SOCCER TEAM DATA ANALYTICS

Prepared by:

Team PANDA

ARINJAY JAIN(A20447307)

AYUSH DADHICH(A20449379)

DIVYANI AUDICHYA(A20443049)

MRINALINI(A20348759)

PARTH GUPTA(A20449774)

CSP 571

DATA PREPARATION AND ANALYSIS

PROF. ADAM MCELHINNEY

MAY 3rd 2020

Abstract

The main focus of this paper is to showcase the soccer analysis[3] done by leveraging two data sets. The most fan clubs are associated with soccer and there's a huge revenue made in this area of sports(link). If a manager wants to buy out a European league team, within a given budget, we try to help this new manager invest his money with the best possible team along with five swaps at max to make the team even better. We filtered out teams having an overall average rating above 75% from the primary dataset(https://sofifa.com/), which is the qualifying criteria. We picked the best possible team in the given budget. To further make sure the team with the best overall ratings really has the best players, we use the remaining amount to swap our players with the better ones from the other teams. We consider a few features variables from the secondary data set to improve the model efficiency.

Introduction

Let's assume that you are a millionaire and are looking to make some money by investing into the European league team. But you have a restricted budget for buying a team of 11-15 players and it must be one of the best teams within this budget. We decided that the team with the best overall rating should be the one. Following are some constraints by the organization conducting the match:

1. The teams which qualify will have an overall team rating which is equal to 75% or higher.
2. Once the team with best overall ratings is filtered out, the top team which can be bought in the given budget will be your team. After this, we can perform at most 5 swaps. Here with the remaining amount after the team selection , we will compare the stats of each player in our team with the ones in the filtered-out teams, the players with better ratings will be swapped until the remaining budget is exhausted. And you got your final team.

More details on the problem statement are explained in section 1.1.
Section 1.2 explains a systematic data collection procedure which was carried out to get some data sets to leverage the features from as this will help us decrease the RMSE value of our prediction model.
Our primary data source was Player Statistics Dataset from https://sofifa.com/ 2009 - 2019. It consists of 18207 rows and 89 columns for the 2019 season. It includes unique id, Country, Club, Overall Rating, Value, Wage, Position, Leagues, Skill moves, Attacking, Power, etc.

The Key Performance Indicators(**KPIs**) of our models are Root Mean Square Error (RMSE), R-squared and Mean Square Error (MSE).
As we are looking for minimum RMSE value for the regression models, hence we decided to use some supplementary source:- https://www.kaggle.com/hugomathien/soccer. Section 1.3 will elaborate how we prepared the data from these sources to make it fit for the usage in the problem statement.
Our purpose is to develop an optimized team from the raw data.

Section 1.4 will further explain the exploratory data analysis right from the very beginning and shows some figures including plots and tables for the data analysis.
Once we have analyzed our data, we start with the data modelling. Here we use various modeling and analytical techniques to predict Player's overall rating, which is explained very thoroughly in Section 1.5.

Finally, after applying the best modelling techniques, the results as shown in Section 1.6 will explain how it finally turns out to be the best predictive model and this whole model is deployed on the web app using the Django framework as shown in Section 1.7. It contains snapshots of the demo which shows predicted overall rating of a particular player.

Like any other project, this too has some limitations and hence the future scope is described in the Section 1.8.

Keywords: *soccer, performance analysis, overall rating prediction, team optimization*

1.1 Problem Statement:

The objective of our algorithm is to maximize the team's overall statistics by accumulating players with the highest 'Ratings' within the constrained budget.

1. The 'sofifa' dataset has extensive data. The first step includes gathering the required domain knowledge of the dataset
2. Perform sanity checks to ensure the data falls in the expected range of values. These ranges will be defined by our knowledge about the sport.
3. We predict 'Overall Rating' of the players for the year 2020 based on 10 years of data from the primary dataset.
4. Then we predict 'Overall Rating' for 2020 by Joining Date of Birth (Age) column from our secondary dataset with our Primary Dataset.
5. We compare the model in 4 and 5 based on Regression Algorithms: Decision Trees Regressor, SVM (SVR), Linear Regression and Random Forest Regressor.
6. We use the best model with the minimum RMSE value to form our dataset for Optimization Algorithm[8].
7. We build a proprietary optimization algorithm to tackle our problem statement and predict the best team in the given budget.

Constraints and Assumptions:
i)Budget does not exceed 'x million' Euros
ii)Make sure I have 11-15 players
iii)Make sure I have 1-2 goalkeepers
iv)Make sure I have 4-5 defenders
v)Make sure I have 4-5 midfielders
vi)Make sure I have 2-3 strikers

- The final evaluation metric is Overall Team rating. We utilized the 2009 - 2019 FIFA dataset for prediction of the player's 'Overall Rating' for 2020.
- The player pool is divided into 15 domains, which are:
    1. ST -> Striker
    2. RW -> Right winger (a more attacking option than a right midfielder)
    3. LWB -> Left Wing Back
    4. RWB -> Right Wing Back
    5. LB -> Left back
    6. CF -> Center forward
    7. CDM -> Central defensive midfielder
    8. CB -> Centre back

9. CAM -> Central attacking midfielder
10. RM -> Right midfielder
11. CM -> Centre midfielder
12. LW -> Left winger (typically more attack-minded than a left midfielder)
13. LM -> Left midfielder
14. RB -> Right back
15. GK -> Goalkeeper

- As the team is created, we use an additional metric 'Age' from our secondary dataset to further evaluate its effect on the overall rating.

Specific Player Position(s)

Figure 1: Player's positions in Soccer game

(Source: https://images.app.goo.gl/GJHogKgaMs3Eerpe9)

Why Soccer and why does it need analysis?

- Resources such as capital are an essential constraint in businesses (sports) and thus, developing a team with best possible players is a necessity for a new team manager / club.
- After carefully analyzing different sports in terms of their global fanbase, TV rights and deals, popularity of the sport in different parts of the world, etc. we observed Soccer has the most potential to grow in terms of investment and returns.
- A value of 1.85 billion dollars was estimated in the 2018 world cup alone, with a viewership of 3.5 billion compared to other sports like cricket with 2.5 billion and basketball with 2.2 billion fans watching all over the world.

https://www.ft.com/content/84aa8b5e-c1a9-11e8-84cd-9e601db069b8

Numbers are not entirely new in the sport: for decades commentators have painstakingly compiled statistics on everything from winning streaks to the most crosses ever delivered in one match. But over the past decade a far more scientific operation has emerged, changing not only teams' results but also how money is deployed on recruiting new talent.
Clubs can now draw up a shortlist of players whose statistics match the profile of their ideal target signing, all without leaving the training ground. Scouts can then assess matches and video footage of a smaller pool of players, saving time and money.

One company in the recruitment field is the 21st Club. The consultant's tool calculates the historical link between players' actions on the pitch and their team's overall performance level and assigns each player a rating. Clubs can use the data to see whether a player would strengthen, weaken or make little difference to their team's overall performance level.

This is what we use exactly, we leveraged the *overall performance level* to predict the future team's overall performance level, which is 2020.

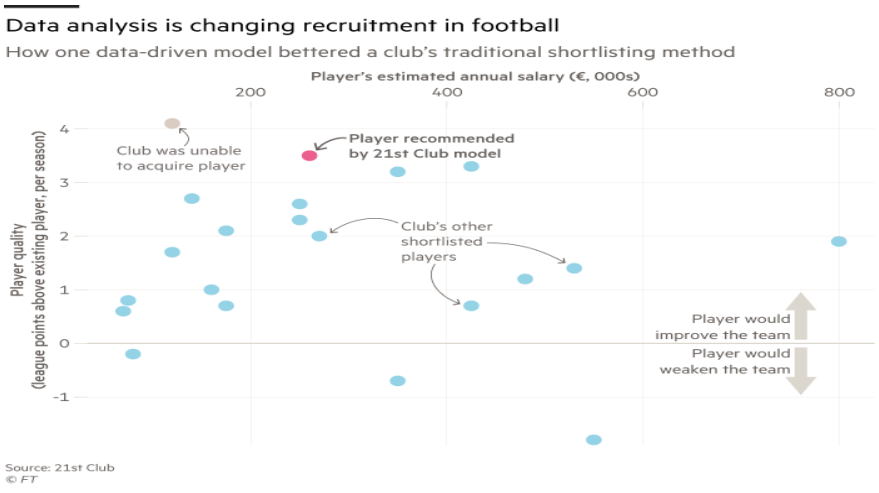

Figure 2: Player's estimated annual salary vs player quality

What Are The Most Popular Sports In The World?

Rank	Sport	Estimated Global Following	Primary Sphere of Influence
1	Soccer (Association Football)	4.0 Billion	Globally
2	Cricket	2.5 Billion	UK and Commonwealth
3	Field Hockey	2 Billion	Europe, Africa, Asia, and Australia
4	Tennis	1 Billion	Globally
5	Volleyball	900 Million	Western Europe and North America
6	Table Tennis	875 Million	Globally
7	Basketball	825 Million	Globally
8	Baseball	500 Million	United States, Caribbean, and Japan
9	Rugby	475 Million	UK and Commonwealth
10	Golf	450 Million	Western Europe, East Asia, and North America

Figure 3: Showing the comparison of various popular sports(Source: www.worldatlas.com)

1.2 Data Acquisition:

As we need to minimize the RMSE value of the model, hence we decided to use some supplementary source: https://www.kaggle.com/hugomathien/soccer

What Data Sources are we using?

Dataset:

- We identify the data from soFIFA(primary dataset) and Kaggle(secondary dataset) which includes more than 100,000 rows of players (before pre-processing) with 15 prediction variables and a target variable of overall rating from years 2009-2019.

Web Scraping:

- We start by web scraping the data from html tables from our primary datasource, using-beautifulsoup4, BS4, HTML5lib, requests, URLlib3, webencoddings.
- Each predictor variable (columns) are then stored into an array.
- This array is converted into a dataframe and the data frame is then pushed into the .csv files.

Primary Data Source:

Player Statistics Dataset from soFIFA 2009 - 2019. Data set consisting of 18207 rows and 89columns for the 2019 season. This dataset includes unique id, Country, Club, Name, Overall Rating, Value, Wage, Position,Leagues, Skill moves, Attacking, Poweretc.

Secondary Data Source:

European Soccer League Dataset on Kaggle. We are using the Date of Birth column to calculate current 'age' of players. We want to record the effects of age on a player's statistics and draw conclusions if a player performs better with age. Here we are assuming that more age means more playing experience. We found an interesting correlation with age to our data, i.e., player performance and overall stats increase with increase in age, however after a certain age(e.g.,35), we see our player's performance/stats declining. This makes sense because human bodies start to break down after a certain age and no amount of experience can compensate for the decline in physical conditions.

Note On why we choose this secondary dataset:We tried to get Height/Weight from the secondary dataset and calculate BMI. But our models did not work well w.r.t. this. It's probably because in the small range of height and weight there are too many entries to make a reliable linear model. (Over-fitting)

We were considering Preferred foot(Left vs Right), but this also shows the same problems as above.

We have other columns: crossing, short_passing, volleys, dribbling, curve, free_kick_accuracy, sprint_speed, agility, shot_power, jumping, stamina, strength etc.
However, these are abnormally skewed for Goal Keeper data. Which brings our Accuracy down.

We also tried to find weather data, related to player stats, but we could not find anything substantial.

So, we thought of evaluating experience(age) from this data set.

1.3 Data Understanding, Cleaning and Preparation[1]:

Predictor Variables

1. Age
2. Name
3. Team
4. Best Overall
5. Position
6. Attacking stats
7. Movement stats
8. Goalkeeper stats
9. Power stats
10. Defense stats
11. Skill sets
12. Total potential
13. market value
14. Wage
15. Predicted overall rating
16. Actual overall rating

Used for the model: [["Age", "Best Overall", "Attacking stats", "Movement stats", "Power stats", "Defense stats", "Skill sets", "Total potential", "Market_value", "Wage"]]

Various approaches for data cleaning

- We first read the CSV files consisting of all data from years, 2009-2019.
- Duplicate Entries: Removing Duplicates by Key - Player ID.
- Missing Values: Drop unnamed columns

- Units- ‘Wage’, ‘market_value’ columns have units in M (10,00000) and K (1,000). We converted from alpha-numeric to numeric values.
- Alpha Numeric Values: Remove Columns such as ‘Team’, ‘Name’, ‘Position’
- Table Joins (Primary and Secondary dataset)- By Player ID
- We dropped the rows which had ‘market_value’=0 and ‘Wage’=0

1.4 Exploratory Data Analysis[2]

External data sources:

https://sofifa.com/?hl=en-US&r=190075&set=true

https://www.kaggle.com/hugomathien/soccer

Measures precisely in business terms the value of these data sets to their business problem.

Secondary Data Set

How did we leverage it?

1. We used the ‘player‘ table from the database present in the link: **https://www.kaggle.com/hugomathien/soccer**
2. From the player table, we used the ‘birthday’ column to calculate the age.
3. We joined the primary and secondary data sets on the ‘id’ column using the inner join.

That’s how we leveraged the secondary data set in our problem statement.

Data Analysis:

- This part includes plotting the data that we cleaned, before prediction, to analyse and make better predictions on our data.
- Some of the plots include:
    - Age vs Overall Rating
    - Age vs Avg Skill sets
    - Age vs Power Stats
    - Heatmaps, etc.

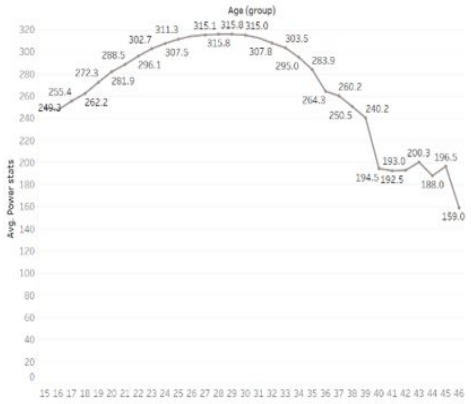

Fig 4: Age vs Avg. Power stats

Fig 5: Age vs Avg. Skill sets

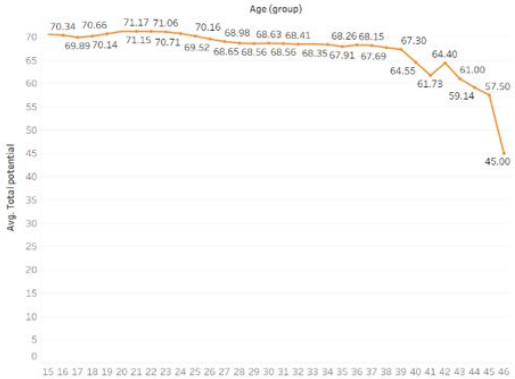

Fig 6: Age vs Avg. Total potential

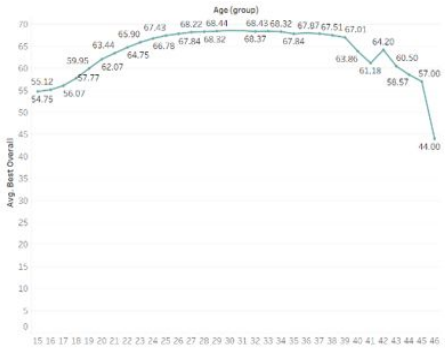

Fig 7: Age vs Avg. Best Overall

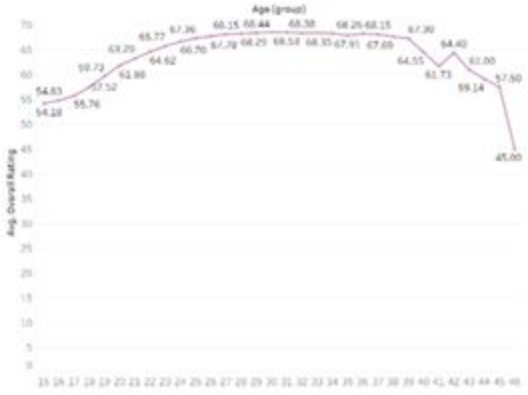

Fig 8: Age vs Avg. Overall Rating

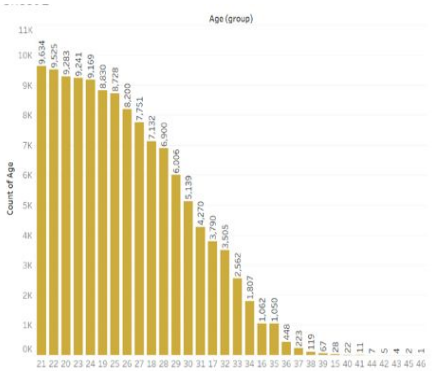

Fig 9: Age vs Count of Age

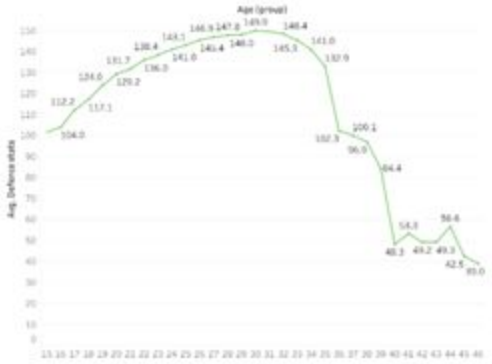

Fig 10: Age vs Avg. Defense stats

CONCLUSION:

1. From the above plots, we observed that age can be an important factor for our model.

VARIABLE SELECTION

Correlation matrix:
We used the correlation matrix for the variable selection for our model. We are considering p-value = 0.5 as threshold.

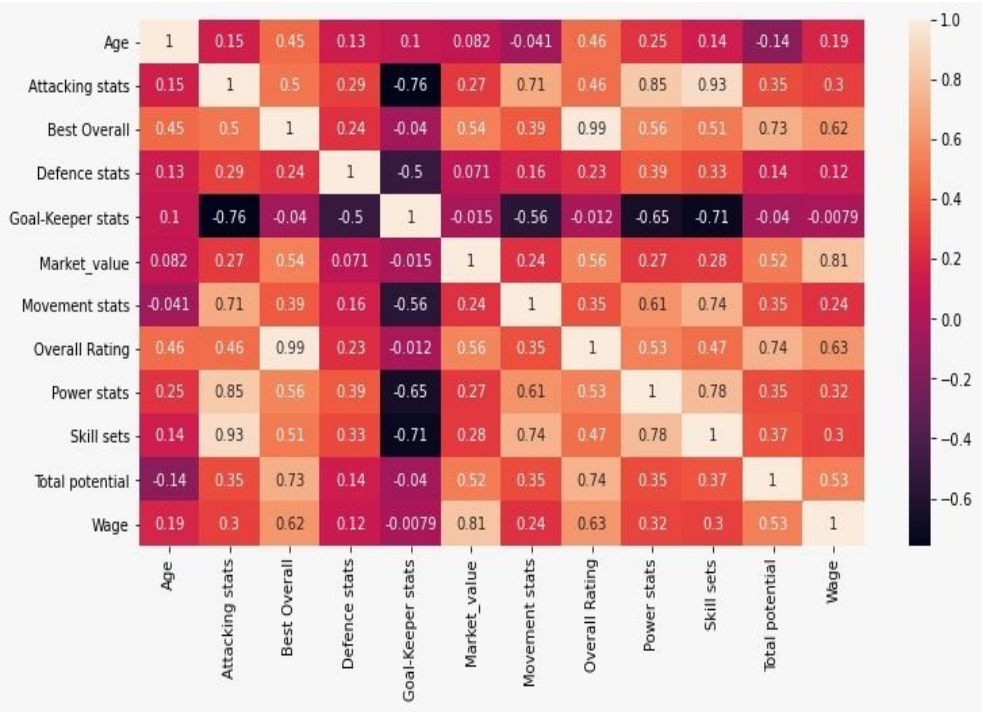

Fig 11: Heatmap for correlation matrix

Apart from the observed analysis from the plots above, we can also observe a strong correlation between 'Age' and 'Overall Rating' which is 0.46. So we used Age from the secondary dataset. Here, we can observe, the correlation between the column 'Goal-Keeper stats' and target variable 'Overall Rating' is -0.012, which is very low, so we dropped the 'Goal-Keeper stats' column, and we will not consider that in the model.

1.5 Data Modelling[4]:

The modeling and analytical techniques to predict Player's overall stats:

Random Forest:

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees.[1][2] Random decision forests correct for decision trees' habit of overfitting to their training set.

Pros:

- The predictive performance can compete with the best supervised learning algorithms
- They provide a reliable feature importance estimate
- They offer efficient estimates of the test error without incurring the cost of repeated model training associated with cross-validation.

Cons:

- An ensemble model is inherently less interpretable than an individual decision tree
- Training many deep trees can have high computational costs (but can be parallelized) and use a lot of memory
- Predictions are slower, which may create challenges for applications.

Support vector machine (SVM):

In machine learning, **support-vector machines** (**SVMs**, also **support-vector networks**[1]) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis.

Pros:

- It works really well with a clear margin of separation
- It is effective in high dimensional spaces.
- It is effective in cases where the number of dimensions is greater than the number of samples.
- It uses a subset of training points in the decision function (called support vectors), so it is also memory efficient.

Cons:

- It doesn't perform well when we have large data set because the required training time is higher
- It also doesn't perform very well, when the data set has more noise i.e. target classes are overlapping
- SVM doesn't directly provide probability estimates, these are calculated using an expensive five-fold cross-validation. It is included in the related SVC method of Python scikit-learn library.

Decision Tree:

A **decision tree** is a decision support tool that uses a tree-like model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility. It is one way to display an algorithm that only contains conditional control statements.

Pros:

- Easy to understand and interpret, perfect for visual representation. This is an example of a white box model, which closely mimics the human decision-making process.
- Can work with numerical and categorical features.
- Requires little data preprocessing: no need for one-hot encoding, dummy variables, and so on.

- Non-parametric model: no assumptions about the shape of data.
- Fast for inference.
- Feature selection happens automatically: unimportant features will not influence the result. The presence of features that depend on each other (multicollinearity) also doesn't affect the quality.

Cons:

It tends to overfit. This usually can be mitigated in one of three ways:

- Limiting tree depth

Linear Regression:

In statistics, linear regression is a linear approach to modeling the relationship between a scalar response and one or more explanatory variables. The case of one explanatory variable is called simple linear regression. For more than one explanatory variable, the process is called multiple linear regression.

Pros:

- least complex compared to other algorithms that also try finding the relationship between independent and dependent variables.

Cons:

- In real life, there aren't many problems in the world that exhibit a clear relationship between the independent and dependent variables.
- linear regression most of the time can be only used when we deal with relationships that graphically look like a line because "linear" means according to the mathematical graphical definition is a straight line.
- Outliers are others that make linear regression more limited in terms of its use because linear regression always considers the case that tends to be the most frequent.

HYPER PARAMETER TUNING[5]

For Hyperparameter tuning, we used 'RandomizedSearchCV' for our problem statement. After using this, we found out that the default parameters of models: 'Random Forest', 'Decision Tree', 'SVM(SVR)', 'Linear Regression' gives minimum RMSE for our problem.
So, we used the default parameters.

Random forest in our case:
We observed an RMSE of 0.385452 using Random Forest which is the minimum among RMSE values of all the models and thus used the predicted rating from the random forest model.
Here, we are considering minimum RMSE value and maximum R-square value for our models because minimum RMSE and maximum R-square values are best for any regression problem.

	Model	RMSE	R-square	MSE
0	Decision Tree	0.684665	0.990329	0.468766
1	Random Forest	0.457672	0.995679	0.209464
2	Support vector machine(SVM)	2.400238	0.881143	5.761140
3	Linear Regression	0.924370	0.982372	0.854460

Fig 12: Table Showing the comparison of metrics between each model without Age

	Model	RMSE	R-square	MSE
0	Decision Tree	0.570402	0.993288	0.325358
1	Random Forest	0.385452	0.996935	0.148574
2	Support vector machine(SVM)	2.400665	0.881101	5.763193
3	Linear Regression	0.836743	0.985556	0.700139

Fig 13: Table Showing the comparison of metrics between each model with Age

1.6 Results:

- We utilized Python and Tableau for visualisation of our project.
- Our focus was to optimize highest rated team within a constrained budget, plots such as:
    - Team total value vs team
    - Average Predicted overall rating vs teams
    - Domain specific plots, such as:
        - Striker vs team
        - Goal-keeper vs team
        - centre-midfielder vs team

- Player specific plots vs yearly performance

ID	Age	Name	Team	Best Overall	Position	Attacking stats	Movement stats	Goal-Keeper stats	Power stats	Defence stats	Skill sets	Total potential	Market_value	Wage	Predicted Overall Rating	Actual Overall Rating
239085	18	E. Háland	Borussia Dortmund	78	ST	342	385	52	382	84	329	90	20000000	32000	79	79
245541	16	G. Reyna	Borussia Dortmund	62	RW	254	371	56	263	67	310	85	550000	1000	62	61
251573	21	Renan Lodi	Atlético Madrid	79	LWB	346	413	55	372	225	373	87	16000000	36000	79	79
233049	19	J. Sancho	Borussia Dortmund	85	RW	363	439	52	320	105	368	93	55000000	66000	84	85
235212	20	A. Hakimi	Borussia Dortmund	81	RWB	348	423	46	373	228	355	87	22000000	54000	82	81
242641	18	R. Aït Nouri	Angers SCO	69	LB	275	352	60	247	211	295	85	3000000	3000	70	70

Fig 14: Table showing Overall Predicted Ratings

Optimization Algorithm

- 13 teams are found to have a rating of 75 and higher.We consider only these 13 teams for our player pool[6].
- The manager of the team acquires the highest rated team with a budget constraint of 30M euros.
- First, we obtain the predicted (average) overall rating for the players.
- Then we calculate overall (predicted) team rating , by summing the predicted overall rating of all players in one team and dividing it by the total number of players in the team mentioned. (Average)
- As we can see in the next fig, the 9th team “Inter” falls in this budget.

Team	Market_value	Predicted Overall Rating
Manchester City	626000000	86.153846
Juventus	346000000	84.181818
FC Barcelona	479000000	84.000000
Real Madrid	493725000	83.846154
Liverpool	550400000	83.250000
Paris Saint-Germain	426650000	83.000000
Napoli	315000000	82.636364
Manchester United	323000000	81.583333
Inter	249000000	81.272727
Lazio	267000000	80.384615
FC Porto	171000000	79.750000
Roma	175000000	79.416667
Milan	163000000	78.727273

```
My team-'Inter' before swap.
```

Fig 15: A table showing Team Standing Before Swapping

ID	Age	Name	Team	Best Overall	Position	Attacking stats	Movement stats	Goal-Keeper stats	Power stats	Defence stats	Skill sets	Total potential	Market_value	Wage	Predicted Overall Rating	Actual Overall Rating
192505	26	R. Lukaku	Inter	83	ST	400	338	54	402	90	369	87	53000000	125000	85	86
152908	33	A. Young	Inter	74	RWB	343	365	72	330	220	372	75	4000000	50000	75	75
172962	28	V. Moses	Inter	76	RM	353	387	56	358	219	369	78	10000000	62000	78	78
198946	30	D. D'Ambrosio	Inter	76	RB	322	365	62	344	236	332	77	7000000	51000	77	77
201389	26	C. Biraghi	Inter	75	LWB	304	369	49	361	225	351	77	8000000	47000	76	76
162835	34	S. Handanović	Inter	86	GK	80	272	420	261	48	88	88	26000000	110000	88	88
216352	26	M. Brozović	Inter	81	CM	355	349	50	390	236	401	84	24000000	73000	82	82
212823	22	Gabriel Barbosa	Inter	80	CF	384	409	44	367	76	361	85	23000000	70000	81	81
212153	25	R. Gagliardini	Inter	77	CDM	337	319	42	365	233	333	81	12000000	51000	78	78
182493	33	D. Godín	Inter	85	CB	307	308	49	352	263	295	87	24000000	110000	86	87
190460	27	C. Eriksen	Inter	85	CAM	385	392	43	372	138	432	88	58000000	125000	87	87

Fig: Table showing the eleven players of Team 'Inter' before swapping

Player Swaps

- The remaining 19M Euros from our overall budget are used for swapping players with other teams in the pool of 13 teams.
- First we compare the overall rating of players of each domain in our to the remaining best players in each domain from 12 teams.
- After comparing, we pick the domain player whose difference is highest because it will increase the average overall rating of the team.
- After picking the player from the above step, we check for our remaining budget[7] if it is within our budget range then we will swap otherwise not.
- This is how players from the picked domain are swapped until the budget is over or the total swap gets exhausted.
- Only 5 swaps are allowed if we don't run out of budget.
- As we can see, our algorithm terminated after 4 swaps because we ran out of budget.

```
True
(212153, R. Gagliardini, Inter) <----> (200145, Casemiro, Real Madrid)
remaining = 139000000
True
(201389, C. Biraghi, Inter) <----> (189332, Jordi Alba, FC Barcelona)
remaining = 112000000
True
(152908, A. Young, Inter) <----> (210514, João Cancelo, Manchester City)
remaining = 87000000
True
(212823, Gabriel Barbosa, Inter) <----> (208722, S. Mané, Liverpool)
remaining = 30000000
False
```

Fig 16: Output of Player Swaps

ID	Age	Name	Team	Best Overall	Position	Attacking stats	Movement stats	Goal-Keeper stats	Power stats	Defence stats	Skill sets	Total potential	Market_value	Wage	Predicted Overall Rating	Actual Overall Rating
192505	26	R. Lukaku	Inter	83	ST	400	338	54	402	90	369	87	53000000	125000	85	86
172962	28	V. Moses	Inter	76	RM	353	387	56	358	219	369	78	10000000	62000	78	78
198946	30	D. D'Ambrosio	Inter	76	RB	322	365	62	344	236	332	77	7000000	51000	77	77
162835	34	S. Handanović	Inter	86	GK	80	272	420	261	48	88	88	26000000	110000	88	88
216352	26	M. Brozović	Inter	81	CM	355	349	50	390	236	401	84	24000000	73000	82	82
182493	33	D. Godín	Inter	85	CB	307	308	49	352	263	295	87	24000000	110000	87	87
190460	27	C. Eriksen	Inter	85	CAM	385	392	43	372	138	432	88	58000000	125000	87	87
200145	27	Casemiro	Inter	86	CDM	349	339	67	436	259	368	90	63000000	280000	89	88
189332	30	Jordi Alba	Inter	84	LWB	373	437	60	362	239	388	86	35000000	220000	86	86
210514	25	João Cancelo	Inter	83	RWB	353	415	58	372	238	395	87	29000000	135000	84	83
208722	27	S. Mané	Inter	89	CF	410	460	56	406	122	391	90	80000000	240000	90	90

Fig 17: Team 'Inter' after swapping

Team	Market_value	Predicted Overall Rating
Manchester City	601000000	85.461538
Inter	409000000	84.818182
Juventus	346000000	84.181818
FC Barcelona	452000000	83.166667
Paris Saint-Germain	426650000	83.000000
Real Madrid	442725000	83.000000
Napoli	315000000	82.636364
Liverpool	493400000	82.500000
Manchester United	323000000	81.583333
Lazio	267000000	80.384615
FC Porto	171000000	79.750000
Roma	175000000	79.416667
Milan	163000000	78.727273

Fig 18: Team's Standings After Swapping

The figure above Fig shows that Our team "Inter" is now in 2nd position in overall team ratings list.

Domain-wise Team Comparison:

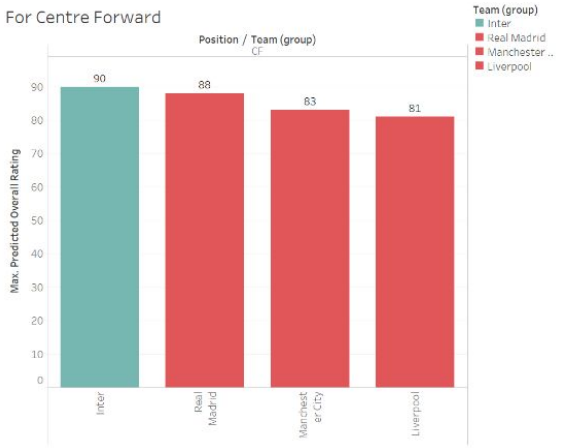

Fig 19: Centre forward position/team vs Max. Predicted Overall Rating

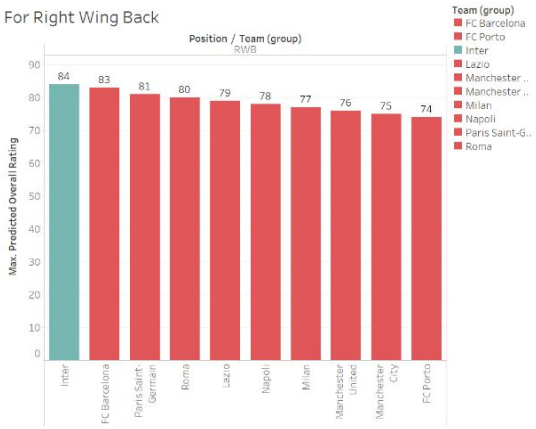

Fig 20: Right Wing Back position/team vs Max. Predicted Overall Rating

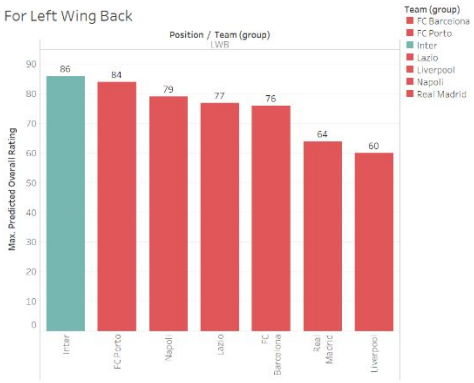

Fig 21: Left Wing Back position/team vs Max. Predicted Overall Rating

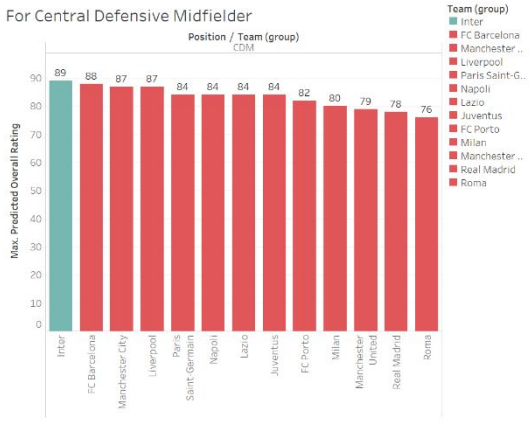

Fig 22: Central Defensive Midfielder position/team vs Max. Predicted Overall Rating

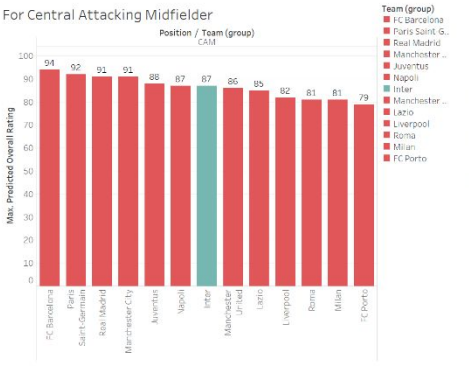

Fig 23: Central Attacking Midfielder position/team vs Max. Predicted Overall Rating

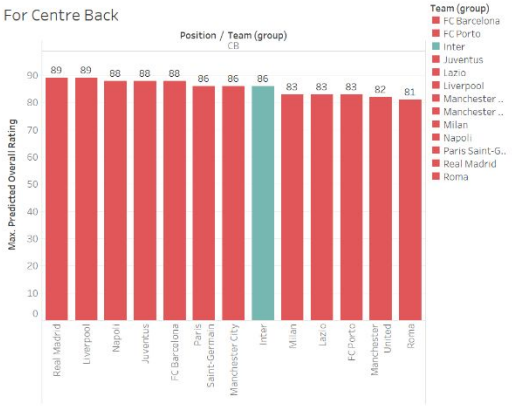

Fig 24: Centre back position/team vs Max. Predicted Overall Rating

Fig 25: Center Midfielder position/team vs Max. Predicted Overall Rating

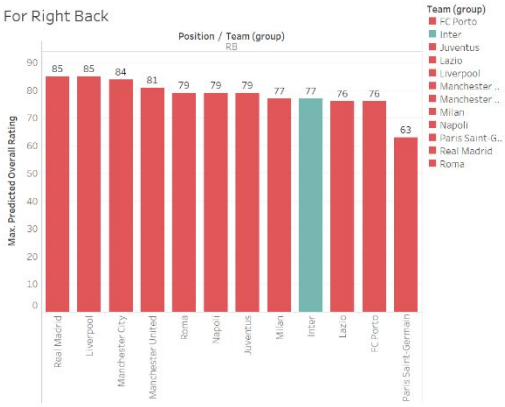

Fig 26: Right Back position/team vs Max. Predicted Overall Rating

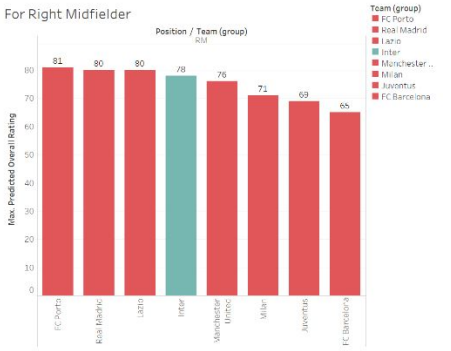

Fig 27: Right Midfielder position/team vs Max. Predicted Overall Rating

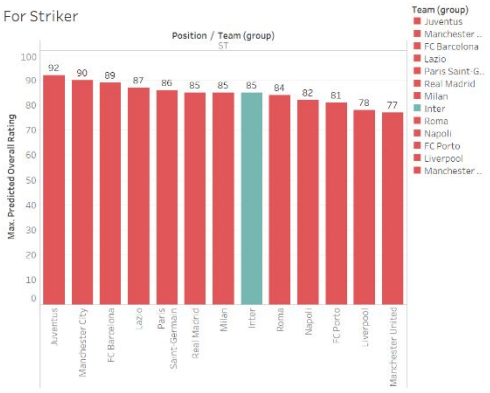

Fig 28: Striker position/team vs Max. Predicted Overall Rating

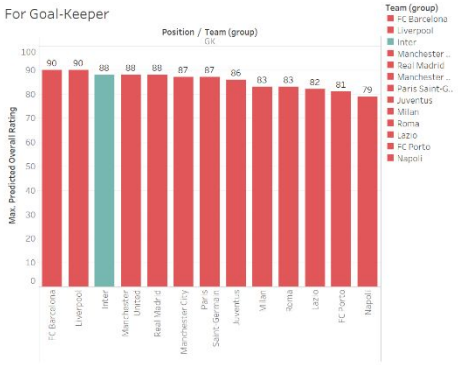

Fig 29: Goalkeeper position/team vs Max. Predicted Overall Rating

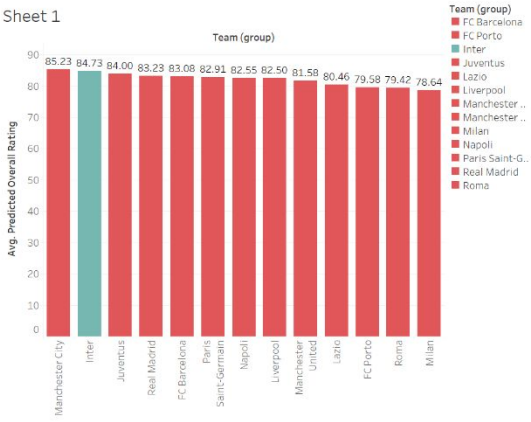

Fig 30: Team vs Avg. Predicted Overall Rating

‘Inter’ player’s past performance:

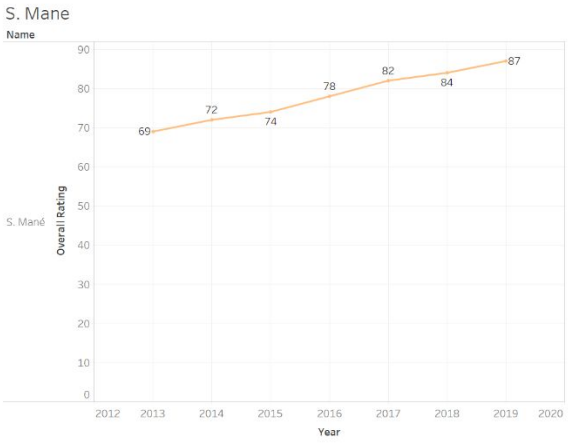

Fig 31: Graph of S. Mane’s Overall Rating vs Year

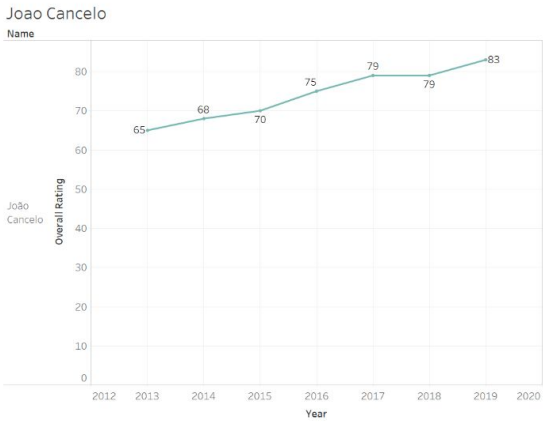

Fig 32: Graph of Joao Cancelo’s Overall Rating vs Year

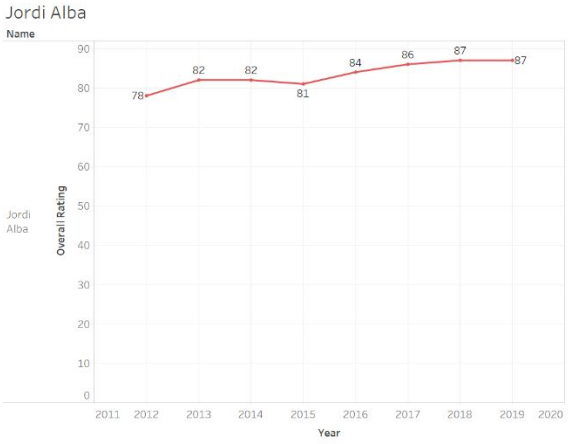

Fig 33: Graph of Jordi Alba’s Overall Rating vs Year

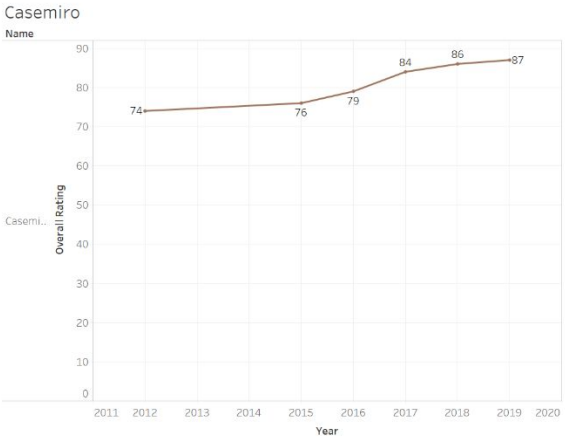

Fig 34: Graph of Casemiro Overall Rating vs Year

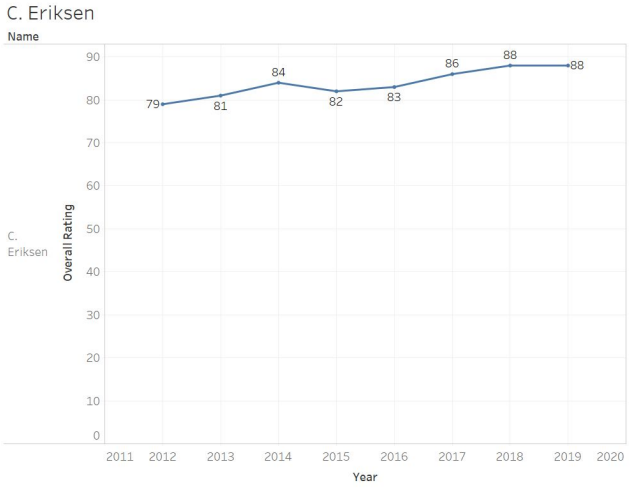

Fig 35: Graph of C.Eriksen's Overall Rating vs Year

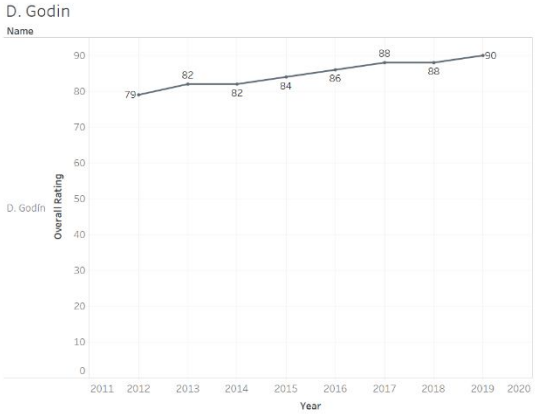

Fig 36: Graph of D.Godin's Overall Rating vs Year

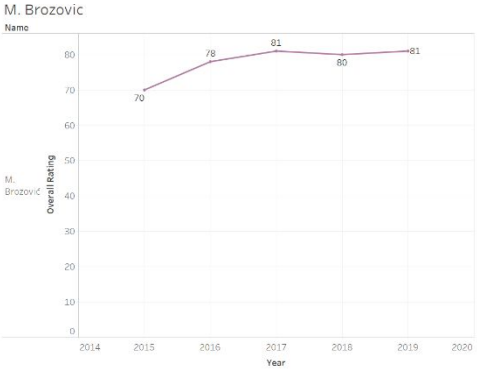

Fig 37: Graph of M.Brozovc's Overall Rating vs Year

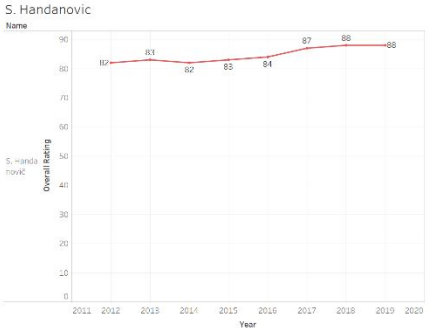

Fig 38: Graph of S.Handanovic Overall Rating vs Year

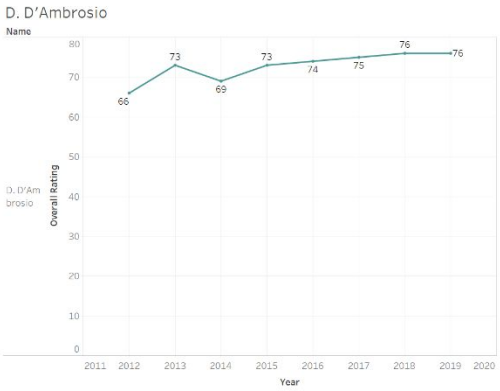

Fig 39: Graph of D.D'Ambrosio Overall Rating vs Year

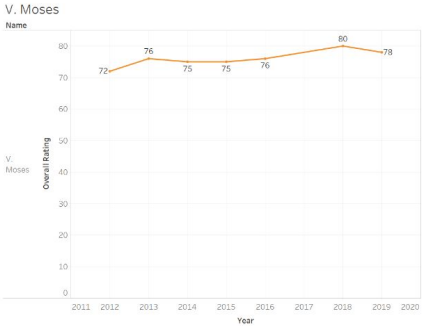

Fig 40: Graph of V.Moses Overall Rating vs Year

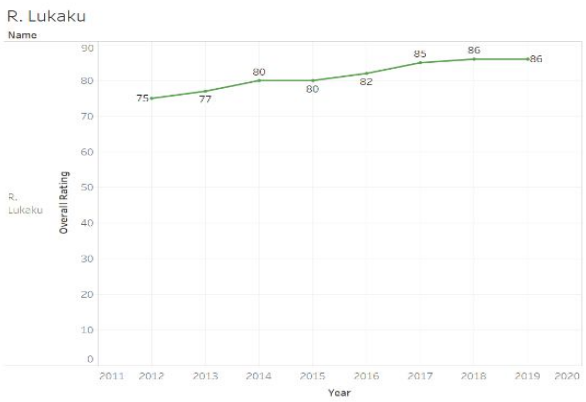

Fig 41: Graph of R. Lukaku Overall Rating vs Year

1.7 Model Deployment:

- Using the Django framework, we created a webapp that uses our model and predicts the Overall Rating for each player, when features of that player are given as input into the webapp.
- Django is a python based free and open source web framework which follows the model-template-view architectural pattern.
- We utilized an AWS (EC2) instance to deploy this webapp.
- Link for the webapp- http://ec2-3-80-79-254.compute-1.amazonaws.com/

Demo:

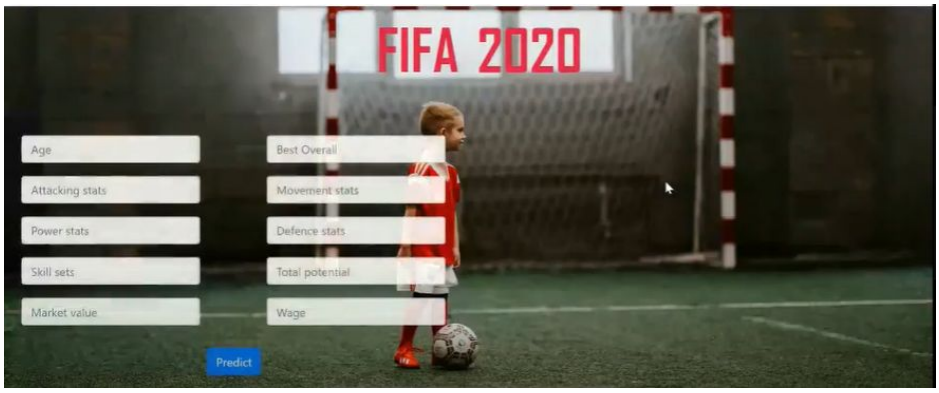

Fig 42: Showing the image of Web App for our prediction model with blank values

Fig 43: Showing the image of Web App for our deployment with filled values and showing 94 as the Predicted overall rating

Fig 44: Showing the image of Web App for our deployment with filled values and based on which shows 77 as the Predicted overall rating

All the code to the project is available in the GitHub repository Link.

1.8 Limitations and recommendations for future Research

- Using web-scraping was a new task that we used and implemented successfully.
- New Machine Learning models include: SVM and Random forest. These were challenging tasks that we learnt throughout the project course.
- Deployment for production needs of the project required assistance of AWS which was quite new to us.
- Also, learning Django to develop the webapp was very interesting and a good experience for us.
- Our current solution is not globally optimal since we only consider the players for swapping, whose teams have overall 75+ ratings. This means that there could be players with higher overall

stats in lower rated teams.We can use Optimization algorithms like Mixed Integer Linear Programming(MILP)[9] and decomposition techniques to find a better global optimal solution.

- For the future implementation of the model, we can utilize the prediction and the optimization techniques to analyze sports that require similar predictor variables but different targets.

1.9 References

- [1] Myatt and Johnson (2014), *Making Sense of Data I*, 2nd Edition, Wiley, ISBN: 978-1-118-40741-7
- [2] Zumel and Mount (2014), *Practical Data Science with R*, Manning Publications Co., ISBN: 978-1-617291-56-2
- [3] Thomas Reilly Christopher Carling, A. Mark Williams.Handbook of Soccer Match Analysis: A Systematic Approach to Improving performance., 2005.
- [4] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion,O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vander-Plas, A. Passos, D. Cournapeau,M. Brucher, M. Perrot, and E. Duch-essay. Scikit-learn: Machine learning in Python.Journal of MachineLearning Research, 12:2825–2830, 2011.
- [5]Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer,Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, ArnaudJoly, Brian Holt, and Ga ¨el Varoquaux. API design for machine learning software:experiences from the scikit-learn project. InECML PKDDWorkshop: Languages for Data Mining and Machine Learning, pages108–122, 2013
- [6] Vineet M. Payyappalli and Jun Zhuang. A data-driven integer programming model for soccer clubs' decision making on player transfers.Environment Systems and Decisions, pages 1–16, 2019
- [7] Mike Joang Andy Peng Ashwin Nair, Archit Dhanani. r24-785 engineering optimization.Developing the simplex method with numpy and matrix operations.
- [8]srome/srome.github.io the mit license (mit), copyright (c) 2015 barry clark.
- [9]George L. Nemhauser; Laurence A. Wolsey (1988). Integer and combinatorial optimization. Wiley. ISBN 978-0-471-82819-8.