



Scouting Potential Premier League Signings with Machine Learning



Aaron Kochman



Project Overview

- Project Question: How can teams better manage player inventory and always have replacement options with valuation in mind?
 - Can machine learning models be used to manage squads?
- Target League: English Premier League (EPL)
 - Most valued league in global football at \$10.63bn (<https://www.transfermarkt.us/wettbewerbe/europa>)
- Data Sources:
 - Transfermarkt, SoFIFA (EA Sports FIFA), Fantasy Premier League
- Prediction models:
 - Linear Regression: XGBoost, Sklearn

Business Case: Premier League Scouting

- Can machine learning models be used to produce a scouting report for team management?
 - Scout players with similar skills of existing players
 - Predict transfer values for scouted players
 - Recommend players to improve squad

Hypotheses and Prediction Models

Null Hypothesis (H_0): There is no relationship between player values and skill level.

Alternative Hypothesis (H_a): There is some relationship between player values and skill level.

Data Sources

#14 Pierre-Emerick Aubameyang 






Arsenal
Premier League
League level:  First Tier
Joined: Jan 31, 2018
Contract until: 30.06.2021

\$79.80m
Last update: Jun 13, 2019

Date of birth/Age: Jun 18, 1989 (30) Height: 1,87 m Current International:  Gabon
Place of birth:  Laval Position: Centre-Forward Caps/Goals: 51/20
Citizenship:  Gabon Agent: Relatives

- Data Sources:
 - Transfermarkt
 - SoFIFA (EA Sports FIFA)
 - Fantasy Premier League



P. Aubameyang (ID: 188567) 
Pierre-Emerick Aubameyang  ST LM Age 30 (Jun 18, 1989) 6'2" 176lbs

88 Overall Rating **88** Potential Value: €57M Wage: €205K

Pierre-Emerick Aubameyang
Forward
Arsenal



Form	GW 12	Total	Price	TSB
3.5	2pts	69pts	£11.0	24.9%

Influence	Creativity	Threat	ICT Index
337.0	177.6	464.0	97.7

Data Wrangling

- Data Sources:
 - Transfermarkt
 - Beautiful Soup HTML wrangling
 - SoFIFA (EA Sports FIFA)
 - CSV provided by Kaggle user stefanoleone992, web scraped (<https://www.kaggle.com/stefanoleone992/fifa-20-complete-player-dataset>)
 - Fantasy Premier League
 - Fantasy Premier League API request
 - JSON parsing

```
if __name__ == '__main__':
    scraper = PageScraper()
    soup = scraper(LEAGUES_URL)
    leagueTables = soup.find('table', class_='items').find('tbody')
    leagues = leagueTables.find_all('tr', href=re.compile('setbnewrb/[A-Z]{2}1'), title=re.compile('W'))
    leagues = leagues[:N_LEAGUES]
    leagueUrlDic = { league.text : BASE_URL + league['href'] for league in leagues}
    leaguesData = []
    for leagueName, leagueUrl in leagueUrlDic.items():
        print( "Scraping the %s..." % leagueName)
        leaguesData.append( League( leagueName, leagueUrl, scraper))

    #flattening all players information to pandas.DataFrame and exporting to csv
    playerProfiles = [player.PlayerData for league in leaguesData for team in league.TeamsData for player in team.PlayersData]
    df = pd.DataFrame( PlayerProfiles)
    df.to_csv('transfer.csv', index=False)

Scraping the Premier League...
['17/18', 'Jul 1, 2017', 'Benfica', 'Man City', '22,00 mil. €', '40,00 mil. €']
['15/16', 'Jul 1, 2015', 'Rio Ave FC', 'Benfica', '1,20 mil. €', '500 K €']
['12/13', 'Jul 1, 2012', 'GD Ribeirão', 'Rio Ave FC', 0, 'Free transfer']
['11/12', 'Jul 1, 2011', 'Benfica U19', 'GD Ribeirão', 0, 'Free transfer']
['10/11', 'Jul 1, 2010', 'Benfica U17', 'Benfica U19', 0, 0]
['09/10', 'Jan 1, 2010', 'São Paulo U17', 'Benfica U17', 0, '?']
Ederson done
```

```
▼ root: {} 8 keys
▶ events: [] 38 items
▶ game_settings: {} 22 keys
▶ phases: [] 11 items
▶ teams: [] 20 items
total_players: 6801617
▼ elements: [] 541 items
```

Data Compiling and Matching Player Names with FuzzyWuzzy

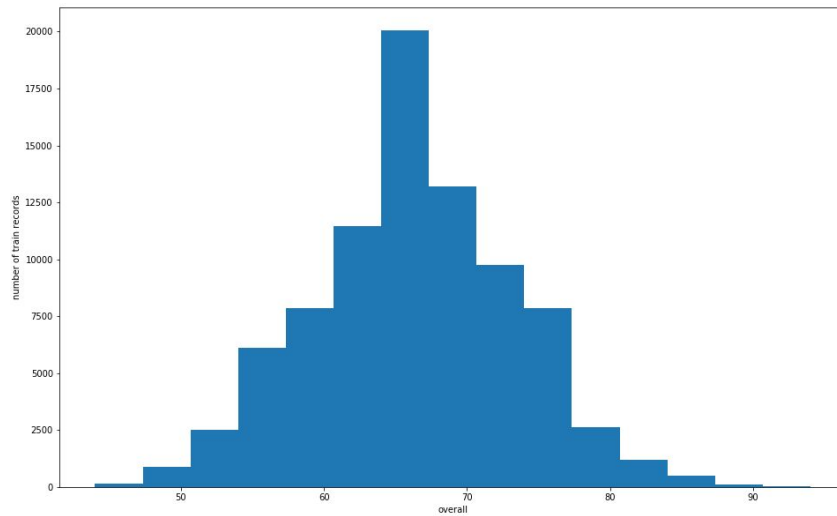
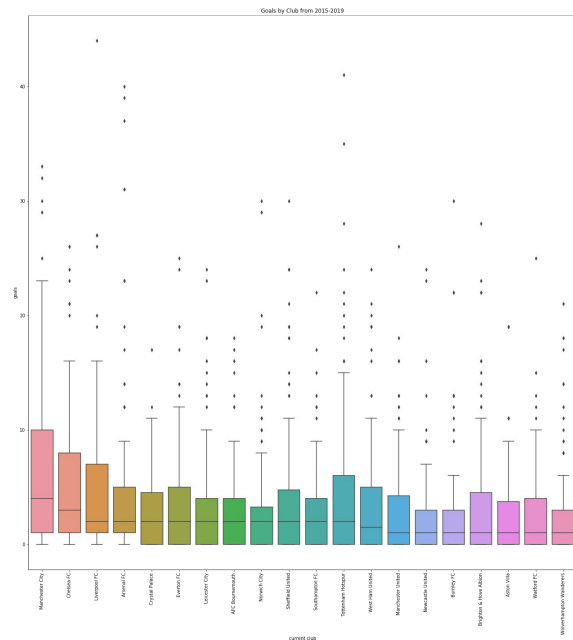
```
# List for dicts for easy dataframe creation
dict_list = []
# iterating over our players without salaries found above
for name in df_fifa.short_name:
    # Use our method to find best match, we can set a threshold here
    match = match_name(name, df_prem_field_players.name, 60)

    # New dict for storing data
    dict_ = {}
    dict_.update({"fifa_name" : name})
    dict_.update({"transfermarkt_name" : match[0]})
    dict_.update({"score" : match[1]})
    dict_list.append(dict_)

merge_table = pd.DataFrame(dict_list)
# Display results
merge_table.head()
```

	fifa_name	transfermarkt_name	score
0	K. De Bruyne	Kevin De Bruyne	81
1	V. van Dijk	Virgil van Dijk	77
2	M. Salah	Mohamed Salah	67
3	H. Kane	Harry Kane	71
4	Alisson		-1

Exploratory Analysis



Exploratory Analysis

