

MODERN DATA ARCHITECTURES FOR BIG DATA II

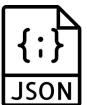
THE MONEYBALL PROJECT – FINDING UNDervalued ASSETS

By: Group 3 – Ignacio Alonso, Baptiste Richard, Kate Bade, Zeyad Al-houti,
Ana Belén Guerra, Enrico Bonilauri

Agenda for today!

[CONTEXT]

The Use Case: Problem Explanation



Data Sources: API-FOOTBALL.COM

Data Sources: Transfermarkt Player Values

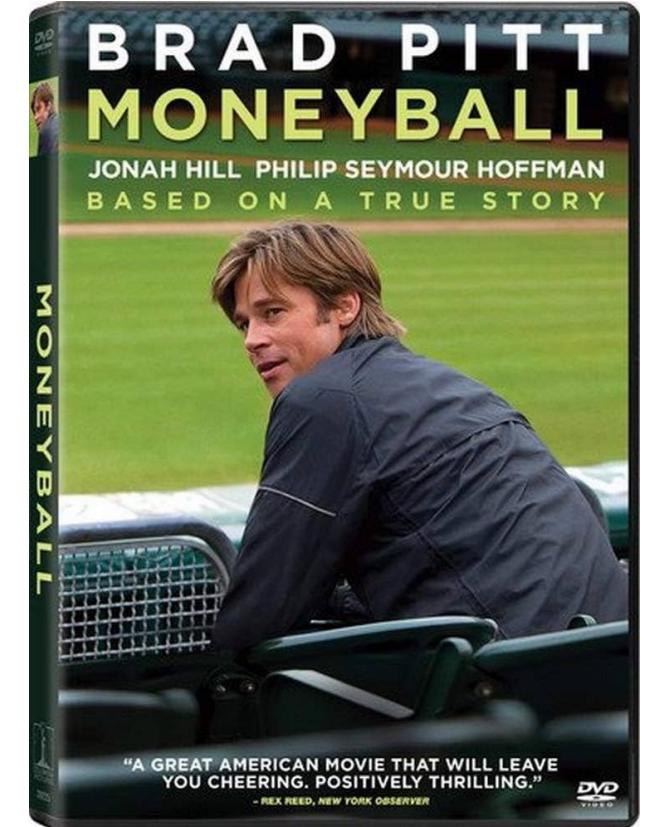


Data Processing:

1. Cleaning the data
2. Machine Learning



Project Results & Our Conclusions



The Use Case: Problem Explanation

With football being the most popular sport, how can you find talent without vast economical resources? using statistics and data.

This is how **Billy Beane and Paul DePodesta brought the Oakland Athletics to win** the Baseball American League West and this is what is portrayed in the movie, Moneyball.

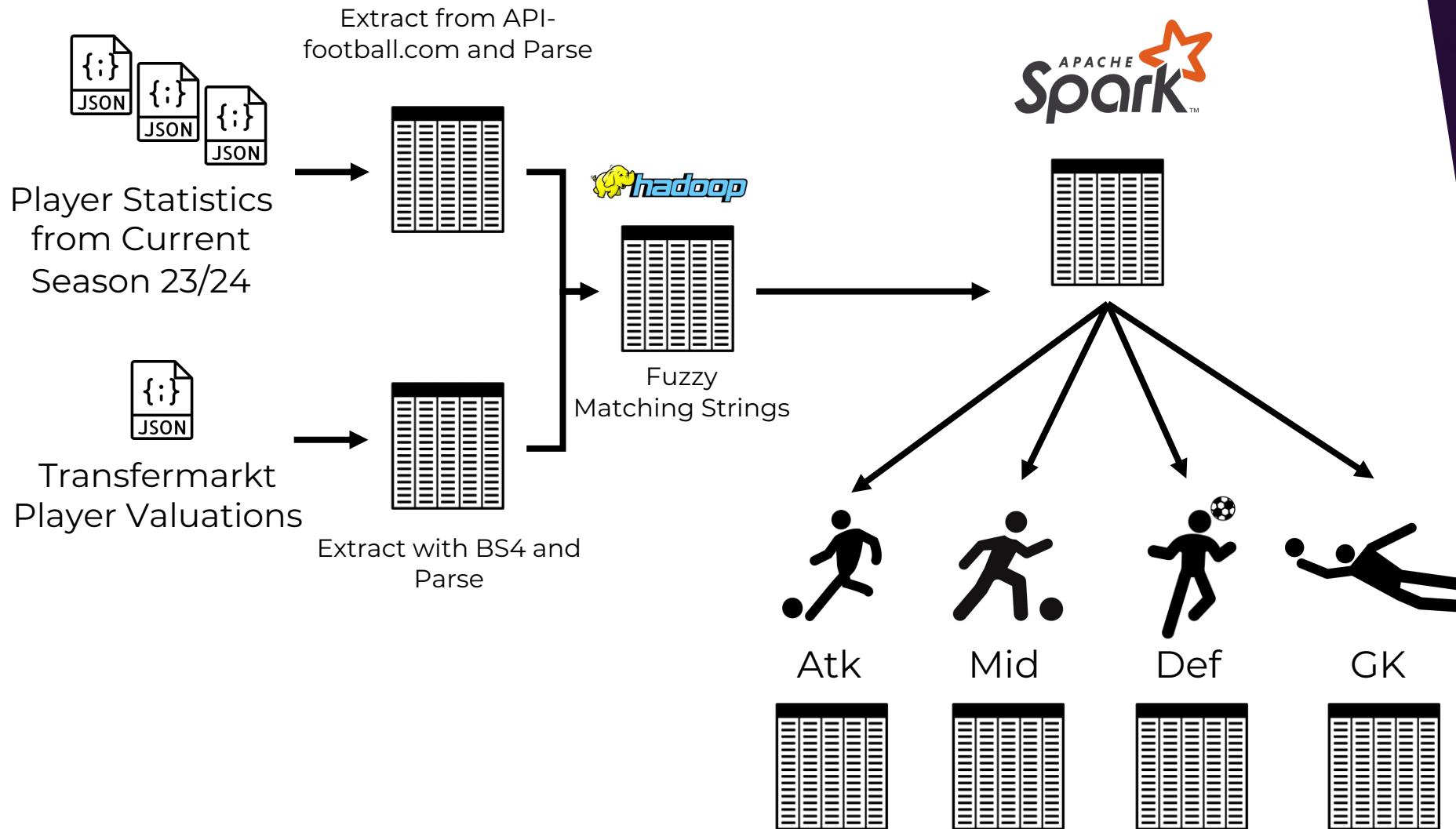
This analytical approach to sports is very useful for various reasons:

- 1. Knowledge:** In Sports, statistics are performance, knowing what to improve and where you are good at. Thus, finding what you lack!
- 2. Value Perception:** Statistics or performance correlate with prices but that is not always the case as we will see later. You can find good players at a discount!
- 3. Ultimately Winning:** with undervalued players that are performing greatly, you can make the most out of your team and eventually win.



In a nutshell, Analytics are essential in sports to discover undervalued players and measure performance to prices

The Use Case: Process Outline



Data Sources: API-FOOTBALL.COM

Popularity: 10 / 10 | Latency: 357ms | Service Level: 100% | Health Check: N/A

Subscribed

Endpoints About Tutorials Discussions Pricing

+1 050 football leagues & cups. Livescore (15s), live & pre-match odds, events, line-ups, coaches, pl Show more...

V1 (Current)

Search endpoints

GET V3 - Timezone

Test Endpoint

Code Snippets Example Responses Results

(Node.js) Axios Copy Code

```
const axios = require('axios');

const options = {
  method: 'GET',
  url: 'https://api-football-v1.p.rapidapi.com/v3/timezone',
  headers: {
    'X-RapidAPI-Key': 'YOUR_RAPIDAPI_KEY',
    'X-RapidAPI-Host': 'api-football-v1.p.rapidapi.com'
  }
};

try {
```

Events

Standings

Statistics

Players

LineUps

Trophies

Personal Account Ignacio Alonso

Language: English

Home > Api Football

© 2024 RapidAPI. All Rights Reserved. About Blog Learn Careers Press Contact Terms Privacy

We extracted data **from the 5 major countries:** UK, Spain, France, Italy and Germany and their first 3 divisions for each totaling almost 15 leagues

Ex. From Spain, Data from La Liga, Segunda División and Primera RFEF

Through the dashboard, you can discover the league ids

We subscribed to API-
football.com, an API that
lets you query over 6K
leagues with many
endpoints.

They also offer other sport
APIs such as Basketball,
Baseball, Formula 1 or NFL!

League Id (v3)	League Id (v2)	Country	Name	Season	Start	End	Current
140	5284	Spain	La Liga	2023	2023-08-11	2024-05-26	True
141	5323	Spain	Segunda División	2023	2023-08-11	2024-06-02	True
142	5576	Spain	Primera División Femenina	2023	2023-09-10	2024-06-16	True
143	5796	Spain	Copa del Rey	2023	2023-10-11	2024-04-06	True
435	5511	Spain	Primera División RFEF - Group 1	2023	2023-08-26	2024-05-26	True
436	5512	Spain	Primera División RFEF - Group 2	2023	2023-08-27	2024-05-26	True
437	2952	Spain	Primera División RFEF -	2020	2020-10-	2021-	True

Data Sources: Transfermarkt Market Values

Scraping Code with BS4

```
from bs4 import BeautifulSoup
from tqdm import tqdm
import unicodedata

ligas = []
equipos = []
nombre = []
posiciones = []
edades_actuales = []
nacionalidades_list = []
valores_mercado = []

urls = ['https://www.transfermarkt.es/laliga/startseite/wettbewerb/ES1/plus/?saison_id=2023',
        'https://www.transfermarkt.es/laliga2/startseite/wettbewerb/ES2',
        'https://www.transfermarkt.es/primer-a-division-r-f-e-f-grupo-i-startseite/wettbewerb/E3G1',
        'https://www.transfermarkt.es/primer-a-grupo-ii-startseite/wettbewerb/E3G2',
        'https://www.transfermarkt.es/serie-a/startseite/wettbewerb/IT1',
        'https://www.transfermarkt.es/b/startseite/wettbewerb/IT2',
        'https://www.transfermarkt.es/liga-1/startseite/wettbewerb/FR1',
        'https://www.transfermarkt.es/liga-2/startseite/wettbewerb/FR2',
        'https://www.transfermarkt.es/championnat-national/startseite/wettbewerb/FR3',
        'https://www.transfermarkt.es/la-liga/startseite/wettbewerb/L1',
        'https://www.transfermarkt.es/la-liga2/startseite/wettbewerb/L2',
        'https://www.transfermarkt.es/premier-league/startseite/wettbewerb/G81',
        'https://www.transfermarkt.es/championship/startseite/wettbewerb/G82',
        'https://www.transfermarkt.es/league-one/startseite/wettbewerb/G83']

headers = {
    'Accept': 'text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,/q=0.8,application/signed-exchange;v=b3;q=0.7',
    'Accept-Encoding': 'gzip, deflate, br, zstd',
    'Accept-Language': 'es-ES,es;q=0.9',
    'Cache-Control': 'max-age=0',
    'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/122.0.0.0 Safari/537.36'
}

for i,url in enumerate(urls):
    url = url.split("/")[-1].replace("-"," ")
    response = requests.get(url, headers=headers)
    html_content = response.text

    soup = BeautifulSoup(html_content, 'html.parser')
    urls_sinrep.add(url)

    links_with_kader = soup.find_all('a', href=lambda href: href and 'kader' in href)
    for link in links_with_kader:
        urls_sinrep.add(link['href'])

    #print(urls_sinrep)

    for link in tqdm(urls_sinrep, desc=f'Liga {liga}'):

        equip = link.split("/")[-1].replace("-"," ")
        player_url = f'https://www.transfermarkt.es' + str(link)
        player_response = requests.get(player_url, headers=headers)
        #print(player_response.text)
        player_soup = BeautifulSoup(player_response.content, 'html.parser')
        jugadores = player_soup.find_all('tr')
```

Website Page

The screenshot shows the Transfermarkt website interface. At the top, there's a navigation bar with links for NOTICIAS, FICHAJES & RUMORES, VALORES DE MERCADO, COMPETICIONES, FOROS, MI TM, EN VIVO, and INICIAR SESIÓN. Below the navigation is a search bar with placeholder text "Introducir término (p.ej. jugador, club,...)". The main content area features a banner for La Liga with the EA logo. To the right of the banner, there's a summary box for Spain (España) in the Primera División, stating it has 20 teams, 507 players, and a total market value of 4.97 million euros. Below this, there's a banner for Premier League with the text "NUEVOS VALORES DE MERCADO" and "A LA ACTUALIZACIÓN". At the bottom of the page, there's a navigation menu with links for VISTA GENERAL, TABLAS, FICHAJES, VALORES DE MERCADO, JUGADORES, CLUBES, DATOS, HISTORIA, and NOTICIAS.

Okay, this part is more difficult. We wrote a program with a web-scraping package: Beautiful Soup 4, that collected through the URL all the players from the same leagues we gathered with the API, pretty sweet huh!

Data Processing: Refining the dataset

After retrieving the data, we:

- 1 Flatten the data, originally in a json file. For every league, we will make a dataframe and then combine them together, with the same structure. A dummy example is displayed.
- 2 Selected only players that played matches or had “appearances”. We are not interested in those players that belong to a league, did not play and thus, have no statistics.
- 3 Corrected club names and other data preparation techniques.

```
... [{"player": {"id": 83,  
            "name": "A. Danjuma",  
            "firstname": "Arnaut",  
            "lastname": "Danjuma Adam Groeneveld",  
            "age": 26,  
            "birth": {"date": "1997-01-31", "place": "Lagos", "country": "Nigeria"},  
            "nationality": "Netherlands",  
            "height": "178 cm",  
            "weight": "74 kg",  
            "injured": False,  
            "photo": "https://media.api-sports.io/football/players/83.png"},  
         "statistics": [{"team": {"id": 533,  
                  "name": "Villarreal",  
                  "logo": "https://media.api-sports.io/football/teams/533.png"},  
                     "league": {"id": 140,  
                               "name": "La Liga",  
                               "country": "Spain",  
                               "logo": "https://media.api-sports.io/football/leagues/140.png"},  
                     "flag": "https://media.api-sports.io/flags/es.svg",  
                     "season": 2023},  
                     "games": {"appearances": None,  
                               "lineups": None,  
                               "minutes": None,  
                               "number": None,  
                               "position": "Attacker",  
                               "rating": None,  
                               "captain": False},  
                     "substitutes": {"in": None, "out": None, "bench": None},  
                     "shots": {"total": None, "on": None},  
                     "goals": {"total": None, "conceded": None, "assists": None, "saves": None},  
                     "passes": {"total": None, "key": None, "accuracy": None},  
                     "tackles": {"total": None, "blocks": None, "interceptions": None},  
                     "duels": {"total": None, "won": None},  
                     "dribbles": {"attempts": None, "success": None, "past": None},  
                     "fouls": {"drawn": None, "committed": None},  
                     "cards": {"yellow": None, "yellowred": None, "red": None},  
                     "penalty": {"won": None, "committed": None},  
                     "scored": None,  
                     "missed": None,  
                     "saved": None}]}],  
    {"player": {"id": 274,
```



Data Processing: Refining the dataset

But how do we merge them? Two different databases: Stats + PlayerValuations...
Fuzzy Wuzzy

```
Original: atletico de madrid, Match: Atletico Madrid, Score: 95
Original: girona fc, Match: Girona, Score: 90
Original: granada cf, Match: Granada CF, Score: 100
original: real betis balompie, Match: Real Betis, Score: 90
Original: valencia cf, Match: Valencia, Score: 95
Original: real madrid cf, Match: Real Madrid, Score: 95
Original: deportivo alaves, Match: Alaves, Score: 90
Original: ud almeria, Match: Almeria, Score: 95
Original: fc barcelona, Match: Barcelona, Score: 95
Original: ca osasuna, Match: Osasuna, Score: 95
Original: rcd mallorca, Match: Mallorca, Score: 90
Original: villarreal cf, Match: Villarreal, Score: 95
Original: getafe cf, Match: Getafe, Score: 90
Original: rc celta de vigo, Match: Celta Vigo, Score: 86
Original: real sociedad, Match: Real Sociedad, Score: 100
Original: rayo vallecano, Match: Rayo Vallecano, Score: 100
Original: cadiz cf, Match: Cadiz, Score: 90
Original: sevilla fc, Match: Sevilla, Score: 95
Original: ud las palmas, Match: Las Palmas, Score: 95
Original: athletic club, Match: Athletic Club, Score: 100
Original: real racing club, Match: Real Betis, Score: 86
Original: albacete balompie, Match: Albacete, Score: 90
Original: cd tenerife, Match: Tenerife, Score: 95
Original: racing club de ferrol, Match: Athletic Club, Score: 86
Original: burgos cf, Match: Burgos, Score: 90
Original: sd amorebieta, Match: Amorebieta, Score: 95
Original: real oviedo, Match: Oviedo, Score: 90
Original: sd huesca, Match: Huesca, Score: 90
Original: cd eldense, Match: Eldense, Score: 95
Original: real zaragoza, Match: Zaragoza, Score: 90
Original: cd leganes, Match: Leganes, Score: 95
Original: cd mirandes, Match: Mirandes, Score: 95
Original: fc andorra, Match: FC Andorra, Score: 100
Original: villarreal cf b, Match: Villarreal, Score: 90
Original: fc cartagena, Match: FC Cartagena, Score: 100
Original: elche cf, Match: Elche, Score: 90
Original: real sporting de gijon, Match: Real Sociedad, Score: 86
Original: rcd espanyol, Match: Espanyol, Score: 90
Original: levante ud, Match: Levante, Score: 95
Original: sd eibar, Match: Eibar, Score: 90
Original: ad alcorcon, Match: Alcorcon, Score: 95
Original: real valladolid cf, Match: Valladolid, Score: 90
Original: cd arentero, Match: Arenteiro, Score: 95
Original: ce sabadell fc, Match: Sabadell, Score: 90
```

concat_id	matched_name	score	full_name
La Liga-Real Madrid-Kepa Arrizabalaga	La Liga-Real Madrid-Kepa Arrizabalaga Revuelta	95	La Liga-Real Madrid-Kepa Arrizabalaga Revuelta
La Liga-Real Madrid-Andriy Lunin	La Liga-Real Madrid-Andrii Lunin	97	La Liga-Real Madrid-Andrii Lunin
La Liga-Real Madrid-Eder Militao	La Liga-Real Madrid-Eder Gabriel Militao	95	La Liga-Real Madrid-Eder Gabriel Militao
La Liga-Real Madrid-David Alaba	La Liga-Real Madrid-David Olatukunbo Alaba	95	La Liga-Real Madrid-David Olatukunbo Alaba
La Liga-Real Madrid-Antonio Rudiger	La Liga-Real Madrid-Antonio Rudiger	100	La Liga-Real Madrid-Antonio Rudiger
...
League One-Charlton-Panutche Camara	League One-Charlton-Panutche Amadu Pereira Camara	95	League One-Charlton-Panutche Amadu Pereira Camara
League One-Charlton-Louie Watson	League One-Charlton-Louie Shaun Watson	95	League One-Charlton-Louie Shaun Watson
League One-Charlton-Tyreece Campbell	League One-Charlton-Tyreece Campbell	100	League One-Charlton-Tyreece Campbell
League One-Charlton-Miles Leaburn	League One-Charlton-Miles Leaburn	100	League One-Charlton-Miles Leaburn
League One-Charlton-Daniel Kanu	League One-Charlton-Daniel Malanga Kanu	95	League One-Charlton-Daniel Malanga Kanu

Transfermarkt has lowercase club names while the API stats dataset has uppercase names. Soccer player names also differ significantly. Using Levenshtein distance, we are able to match both datasets. The score is the confidence threshold for making the match - We only considered matches > 90.



Data Processing: Machine Learning Lib.

Instead of making 1 model, we made 4, but why? Football is a specialized sport so different positions have different important features. Here is what we did:

GoalKeeper

```
assembler = VectorAssembler(inputCols=["player_age", "player_height", "player_weight", "player_injured", "minutes",  
                                         "captain", "goals_saves", "penalty_saved"], outputCol="goalie_features")  
  
GoalieFeatures_df = assembler.transform(GoalieFeatures_df)  
GoalieFeatures_df.limit(5).toPandas()
```



Attacker

```
assembler = VectorAssembler(inputCols=["player_age", "player_height", "player_weight", "player_injured", "minutes", "captain",  
                                         "shots_total", "shots_on", "goals_total", "goals_assists", "passes_total", "passes_key",  
                                         "duels_won", "dribbles_attempts", "dribbles_success", "fouls_drawn", "cards_yellow", "cards_red",  
                                         "penalty_scored", "penalty_missed"], outputCol="attacker_features")  
  
AttackerFeatures_df = assembler.transform(AttackerFeatures_df)  
AttackerFeatures_df.limit(5).toPandas()
```



Midfielder

```
assembler = VectorAssembler(inputCols=["player_age", "player_height", "player_weight", "player_injured", "minutes", "captain",  
                                         "shots_total", "shots_on", "goals_total", "goals_assists", "passes_total", "passes_key",  
                                         "tackles_total", "tackles_blocks", "tackles_interceptions", "duels_total", "duels_won",  
                                         "dribbles_attempts", "dribbles_success", "fouls_drawn", "fouls_committed",  
                                         "cards_yellow", "cards_yellowred", "cards_red", "penalty_scored", "penalty_missed"], outputCol="midfielder_features")  
  
MidfieldFeatures_df = assembler.transform(MidfieldFeatures_df)  
MidfieldFeatures_df.limit(5).toPandas()
```



Defender

```
assembler = VectorAssembler(inputCols=["player_age", "player_height", "player_weight", "player_injured", "minutes", "position_encoded", "captain",  
                                         "shots_total", "shots_on", "goals_total", "goals_conceded", "goals_assists", "passes_total", "passes_key",  
                                         "tackles_total", "tackles_blocks", "tackles_interceptions", "duels_total", "duels_won",  
                                         "fouls_drawn", "fouls_committed", "cards_yellow", "cards_red",  
                                         "penalty_scored", "penalty_missed"], outputCol="defender_features")  
  
DefenderFeatures_df = assembler.transform(DefenderFeatures_df)  
DefenderFeatures_df.limit(5).toPandas()
```



Data Processing: Insights :D

Let's practice with just the attacker's output, historical rating vs prediction

	player_name	team_name	league	attacker_features	rating	prediction	MarketValue	valuation	difference
0	C. Ngonge	Napoli	Serie A	[24.0, 179.0, 73.0, 0.0, 71.0, 0.0, 28.0, 14.0, 1.0, 2.0, 306.0, 21.0, 89.0, 62.0, 29.0, 22.0, 0.0, 0.0, 0.0, 0.0]	6.994736	7.998373	12000000	Undervalued	1.003637
1	J. Alvarez	Manchester City	Premier League	[24.0, 170.0, 71.0, 0.0, 2140.0, 0.0, 55.0, 30.0, 8.0, 7.0, 888.0, 60.0, 48.0, 33.0, 16.0, 10.0, 2.0, 0.0, 1.0, 0.0]	7.388888	7.816999	90000000	Undervalued	0.428111
2	M. Gregoritsch	SC Freiburg	Bundesliga 1	[30.0, 193.0, 91.0, 0.0, 1088.0, 0.0, 45.0, 21.0, 5.0, 3.0, 261.0, 11.0, 86.0, 9.0, 3.0, 12.0, 1.0, 0.0, 0.0, 0.0]	7.022727	7.376467	8000000	Undervalued	0.353740
3	A. Nemeth	Hamburger SV	Bundesliga 2	(22.0, 187.0, 85.0, 0.0, 297.0, 0.0, 12.0, 8.0, 0.0, 0.0, 57.0, 6.0, 23.0, 6.0, 3.0, 3.0, 0.0, 0.0, 0.0, 0.0)	6.457894	6.803342	1500000	Undervalued	0.345448
4	F. El Melali	Angers	Ligue 2	[27.0, 168.0, 65.0, 0.0, 1416.0, 0.0, 39.0, 17.0, 7.0, 2.0, 507.0, 12.0, 127.0, 92.0, 50.0, 45.0, 4.0, 0.0, 0.0, 0.0]	7.008333	7.332610	800000	Undervalued	0.324277
5	M. El Hankouri	FC Magdeburg	Bundesliga 2	[27.0, 176.0, 68.0, 0.0, 1002.0, 0.0, 20.0, 15.0, 3.0, 1.0, 628.0, 8.0, 52.0, 38.0, 22.0, 16.0, 4.0, 0.0, 1.0, 0.0]	7.057142	7.380216	800000	Undervalued	0.323074
6	B. Zivzividze	Karlsruher SC	Bundesliga 2	[30.0, 189.0, 85.0, 0.0, 850.0, 0.0, 31.0, 17.0, 8.0, 0.0, 183.0, 7.0, 36.0, 33.0, 8.0, 2.0, 0.0, 0.0, 0.0, 0.0]	6.791304	7.007419	700000	Undervalued	0.216115
7	K. Corredor	Rodez	Ligue 2	[24.0, 182.0, 72.0, 0.0, 2287.0, 0.0, 65.0, 28.0, 8.0, 5.0, 528.0, 24.0, 110.0, 63.0, 31.0, 34.0, 1.0, 0.0, 0.0, 0.0]	7.100000	7.287124	900000	Undervalued	0.187124
8	G. Scamacca	Atalanta	Serie A	[25.0, 196.0, 85.0, 0.0, 971.0, 0.0, 38.0, 18.0, 6.0, 3.0, 274.0, 13.0, 61.0, 20.0, 14.0, 19.0, 1.0, 0.0, 0.0, 0.0]	7.028571	7.211819	25000000	Undervalued	0.183248
9	E. Saad	FC St. Pauli	Bundesliga 2	[25.0, 185.0, 72.0, 0.0, 1763.0, 0.0, 41.0, 19.0, 7.0, 1.0, 575.0, 39.0, 124.0, 99.0, 53.0, 37.0, 5.0, 0.0, 0.0, 0.0]	7.200000	7.366064	2000000	Undervalued	0.166064
10	V. Muriqi	Mallorca	La Liga	[30.0, 194.0, 92.0, 0.0, 1596.0, 0.0, 39.0, 18.0, 5.0, 2.0, 443.0, 29.0, 157.0, 15.0, 8.0, 28.0, 2.0, 0.0, 1.0, 2.0]	7.185000	7.341115	15000000	Undervalued	0.156115
11	D. Berardi	Sassuolo	Serie A	[30.0, 183.0, 72.0, 0.0, 1437.0, 0.0, 38.0, 24.0, 9.0, 3.0, 559.0, 27.0, 65.0, 25.0, 11.0, 28.0, 6.0, 0.0, 5.0, 0.0]	7.382352	7.535386	12000000	Undervalued	0.153034
12	T. Ito	FC Magdeburg	Bundesliga 2	[27.0, 166.0, 59.0, 0.0, 603.0, 0.0, 18.0, 11.0, 2.0, 1.0, 325.0, 19.0, 41.0, 44.0, 26.0, 12.0, 2.0, 0.0, 0.0, 0.0]	7.023809	7.172910	700000	Undervalued	0.149101

Okay, in our model we predict with the current stats a given **rating**: if the predicted is bigger than the current, we say its **undervalued**. Let's imagine that we are on a budget, and we are a bigger team. Which player would you pick? The one that costs less! This is the importance of the **MarketValue** from Transfermarkt



Data Processing: Insights II :D

Let's practice with just the attacker's output, historical rating vs prediction

	player_name	team_name	league	attacker_features	rating	prediction	MarketValue	valuation	difference
0	C. Ngonge	Napoli	Serie A	[24.0, 179.0, 73.0, 0.0, 71.0, 0.0, 28.0, 14.0, 1.0, 2.0, 306.0, 21.0, 89.0, 62.0, 29.0, 22.0, 0.0, 0.0, 0.0, 0.0]	6.994736	7.998373	12000000	Undervalued	1.003637
1	J. Alvarez	Manchester City	Premier League	[24.0, 170.0, 71.0, 0.0, 2140.0, 0.0, 55.0, 30.0, 8.0, 7.0, 888.0, 60.0, 48.0, 33.0, 16.0, 10.0, 2.0, 0.0, 1.0, 0.0]	7.388888	7.816999	90000000	Undervalued	0.428111
2	M. Gregoritsch	SC Freiburg	Bundesliga 1	[30.0, 193.0, 91.0, 0.0, 1088.0, 0.0, 45.0, 21.0, 5.0, 3.0, 261.0, 11.0, 86.0, 9.0, 3.0, 12.0, 1.0, 0.0, 0.0, 0.0]	7.022727	7.376467	8000000	Undervalued	0.353740
3	A. Nemeth	Hamburger SV	Bundesliga 2	(22.0, 187.0, 85.0, 0.0, 297.0, 0.0, 12.0, 8.0, 0.0, 0.0, 57.0, 6.0, 23.0, 6.0, 3.0, 3.0, 0.0, 0.0, 0.0, 0.0)	6.457894	6.803342	1500000	Undervalued	0.345448
4	F. El Melali	Angers	Ligue 2	[27.0, 168.0, 65.0, 0.0, 1416.0, 0.0, 39.0, 17.0, 7.0, 2.0, 507.0, 12.0, 127.0, 92.0, 50.0, 45.0, 4.0, 0.0, 0.0, 0.0]	7.008333	7.332610	800000	Undervalued	0.324277
5	M. El Hankouri	FC Magdeburg	Bundesliga 2	[27.0, 176.0, 68.0, 0.0, 1002.0, 0.0, 20.0, 15.0, 3.0, 1.0, 628.0, 8.0, 52.0, 38.0, 22.0, 16.0, 4.0, 0.0, 1.0, 0.0]	7.057142	7.380216	800000	Undervalued	0.323074
6	B. Zivzividze	Karlsruher SC	Bundesliga 2	[30.0, 189.0, 85.0, 0.0, 850.0, 0.0, 31.0, 17.0, 8.0, 0.0, 183.0, 7.0, 36.0, 33.0, 8.0, 2.0, 0.0, 0.0, 0.0, 0.0]	6.791304	7.007419	700000	Undervalued	0.216115
7	K. Corredor	Rodez	Ligue 2	[24.0, 182.0, 72.0, 0.0, 2287.0, 0.0, 65.0, 28.0, 8.0, 5.0, 528.0, 24.0, 110.0, 63.0, 31.0, 34.0, 1.0, 0.0, 0.0, 0.0]	7.100000	7.287124	900000	Undervalued	0.187124

#28 Farid El Melali

Angers SCO
Ligue 2
League level: Second Tier
Joined: Aug 9, 2018
Contract expires: Jun 30, 2025

Date of birth/Age: Jul 13, 1997 (26) Height: 1,68 m Position: Left Winger Former International: Algeria
Place of birth: Béjaia Citizenship: Algeria

€800k

PREMIER LEAGUE MARKET VALUE UPDATE

PROFILE STATS MARKET VALUE TRANSFERS RUMORS NATIONAL TEAM NEWS ACHIEVEMENTS CAREER

Ligue 2 23/24 Statistics
Appearances: 24 Yellow Cards: 4
Goals: 7 Second Yellows: -
Assists: 2 Red cards: -

Starting eleven: 64% Minutes: 56% Goal participation: 22%

NEXT MATCHES
Ligue 2 - 29. Matchday Saturday, 03/16/2024 - 7:00 PM
Angers SCO vs Amiens SC

#12 Killian Corredor

Rodez AF
Ligue 2
League level: Second Tier
Joined: Jul 1, 2021
Contract expires: Jun 30, 2026

Date of birth/Age: Nov 4, 2000 (23) Height: 1,82 m Position: Centre-Forward
Place of birth: Montpellier Citizenship: France

€900k

PREMIER LEAGUE MARKET VALUE UPDATE

PROFILE STATS MARKET VALUE TRANSFERS RUMORS NEWS ACHIEVEMENTS CAREER

Ligue 2 23/24 Statistics
Appearances: 28 Yellow Cards: 1
Goals: 8 Second Yellows: -
Assists: 6 Red cards: -

Starting eleven: 89% Minutes: 91% Goal participation: 32%

NEXT MATCHES
Ligue 2 - 29. Matchday Saturday, 03/16/2024 - 7:00 PM
Rodez AF vs Grenoble

Project Results & Our Conclusions

- 1 A combination of data sources offers an insight performance that is simply unmatched
- 2 An API is not always available, when it comes to this, maybe the API is the website itself, use web-scraping!
- 3 Pandas is versatile but extremely slow when computing operations. Spark feels rigid and clunky but extremely fast.
- 4 Data is essential in all industries but in sports, everything is measurable and almost public. To any sports team out there, if you are not using data, you are missing out big time!
- 5 With a couple of easy models and valuations, we are able to build a product that could be sold to any football team looking to improve its scouting abilities!



For Further Reference

API documentation: <https://www.api-football.com/documentation-v3>

Fuzzy Wuzzy: <https://pypi.org/project/fuzzywuzzy/>

BS4: <https://pypi.org/project/beautifulsoup4/>



CONGRATS! WE ARE DONE!



THANKS FOR LISTENING :D

