QRT Data Challenge: Can you guess the winner? - A Football Game Predictor

JAROD LEVY*, Mines Paris - PSL and ENS Paris Saclay - jarod.levy@etu.minesparis.psl.eu
PAUL MEDDEB, Mines Paris - PSL and ENS Paris Saclay - paul.meddeb@etu.minesparis.psl.eu

Fig. 1. Football illustration generated using DALL-E

1 INTRODUCTION

1.1 Challenge Presentation

This scientific report presents a solution to the challenge proposed by QRT as part of the ENS data challenge. The objective of this challenge is to predict the outcome of a football game, which is a three-class classification problem: the home team wins, the away team wins, or the game ends in a draw. The challenge set by QRT follows the current trend in the sports world, which is increasingly relying on data for decision-making. The analysis of metrics is ubiquitous: in fan discussions, team strategies, television broadcasts, and sports betting. Leveraging data science could be promising in building increasingly high-performing teams and enhancing decision-making processes. This trend is fueled by the exponential availability of such data, with companies like *Sport Monks* providing the data used in this challenge.

1.2 Data Description

In this data challenge, the provided data represents real historical data at the level of teams and their players. This data is highly diverse and covers numerous leagues worldwide and various divisions. The model must be as general as possible to adapt to different levels and geographical locations. The data is distributed across 4 separate databases describing the home team, home players, away team, and away players. These databases are linked through a game identifier. Each player or team is associated with a list of attributes (25 for the team and 52 for the players), ranging from discipline to ball possession, player ratings, or number of season victories. These attributes are available at several levels: the last five games, the entire season, and are presented as cumulative sums, averages, and standard deviations. The goal of the challenge is to achieve the highest possible accuracy in predicting the outcome of the football game. The use of external data is disqualifying. Predicting that the home team will always win yields an accuracy of 46%, setting a preliminary benchmark for the challenge.

1.3 Literature Review

In recent years, football has become inseparable from data, marking a growing interest in predicting game outcomes. Such predictions could offer a real advantage to football teams in preparing against future opponents. On another hand, a highly effective model could be competitive in the sports betting market by setting increasingly challenging odds or achieving significant financial gains.

One of the major challenges in predicting football game outcomes is the unbalanced nature of available datasets, given that there are, on average, more games ending in a home team victory. A study by [8] revealed this behavior exists across many different sports. Given football's chaotic nature, statistical predictive models have been favored over the years to bring a scientific methodology to this task. The behavior and performance of players seem unpredictable. With the increasing availability of data, machine learning appears to be a promising set of techniques for solving

*Both authors contributed equally to this research.

this task with sufficient accuracy. Numerous papers and articles have been specifically published on game outcome prediction [3, 5, 8, 14, 25, 28], while others focus on different tasks: injury predictions [15], enhancing player and team performance [16], and even predicting players with the highest potential [4]. On a broader scale, some reviews have summarized the use of predictive techniques in team sports (Basketball, Baseball, American Football, Cricket, and Football) [12].

A very common problem in these projects is predicting draws. [10] and [13] both noted that their models, although very different (long-short term memory and logistic regression), predicted no draws. It is easy to see how a series of features could indicate one team's superiority over another, but predicting a draw seems subtle. This challenge is further underscored by the typically lower number of draws in datasets. Many studies have therefore decided to simplify the problem to "the home team wins or not" [18].

Machine learning algorithms are divided between supervised learning (classification and regression) based on data and labels and unsupervised learning (clustering) with only input data. Football game data is often tabular, leading to the use of classical machine learning models such as decision trees [5, 16], random forests [19, 21], support vector machines [7, 20] and logistic regression [18]. The performance of these models heavily depends on the quality of the datasets and the features included in these data. Logistic regression often returns as the method yielding the most promising results. The growing availability of data has also allowed for the consideration of deep [23, 29] and recurrent models [1].

Another fundamental aspect not yet discussed is the importance of feature engineering. Databases are often filled with irrelevant or poorly exploited data, and it must be modified so its football significance is highlighted. [3] described that only 2 out of 12 used features were not created by themselves. [21] and [2] use features by aggregating statistical indicators for home and away teams in a manner very similar to ours. FIFA ratings from the EA Sports video game, not provided in our project, often represent the skills and attributes of players and teams [3, 21, 24, 26]. The company hires specialists around the globe to have the most realistic ratings possible each year [9]. [18] uses these FIFA ratings and achieves an impressive accuracy of 69.5%. Some studies use increasingly broad attributes: tactical, technical, psychological, and contextual variables to predict future performance of teams and players [6, 17, 22]. Other interesting features used include the month to model the impact of weather on a game [11] or the number of days since the last game [27] to illustrate team fatigue. Finally, some projects use the ability to beat bookmakers as a metric. Thus, [21] evaluates their model based on the amount of money won by betting on predicted games. Their most successful model for this study was the random forest, with a profit margin of 26.74%.

2 METHOD

The training dataset contains 12 042 games, each described by 280 team-level features and 606 player-level features. With an average of 15,4 players per team, a game is described by 4680 player-level features in average. Therefore, one of the most important challenges in achieving good prediction is to separate the signal from the noise in the data. This is done by cleaning the data, selecting and creating features as relevant as possible, and reducing the dimension of the problem.

2.1 NaN values

When dealing with tabular datasets, cleaning the data mainly consists in handling NaN (Not a Number) values. After assessing the extent to which NaNs affect data quality, it is essential to define a proper NaN handling strategy: this can range from removing samples or features with at least one missing value, to replacing NaNs with a fixed value, which may be the mean value, or with a more sophisticated imputed value. We perform an analysis to assess the impact of NaN values within the team's dataset on overall model performance, with the aim of designing an appropriate NaN processing strategy applicable to the whole dataset.

Figure 2 shows NaNs accross the team dataset table. The first important takeaway is that not all the samples and features are responsible for the NaN values. Actually most of the columns and most of the samples don't have any. A few columns related to injuries have a rate of about 30% missing values. Moreover these columns, if we keep them, appear not to be very useful for predicting the outcome. Hence, we've decided to remove them. The remaining columns containing NaNs were still informative and we didn't want to remove them. For example features such as the rate of succesful passes or shots have a lot of NaN values. Rather than removing the entire column we instead remove the samples with too many NaNs and replace the remaining NaNs with the mean value of the feature. Table 1 gives a glimpse of how impactful these kind of decisions can be on the overall performance of the model.

Fig. 2. Heatmap of the team dataset, where NaN values are highlighted in yellow

Table 1. Impact of NaN handling strategy on performance

Remaining rows	Remaining columns	Final model Accuracy
12042	280	0.4819
10089	252	0.4995
9321	252	0.4965

Hence, the thresholds for the number of NaN values at which we decide to remove either a column or a row were hyperparameters that have been optimized.

As for the player dataset, it contains a lot more NaN values than the team dataset. In particular some columns are entirely made up of NaNs, so we removed them first. For the rest of the player dataset, as it was more complex, we addressed NaNs as follows:

- Position Column: use other appearances of the player's name to determine their position. Use columns with non-null values for goal-related features to find the position. Remove remaining players with still unknown positions.
- Other Columns: All rows containing more than 50% NaN values were removed.

2.2 Extracting Features

After cleaning the dataset, the main task of the data challenge remains: relevant information still has to be extracted from the data. Among all the features describing every sample, a lot of them is noise. The challenge is to remove this noise while intelligently combining relevant features to create a signal as informative as possible. We conduct two separate analyses on the team dataset and on the player dataset.

2.2.1 *Feature Engineering.*

Team Dataset. The most general features about the games are available in the team dataset of the challenge. These features comprise 25 statistics, which are aggregated by sum, average and standard deviation. They are summaries of the last 5 games prior to the game, as well as season-to-date statistics of the game being predicted. As the sum and average columns were redundant we decided to keep only the average features, as they contained less NaN values. Moreover we also decided not to consider the standard deviations on the last 5 games, as it might not always be a meaningful value. To take advantage of the information about the last 5 games available to us, we decided to compute for every feature an indicator of the current performance of the team with respect to the feature:

$$\textbf{Current Performance} = (\text{average of the last 5 games} - \text{season average}) \times \text{season standard deviation} \qquad (1)$$

This indicator should be high when the team is going through a good period in terms of the feature considered, and low when the team is in poor form.

In summary, for every feature we only consider the season average and the current performance indicator.

To further reduce the number of features we create categories of features, which are meant to be meaningful:

- Possession: ball possession, ball safe, passes
- Attack: Attacks, corners, dangerous attacks, goals, penalties, shots total
- Dicipline: fouls, offsides, red cards, yellow cards
- Efficiency: (dangerous attacks / attacks) + (shots insidebox + shots on target - shots off target) / shots total

Player Dataset. As a second part, the challenge provides datasets at the level of the players. This time, we are supplied with all players from all games listed in the first dataset, described by 52 metrics, presented in the same manner as the first dataset, with the addition of their position and player name. For the same reasons as with the first dataset, we will only use the season's average and current performance statistics.

The initial step was also to sort the features into meaningful groups, distinguishing 8 categories:

- Midfield features: passes, crosses
- Attacker features: duels, assists, goals, shots, dribbles, created chances
- Negative attacker features: missed chances, offsides, missed penalties, lost duels, ball losses
- Goalkeeper features: number of saves, number of penalties saved
- Defender features: number of successful tackles, number of blocked shots, number of interceptions
- Negative defense features: number of errors leading to a goal, dribbles successful against the player, own goals, goals conceded per game
- Dicipline features: yellow cards, red cards, fouls, penalties conceded
- Importance features: player rating, number of minutes played, number of starts

Additional features were created, notably ratios: ratio of successful crosses, shots on target, goals per number of shots, won duels, lost duels. Then, as many columns as groups were added. The values correspond to the average of the features of each group.

Two features: *Top Player* and *Bad Player* indicate whether a player has a Player Rating feature above 80 or below 50, respectively. This, after team aggregation, will allow proportions of good and bad players per team, which could represent valuable information for predicting the game outcome.

2.2.2 *Aggregating Team and Player Features.*

. To combine team features with player features, all players of a team must be aggregated into a single line representing the game. While simply averaging all columns is an option, our experiments showed that a more nuanced approach could yield better results. From a football perspective, evaluating a team's attack or defense does not involve averaging all players. It primarily considers the team's key players who guide it towards success. We adopted this approach. For a given team, players are grouped as follows: the features corresponding to defense, attack, midfield, or goalkeeper will represent the average of the 5 most representative players (or fewer for goalkeepers) in that position on the team. To find the most representative players, we simply ranked these players by the number of starts in the season (*Starting Lineup* feature).Other features (discipline, importance, etc.) were simply averaged. This aggregation approach is, in our opinion, more realistic and leads in an accuracy improvement of 0.5%[1]. With this process, we obtain a hundred features representing the players by team per game.

2.2.3 *Feature Selection.*

. Once all the features have been created, and the player and team features can be considered jointly, the importance of each one of them needs to be assessed. Reducing the number of features, only keeping the most informative ones, is essential to ensure generalisation when predicting the outcome. This is done in 3 steps.

Evaluating Feature Importance. Several criteria exist to assess the importance of a feature for predicting the outcome. Most of them consist in evaluating in a way or another the correlation between the feature and the label. We consider two of these criteria: chi-2 statistics and coefficients of a trained logistic regression. The chi-2 statistic quantifies the extent of the discrepancy between the observed frequencies in a contingency table and the frequencies

[1]All accuracy comparisons are conducted using 5-fold cross-validation, with all other parameters remaining constant.

that would be expected if the variables were independent. The logistic regression coefficient quantifies how much a one-unit change in the predictor variable affects the log-odds of the outcome variable, indicating its importance in predicting the outcome. We then sort the features by order the average of these two importance criteria and keep the top 120 features (among 320 columns). Figure 3 shows the sorted table and two boxplots of one bad and one good feature. The box plots allow us to confirm that the importance measure is relevant, since we can see that a high importance score is linked to having distinct distributions for each label.

	Feature	Importance_chi2	Importance_logreg	Mean_Importance
10	AWAY_PASSES_season_average	0.823759	0.921482	0.872621
11	AWAY_SHOTS_TOTAL_season_average	0.791851	0.680359	0.736105
2	HOME_GAME_WON_season_average	0.909478	0.448818	0.679148
24	AWAY_GAME_LOST_season_average	0.578280	0.776101	0.677191
4	AWAY_SHOTS_INSIDEBOX_season_average	0.894145	0.380200	0.637172
...	...	...	...	...
198	AWAY_PLAYER_ASSISTS_variance	0.000831	0.008050	0.004440
136	HOME_SHOTS_INSIDEBOX_curr_perf	0.003590	0.001787	0.002688
245	HOME_PLAYER_DUELS_LOST	0.000333	0.004232	0.002283
275	AWAY_PLAYER_SAVES_INSIDE_BOX_variance	0.000117	0.002738	0.001428
310	AWAY_PLAYER_DUELS_WON_variance	0.000012	0.000000	0.000006

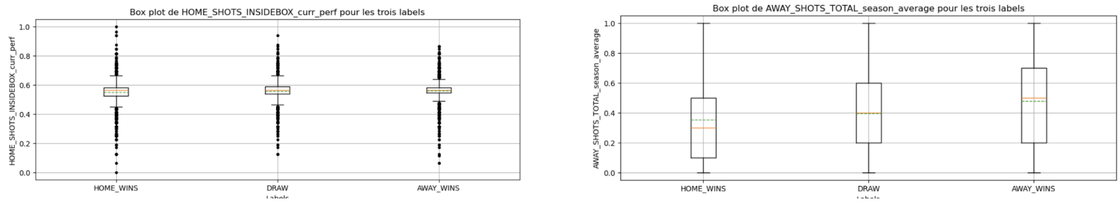

Fig. 3. Features sorted by the average of 2 measures of different importance. Bottom left box plot of a bad feature and right of a good feature

Group together correlated features. The pre-selection we've just made of the features has eliminated the bad features, i.e. the noise in the data. We now know that there is signal in each of the features we have left. However, some of these features are redundant. For example, the number of wins and losses over the season are highly related. It's important to eliminate these redundancies as much as possible, in order to reduce the size of the problem and obtain the most robust signal possible. To do this, we compute the correlation between each feature and use a hierarchical clustering algorithm to group highly correlated features into clusters. Note that features linked to defeat have been taken as negative, so there should be no features negatively correlated with each other. The number of clusters is a hyper-parameter set according to the dendogram given by the hierarchical clustering, which we then optimize. Finally, we take the average of the features within each cluster, enabling us to reduce the number of features and eliminate redundancies a priori. Figure **??** shows the correlation matrix of the pre-selected features and the clusters obtained after the hierarchical clustering algorithm. The formed clusters make sense from a soccer point of view.

PCA - Dimension Reduction. Finally, we reduce the dimension and make our features independent by performing a PCA and keeping only the first components. This eliminates the remaining noise, making the features independent. The number of components we keep is again a hyper-parameter that we choose according to the figure 6 and optimize later.

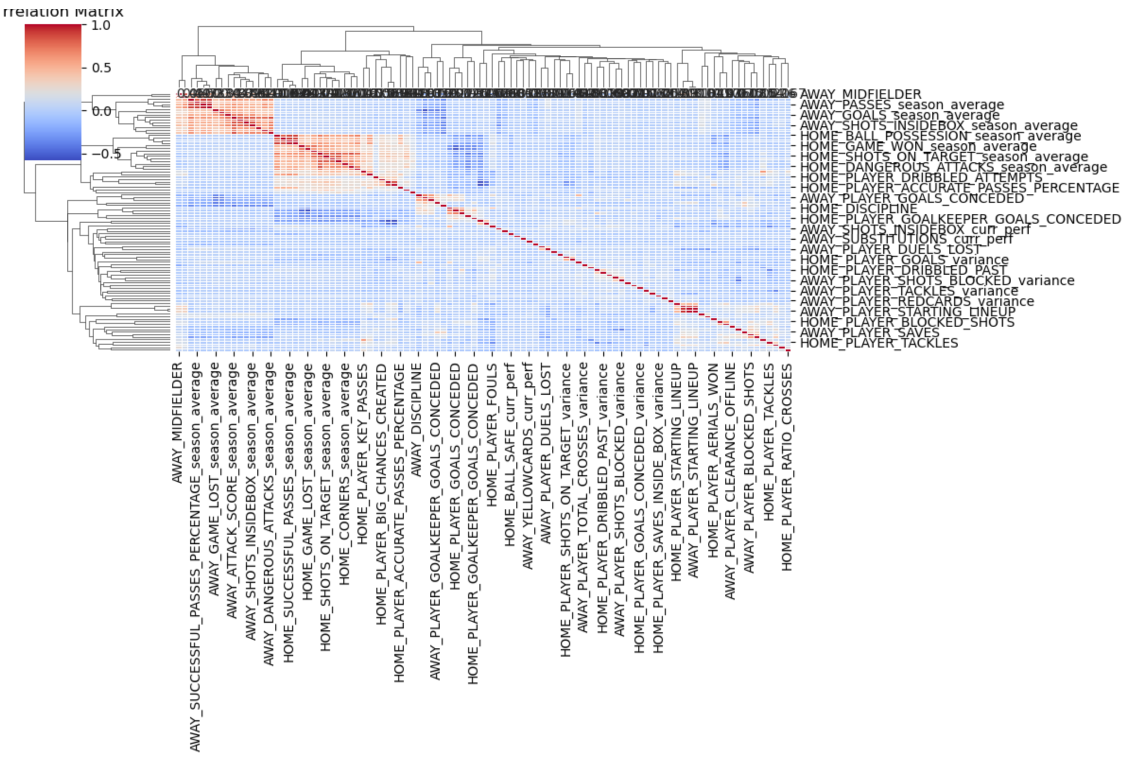

Fig. 4. Heatmap of the correlation matrix of selected features

```
Cluster 6: HOME_PLAYER_SHOTS_ON_TARGET, HOME_PLAYER_SHOTS_TOTAL
Cluster 9: HOME_BALL_POSSESSION_season_average, HOME_SUCCESSFUL_PASSES_season_average, HOME_PASSES_season_average, HOME_SUCCESSFUL_PASSES_PERCENTAGE_season_average
Cluster 10: HOME_GAME_WON_season_average, HOME_GAME_LOST_season_average
Cluster 11: HOME_CORNERS_season_average, HOME_DANGEROUS_ATTACKS_season_average, HOME_ATTACKS_season_average
Cluster 12: HOME_SHOTS_INSIDEBOX_season_average, HOME_ATTACK_SCORE_season_average, HOME_SHOTS_ON_TARGET_season_average, HOME_SHOTS_TOTAL_season_average
Cluster 14: AWAY_MIDFIELDER, AWAY_PLAYER_ACCURATE_PASSES
Cluster 15: AWAY_PASSES_season_average, AWAY_BALL_POSSESSION_season_average, AWAY_SUCCESSFUL_PASSES_season_average, AWAY_SUCCESSFUL_PASSES_PERCENTAGE_season_average
Cluster 16: AWAY_GAME_LOST_season_average, AWAY_GAME_WON_season_average
Cluster 17: AWAY_ATTACKS_season_average, AWAY_DANGEROUS_ATTACKS_season_average
Cluster 19: AWAY_SHOTS_TOTAL_season_average, AWAY_SHOTS_INSIDEBOX_season_average, AWAY_ATTACK_SCORE_season_average, AWAY_SHOTS_ON_TARGET_season_average
Cluster 21: HOME_PLAYER_SHOTS_ON_TARGET_variance, HOME_PLAYER_GOALS_variance
Cluster 52: AWAY_PLAYER_MINUTES_PLAYED, AWAY_PLAYER_STARTING_LINEUP, AWAY_IMPORTANCE
Cluster 54: HOME_PLAYER_BLOCKED_SHOTS, HOME_PLAYER_CLEARANCES
Cluster 63: HOME_PLAYER_GOALS_CONCEDED, HOME_DEFENSER_NEGATIVE
Cluster 67: AWAY_PLAYER_GOALS_CONCEDED, AWAY_DEFENSER_NEGATIVE
```

Fig. 5. Some of the clusters formed by the grouping of features

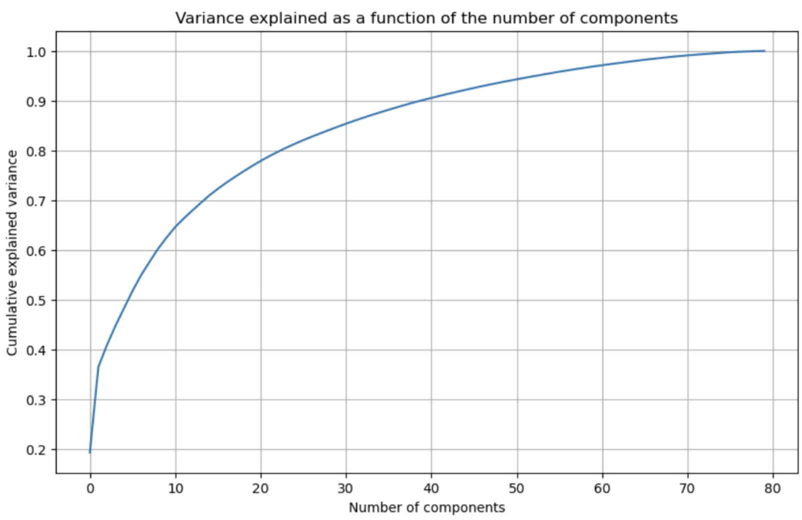

Fig. 6. Variance explained as a function of the number of PCA components

2.3 Prediction

2.3.1 Model selection.

. Now that we've extracted and selected our features, we need to find the best model for our prediction task. We tried out a number of models from the scikit learn library, and the results are summarized in the table 2. We find that the best model is logistic regression, which confirms the literature. In addition, we train a voting model that predicts the majority label predicted by the models in the table. This voting model has the advantage of being more robust and performs similarly to logistic regression.

Model	Accuracy
Logistic Regression	**0.4981**
Random Forests	0.4885
XGBoost	0.4965
SVM	0.4906
Naive Bayes	0.4922
Adaboost	0.4670
Gradient Boosting	0.4869
Voting Model	0.4965

Table 2. Accuracy of different models

2.3.2 The Draw Problem.

Adding a Prior. A closer look at our model's predictions reveals that the confusion matrix is very unbalanced. The model mainly predicts home wins and has particular difficulty in correctly predicting draws. In fact, it hardly predicts any at all. In reviewing the literature, it has been repeatedly reported that the type of model we were attempting to develop struggled significantly in predicting draws. Our findings confirmed this trend. Initially, our model only predicted a draw 1.3% of the time. However, examining football game statistics reveals that about 46% of the time, the home team wins (the project's benchmark), with the remaining time equally split between away team victories and draws, at 27% each. To attempt to correct our results after the model's initial predictions, we implemented a "prior knowledge" that forces the model to adhere to these three percentages. This is achieved by taking the highest probabilities to reach the 46% for home wins and then distributing the remaining predictions between "away team wins" and "draws" by comparing their two probabilities. This adjustment resulted in an approximate 0.5% increase in overall accuracy. The results are shown in table 3.

Yet again, we find ourselves slightly disappointed with this outcome. Given the strength of the prior, we had anticipated a significant impact on the final result. The fact that it did not markedly change the outcome leads us to believe that many "Home Wins" with a low probability (which remains the highest of the three) turn out to be true positives. This once again implies the weakness of our model and its features.

	HOME WINS	AWAY WINS	DRAW
WITHOUT PRIOR	17076	7958	334
WITH PRIOR	11671	6849	6848

Table 3. Games outcomes with and without prior

ML Methods for Imbalanced Classes. We have also tried to implement more basic training adaptation techniques for datasets with unbalanced classes. In particular, the scikit learn library proposes to overweight the cost function for rare labels during logistic regression training, which we have tried to do. However, even if this effectively forces the model to predict more null games, as shown on the confusion matrices in figure 7, the nulls are still not predicted in the right place, and the model's accuracy is no better.

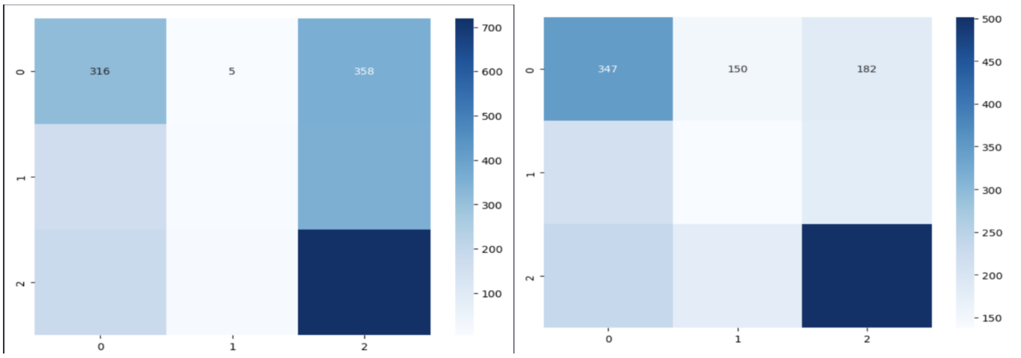

Fig. 7. On the left, confusion matrix of a simple logistic regression. On the right, confusion matrix for a logistic regression specially trained to correct class imbalance. We can see that more draws are predicted, but the global confusion matrix is no better.

An Idea Often Mentioned but Ineffective in this Context. In an attempt to incorporate from the outset the information that predicting draws was challenging, we developed a strategy that slightly altered the initial problem. We transformed the issue into a binary classification problem: "Will the home team win or not?". Our approach was to monitor the model's predictions and convert all low probabilities into draws. We established a range within which we would make this transformation: when the maximum probability was below 55%. Even after tuning this parameter, the final accuracy remained very low. Yet, this type of approach seems to have been successful in many projects.

3 RESULTS

Our team achieved a private score of 0.4838 in the data challenge, which is a slight difference of 0.0067 from the first place score. Hence, our public ranking is 66th, our Public Academic ranking is 13th and our Private Academic ranking is 18th. The difference between our private and public score shows that our model suffered from slight overfitting, likely because the public test data is more similar to our training data than the private test data. It is unlikely that we have significantly overfitted the public testing data, given the relatively small number of submissions and our emphasis on football considerations rather than maximizing accuracy.

Score	Accuracy
Public Score	0.4888
Private Score	0.4838

Table 4. Public and Private Scores

Starting from Scratch, What Strategy Would We Have Employed? Interesting insights have been gained from other studies. Upon reviewing various papers, it appears that the most promising models utilize features from the FIFA video game by EA Sports. This video game includes a continuously updated database that accurately characterizes teams and players. Experts are deployed worldwide to report changes in the performance of players and teams. It would have been promising to use these data. Additionally, some projects directly incorporate bettors' odds into the features. These odds are themselves based on private statistics used by bookmakers to maximize their profits and are thus potentially very precise. Interesting features to add were also mentioned, such as the month of the game to account for weather impact or the number of games in the last week to assess the team and players' fatigue level. Finally, we would have liked to have access to raw data without normalization. The normalized data were quite hard to interpret.

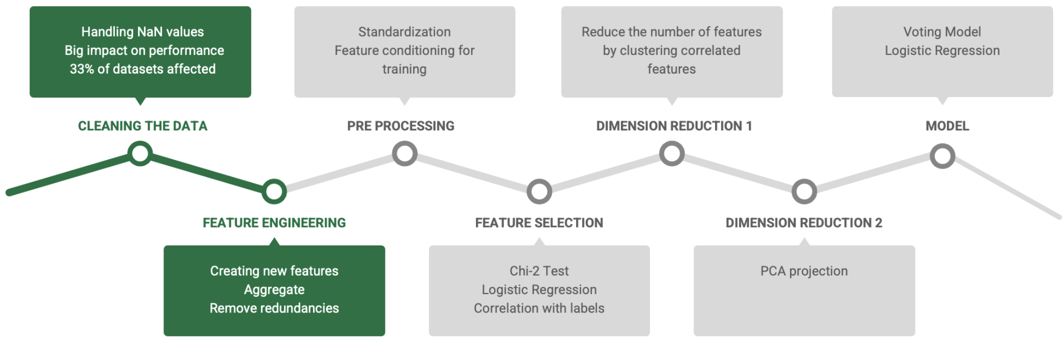

Fig. 8. Summary Diagram of our Pipeline

4 CONCLUSION

It was highly engaging to keep only the valuable features and to create new ones that made sense in the context of football. This project enabled us to follow a rigorous approach applicable to many other challenges: reviewing the state of the art, data exploration, data cleaning, feature engineering, establishing an evaluation procedure, model creation, and making continuous adjustments until achieving satisfactory performance levels. Our approach is summarized in figure 8.

Overall, it was quite disappointing to see that almost none of our ideas led to a significant improvement in performance. Nonetheless, we insisted on retaining ideas that did not necessarily enhance performance but made sense in our understanding of the model, leading to a logical and scientific pipeline. Looking at the general leaderboard, one can see the difficulty of this problem.

Football exhibits chaotic behavior, which contributes to its beauty, and perhaps if we had achieved a 70% score, we might not have written this report and kept the results to ourselves...

REFERENCES

[1] Jassim AlMulla, Mohammad Tariqul Islam, Hamada R. H. Al-Absi, and Tanvir Alam. 2023. SoccerNet: A Gated Recurrent Unit-based model to predict soccer match winners. *PLOS ONE* 18, 8 (2023), e0288933.

[2] S. Anfilets, S. Bezobrazov, V. Golovko, A. Sachenko, M. Komar, R. Dolny, V. Kasyanik, P. Bykovyy, E. Mikhno, and O. Osolinskyi. 2020. Deep multilayer neural network for predicting the winner of football matches. *International Journal of Computing* 19, 1 (2020), 70–77.

[3] R. Baboota and H. Kaur. 2019. Predictive analysis and modelling football results using machine learning approach for English Premier League. *International Journal of Forecasting* 35, 2 (2019), 741–755.

[4] D. Barron, G. Ball, M. Robins, and C. Sunderland. 2018. Artificial neural networks and player recruitment in professional soccer. *PLoS ONE* 13, e0205818 (2018).

[5] G. Bilek and E. Ulas. 2019. Predicting match outcome according to the quality of opponent in the English Premier League using situational variables and team performance indicators. *Journal of Performance Analysis in Sport* 19 (2019), 930–941.

[6] J. Brooks, M. Kerr, and J. Guttag. 2016. Using machine learning to draw inferences from pass location data in soccer. *Statistical Analysis and Data Mining: The ASA Data Science Journal* 9 (2016), 338–349.

[7] S. Chawla, J. Estephan, J. Gudmundsson, and M. Horton. 2017. Classification of Passes in Football Matches Using Spatiotemporal Data. *ACM Transactions on Spatial Algorithms and Systems* 3 (2017), 1–30.

[8] João Gustavo Claudino, Daniel de Oliveira Capanema, Thales Vianna de Souza, Julio Cerca Serrão, Ana Carolina Machado Pereira, and George P. Nassis. 2019. Current Approaches to the Use of Artificial Intelligence for Injury Risk Assessment and Performance Prediction in Team Sports: a Systematic Review. *Sports Medicine - Open* 5, 1 (2019), 41.

[9] L. Cotta, P. de Melo, F. Benevenuto, and A. Loureiro. 2016. Using FIFA soccer video game data for soccer analytics. In *Workshop on Large Scale Sports Analytics*.

[10] N. Danisik, P. Lacko, and M. Farkas. 2018. Football match prediction using players attributes. In *2018 World Symposium on Digital Intelligence for Systems and Machines (DISA)*. IEEE, 201–206.

[11] H. Elmiligi and S. Saad. 2022. Predicting the outcome of soccer matches using machine learning and statistical analysis. In *2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC)*. IEEE, 1–8.

[12] T. Horvat and J. Job. 2020. The use of machine learning in sport outcome prediction: A review. *WIREs Data Mining and Knowledge Discovery* 10 (2020).

[13] C. P. Igiri and E. O. Nwachukwu. 2014. An improved prediction system for football a match result. *IOSR Journal of Engineering* (2014).

[14] Matheus Kempa. Accessed: Year. Machine Learning Algorithms for Football Predictions. https://towardsdatascience.com/machine-learning-algorithms-for-football-prediction-using-statistics-from-brazilian-championship-51b7d4ea0bc8.

[15] Jon L. Oliver, Francisco Ayala, Mark B.A. De Ste Croix, Rhodri S. Lloyd, Gregory D. Myer, and Paul J. Read. 2020. Using machine learning to improve our understanding of injury risk and prediction in elite male youth football players. *Journal of Science and Medicine in Sport* 23 (2020), 1044–1048.

[16] L. Pappalardo and P. Cintia. 2018. Quantifying the relation between performance and success in soccer. *Advances in Complex Systems* 21, 1750014 (2018).

[17] E. Perri, S. Carlo, A. Rossi, A. Trecroci, A. Alberti, and M. Laia. 2020. Relationship Between Wellness Index and Internal Training Load in Soccer: Application of a Machine Learning Model. *International Journal of Sports Science Coaching* (2020).

[18] D. Prasetio et al. 2016. Predicting football match results with logistic regression. In *2016 International Conference On Advanced Informatics: Concepts, Theory And Application (ICAICTA)*. IEEE, 1–5.

[19] M. A. Raju, M. S. Mia, M. A. Sayed, and M. R. Uddin. 2020. Predicting the outcome of English Premier League matches using machine learning. In *2020 2nd International Conference on Sustainable Technologies for Industry 4.0 (STI)*. IEEE, 1–6.

[20] G. Rodas, L. Osaba, D. Arteta, D. Fernández, and A. Lucía. 2019. Genomic Prediction of Tendinopathy Risk in Elite Team Sports. *International Journal of Sports Physiology and Performance* 14 (2019), 1–7.

[21] F. Rodrigues and A. Pinto. 2022. Prediction of football match results with machine learning. *Procedia Computer Science* 204 (2022), 463–470.

[22] A. Rossi, L. Pappalardo, P. Cintia, F. M. Iaia, J. Fernàndez, and D. Medina. 2018. Effective injury forecasting in soccer with GPS training data and machine learning. *PLoS ONE* 13, e0201264 (2018).

[23] D. Rudrapal, S. Boro, J. Srivastava, and S. Singh. 2020. A deep learning approach to predict football match result. In *Computational Intelligence in Data Mining: Proceedings of the International Conference on ICCIDM 2018*. Springer, 93–99.

[24] S. Saiedy, M. Qachmas, and F. Amanullah. 2020. Predicting epl football matches results using machine learning algorithms. *International Journal of Engineering Applied Sciences and Technology* 5 (2020), 83–91.

[25] Rishi Sanke. Accessed: Year. Using Machine Learning to Predict Football Game Outcomes. https://python.plainenglish.io/creating-machine-learning-models-to-predict-football-game-outcomes-70b6bf02885c.

[26] J. Stübinger, B. Mangold, and J. Knoll. 2019. Machine learning in football betting: Prediction of match results based on player characteristics. *Applied Sciences* 10, 1 (2019), 46.

[27] N. Tax and Y. Joustra. 2015. Predicting the Dutch football competition using public data: A machine learning approach. *IEEE Transactions on Knowledge and Data Engineering* 10, 10 (2015), 1–13.

[28] A. Titu. Accessed: Year. Predicting football matches with machine learning. https://github.com/aziztitu/football-match-predictor.

[29] E. Tiwari, P. Sardar, and S. Jain. 2020. Football Match Result Prediction Using Neural Networks and Deep Learning. In *2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)*.