



Tecnológico de Monterrey

PROYECTO INTEGRADOR

Alumnos: LUIS ENRIQUE AGUILERA NOVOA – A01646551
PABLO EMILIANO GONZALES RIOS – A01647192
ANGELO DAYVIS FARFAN TORRES – A01647200

Curso: Pensamiento Computacional

Grupo: 418

Fecha: 16 de octubre de 2024

SEMESTRE AGOSTO – DICIEMBRE

ÍNDICE:

PROYECTO INTEGRADOR	1
I. INTRODUCCIÓN:	3
II. MARCO TEÓRICO:	4
III. OBJETIVOS DEL TRABAJO:	5
IV. DESARROLLO DEL PROGRAMA:.....	5
V. MÉTODOS DE IMPLEMENTACIÓN:.....	11
VI. ANÁLISIS DE RESULTADOS:.....	12
VII. CONCLUSIÓN INDIVIDUAL:	13
VIII. CONCLUSIÓN POR EQUIPO:.....	14

I. INTRODUCCIÓN:

En los últimos años, el fútbol moderno se ha convertido en una industria multimillonaria, la inyección de dinero en clubes de fútbol provenientes en su gran mayoría de países pertenecientes a la península arábiga ha transformado de manera significativa el panorama competitivo del fútbol europeo de elite. La Premier League es el ejemplo más claro de cómo los clubes pueden mejorar sustancialmente su competitividad con la inyección de capital externo, especialmente la presencia de países pertenecientes a la península arábiga fue haciendo presencia en la mejor liga del mundo. Sin embargo, estas inversiones no siempre garantizan un éxito inmediato.

Equipos como el Newcastle United que fue adquirido en 2021 por el Fondo de Inversión Pública de Arabia Saudita y el Everton que ha recibido inyecciones de capital desde que Farhad Moshiri se convirtió en accionista mayoritario en 2016, proporcionaron a los clubes más recursos para reforzar los planteles con la contratación de jugadores y staff de elite, lograr una mejora significativa en infraestructura y sin duda alguna competir a un nivel más alto dentro del fútbol europeo. Por un lado, el Newcastle ha pasado de ser un equipo de tres cuartos de tabla a uno que compite clasificación a torneos europeos y por otro lado, el Everton tomó el camino contrario, estando en los últimos lugares de la liga y peleando su estancia. Estos casos presentan una oportunidad única para estudiar cómo las inversiones afectan a los clubes en distintos niveles de recursos y enfoques estratégicos.

Es por ello que, a base de datos recopilados en 7 temporadas, desde la 2017-2018 hasta la 2023-2024, utilizando herramientas de análisis de datos y programación en Python buscaremos dar respuesta a la cuestión de ¿Qué tan relevante es la inyección del capital para mantenerse en un estado de competencia constante en la Premier League de Inglaterra? Y de igual, no solo busquemos cuantificar el efecto de las inyecciones de capital, sino también busquemos proporcionar una base de datos que permita proyectar futuros escenarios en la Premier League, una liga que cada vez tiene más presencia de inyección extranjera.

La analítica de los datos consta de 7 ejes fundamentales para su desarrollo, se basará en diversas estadísticas de rendimiento, como la posición en la tabla de clasificación, los puntos obtenidos por temporada, los goles anotados por temporada, goles recibidos por temporada, victorias y derrotas en la temporada y el gasto neto con respecto al valor de la plantilla y su relación con los resultados deportivos. Además, se explorarán las tendencias de gasto en fichajes y salarios, comparando los efectos de las inversiones en ambos clubes.

El proyecto está realizado a través del uso de bibliotecas como pandas, matplotlib y seaborn, el uso de funciones, lectura y creación de archivos, bucles while, condicionales if, impresión de textos y gráficas, listas, matrices y tuplas; con todo lo anterior se presentarán visualizaciones y modelos que permitan una comprensión más detallada del impacto financiero en el rendimiento de estos equipos.

II. MARCO TEÓRICO:

A continuación, para lograr entender este proyecto de análisis de datos de forma plena, es importante definir algunos conceptos claves que guiarán nuestro estudio:

- **Inyección de capital:** Es la introducción de fondos adicionales en un club por parte de inversores externos. Estos fondos generalmente suelen destinarse a la contratación de jugadores, mejoras de infraestructura, salarios y otros aspectos más.
- **Rendimiento deportivo:** En este estudio, el rendimiento deportivo se mide principalmente a través de la posición en la tabla de clasificación de la Premier League, puntos obtenidos por temporada, victorias, derrotas, goles anotados y goles recibidos.
- **Rendimiento financiero:** Es el análisis de los ingresos y gastos de los clubes, especialmente en términos de gasto neto por temporada y valor de la plantilla.
- **Gasto neto:** Es el excedente positivo o negativo obtenido por un club en una temporada, derivado de la venta de jugadores, entradas vendidas o contratación de nuevos jugadores.
- **Biblioteca de programación:** Es una colección de código desarrollado previamente que los programadores pueden utilizar para desarrollar software de manera más ágil. Este trabajo utilizará herramientas de análisis de datos, principalmente bibliotecas que permiten procesar grandes volúmenes de información y visualizarlos de manera clara.
 - **Pandas:** Pandas es una librería de Python especializada en la manipulación y el análisis de datos. Ofrece estructuras de datos y operaciones para manipular tablas numéricas y series temporales. En el presente trabajo se renombrará pandas como pd
 - **Matplotlib:** Es una biblioteca para la generación de gráficos en dos dimensiones, a partir de datos contenidos en listas o arrays en el lenguaje de programación Python. En el presente trabajo matplotlib.pyplot será renombrada como plt.

- **Seabor:** Es una biblioteca para crear gráficos estadísticos en Python. Está basada en Matplotlib, y se integra con las estructuras de Pandas.
- **Análisis descriptivo:** Consiste en el uso de estadísticas y gráficos para resumir los datos. Será utilizado para observar cómo han cambiado las posiciones de liga, las victorias, los goles marcados y otros indicadores luego la inyección de capital en los clubes.
- **Limpieza de datos:** Se asegurará que los datos sean completos y estén formateados correctamente, esto implica eliminar datos duplicados, tratar valores faltantes y estructurar la información de manera que sea fácil de analizar, será realizado con pandas.
- **Análisis de resultados:** Usando matplotlib y seaborn, se generarán gráficos que muestren la evolución de las posiciones en la liga, los puntos obtenidos por temporada y el gasto neto.

III. OBJETIVOS DEL TRABAJO:

1. **Evaluar el impacto de la inyección de capital en el rendimiento deportivo:** Se busca analizar si los equipos que reciben mayores inversiones logran mejorar su posición en la tabla y si existe una correlación directa entre el gasto en fichajes y los puntos obtenidos.
2. **Comparar el rendimiento deportivo del Newcastle United y el Everton:** Se busca encontrar el contraste entre las inversiones realizadas por ambos equipos y cuales fueron los resultados obtenidos por cada uno en un periodo de siete temporadas.
3. **Desarrollar un programa en Python para analizar datos de clubes de fútbol (desde archivos (CSV):** Se busca formar parte de la escena con un programa que pueda leer, procesar una serie de archivos .csv y luego retornarlos en gráficos, estadísticas y reportes.

IV. DESARROLLO DEL PROGRAMA:

```
#Librerías
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

En estas primeras líneas lo que hacemos es importar las librerías que serán usadas posteriormente para procesar los datos.

```
#Funciones
def victorias_en_temporada(Temporadas_str,Victorias_Newcastle_Total,Victorias_Everton_Total):
    plt.figure(figsize=(10, 6))
    plt.plot(Temporadas_str, Victorias_Newcastle_Total, label='Proyección de victorias Newcastle')
    plt.plot(Temporadas_str, Victorias_Everton_Total, label='Proyección de victorias Everton')
    plt.title('Victorias por temporada')
    plt.xlabel('Temporada')
    plt.ylabel('Victorias')
    plt.legend()
    plt.grid()
    plt.show()
```

Definimos nuestra primera función de victorias en temporada con `plt.figure` da la dimensiones de la imagen, `plt.plot` crea la grafica en base a los datos brindados, `plt.title` se encarga de poner el titulo a la gráfica, `plt.xlabel` y `plt.ylabel` se encargan de dar nombre a los ejes X y Y, `plt.legend` crea la leyenda de la gráfica, `plt.grid` crea el cuadriculado de la grafica y `plt.show` se utiliza para mostrar el resultado en una gráfica.

Utilizaremos las mismas funciones para el resto de funciones.

```
def goles_ anotados(Goles_Totales_Newcastle,Goles_Totales_Everton,Temporadas_str):
    plt.figure(figsize=(10, 6))
    plt.plot(Temporadas_str, Goles_Totales_Newcastle, label='Proyección de goles Newcastle')
    plt.plot(Temporadas_str, Goles_Totales_Everton, label='Proyección de goles Everton')
    plt.title('Goles anotados por temporada por equipo')
    plt.xlabel('Temporada')
    plt.ylabel('Goles anotados')
    plt.legend()
    plt.grid()
    plt.show()

def goles_recibidos(Goles_Recibidos_Newcastle,Goles_Recibidos_Everton,Temporadas_str):
    plt.figure(figsize=(10, 6))
    plt.plot(Temporadas_str, Goles_Recibidos_Totales_Newcastle, label='Proyección de goles recibidos Newcastle')
    plt.plot(Temporadas_str, Goles_Recibidos_Totales_Everton, label='Proyección de goles recibidos Everton')
    plt.title('Goles recibidos por temporada por equipo')
    plt.xlabel('Temporada')
    plt.ylabel('Goles recibidos')
    plt.legend()
    plt.grid()
    plt.show()

def derrotas_en_temporada(Temporadas_str,Victorias_Newcastle_Total,Victorias_Everton_Total):
    plt.figure(figsize=(10, 6))
    plt.plot(Temporadas_str, Derrotas_Newcastle_Total, label='Proyección de derrotas Newcastle')
    plt.plot(Temporadas_str, Derrotas_Everton_Total, label='Proyección de derrotas Everton')
    plt.title('Derrotas por temporada')
    plt.xlabel('Temporada')
    plt.ylabel('Derrotas')
    plt.legend()
    plt.grid()
    plt.show()
```

```

def gasto_neto_vs_valor_plantilla(Temporadas_str, Gasto_Neto_Newcastle, Valor_Plantillas_Newcastle, Gasto_Neto_Everton, Valor_Plantillas_Everton):

    plt.figure(figsize=(10, 6))
    plt.plot(Temporadas_str, Gasto_Neto_Newcastle, label='Gasto Neto Newcastle', marker='o')
    for i, valor in enumerate(Gasto_Neto_Newcastle):
        plt.annotate(f'{valor}', (Temporadas_str[i], Gasto_Neto_Newcastle[i]), textcoords="offset points", xytext=(0,5), ha='center')
    plt.plot(Temporadas_str, Valor_Plantillas_Newcastle, label='Valor de Plantilla Newcastle', marker='o')
    for i, valor in enumerate(Valor_Plantillas_Newcastle):
        plt.annotate(f'{valor}', (Temporadas_str[i], Valor_Plantillas_Newcastle[i]), textcoords="offset points", xytext=(0,5), ha='center')

    plt.title('Gasto contra valor de plantillas Newcastle')
    plt.xlabel('Temporada')
    plt.ylabel('Gasto Neto / Valor de Plantilla')
    plt.legend()
    plt.grid()

    plt.figure(figsize=(10, 6))

    plt.plot(Temporadas_str, Gasto_Neto_Everton, label='Gasto Neto Everton', marker='o')

    for i, valor in enumerate(Gasto_Neto_Everton):
        plt.annotate(f'{valor}', (Temporadas_str[i], Gasto_Neto_Everton[i]), textcoords="offset points", xytext=(0,5), ha='center')

    plt.plot(Temporadas_str, Valor_Plantillas_Everton, label='Valor de Plantilla Everton', marker='o')

    for i, valor in enumerate(Valor_Plantillas_Everton):
        plt.annotate(f'{valor}', (Temporadas_str[i], Valor_Plantillas_Everton[i]), textcoords="offset points", xytext=(0,5), ha='center')

    plt.title('Gasto contra valor de plantillas Everton')
    plt.xlabel('Temporada')
    plt.ylabel('Gasto Neto / Valor de Plantilla')
    plt.legend()
    plt.grid()

    plt.show()

def posiciones_vs_temporada(Temporadas_str, Posicion_Newcastle, Posicion_Everton):
    plt.figure(figsize=(10, 6))
    plt.plot(Temporadas_str, Posicion_Everton, label='Posiciones Everton')
    plt.plot(Temporadas_str, Posicion_Newcastle, label='Posiciones Newcastle')
    plt.title('Posición al final de temporada')
    plt.xlabel('Temporada')
    plt.ylabel('Posición')
    plt.legend()
    plt.grid()
    plt.gca().invert_yaxis()
    plt.show()

def puntos_vs_temporada(Temporadas_str, Puntos_Newcastle, Puntos_Everton):
    plt.figure(figsize=(10, 6))
    plt.plot(Temporadas_str, Puntos_Everton, label='Puntos Everton')
    plt.plot(Temporadas_str, Puntos_Newcastle, label='Puntos Newcastle')
    plt.title('Puntos al final de temporada')
    plt.xlabel('Temporada')
    plt.ylabel('Puntos')
    plt.legend()
    plt.grid()
    plt.show()

def correlaciones(Data_Frame_Newcastle_Transfermarkt, Data_Frame_Everton_Transfermarkt):
    plt.figure(figsize=(8,2))
    sns.heatmap(Data_Frame_Newcastle_Transfermarkt.corr(),annot=True, linewidth=0.5, cmap = 'magma')
    plt.title('Matriz de correlaciones de Newcastle')
    plt.figure(figsize=(8,2))
    sns.heatmap(Data_Frame_Everton_Transfermarkt.corr(),annot=True, linewidth=0.5, cmap = 'magma')
    plt.title('Matriz de correlaciones de Everton')

```

En esta última función notamos a presencia de una nueva función `sns.heatmap` la cual se encarga de darle un color a la gráfica desde más tenue a más oscura según la correlación encontrada.


```
#Preparación para el data analysis (se suben los documentos para que se cree un pandas series)
Data_Frame_Newcastle_Transfermarkt = pd.read_csv('/Newcastle_Transfermarkt (1).csv')
Data_Frame_Everton_Transfermarkt = pd.read_csv('/Everton_Transfermarkt (1).csv')
Data_Frame_2017_2018 = pd.read_csv('/2017-18.csv')
Data_Frame_2018_2019 = pd.read_csv('/2018-19.csv')
Data_Frame_2019_2020 = pd.read_csv('/2019-20.csv')
Data_Frame_2020_2021 = pd.read_csv('/2020-21.csv')
Data_Frame_2021_2022 = pd.read_csv('/2021-22.csv')
Data_Frame_2022_2023 = pd.read_csv('/2022-23.csv')
Data_Frame_2023_2024 = pd.read_csv('/2023-24.csv')
```

Con `pd.read_csv` hacemos que se lean los archivos subidos y se almacenen en un variable `Data_Frame` como una tabla.

```
#Nos deshacemos de los tipos de datos "object", ya que Python no los entiende. Nótese que esto sólo se hace para los de las temporadas
Data_Frame_2017_2018_limpio=Data_Frame_2017_2018.select_dtypes(include=object).columns.tolist()
Data_Frame_2017_2018[Data_Frame_2017_2018_limpio] = Data_Frame_2017_2018[Data_Frame_2017_2018_limpio].astype(str)
Data_Frame_2018_2019_limpio=Data_Frame_2018_2019.select_dtypes(include=object).columns.tolist()
Data_Frame_2018_2019[Data_Frame_2018_2019_limpio] = Data_Frame_2018_2019[Data_Frame_2018_2019_limpio].astype(str)
Data_Frame_2019_2020_limpio=Data_Frame_2019_2020.select_dtypes(include=object).columns.tolist()
Data_Frame_2019_2020[Data_Frame_2019_2020_limpio] = Data_Frame_2019_2020[Data_Frame_2019_2020_limpio].astype(str)
Data_Frame_2020_2021_limpio=Data_Frame_2020_2021.select_dtypes(include=object).columns.tolist()
Data_Frame_2020_2021[Data_Frame_2020_2021_limpio] = Data_Frame_2020_2021[Data_Frame_2020_2021_limpio].astype(str)
Data_Frame_2021_2022_limpio=Data_Frame_2021_2022.select_dtypes(include=object).columns.tolist()
Data_Frame_2021_2022[Data_Frame_2021_2022_limpio] = Data_Frame_2021_2022[Data_Frame_2021_2022_limpio].astype(str)
Data_Frame_2022_2023_limpio=Data_Frame_2022_2023.select_dtypes(include=object).columns.tolist()
Data_Frame_2022_2023[Data_Frame_2022_2023_limpio] = Data_Frame_2022_2023[Data_Frame_2022_2023_limpio].astype(str)
Data_Frame_2023_2024_limpio=Data_Frame_2023_2024.select_dtypes(include=object).columns.tolist()
Data_Frame_2023_2024[Data_Frame_2023_2024_limpio] = Data_Frame_2023_2024[Data_Frame_2023_2024_limpio].astype(str)
```

En este apartado nos deshacemos de los datos “object”, ya que, este tipo de datos no son reconocibles por Python, generan confusión dentro del código.

```
Datos Newcastle y Everton por temporada (separación de datos)
"""
#2017-18
Newcastle_Local_2017_2018 = Data_Frame_2017_2018[Data_Frame_2017_2018['HomeTeam']=='Newcastle']
Newcastle_Visita_2017_2018 = Data_Frame_2017_2018[Data_Frame_2017_2018['AwayTeam']=='Newcastle']
Everton_Local_2017_2018 = Data_Frame_2017_2018[Data_Frame_2017_2018['HomeTeam']=='Everton']
Everton_Visita_2017_2018 = Data_Frame_2017_2018[Data_Frame_2017_2018['AwayTeam']=='Everton']
#2018-19
Newcastle_Local_2018_2019 = Data_Frame_2018_2019[Data_Frame_2018_2019['HomeTeam']=='Newcastle']
Newcastle_Visita_2018_2019 = Data_Frame_2018_2019[Data_Frame_2018_2019['AwayTeam']=='Newcastle']
Everton_Local_2018_2019 = Data_Frame_2018_2019[Data_Frame_2018_2019['HomeTeam']=='Everton']
Everton_Visita_2018_2019 = Data_Frame_2018_2019[Data_Frame_2018_2019['AwayTeam']=='Everton']
#2019-20
Newcastle_Local_2019_2020 = Data_Frame_2019_2020[Data_Frame_2019_2020['HomeTeam']=='Newcastle']
Newcastle_Visita_2019_2020 = Data_Frame_2019_2020[Data_Frame_2019_2020['AwayTeam']=='Newcastle']
Everton_Local_2019_2020 = Data_Frame_2019_2020[Data_Frame_2019_2020['HomeTeam']=='Everton']
Everton_Visita_2019_2020 = Data_Frame_2019_2020[Data_Frame_2019_2020['AwayTeam']=='Everton']
#2020-21
Newcastle_Local_2020_2021 = Data_Frame_2020_2021[Data_Frame_2020_2021['HomeTeam']=='Newcastle']
Newcastle_Visita_2020_2021 = Data_Frame_2020_2021[Data_Frame_2020_2021['AwayTeam']=='Newcastle']
Everton_Local_2020_2021 = Data_Frame_2020_2021[Data_Frame_2020_2021['HomeTeam']=='Everton']
Everton_Visita_2020_2021 = Data_Frame_2020_2021[Data_Frame_2020_2021['AwayTeam']=='Everton']
#2021-22
Newcastle_Local_2021_2022 = Data_Frame_2021_2022[Data_Frame_2021_2022['HomeTeam']=='Newcastle']
Newcastle_Visita_2021_2022 = Data_Frame_2021_2022[Data_Frame_2021_2022['AwayTeam']=='Newcastle']
Everton_Local_2021_2022 = Data_Frame_2021_2022[Data_Frame_2021_2022['HomeTeam']=='Everton']
Everton_Visita_2021_2022 = Data_Frame_2021_2022[Data_Frame_2021_2022['AwayTeam']=='Everton']
#2022-23
Newcastle_Local_2022_2023 = Data_Frame_2022_2023[Data_Frame_2022_2023['HomeTeam']=='Newcastle']
Newcastle_Visita_2022_2023 = Data_Frame_2022_2023[Data_Frame_2022_2023['AwayTeam']=='Newcastle']
Everton_Local_2022_2023 = Data_Frame_2022_2023[Data_Frame_2022_2023['HomeTeam']=='Everton']
Everton_Visita_2022_2023 = Data_Frame_2022_2023[Data_Frame_2022_2023['AwayTeam']=='Everton']
#2023-24
Newcastle_Local_2023_2024 = Data_Frame_2022_2023[Data_Frame_2022_2023['HomeTeam']=='Newcastle']
Newcastle_Visita_2023_2024 = Data_Frame_2022_2023[Data_Frame_2022_2023['AwayTeam']=='Newcastle']
Everton_Local_2023_2024 = Data_Frame_2022_2023[Data_Frame_2022_2023['HomeTeam']=='Everton']
Everton_Visita_2023_2024 = Data_Frame_2022_2023[Data_Frame_2022_2023['AwayTeam']=='Everton']
```

Lo que hacemos en esta parte del código es filtrarlo de formas más específica en pequeños pandas series, o parecidos a un tabla.


```

Datos Transfersmarkt por equipo
"""

#Newcastle
Valor_Plantillas_Newcastle=Data_Frame_Newcastle_Transfermarkt['Valor Plantilla']
Gasto_Neto_Newcastle=Data_Frame_Newcastle_Transfermarkt['Gasto Neto']
Posicion_Newcastle=Data_Frame_Newcastle_Transfermarkt['Standings']
Puntos_Newcastle=Data_Frame_Newcastle_Transfermarkt['Puntos']
#Everton
Valor_Plantillas_Everton=Data_Frame_Everton_Transfermarkt['Valor Plantilla']
Posicion_Everton=Data_Frame_Everton_Transfermarkt['Standings']
Gasto_Neto_Everton=Data_Frame_Everton_Transfermarkt['Gasto Neto']
Puntos_Everton=Data_Frame_Everton_Transfermarkt['Puntos']
"""

Temporadas que se analizarán
"""

temporadas=[["2017-18"],["2018-19"],["2019-20"],["2020-21"],["2021-22"],["2022-23"],["2023-24"]]
"""

Extracción de datos para análisis deportivo
"""

#2017-16
#Newcastle
Resultados_Local=list(Newcastle_Local_2017_2018['FTR'])
Resultados_Visita=list(Newcastle_Visita_2017_2018['FTR'])
Victorias_Local=Resultados_Local.count('H')
Victorias_Visita=Resultados_Visita.count('A')
Victorias2017_Newcastle=Victorias_Local+Victorias_Visita
Derrotas_Local=Resultados_Local.count('A')
Derrotas_Visita=Resultados_Visita.count('H')
Derrotas2017_Newcastle=Derrotas_Local+Derrotas_Visita
Empates_Local=Resultados_Local.count('D')
Empates_Visita=Resultados_Visita.count('D')
Empates2017_Newcastle=Empates_Local+Empates_Visita
Goles_Local_Newcastle_2017=list(Newcastle_Local_2017_2018['FTGH'])
Goles_Visita_Newcastle_2017=list(Newcastle_Visita_2017_2018['FTAG'])
Goles_Newcastle_2017=sum(Goles_Local_Newcastle_2017)+sum(Goles_Visita_Newcastle_2017)
Goles_Recibidos_Local_Newcastle_2017=list(Newcastle_Local_2017_2018['FTAG'])
Goles_Recibidos_Visita_Newcastle_2017=list(Newcastle_Visita_2017_2018['FTGH'])
Goles_Recibidos_Newcastle_2017=sum(Goles_Recibidos_Local_Newcastle_2017)+sum(Goles_Recibidos_Visita_Newcastle_2017)

#Everton
Resultados_Local=list(Everton_Local_2017_2018['FTR'])
Resultados_Visita=list(Everton_Visita_2017_2018['FTR'])
Victorias_Local=Resultados_Local.count('H')
Victorias_Visita=Resultados_Visita.count('A')
Victorias2017_Everton=Victorias_Local+Victorias_Visita
Derrotas_Local=Resultados_Local.count('A')
Derrotas_Visita=Resultados_Visita.count('H')
Derrotas2017_Everton=Derrotas_Local+Derrotas_Visita
Empates_Local=Resultados_Local.count('D')
Empates_Visita=Resultados_Visita.count('D')
Empates2017_Everton=(Empates_Local+Empates_Visita)
Goles_Local_Everton_2017=list(Everton_Local_2017_2018['FTGH'])
Goles_Visita_Everton_2017=list(Everton_Visita_2017_2018['FTAG'])
Goles_Everton_2017=sum(Goles_Local_Everton_2017)+sum(Goles_Visita_Everton_2017)
Goles_Recibidos_Local_Everton_2017 = list(Everton_Local_2017_2018['FTAG'])
Goles_Recibidos_Visita_Everton_2017 = list(Everton_Visita_2017_2018['FTGH'])
Goles_Recibidos_Everton_2017 = sum(Goles_Recibidos_Local_Everton_2017) + sum(Goles_Recibidos_Visita_Everton_2017)

```

En este apartado filtramos los datos por columnas de datos y repetimos el proceso para todas las temporadas.

```

Programa de análisis de datos
"""

usuario = str(input("Introduce el usuario: "))
with open("texto.txt", "w") as archivo:
    archivo.write(f"Usuario: {usuario}\nHistorial de búsqueda: \n")
while True:
    print("¿Qué acción deseas realizar?")
    print("1: Visualizar Gráficas de rendimiento por partido")
    print("2: Visualizar Gráficas económicas")
    print("3: Visualizar Gráficas de rendimiento por temporada")
    print("4: Desas salir?")
    accion=int(input("Introduce tu acción: "))
    if accion==1:
        print("Visualizar Gráficas de rendimiento por partido")
        with open("texto.txt", "a") as archivo:
            archivo.write("Accedio: Graficas de rendimiento deportivo\n") #Escribiendo el archivo
        archivo.close()#Liberamos el archivo

```

```

while True:
    print("1: Visualizar Gráfica de Victorias")
    print("2: Visualizar Gráfica de Goles Anotados")
    print("3: Visualizar Gráfica de Goles Recibidos")
    print("4: Visualizar Gráfica de Derrotas")
    print("5: Salir de Gráficas de rendimientos por partido")
    rendimiento=int(input("¿Qué deseas visualizar? "))
    if rendimiento==1:
        temporadas_str = [t[0] for t in temporadas]
        victorias_en_temporada(temporadas_str,Victorias_Newcastle_Total,Victorias_Everton_Total)
        print("Se observa la gráfica de victorias por equipo.")
        with open("texto.txt", "a") as archivo:
            archivo.write("Accedio: Grafica de Victorias\n") #Escribiendo el archivo
            archivo.close()#Liberamos el archivo
        with open("texto.txt", "a") as archivo:
            archivo.write("Accedio: Grafica de Victorias\n") #Escribiendo el archivo
            archivo.close()#Liberamos el archivo
    elif rendimiento==2:
        temporadas_str = [t[0] for t in temporadas]
        goles_anotados(Goles_Totales_Newcastle,Goles_Totales_Everton,temporadas_str)
        print("Se observa la gráfica de goles anotados.")
        with open("texto.txt", "a") as archivo:
            archivo.write("Accedio: Grafica de Goles Anotados\n") #Escribiendo el archivo
            archivo.close()#Liberamos el archivo
    elif rendimiento==3:
        temporadas_str = [t[0] for t in temporadas]
        goles_recibidos(Goles_Recibidos_Totales_Newcastle,Goles_Recibidos_Totales_Everton,temporadas_str)
        print("Se visualiza la grafica de goles recibidos.")
        with open("texto.txt", "a") as archivo:
            archivo.write("Accedio: Grafica de Goles Recibidos\n") #Escribiendo el archivo
            archivo.close()#Liberamos el archivo
    elif rendimiento==4:
        temporadas_str = [t[0] for t in temporadas]
        derrotas_en_temporada(temporadas_str,Victorias_Newcastle_Total,Victorias_Everton_Total)
        print("Se visualiza la grafica de derrotas.")
        with open("texto.txt", "a") as archivo:
            archivo.write("Accedio: Grafica de Derrotas\n") #Escribiendo el archivo
            archivo.close()#Liberamos el archivo
    elif rendimiento==5:
        print("Saliendo de Gráficas de Rendimiento por partido")
        with open("texto.txt", "a") as archivo:
            archivo.write("Salio: Graficas de Rendimiento por partido\n") #Escribiendo el archivo
            archivo.close()#Liberamos el archivo
        break
    else:
        print("Esa opción no se encuentra")
        break
    elif accion==2:
        print("Gráficas Económicas")
        with open("texto.txt", "a") as archivo:
            archivo.write("Accedio: Graficas Economicas\n")
            archivo.close()
        while True:
            print("1: Gráfica de Gasto Neto vs valor de la plantilla")
            print("2: Salir de Gráficas Económicas")
            economia=int(input("¿Qué deseas Visualizar? "))
            if economia==1:
                temporadas_str = [t[0] for t in temporadas]
                gasto_neto_vs_valor_plantilla(temporadas_str,Gasto_Neto_Newcastle,Valor_Plantillas_Newcastle,Gasto_Neto_Everton,Valor_Plantillas_Everton)
                print("Se visualiza la Gráfica de Gasto Neto vs Valor de la plantilla")
                with open("texto.txt", "a") as archivo:
                    archivo.write("Accedio: Grafica de Gasto Neto vs Valor de la Plantilla \n")
                    archivo.close()
            elif economia==2:
                print("Saliendo de Gráficas Económicas")
                with open("texto.txt", "a") as archivo:
                    archivo.write("Salio: Graficas Economicas \n")
                    archivo.close()
                break
            else:
                print("Esa opción no se encuentra")
                break

```

```

elif accion==3:
    print("Gráficas de rendimiento por temporada")
    with open("texto.txt", "a") as archivo:
        archivo.write("Accedio: Visualizar Gráficas de rendimiento por temporada \n")
        archivo.close()

while True:
    print("1: Gráfica de posiciones vs temporada ")
    print("2: Grafica de puntos vs temporada")
    print("3: Graficas de correlación")
    print("4: Salir de Gráficas de rendimiento por temporada \n")
    eleccion=int(input("¿Qué deseas Visualizar? "))
    if eleccion==1:
        temporadas_str = [t[0] for t in temporadas]
        posiciones_vs_temporada(temporadas_str, Posicion_Newcastle, Posicion_Everton)
        print("Se visualiza la grafica de posiciones versus temporadas.")
        with open("texto.txt", "a") as archivo:
            archivo.write("Accedio: Posiciones por temporada \n")
            archivo.close()
    elif eleccion==2:
        temporadas_str = [t[0] for t in temporadas]
        puntos_vs_temporada(temporadas_str, Puntos_Newcastle, Puntos_Everton)
        print("Se visualiza la grafica de puntos versus temporadas.")
        with open("texto.txt", "a") as archivo:
            archivo.write("Accedio: Puntos por temporada \n")
            archivo.close()
    elif eleccion==3:
        temporadas_str = [t[0] for t in temporadas]
        correlaciones(Data_Frame_Newcastle_Transfermarkt, Data_Frame_Everton_Transfermarkt)
        print("Se visualiza la grafica de correlación.")
        with open("texto.txt", "a") as archivo:
            archivo.write("Accedio: Correlación entre variables \n")
            archivo.close()
    elif eleccion == 4:
        with open("texto.txt", "a") as archivo:
            print("Saliendo de Gráficas de rendimiento por temporada")
            archivo.write("Salió de Gráficas de rendimiento por temporada \n")
            archivo.close()
        break
    else:
        print("Esa opción no se encuentra")
        break
elif accion==4:
    print("Adiós")
    with open("texto.txt", "a") as archivo:
        archivo.write("Salio\n")
        archivo.close()
    break
else:
    print("No se encuentra esa opción")
    break

```

La creación del menú se hizo a través de bucles while true y condicionales if con una respuesta numérica int en un rango de opciones según lo elegido.

V. MÉTODOS DE IMPLEMENTACIÓN:

- **Fase de Recolección de Datos:**

Fuentes de datos: Los datos para este proyecto se recopilaron de diversas fuentes, como Transfermarkt, y Football Data (para resultados de la liga y tablas).

- **Limpieza y Preparación de Datos:**

Estandarización: Como los valores abarcan varias temporadas, deben ajustarse a términos reales y filtrados por temporada para asegurar un análisis consistente y lineal.

- **Análisis Descriptivo:**

Visualización de tendencias: Se elaborarán gráficos de líneas mostrando la transformación con respecto al rendimiento deportivo del Newcastle y Everton en relación con sus posiciones en la tabla y los puntos obtenidos.

- **Correlación y Causalidad:**

- Correlación: Se analizará la correlación entre la inversión y el rendimiento en la liga.

- **Análisis Comparativo:**

Benchmarking: Se comparará la evolución de Newcastle y Everton.

- **Visualización de Datos:**

Gráficas interactivas: Se utilizarán herramientas como Matplotlib o Seaborn en Python para crear visualizaciones interactivas que muestren la relación entre inversión y rendimiento.

VI. ANÁLISIS DE RESULTADOS:

El análisis de los resultados obtenidos mediante las gráficas, reportes y estadísticas nos muestra un cambio significativo en el rendimiento del Newcastle United, que coincide con la llegada de capital externo al club. Por otro lado, el Everton, a pesar de su tradición como equipo competitivo, ha sufrido un constante declive en su posición dentro de la liga. Por un lado, el Newcastle adoptó nuevas políticas y se hizo de nuevos staff y nuevos jugadores para poder aumentar la competitividad de su plantilla, pasando de ser un equipo ubicado generalmente en tres cuartos de tabla a uno que se ubica dentro de los 7 mejores, tuvo un excedente negativo de 519 millones de euros en estas 7 temporadas, el valor de su plantilla subió un 315% con respecto a la fijada al inicio, se notó una considerable mejora con respecto a los goles anotados por temporada que ascendieron de 39 en su primera temporada a 62 en la última, los goles recibidos en la última temporada fue de 85 y las victorias obtenidas que en su primera temporada fue de 12 y en la última llegó a 18, por otro lado, tenemos al Everton el cual a medida que fue adoptando una política de formación de jóvenes jugadores y ha reducido su inversión en fichajes, ha logrado obtener un excedente negativo en su gran mayoría por la venta de jugadores de 181 millones de euros en estas 7 temporadas, el valor de su plantilla actual con respecto a la inicial sufre variaciones durante el tiempo pero en la actualidad solo se encuentra 30 millones por debajo de la inicial, de igual

manera su posición y rendimiento en la liga se vieron afectados de manera considerable, pasando de ser un equipo ubicado dentro de los 10 mejores a pelear hasta las últimas fechas para no descender y en consecuencia de todo ello se obtuvieron más derrotas y más goles fueron recibidos durante las temporadas analizadas.

VII. CONCLUSIÓN INDIVIDUAL:

Luis Enrique Aguilera Novoa:

“Los resultados del código realizado comprueban nuestra hipótesis inicial. La inversión extranjera es un factor clave en el éxito en el fútbol de hoy en día. Nuestro código tiene un menú muy amplio el cual le permite al usuario acceder a demasiados datos, los cuales comprueban nuestra hipótesis. Poder utilizar archivos csv para manipular datos en python, y poder graficarlos, fue bastante útil, para exponer nuestros resultados de manera correcta, formal e interactiva. Sobre el código se puede concluir que, el uso de librerías hace que la creación del código sea más eficiente y fácil, sin ellas el código sería más extenso y limitado.”

Pablo Emiliano Gonzales Rios:

“Este proyecto fue especialmente interesante porque me permitió desarrollar aún más mis habilidades del trabajo con datos: la limpieza y la preparación. Al trabajar con archivos CSV y aplicar técnicas de procesamiento de datos, me di cuenta de lo importante que es que los datos estén ordenados y sean consistentes en su formato para garantizar resultados. El uso de ayudas visuales como gráficas, me ayudó a que los datos fueran más comprensibles para mí.”

Angelo Dayvis Farfan Torres:

“Del trabajo realizado puedo concluir que, en efecto, la inyección de capital en un equipo profesional, esta directamente relacionado con el rendimiento deportivo que este enfrente a lo largo de la temporada, esto se puede demostrar al contrastar en las gráficas que el punto de inflexión llega en la temporada 2020/21 en donde el Everton dejó de invertir y su rendimiento bajo, por otra parte, el Newcastle invirtió mucho más, logrando así, una de sus mejores temporadas. Por último, con este proyecto pusimos a prueba los conocimientos adquiridos durante estas 10 semanas y lo utilizamos para aportar a la escena de programadores con un software el cual analiza los datos y crea gráficas, reportes y estadísticas a partir de ello; dejando de base un programa que esperamos que a futuro sea mejorado.”

VIII. CONCLUSIÓN POR EQUIPO:

Podemos concluir que, la inyección de capital tiene un impacto directo en el rendimiento deportivo de un equipo, los gastos en fichaje guardan cierta relación con los puntos obtenidos en la liga y los goles marcados, pues al Newcastle obtener nuevos jugadores y mejor preparados para competir en el fútbol de elite, se visualizó un incremento del valor de plantilla y en consecuencia un ascenso casi meteorico con respecto a los puntos obtenidos, partidos ganados, goles anotados y posición obtenida en la liga; pero no es lo único que se identificó, también existen factores, como la gestión deportiva que son importantes, en el caso del Everton al cambiar radicalmente su política de fichajes por una más conservadora y centrada en la formación de jugadores jóvenes, solamente logro que su valor de plantilla baje, al igual que su sus goles anotados y las demás estadísticas, es por ello que, en los casos estudiados se logró identificar el patrón que mientras más invierta un equipo para reforzar su plantilla, obtendrá mejores resultado en la temporada en términos de rendimiento deportivo.

Este análisis de datos no solo contribuye a la comprensión del impacto de la inyección de capital extranjero en el fútbol y su efecto en el rendimiento deportivo, sino que también proporcionará una base para futuros estudios en el área del análisis deportivo. Mediante la aplicación de librerías que facilitan el proceso de análisis de datos en Python, este trabajo ofrece una visión detallada del efecto de las inversiones en el rendimiento de equipos como el Newcastle United y el Everton, permitiendo a los lectores extraer conclusiones fundamentadas sobre la relación entre el dinero y el éxito en el fútbol moderno.