada completa) utilizando números aleatorios para muestrear de las distribuciones definidas.
3. Observar y resumir las salidas del modelo (por ejemplo, la distribución de puntos de cada equipo o la probabilidad de que supere cierto umbral).

Cada ejecución completa de la temporada se denomina **réplica** o **corrida de simulación**. A medida que aumenta el número de réplicas, las estimaciones de interés (por ejemplo, la probabilidad de que un equipo salga campeón) se estabilizan gracias a la **Ley de los Grandes Números**.

Para garantizar reproducibilidad se utiliza un **generador de números pseudoaleatorios** con una semilla fija, lo que permite obtener los mismos resultados si se vuelve a ejecutar el código bajo las mismas condiciones.

### Justificación de la técnica elegida

Se opta por Simulación de Monte Carlo porque:

- El sistema tiene una estructura relativamente sencilla (partidos independientes con un fixture dado), pero los resultados posibles de una temporada son combinatorialmente enormes, lo que dificulta un tratamiento **analítico exacto**.
- La simulación permite incorporar de forma flexible distintos supuestos (por ejemplo, diferenciar la fuerza de ataque de local y visitante) sin necesidad de resolver expresiones matemáticas complejas.
- La técnica es directamente compatible con los contenidos de la materia y con el uso de Python como herramienta de apoyo.

Se descartaron otras aproximaciones, como modelos puramente **deterministas** (que no representarían la variabilidad real de los resultados) o modelos de **agentes** detallados (que requerirían información muy específica a nivel de jugadores y decisiones tácticas, no disponible en este trabajo). También se dejó de lado la construcción de modelos de ecuaciones diferenciales o de dinámica de sistemas, que son más adecuados para fenómenos continuos en el tiempo y no para conteos de goles en eventos discretos.

### 5.3. Fundamentos estadísticos y de análisis de datos

La base estadística del modelo se apoya en el concepto de **variable aleatoria discreta**, que en este caso representa la cantidad de goles anotados por un equipo en un partido. A partir de los datos históricos de la Premier League se calculan, para cada condición (local y visitante) y, eventualmente, para cada equipo, medidas como:

- media de goles anotados,
- media de goles recibidos,
- frecuencias relativas de los distintos resultados posibles.

Una elección habitual para modelar conteos de eventos en un intervalo fijo de tiempo es la **distribución de Poisson**, cuyo parámetro  $\lambda$  representa la cantidad media de eventos (goles) por partido. En este trabajo, el valor de  $\lambda$  se estima a partir de los datos observados, utilizando como estimador la **media muestral de goles**. Esto supone que:

- los goles se producen de manera relativamente “independiente” a lo largo del partido,
- la probabilidad de que ocurran dos o más goles simultáneamente es despreciable,
- la tasa media de goles se mantiene aproximadamente constante para una determinada combinación equipo–condición (por ejemplo, Manchester City jugando de local).

Además, se realizan análisis descriptivos complementarios (tablas de frecuencia, histogramas de goles, promedios por temporada) para evaluar qué tan razonable es asumir una forma de distribución determinada. Cuando la Poisson logra ajustar de manera aceptable, se utiliza como base del modelo; si no, en trabajos futuros podrían explorarse variantes (por ejemplo, distribuciones empíricas obtenidas directamente de los datos).

Desde el punto de vista de la simulación, cada valor de goles generado para un partido es una **muestra** de la distribución elegida. Repetir la simulación de toda la temporada muchas veces permite obtener distribuciones empíricas para variables de interés más agregadas, como:

- **puntos finales de cada equipo,**
- diferencia de gol,
- **posición en la tabla.**

La comparación entre estas distribuciones simuladas y los valores realmente observados aporta la primera forma de **validación del modelo** a nivel descriptivo.

## 6. Caso de Estudio

### 6.1. Descripción del trabajo de simulación de temporada

El caso de estudio se centra en la **Premier League inglesa**, una liga de fútbol profesional en la que participan 20 equipos que se enfrentan todos contra todos, a dos ruedas, a lo largo de una temporada. Cada partido otorga 3 puntos por victoria, 1 punto por empate y 0 puntos por derrota, y al finalizar el fixture se ordena a los equipos según la cantidad total de puntos obtenidos.

El problema que se aborda en este trabajo es el siguiente:

A partir de los resultados históricos de la Premier League en el período 2020–2023, ¿es posible construir un modelo probabilístico que permita **simular temporadas alternativas** y estudiar cuánta variabilidad puede haber en la tabla de posiciones, manteniendo constantes las tasas medias de goles observadas para cada equipo y condición?

En otras palabras, se busca responder preguntas del tipo:

- ¿Qué tan diferente podría haber sido la temporada si se repitiera muchas veces bajo las mismas condiciones estadísticas?
- ¿Cuál es la distribución de puntos posible para cada equipo?
- ¿Qué probabilidad tendría un equipo de alcanzar o superar cierto umbral de puntos?

### 6.2. Supuestos y restricciones para el modelo

Para poder modelar y simular el sistema, se adoptan una serie de **supuestos** que simplifican la realidad que alojan estos eventos deportivos:

#### 1. Independencia de los partidos

Se asume que los partidos son independientes entre sí: el resultado de un encuentro no afecta directamente los parámetros del siguiente (no se modelan

rachas, lesiones, cansancio, etc.).

## 2. Parámetros constantes en la temporada

Se considera que las tasas medias de goles de cada equipo, así como el efecto de la localía, se mantienen constantes a lo largo de la temporada. Los cambios de entrenador, fichajes o lesiones no se incorporan explícitamente.

## 3. Modelo de goles por equipo y condición

La cantidad de goles que anota cada equipo en un partido se modela mediante una distribución de Poisson, cuyo parámetro depende del equipo y de si juega de local o visitante. Estos parámetros  $\lambda_{i,local}$  y  $\lambda_{i,visitante}$  se estiman a partir de los datos históricos como la media de goles a favor del equipo en cada condición.

## 4. Reglamento fijo

Se toma el reglamento estándar de la liga: 3 puntos por victoria, 1 por empate, diferencia de gol como criterio secundario, sin desempates más complejos.

## 5. Mismo fixture que en la realidad

El orden y las combinaciones de partidos (quién juega contra quién y en qué condición) se mantienen según el fixture real de la temporada.

Además, el trabajo presenta las siguientes **restricciones**:

- Solo se utilizan las variables disponibles en la base de datos (goles, local/visitante, fecha, equipo, rival, etc.); no se incorporan datos avanzados como xG, posesión o estadísticas individuales de jugadores.
- El análisis se limita a las temporadas seleccionadas y no se modelan interacciones entre temporadas (por ejemplo, cambios de categoría o promociones/descensos).

## 6.3. Variables involucradas

Las principales variables utilizadas en el caso de estudio son:

- **team**: nombre del equipo considerado en ese registro.
- **opponent**: nombre del rival al que enfrenta *team* en ese partido.
- **gf (goals for)**: cantidad de goles anotados por *team* en el partido (goles a favor).
- **ga (goals against)**: cantidad de goles recibidos por *team* en el partido (goles en contra).
- **venue / home\_away** : indica si *team* jugó como **local** o como **visitante** en ese encuentro.

- **season**: temporada a la que pertenece el partido (por ejemplo, 2020–2021).
- **date**: fecha en la que se jugó el encuentro.

En la etapa de análisis descriptivo, estas variables permiten:

- reconstruir los resultados de cada partido,
- calcular los puntos obtenidos por cada equipo en cada encuentro,
- y estimar parámetros como la **media de goles de local** y la **media de goles de visitante** para cada equipo.

A partir de estas medias, agregando por equipo y condición (local/visitante), se obtienen los parámetros  $\lambda_{i,local}$  y  $\lambda_{i,visitante}$  que alimentan el modelo de simulación de Poisson.

## 6.4. Representación del sistema

Desde el punto de vista de la simulación, el sistema puede representarse de la siguiente manera:

### 1. Nivel de equipos

Cada equipo  $i$  está caracterizado por parámetros que resumen su comportamiento goleador en cada condición de juego. En el modelo principal utilizado, cada equipo tiene:

$\lambda_{i,local}$ : media de goles anotados por el equipo  $i$  cuando juega de local.

$\lambda_{i,visitante}$ : media de goles anotados por el equipo  $i$  cuando juega de visitante.

Estos parámetros se estiman a partir de los datos históricos, calculando la media de goles a favor en los partidos en los que el equipo actúa como local o como visitante.

### 2. Nivel de partidos

El fixture define una lista ordenada de partidos (equipo\_local,equipo\_visitante) Para cada partido entre un equipo local  $h$  y un equipo visitante  $a$ , el modelo genera dos variables aleatorias:

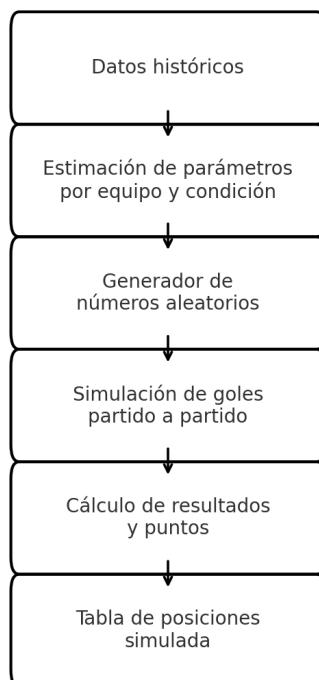
- Goles del local, muestreados de la distribución asociada al equipo  $h$  jugando en casa, con intensidad  $\lambda_{h,\text{local}}$
- Goles del visitante, muestreados de la distribución asociada al equipo a jugando fuera de casa, con intensidad  $\lambda_{a,\text{visitante}}$

En la implementación concreta, estas variables se modelan como distribuciones de Poisson con los parámetros estimados para cada equipo y condición.

### 3. Nivel de temporada

A partir de los resultados de todos los partidos simulados se calculan puntos, diferencia de gol y posición final para cada equipo, obteniendo así una temporada “ficticia” pero consistente con los parámetros estimados.

El flujo del sistema sería:



**Figura 1.** Flujo del modelo de simulación: desde los datos históricos hasta la tabla de posiciones simulada.

Este esquema resume la conexión entre el caso de estudio real (Premier League) y el modelo de simulación implementado en Python.

## 7. Metodología

### 7.1. Descripción general del modelo aplicado

El modelo propuesto es un **modelo estocástico de goles por partido**, en el cual la cantidad de goles que anota cada equipo se considera una variable aleatoria discreta.

A partir de los datos históricos de la Premier League se reconstruye primero la información a nivel partido (fecha, equipo local, equipo visitante, goles de cada uno). Sobre esa base se estiman parámetros que resumen el comportamiento de los equipos cuando juegan de **local** y de **visitante**, y estos parámetros se utilizan para generar resultados simulados para cada partido del fixture.

El modelo se integra luego en una **simulación de Monte Carlo** de temporadas completas: para cada réplica se simulan todos los partidos, se calculan los puntos de cada equipo y se obtiene una tabla de posiciones simulada. La repetición de este procedimiento muchas veces permite estudiar la distribución de posibles desenlaces de la competición.

### 7.2. Tipo de modelo

- **Tipo de modelo:** estocástico, basado en variables aleatorias de conteo.
- **Nivel de detalle:** se trabaja a nivel de **partido completo**, sin modelar eventos internos (remates, posesiones, etc.).
- **Enfoque temporal:** estático por partido; la temporada se obtiene como la suma de todos los partidos simulados según el fixture real.

### 7.3. Ecuaciones y algoritmos utilizados

Para modelar la cantidad de goles se utiliza, en primera instancia, una **distribución de Poisson**.

Sea:

- $G_{i,\text{local}}$ : goles que anota el equipo  $i$  cuando juega de local.
- $G_{i,\text{visitante}}$ : goles que anota el equipo  $i$  cuando juega de visitante.

Se supone que:

$$G_{i,\text{local}} \sim \text{Poisson}(\lambda_{i,\text{local}}), \quad G_{i,\text{visitante}} \sim \text{Poisson}(\lambda_{i,\text{visitante}})$$

donde los parámetros  $\lambda$  se estiman a partir de los datos históricos como la **media muestral** de goles en cada condición (local/visitante), ya sea:

- a nivel de cada equipo, o
- a nivel global de la liga, diferenciando solo local y visitante

Para simular un partido entre el equipo A (local) y B (visitante) se siguen los pasos:

1. Generar un valor  $g_A$  muestrado de Poisson ( $\lambda_{A,\text{local}}$ )
2. Generar un valor  $g_B$  muestrado de Poisson ( $\lambda_{B,\text{visitante}}$ ).
3. Determinar el resultado del partido:
  - si  $g_A > g_B$  : victoria local (3 puntos para A, 0 para B);
  - si  $g_A = g_B$ : empate (1 punto para cada uno);
  - si  $g_A < g_B$ : victoria visitante (0 puntos para A, 3 para B).

Este algoritmo se repite para todos los partidos del fixture en cada réplica de la simulación.

## 7.4. Herramientas y lenguajes utilizados

La implementación se realiza en **Python**, utilizando principalmente:

- **pandas** para el manejo y limpieza de la base de datos de partidos,
- **numpy** para la generación de números aleatorios y operaciones numéricas,
- eventualmente **matplotlib** o **seaborn** para la generación de gráficos de resultados.

Se emplea un **generador de números pseudoaleatorios** con semilla fija (`set_seed(seed)` basado en `np.random.seed`) para asegurar la reproducibilidad de los experimentos.

## 7.5. Procedimiento paso a paso

El procedimiento metodológico puede resumirse en las siguientes etapas:

1. **Preparación de los datos**
  - Importación del archivo con los partidos de la Premier League.

- Reconstrucción de la información a nivel partido mediante una función de limpieza (reconstruct\_matches\_level), obteniendo columnas home, away, gh, ga, season, date, round.
- Estandarización de nombres de equipos y verificación de consistencia de los resultados (goles de local/visitante).
- Cálculo de variables derivadas: resultado del partido, puntos obtenidos por cada equipo, etc.

```
from sim_utils import reconstruct_matches_level

RAW = Path(r"FILES/df_matches_limpio.xlsx")
df_raw = pd.read_excel(RAW)
clean = reconstruct_matches_level(df_raw)
CLEAN = Path(r"FILES/matches_clean.csv")
```

## 2. Partición en muestra de estimación y validación

**Separados los datos en:**

- Una muestra de **estimación** (temporadas anteriores), utilizada para calibrar los parámetros del modelo.
- Una muestra de **validación** (por ejemplo, la última temporada), utilizada como fixture objetivo para comparar los resultados simulados con los resultados reales.

```
from sim_utils import split_est_val

CLEAN = Path(r"FILES/matches_clean.csv")
df_clean = pd.read_csv(CLEAN, parse_dates=["date"])
df_est, df_val, val_season = split_est_val(df_clean)
EST = Path(r"FILES/matches_est.csv")
VAL = Path(r"FILES/matches_val.csv")
df_est.to_csv(EST, index=False)
df_val.to_csv(VAL, index=False)
```

## 3. Estimación de parámetros del modelo

- Cálculo, sobre la muestra de estimación, de la media de goles para cada equipo cuando juega de local y cuando juega de visitante.

- Construcción de una tabla de parámetros  $\lambda_i$ , local y  $\lambda_i$ , visitante, que se utilizarán en la simulación.
- En caso de equipos con información insuficiente, se recurre a la media global de la liga en la condición correspondiente.

```
from sim_utils import empirical_global, empirical_by_team, poisson_lambdas

EST = Path(r"FILES/matches_est.csv")
df_est = pd.read_csv(EST, parse_dates=["date"])

emp_global = empirical_global(df_est)
emp_global_path = Path(r"FILES/empirical_cdf_global.csv")
emp_global.to_csv(emp_global_path, index=False)

emp_team = empirical_by_team(df_est, min_n_home=35, min_n_away=35)
if len(emp_team):
    emp_team_path = Path(r"FILES/empirical_cdf_by_team.csv")
    emp_team.to_csv(emp_team_path, index=False)
    print("Guardado empírico por equipo:", emp_team_path)
else:
    print("No hay suficiente N para tablas empíricas por equipo. Se usará la global.")

lam = poisson_lambdas(df_est)
lam_path = Path(r"FILES/poisson_lambdas_by_team.csv")
lam.to_csv(lam_path, index=False)
```

#### 4. Definición del experimento de simulación

- Fijación del número de réplicas o temporadas simuladas N (en el orden de los miles, por ejemplo).
- Definición del fixture a simular, utilizando los partidos de la temporada de validación (lista de pares `home-away`).

#### 5. Elección de la semilla del generador aleatorio

Para garantizar la reproducibilidad de los resultados, se fijó una semilla específica en el generador de números pseudoaleatorios utilizado en la simulación (`seed = 123`). El valor numérico de la semilla es arbitrario: su función no es “mejorar” el modelo, sino permitir que, bajo las mismas condiciones, se obtengan exactamente los mismos resultados al volver a ejecutar el código.

Como verificación adicional, se repitieron algunas corridas de simulación utilizando semillas alternativas (por ejemplo, 42 y 2025) y se compararon las medias de puntos simulados y las probabilidades de eventos de interés. Las diferencias observadas fueron mínimas y atribuibles únicamente a la variabilidad Monte Carlo, lo que indica que las conclusiones del trabajo no dependen de la elección puntual de la semilla, sino del número de réplicas y del modelo especificado.

```

# Parámetros de simulación
R = 1000      # cantidad de réplicas
SEED = 123    # semilla aleatoria para reproducibilidad

```

## 6. Simulación de temporadas

Para cada réplica :

- Recorrer secuencialmente todos los partidos del fixture.
- Para cada partido, generar goles de local y visitante usando las distribuciones de Poisson con los parámetros correspondientes.
- Actualizar los puntos de cada equipo según el resultado.
- Al finalizar todos los partidos, registrar para cada equipo: puntos totales, diferencia de gol y posición en la tabla.

```

from sim_utils import simulate_season_empirical, summarize_points
emp_global = pd.read_csv(r"FILES/empirical_cdf_global.csv")
emp_team_path = r"FILES/empirical_cdf_by_team.csv"
emp_team = pd.read_csv(emp_team_path)

emp_points = simulate_season_empirical(
    df_fixture=df_val,
    emp_global=emp_global,
    emp_by_team=emp_team,
    R=R,
    seed=SEED
)
emp_summary = summarize_points(emp_points)

emp_points.to_csv(r"FILES/points_empirical_long.csv", index=False)
emp_summary.to_csv(r"FILES/points_empirical_summary.csv", index=False)

```

```

from sim_utils import simulate_season_poisson, summarize_points
lambdas = pd.read_csv(r"FILES/poisson_lambdas_by_team.csv")

poi_points = simulate_season_poisson(
    df_fixture=df_val,
    Lambdas=lambdas,
    R=R,
    seed=SEED
)
poi_summary = summarize_points(poi_points)

poi_points.to_csv(r"FILES/points_poisson_long.csv", index=False)
poi_summary.to_csv(r"FILES/points_poisson_summary.csv", index=False)

```

## 7. Almacenamiento y organización de las salidas

- Guardar, para cada réplica, una tabla con los resultados finales de todos los equipos (`replica`, `team`, `points`).
- Unir estas tablas para obtener una base de datos de resultados simulados que permita análisis posteriores (cálculo de medias, probabilidades, etc.).

## 8. Análisis de resultados

- Calcular, para cada equipo, la distribución de puntos obtenidos en las N temporadas simuladas (media, desvío estándar, percentiles).
- Estimar probabilidades de eventos de interés (por ejemplo, probabilidad de superar cierto umbral de puntos o de terminar en las primeras posiciones).
- Comparar las distribuciones simuladas con los valores observados en la temporada real de validación.

team	points_real	mean_poi	mean_emp	mean_xg	err_mean_poi	err_mean_emp	err_mean_xg
Manchester City	91	78.336	70.065	82.131	12.664	20.935	8.869
Arsenal	89	62.033	60.282	60.376	26.967	28.718	28.624
Liverpool	82	71.513	68.798	78.983	10.487	13.202	3.017
Aston Villa	68	52.020	49.382	52.377	15.980	18.618	15.623
Tottenham	66	63.543	60.346	63.074	2.457	5.654	2.926
Chelsea	63	59.206	54.830	58.072	3.794	8.170	4.928
Manchester United	60	61.380	60.591	63.579	-1.380	-0.591	-3.579
Newcastle United	60	50.851	51.783	52.285	9.149	8.217	7.715
West Ham	52	54.069	51.499	53.925	-2.069	0.501	-1.925
Crystal Palace	49	44.338	48.477	36.756	4.662	0.523	12.244

team	mean_points	sd_points	p10	p50	p90
Manchester City	78.336	7.348000	69.0	78.0	87.0
Liverpool	71.513	7.949656	61.0	72.0	82.0
Tottenham	63.543	8.182488	53.0	64.0	74.0
Arsenal	62.033	7.831810	52.0	62.0	72.0
Manchester United	61.380	7.917846	51.0	61.0	72.0
Chelsea	59.206	8.018095	48.0	59.0	70.0
West Ham	54.069	7.633775	44.0	54.0	64.0
Brentford	53.811	7.998453	43.0	54.0	64.0
Aston Villa	52.020	7.766848	42.0	52.0	62.0
Luton Town	51.228	7.896478	41.0	51.0	61.0
Newcastle United	50.851	7.875203	40.0	51.0	61.0
Brighton	50.077	7.942427	40.0	50.0	60.0
Everton	45.879	7.775656	36.0	46.0	56.0
Crystal Palace	44.338	8.020853	34.0	45.0	54.0
Fulham	44.182	7.264134	35.0	44.0	54.0
Wolverhampton	43.225	7.573489	34.0	43.0	53.0
Bournemouth	42.614	7.440589	33.0	42.0	53.0
Nottingham Forest	41.598	7.280343	32.0	42.0	51.0
Burnley	40.925	7.598362	32.0	41.0	51.0
Sheffield United	35.044	6.841116	26.0	35.0	44.0

## 7.6. Justificación de decisiones metodológicas

Las principales decisiones metodológicas adoptadas se justifican de la siguiente manera:

- **Uso de un modelo estocástico por goles**

El fútbol es un deporte con marcadas fluctuaciones aleatorias; por ello, un modelo determinista (en el que cada equipo tenga siempre el mismo resultado) sería poco realista. El enfoque basado en variables aleatorias de goles permite capturar de manera sencilla la incertidumbre inherente al juego.

- **Elección de la distribución de Poisson**

La Poisson es una elección natural para modelar conteos de eventos raros en un intervalo fijo, como los goles en un partido. Además, es una distribución simple, con un único parámetro que se puede estimar fácilmente a partir de la media observada, lo que la hace adecuada para un trabajo de carácter introductorio en modelos de simulación.

- **Diferenciación entre local y visitante**

La literatura y la experiencia futbolística indican la existencia de una “ventaja de la localía”. Distinguir parámetros para partidos de local y de visitante permite incorporar este efecto de forma explícita, sin complejizar en exceso el modelo.

- **Uso del fixture real de la temporada**

Mantener el mismo calendario de partidos que se jugó en la realidad facilita la comparación entre los resultados simulados y los observados, y evita introducir un nivel adicional de complejidad en el modelado de la estructura del torneo.

- **Número elevado de réplicas (N grande)**

La simulación de un número grande de temporadas (miles de réplicas) permite obtener estimaciones más estables de las probabilidades de interés y observar con mayor claridad la forma de las distribuciones de puntos y posiciones.

Estas decisiones buscan encontrar un equilibrio entre **realismo** y **simplicidad**, de manera que el modelo sea interpretable, esté alineado con los contenidos de la materia y, al mismo tiempo, resulte suficientemente rico como para mostrar la variabilidad de una competencia real como la Premier League.

## 8. Implementación y Simulación

### 8.1. Descripción de la implementación

La implementación del modelo se realizó en **Python**, organizando el código en tres bloques principales:

1. **Carga y preparación de datos**

- Lectura del archivo `.csv` con los partidos de la Premier League.

- Uso de una función de limpieza para reconstruir la información a **nivel partido**, obteniendo un DataFrame con las columnas `date`, `home`, `away`, `gh`, `ga`, `result`, `season` y `round`.
- Selección de las temporadas analizadas y verificación de consistencia de los resultados (goles del local y del visitante).

## 2. Estimación de parámetros

- Separación de los datos en una muestra de **estimación** (temporadas iniciales) y una muestra de **validación** (última temporada), que se utilizará como fixture sobre el que se simula.
- Agrupación de la información de la muestra de estimación por equipo y condición (local / visitante).
- Cálculo de la media de goles anotados por cada equipo cuando juega de local y cuando juega de visitante. Estas medias se utilizan como estimadores de los parámetros  $\lambda_{i,\text{local}}$  y  $\lambda_{i,\text{visitante}}$  de la distribución de Poisson.
- Almacenamiento de estos parámetros en una tabla que relaciona cada equipo con sus intensidades de goles en ambas condiciones.

## 3. Simulación de temporadas

- Definición de una función que, dado un fixture (muestra de validación) y los parámetros  $\lambda$ , simula una temporada completa: para cada partido genera los goles del local y del visitante, asigna los puntos y actualiza la tabla de cada equipo.
- Repetición de esta función un número elevado de veces, guardando para cada réplica la información final de puntos por equipo en un formato largo (`replica`, `team`, `points`).

La lógica principal se resume en un bucle externo sobre el número de temporadas simuladas y un bucle interno sobre la lista de partidos del fixture.

## 8.2. Parámetros de entrada

Los principales parámetros de entrada del sistema de simulación son:

- **Datos históricos de partidos:** archivo con la información de la Premier League (equipos, goles, condición de local/visitante, temporada).

- **Parámetros de goles:** valores  $\lambda_i$ , local y  $\lambda_i$ , visitante estimados para cada equipo  $i$  a partir de la muestra de estimación.
- **Fixture de la temporada de validación:** lista de partidos a simular (equipo local, equipo visitante) obtenida de la última temporada disponible.
- **Número de réplicas  $N$ :** cantidad de temporadas simuladas (en este trabajo se utilizaron valores del orden de los miles, por ejemplo  $N = 1.000$  temporadas).
- **Semilla del generador aleatorio:** valor entero fijado para garantizar reproducibilidad (por ejemplo, `seed = 123`).

### 8.3. Escenarios simulados

En este trabajo se consideró un **escenario base**:

- Los parámetros de goles se estiman a partir de los datos históricos de las temporadas de estimación y se mantienen constantes durante todas las simulaciones.
- El fixture utilizado en la simulación coincide exactamente con el de la temporada de validación.
- No se modelan cambios de plantel, lesiones ni variaciones de rendimiento a lo largo del tiempo.

Sobre este escenario base se generan múltiples temporadas ficticias. En términos de la guía del trabajo práctico, cada temporada simulada constituye un **escenario posible** compatible con los datos históricos y con los supuestos del modelo.

### 8.4. Configuración de la simulación

La simulación se configuró de la siguiente manera:

#### 1. Inicialización

- Se fija la semilla del generador de números pseudoaleatorios.
- Se preparan estructuras vacías para guardar los resultados de cada réplica (listas o DataFrames).

#### 2. Bucle de réplicas

Para cada réplica:

- Se inicializa una tabla de puntos en cero para todos los equipos presentes en el fixture.
- Se recorre el fixture partido por partido.
- Para cada partido se obtienen los parámetros  $\lambda_A$ ,local y  $\lambda_B$ ,visitante correspondientes a los equipos local y visitante, y se generan los goles simulados utilizando distribuciones de Poisson con esas intensidades.
- Se asignan los puntos en función del resultado y se actualiza la tabla de cada equipo.
- Al terminar todos los partidos, se guarda la tabla final de la réplica en formato largo (`replica, team, points`).

### 3. Salida de la simulación

- Se unifican las tablas de todas las réplicas en una sola base de datos, añadiendo un identificador de réplica.
- Esta base consolidada es la entrada para la etapa de análisis de resultados (cálculo de medias, probabilidades, percentiles, etc.), que se realiza agrupando por equipo.

## 8.5. Número de repeticiones o iteraciones

El número de temporadas simuladas se fijó en un valor grande (por ejemplo,  $N=1.000$ ), buscando un equilibrio entre:

- **Estabilidad estadística** de las estimaciones (la variabilidad entre corridas disminuye al aumentar  $N$ )
- **Costo computacional** (tiempo de ejecución y uso de memoria).

Durante la etapa exploratoria se probaron valores menores de  $N$  (por ejemplo, 500 o 1.000 réplicas) para verificar que las distribuciones de puntos y las probabilidades estimadas se estabilizaran. Una vez observado este comportamiento, se adoptó el valor final de  $N$  para el análisis presentado en el informe.

## 9. Resultados

### 9.1. Análisis descriptivo de los datos históricos

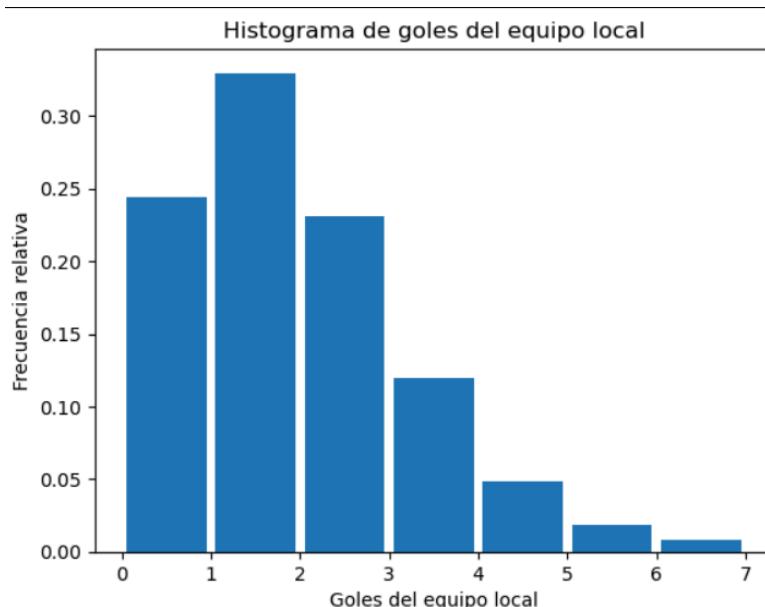
A partir de la base de partidos reales de la Premier League en las temporadas utilizadas como muestra de estimación, se realizó primero un análisis descriptivo con el objetivo de entender el comportamiento de los goles y de los puntos:

- **La media de goles del equipo local resultó mayor que la del equipo visitante, lo que confirma la presencia de ventaja de localía en el período analizado.**

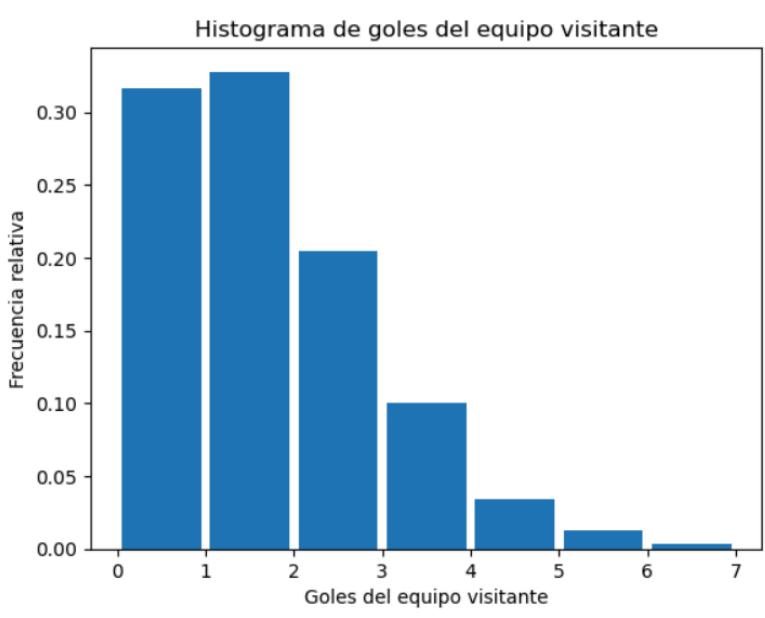
Condición	Media de goles
Local	1.503947
Visitante	1.267763

**Tabla 1.** Media de goles local y visitante

- Los histogramas de goles de local y de visitante muestran que la mayoría de los partidos termina con valores bajos (0, 1, 2 o 3 goles por equipo), con frecuencias decrecientes a medida que aumenta la cantidad de goles.

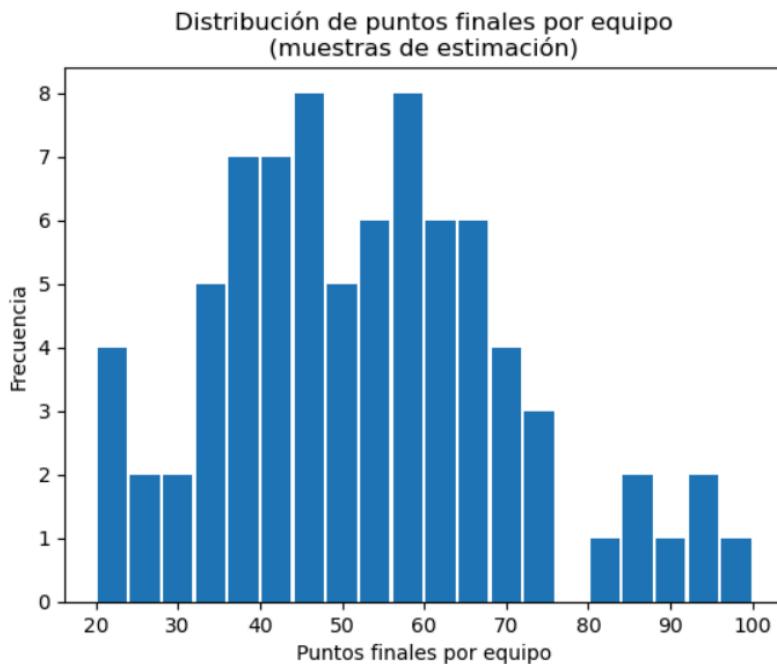


**Figura 1.** Histograma de goles del equipo local en la muestra de estimación de la Premier League



**Figura 2.** Histograma de goles del equipo visitante en la muestra de estimación de la Premier League

- La distribución de puntos finales por equipo presenta una forma relativamente extendida: algunos equipos se concentran en la zona alta de la tabla con muchos puntos, mientras que otros se agrupan en la parte baja con registros considerablemente menores.



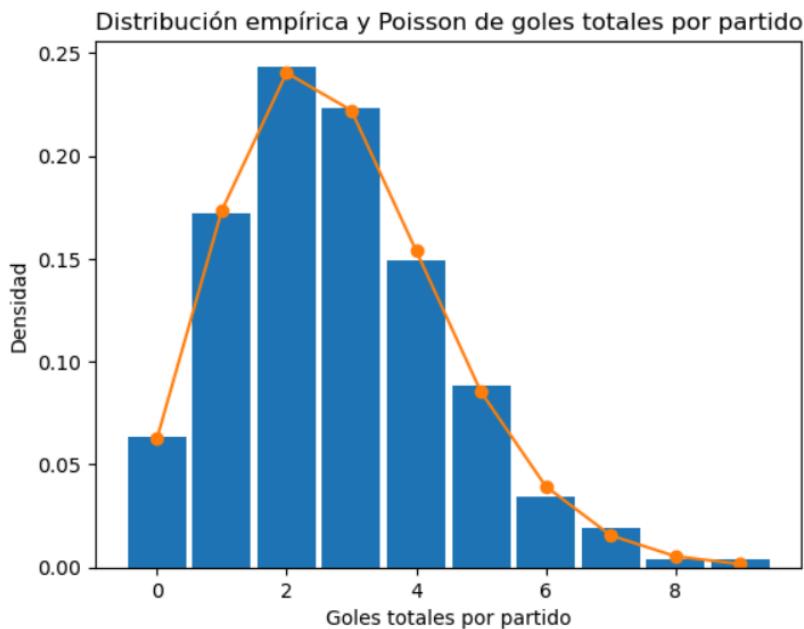
**Figura 3.** Distribución de puntos finales por equipo en las temporadas utilizadas como muestra de estimación

Estos resultados descriptivos justifican el uso de un **modelo de conteo (Poisson)** para los goles y sirven como referencia para evaluar luego la calidad de las simulaciones.

## 9.2. Evaluación básica del ajuste del modelo

Para verificar de manera preliminar si el modelo propuesto es compatible con los datos observados se realizaron las siguientes comparaciones:

- Se comparó la **distribución empírica de goles por partido** (histórica) con la distribución teórica de Poisson usando los parámetros  $\lambda$  estimados. Visualmente, las curvas resultaron razonablemente cercanas para goles bajos (0–3), que concentran la gran mayoría de los casos.



**Figura 4.** Distribución empírica y distribución de Poisson de los goles totales por partido en la muestra de estimación

- Se contrastó el **promedio de goles por partido** (local y visitante) observado en los datos con el promedio obtenido en un conjunto de temporadas simuladas. Las medias simuladas se ubicaron muy próximas a las históricas, lo que indica que el modelo reproduce correctamente el nivel global de anotación.

Condición	Media histórica	Media simulada
Local	1.800000	1.508297
Visitante	1.478947	1.225163

**Tabla 2.** Medias de goles por partido según localia: valores históricos y simulados

Si bien no se realizó un test estadístico de bondad de ajuste para todas las combinaciones equipo-condición, estas verificaciones descriptivas sugieren que el modelo es **adecuado como aproximación inicial** para la simulación de temporadas.

### 9.3. Comportamiento de los puntos en las temporadas simuladas

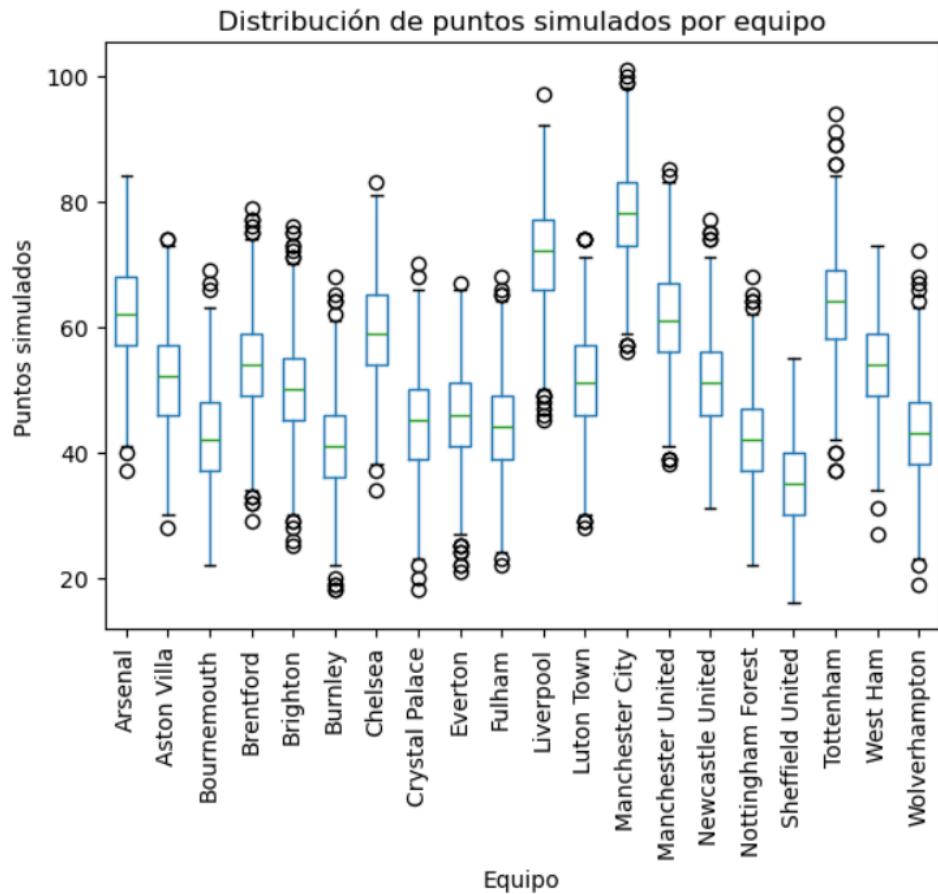
Con el modelo calibrado se procedió a simular un número elevado de temporadas completas sobre el fixture de la temporada de validación. Para cada equipo se analizó la distribución de puntos obtenidos en las N réplicas, a partir de la tabla de resultados generada y resumida con funciones de agregación (media, desvío estándar y percentiles):

team	mean_points	sd_points	p10	p50	p90
Manchester City	78.336	7.348000	69.0	78.0	87.0
Liverpool	71.513	7.949656	61.0	72.0	82.0
Tottenham	63.543	8.182488	53.0	64.0	74.0
Arsenal	62.033	7.831810	52.0	62.0	72.0
Manchester United	61.380	7.917846	51.0	61.0	72.0
Chelsea	59.206	8.018095	48.0	59.0	70.0
West Ham	54.069	7.633775	44.0	54.0	64.0
Brentford	53.811	7.998453	43.0	54.0	64.0
Aston Villa	52.020	7.766848	42.0	52.0	62.0
Luton Town	51.228	7.896478	41.0	51.0	61.0
Newcastle United	50.851	7.875203	40.0	51.0	61.0
Brighton	50.077	7.942427	40.0	50.0	60.0
Everton	45.879	7.775656	36.0	46.0	56.0
Crystal Palace	44.338	8.020853	34.0	45.0	54.0
Fulham	44.182	7.264134	35.0	44.0	54.0
Wolverhampton	43.225	7.573489	34.0	43.0	53.0
Bournemouth	42.614	7.440589	33.0	42.0	53.0
Nottingham Forest	41.598	7.280343	32.0	42.0	51.0
Burnley	40.925	7.598362	32.0	41.0	51.0
Sheffield United	35.044	6.841116	26.0	35.0	44.0

**Tabla 3.** Resumen de puntos simulados por equipo: media, desvío estándar y percentiles 10, 50 y 90

- Los equipos que en la temporada real se ubicaron en la parte alta de la tabla tienden, en las simulaciones, a concentrarse también en la zona superior de puntos,

aunque con una distribución extendida que muestra la **variabilidad posible** en su rendimiento.

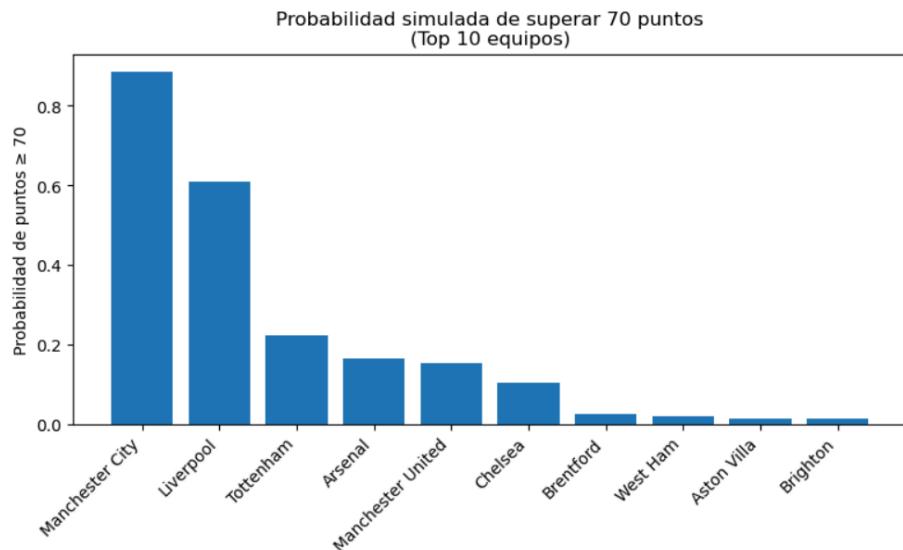


**Figura 6.** Distribución de puntos simulados por equipo en las temporadas ficticias generadas

- Del mismo modo, los equipos que históricamente estuvieron cerca del descenso presentan en las simulaciones una alta probabilidad de terminar en la parte baja, pero no siempre en la misma posición, reflejando que pequeños cambios en los resultados de algunos partidos pueden modificar el desenlace final.

A partir de estas distribuciones se pueden extraer indicadores como:

- la **media** y el **desvío estándar** de puntos para cada equipo,
- los **percentiles** (por ejemplo, 10, 50 y 90) que describen un rango típico de puntos,
- y la **probabilidad estimada** de que cada equipo supere cierto umbral (por ejemplo, clasificar a copas, evitar el descenso, etc.).



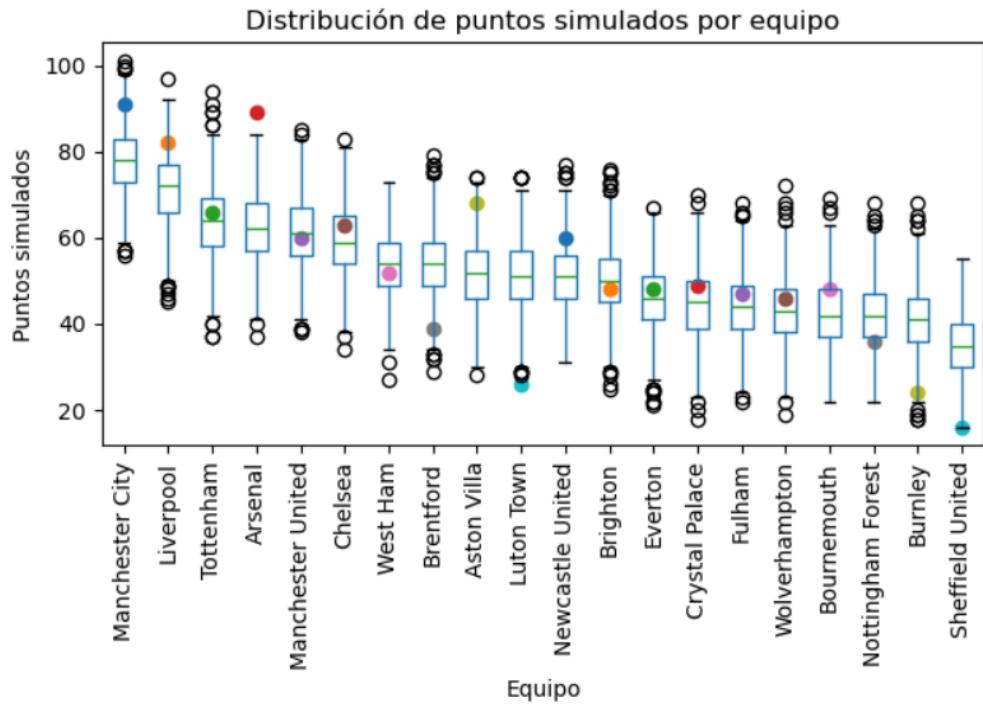
**Figura 7.** Probabilidad simulada de que los equipos superen un umbral de puntos en la temporada

En general, se observa que, incluso manteniendo fijos los parámetros derivados de los resultados reales, la tabla de posiciones resultante puede variar significativamente entre una temporada simulada y otra.

#### 9.4. Comparación con la temporada real

Para evaluar la capacidad del modelo de reproducir la realidad, se compararon las simulaciones con la temporada real de validación:

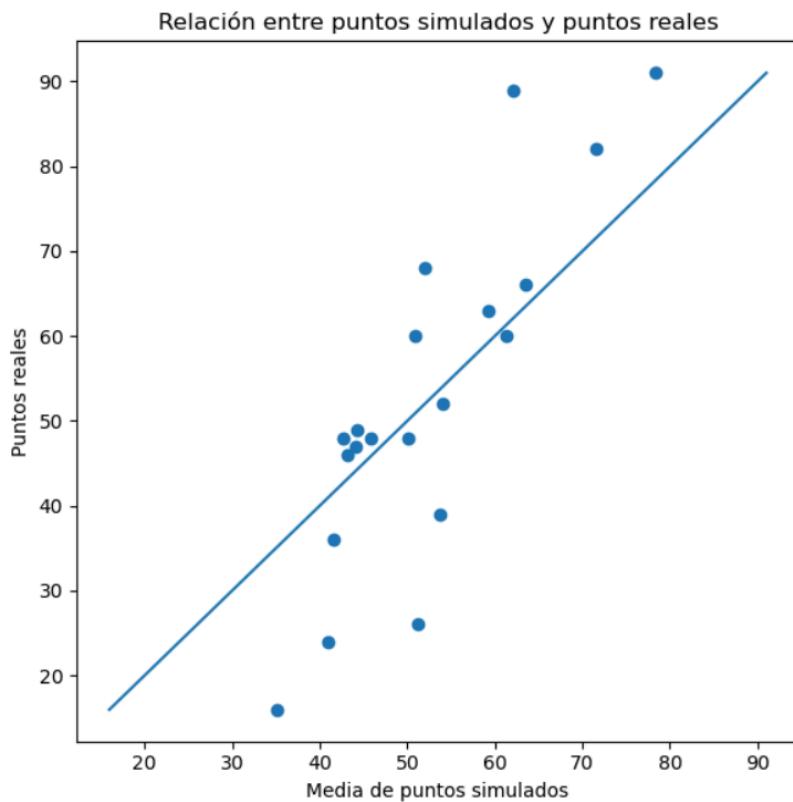
- En muchos casos, el número de puntos reales de los equipos se ubica dentro del rango central de la distribución simulada (por ejemplo, cerca de la mediana o dentro del intervalo intercuartil), lo que sugiere que el desenlace observado es **compatible** con el modelo probabilístico planteado.



**Figura 8.** Distribución de puntos simulados por equipo y posición del valor real de la temporada de validación

- En otros casos puntuales, ciertos equipos muestran desempeños reales que se encuentran en la parte más extrema de la distribución simulada, lo que podría interpretarse como temporadas “atípicamente buenas” o “atípicamente malas” según

el modelo.



**Figura 9.** Relación entre la media de puntos simulados y los puntos reales por equipo

Esta comparación no busca “demostrar” que el modelo sea exacto, sino mostrar que:

- La temporada real es solo **una realización posible** dentro de un conjunto amplio de desenlaces plausibles.
- El modelo permite cuantificar cuán “probables” o “extraordinarios” pueden ser algunos rendimientos cuando se tiene en cuenta el componente aleatorio del juego.

## 9.5. Visualizaciones y tablas resumen

Los principales hallazgos se acompañan con:

- **Tablas resumen** que muestran, para cada equipo, la media de puntos simulados, el desvío estándar y algunos percentiles relevantes (por ejemplo, p10, p50 y p90).
- **Gráficos de distribución** (diagramas de caja o histogramas) de los puntos obtenidos en las simulaciones, comparando la posición del valor real dentro de esa distribución.

- **Gráficos de barras** que ilustran, para ciertos equipos seleccionados, la probabilidad de terminar en diferentes rangos de la tabla (primeros puestos, zona media, zona baja).

Estas visualizaciones ayudan a interpretar de manera intuitiva la variabilidad de los resultados y complementan el análisis numérico presentado en las secciones anteriores.

## 10. Discusión

### 10.1. Interpretación de los resultados obtenidos

Los resultados de la simulación muestran que, aun manteniendo constantes los parámetros estimados a partir de los datos históricos, la **tabla de posiciones de la Premier League puede variar considerablemente** de una temporada a otra. Equipos que suelen ocupar los primeros lugares mantienen, en general, una alta probabilidad de terminar en la parte alta de la tabla, pero no siempre en la misma posición ni con la misma cantidad de puntos. De manera análoga, los equipos más débiles tienden a ubicarse en la parte baja, aunque con un rango amplio de posibles desempeños.

Esto refleja una idea central: el rendimiento observado en una temporada concreta no es únicamente consecuencia de la “calidad” de los equipos, sino también del **azar inherente** a cada partido. Pequeñas variaciones en algunos resultados (goles sobre la hora, rebotes, penales, etc.) pueden acumularse y producir desenlaces finales muy distintos.

La comparación entre las simulaciones y la temporada real de validación sugiere que, en la mayoría de los casos, los puntos efectivos de los equipos se encuentran dentro del rango de valores plausibles según el modelo. Cuando esto no ocurre, es decir, cuando un equipo queda muy por encima o muy por debajo de lo esperado, puede interpretarse como una temporada “excepcional” desde el punto de vista del modelo, lo cual abre la puerta a pensar en **factores no incluidos** (cambios tácticos relevantes, lesiones, refuerzos, calendario, etc.).

### 10.2. Fortalezas del enfoque utilizado

El enfoque adoptado presenta varias fortalezas:

- **Simplicidad e interpretabilidad:** el modelo de goles basado en una distribución de Poisson es relativamente sencillo, requiere pocos parámetros y se conecta de forma directa con estadísticas básicas de los datos (medias de goles por equipo y condición).
- **Alineación con los contenidos de la materia:** el trabajo integra de forma clara conceptos de modelos estocásticos, variables aleatorias, distribuciones de probabilidad y simulación de Monte Carlo, aplicados a un caso de estudio concreto.

- **Capacidad de generar escenarios alternativos:** la simulación de un gran número de temporadas permite cuantificar explícitamente la variabilidad de los resultados y estimar probabilidades de eventos de interés (clasificar a copas, evitar el descenso, etc.).
- **Uso eficiente de datos reales:** la calibración de los parámetros a partir de datos históricos otorga realismo al modelo dentro de su nivel de complejidad, aprovechando la información disponible sin requerir datos adicionales.

### **10.3. Limitaciones del modelo**

Al mismo tiempo, el modelo presenta limitaciones importantes que deben ser tenidas en cuenta al interpretar los resultados:

- **Independencia entre partidos**  
Se asume que los partidos son independientes y que los parámetros de cada equipo no cambian con el tiempo. En la realidad, el rendimiento puede verse afectado por rachas, confianza, lesiones, acumulación de partidos o cambios tácticos, que no están representados en el modelo.
- **Ausencia de información contextual**  
El modelo solo utiliza goles y condición de local/visitante. No se incorporan datos más ricos como ocasiones creadas, disparos al arco, tarjetas, lesiones o calendario de competencias paralelas, que podrían ayudar a explicar diferencias en rendimiento.
- **Suposición de Poisson para los goles**  
Si bien la Poisson es una elección razonable para conteos, en algunos contextos el número de goles puede presentar sobredispersión o patrones que no se ajustan del todo a esta distribución. El trabajo no explora en profundidad alternativas como la binomial negativa o modelos mixtos, ni compara sistemáticamente distintos ajustes.
- **Fixture fijo y reglas simplificadas**  
Se replica el fixture real de la temporada de validación y se consideran reglas estándar de la liga. Otros factores como criterios detallados de desempate, decisiones de árbitros o cambios reglamentarios no se incluyen.

Estas limitaciones no invalidan el modelo, pero sí marcan el **alcance** de las conclusiones: se trata de una representación simplificada y útil con fines académicos, más que de una herramienta de predicción profesional.

### **10.4. Posibles extensiones y mejoras**

El trabajo deja abiertas varias líneas posibles de mejora:

- Incorporar parámetros más específicos por **ataque y defensa**, separando tasas de goles a favor y en contra, y combinándolas para determinar la intensidad esperada de goles en cada partido.
- Considerar modelos que dependan no solo de la localía, sino también de la **fortaleza relativa del rival**, de modo que la intensidad de goles varíe según la combinación de equipos.
- Evaluar el uso de otras distribuciones de probabilidad para los goles, o incluso **distribuciones empíricas** directamente derivadas de los datos históricos.
- Introducir una **dinámica temporal sencilla**, donde los parámetros de los equipos puedan variar ligeramente a lo largo de la temporada (por ejemplo, según su rendimiento acumulado o por tramos del calendario).
- Ampliar el análisis a más temporadas o a otras ligas para comparar comportamientos y verificar la robustez del modelo.

Estas extensiones permitirían aumentar el realismo del modelo, aunque al costo de agregar complejidad y requerir más datos y trabajo de calibración.

## 11. Conclusiones

### 11.1. Conclusiones generales del trabajo

El trabajo mostró que las técnicas de **modelos y simulación** son herramientas adecuadas para analizar competiciones deportivas como la Premier League. A partir de un modelo probabilístico relativamente simple para la generación de goles por partido (basado en distribuciones de Poisson) y de la aplicación del método de **Simulación de Monte Carlo**, fue posible:

- recrear temporadas ficticias consistentes con los datos históricos,
- cuantificar la variabilidad potencial en los puntos y posiciones finales,
- y poner en perspectiva el desenlace real de una temporada como una realización más dentro de un abanico amplio de escenarios posibles.

En este sentido, el estudio refuerza la idea de que el fútbol es un sistema en el que conviven **habilidad, estrategia y azar**, y en el que el componente aleatorio puede tener un peso significativo en la determinación de los resultados finales, incluso cuando se mantienen fijas las tasas medias de gol de los equipos.

### 11.2. Aprendizajes desde la perspectiva de Modelos y Simulación

Desde el punto de vista de la materia, el trabajo permitió:

- Aplicar de manera concreta los conceptos de **modelo estocástico, variables aleatorias discretas y distribuciones de probabilidad** a un caso de estudio real.
- Implementar un esquema completo de **Simulación de Monte Carlo**, desde la calibración de parámetros a partir de datos históricos hasta la generación masiva de réplicas y el análisis estadístico de las salidas.
- Experimentar con la importancia del **número de réplicas** y con la interpretación de medidas como medias, desvíos estándar, percentiles y probabilidades de eventos.

En síntesis, el trabajo integró el **marco teórico** de la asignatura con una **aplicación práctica**, facilitando la comprensión de los conceptos a través de un ejemplo conocido y motivador como es una liga de fútbol.

### 11.3. Reflexión final

Aunque el modelo utilizado es deliberadamente sencillo, sus resultados muestran de manera clara que:

- la temporada observada no es “la única” que podría haber ocurrido con esos equipos y ese fixture,
- y que la simulación es una herramienta poderosa para explorar “**mundos posibles**” a partir de información limitada.

En trabajos futuros, profundizar en modelos más detallados y en datos adicionales podría permitir respuestas más finas a preguntas específicas (como probabilidad de título, clasificación a copas o impacto de ciertos cambios en el reglamento), manteniendo siempre el equilibrio entre **complejidad del modelo e interpretabilidad de los resultados**.

## 12. Referencias bibliográficas

- **Apuntes de cátedra**
  - Cátedra de Modelos y Simulación. (2025). *Apuntes de clase y material de cátedra*. Universidad Nacional de San Luis.
- **Libro de referencia en simulación**
  - Law, A. M. (2015). *Simulation Modeling and Analysis* (5<sup>a</sup> ed.). McGraw-Hill.

- **Libro/Apunte de probabilidad/estadística**
  - Ross, S. M. (2014). *Introduction to Probability Models* (11<sup>a</sup> ed.). Academic Press.
- **Fuente de datos (estadísticas de la Premier League)**
  - FBref. (s. f.). *Premier League Scores & Fixtures, Seasons 2020–2023*. Recuperado de Kaggle. “Premier League Matches 2020 to 2024 Dataset”
  - <https://www.kaggle.com/datasets/sajkazmi/premier-league-matches?resource=download>
- **Librerías de software**
  - Python Software Foundation. *Python (versión 3.11)*
  - The pandas development team. *pandas*
  - Harris, C. R. et al. (2020). *Array programming with NumPy*. *Nature*, 585, 357–362.

## 13. Anexo

- [Jupyter Notebook](#)
- [Base](#)
- [Modulo](#)