Soccer Data Analysis

CSC 440 - Final Project Report

Ankit Ranjan
Avinash

INTRODUCTION

Soccer is the most followed sports in the world with an estimated following that comprises of more than half of the world's population. The last World Cup(2014) had an estimated TV viewership of 3.2 billion and a billion people tuned in to watch the final that saw Germany become world champions. Being soccer enthusiasts we have always had interest in the game and the in-game statistics like possession, shots taken, number of tackles etc. Every game has in-depth analysis available and there has been lot of analysis to predict the outcomes. But what was not readily available was the "big picture",that is an analysis on the leagues, teams and players that would span more than just one game. Thus we decided on doing an analysis on the major European Leagues, over the span of a few seasons.

Now EA Sports and FIFA have been partners for quite sometime and this partnership has resulted in one of the most comprehensive and systematic data collection ever done for soccer. The two associations have paired up to collect data of every match that has taken place in the European Leagues for the past 8 years or so and this data has been made available to everyone to do any analysis on. The list of every player that has played in every match, their ratings for the game, their overall rating, which is calculated using a proprietary algorithm, stats regarding every attribute(upto 40) is readily available. The teams have also been assigned various attributes that describe their playing style. All of this has been combined with match details to give a dataset that describes the complete soccer season in a very detailed manner.

In this project we wish to answer some of the questions like which teams have similar playing styles across all the leagues and within the league too. We hope to identify teams that tend to transfer players among themselves and see if there is affinity between teams that was hitherto unknown and in same line identify which leagues tend to move players only among themselves and which leagues are involved in transfers with another league. Our plan is to also try and classify players into the type of team that they tend to belong too, if such a pattern exists.

Problem Statement

Our problem statement thus comes to answering the following questions:

1. Classify players into teams clusters.
2. Identify teams across Europe that have similar playing styles.
3. Identify teams within each league that have similar playing styles.
4. Identify transfer trends between teams and also leagues.

Related Work

Not much has been done in the soccer database in general. Most of the analysis we see is in-depth analysis if one game at max. There have been many research done in the files of predicting the outcome of the game but that's about all the research work that has been done so far.

Sérgio Nunes and Marco Sousa. Applying data mining techniques to football data from european championships. In Actas da 1a Conferência de Metodologias de Investigação Científica

(CoMIC'06), 2006 was one paper we found that had done work on similar lines, but all the work they did was done for Portuguese League only and they didn't get satisfactory results either. There is some unofficial work that has been done by Kaggle users. The majority of which basically is nothing more than execution of a simple query and then displaying the results in a fancy manner.

METHODOLOGY

Data Collection

The dataset used for this project was compiled at the end of 2015-2016 soccer season and consists of all the data from season 2008-2009 onwards. This data was downloaded from https://www.kaggle.com/hugomathien/soccer. We also scraped transfer data from http://www.soccernews.com/soccer-transfers/ using BeautifulSoup of Python, but the data was too noisy for us to go ahead with. Thus we ended up using only the data that we obtained from Kaggle.

Data Description

Before we delve into how the data was preprocessed, a little overview of the database used. The dataset contains data about 10,644 players and 25,979 matches. We have information regarding the To League of 11 European countries and 285 teams.
The tables and their attributes:

- Country - id, name. Contains the names of 11 countries and their corresponding id.
- League - id,country_id,name. Contains the name and id of each league and their respective countries has been mapped using the country id.
- Match - Has over 50 attributes. Contains the stats regarding a game, including the team names, number of goals scored, players who played, ratings for players in the game etc.
- Player - Contains basic info about a player. Like their height, weight and DOB.
- Player_Attributes - Has about 40 attributes. Contains ratin gof player for every attribute. These attributes span from curve, crossing, dribbling to tackles and gk handling etc.
- Team - Has team name and team ids.
- Team_Attributes - Just like Player_Attributes contains all the attribute values for a team, has nominal as well as numeric values for each attribute. The attributes range from chanceCreationCrossing, chanceCreationPassing to defenceWidth, buildUpPlay etc.

We created a few views for our use. These were join of Team and team_Attributes, Player and PLayer_Attributes. The amalgamation of above mentioned views and the Match table was another one, which was used for transfer trends calculation.

Data Preprocessing

The database had many features that were recorded multiple times for the same entity, whether a player's rating or a team's rating for an attribute. So, we only used the latest ones for our analysis. These duplicates were not removed directly from the database, we didn't want to compromise the integrity of data.So,we would write queries to extract the data that we needed. We created Views from the relevant data and then exported them out as CSV files. For eg, when our scraping data didn't pan out to be very well, we decided to use the database and try to extract the players, across seasons and the teams where these players have played. One

assumption we made for this was that every team plays at their home grounds for sure. Thus we used following query to get the transfer data:

```
Select Distinct League.name,Team.team_long_name,Match.home_player_1,Match.season from Match ,League,Team Where Match.home_team_api_id in
(Select home_team_api_id from Teams_League)  AND League.id = Match.league_id AND Team.team_api_id=Match.home_team_api_id
UNION
Select Distinct League.name,Team.team_long_name,Match.home_player_2,Match.season from Match ,League,Team Where Match.home_team_api_id in
(Select home_team_api_id from Teams_League)  AND League.id = Match.league_id AND Team.team_api_id=Match.home_team_api_id
UNION
Select Distinct League.name,Team.team_long_name,Match.home_player_3,Match.season from Match ,League,Team Where Match.home_team_api_id in
(Select home_team_api_id from Teams_League)  AND League.id = Match.league_id AND Team.team_api_id=Match.home_team_api_id
UNION
Select Distinct League.name,Team.team_long_name,Match.home_player_4,Match.season from Match ,League,Team Where Match.home_team_api_id in
(Select home_team_api_id from Teams_League)  AND League.id = Match.league_id AND Team.team_api_id=Match.home_team_api_id
UNION
Select Distinct League.name,Team.team_long_name,Match.home_player_5,Match.season from Match ,League,Team Where Match.home_team_api_id in
(Select home_team_api_id from Teams_League)  AND League.id = Match.league_id AND Team.team_api_id=Match.home_team_api_id
UNION
Select Distinct League.name,Team.team_long_name,Match.home_player_6,Match.season from Match ,League,Team Where Match.home_team_api_id in
(Select home_team_api_id from Teams_League)  AND League.id = Match.league_id AND Team.team_api_id=Match.home_team_api_id
UNION
Select Distinct League.name,Team.team_long_name,Match.home_player_7,Match.season from Match ,League,Team Where Match.home_team_api_id in
(Select home_team_api_id from Teams_League)  AND League.id = Match.league_id AND Team.team_api_id=Match.home_team_api_id
UNION
Select Distinct League.name,Team.team_long_name,Match.home_player_8,Match.season from Match ,League,Team Where Match.home_team_api_id in
(Select home_team_api_id from Teams_League)  AND League.id = Match.league_id AND Team.team_api_id=Match.home_team_api_id
UNION
Select Distinct League.name,Team.team_long_name,Match.home_player_9,Match.season from Match ,League,Team Where Match.home_team_api_id in
(Select home_team_api_id from Teams_League)  AND League.id = Match.league_id AND Team.team_api_id=Match.home_team_api_id
UNION
Select Distinct League.name,Team.team_long_name,Match.home_player_10,Match.season from Match,League,Team Where Match.home_team_api_id in
(Select home_team_api_id from Teams_League) AND League.id = Match.league_id AND Team.team_api_id=Match.home_team_api_id
UNION
Select Distinct League.name,Team.team_long_name,Match.home_player_11,Match.season from Match,League,Team Where Match.home_team_api_id in
(Select home_team_api_id from Teams_League) AND League.id = Match.league_id AND Team.team_api_id=Match.home_team_api_id

ORDER BY Match.season
```

This gave us the required data and using DB Browser for SQLite we exported it as a csv file. Similarly, we acquired data for different phases of our project. The team names and the league they belong to, the player attributes with player names, the top 20 players etc were all extracted in a similar fashion.

As is common with large databases there were issues with data integrity too. There were numeric attributes for nominal features and sometimes the data was completely missing. So, for the nominal attributes we discretized the numeric values. For eg, an attribute ‘defence_work_rate’ for Player_Attribute table had three nominal values - ‘high’,’low’ and ‘medium’. We encountered some rows where the values were numeric and ranged from 0-10. So, we converted the values in range [0-4] to ‘low’, [5-7] as ‘medium’ and [8-10] as high. On the other hand whenever we came across missing values which couldn’t be directly replaced,we first calculated the amount of data that was missing. In our case, luckily, we never had more than 1% of data missing. So, we ignored the rows with missing attribute values in our final analysis.

DATA MINING

Our next step was the processing of data. We did a few preliminary analysis on the data and extracted Top 20 current players and Top 20 players with highest potential. Again this was done for all the current players only. We have data for 11 leagues across 11 different countries and for 8 years.

Our next step was to get the teams with similar playing styles in one cluster(problem statement 1) and we decided to use EM algorithm for this clustering. So, we used the team names and id from table 'Team' and then joined it with the table 'Team_Attributes' to get the relevant data. Then ran the data through EM algorithm and got 4 resultant clusters. These clusters were well defined and we thus classified the playing style of the teams into 4 types.(the result has been discussed in the Results section).

Next up tackled the similar teams clustering for every league(problem statement 2). Now our approach could have been to use the cluster results from the similar team result across Europe but we decided to go separately for each league. But soon, we realised this would be an issue as the amount of data for every league was very less. We had ~30 teams corresponding to every league. Running the EM clustering algorithm returned us only one cluster, which is basically pointless and running K-means with 4 cluster option gave us 4 clusters but the result was analogous to a Gaussian model and thus could not be relied upon. Thence we decided to keep the result from clustering the teams across Europe(it has about 285 teams' data) than to do a separate clustering for individual leagues.

Now by exporting the results of problem statement 1 as csv files we had the team and its attributes along with the clusters it had been assigned to,this was file was essential to our classification of players to different team clusters(problem statement 4).Coming to this problem, we first used the 'Match' and 'Team' tables to get all the teams a player has ever played for.

```
Select  A.player_api_id,A.player_name, A.birthday,B.date,B.overall_rating,B.potential,B.preferred_foot,B.attacking_work_rate,B.defensive_work_rate, B.crossing,
B.finishing,B.heading_accuracy,B.short_passing,B.volleys,B.dribbling,B.curve,B.free_kick_accuracy,B.long_passing,B.ball_control,B.acceleration,B.sprint_speed,
B.agility,B.reactions,B.balance,B.shot_power,B.jumping,B.stamina,B.strength,B.long_shots,B.aggression,B.interceptions,
B.positioning,B.vision,B.penalties,B.marking,B.standing_tackle,B.slide_tackle,B.gk_diving,B.gk_handling,B.gk_kicking,B.gk_positioning,B.gk_reflexes
from Player A left Join Player_Attributes B on Player_Attributes.player_api_id = Player.player_api_id
where Player_Attributes.player_api_id = Player.player_api_id Group by Player_Attributes.player_api_id;
```

Then we left joined the players with their attributes to get the player attributes along with their id.

We again left joined this data to the player and their teams data and got the teams that every player played for along with player attributes. The team names were subsequently replaced with their cluster number and was used as the class attribute for the players in further classification. The plan was to get a distribution of player attributes corresponding to every team cluster. We ran a few Classifiers on this data, but data was not as great as we expected it to be. We tried J48, RandomForest,Random Tree, ADABoost and K-star algorithms but the best result we got was from the Naive Bayes algorithm, so we have included the result for Naive Bayes only. The result for Naive Bayes was less than 40%.

Now coming to our final problem statement. We wanted to analyse the transfer data across all the European leagues. Now, as mentioned in Data Preprocessing section we first joined the League and Team table keeping only the distinct values to get all the teams along with the league that these teams belonged to, across all the seasons. Using the fact that every team players at least half it matches every season at home, we used the data for home team like the home players1 through home players11 to get all the players for a team in a particular season (this query has been mentioned in the data preprocessing section). Once we had the players and the teams that they played for, we used Python scripts to get transfer matrices for the players. There were a couple of transfer matrices created. One was done across leagues and the ith row would give the number of players transferred from the ith league that season, jth columns would tell us which league were they transferred to. A similar matrix was created to find out the team transfer trends across a league. Finally we created another transfer trend matrix for all the big teams of Europe(FC Barcelona, Real Madrid CF, Arsenal, Manchester United, Borussia Dortmund,etc.). The results of all has been mentioned in the Result section below.

We also tried to see if we could identify the player type as in forward, midfielder, defender or goalkeeper from the player attribute values that we had. So, running 4 cluster K-means gave us two very distinct clusters to identify a Goalkeeper and a Forward. We had two more clusters out of which one was for midfielders and the other for defenders but a few of the attribute means were a little close, which could be the result of dynamic plays of today. Many midfielders drop back to defend at times too(Sergio Busquets for FC Barcelona and Wayne Rooney for Manchester United last season). Also, defenders like Sergio Ramos, Gerard Pique, Daniel Alves tend to make runs forward. Thus the line between a midfielder and defender has become blurry these days, but still barring a few prominent defenders/midfielders the clusters were very well defined for the respective player types.

RESULTS

Team Clusters

The idea was to identify teams across Europe that have a similar playing style. Using a simple k-means algorithm for this purpose has a shortcoming - we have to specify the number of clusters required. To overcome this, we used the iterative EM algorithm. By visual inspection of the attribute means, not much can be said about what each cluster represents. In order to understand the output, we look at the cluster assignments.

Among other teams, we found FSV Mainz, Borussia Dortmund and Leicester City belonging to cluster 0. With our domain knowledge, we could identify this cluster as teams that play with a very high attacking pressure. This can be verified by the 'chanceCreationShooting' attribute that has the highest mean for cluster 0.

Cluster 1 had teams like Arsenal, FC Barcelona and FC Bayern Munich that are known to play the passing game with not so much attacking pressure. This can be verified by the low 'buildUpPlaySpeed' and 'chanceCreationPassing' means. As these teams pass the ball a lot, only a small fraction of passes actually create a goal scoring opportunity.

Cluster 2 had teams including Atletico Madrid and Newcastle United, both teams that play counter attacking football. These teams do not build up plays and hence have the second lowest mean among the clusters. Given that most teams across europe play primarily counter attacking football, it makes sense that this cluster has the most number of teams.

Cluster 3 had teams that play different from the above three types.

	Cluster			
Attribute	0	1	2	3
buildUpPlaySpeed				
mean	59.2693	45.7432	52.8984	56.4504
buildUpPlayDribbling				
mean	48.2363	47.6059	49.0121	45.9915
buildUpPlayPassing				
mean	51.6673	39.8413	51.6543	52.8596
chanceCreationPassing				
mean	60.1238	45.8499	51.6508	52.1232
chanceCreationCrossing				
mean	55.1138	50.2412	54.7417	51.0575
chanceCreationShooting				
mean	56.4755	47.4222	51.2274	48.5528
defencePressure				
mean	55.3824	50.8609	44.9026	35.9719
defenceAggression				
mean	58.1193	50.2745	46.6227	43.8814
defenceTeamWidth				
mean	54.0041	53.6904	52.9925	45.895

Clustered Instances

0	39 (14%)
1	56 (20%)
2	120 (42%)
3	70 (25%)

Figure 1. Team Cluster Means of each attribute as calculated by EM Algorithm

Team Clusters Within A League

We wanted to analyze the playing styles of teams within each league to see if we get results different from the previous clusters. However, since each league has around 30-40 teams, clustering algorithms could not give meaningful results. There is simply not enough variety.
EM algorithm detected a single cluster. Specifying the number of clusters as 4 for k-means did not give meaningful results either. Because of the small variance in the data, neither of the clustering algorithms were successful.
As a result, we had to use the results of the previous algorithm to cluster teams within the league as well. Using the cluster labels assigned by EM run on all the teams, we got the following results for teams in EPL(English Premier League)
Cluster 0 : Burnley, Leicester City…
Cluster 1 : Everton, Liverpool, Manchester United…
Cluster 2 : Blackburn, Newcastle United…
Cluster 3 : Aston Villa, Crystal Palace…
each cluster having the same interpretation as the previous clusters.
The same was done for all other leagues as well.

Classification Of Players Into Identified Team Clusters

We wanted to know if a player can be assigned to the team clusters we identified based on his playing attributes. For this purpose, we used the match log to find out all the teams that a player has played for and hence all the clusters he belonged to. With this additional information, we classified players using the cluster number as the class label.
However, we found that none of the classifiers could not give us good accuracy. In order to explain this, we went back to the data to find out how many players transferred to teams belonging to different clusters. 37% of all players had transferred to teams belonging to different clusters. This explained the poor results of the classifiers. This explanation also makes sense as a player need not only play in teams that have a similar playing style. For example, there have been several high profile transfers of players from FC Barcelona to Real Madrid CF, both teams playing very different kind of football.

```
=== Summary ===

Correctly Classified Instances        6539               37.8633 %
Incorrectly Classified Instances     10731               62.1367 %
Kappa statistic                          0.0678
Mean absolute error                      0.3434
Root mean squared error                  0.4419
Relative absolute error                 96.355  %
Root relative squared error            104.6932 %
Total Number of Instances            17270

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Area  Class
               0.185     0.114     0.286       0.185   0.225      0.085    0.557     0.248     B
               0.742     0.645     0.440       0.742   0.553      0.102    0.564     0.446     C
               0.025     0.020     0.294       0.025   0.046      0.016    0.539     0.274     D
               0.230     0.156     0.207       0.230   0.218      0.071    0.592     0.189     A
Weighted Avg.  0.379     0.313     0.339       0.379   0.313      0.073    0.561     0.326

=== Confusion Matrix ===

   a    b    c    d   <-- classified as
 631 2050   66  659 |    a = B
 736 5206  125  948 |    b = C
 409 3068  106  678 |    c = D
 430 1498   64  596 |    d = A
```

Figure 2. Naive Bayes' Classifier Output

League Transfers

We wanted to identify transfer trends between the leagues of Europe. In order to so, a transfer matrix was constructed where Matrix[i,j] represented the number of transfers from league 'i' to league 'j'.

Some of the inferences we made are:

1. Two clusters of leagues emerged. The big leagues of europe including England, France, Germany, Italy and Spain tended to make transfers among themselves and the rest of the leagues transferred players among themselves.
2. Within these clusters, we looked for leagues that had almost equal number of transfers in either direction. Among the big leagues, Italy and Spain had the most symmetric transfer ratio and among the small leagues, Belgium and Netherlands had a large number of symmetric transfers.
3. In terms of unidirectional flow of players, the most evident pair was between the two clusters. The small leagues sold players to big leagues but did not receive players from the big leagues. However, portugal was found to be an exception to this generalisation and this can be attributed to the geographic proximity of Spain and Portugal

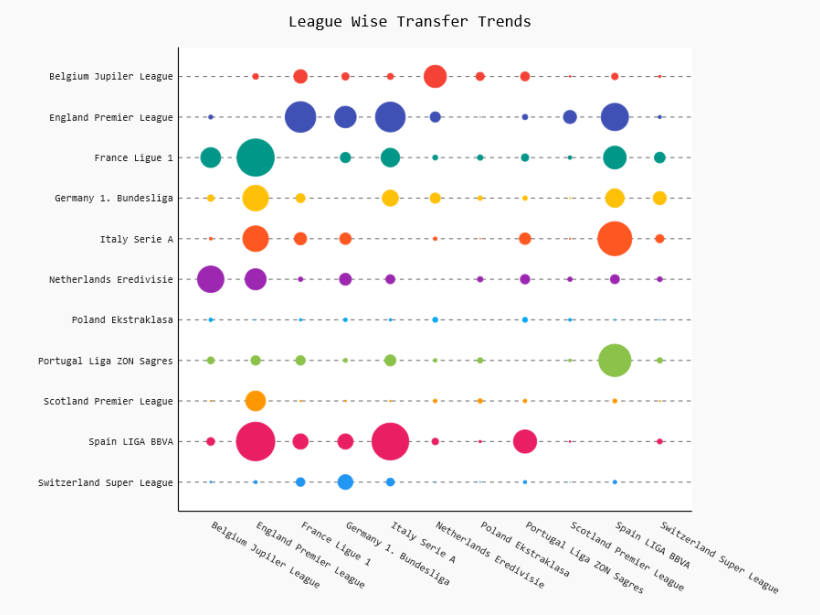
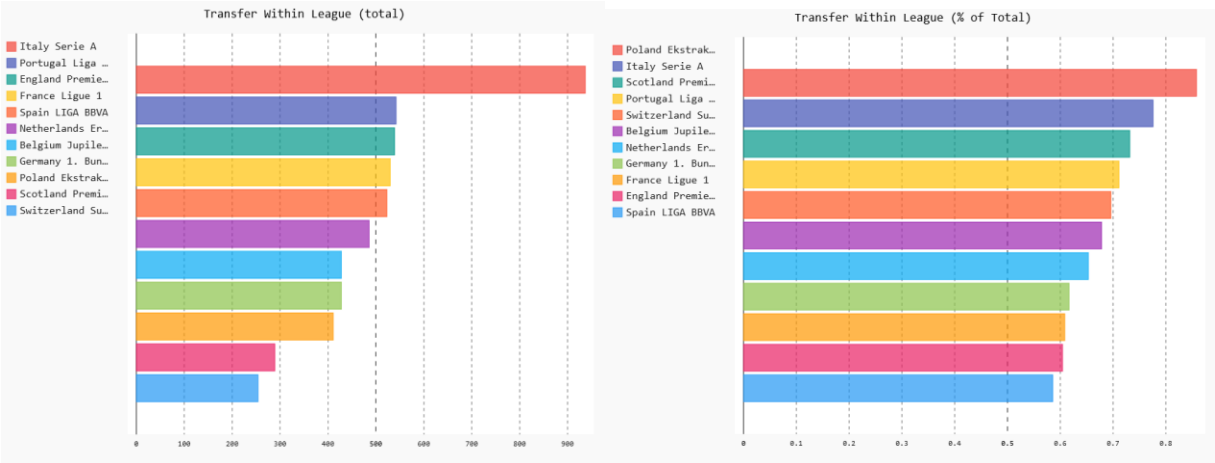


Figure 3. League Wise Transfer Trends

Transfers Within A League

Figure 3 does not include the transfers of players within the same league. We analyse such transfers in this section. Italy was found to have the most number of transfers within the same league while Switzerland was at the bottom of this list.

However, if we were to compare the ratio of within-league transfer to all outgoing transfers for each league, Poland leap frogs the rest of the leagues to claim top spot. All the big leagues except for Italy can be found at the bottom of this list implying that players in these leagues would rather transfer out to teams in other leagues than stay in the same league.



A. B.

Figure 4. A) Total number of transfers within the same league
B) Percentage of transfers within the same league

Transfer Trends In Individual Leagues

We now further analyse within league transfers. For 3 of the big leagues namely England, Spain and Germany, we created a transfer matrix to look for trends.

One of the most striking observations was that the big clubs did not buy too many players from the same league. These clubs have the money and can afford to buy quality players from the rest of Europe. This trend continued across the other big leagues as well. For example, in the EPL, Manchester United barely bought any player from other teams. In Spain's La Liga, Real Madrid also did not buy too many players from other clubs. In Germany's Bundesliga, Borussia Dortmund follows the same trend.

Looking for bidirectional transfer trends proved most useful in the case of Bundesliga. We were surprised to find FC Bayern Munich and Hoffenheim shared such a relationship. In addition to this pair, Borussia Dortmund and Stuttgart also had bidirectional transfers

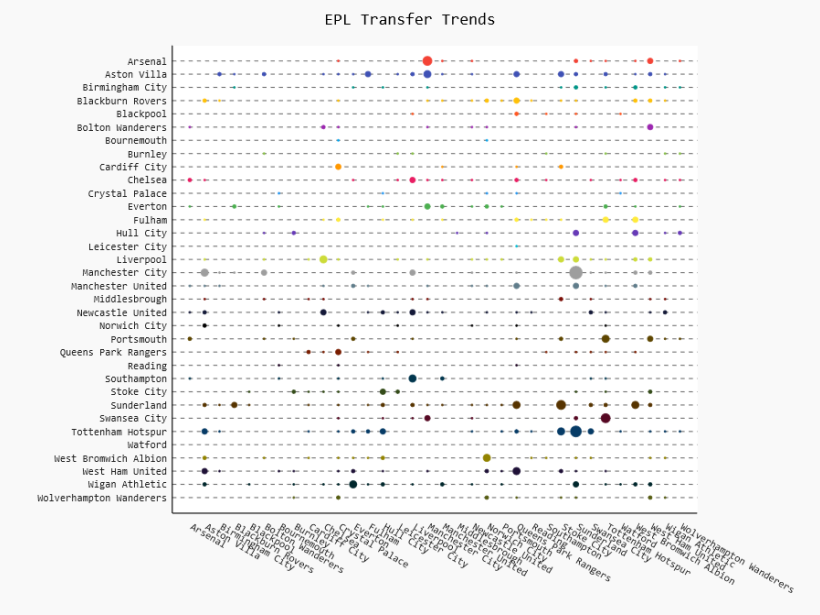


Figure 5. EPL Transfer Trends

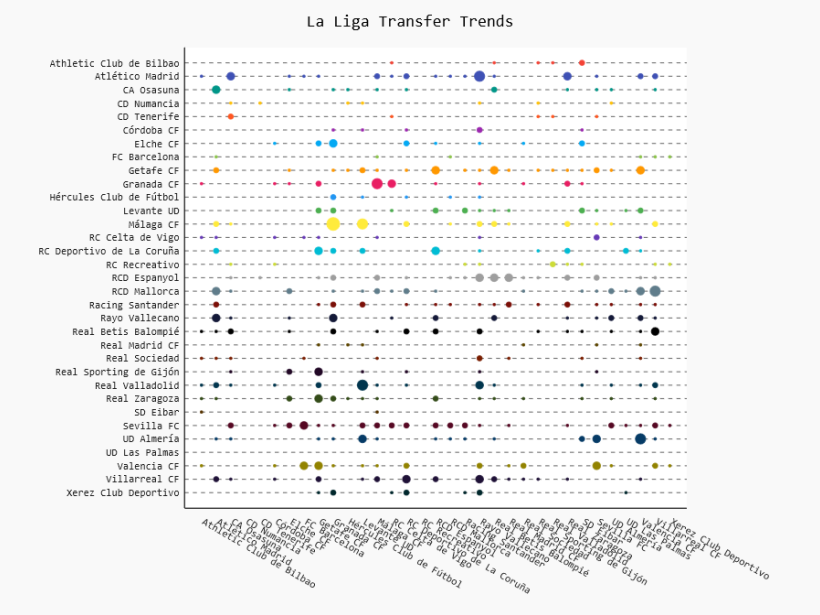


Figure 6. Spain La Liga Transfer Trends

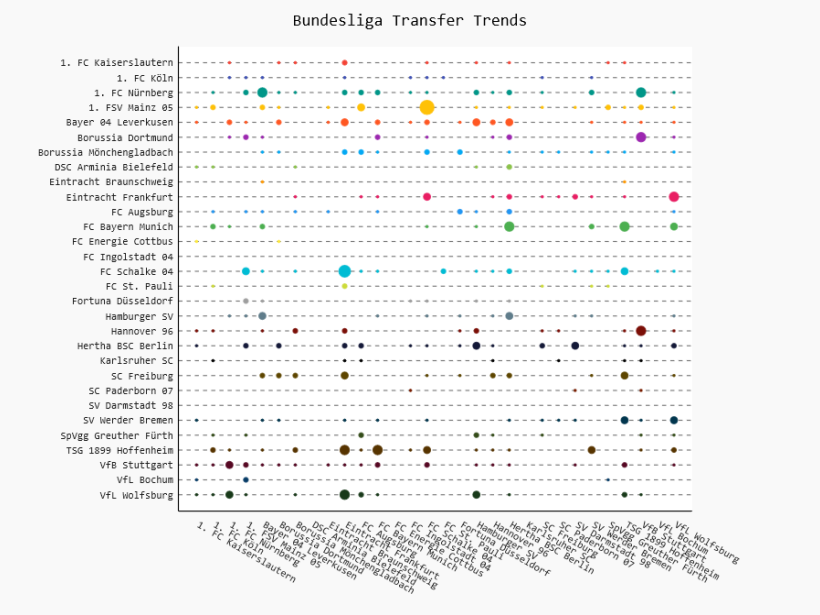


Figure 7. Germany Bundesliga transfer Trends

Transfers Among Big Clubs

We took a deeper look at transfer trends between the big clubs of Europe including Arsenal, Juventus, PSG, Bayern and Barcelona among others. Among these teams we found Dortmund to have the smallest number of incoming transfers while Inter Milan had the largest. But to get a better understanding of which clubs buy most players from their competitors, we needed to look at percentages. Surprisingly, FC Barcelona topped this list. The general feel in the football

community is that Real Madrid tend to buy most number of players from competitors but this turned out to be incorrect.

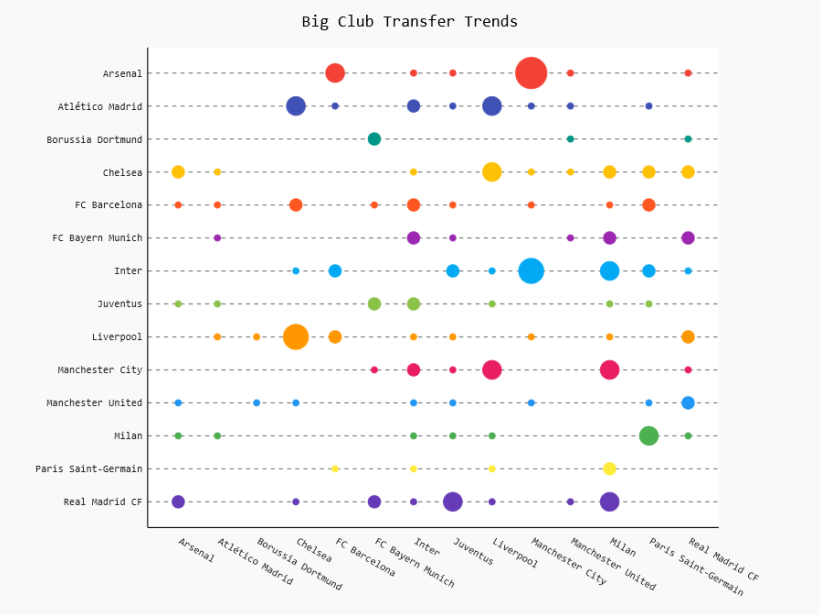


Figure 7. Big Club Transfer Trends

Team	In from Other Big Clubs	Total incoming	Percent
Barcelona	12	28	42.85
Chelsea	15	42	35.71
Madrid	14	42	33.33
Arsenal	12	37	32.43
Liverpool	14	49	28.57
ManU	9	32	28.12
Inter	16	59	27.11
Bayern	9	39	23.07
Atletico	13	58	22.41
ManC	11	55	20
Juventus	9	51	17.64
Milan	9	52	17.31
Dortmund	4	25	16
PSG	5	34	14.7

Table 1. Comparison of player acquisition of big clubs

Player Clustering

In addition to our problem statement, we wanted to see if we can identify a player's position on the pitch from his attribute ratings. As this is an unsupervised process, we went ahead with clustering. This data is highly multidimensional and so we used k-means.

Final cluster centroids:

Attribute	Full Data (10664.0)	Cluster# 0 (3625.0)	1 (3242.0)	2 (939.0)	3 (2858.0)
weight	168.3789	164.4894	164.0605	184.4079	172.9447
height	181.8975	179.9399	179.8182	188.7282	184.4948
acceleration	64.751	72.248	68.0845	43.279	58.5154
aggression	60.9582	53.864	70.5509	31.6603	68.7005
agility	64.2216	71.568	68.7379	45.9066	55.7979
balance	63.6579	68.3862	68.7628	45.4576	57.8495
ball_control	62.1407	69.2844	70.4522	22.3674	56.719
crossing	54.0137	58.848	65.5583	16.3078	47.1746
curve	51.9378	60.4801	62.3814	17.6528	40.5206
dribbling	57.8547	68.8163	66.3837	16.5453	47.8488
finishing	48.2346	65.4246	52.5213	15.8062	32.2232
free_kick_accuracy	48.1736	54.9672	58.7557	16.1171	38.085
gk_diving	15.5915	10.4301	10.397	69.377	10.3593
gk_handling	15.7251	10.885	10.9343	65.9585	10.7943
gk_kicking	16.6822	11.5639	12.7699	64.0128	12.0616
gk_positioning	15.8751	10.7804	10.9849	67.5037	10.9216
gk_reflexes	16.1022	10.7666	10.9019	70.6667	10.8415
heading_accuracy	56.7583	58.4877	60.4053	16.4292	63.6777
interceptions	50.9326	33.5514	66.9226	22.1033	64.3118
jumping	66.4517	65.1798	68.0953	63.2598	67.2493
long_passing	56.3588	54.8921	67.7273	28.2066	54.5724
long_shots	52.1002	62.6284	62.839	16.475	38.2694
marking	47.0259	26.8491	62.814	15.8882	64.9384
penalties	68.0466	68.0913	70.3396	67.8328	65.4591
positioning	53.3907	63.5225	58.8492	23.7252	44.0945
potential	53.4089	67.0149	61.6622	15.8978	39.1134
preferred_foot	70.9917	71.4623	72.5842	70.148	68.8656
reactions	65.4962	65.587	69.2323	64.4899	61.4734
short_passing	61.8211	64.5228	71.1669	28.0266	58.8961
shot_power	60.9919	68.9639	69.6064	25.2268	52.8593
sliding_tackle	48.0451	28.6	64.3761	17.1028	64.3496
sprint_speed	65.1311	72.1437	67.9877	43.2034	60.2005
stamina	64.5647	64.4022	73.3896	35.442	64.3286
standing_tackle	50.4978	30.9026	67.4454	16.1896	67.3992
strength	68.2289	64.4171	68.8806	64.8978	73.4188
vision	56.1553	61.9503	64.7394	34.2376	46.2686
volleys	47.8235	61.2047	53.7375	17.2952	34.1728
sprint_speed	65.1311	72.1437	67.9877	43.2034	60.2005
stamina	64.5647	64.4022	73.3896	35.442	64.3286
standing_tackle	50.4978	30.9026	67.4454	16.1896	67.3992
strength	68.2289	64.4171	68.8806	64.8978	73.4188
vision	56.1553	61.9503	64.7394	34.2376	46.2686
volleys	47.8235	61.2047	53.7375	17.2952	34.1728

Figure 8. Player attribute means as obtained from K-Means

The simple algorithm proved to be highly accurate with each of clusters representing either forward, mid-field, defence or goalkeepers.

1. Cluster 0 represents forward players such as Cristiano Ronaldo and Jamie Vardy
2. Cluster 1 represents mid-fielders such as Andres Iniesta and Frank Lampard

3. Cluster 2 represents goalkeepers such as Claudio Bravo and Roman Burki
4. Cluster 3 represents defenders such as Pepe and Jerome Boateng

While the algorithm works with a very high accuracy for clusters 0 and 2, it had some problems with clusters 1 and 3. Ball playing defenders such as David Luiz, Mats Hummels and Sergio Ramos were all clustered into cluster 1 while in actuality they should belong to cluster 3.

Top 20 Players (Current)

player_api_id	player_name	birthday	overall_rating
30981	Lionel Messi	24-06-1987	94
30893	Cristiano Ronaldo	05-02-1985	93
19533	Neymar	05-02-1992	90
27299	Manuel Neuer	27-03-1986	90
40636	Luis Suarez	24-01-1987	90
30834	Arjen Robben	23-01-1984	89
35724	Zlatan Ibrahimovic	03-10-1981	89
30955	Andres Iniesta	11-05-1984	88
36378	Mesut Oezil	15-10-1988	88
37412	Sergio Aguero	02-06-1988	88
80562	Thiago Silva	22-09-1984	88
93447	Robert Lewandowski	21-08-1988	88
107417	Eden Hazard	07-01-1991	88
30894	Philipp Lahm	11-11-1983	87
30962	Sergio Ramos	30-03-1986	87
31097	Luka Modric	09-09-1985	87
31921	Gareth Bale	16-07-1989	87
36183	Jerome Boateng	03-09-1988	87
95078	Toni Kroos	04-01-1990	87
164684	James Rodriguez	12-07-1991	87

Top 20 Players (Potential)

player_api_id	player_name	birthday	potential
19533	Neymar	05-02-1992	94
30981	Lionel Messi	24-06-1987	94
30893	Cristiano Ronaldo	05-02-1985	93
164684	James Rodriguez	12-07-1991	93
248453	Paul Pogba	15-03-1993	91
27299	Manuel Neuer	27-03-1986	90
40636	Luis Suarez	24-01-1987	90
107417	Eden Hazard	07-01-1991	90
170323	Thibaut Courtois	11-05-1992	90
182917	David De Gea	07-11-1990	90
395154	Alen Halilovic	18-06-1996	90
30834	Arjen Robben	23-01-1984	89
31921	Gareth Bale	16-07-1989	89
35724	Zlatan Ibrahimovic	03-10-1981	89
36378	Mesut Oezil	15-10-1988	89
93447	Robert Lewandowski	21-08-1988	89
95078	Toni Kroos	04-01-1990	89
169200	Kevin de Bruyne	28-06-1991	89
190972	Marco Verratti	05-11-1992	89
230982	Raphael Varane	25-04-1993	89

CONCLUSION

Even though our data set is rather small in comparison to data sets available today, the different ways to analyze it and the possible interpretations are manifold. The clustering of teams to reveal playing styles in our project, proves that the proprietary algorithm used by EA Sports and FIFA for assigning ratings is very accurate. Maybe including more attributes such as ‘attackingPressure’ could improve the clustering even further.

The aim of classifying players into the team clusters remained unfulfilled but the reason for this was revealed by our analysis. Players need not always play for teams that have similar playing styles. Perhaps a probabilistic classifier would give us better results.

We were able to analyse transfer trends the match log revealing some interesting findings such as the bidirectional relationship between FC Bayern Munich and Hoffenheim. We also clustered players according to the positions they play in.

REFERENCES

1. https://www.kaggle.com/hugomathien/soccer
2. *Sérgio Nunes and Marco Sousa. Applying data mining techniques to football data from european championships. In Actas da 1a Conferência de Metodologias de Investigação Científica (CoMIC'06), 2006*

Technologies Use:

- Python 3.5: with pygal, cairociff as visualisation packages. Numpy and sqlite3 were used too.
- WEKA
- DB Browser for SQLite