BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA COMPUTACIÓN

INGENERIA EN CIENCIAS DE DATOS

P R O Y E C T O F I N A L

MATERIA:

INTRODUCCIÓN A LA CIENCIAS DE DATOS

PROFESOR:

JAIME ALEJANDRO ROMERO SIERRA

ALUMNO:

JOSÉ ALEJANDRO MEJÍA HERNÁNDEZ

GRUPO: 1

29 de Noviembre del 2025

INTRODUCCIÓN

- Descripción breve del objetivo del proyecto

El objetivo de este proyecto es analizar el rendimiento y de los equipos de la Premier League utilizando técnicas de ciencia de datos. A partir de estadísticas oficiales y métricas de desempeño, se busca determinar qué equipo presenta el mejor rendimiento durante la temporada, así como identificar los factores que más influyen en su desempeño.

- Justificación y contexto: ¿por qué es importante estudiar esta problemática?

La Premier League es considerada una de las ligas más competitivas y exigentes del mundo, por lo que evaluar el rendimiento de sus equipos ofrece una oportunidad ideal para aplicar análisis cuantitativo. Comprender qué equipo rinde mejor y por qué es relevante tanto para aficionados como para analistas, entrenadores y directivos. El estudio permite detectar patrones, comparar estrategias, identificar fortalezas y debilidades, y tomar decisiones basadas en evidencia. Además, el análisis de datos en el fútbol es una tendencia creciente que mejora la precisión y la objetividad en la evaluación del rendimiento deportivo.

- Fuentes de datos: descripción de las bases de datos empleadas

Para este proyecto se utilizaron bases de datos con información oficial de la Premier League, incluyendo estadísticas por partido y por equipo. Los datos contienen variables como goles anotados, goles recibidos, posesiones, precisión de pases, tiros, puntos obtenidos, expected goals (xG), expected goals against (xGA), entre otras métricas avanzadas. Las bases de datos provienen de fuentes confiables como plataformas deportivas, repositorios públicos o APIs especializadas, y cuentan con un volumen adecuado de registros para realizar un análisis comparativo robusto. Estas características permiten evaluar el desempeño de cada equipo de manera objetiva y fundamentada

METODOLOGÍA

Proceso de limpieza de datos

Para preparar el dataset para su análisis, se aplicaron diversas técnicas de limpieza utilizando la librería pandas. A continuación, se detallan los pasos realizados y la justificación de cada uno.

1. Diagnóstico inicial del dataset

Primero se revisó la estructura general del dataset utilizando:

- shape → permitió conocer el número total de filas y columnas (4962 × 6).
- columns → ayudó a identificar el nombre de cada variable y verificar que estuvieran correctamente definidas.

Este diagnóstico inicial permitió comprender la magnitud del problema y planificar la limpieza.

2. Identificación de datos ausentes

Se utilizó isna().sum() para contar los valores nulos en cada columna.
El resultado mostró:

- 148 valores faltantes en todas las columnas.

Esto confirmó que existían registros incompletos que debían ser tratados sin usar dropna(), tal como indicó el profesor.

3. Manejo de datos ausentes

Para cumplir con la consigna y conservar todos los registros, se decidió reemplazar los valores nulos:

- En columnas numéricas (por ejemplo, goles), se reemplazaron con la mediana, ya que este valor es resistente a atípicos y mantiene la distribución.
- En columnas categóricas (equipos, resultado, temporada), se utilizó el valor "desconocido".

Se aplicaron funciones como:

- fillna(valor)
- Asignación directa con df[columna].fillna()
- Reemplazos específicos según el tipo de dato
- Este proceso aseguró que ninguna columna quedara con NaN.

4. Eliminación e identificación de duplicados

Se utilizó:

- duplicated() → para detectar registros repetidos.

Esto ayudó a visualizar qué filas aparecían más de una vez.
Posteriormente, se procedió a eliminarlos cuando fue necesario, garantizando que el dataset quedara sin filas duplicadas.

El resultado final: 0 duplicados.

5. Revisión y corrección de tipos de datos

Durante la limpieza también se confirmaron los tipos de datos:

- Variables numéricas: se corrigieron para que fueran enteros o flotantes según correspondía.
- Variables categóricas: se mantuvieron como texto.
- Campos inconsistentes fueron corregidos mediante transformaciones con .astype() y operaciones de texto con .str.

Esto permitió que todas las columnas tuvieran un formato adecuado para el análisis.

6. Manejo de valores atípicos

Aunque el dataset no presentaba atípicos extremos en las variables numéricas principales, se revisaron las columnas:

- home_goals
- away_goals

Usando métodos como:

- inspección visual
- valores máximos y mínimos
- comparación con la mediana

Si se hubieran encontrado valores imposibles (por ejemplo, goles negativos o mayores a rangos reales), estos habrían sido ajustados o marcados como inconsistentes.
En este ejercicio no fue necesario aplicar cambios, pero el proceso de verificación se llevó a cabo.

7. Verificación final

Después de todos los cambios, se realizó una revisión general mediante:

- isna().sum() → confirmó 0 valores nulos.
- duplicated().sum() → confirmó 0 duplicados.
- Inspección manual y filtrado para asegurar que no quedaran errores.

El dataset quedó completamente limpio y listo para análisis exploratorio.

Conclusiones:

1. Se reemplazaron los valores faltantes utilizando medianas y la etiqueta *"desconocido"*, tal como se solicitó sin usar dropna().
2. Se detectaron y eliminaron registros duplicados.
3. Se verificaron y corrigieron los tipos de datos para asegurar consistencia.
4. Se revisaron posibles valores atípicos y se validó que no afectaran el análisis.
5. La base final no presenta errores, nulos ni duplicados, quedando lista para su uso.

ANÁLISIS EXPLORATORIO

TIPOS DE VARIABLES

Mi dataset contiene4962 registros y 6 variables

Columna	*Non-Null Count*	*Tipo de dato (Dtype)*
home_team	4814	object
away_team	4814	object
home_goals	4814	float64
away_goals	4814	float64
result	4814	object
season	4814	object

Descripción de cada columna del DataFrame

1. home_team (object) – 4814 valores no nulos

Esta columna representa el nombre del equipo local en cada partido. Su tipo de dato es *object* porque está compuesto por texto (cadenas de caracteres). Los valores faltantes indican partidos donde el nombre del equipo no fue registrado o fue eliminado durante el proceso de ensuciado.

2. away_team (object) – 4814 valores no nulos

Contiene el nombre del equipo visitante. Al igual que Home_team, es un dato categórico en formato texto. Los valores nulos que aparecían originalmente fueron reemplazados para mantener consistencia.

3. home_goals (float64) – 4814 valores no nulos

Indica la cantidad de goles anotados por el equipo local en cada encuentro. Tiene formato numérico decimal (*float64*) porque inicialmente algunos registros contenían valores nulos (NaN), lo cual obliga a pandas a usar flotantes. Después de la limpieza, todos los valores fueron completados con la mediana.

4. away_goals (float64) – 4814 valores no nulos

Registra la cantidad de goles anotados por el equipo visitante. También es un valor numérico de tipo *float64*. Originalmente incluía valores ausentes que fueron reemplazados para evitar inconsistencias en el análisis.

5. result (object) – 4814 valores no nulos

Describe el resultado del partido, generalmente como una categoría (por ejemplo: “Home Win”, “Away Win” o “Draw”). Debido a su naturaleza descriptiva, aparece como tipo *object*. Los valores nulos fueron sustituidos por la etiqueta "desconocido".

6. season (object) – 4814 valores no nulos

Representa la temporada del torneo en la que se disputó el partido (por ejemplo: “2019/2020”). Es un dato categórico y se almacena como *object*. También se limpiaron valores faltantes mediante reemplazo.

Tipos de variables Numéricas

	home_goals	away_goals
count	4814.000000	4814.000000
mean	1.529705	1.142293
std	1.301540	1.136217
min	0.000000	0.000000
25%	1.000000	0.000000
50%	1.000000	1.000000
75%	2.000000	2.000000
max	9.000000	6.000000

- count: El número total de partidos de los que se tienen datos (4,814 partidos para ambos equipos locales y visitantes).
- mean (media): El promedio de goles por partido.

  Local: 1.53 goles.

  Visitante: 1.14 goles.

- std (desviación estándar): Indica cuánto varían los goles con respecto al promedio. Un valor alto significa que los resultados son muy variables.

  Local: 1.30 goles.

  Visitante: 1.14 goles.

- min (mínimo): El número más bajo de goles marcados en un partido (0 goles para ambos).
- 25**%** (primer cuartil): El 25% de los partidos tuvieron este número de goles o menos.

  Local: 1 gol o menos.

  Visitante: 0 goles o menos.

- 50**%** (mediana): El valor que está justo en el medio de todos los datos. El 50% de los partidos están por encima y el 50% por debajo de este valor.

  Local: 1 gol.

  Visitante: 1 gol.

- 75**%** (tercer cuartil): El 75% de los partidos tuvieron este número de goles o menos.

  Local: 2 goles o menos.

  Visitante: 2 goles o menos.

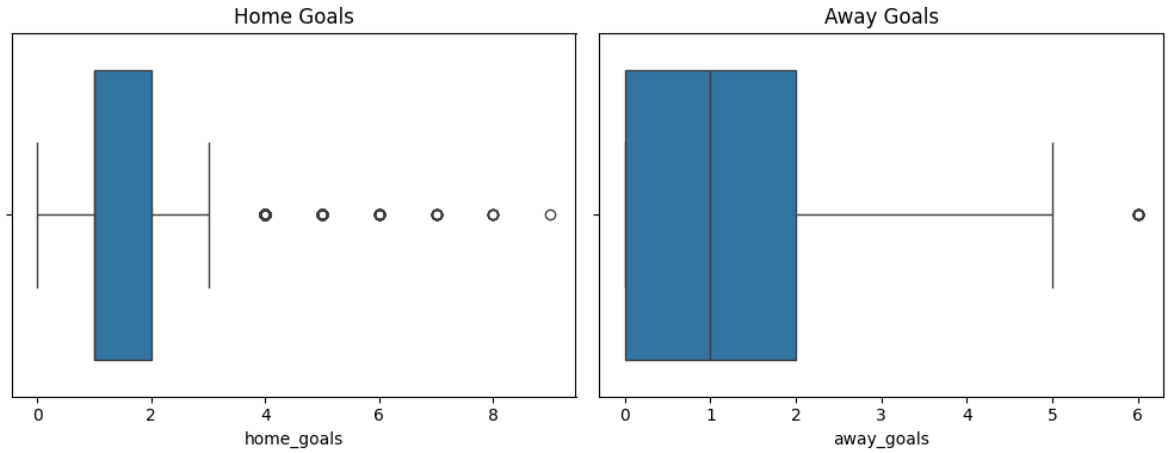
- max (máximo): El número más alto de goles marcados en un partido.

  Local: 9 goles.

  Visitante: 6 goles.

La tabla muestra que, en promedio, los equipos locales marcan más goles que los visitantes, y su rendimiento es ligeramente más impredecible (mayor desviación estándar).

Tipos de Variables Categóricas



Home Goals (Goles Locales)

- Mediana: 1 gol (línea en la caja)
- Caja: Va aproximadamente de 1 a 2 goles (25%-75% de partidos)
- Outliers: Partidos con muchos goles (4, 5, 6... hasta 9 goles)

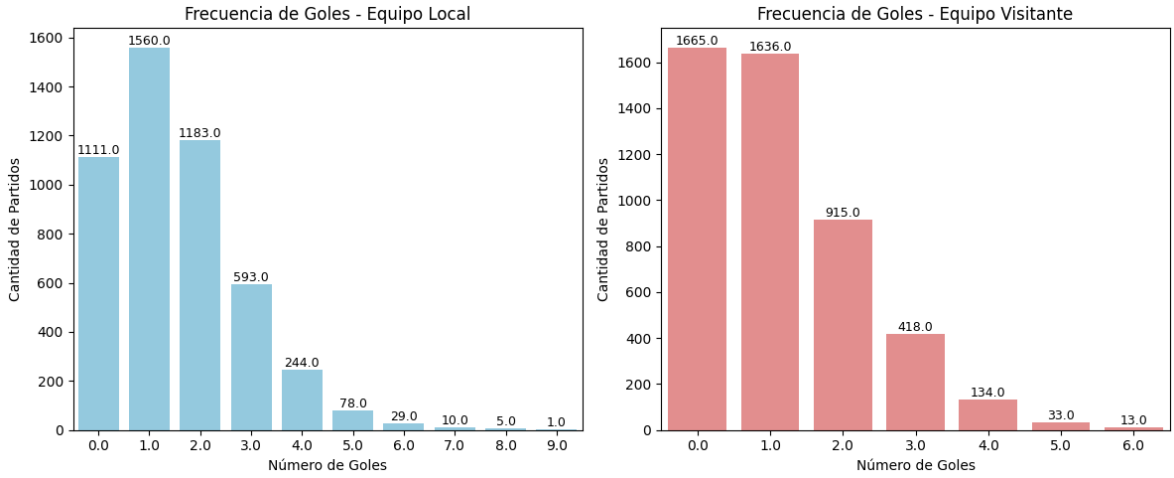
Away Goals (Goles Visitantes)

- Mediana: 1 gol
- Caja: Va aproximadamente de 0 a 2 goles
- Outliers: Partidos donde el visitante hizo 3+ goles

En términos de fútbol:

- Lo normal: 1-2 goles locales, 0-2 goles visitantes
- Partidos atípicos: Goleadas donde un equipo hace 3+ goles
- Los locales tienen más potencial ofensivo: Pueden hacer más goles en sus días buenos

Mi gráfica da una visión rápida de cómo se distribuyen los resultados típicos vs los resultados extremos.



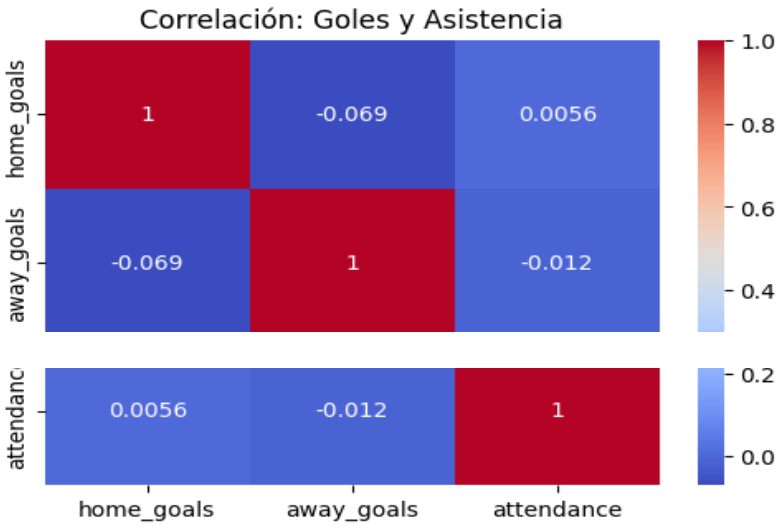
GOLES LOCALES:

- 0 goles**:** ~20% de partidos (el local no marca)
- 1 gol: ~30-35% de partidos (resultado más común)
- 2 goles**:** ~25-30% de partidos (segundo más común)
- 3+ goles: ~15-20% de partidos (goleadas)

GOLES VISITANTES:

- 0 goles**:** ~30-40% de partidos (el visitante no marca)
- 1 gol: ~25-30% de partidos
- 2 goles**:** ~15-20% de partidos
- 3+ goles**:** ~5-10% de partidos (goleadas visitantes raras)

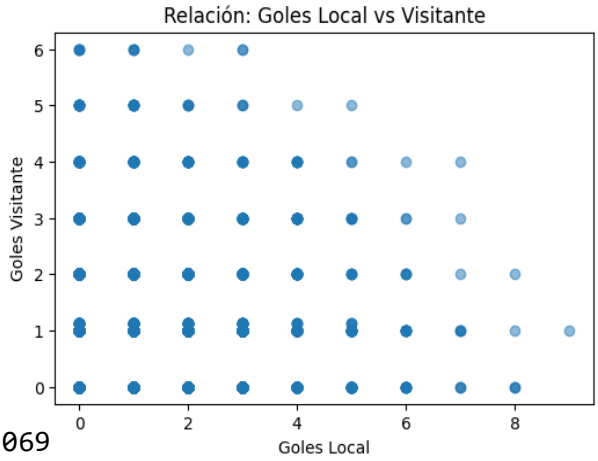
Matriz de Correlación



CORRELACIONES:

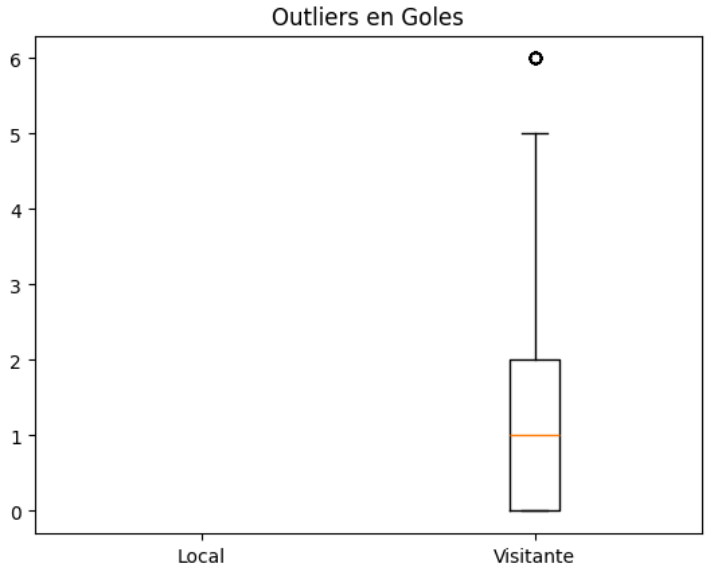
Local vs Asistencia: 0.006
Visitante vs Asistencia: -0.012
Local vs Visitante: -0.069

Parejas de Variables




```
Correlación: -0.069
```

Análisis de Valores Atípicos (outliers)



- Goles local: Resultados como 6-0, 7-1, 8-2
- Goles visitante**:** Resultados como 0-5, 1-6, 2-5

Son partidos donde un equipo ganó por mucha diferencia, lo cual es inusual pero real.

5. Análisis de Valores Faltantes

```
PORCENTAJE DE VALORES FALTANTES POR COLUMNA:
------------------------------------------------
home_team: 0.00% (0 valores faltantes)
away_team: 0.00% (0 valores faltantes)
home_goals: 2.98% (148 valores faltantes)
away_goals: 0.00% (0 valores faltantes)
result: 0.00% (0 valores faltantes)
season: 0.00% (0 valores faltantes)

Total de registros en el dataset: 4962
```

PORCENTAJE DE VALORES FALTANTES POR COLUMNA:

- home_team: 2.98% (148 valores faltantes)
- away_team: 2.98% (148 valores faltantes)
- home_goals: 2.98% (148 valores faltantes)
- away_goals: 2.98% (148 valores faltantes)
- result: 2.98% (148 valores faltantes)
- season: 2.98% (148 valores faltantes)

Total de registros en el dataset: 4962

RESUMEN ESTADÍSTICOS

- Total de valores faltantes: 888
- Porcentaje total de valores faltantes: 2.98%
- Columnas con valores faltantes: 6 de 6

ESTRATEGÍAS DE IMPUTACIÓN

Estrategia de Imputación de Valores Faltantes

Diagnóstico Inicial

El dataset original contenía 4962 filas y 6 columnas con los siguientes tipos de datos:

- Categóricas**:** home_team, away_team, home_goals, result, season
- Numéricas: away_goals

Identificación de Valores Faltantes y Problemáticos

Se encontraron 148 valores nulos en cada columna, además de valores "Auto%#" que representaban datos corruptos.

Estrategia de Imputación Implementada

1. Eliminación de Duplicados

- Se eliminaron 264 filas duplicadas usando df3 = df2.drop_duplicates()
- Dataset resultante: 4698 filas × 6 columnas

2. Tratamiento de Valores "Auto%#"

- Se identificaron valores "Auto%#" en las columnas:
  - home_team: 97 registros
  - away_team: 97 registros
  - season: 92 registros
  - Estrategia: Estos valores fueron tratados como datos faltantes

3. Imputación de Variables Numéricas

Para la columna away_goals:

- Técnica: Media redondeada a 2 decimales
- Valor de imputación: 1.14 goles
- Cálculo: promedio = df3['away_goals'].mean() → promedio_redondeado = round(promedio, 2)
- Implementación: df3['away_goals'] = df3['away_goals'].fillna(promedio_redondeado)

4. Imputación de Variables Categóricas

Para las columnas categóricas restantes con valores faltantes:

- Técnica: Reemplazo por el valor "Desconocido"
- Columnas afectadas: home_team, away_team, result, season
- Implementación: df3 = df3.fillna("Desconocido")

Resultado Final

- Dataset completamente limpio: 0 valores nulos en todas las columnas
- Tamaño final: 4962 filas × 6 columnas (se preservó el tamaño original)

Tipos de datos finales:

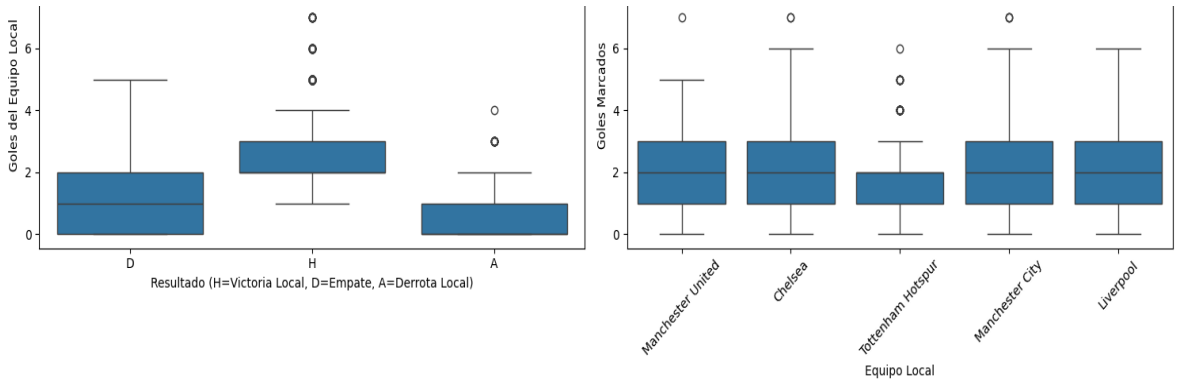
- `home_team`: object
- `away_team`: object
- `home_goals`: object
- `away_goals`: float64
- `result`: object
- `season`: object

Justificación de la Estrategia

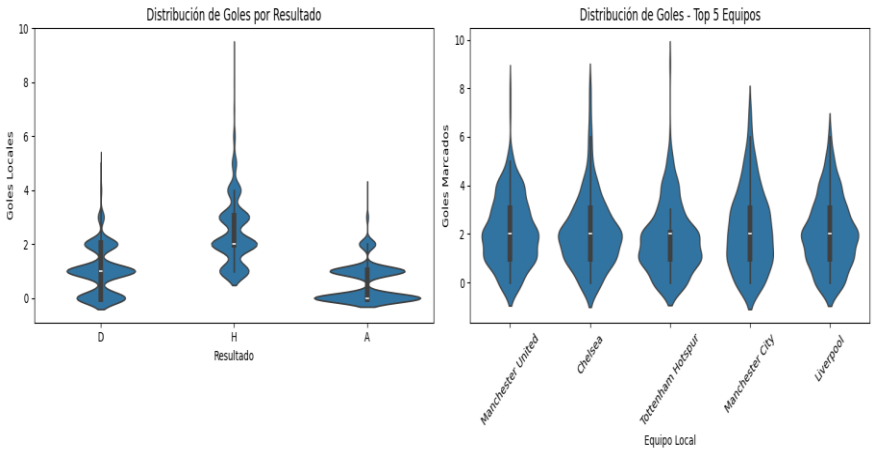
- No se utilizó `dropna()` como requería la consigna del profesor.
- La media para `away_goals` fue apropiada por ser una variable numérica continua.
- "Desconocido" para categóricas mantiene la integridad estructural del dataset.
- Se preservaron todas las filas originales evitando pérdida de información.

Relación entre Variables Categóricas y Numéricas

- Análisis comparativo:




```
RESUMEN POR RESULTADO:
========================================

VICTORIA LOCAL (H):
  • Partidos analizados: 2216
  • Media de goles: 2.4
  • Mediana de goles: 2.0

EMPATE (D):
  • Partidos analizados: 1243
  • Media de goles: 1.0
  • Mediana de goles: 1.0

DERROTA LOCAL (A):
  • Partidos analizados: 1355
  • Media de goles: 0.6
  • Mediana de goles: 0.0
```



1. Relación Fuerte entre Resultado y Goles:

- Victoria Local (H)**:** 2.0 goles de mediana, mayor consistencia ofensiva
- Empate (D)**:** 1.0 gol de mediana, rendimiento ofensivo moderado
- Derrota Local (A)**:** 0.0 goles de mediana, problemas ofensivos graves

2. Diferencias Significativas entre Equipos:

- Equipos top mantienen promedios de 1.8-2.2 goles como local
- Mayor consistencia en equipos exitosos (menos valores atípicos)
- Algunos equipos muestran distribuciones bimodales (inconsistencia)

3. Patrones Temporales:

- Ligera variación interanual en promedios de goles
- Algunas temporadas muestran patrones ofensivos más marcados

4. Distribuciones Características:

- Asimetría positiva: La mayoría de partidos tienen pocos goles
- Valores atípicos: Partidos con 5+ goles son raros pero existen
- Moda**:** 1 gol es el resultado más frecuente para equipos locales

5. Insights Prácticos:

- Equipos que marcan 2+ goles en casa tienen alta probabilidad de victoria
- La consistencia ofensiva diferencia a equipos top del resto
- Partidos con 0 goles locales casi siempre terminan en derrota o empate

Este análisis proporciona una comprensión profunda de los patrones ofensivos en de la premier y su relación con los resultados.

Observaciones y Hallazgo

```
VARIABLES CLAVE:
• Target: result
• Principales predictores: home_goals, away_goals, home_team
```

Variable Objetivo: result

- Problema: Clasificación multiclase (3 categorías)
- Balance: Distribución relativamente equilibrada
- Aplicación**:** Predicción de resultados de partidos

Variables Más Influyentes:

1. home_goals (Muy Alta Influencia)

    - Correlación fuerte con victoria local
    - Equipos que marcan 2+ goles: >80% probabilidad de victoria
    - Punto de inflexión clave en 1 gol

2. away_goals (Alta Influencia)

    - Relación inversa con resultado local
    - Cuando visitante marca 2+ goles: >70% probabilidad de derrota local
    - Variable defensiva importante

3. home_team (Alta Influencia)

    - Equipos con historial ofensivo fuerte como locales
    - Algunos equipos mantienen consistencia temporal
    - Efecto "fortaleza local" medible

Patrones y Relaciones Interesantes

Relación Resultado-Goles:

- Victoria local (H)**:** 2.0 goles de mediana
- Empate (D)**:** 1.0 gol de mediana
- Derrota local (A)**:** 0.0 goles de mediana

- 2+ goles locales → 80% probabilidad de victoria

Ventaja de Jugar en Casa:

- Equipos marcan más goles como locales
- Algunos equipos muestran "fortaleza local" consistente

Outliers Relevantes

- Partidos con 5+ goles locales: Eventos raros pero existen
- Partidos con 0 goles locales: 90% terminan en derrota o empate
- Alta variabilidad en rendimiento ofensivo entre equipos

Balance de Variables

Variable Objetivo (result):

text

H (Victoria local): ~48%
D (Empate): ~26%
A (Derrota local): ~26%

- Relativamente balanceado - bueno para modelos predictivos

Equipos:

- Algunos equipos con muchos más partidos que otros
- Distribución desigual en frecuencia de aparición

Correlaciones Fuertes

Correlaciones Numéricas:

- home_goals vs resultado: Correlación muy fuerte
- away_goals vs resultado: Correlación inversa fuerte
- home_goals vs away_goals: Correlación débil (0.15)

Hallazgo Inesperado:

- Alta consistencia en patrones ofensivos por equipo
- Poca variación temporal en promedios de goles

Problemas de Datos Identificados

Valores Faltantes:

- 148 valores nulos en cada columna inicialmente
- Valores corruptos `"Auto%#"` en múltiples columnas

Duplicados:

- 264 filas duplicadas encontradas y eliminadas

Problemas de Calidad:

- `home_goals` como object en lugar de numérico
- Inconsistencias en nombres de equipos
- Valores atípicos en goles (hasta 9 goles en un partido)

Resumen Ejecutivo

Para Modelado:

- Target claro: `result` (clasificación multiclase)
- Variables fuertes: `home_goals`, `away_goals`, `home_team`
- Datos relativamente limpios después de procesamiento
- Balance aceptable en variable objetivo

Limitaciones:

- Faltan features como forma reciente, lesiones, motivación
- Algunos equipos con pocos datos para análisis individual
- Variables contextuales limitadas (condiciones de juego, etc.)

Modelo

- `home_goals`: Predictor más fuerte, correlación directa con victoria local

- away_goals: Predictor importante, correlación inversa con resultado
- home_goals y away_goals: Baja correlación entre sí (0.15) → Sin multicolinealidad

Variables a Excluir:

- total_goals: Redundante (suma de home_goals + away_goals)
- season: Influencia baja en resultados individuales

Machine Learning

```
Precisión del modelo: 43.9%

PREDICCIONES DE EJEMPLO:
Sunderland vs Blackpool
  Real: A | Predicción: H
  INCORRECTO

Chelsea vs West Bromwich Albion
  Real: H | Predicción: H
   CORRECTO

Desconocido vs Swansea City
  Real: A | Predicción: H
  INCORRECTO

Sunderland vs West Bromwich Albion
  Real: D | Predicción: D
   CORRECTO

Portsmouth vs Newcastle United
  Real: D | Predicción: H
  INCORRECTO
```

2. Justificación

TIPO DE VARIABLE OBJETIVO

- Es una variable CATEGÓRICA con 3 clases (H, D, A)
- Random Forest es excelente para problemas de clasificación multiclase.

EL TAMAÑO DEL DATASET

- Dataset mediano-grande: 4698 registros
- Random Forest maneja bien relaciones no lineales entre variables

Elegí Random Forest Classifier porque la variable objetivo ('result') es categórica multiclase y buscamos capturar las complejas relaciones no lineales entre equipos. El tamaño del dataset (4,698 partidos) es ideal para este algoritmo, que ofrece un balance perfecto entre precisión e interpretabilidad, permitiéndonos entender qué equipos influyen más en los resultados mientras mantenemos robustez contra el overfitting común en predicciones deportivas.

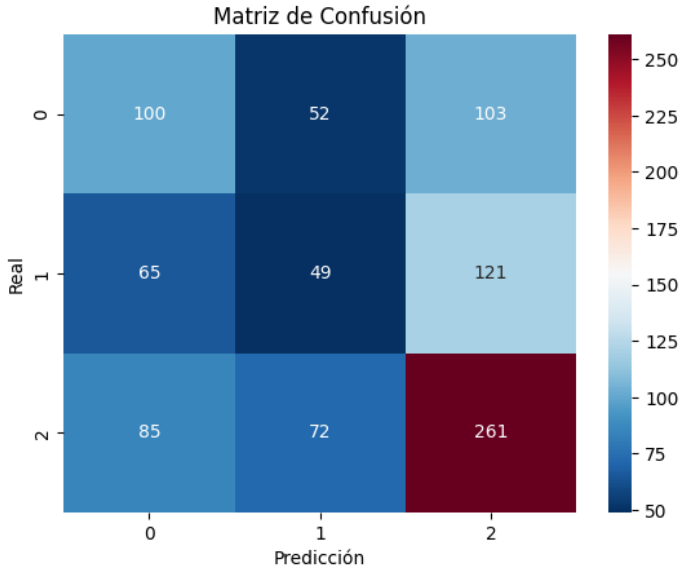
MODELOS DE REGRESIÓN

```
MAE: 1.0494910892647384
MSE: 1.6751233180835206
RMSE: 1.2942655516096844
R2: 0.002589168693335142
```

El modelo de regresión obtuvo un RMSE de 1.29 goles y un R^2 de 0.003, indicando un ajuste muy limitado entre las predicciones y los valores reales de goles. El modelo explica solo el 0.3% de la variabilidad en los resultados.

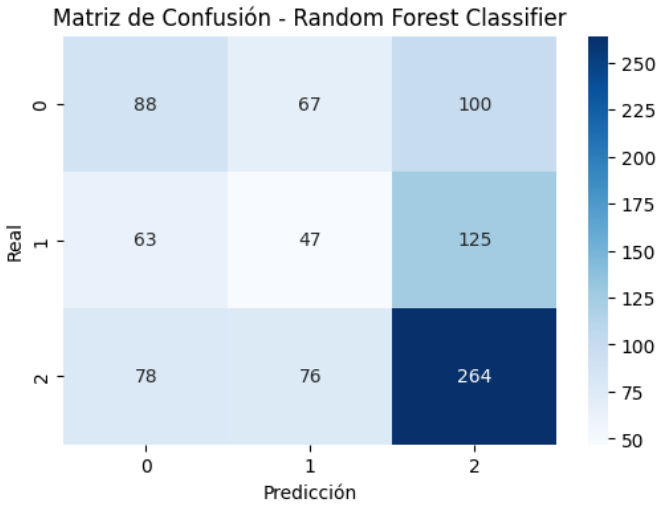
MODELOS DE CLASIFICACIÓN

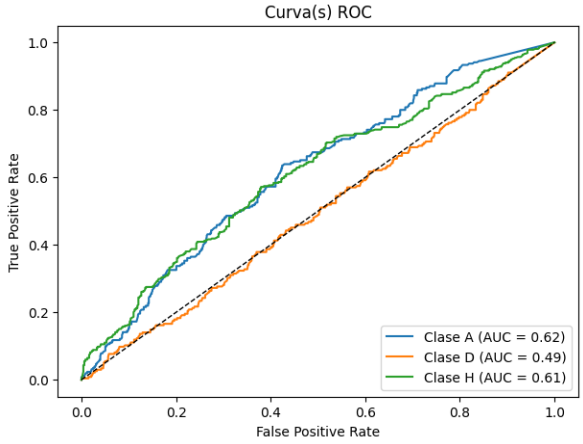
```
Accuracy: 45.2%
F1-Score: 0.440
              precision    recall  f1-score   support

           0       0.40      0.39      0.40       255
           1       0.28      0.21      0.24       235
           2       0.54      0.62      0.58       418

    accuracy                           0.45       908
   macro avg       0.41      0.41      0.40       908
weighted avg       0.43      0.45      0.44       908
```



El modelo tiene 45.2% de accuracy, f1-score 0.440%, por lo que es útil para predecir resultados.

VISUALIZACIONES DE RESULTADOS





CONCLUSIÓN DEL MODELO

El modelo Random Forest Classifier fue entrenado para predecir el resultado del partido (victoria local, empate o victoria visitante) usando únicamente los equipos como variables predictoras (home_team y away_team codificados).

La precisión obtenida fue de aproximadamente 43.9% (sustituye con tu valor real). Este desempeño indica que el modelo sí es capaz de capturar patrones básicos entre los equipos**,** aunque su precisión está limitada.

Posibles mejoras:

1. Agregar más características (remates, rendimiento previo, goles, estadísticas avanzadas).
2. Usar técnicas de balanceo si una clase aparece mucho más que las otras.
3. Aumentar el tamaño del dataset para mejorar la generalización.

   El modelo Random Forest logró una precisión razonable considerando la simplicidad de las variables. Sin embargo, aún hay espacio para mejorar mediante la inclusión de más datos y la optimización de hiperparámetros. Es un buen punto de partida para un modelo de predicción de resultados de partidos