Práctica 1

1. Contexto. Explicar en qué contexto se ha recolectado la información. Explique por qué el sitio web elegido proporciona dicha información.

Este trabajo ha sido realizado como parte de la Práctica 1 de la asignatura Tipología y ciclo de vida de los datos, enmarcada en el Máster de Ciencia de Datos de la UOC.

El proyecto consiste en recolectar, en un repositorio de Github, un dataset que haya sido obtenido a través de la técnica de *web scrapping*. Para ello se han tenido en cuenta las fuentes de información más abajo citadas. El resultado de este proceso, tanto el código Python como el CSV que se obtiene, se incluyen en el repositorio NBA_Statistics.

Por defecto, los datos se refieren a la temporada 2020/2021 a través de datos medios por partido. Sin embargo, a través del código, se puede modificar el tipo de datos que se obtiene y la temporada a la que hacen referencia los indicadores.

La importancia que reside en este dataset está en el análisis del rendimiento de los diferentes equipos de la NBA a través de la evolución durante cada temporada de sus estadísticas principales.

Cabe mencionar que la web utilizada para recopilar los datos específica en su archivo robots.txt que no permite el uso de técnicas de web scraping. Sin embargo, puesto que el objetivo de esta práctica es meramente académico y que la estructura HTML de la web permite seleccionar la temporada y el tipo de datos (algo que no permite la web oficial de la NBA), se ha decidido seguir adelante con el web scraping de la misma.

2. Definir un título para el dataset. Elegir un título que sea descriptivo.

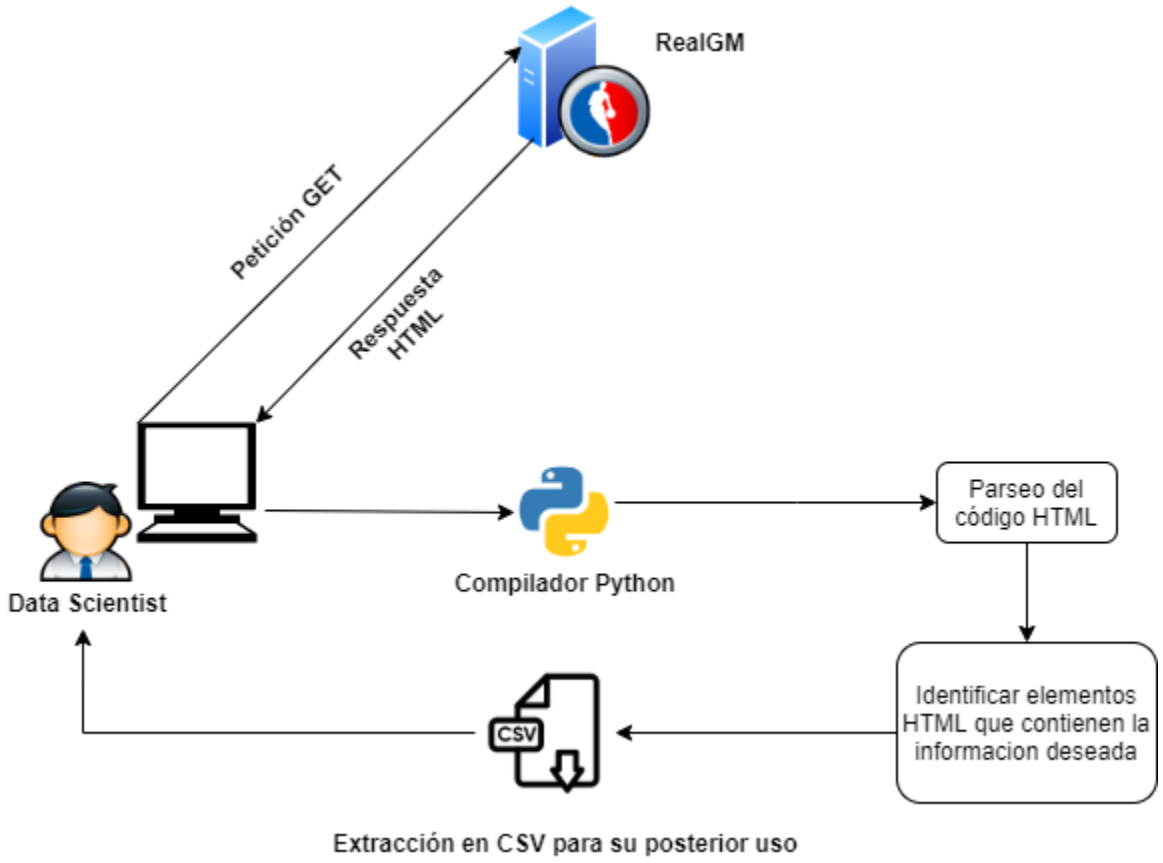
El título elegido es nbateams_stats_20-21.csv

3. Descripción del dataset. Desarrollar una descripción breve del conjunto de datos que se ha extraído (es necesario que esta descripción tenga sentido con el título elegido).

El *dataset* al que se refiere este proyecto ha sido obtenido de la web del proveedor de servicios de la NBA RealGM.com. En particular, se ha obtenido un archivo CSV que incluye diferentes estadísticas de juego de los diferentes equipos de la NBA para la temporada 2020/2021 expresadas como media por partido, tales como Triples Encestados o Puntos por Partido.

Los datos obtenidos incluyen un total de 20 indicadores de tipo numérico para los 30 equipos que disputan la NBA. Por esa razón obtenemos un dataframe con 21 columnas, teniendo en cuenta la columna que hace referencia a los nombres de los equipos (tipo string). A través de un estudio más en profundidad, como los llevados a cabo en los siguientes apartados, se puede discriminar entre equipos mejores y peores, más defensivos y ofensivos, aquellos que más convierten sus tiros y los que más fallos tienen.

4. Representación gráfica. Presentar esquema o diagrama que identifique el dataset visualmente y el proyecto elegido.



5. Contenido. Explicar los campos que incluye el dataset, el periodo de tiempo de los datos y cómo se ha recogido.

Se incluyen las siguientes estadísticas para cada equipo de la NBA (media por partido):

- GP: Games Played - Partidos Jugados
- MPG: Minutes Per Game - Minutos por Partido
- FGM: Field Goals Made - Tiros de Campo Encestados
- FGA: Field Goals Attempts - Tiros de Campo Intentados
- FG%: Field Goals Percentage - Porcentaje de Tiros de Campo Encestados
- 3PM: Three-Points Field Goals Made - Triples Encestados

- 3PA: Three-Points Field Goals Attempts - Triples Intentados
- 3P%: Three-Points Field Goals Percentage - Porcentaje de Triples Encestados
- FTM: Free Throws Made - Tiros Libres Encestados
- FTA: Free Throws Attempts - Tiros Libres Intentados
- FT%: Free Throws Percentage - Porcentaje de Tiros Libres Encestados
- TOV: TurnOvers - Pérdidas de Balón
- PF: Personal Faults - Faltas Personales
- ORB: Offensive Rebounds - Rebotes en Ataque
- DRB: Defensive Rebounds - Rebotes en Defensa
- RPG: Total Rebounds Per Game - Rebotes Totales
- APG: Assist Per Game - Asistencias
- SPG: Steal Per Game - Robos
- BPG: Block Per Game - Bloqueos
- PPG: Points Per Game - Puntos

Los datos obtenidos corresponden a la temporada 2020-2021 aunque el usuario puede introducir la fecha que desea consultar durante la ejecución del programa. La manera en la que se han recogido los datos ha sido mediante la técnica de web scrapping. Para ello, se ha generado un script en python que accede a la web que contiene los datos. Este script, itera sobre la tabla que contiene los datos y genera un dataframe que los incluye. Posteriormente, se exporta dicho dataframe en formato csv.

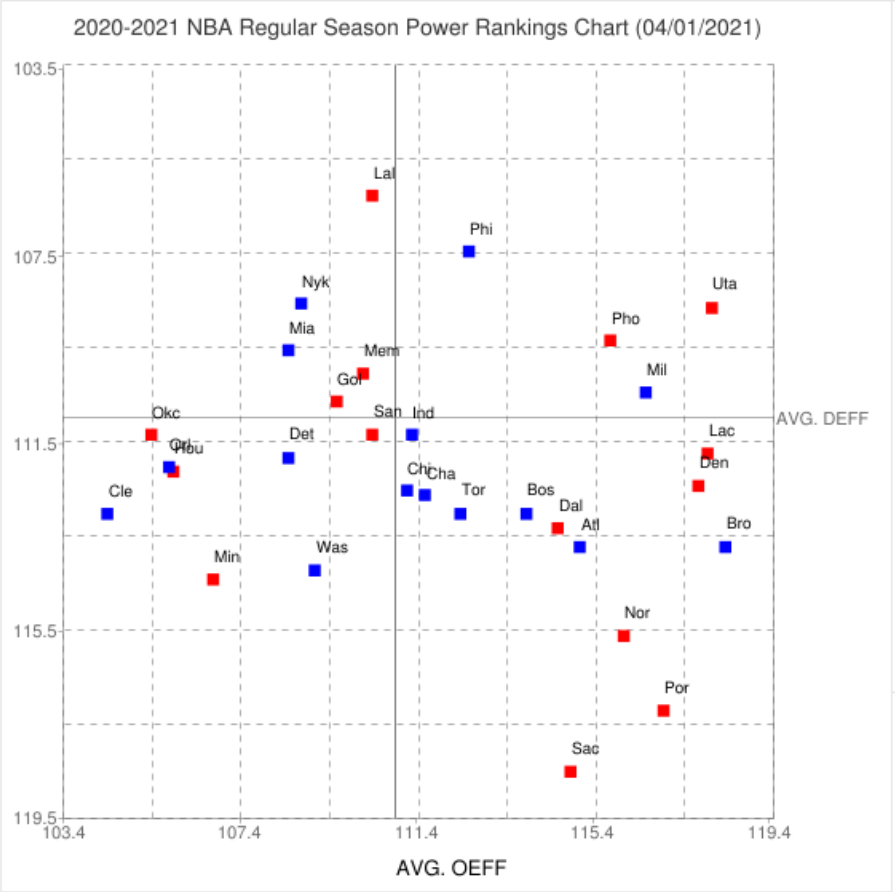
La recolección de los datos se realiza a través del programa con el código Python incluido en el repositorio de Github (https://github.com/emgestoso/NBA_Statistics). En él, principalmente a través del paquete BeautifulSoup, se puede analizar el contenido HTML de la página y extraer los datos de la tabla indexando cada uno de sus elementos con un bucle for. Una vez obtenido el dataframe, se escribe en un documento CSV que es el resultado de la práctica e igualmente se incluye en el repositorio de Github.

6. Agradecimientos. Presentar al propietario del conjunto de datos. Es necesario incluir citas de análisis anteriores o, en caso de no haberlas, justificar esta búsqueda con análisis similares.

El propietario del conjunto de datos es RealGM. Se trata de una empresa de licencias que ofrece software patentado a los equipos de la NBA como proveedor de servicios de aplicaciones.

En su página web no encontramos ningún tipo de analitica basada en datos estadisticos por lo que para esta práctica consideraremos que no se realizan. Sin embargo, es bien sabido que la NBA cuenta con un gran número de analistas que realizan sus estudios basados en datos similares.

En la web 'https://www.nbastuffer.com/2020-2021-nba-power-rankings-chart/' podemos encontrar un cuadrante donde se muestran los ratios tanto ofensivos como defensivos de los equipos.



De esta manera, podemos clasificar los equipos en cuatro zonas:

- Los buenos equipos aparecen en el cuadrante superior derecho.
- Los malos equipos aparecen en el cuadrante inferior izquierdo.
- Los equipos con buena defensa y mala ofensiva aparecen en el cuadrante superior izquierdo.

- Los equipos con buena ofensiva y mala defensa aparecen en el cuadrante inferior derecho.

Otra manera de realizar análisis es mediante el uso de indicadores generados a partir de los hechos. Pongamos como ejemplo la posesión.

La posesión es posiblemente la métrica más importante para comprender los conceptos básicos del baloncesto. Actualmente, la posesión para un equipo se cuenta cada vez que un jugador de dicho equipo realiza una de las siguientes acciones:

1. Intenta un tiro de campo
2. Falla un tiro y no consigue el rebote ofensivo
3. Pierde el balón
4. Va a la área de 2 o 3 puntos y hace el último tiro o no recibe el rebote de un último tiro fallado

Una fórmula básica para el cálculo de la posesión sería la siguiente:

Fórmula posesión básica = 0.96 * [(FGA) + (TOV) + 0.44 * (FTA) - (ORB)]

Debe tenerse en cuenta el multiplicador de 0.44 porque no todos los tiros libres toman posesión. El multiplicador de 0.96 tiene en cuenta los rebotes ofensivos del equipo en situaciones en las que un jugador defensivo lanza un tiro fallado fuera de los límites, continuando la posesión sin que se acredite un rebote ofensivo.

7. Inspiración. Explique por qué es interesante este conjunto de datos y qué preguntas se pretenden responder. Es necesario comparar con los análisis anteriores presentados en el apartado 6.

Este conjunto de datos es de especial interés para los amantes del baloncesto, en particular para los de la NBA. Algunas de las cuestiones que puede resolver el análisis de este dataset son:

- Listado de puntos anotados por conferencia y división de cualquier temporada registrada de la NBA.
- Ranking por eficiencia (TOP 10) de los equipos durante cualquier temporada.
- Listado de equipos de la NBA, ordenados por el número de rebotes ofensivos realizados en la temporada 2019-2020.
- Evolución temporal desde la temporada 1946-1947 hasta la 2020-2021 del total de tiros libres realizados, acertados y fallados y tanto por ciento de acierto por equipos.

Basándonos en el análisis por cuadrantes de ratios ofensivos y defensivos planteado en el apartado 6, podemos afirmar que los cuatro mejores equipos son PHI, PHO, UTA y MIL (en el cuadrante superior derecho) mientras que los tres peores equipos son CLE, MIN y WAS, situados en el cuadrante inferior izquierdo.

8. Licencia. Seleccione una de estas licencias para su dataset y explique el motivo de su selección:

La licencia escogida, debido al problema con el archivo de robots.txt, será la de **Released Under CC BY-NC-SA 4.0 License.** Este tipo de licencia da derecho a terceros a compartir (copiar y distribuir el dataset en cualquier medio o formato) y a adaptar (mezclar, transformar y construir sobre el dataset), siempre que se atribuye la autoría del dataset correctamente, no se use con fines comerciales y, caso de adaptar el dataset, se utilice esta misma licencia.

Esta licencia se justifica atendiendo al hecho de que los datos son de dominio público. Sin embargo, dado que el uso que se les ha dado es meramente académico, no se le puede otorgar licencia con uso comercial.

9. Código. Adjuntar el código con el que se ha generado el dataset, preferiblemente en Python o, alternativamente, en R.

```
# Importamos las librerias necesarias

from bs4 import BeautifulSoup

import requests

import pandas as pd

import os



# Declaramos la variable URL como string del link donde descargamos la información

url = 'https://basketball.realgm.com/nba/team-stats'



# Opción para que el usuario introduzca la temporada que quiere consultar

# season = input("Introduce la temporada que quieres consultar:\n")



# Tipo de estadísticas a consultar, en este caso siempre serán estadísticas medias

# datatype = 'Averages'
```

```
# url2 = 'https://basketball.realgm.com/nba/team_stats/'+ season +'/'
+ datatype +'/Team_Totals/Regular_Season/gp/desc'



# Haciendo uso del método get de la clase requests, descargamos la
pagina web y la almacenamos en la variable page

page = requests.get(url)


# Instanciamos el objeto beautifulsoup

soup = BeautifulSoup(page.content, 'html.parser')


# Declaramos las columnas de nuestro dataframe

columns = ['#', 'Team', 'GP', 'MPG', 'FGM', 'FGA', 'FG%', '3PM',
'3PA', '3P%', 'FTM', 'FTA', 'FT%', 'TOV', 'PF', 'ORB',

          'DRB', 'RPG', 'APG', 'SPG', 'BPG', 'PPG']


# Indicamos que las columnas del dataframe corresponden a las
almacenadas en la lista "columns"

df = pd.DataFrame(columns= columns)


# Hacemos uso del método find de la clase beautifulsoup sobre nuestro
objeto "soup"

# para identificar el elemento html table

table = soup.find('table', attrs={'class': 'tablesaw',
'data-tablesaw-mode': 'swipe'}).tbody


# Dentro de la tabla almacenada en el objeto "table", encontramos los
elementos html tr
```

```
# que identifican las filas de la tabla

trs = table.find_all('tr')


# Recorremos cada una de las filas de la tabla, iterando sobre los
elementos que contienen y que se corresponden con las

# columnas de la tabla

for tr in trs:

   tds = tr.find_all('td')

   row = [td.text.replace('\n', '')for td in tds]

   df = df.append(pd.Series(row, index= columns), ignore_index=True)


# Mostramos por pantalla el dataframe obtenido de la iteración sobre
la tabla html

print(df)


# Hacemos uso de la librería "os" para identificar el path del codigo

directory_of_python_script                                          =
os.path.dirname(os.path.abspath(__file__))


# Transformamos el dataframe a formato CSV y lo guardamos en la
ubicación del código

df.to_csv(os.path.join(directory_of_python_script,
"nbateams_stats_20-21.csv"))
```

10. Dataset. Publicación del dataset en formato CSV en Zenodo (obtención del DOI) con una breve descripción.

Debido a que el *web scraper* no respeta el archivo robots.txt, como ya se ha explicado en el primer apartado, al ser el proyecto con fines meramente académicos, no se va a publicar el CSV en Zenodo.

Contribuciones al trabajo

Contribuciones	Firma
Investigación previa	Toni de la Rubia Enrique Martínez Gestoso
Redacción de las respuestas	Toni de la Rubia Enrique Martínez Gestoso
Desarrollo del código	Toni de la Rubia