# OBTENCIÓN DE DATOS

## WEB SCRAPING

ESTADÍSTICAS 130 MEJORES JUGADORES 2022

- https://www.mlb.com/es/stats/2022

UBICACIÓN DE LOS ESTADIOS

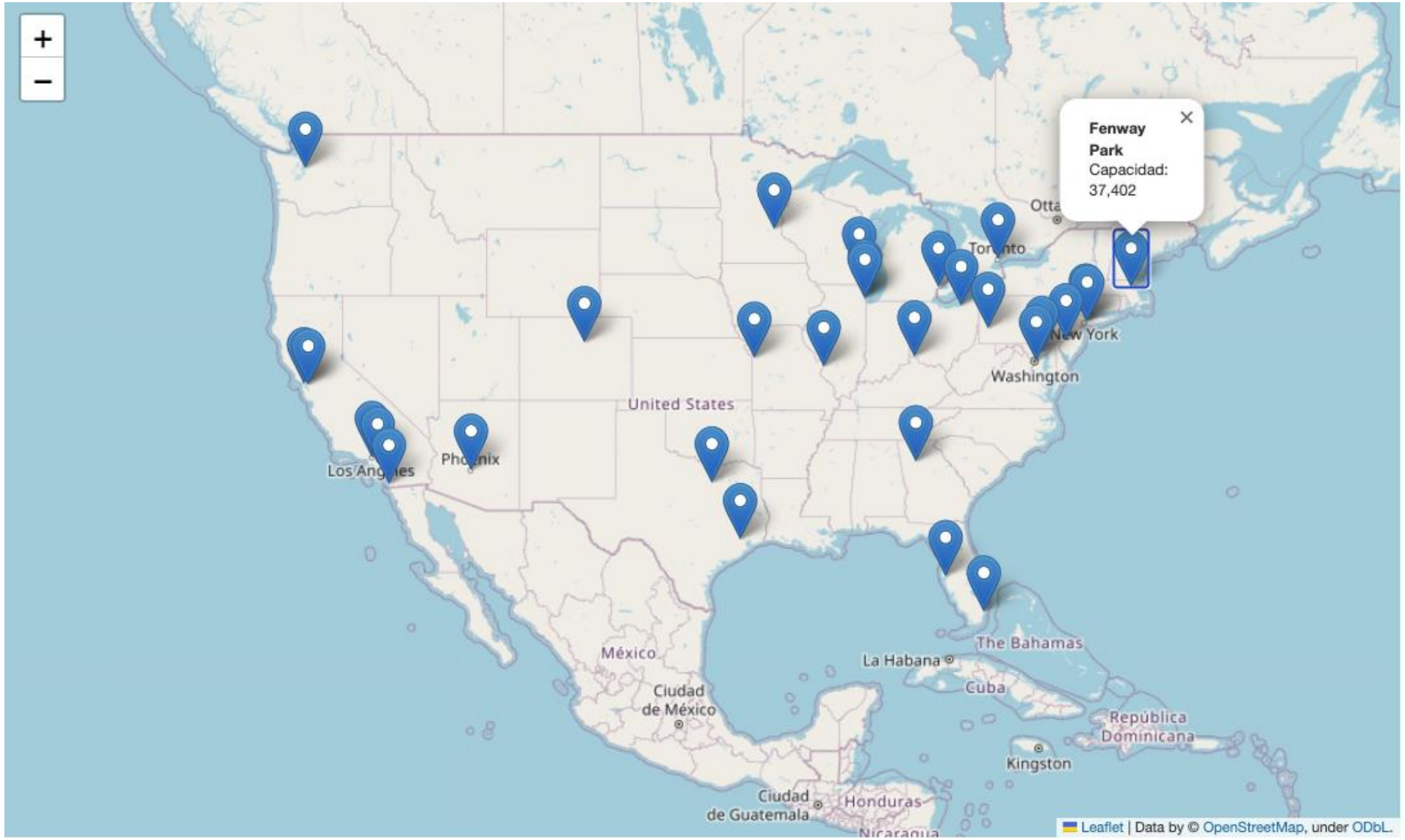
- https://es.m.wikipedia.org/wiki/Anexo:Estadios_de_B%C3%A9isbol_de_las_Grandes_Ligas

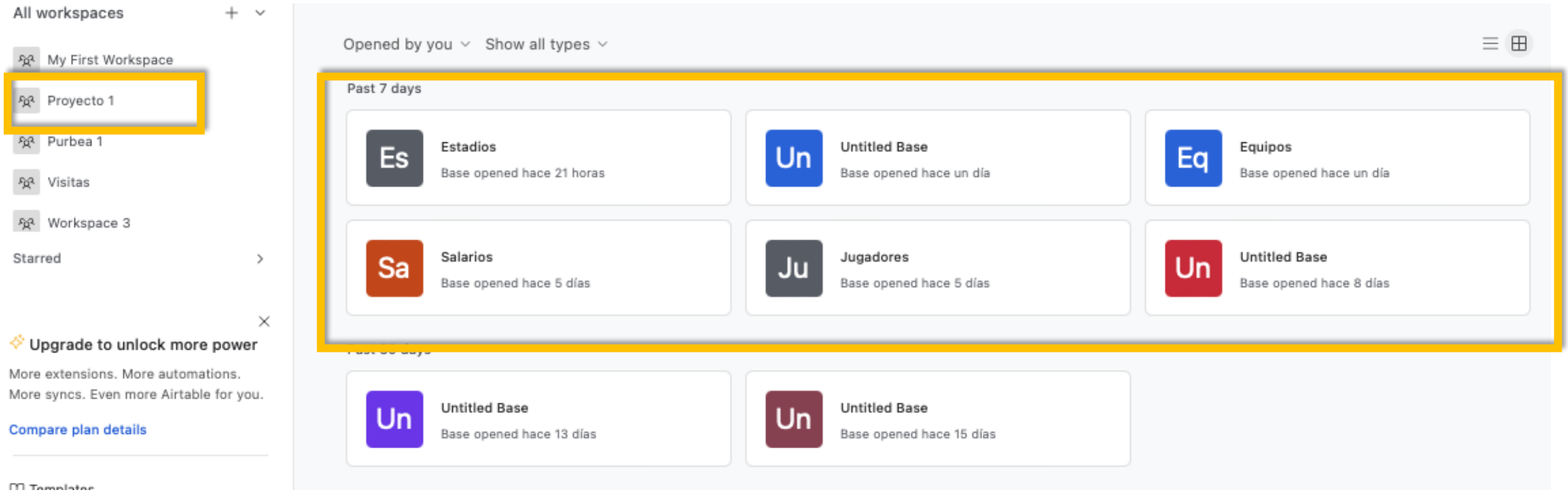
## API

ESTADÍSTICAS DE LOS EQUIPOS

- https://rapidapi.com/api-sports/api/api-baseball/

# ESTADIOS


Fenway Park
Capacidad: 37,402
Toronto
New York
Washington
United States
Phoenix
México
Ciudad de México
La Habana
Cuba
The Bahamas
República Dominicana
Kingston
Ciudad de Guatemala
Honduras
Leaflet | Data by © OpenStreetMap, under ODbL.




**Generamos un proyecto con varias tablas.**

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 130 entries, 0 to 129
Data columns (total 31 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   id                     130 non-null    object
 1   Nombre                 130 non-null    object
 2   Apellido               130 non-null    object
 3   Posicion               130 non-null    object
 4   Dorsal                 130 non-null    object
 5   Equipo                 130 non-null    object
 6   Fecha Nac              130 non-null    object
 7   Edad                   130 non-null    int64
 8   Lugar_Nacimient        130 non-null    object
 9   F_Debut                130 non-null    object
 10  Peso                   130 non-null    object
 11  Estatura(mtrs)         130 non-null    float64
 12  Mano_Bateo             130 non-null    object
 13  Mano_Lanzar            130 non-null    object
 14  Link_Ficha_Play r      130 non-null    object
 15  Team                   130 non-null    object
 16  Juegos_Jugados         130 non-null    object
 17  Turnos_Bate            130 non-null    object
 18  Carrera                130 non-null    object
 19  Hits                   130 non-null    object
 20  Dobles                 130 non-null    objec t
 21  Triple                 130 non-null    objec t
```

**La información almacenada como fecha la pasamos a texto.**

**La información almacenada como int o str no da ese problema.**

```
In [*]: df_fullplayer_stat_subir["Fecha Nac"] = df_fullplayer_stat_subir["Fecha Nac"].astype("string")
```

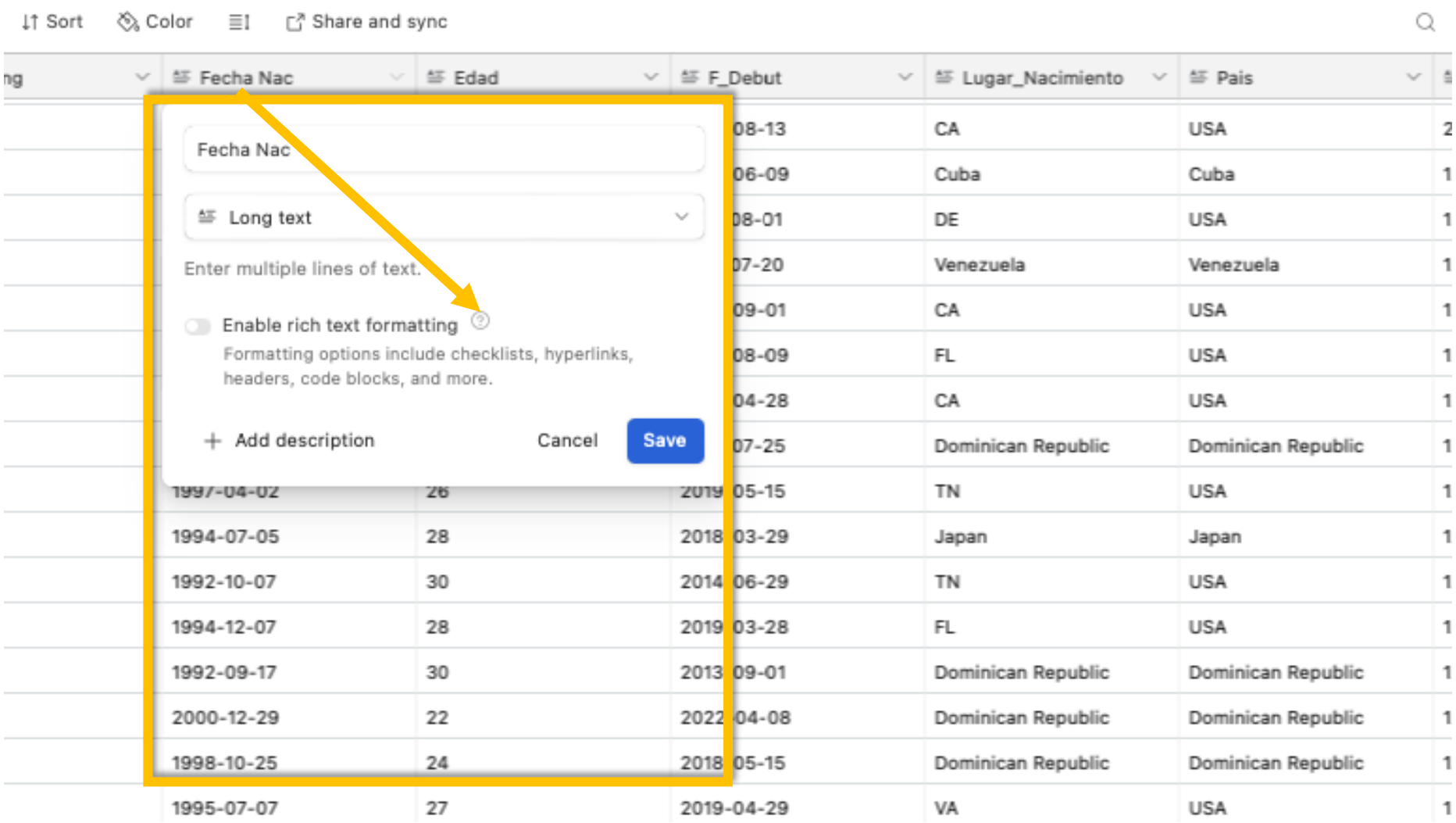
```
In [*]: df_fullplayer_stat_subir["F_Debut"] = df_fullplayer_stat_subir["F_Debut"].astype("string")
```

**Espacios almacenados en los textos de las columnas.**

```
df_fullplayer_stat_subir2 = [x.strip() for x in df_estadios_subir.columns]
```

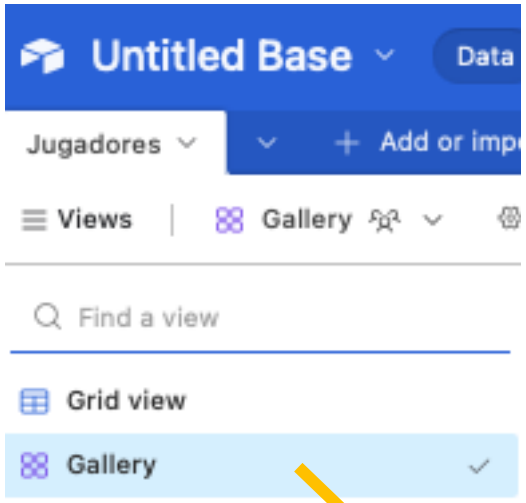
**Generamos todas las columnas como ”Long Text”, incluso las fechas, ya que es adecuado para el uso que le damos.**

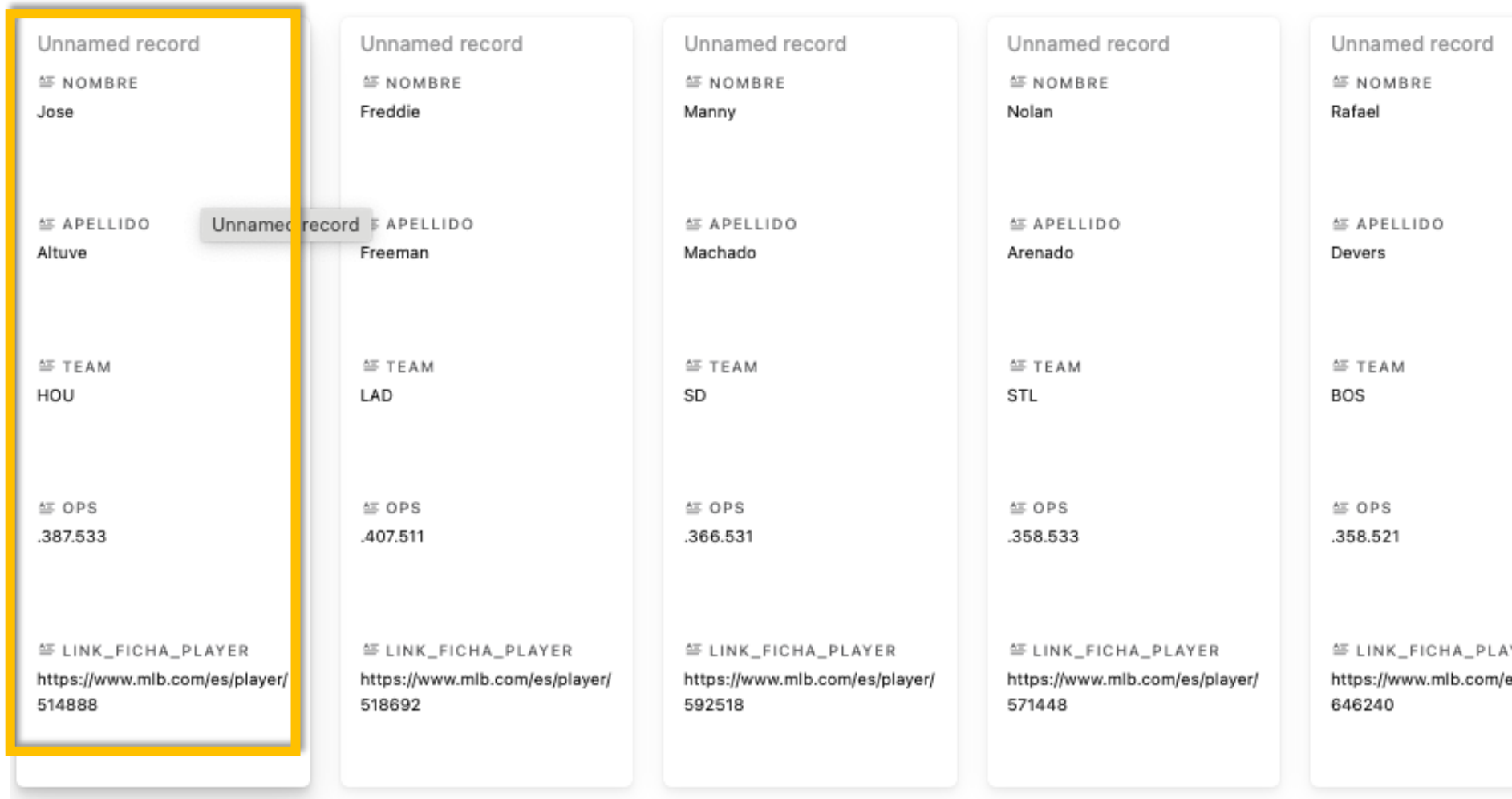
Se puede realizar con otros formatos.



## Una posible utilidad de las formas de visualización

Airtable permite **combinar información** que subimos con entradas manuales



# TOP PLAYERS 2022 (OPS> 500) 130 JUGADORES

**Jugador** **Equipo**

**Bateo** Pitcheo

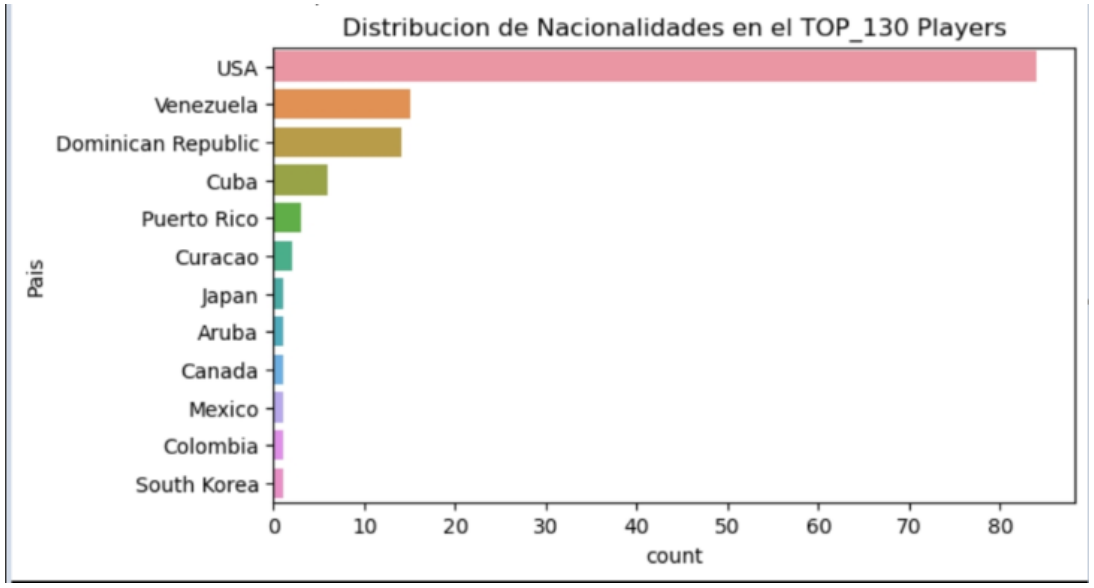
Reajusta filtros

2022 | Temporada Regular | MLB | Todas Posiciones | Elige al Jugador | Elige una Dividida | Estándar | Expandido | Statcast

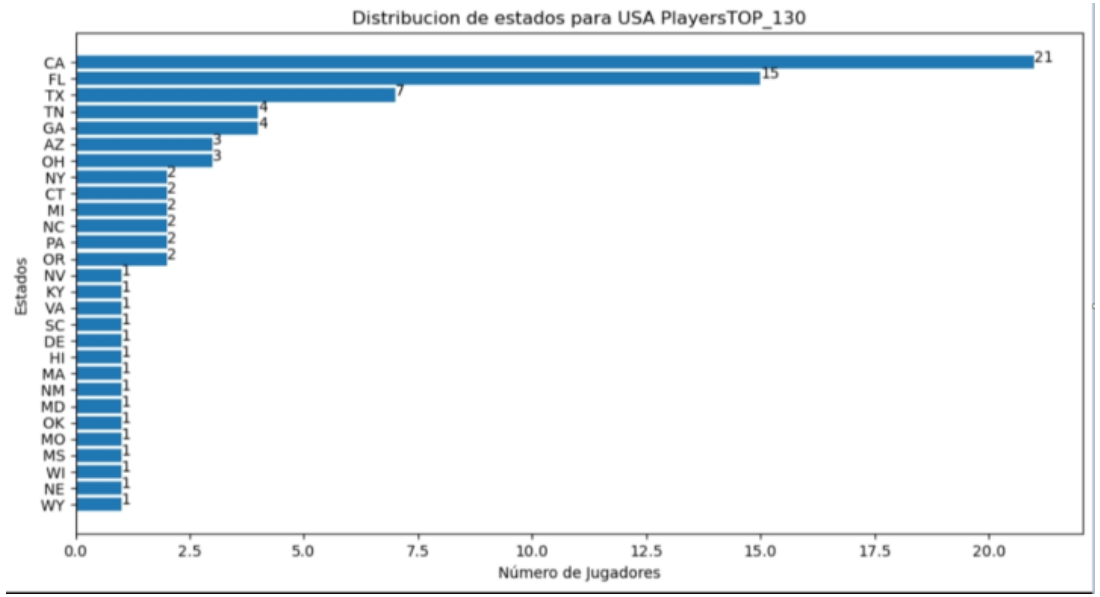
| | JUGADOR | EQUIPO | J | TB | C | H | 2B | 3B | HR | CI | BB | P | BR | AR | PRO | OBP | SLG | OPS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Aaron Judge CF | NYY | 157 | 570 | 133 | 177 | 28 | 0 | 62 | 131 | 111 | 175 | 16 | 3 | .311 | .425 | .686 | 1.111 |
| 2 | Yordan Alvarez DH | HOU | 135 | 470 | 95 | 144 | 29 | 2 | 37 | 97 | 78 | 106 | 1 | 1 | .306 | .406 | .613 | 1.019 |
| 3 | Paul Goldschmidt 1B | STL | 151 | 561 | 106 | 178 | 41 | 0 | 35 | 115 | 79 | 141 | 7 | 0 | .317 | .404 | .578 | .982 |
| 4 | Jose Altuve 2B | HOU | 141 | 527 | 103 | 158 | 39 | 0 | 28 | 57 | 66 | 87 | 18 | 1 | .300 | .387 | .533 | .920 |
| 5 | Freddie Freeman 1B | LAD | 159 | 612 | 117 | 199 | 47 | 2 | 21 | 100 | 84 | 102 | 13 | 3 | .325 | .407 | .511 | .918 |
| 6 | Manny Machado 3B | SD | 150 | 578 | 100 | 172 | 37 | 1 | 32 | 102 | 63 | 133 | 9 | 1 | .298 | .366 | .531 | .897 |

# EL BASEBALL ES UN DEPORTE CARIBEÑO

## NACIONALIDAD JUGADORES TOP 130

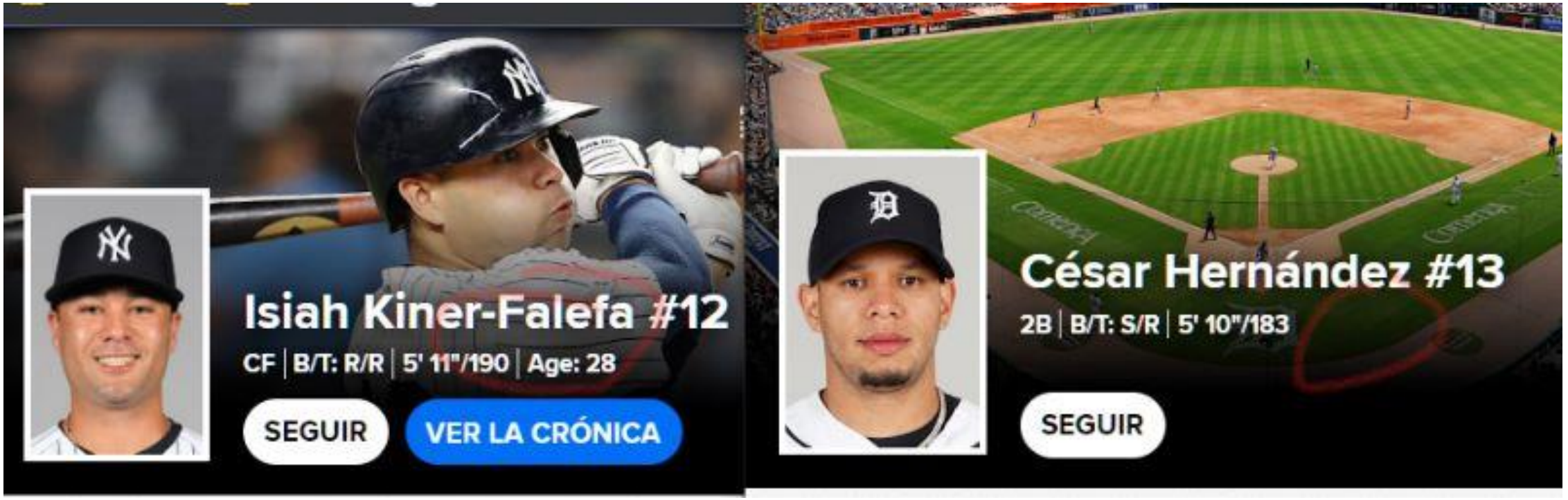



## ESTADOS QUE MÁS APORTAN AL TOP 130



# Algunos “problemas” a solucionar…..

Faltaban datos en algunos jugadores  o no todos estaban en la misma posición en la pagina




```
for i in Fecha_nac:
    edad=(datetime.now() - datetime.strptime(i, "%m/%d/%Y")).days // 365
    Edad.append(edad)
```

# Otros aspectos de limpieza y manipulación

- Estatura en Pie y pulgadas
- Peso en Libras
- Fechas sistema Mes / Dia / Año
- Añadir el IMC

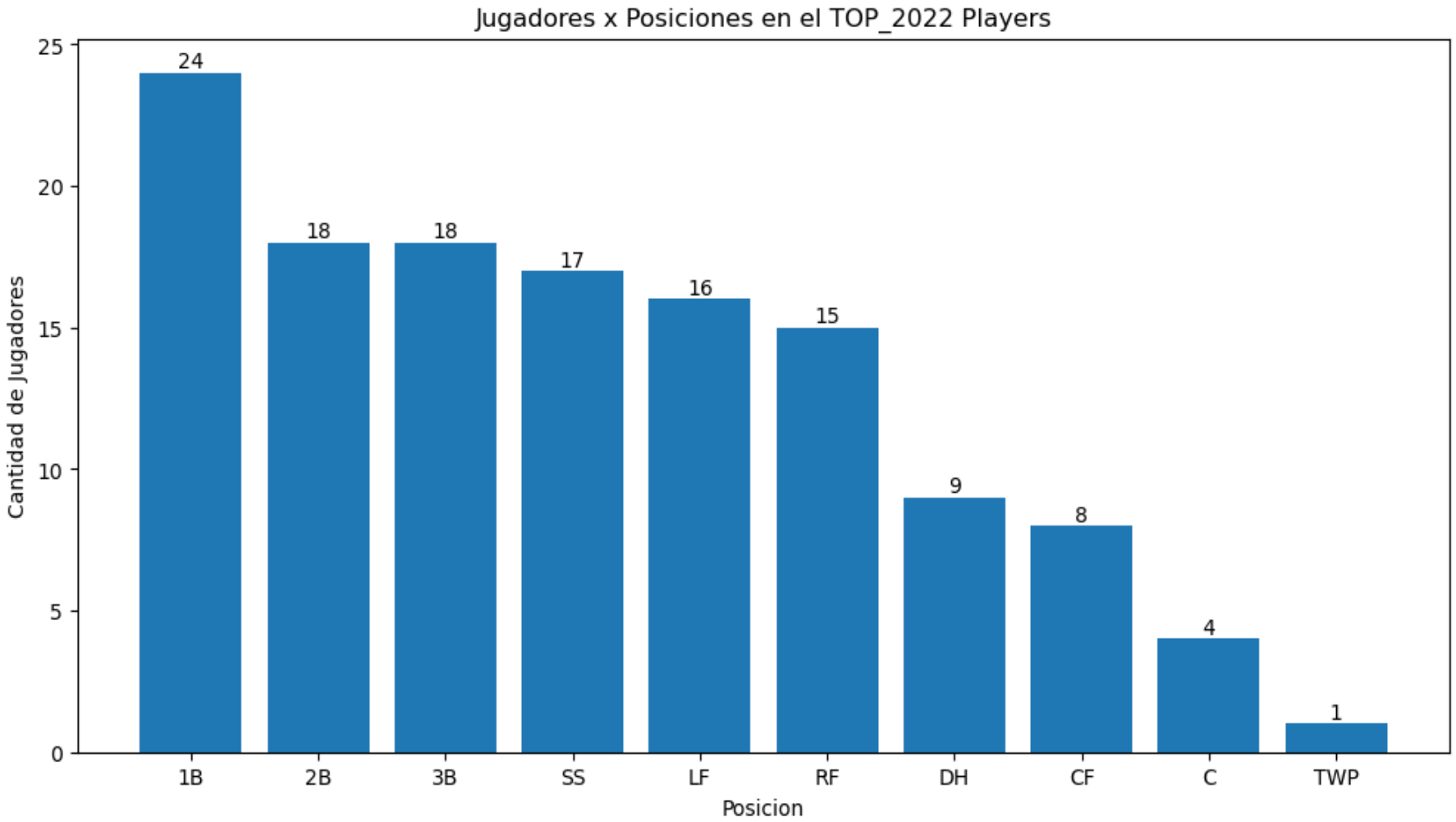
```
c=0
for jugador in (lista_link_player):

    print(f"Faltan {len(lista_link_player)- c } jugadores")
    print(jugador)
    c+=1

    response = requests.get(jugador)
    soup = BeautifulSoup(response.text, "html.parser")
    sleep(0.02)
```

Mejoramos la “experiencia” de espera mientras se ejecuta el código del Web Scrapping

```
https://www.mlb.com/es/player/663757
Faltan 4 jugadores
https://www.mlb.com/es/player/624428
Faltan 3 jugadores
https://www.mlb.com/es/player/500743
Faltan 2 jugadores
https://www.mlb.com/es/player/664702
Faltan 1 jugadores
https://www.mlb.com/es/player/570731
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 130 entries, 0 to 129
Data columns (total 33 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   id                  130 non-null    int64
 1   Dorsal              130 non-null    int64
 2   Nombre              130 non-null    object
 3   Apellido            130 non-null    object
 4   Team                130 non-null    object
 5   Posicion            130 non-null    object
 6   OPS                 130 non-null    float64
 7   %_On_Base           130 non-null    float64
 8   Slugging            130 non-null    float64
 9   Fecha Nac           130 non-null    object
 10  Edad                130 non-null    int64
 11  F_Debut             130 non-null    object
 12  Lugar_Nacimiento    130 non-null    object
 13  Pais                130 non-null    object
 14  Peso                130 non-null    float64
 15  Estatura(mtrs)      130 non-null    float64
 16  Mano_Bateo          130 non-null    object
 17  Mano_Lanzar         130 non-null    object
 18  Juegos_Jugados      130 non-null    int64
 19  Turnos_Bate         130 non-null    int64
 20  Carrera             130 non-null    int64
 21  Hits                130 non-null    int64
 22  Dobles              130 non-null    int64
 23  Triple              130 non-null    int64
 24  Jonrones            130 non-null    int64
 25  Carreras_Impulsadas 130 non-null    int64
 26  Boletos             130 non-null    int64
 27  Ponches             130 non-null    int64
 28  Bases_Robadas       130 non-null    int64
 29  Atrapado_Robando    130 non-null    int64
 30  Promedio            130 non-null    float64
 31  Link_Ficha_Player   130 non-null    object
 32  IMC                 130 non-null    float64
dtypes: float64(7), int64(15), object(11)
memory usage: 33.6+ KB
```

Dataframe Resultante con 130 Filas(Jugadores) y 33 Columnas



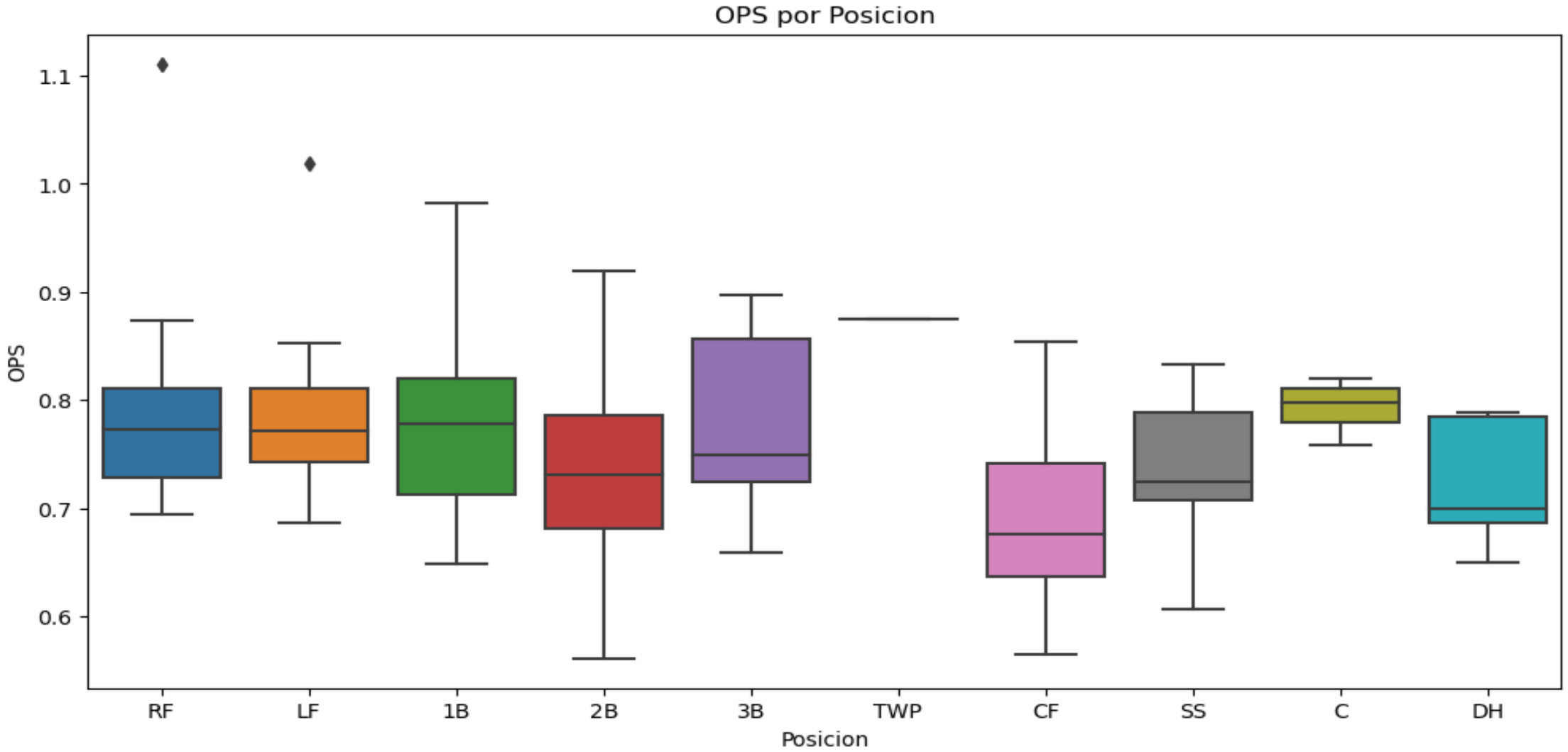

- P: Pitcher /Lanzador
- C: Catcher/Receptor
- 1B: First Base
- 2B: Second Base
- 3B: Third Base
- SS: Shortstop/Campo Corto
- LF: Left Field/Jardinero Izquierdo
- CF: Center Field/Jardinero Central
- RF: Right Field/Jardinero Derecho
- DH: Designated hitter/ Bateador designado
- TWP: Two way player/Jugador bidireccional

•Hay más jugadores de infield(1B,2B,3B,SS) que de outfield(CF. RF.LF)

•Destaca mayor cantidad jugadores de 1B.(Mas bateadores¿?)

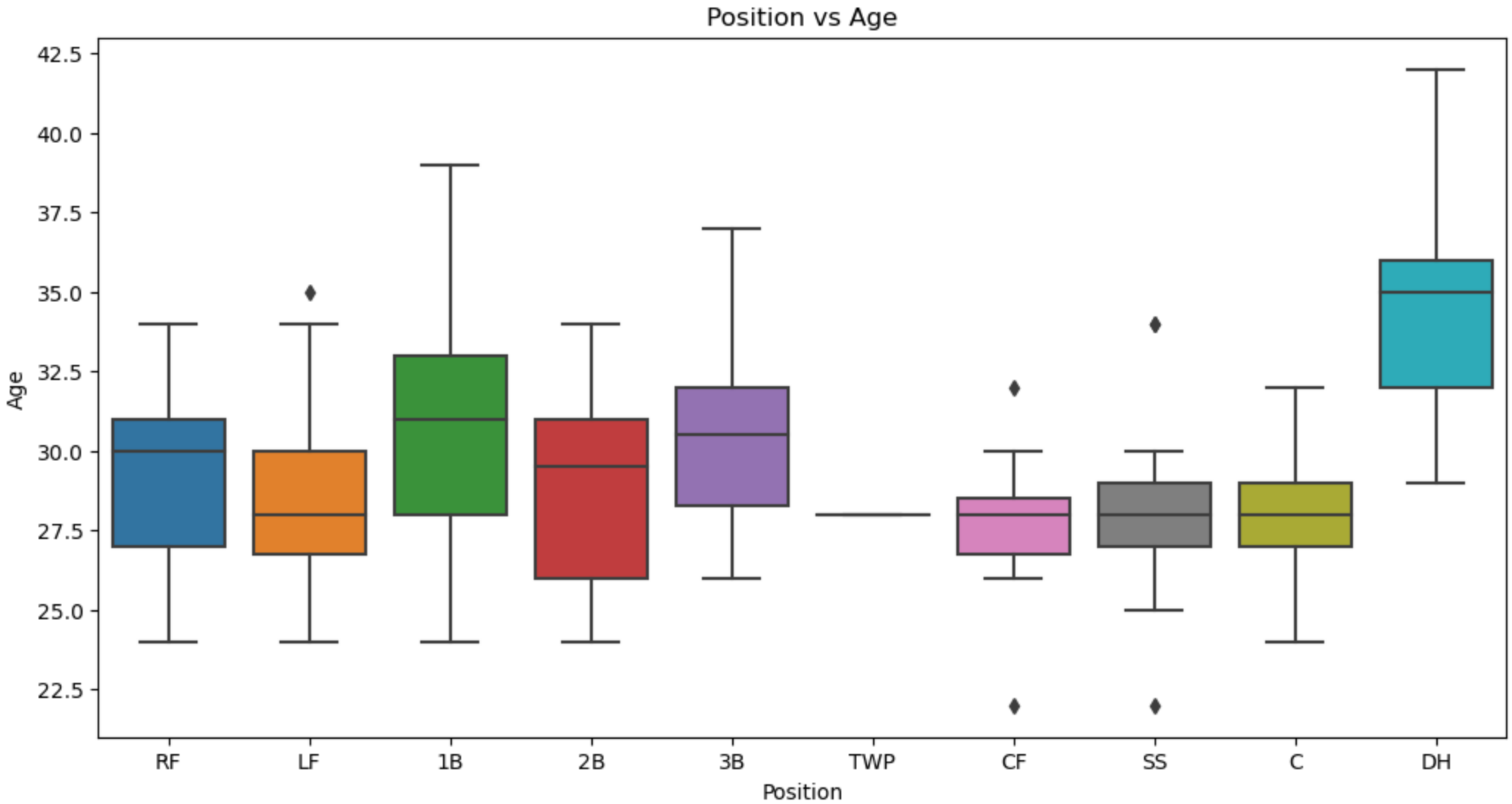
•Aparece la "categoría especial TWP es un jugador que es picher y bateador".. (no se contabiliza como pitcher en la alineación del equipo)Othani

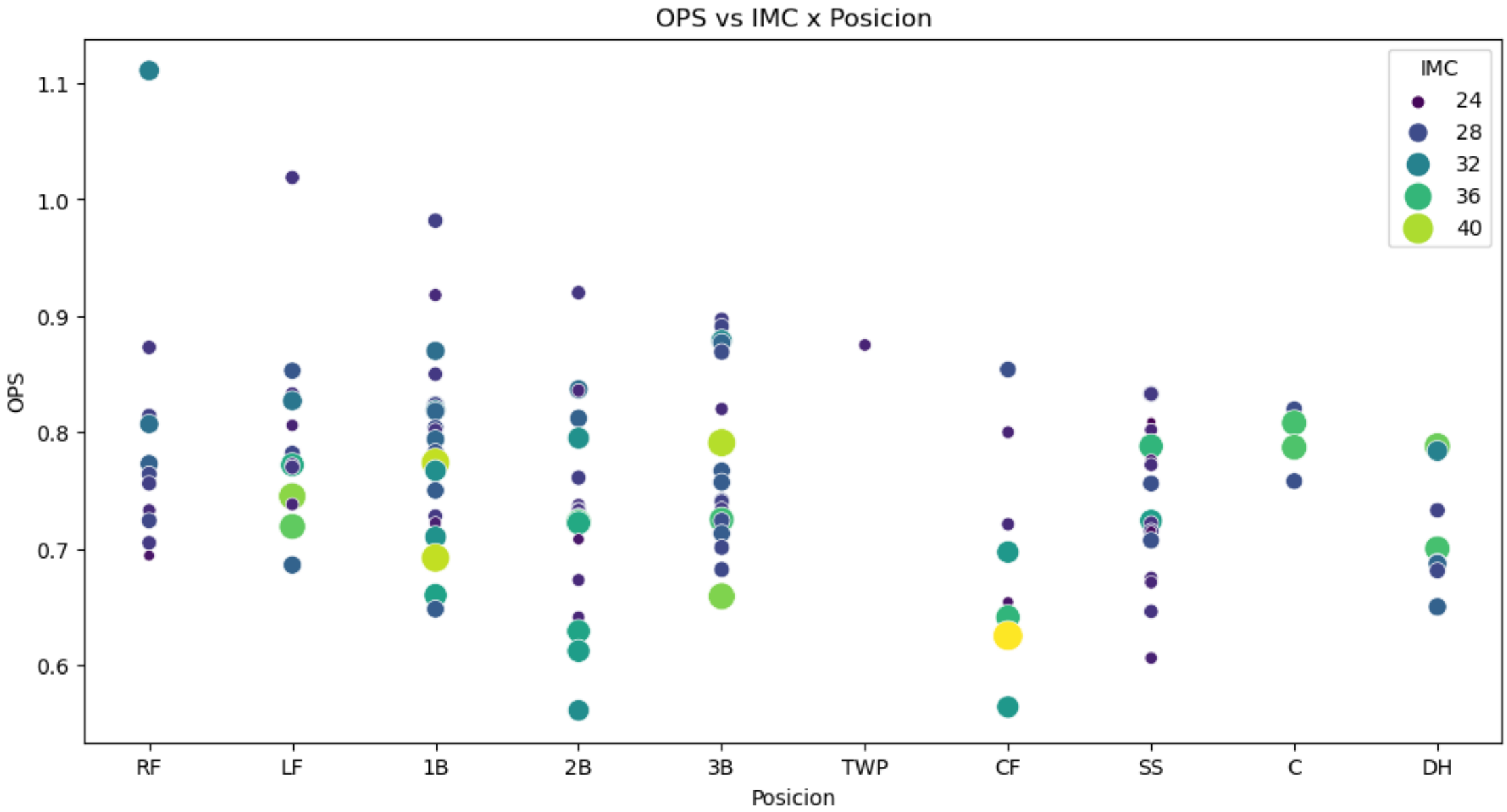
# Análisis del comportamiento del OPS x Posición



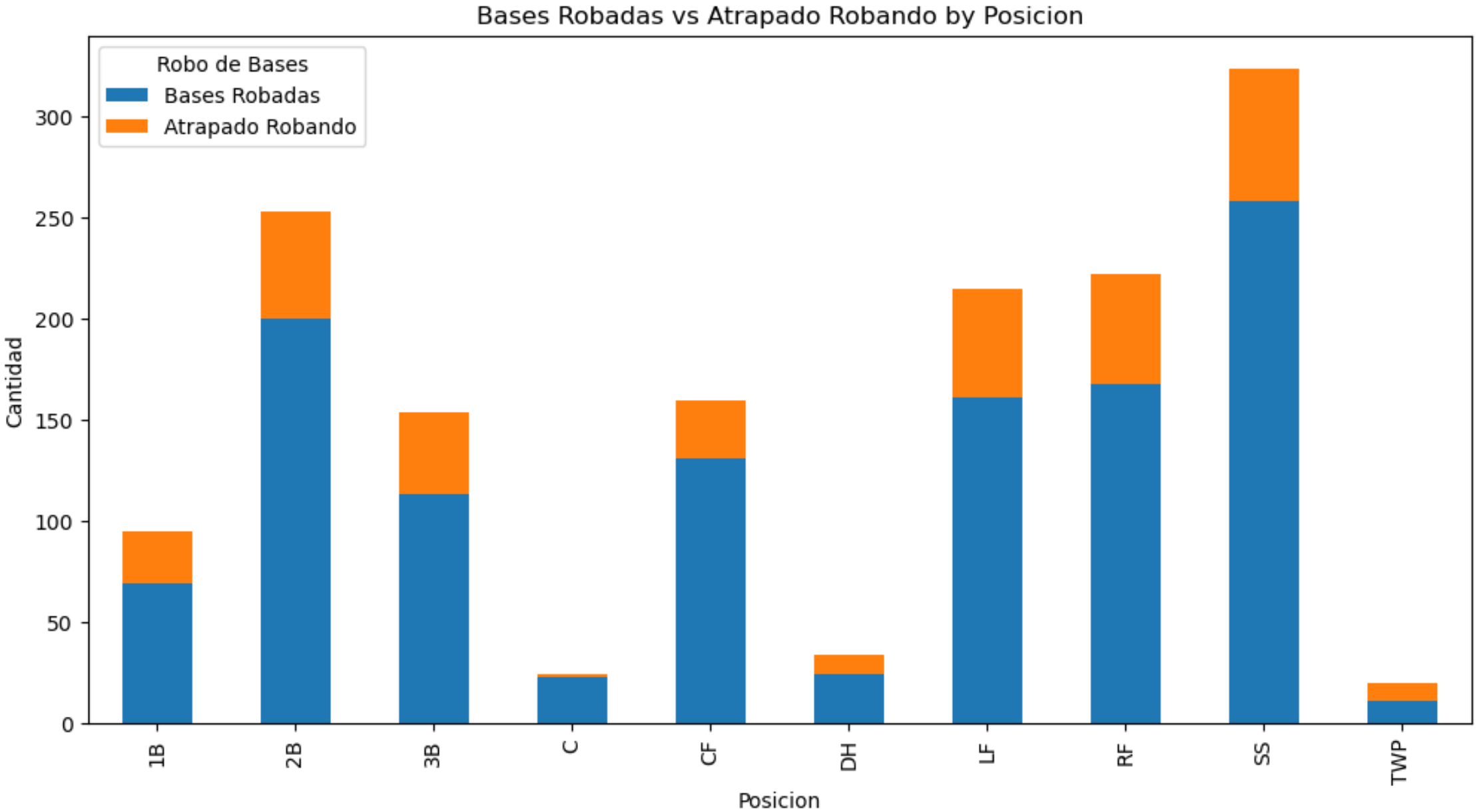
Conclusiones: De manera general el tamaño de los bigotes nos representa el amplio rango de OPS de los jugadores en cada posición

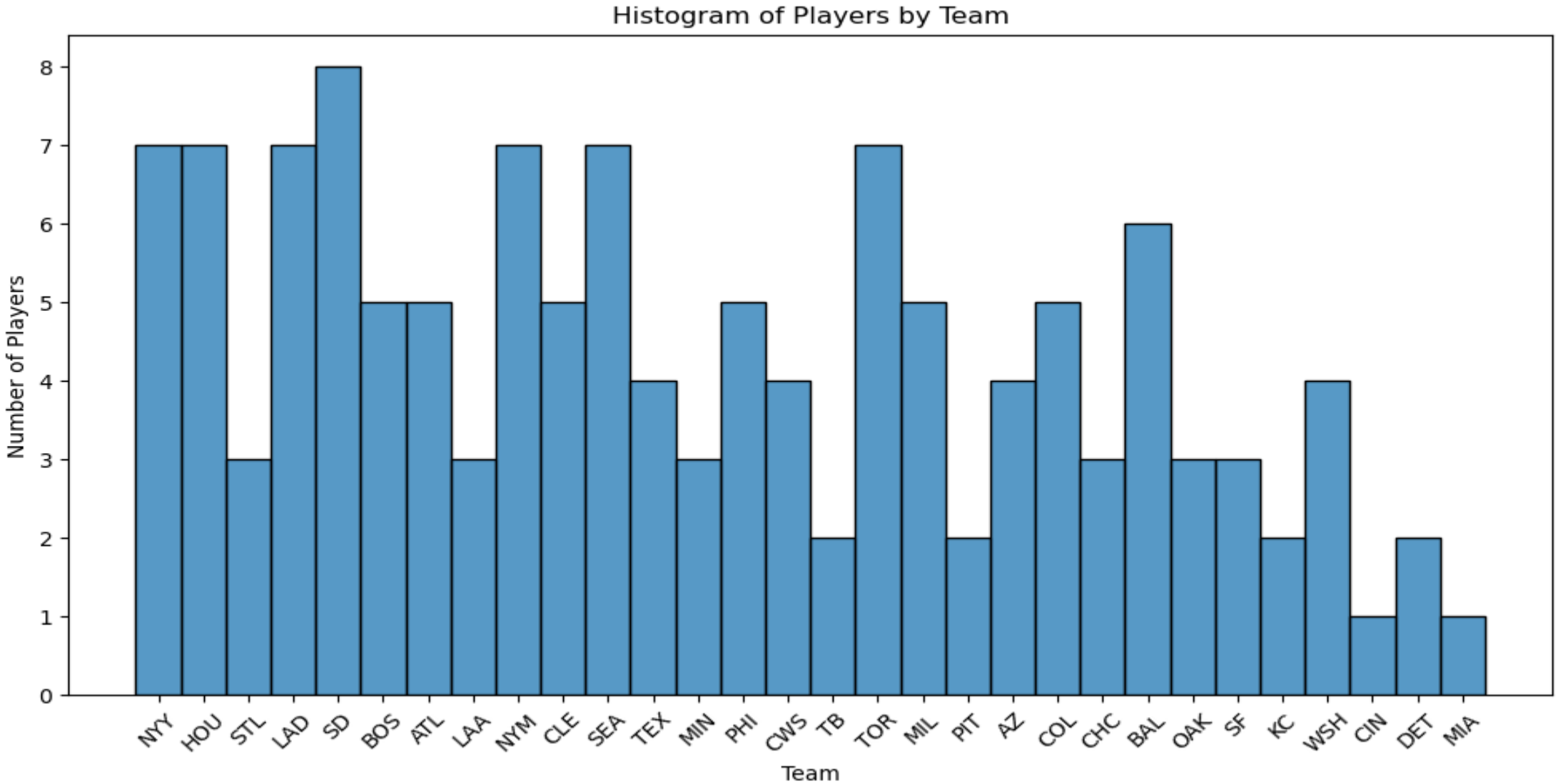
1. La posición con el valor medio de OPS más alto es: TWP(esta categoría la tiene solo 1 jugador), pero no se puede tomar en consideración, como valor medio ya que es un único jugador el de esa posición.
2. La posición con el valor medio más bajo de OPS es: CF con lo cual tienen menor rendimiento ofensivo 3.La posición con mayor IQR es: 3B, los jugadores de estas posiciones tienen gran dispersión en su ofensiva
3. La posición con el rango Inter cuartil más pequeño (RIC) es el CATCHER
4. En el caso de los Outliers tenemos dos jugadores que son los que encabezaron la ofensiva de la temporada con valores considerables, inclusive son outliers en cualquiera de las posiciones
5. De manera general influyen dos factores las habilidades técnicas y la experiencia


Position vs Age
42.5
40.0
37.5
35.0
32.5
30.0
27.5
25.0
22.5
Age
RF
LF
1B
2B
3B
TWP
CF
SS
C
DH
Position


OPS vs IMC x Posicion
IMC
24
28
32
36
40
OPS
1.1
1.0
0.9
0.8
0.7
0.6
RF
LF
1B
2B
3B
TWP
CF
SS
C
DH
Posicion

- En el diagrama de dispersión anterior, podemos observar:
- 1. Existe una amplia gama de valores de OPS en diferentes posiciones de acuerdo a las IMC, lo que indica niveles variables de desempeño ofensivo. Tendiendo a estar mas elevado en el caso de los Jardineros derechos
- 2. La distribución de los valores de OPS dentro de cada posición también varía, con algunas posiciones que tienen un rango de valores más concentrado, mientras que otras tienen un rango más disperso
- 3. El IMC (índice de masa corporal) de los jugadores parece tener cierta influencia en los valores de OPS, como lo indican el color y el tamaño de los puntos. Sin embargo, no está claro si existe una fuerte correlación entre IMC y OPS.
- 4. Es posible que se necesite un análisis adicional para determinar si hay diferencias significativas en los valores de OPS entre posiciones o si hay una fuerte relación entre IMC y OPS.
- 5. De cualquier modo los de mayor IMC sus OPC están por debajo de la media de OPC(0.79)


Bases Robadas vs Atrapado Robando by Posicion
Robo de Bases
Bases Robadas
Atrapado Robando
Cantidad
0
50
100
150
200
250
300
1B
2B
3B
C
CF
DH
LF
RF
SS
TWP
Posicion


Histogram of Players by Team
Number of Players
0
1
2
3
4
5
6
7
8
NYY
HOU
STL
LAD
SD
BOS
ATL
LAA
NYM
CLE
SEA
TEX
MIN
PHI
CWS
TB
TOR
MIL
PIT
AZ
COL
CHC
BAL
OAK
SF
KC
WSH
CIN
DET
MIA
Team

| | Team | Total_OPS | Num_Jugadores | Mean_OPS |
|---|---|---|---|---|
| 22 | SD | 6.123 | 8 | 0.765375 |
| 10 | HOU | 5.640 | 7 | 0.805714 |
| 28 | TOR | 5.459 | 7 | 0.779857 |
| 23 | SEA | 5.086 | 7 | 0.726571 |
| 18 | NYY | 5.519 | 7 | 0.788429 |
| 17 | NYM | 5.604 | 7 | 0.800571 |
| 13 | LAD | 5.563 | 7 | 0.794714 |
| 2 | BAL | 4.247 | 6 | 0.707833 |
| 0 | ATL | 3.906 | 5 | 0.781200 |
| 20 | PHI | 3.848 | 5 | 0.769600 |
| 15 | MIL | 3.768 | 5 | 0.753600 |
| 6 | CLE | 3.758 | 5 | 0.751600 |
| 3 | BOS | 3.920 | 5 | 0.784000 |
| 7 | COL | 3.714 | 5 | 0.742800 |
| 8 | CWS | 2.962 | 4 | 0.740500 |
| 27 | TEX | 3.111 | 4 | 0.777750 |
| 29 | WSH | 2.694 | 4 | 0.673500 |

Como conclusión de estas visualizaciones en el dataframe que la mayor influencia para ser mejor equipo está dada por la combinación (Mayor cantidad de jugadores + Mayor OPS promedio).. esta combinación lleva a determinar cuáles pueden ser uno de los favoritos del campeonato EN ESTE CASO Houston es el equipo que aporta la mayor combinación cantidad de jugadores + mayor media OPS( EN ESE ORDEN)