# Classifying Opponent Attacking Tendencies Within La Liga

Eric Bradford, Alex Cauneac, Emma Chesley, Sean Ko, Kevin Lee, Garrett Souza, and Tim Yang

**Abstract.** In this report, we take an unsupervised approach to clustering the attacks of the FC Barcelona (FCB) Soccer Club. Through incorporating spatial, contextual, and statistical information into the clustering framework, we developed a holistic assessment of team attacking tendencies. We have created and refined a model which clusters opponent attacks into homogeneous subsets. This model is capable of assessing the degree of similarity across team attacking tendencies, providing information on which teams in La Liga have the most comparable attacks. Our deliverables to FC Barcelona include a scouting report and python library.

## 1. Introduction

Soccer is a sport of ambiguity and intuition. Many factors go into the decision making processes of players at each moment of the game. These factors range from team attacking style and individual tendencies to situational context and spatial positioning. The goal of this project is to quantify and conceptualize these factors, and develop a method for assessing team attacking tendencies. In partnership with FC Barcelona's data analytics team, we have implemented and refined a model capable of clustering opponent attacks into homogeneous subsets. These subsets can then be analyzed and described. Furthermore, the model is capable of assessing the degree of similarity of team attacking tendencies, providing information on which teams in La Liga have the most comparable attacks. Through incorporating spatial, contextual, and statistical information into the clustering framework, our team sought to develop a more holistic assessment of team attacking tendencies. Our deliverables to FC Barcelona include a scouting report and python library. This work will be used by FC Barcelona to better understand and prepare for their opponents.

## 2. Data Overview

We were provided 506 distinct events data sets, 380 of which were La Liga matches from the 2016-2017 season and 126 of which were Champions league matches from the 2016-2017 season. Each data set represented a single game, and provided information on the teams playing as well as in-game events that occurred throughout the match. Each data set was cleaned and divided into separate attacks. The data cleaning and processing laid the foundation for later feature engineering. To correctly segment the events data into attacks, we first removed all unsuccessful tackles and unsuccessful aerials. We then ensured that attacks did not start with "Foul", "Out", or "Attempt Saved". By performing this data processing, our group was able to parse the events data into attacks. These attacks were then indexed for ease of access.

### 2.1. Feature engineering

For the clustering model, we represented each attack as a 21-dimensional feature vector. The 21 features fell into 3 categories: macro-events, passing sequence flow motifs, and spatial region features.

**2.1.1. Macro-events.** Macro-events give a high-level overview of the attack. Features in this component describe attack characteristics such as duration, number of passes, speed, directness, and success. The entire list of features in this section are as follows:

- Duration: Total duration of attack in minutes. Calculated as time of last event in attack minus time of first event in attack
- Number of passes: Total number of successful passes in the attack
- Vertical distance: Total distance travelled by the ball in the x-direction (goal to goal)
- Vertical-Horizontal Ratio: Ratio of total distance travelled by the ball in the x-direction to total distance travelled by the ball in the y-direction
- Average attacking speed: Total distance travelled by the ball divided by total time of attack
- Number of long passes: number of successful passes of length greater than 40 meters
- Time per pass: Time of attack divided by number of passes
- Arrival: 1 if team had possession in attacking third at some point in the attack, 0 otherwise

**2.1.2. Passing sequence flow motifs.** Passing sequence flow motifs describe how players interact in an attack[1]. In our iteration, there are no player specific features. This is a possible area for improvement in the future. Instead we use flow motifs for the sequence of players involved. Each attack was divided into overlapping three-pass sequences. For example, a four-pass attack would be divided into sequences 1-2-3 and 2-3-4. For each sequence of three passes, we consider the 4 (not necessarily unique) players involved. We then abstract away player names by denoting each player with a unique letter in that sequence. This means the first unique player involved in a four pass sequence is player A, the second unique player involved is player B, and so on. If a player is involved in multiple passes, their letter is repeated in the sequence. For each attack, the number of occurrences of each flow motif was counted. The feature vector uses the percentage of each motif in the attack. There are 5 motifs (ABCD, ABCB, ABCA, ABAC, ABAB). Below are three examples.

- ABCD: Iniesta → Messi → Suarez → Busquets
- ABCB : Busquets → Messi → Suarez → Messi
- ABCA : Iniesta → Suarez → Busquets → Iniesta

**2.1.3. Spatial Region Features.** Spatial region features provide details on the areas of the field that a team uses in the attack. The field was first divided into a 3x3 grid of equally sized cells. Then, we calculated the percentage of time the ball spent in the defensive, midfield, and in the attacking third. Similarly, we calculated the percentage of time the ball spent on the left side, in the middle, and on the right side. Since each event only has one time associated with it, we assumed that the events happened instantaneously. When consecutive events occurred in separate zones, we attributed half of the time between them to the first zone and half to the second zone regardless of how close to the zone boundaries the events occurred. Additionally, we consider the zone in which possession was lost in. We represent the cell in which possession was lost with 2 components as follows:

- Possession loss x: 0 if defensive third, 1 if midfield, 2 if attacking third
- Possession loss y: 0 if right, 1 if center, 2 if left

**2.1.4. Complete Feature Vector and Scaling Factors.** Because of the varying units associated with the features we used, the order of magnitude of each component varied significantly. Since our clustering algorithm uses the L2 norm between vectors, using an unscaled vector would give large components significantly more weight than smaller ones. To avoid these effects, we scaled each component such that the average value was roughly in the range [0.5,1]. Additionally, for some components we decided to put a limit on the maximum value it could take so that the difference in one component would not dominate all other components. The complete 21-dimensional vector, along with its scaling factor and maximum value (if applicable) is given in the table below:

| Index | Description | Scaling Factor | Maximum Value |
|---|---|---|---|
| 1 | Duration | 1 | |
| 2 | Number of passes | 1/5 | 4 |
| 3 | Total vertical distance | 1/140 | |
| 4 | Vertical-horizontal ratio | 1/2 | 4 |
| 5 | Average attacking speed | 1/1000 | |
| 6 | Number of long passes | 3 | 4 |
| 7 | Time per pass | 15 | |
| 8 | Arrival (binary) | 1 | |

| | | | |
|---|---|---|---|
| 9 | ABAC percentage | 2 | |
| 10 | ABAB percentage | 2 | |
| 11 | ABCA percentage | 2 | |
| 12 | ABCB percentage | 2 | |
| 13 | ABCD percentage | 2 | |
| 14 | Defensive third percentage | 4 | |
| 15 | Midfield percentage | 4 | |
| 16 | Attacking third percentage | 4 | |
| 17 | Left side percentage | 4 | |
| 18 | Center percentage | 4 | |
| 19 | Right side percentage | 4 | |
| 20 | Possession loss x | 1 | |
| 21 | Possession loss y | 1 | |

Table 1. Descriptions of Engineered Features

### 2.2. Clustering algorithm(s)

We proposed two possible clustering algorithms - K-means and E-M Algorithm. The former algorithm makes hard assignments to a single cluster upon convergence and uses the L2 norm (insert equation here) as its distance metric. The latter algorithm makes soft assignments to at least one cluster upon convergence, which means that it computes a probability of how likely a given data point belongs to a certain cluster. Instead of the L2 norm, its objective is to maximize the likelihood a point belongs to a particular cluster. With our chosen clustering method, we tested and evaluated several different metrics for determining the optimal number of clusters to be used. These metrics are outlined as follows:

- Elbow Method: For each cluster size k, calculate the ”within sum of squares”, which is a measure of how close each point is to its assigned cluster's center. The lower the ”within sum of squares”, the closer points are to their respective centers and the better the fit. Then, we plot those values for the range of cluster sizes and look for an ”elbow point”. This is a point where adding an extra cluster does not decrease the within sum of squares by much.
- Average Silhouette Score: For each cluster size k, calculate the average silhouette score, which is a measure of the average score of points' proximity to other points in its cluster compared to distance to nearest cluster. The greater this number is, the better the clusters are. Once we have all the values, we find the cluster size with the highest average silhouette score.
- Gap Statistic: For each cluster size k, calculate the gap statistic, which is a measure of how different the clustering structure is from a random distribution of points. The greater this number is, the better the clusters are so once we have all the values, we

want to find the cluster size with the highest value. Affinity Propagation: An algorithm that finds points that are most representative of other points. Once the algorithm finds all the representative points, the optimal number of clusters will just be the number of representative points.

- Bayesian Information Criterion (BIC) Score: For each cluster size k, calculate the BIC score, which is a measure of the variance within each cluster while penalizing for the number of clusters. This tries to balance the accuracy of the clustering and the complexity of the model. Once we have all these values, we want the cluster size with the lowest BIC score.

After testing these different algorithms and metrics, we decided to train our final model using the E-M algorithm. **We determined the optimal number of clusters being 16 using the BIC score**.

### 2.3. Filtering mechanism

We conduct the filtering algorithm by filtering out different subsets of the full dataset (Figure 1). We first filter by specifying an attacking team of interest (i.e. FCB) to understand each team's specificity in their attacking phase. To understand how team tendencies change depending on their opponent, we also allow the opponent team(s) to be specified. We then try to compare certain tendencies of each team in particular setting (i.e. Home/Away). These simple filtering options, give users the ability to compare breakdowns of a specific team's attacking style across many different situations. After filtering out teams with similar tendencies and attacking statistics, we can cross-validate matches of different teams with different style and how they affect the clustering effect.

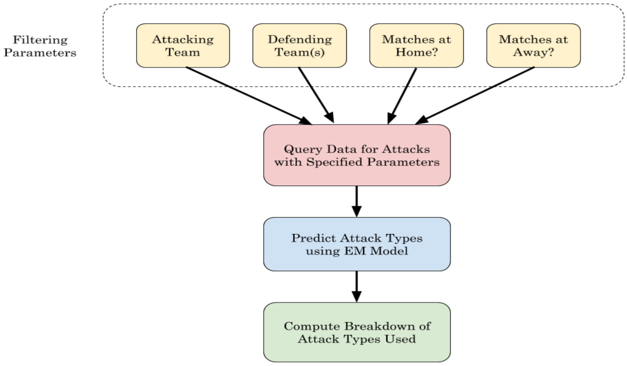

Filtering Parameters
Attacking Team
Defending Team(s)
Matches at Home?
Matches at Away?
Query Data for Attacks with Specified Parameters
Predict Attack Types using EM Model
Compute Breakdown of Attack Types Used

Figure 1. Flowchart of the Filtering System

### 2.4. Analysis of Clusters

Once clusters are generated from the model for the entire data set of the league, textual descriptions for each cluster are generated manually to contextualize the data (in Table 2 below). This is done by first selecting the most indicative attacks for a given cluster, and then analyzing both data points and in-game footage for these attacks to determine characteristics of the entire cluster. The most indicative attacks for a given cluster were deemed to be those attacks which were assigned the highest value in the expectation maximizing model for a given cluster. Each attack is assigned a probability for being in a specific cluster, with the attacks most certainly in a given cluster assigned numbers closer to 1. In future iterations of the model, it is possible that this generation of textual descriptions for clusters be made more automated and objective.

| | |
|---|---|
| 0 | Passing in defensive third and midfield, both sides used equally. Often includes long ball forward towards attacking third. |
| 1 | Short attacks beginning in defensive areas. Team not fully in control, often play long unsuccessful clearance |
| 2 | Quick attacks on right side, all making it to final third. Often ending in crosses |
| 3 | Quick attacks on left side, all making it to final third. Often ending in crosses |
| 4 | Passing in defensive half, losing possession in midfield. Never making it to final third. |
| 5 | Controlled possession in midfield, not making it to final third |
| 6 | Long possessions in midfield with lots of passes. Usually makes it to final third. |
| 7 | Possession in defensive areas and midfield, mainly center and left side. Never making it to final third |
| 8 | Long ball in the air to the final third, either from goalkeeper or defender, fail to get the second ball |
| 9 | Long ball mostly on the right hand side, distributed by the defender or goalkeeper, more success |
| 10 | Clearance from the back, by defenders, clear to sideline, didn't pass through half of the field |
| 11 | Try to break through lines by dribbling, mostly on the left hand side, some successful ending with crossing |
| 12 | Set pieces, throw, corners. Often ending in crosses |
| 13 | Slow backfield build up, pass through lines, ending with crosses |
| 14 | Patient, midfield wide build up, try to play through the lines, try to control tempo, a lot of back pass |
| 15 | In own half try to break out through short pass, rarely make through the final third |

Table 2. Descriptions of the 16 Clusters

## 3. Results/Findings

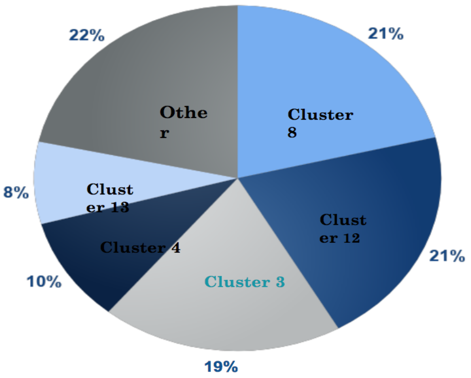
### 3.1. Scouting Report Overview

The scouting report is designed to aggregate meaningful, team-specific conclusions from the clustering model into a concise and quickly digestible format. The central piece of the scouting report is a bar graph indicating the attacking tendencies of a team that diverge most from league average. Since clusters are defined across the entire data set, this graphic provides a quick visual representation of a team's most meaningful attacking tendencies. The other main component of the scouting report is a section listing teams within La Liga with similar attacking tendencies to FCB. This section adds context to the clustering breakdown of the bar graph, and is useful information for coaches and team members with deep prior knowledge on the attacking styles of various teams in La Liga.

### 3.2. Validation and visualization on clusters

*"El Clásico"* refers to the match between FC Barcelona and Real Madrid, its fiercest rival domestically. The figure below shows how our clustering algorithm will help FC Barcelona in terms of breaking down Real Madrid's attacks. The pie chart on the left-hand side shows that Real Madrid attack a lot corresponding to cluster 8, cluster 12 and cluster 3. According to the table on the right-hand side, we can see that FC Barcelona should keep an eye on cluster 3 and cluster 12, in which Real Madrid quickly build up their attacks on both wings and end up with crosses to their center forwards. On the other hand, it seems that FC Barcelona should not worry that much on cluster 8 since they never reach the final third. However, the following league-wise analysis shows that Real Madrid plays much less cluster 8 among all teams in La Liga (Figure 3, 4, and 5).

**Real Madrid Attacking Breakdown**



| CLUSTER | CHARACTERISTICS | LOCATION | RESULT |
|---|---|---|---|
| 3 | Quick | Right Flank | Cross |
| 4 | Quick | Left Flank | Cross |
| 8 | Patient, Many Passes | Center and Left of Defensive and Middle Thirds | Never Reaches Final Third |
| 12 | Dribble through lines | Left Flank | Cross |
| 13 | Set Pieces, Throws, Corners | N/A | Cross |

Figure 2. Sample Attacking Breakdown for Real Madrid

### 3.3. League Trends Overview

The analysis on Real Madrid will not hold without comparing it with other teams in La Liga. On a league-wise basis, we devised a metric called **”net difference from league average”** to evaluate the variability in attacking clusters for each team. It is mathematically defined as the cumulative sum on the absolute value of the difference between the team's percentage in each cluster and the league-wise average percentage on that cluster. Figure 3 shows the net difference from league average among all clusters, ordered by league standing of La Liga 2016-17 season with the champions Real Madrid on the very left. We observe that the values are higher on both ends, which hypothesize that top teams tend to attack in more advantageous clusters while the bottom teams tend to attack in more disadvantageous clusters.

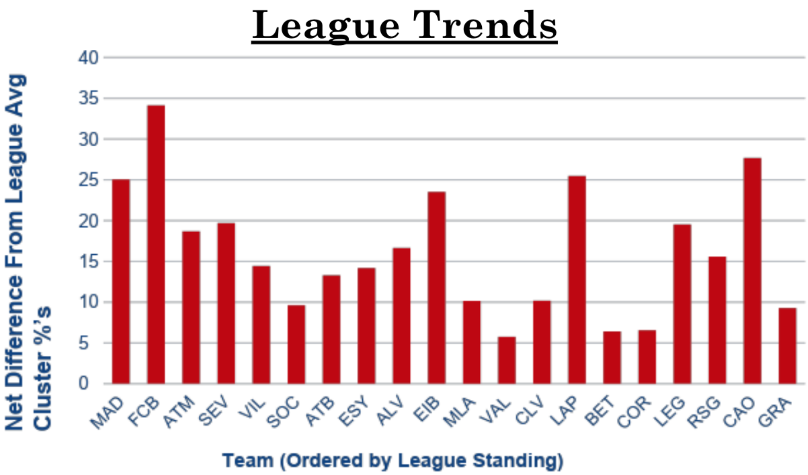
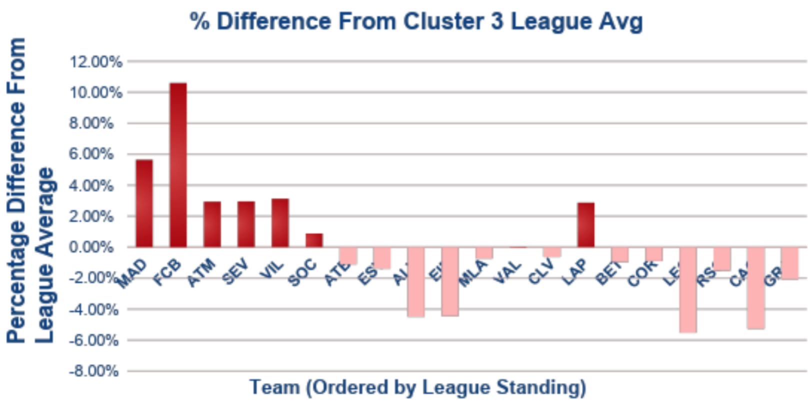


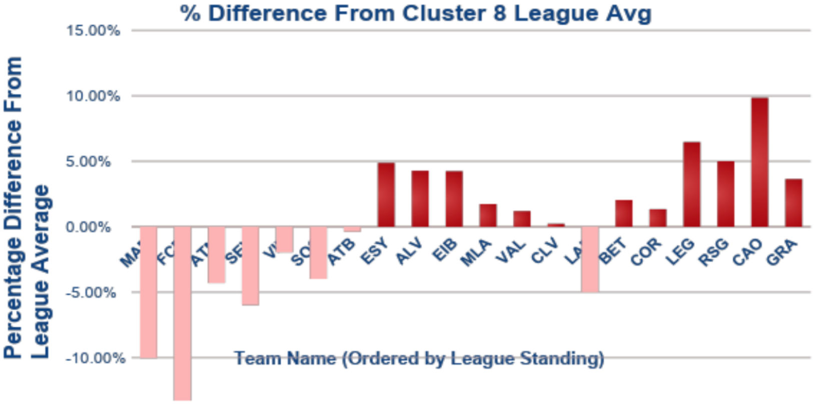
Figure 3. League-wise Net Difference from League Average among All Clusters

Take cluster 3 (an advantageous cluster) for example, in Figure 4 we observe top six teams in the league with positive percentage difference while for median and bottom teams they attack less in this cluster. Also in Figure 5 we can see top teams play relatively less cluster 8 (a disadvantageous cluster) while bottom teams play more. The only team that contradicts the league-wise trend is Las Palmas where we hypothesize that they attack like top teams but they suffer from not having the right players to make their attacks more effectively.



| CLUSTER | CHARACTERISTICS | LOCATION | RESULT |
|---|---|---|---|
| 3 | Quick | Right Flank | Cross |

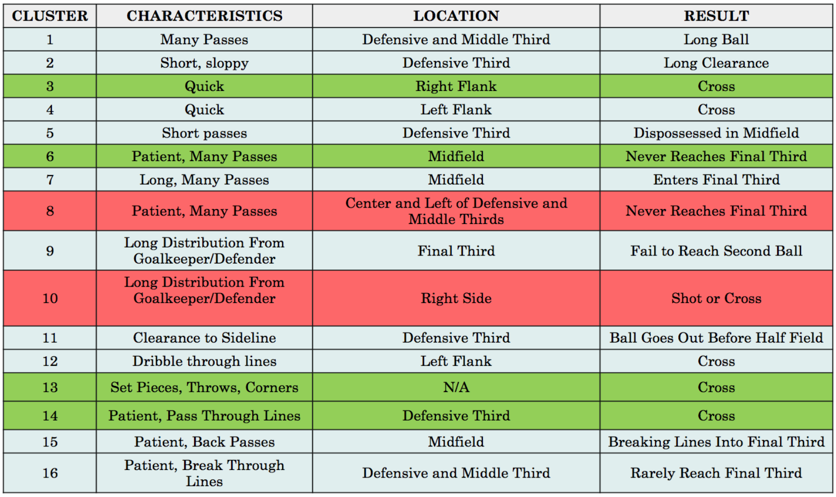
FIGURE 4. League-wise Advantageous Clusters Analysis



| CLUSTER | CHARACTERISTICS | LOCATION | RESULT |
|---|---|---|---|
| 8 | Patient, Many Passes | Center and Left of Defensive and Middle Thirds | Never Reaches Final Third |

FIGURE 5. League-wise Disadvantageous Clusters Analysis

Similarly, we end up with a table (Figure 6) that further breaks down the 16 clusters with respect to characteristics, location and result. Here we mark green the advantageous attack clusters, namely cluster 3, 6, 13, and 14 which are associated with top teams, while cluster 8 and 10 are disadvantageous attack clusters linked to bottom teams.

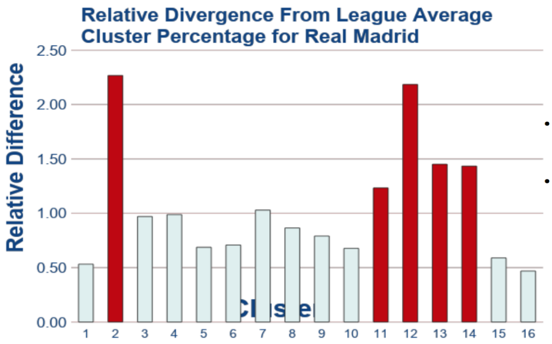


| CLUSTER | CHARACTERISTICS | LOCATION | RESULT |
|---|---|---|---|
| 1 | Many Passes | Defensive and Middle Third | Long Ball |
| 2 | Short, sloppy | Defensive Third | Long Clearance |
| 3 | Quick | Right Flank | Cross |
| 4 | Quick | Left Flank | Cross |
| 5 | Short passes | Defensive Third | Dispossessed in Midfield |
| 6 | Patient, Many Passes | Midfield | Never Reaches Final Third |
| 7 | Long, Many Passes | Midfield | Enters Final Third |
| 8 | Patient, Many Passes | Center and Left of Defensive and Middle Thirds | Never Reaches Final Third |
| 9 | Long Distribution From Goalkeeper/Defender | Final Third | Fail to Reach Second Ball |
| 10 | Long Distribution From Goalkeeper/Defender | Right Side | Shot or Cross |
| 11 | Clearance to Sideline | Defensive Third | Ball Goes Out Before Half Field |
| 12 | Dribble through lines | Left Flank | Cross |
| 13 | Set Pieces, Throws, Corners | N/A | Cross |
| 14 | Patient, Pass Through Lines | Defensive Third | Cross |
| 15 | Patient, Back Passes | Midfield | Breaking Lines Into Final Third |
| 16 | Patient, Break Through Lines | Defensive and Middle Third | Rarely Reach Final Third |

FIGURE 6. Summary of Advantageous and Disadvantageous Attack Clusters

### 3.4. Real Madrid Case Continued: Sample Scouting Report

A more mature sample scouting report are as follows. For Real Madrid in season 2016-17, in Figure 7, we illustrate the distribution of **relative difference** defined by the ratio of percentage in each cluster by Real Madrid against the average percentage in that cluster of the entire league. The insights from this graph are from two aspects. On one hand, it extracts clusters which stray most from league average. On the other hand, it accounts for less common attacks by utilizing the standard deviation of clusters.



- Extract Clusters which stray most from league average
- Account for less common attacks by utilizing standard deviation of clusters

FIGURE 7. Real Madrid: Sample Scouting Report on Cluster Distributions

Incorporating the contents in Figure 7 with our text description of clusters, we will end up with our final version of sample scouting report shown in Figure 8. The bars shows the comparison in between Real Madrid and the league average in terms of some important clusters such as "*Dribble to break lines, left side, ending in cross*" and "*Set pieces, throw, corners, often ending in crosses*". The team coaches should have an overview of how different Real Madrid's attacking style compared with other opponents.

On the top-right corner, we generate the teams (FC Barcelona, Real Sociedad, Sevilla and Atlético de Madrid) that plays similarly with Real Madrid. This feature makes the scouting report more helpful for coaches at FC Barcelona better prepare *El Clásico* via making references to a broader group of La Liga teams.

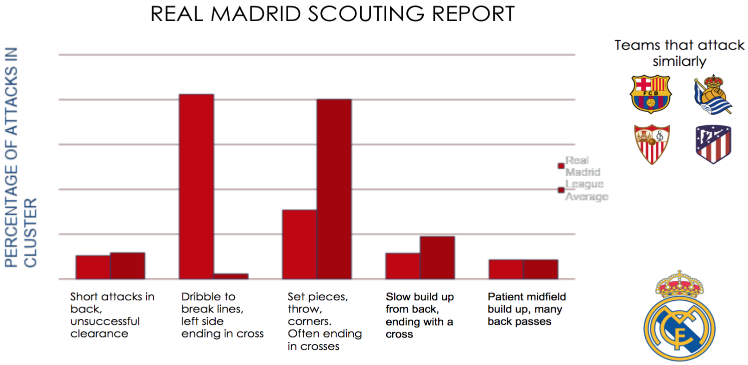


FIGURE 8. Real Madrid: Sample Scouting Report Final Version

## 4. Python Library Overview

Our final deliverable for the data science team at FC Barcelona is a Python 2 library so that they can perform the same clustering analysis we did and gain key insights into opponent attacking tendencies that can be passed on to the coach. The following outlines our final Python library tested with Python 2.7.

### 4.1. Prerequisite Python Libraries

- pandas
- matplotlib
- numpy
- sklearn
- xml
- re
- os

### 4.2. Library Files

- `xml_to_csv.py`: Converts the initial raw event xml data from Opta into easier compatible csv files for our other scripts. The original xml data had event tags for each event with different child tags for each event depending on what type of event it was. Our conversion method essentially flattens all possible child tags into columns so that the whole event becomes one row in a csv file.

- csv_to_features.py: Gets event data files corresponding to the teams of interest, parses each file for attacks, and computes the feature vector for that attack containing our specified 21 features. Please refer to the appendix for details of functions.
- cluster_training.py: Trains a Gaussian Mixture Model using the EM clustering algorithm on all the attacks from the full season's worth of data and saves it along with all complementary data files as pickle files to be used in generating results. Please refer to the appendix for details of parameters.
- cluster_results.py: Uses the trained model to generate results regarding each team's breakdown of attacks used and lists of most representative attacks from each cluster. Please refer to the appendix for details of parameters.
- filtering_results.py: Gets attacks from specified teams against specified opponents using the filtering functions and uses the trained model to classify the attacks and compute the breakdowns for that situation. Please refer to the appendix for details of parameters.

## 5. Conclusion

The goal of this project was to produce meaningful insights into opponent attacking tendencies from data aggregated across an entire season of attacks. We believe that our model is able to accomplish this task through a holistic rendering of attack characteristics, including both contextual, spatial, and temporal features. In addition, the sample scouting report which we provide gives an indication into how this model can be used to deliver results in a actionable and digestible way for coaches and players alike. Together, the clustering model and scouting report provide insights into team attacking tendencies and develop a framework for FC Barcelona to better understand their opponents. We believe that this project is a step in the right direction for applying data analytics to the soccer pitch.

## References

[1] L. Gyarmati, H. Kwak, P. Rodriguez, *Searching for a Unique Style in Soccer.* arXiv:1409.0308 (2014)

## Appendix

**Library Files Details.**

- Functions in `csv_to_features.py`:
  - dict_to_feature_vector(attack_dict)
  - dict_to_spatial_sequence(attack_dict)
  - get_filenames(directory, attacking_team, opponents=None, home=True, away=True)
  - get_attacks(attacking_team,directory,filenames,min_time=0.08)
  - time_duration(attack_dict)
  - number_of_passes(attack_dict)
  - total_vertical(attack_dict)
  - total_horizontal(attack_dict)
  - end_box(attack_dict)
  - if_arrival(attack_dict)
  - flow_motif_occurences(attack_dict)
  - total_distance(attack_dict)
  - average_attacking_speed(attack_dict)
  - number_long(attack_dict)
  - number_short(attack_dict)
  - vertical_horizontal_ratio(attack_dict)
  - pctg_time_back(attack_dict)
  - pctg_time_midfield(attack_dict)
  - pctg_time_last_third(attack_dict)
  - pctg_time_left(attack_dict)
  - pctg_time_center(attack_dict)
  - pctg_time_right(attack_dict)
  - possession_loss(attack_dict)
- Parameters in `cluster_training.py`:
  - DATA_DIRECTORY: relative location of the full season's worth of event data in csv format
  - TEAM_ID_FILE: relative location of the csv file mapping team id numbers to team names
  - NUM_FEATURES: number of features being used (21 in our case)
  - BEST_K: optimal number of clusters to be used
  - BEST_PARAM: optimal covariance parameter to use out of 'full', 'tied', 'diag', 'spherical' options
  - data_from_pickle: boolean flag set to True if loading data from previously saved pickles for quicker data processing
- Parameters in `cluster_results.py`:
  - NUM_FEATURES: number of features being used (21 in our case)
  - BEST_K: optimal number of clusters to be used
  - TOP_N: number of most representative attacks to output for each cluster
  - REP_DIREC: relative directory location to output the csvs of lists of most representative attacks for each cluster

  - VID_DIREC: relative directory location to output the csvs of lists of most representative attacks for each
  - cluster that we have video for as specified from the list video_matches
  - DIST_DIREC: relative directory location to output the pie charts of each team's breakdown of types of attacks
  - video_matches: lists of tuples specifying home and away teams of matches we have video for
- Parameters in `filtering_results.py`:
  - DATA_DIRECTORY: relative location of the full season's worth of event data in csv format
  - BEST_K: optimal number of clusters to be used
  - NUM_FEATURES: number of features being used (21 in our case)
  - ATTACKING_TEAM: name of attacking team of interest
  - OPPONENTS: lists of names of opponent teams of interest
  - HOME_FLAG: True if including ATTACKING_TEAMs home matches
  - AWAY_FLAG: True if including ATTACKING_TEAMs away matches

Eric Bradford, Alex Cauneac, Emma Chesley, Sean Ko, Kevin Lee, Garrett Souza, and Tim Yang