Università degli Studi di Milano

Data Science for Economics

UNIVERSITÀ
DEGLI STUDI
DI MILANO

Statistical Learning project

"Occhio malocchio"

Can Statistical Learning help more than superstitious rituals to understand football?

Student:
Leonardo Acquaroli

ACADEMIC YEAR 2022-2023

Abstract

In this work Supervised and Unsupervised Statistical learning techniques are used in order to discover useful insights in to areas of interest of the football sphere.

1. Supervised algorithms were used to develop an **Expected Goal (xG) model**, that is the estimation of the probability that a shot ends in the net.

2. Unsupervised clustering algorithms to build a **Role detector** that can assign a role to a player not only based on their position but also taking into account other 27 in-game metrics like touches, shots on target, shots-creating-actions, etc.

Dataset

Two datasets were exploited as described in details in the subsections below. The first one comes from a publication by L. Pappalardo and P. Cintia, coordinated by F. Giannotti and D. Pedreschi[3], which, with the help of some professionals by the sport-tech company **Wyscout, made public the biggest dataset of football *events data***. An *event*, in football analytics, is a very granular action of the game that can be recorded and thus analyzed and enriched with a lot of information. For example, the most common *event*, that is the pass, contains in a row: the starting location in (x, y) coordinates, the end location, the type of pass, the player receiving the ball and much more. Wyscout's dataset hosts 3'251'294 rows (*events*) and 12 columns and it refers to the 2017/2018 season of the five most famous leagues (Serie A, Bundesliga, La Liga, Premier League, Ligue 1), to the 2016 European championship and eventually to the 2018 World cup.

The second dataset is the **FBref repository that is full of aggregated metrics**, like the abovementioned, ready to be used. The provider of those data is StasPerform through Opta, one of the leaders in this market. FBref data can be scraped automatically via the library worldfootballR but for computational and technical issues with the library functions the retrieved metrics only involve Serie A, La Liga and Premier League's *events*. Still the stastics on more than 1000 players were accessed and so an adequate number to perform Statistical learning techniques.

Data to train the xG model

The xG model was trained using a subset of all the Wyscout's *events* containing only shots, enriched with some information from the *events* occurred before. In particular it was stored in a $43'054 \times 25$ dataframe that contained some identification variables: eventId, subEventName, tags, playerId, positions, matchId, eventName, teamId, matchPeriod, eventSec, subEventId, id, end_x, end_y; and eleven variables that have been used to predict the Expected Goals.

- ***start_x***: The shot's x coordinate on 100x100 pitch (as supplied by Wyscout), where the coordinates are expressed as percentage of the pitch length. The x coordinate has to be seen as a point in the longest side of the pitch.
- ***start_y***: The shot's y coordinate.
- ***fromSmart_pass***: 1 if the previous *event* was a smart pass, 0 otherwise.

- ***fromCross***: 1 if the previous *event* was a cross, 0 otherwise.
- ***fromSave***: 1 if the previous *event* was a save by the golkeeper, 0 otherwise.
- ***fromDBL***: 1 if in the previous 6 *events* a Dangerous Ball Lost was recorded, 0 otherwise.
- ***head_OR_body***: 1 if the shot was a header or a hit with a part of the body different from foot, 0 if it was a kick.
- ***strongFoot***: 1 if the shot was taken with the preferred foot, 0 otherwise.
- ***distance_to_goal***: Euclidean distance from the center of the goal.
- ***angle***: The shooting angle between the two goal posts and the shooting player, calculated as $\theta = \arctan\left(\frac{7.32x}{x^2+y^2-(7.32/2)^2}\right)$, transformed in positive when it was negative and then multiplied by $\frac{180}{\pi}$ passing from radians to degrees. For the formula explanation see [2].
- ***Goal***: The target variable that shows if the shot was a goal (1) or not (0).

Some descriptive statistics are now reported. First the boxplots for the distributions of the continuous variables: distance, angle, x and y coordinates.

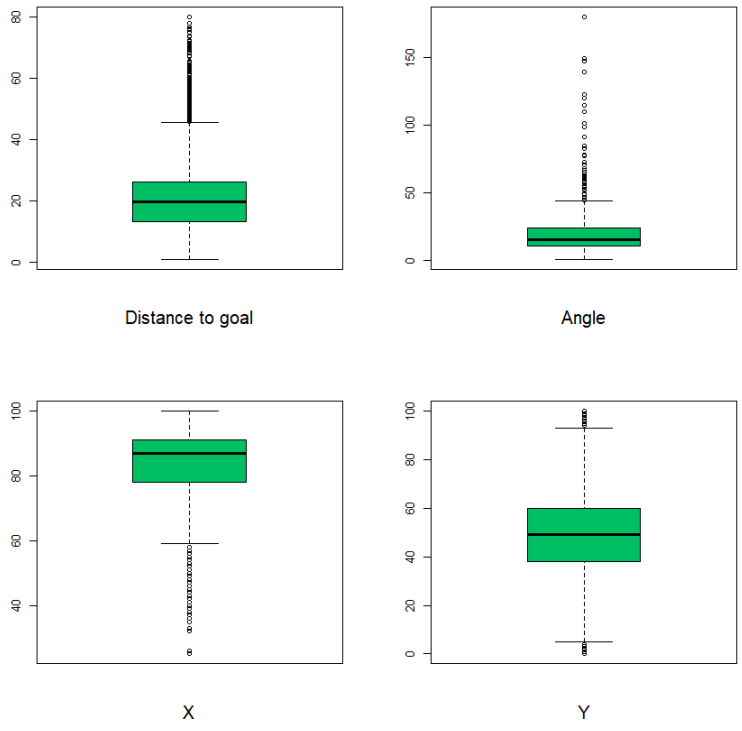


Figure 1: Boxplots for continuous variables.

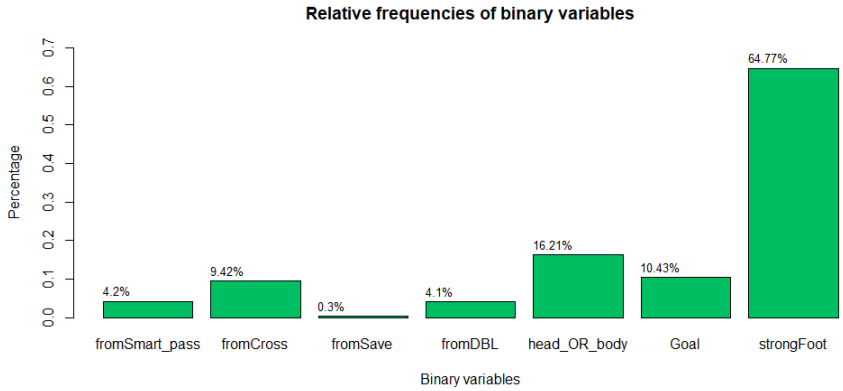


Figure 2: Percentage of 1 in the binary variables.

On the basis of those boxplots the shots with *start_x* smaller than 20 were dropped and treated as mistakes, since it would mean a shot, more or less, from a player's own penalty area.
Then there are the histograms for the binary variables.

Finally we have a correlogram for all the variables.

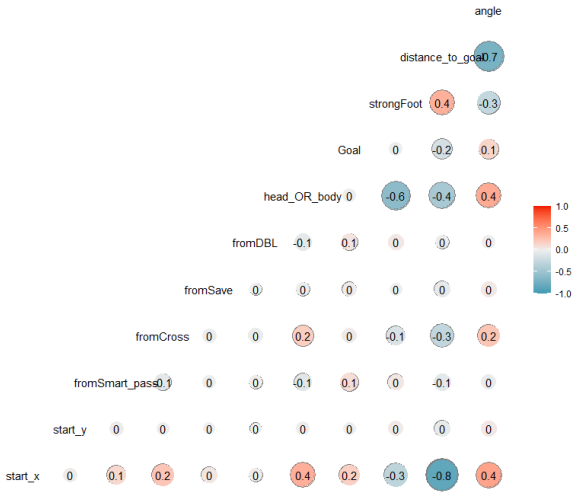


Figure 3: Correlation plot among all the variables of the dataset.

The very last descriptive information is that the dataset is similar, in terms of balance between Goals and Non-goals shots, to the empirical evidence that **scoring a goal takes on average 10 shots**, firstly brought to the public by the father of the football analytics Charles Reep in the work that is considered as the foundation paper of the subject [5].

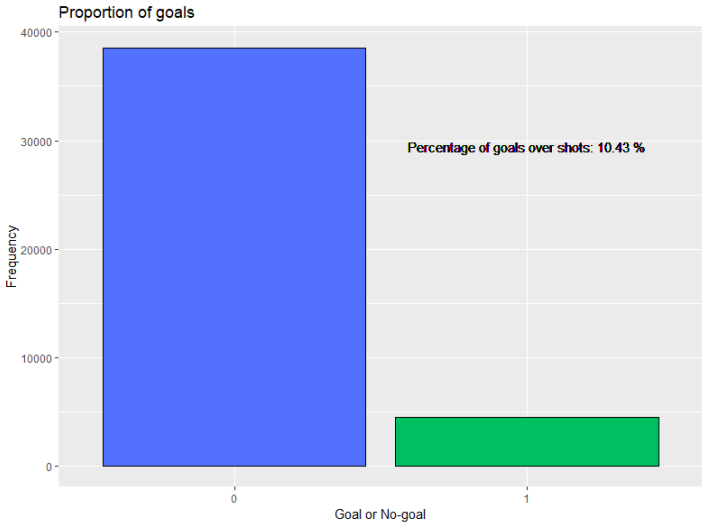


Figure 4: Percentage of goals in the shots

To train the supervised algorithms the complete dataset was split into a training set (70% of the observations) and a test set (remaining 30%).

Data for the Role detector

While the xG models only need *events* data for thier training, the Role detector module needed a merge of the two datasets in order **to combine** the mean positions of players' *events* (that we will call **Centers of Performance**) **and the aggregated metrics** in the FBref dataset.
The resulting merged dataset, thus, contains 29 covariates, 24 of which come from FBref and were normalized using *p90* normalization[1]:

- ***mean_x*** : the x coordinate a player's Center of Performance (from Wyscout's data)
- ***mean_y*** : the y coordinate a player's Center of Performance (from Wyscout's data)
- ***Goals*** : Goals *p90*
- ***Assists*** : Assists *p90*
- ***Penalties*** : Penalties scored *p90*
- ***Penalties_att*** : Penalties attempted *p90*
- ***Shots*** : Shots *p90*
- ***Shots_oT*** : Shots on Target *p90*

[1] *p90* means «per 90 minutes».
The formula to normalize is indeed: $p90 \quad Metric = \frac{Raw \quad metric}{Min. \quad played} \cdot 90$

- ***Ycard*** : Yellow cards *p90*
- ***Rcard*** : Red cards *p90*
- ***Touches*** : Touches *p90*
- ***Tackle*** : Tackles *p90*
- ***Interceptions*** : Interceptions *p90*. They differ from block because they are recorded as a voluntary movement to intercept the ball
- ***Blocks*** : Blocks *p90* are instead when a players gets hit by the ball (for example a shot)
- ***xG*** : Opta's Expected Goals *p90*. They do not include penalties shotouts but include in-game penalties
- ***npxG*** : Opta's non-penalty Expected Goals *p90*
- ***xAG*** : Expected Assisted Goals, i.e. the xG from a pass the preceeds a shot
- ***SCA*** : Shot Creating Actions
- ***GCA*** : Goal Creating Actions
- ***Passes_cmp*** : Completed passes *p90*
- ***Passes_att*** : Attempted passes *p90*
- ***Passes_prg*** : Progressive passes *p90*, that are passes who close the distance with the opponents' goal of at least the 25%
- ***Carries*** : Ball carries *p90*
- ***Carries_prg*** : Progressive carries *p90*, i.e. carries that get the ball closer to the opponents' goal. They are counted only if they end in the opponents' half.
- ***TakeOns_att*** : Attempted take-ons *p90*
- ***TakeOns_succ*** : Succesfull take-ons *p90*
- ***Weight*** : Player's weight (from Wyscout's data)
- ***Height*** : Player's height (from Wyscout's data)
- ***Rfoot*** : 1 if right-footed, 0 if left-footed (from Wyscout's data)

Since the data are provided by two different companies (Wyscout for the *events* and Opta for the metrics) the join was quite painful, because it had to be done by matching players' names that, along with the Murphy's law, are recorded in different formats by the two providers. This resulted in the lost information for about 100 players. Nonetheless, the most famous and, above all, the ones with more minutes were kept. Data were then filtered to exclude players who played less then 450 minutes thus leading to the final dataset of 1056 players.

xG model (Supervised)

Introduction

Expected Goal is the queen of the football metrics developed in the last 10 years because it is easy to understand and is the most direct figure that can be used to say whether a team or a player deserved more or less for the performance played. The decision about the variables used to estimate the models come from the FBref blog in which they explain xG appearing their data. It has to be clearly stated that the aim of such models is not to exactly predict when a shot will be a goal and when not but rather to generate a probability and, above all, to dig into the features importance. In fact, while for the analysts xG is a way to analyze teams' work and players' performances, for the footballers the focus falls on the actions and the positions they should take or avoid in order to improve the likelihood of their shots ending in the net. For these reasons, each model section will be composed of a screening of the features importance and then the evaluation metrics.

Logistic model

The first model is the most commonly used to calculate probability and in particular xG.

The first step to build the logistic model is to check for problems in the variables like multicollinearity and outliers (both for the model and the univariates).

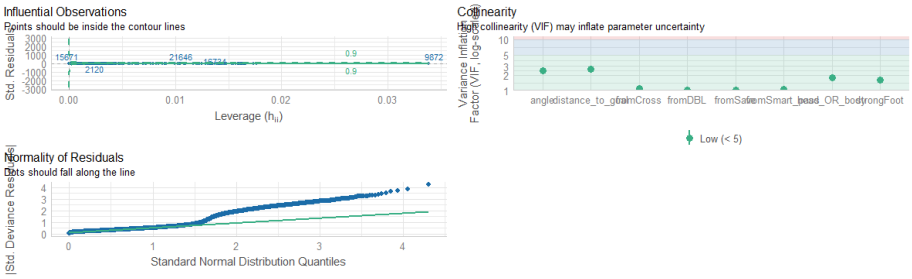


Figure 5: Diagnostics of Logistic model: no problems arise.

Table 1: Logistic Regression Coefficients

Variable	**Estimate**	**Std. Error**	**z value**	**Pr($>\|z\|$)**
(Intercept)	-0.217	0.126	-1.730	0.084
fromSmart_pass	0.557	0.079	7.011	< 0.001 ***
fromCross	-0.003	0.058	-0.049	0.961
fromSave	0.520	0.256	2.030	0.042 *
fromDBL	0.885	0.080	11.040	< 0.001 ***
head_OR_body	-0.824	0.067	-12.311	< 0.001 ***
strongFoot	0.140	0.053	2.670	0.008 **
distance_to_goal	-0.134	0.005	-26.358	< 0.001 ***
angle	0.013	0.002	7.344	< 0.001 ***
Null deviance: 20083 on 30137 degrees of freedom				
Residual deviance: 16921 on 30129 degrees of freedom				
AIC: 16939				

Features importance

We first analyze the impact of the variables through the coefficients Table 1. Even if, for a logistic regression, those estimates cannot be seen as marginal effects their sign and magnitude (compared to their measurement unit) can easily point out which are the most influential and in which direction. We can see that they all follow common sense and in fact, the positive coefficients are linked to the shooting angle (0.013) and to the binary variables indicating when a shot is taken with the strong foot (0.140), occurs after a Dangerous Ball Lost (0.885) or comes from a rebound of the opposing goalkeeper (0.520); the negative ones are related to the distance (-0.134), shots taken with body parts different from feet (-0.824) and shots coming after a cross (-0.003) though this last one is not significant and very close to zero. A second specification was tried excluding *fromCross* but the results were very similar both in performances and variables influence.

Evaluation metrics

After having built the AUC-ROC plot (Figure 6) and, through this, found the best threshold beyond which to label a shot as a goal ($xG \geq 0.1119$), MSE (Means Square Error), Accuracy, Precision and Recall were computed and they are displayed below in Table 2.
For the logistic model the Deviance is also reported since it is better suited than the MSE.

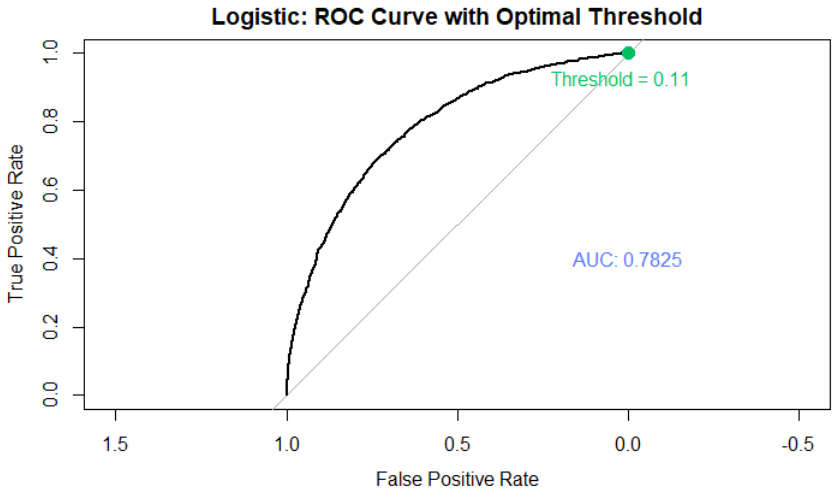


Figure 6: The Optimal threshold is the value the allows the model to reach the highest and right-most point in the ROC.
AUC is the value of the Area Under the Curve: the closest to 1, the better the model's predictions.

Table 2: Logistic Evaluation Metrics

Metric	**Accuracy**	**Precision**	**Recall**	**MSE**	**Deviance**
Value	0.725	0.232	0.698	0.082	16921

Finally for a more thorough understanding of the predictions the confusion matrix (for the threshold of 0.1119) is reported.

Table 3: Logistic model: confusion matrix

	Actual	
Predicted	**No goal**	**Goal**
No goal	8411	411
Goal	3142	952

Decision tree

Decision tree represented an alternative to the usual logistic model. Yet, its performances were actually slightly worse than the ones of the previous model. See Figure 8, Table 4 and Table 5.

Features importance

As we can see from Figure 7 the decision tree also looses a bit of interpretability since it performs splits using only two variables.

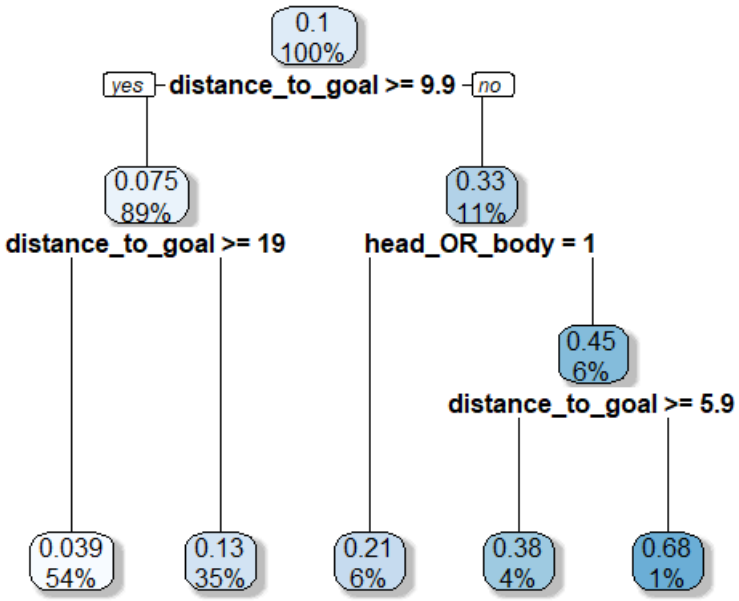


Figure 7: Graphical representation of the tree. It only uses two variables, the distance and the shot taken with head or body, to perform the splits.

Evaluation metrics

The only metric that the tree manages to improve is the recall, i.e. it guesses more goals but at the cost of producing a lot of false positives, making Precision fall.

Table 4: Decision tree Evaluation Metrics

Metric	**Accuracy**	**Precision**	**Recall**	**MSE**
Value	0.604	0.183	0.792	0.085

Table 5: Decision tree: confusion matrix

	Actual	
Predicted	**No goal**	**Goal**
No goal	6727	283
Goal	4826	1080

Random forest

The random forest algorithm was run *"ensambling"* 500 trees and, since this is set as a classification problem, using exactly $3 \approx \sqrt{8}$ predictors at each split, where 8 is the number of variables in the dataset.

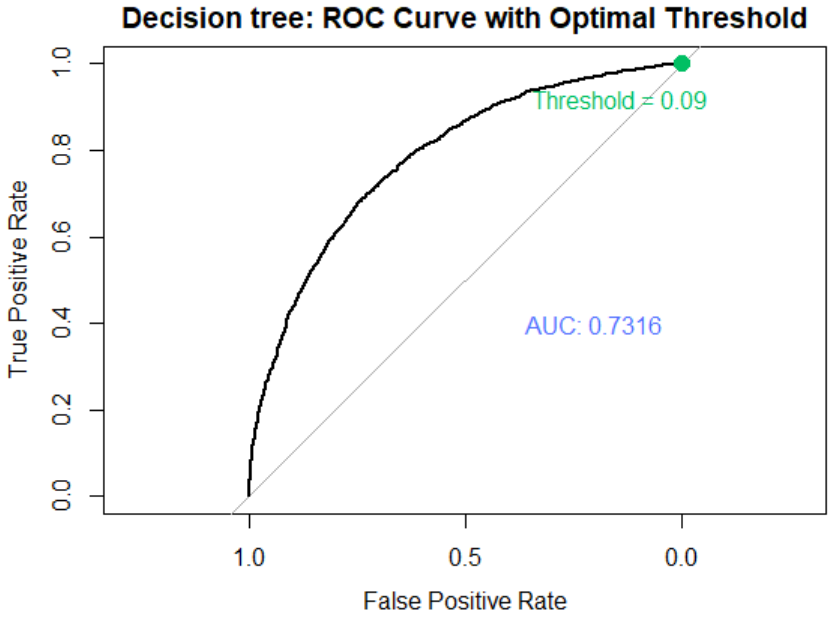


Figure 8: AUC is smaller than the one with logistic and the threshold to classify a shot as a goal is even smaller than the tiny one in the previous model.

Features importance

Variables importance for random forests is calculated as the mean impact that the splits involving each variables have on two metrics: the accuracy and the leaves purity measured through the Gini entropy.

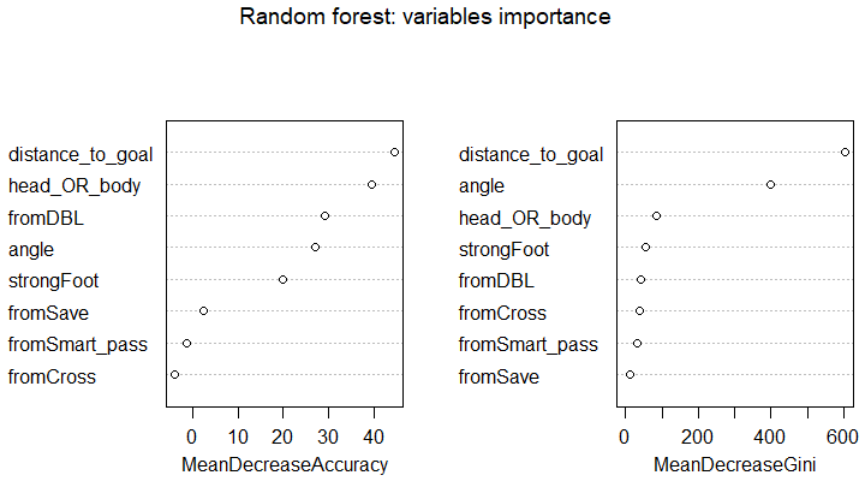


Figure 9: Distance to goal is the most important variable overall. Yet the other variables are shuffled in the two different measurements.

Evaluation metrics

Random forest achieves a great Accuracy and Precision metrics at the expense of the Recall (Table 6), i.e. it is more precise in guessing whether a shot is not a goal but finds extremely hard to guess when a shot turns into goal (see Table 7.

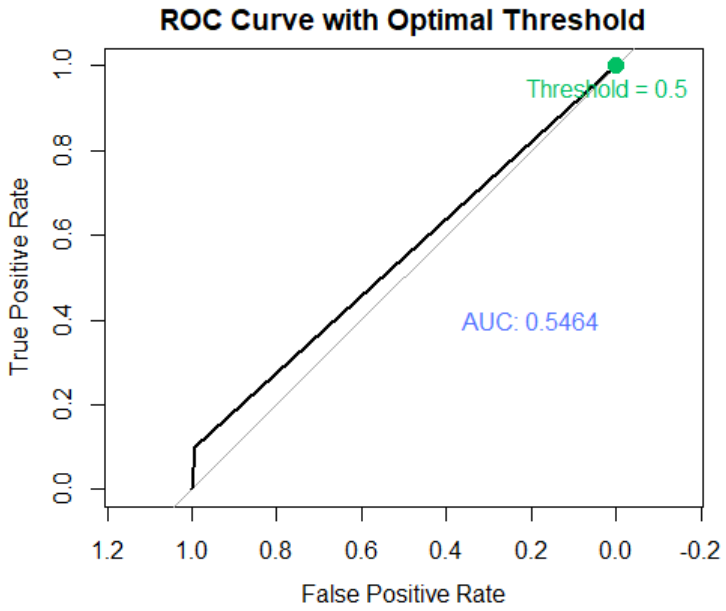


Figure 10: Random forest ROC is the worst of all the models, the AUC is the smallest.

Table 6: Random forest Evaluation Metrics

Metric	Accuracy	Precision	Recall	MSE
Value	0.898	0.603	0.100	0.102

Table 7: Random forest: confusion matrix

	Actual	
Predicted	**No goal**	**Goal**
No goal	11463	1126
Goal	90	137

XGBoost

The last attempt to outperform logistic regression was Extreme Gradient Boosting (XGBoost). This is an algorithm well know for having exceptional perfor-

mance in some Machine learning tasks and, indeed, it is the most used and winning in Kaggle's competitions [1].

Features importance

The XGBoost features importance evaluation shows the same result of the random forest for the most influential variable that is the distance to the goal.

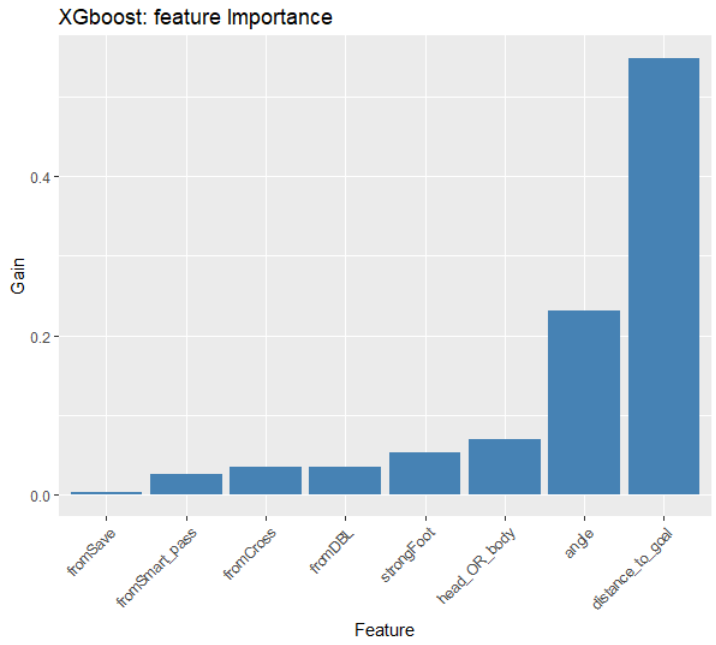


Figure 11: Distance and angle, that are also the founding variables of each xG model, are the most important.

Evaluation metrics

Unfortunately the XGBoost does not manage in generating better predictions than the logistic model.

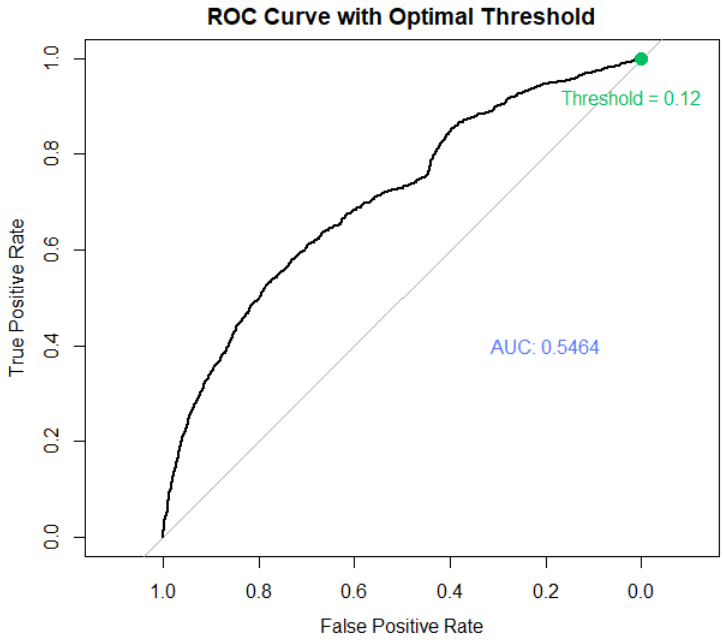


Figure 12: The AUC is the same as using the random forest.

Table 8: XGBoost Evaluation Metrics

Metric	**Accuracy**	**Precision**	**Recall**	**MSE**
Value	0.754	0.222	0.532	0.095

Table 9: XGBoost: confusion matrix

	Actual	
Predicted	**No goal**	**Goal**
No goal	9015	638
Goal	2538	725

Conclusion

To conclude this first part of the project, since the aim was to estimate the probabilities of shots ending in goals, the logistic model was chosen in order to assign to each shot its xG value. Moreover, the logistic model achieved the best MSE (0.082) keeping Accuracy and Recall near the 70%.
Finally the xG produced by each player were summarized and shown in a leaderboard in Table 10 with no surprises at all for the names at its top. Yet, before discover the best forwards of '17/'18 a disclaimer must be done. Summing xG goals is a well spread practice in the field of football analytics though we are summing probabilities assuming that they are always independent. This can be false when we take into account, for example, two shots in the same match or

even in the same action.
What reassures is the number of observations for each player and the fact that the leaderboard order is more or less consistent with other xG providers.

Table 10: Players and their xG Sum

xG sum	**Player**
25.2	C. Ronaldo
24.7	H. Kane
22.2	R. Lewandowski
20.3	L. Suárez
19.4	L. Messi
19.2	E. Cavani
19.2	M. Salah
18.6	E. Džeko
18.5	R. Lukaku
15.3	M. Icardi

Role detector (Unsupervised)

Introduction

The concept of Role detector stems from another work of the abovementioned P. Cintia and L. Pappalardo [4]. Yet, the researchers developed a simple model that clusters players only on the basis of their Centers of Performance (i.e. their events' mean position) that is replicated in Figure **??** on the '17/'18 data.
Nowadays there is a **ongoing debate** in the football analytics industry **about the concept of *role***. Indeed, other researchers developed **new clustering models that take into account, not only the position of players, but also the actions** they perform and so the associated metrics. One excellent example is the book "The Clustering project", by the sport-tech company Soccerment [6].

In this second part of the project, different clustering techniques were put in place in order to find different clusters that represent a new concept of *role*.

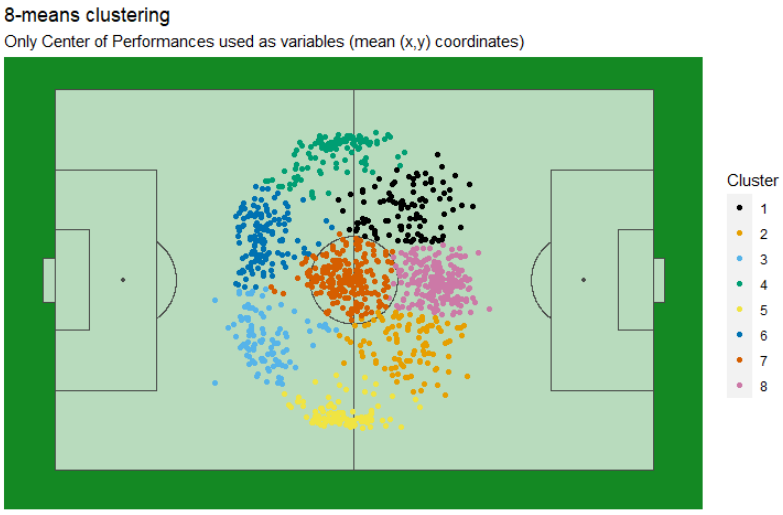
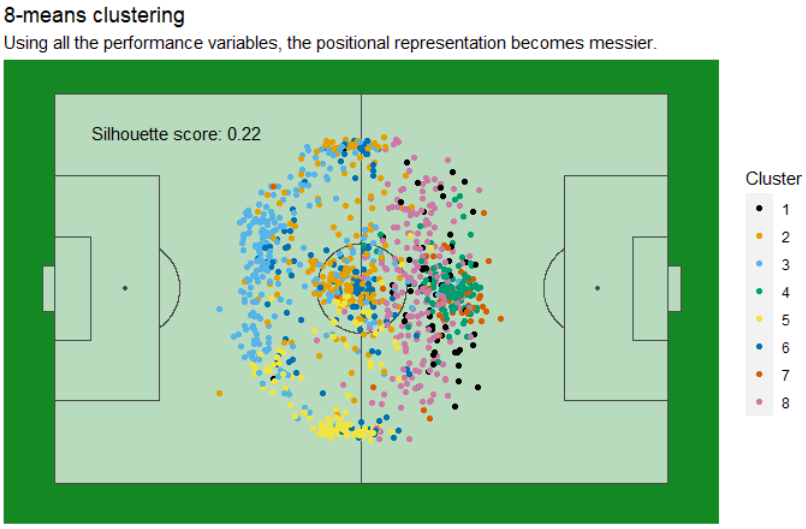
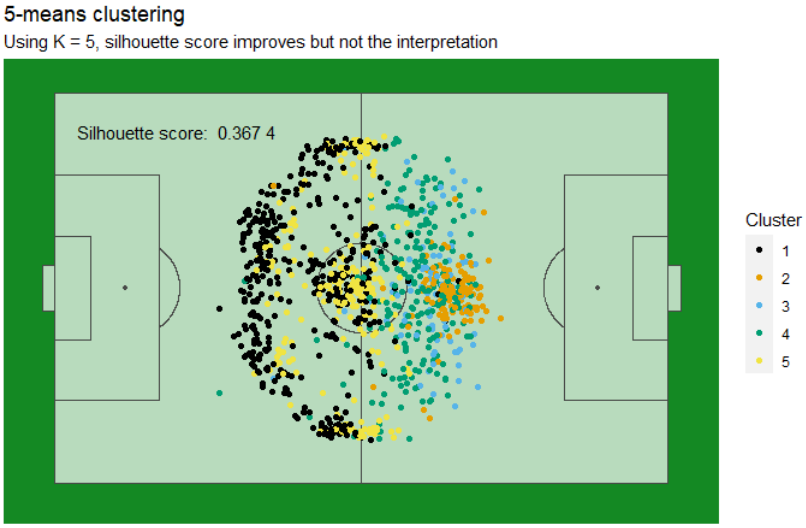


Figure 13: The replication of Cintia's and Pappalardo's experiment.
K = 8 was chosen to match that of the researchers

K-means

The first attempt made is a K-means clustering performed over the entire set of variables. The best K, found minimizing the WCSS (Within-Clusters Sum of Squares) and maximizing the Silhouette score, was 5 but to have a comparison with the just mentioned clustering of Cintia and Pappalardo also the 8means plot in two dimensions $(mean\_x, mean\_y)$ is shown.





It is clear that, augmenting the dimensionality, the mere positional representation is not so useful anymore.

K-means on PCA projections

The football world is obviously populated of few people that can and want to understand the statistics beside such methods. Thus, arises the need of a easier-to-visualize (and also explain) solution. So the same K-means algorithm was run on the projections of each observation onto the two principal components.

PCA

The Principal Component Analysis, extracted two directions onto which projecting all the variables keeping as much variance as it can.

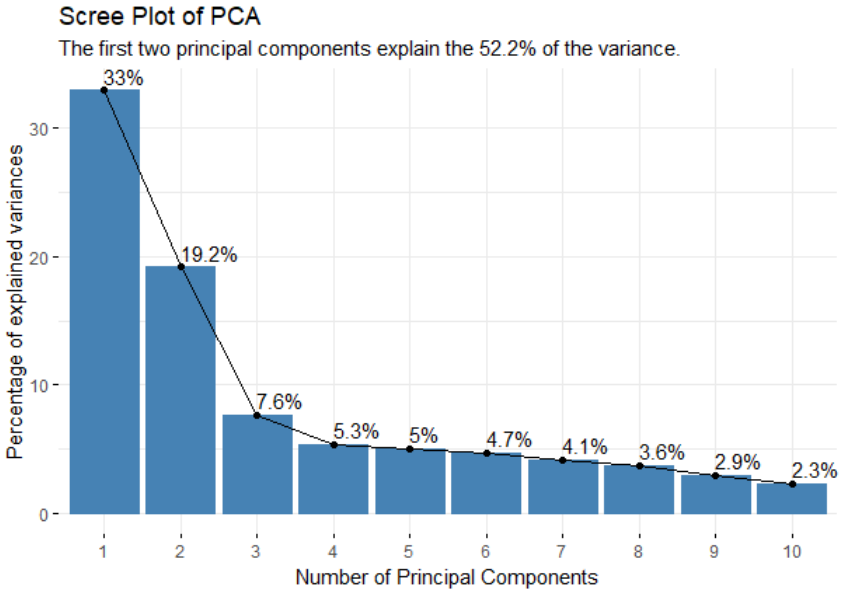


Figure 14: A third component could be used but that would have complicated the plot and the clusters' explanation

The results of the dimensionality reduction can be seen in details in Figure 15 and in Figure 16.

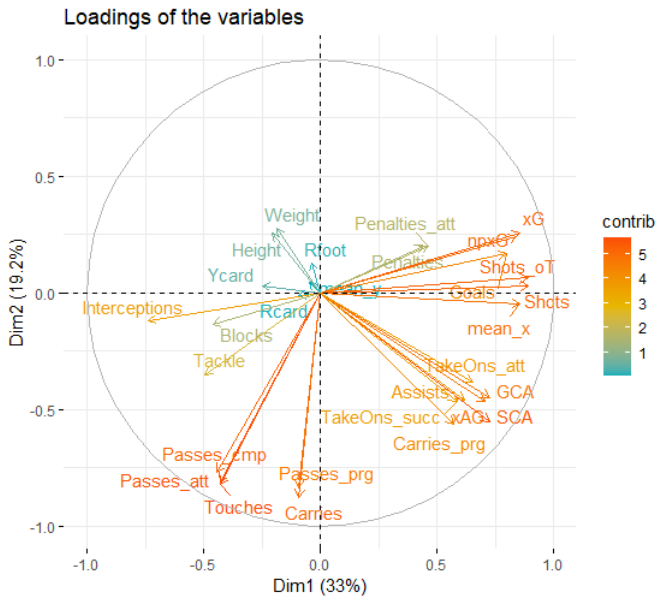


Figure 15: Normalized to 1 loadings of the variables onto the principal components

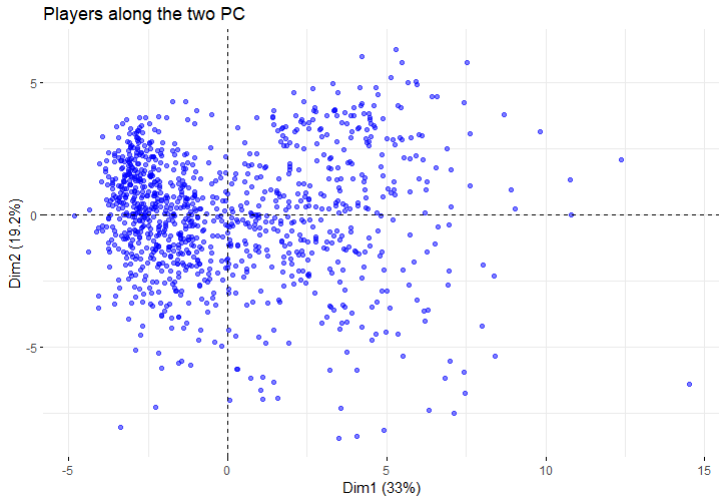


Figure 16: Scatterplot over the two principal components

K-means application

Now that the data can be expressed as bivariate K-means algorithm is run again, still obtaining a best K value of 5 (Figure 17).

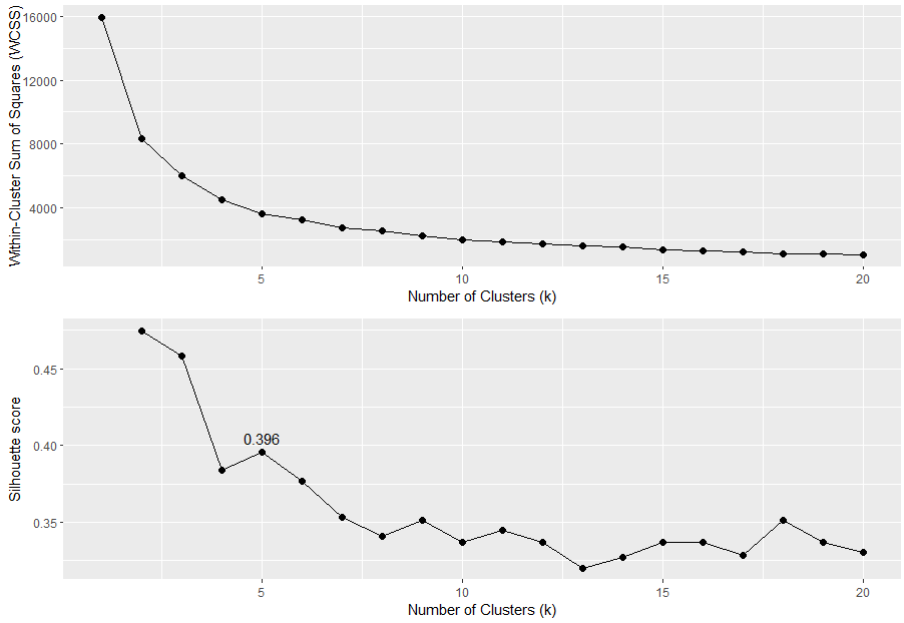


Figure 17: The two screeplots to apply the *elbow rule*

The final visualization summarise all the previous information in one easier-to-digest plot (Figure 18).

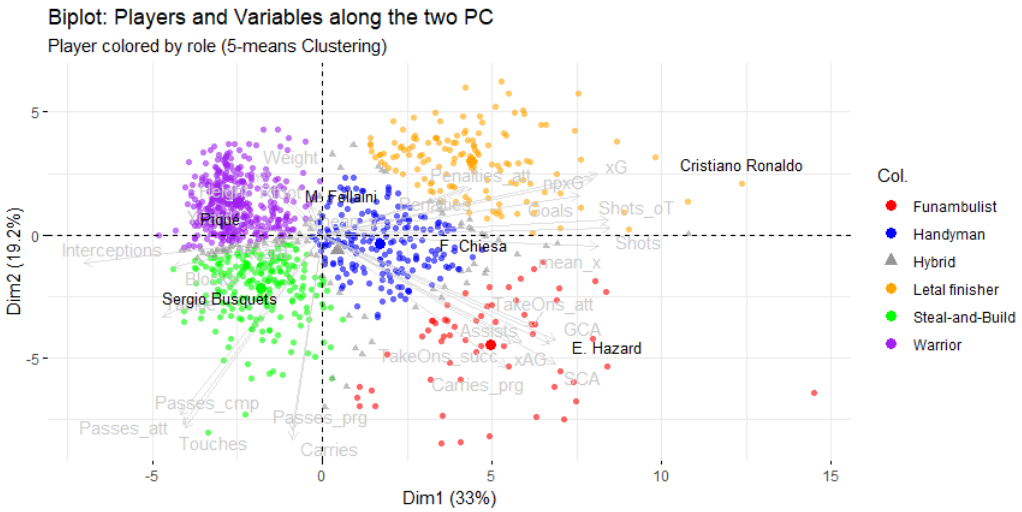


Figure 18: Players are clustered based on the actions they mostly perform.

The more attentive will have noticed that there is an extra cluster in grey. This is the Hybrid cluster composed of those players with Silhouette score smaller than a threshold taken equal to 0.1, as in [4].

The highlighted names in the previous picture are well-know examples of those clusters, or in simpler words players that can be easily associated with variables closest to them in the plot. Their characteristics will be shown in details in a future presentation and then made available upon request.

Hierarchical clustering

The last clustering method was the hierarchical clustering. Run on the standardized variables (as for the K-means) this algorithm produces a dendogram (Figure 19) that could be useful to visualize clusters in an alternative way.
It uses between-clusters distance functions (linkage) different from the euclidean used in the K-means. In our case the linkage that produced the best results was the Ward's linkage that minimizes the Error Sum of Squares, i.e. the distances between the observations and the centroids.

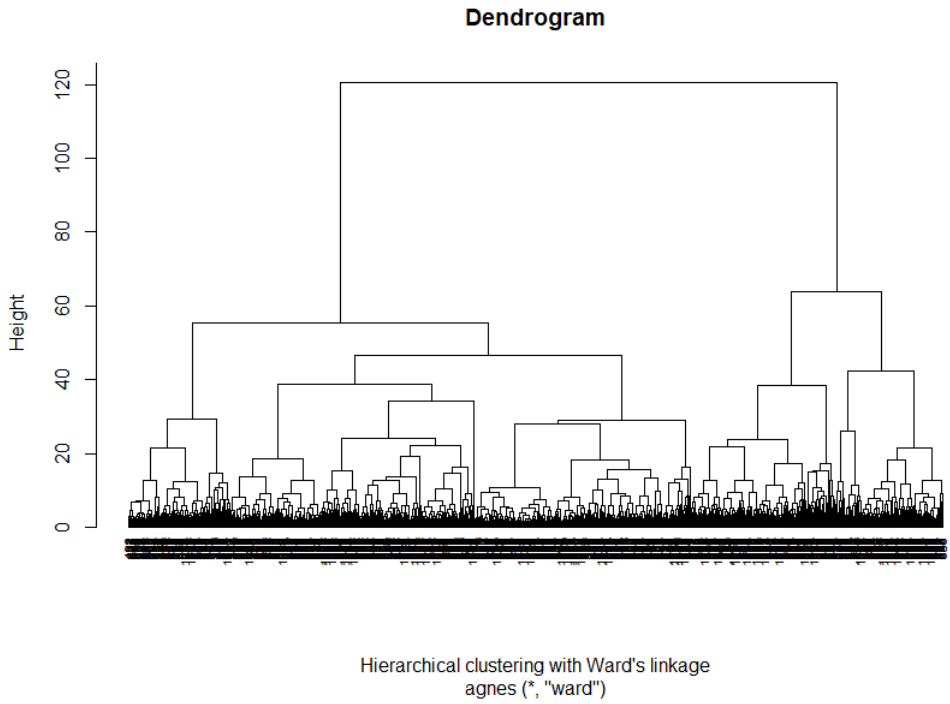


Figure 19: Hierarchical clustering complete dendogram

The best number of clusters was choosen using the Gap statistic, developed by Tibshirani and explained here [7].

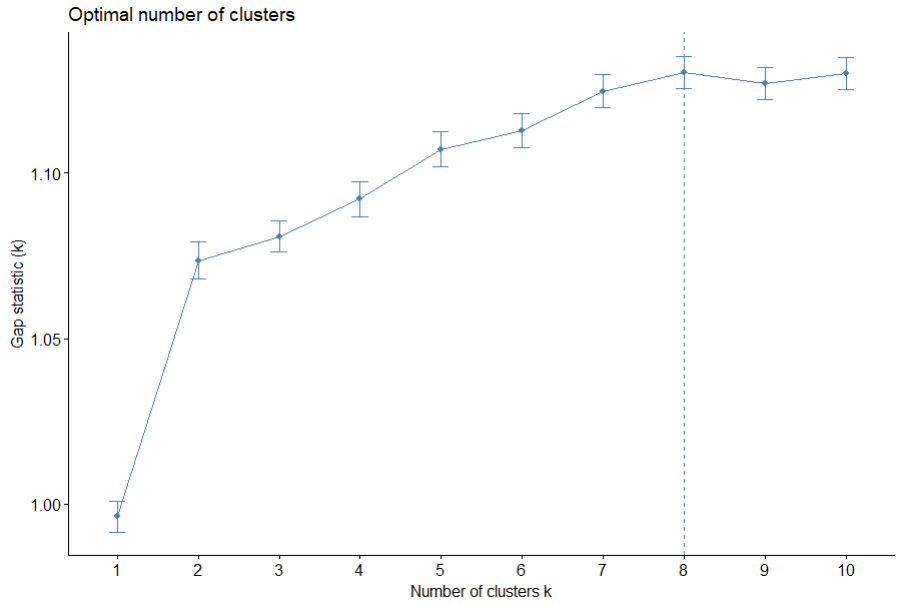


Figure 20: Gap statistic screeplot

Finally, a two dimensional plot is shown also for this method in order to compare it with the original clustering by Cintia and Pappalardo and with the 5-means on the 2 PC.

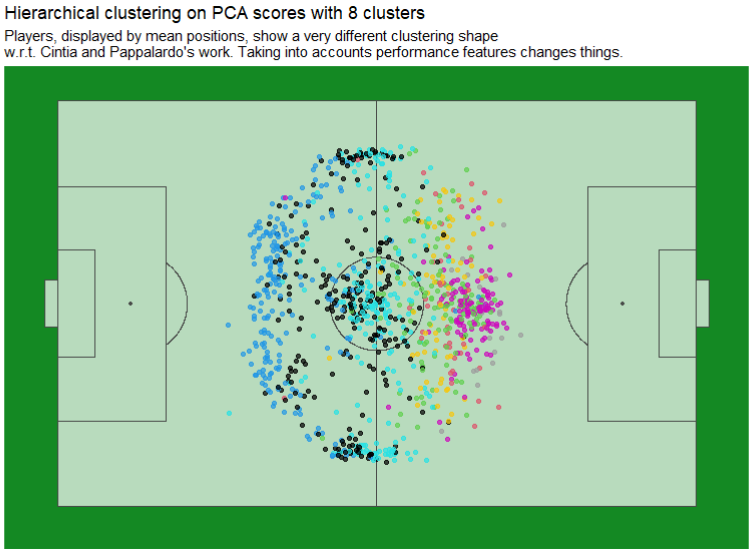


Figure 21: Bi-dimensional plot of the 8 clusters found by the Hierarchical clustering

Conclusion

The 5-means applied on the PCA projections results to be the best clustering (for Silhouette score) and thus it was used in order to go deeper in the details of each cluster, that means understanding the characteristics of the players that are clustered in each different ***role***. The roles characteristics and an example for each will be shown in the academic presentation of June 26^{th} 2023.

Bibliography

[1] Kaggle Community. *Kaggle - General Discussion*. Kaggle Discussion. Unknown. URL: https://www.kaggle.com/discussions/general/25913.

[2] César A. Morales. "A mathematics-based new penalty area in football: tackling diving". In: *Journal of Sports Sciences* (2016). DOI: 10.1080/02640414.2016.1177657.

[3] Luca Pappalardo et al. "A public data set of spatio-temporal match events in soccer competitions". In: *Scientific Data* 6 (Oct. 2019). DOI: 10.1038/s41597-019-0247-7.

[4] Luca Pappalardo et al. "PlayeRank". In: *ACM Transactions on Intelligent Systems and Technology* 10.5 (2019), pp. 1–27. DOI: 10.1145/3343172. URL: https://doi.org/10.1145\%2F3343172.

[5] C. Reep and B. Benjamin. "Skill and Chance in Association Football". In: *Journal of the Royal Statistical Society. Series A (General)* 131.4 (1968), pp. 581–585. ISSN: 00359238. URL: http://www.jstor.org/stable/2343726 (visited on 06/24/2023).

[6] Soccerment. *The Clustering Project*. Soccerment, 2022. URL: https://shop.soccerment.com/products/the-clustering-project-ebook-ita.

[7] Robert Tibshirani, Guenther Walther, and Trevor Hastie. "Estimating the number of clusters in a data set via the gap statistic". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 63.2 (2001), pp. 411–423. DOI: https://doi.org/10.1111/1467-9868.00293. eprint: https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/1467-9868.00293. URL: https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/1467-9868.00293.