Football Forecast

5001 Project Report

Harsh Sunil LAKHANI   
20910249  
harshslakhani@connect.ust.hk

Ye HUANG  
20936499  
yhuanggc@conncet.ust.hk Ho Pun CHENG  
20873245  
carson.cheng@connect.ust.hk

Yueyin TAN  
20922620  
ytanax@connect.ust.hk Yau Wai YIM  
20876194  
ywyimaa@connect.ust.hk

# Abstract

The quadrennial World Cup has returned and rekindled the passion towards football. This is a question that always lingers in our mind: Can we predict the outcome of a match? If so, what are the factors? In this project, we restructured the La Liga dataset by combining the original dataset with the data scraped from several different sources using Power Automate to analyze the effect of external factors, including location, temperature and precipitation. We also used Power Automate for imputing the missing data and dropping redundant data. In the exploratory data analysis section, we reveal the correlation between variables through the heatmap and find some intuitionistic or non-intuitionistic correlations. Considering the robustness and computational efficiency of ensembled models, we selected AdaBoost, GradientBoost and Random Forest as our candidate models. Since our dataset is imbalanced, we selected the F1-score as our metric to evaluate the models’ performance. After comparing the experiment results of the different models, we selected GradientBoost as our finalized model and produced a demo program for users to make a prediction on an upcoming match.

# Introduction

The aim of this project was to produce a highly accurate prediction of the outcome of an upcoming match. We aimed to achieve this by training a model on a robust dataset combined with external factors that we assumed will help with generating a more accurate prediction. Along with a highly informative and complete dataset, we believed that our intuition and understanding of the sport helped us make substantial analysis and in turn, helped create a highly accurate model. We conducted several statistical tests and made use of analysis tools to understand the features and the correlations between one another. Predicting the outcome of an upcoming match can help coaches analyze and prepare their team according to the prediction. Moreover, people who are avid gamblers can take advantage of the model to make a more informed decision on their next bet.

# Design and implementation

## Data collection

### La Liga dataset

The Spanish La Liga dataset[[1]](#footnote-2) acquired from Kaggle, was selected for our study. The dataset is one of the most informative and complete dataset available online. It covers a total of 380 matches held every league year[[2]](#footnote-3) from 2014 to 2020, i.e. 2,660 matches. Each match has 40 columns:

* League year
* Names of home team and away team
* Half-time and final scores
* Match excitement (0 to 10)
* 17 match statistics for the home team and away team:
  + Rating
  + Possession %
  + Off Target Shots
  + On Target Shots
  + Total Shots
  + Aerials Won
  + Blocked Shots
  + Clearances
  + Corners
  + Fouls
  + Pass Success %
  + Goals Conceded
  + Goals Scored
  + Throw Ins
  + Yellow Cards
  + Second Yellow Cards
  + Red Cards

The dataset requires a small amount of preprocessing, which is described in the following section.

### External data

Suggested by the professor, we decided to add external data to our dataset. The weather data was selected as it was assumed to have the most effect on the matches.

We found a site called Weather Underground[[3]](#footnote-4) containing the historic daily weather data – temperature and precipitation (i.e. rain) – for most of the cities in Europe (Figure 1).

A picture containing calendar

Description automatically generated

Figure 1: Historic daily weather data sample from Weather Underground

To obtain the weather data for each match, we needed the date and the location (i.e. city) of each match.

#### Date

The La Liga dataset has only consisted of the league year for each match but not the exact date. Therefore, we needed to find the exact date for each match online and combine it with our dataset. We found a website called Sky Sports[[4]](#footnote-5) that shows the fixtures of La Liga League (Figure 2).

Graphical user interface, application

Description automatically generated

Figure 2: Dates of matches from Sky Sports

Since the website has a protection mechanism against web scraping, the data of the 7 years were manually copied to Microsoft Excel (Excel) for processing.

#### Location

As the matches are held in the city of the Home Team of every game, the match locations were deduced with a heuristic. Exceptions such as the promotion or relegation of teams from La Liga, increased the number of match locations compared to the usual 20 teams in a league year. After inspection, there were only 26 unique locations in total.

To facilitate the web scraping process, the website path to the cities was looked up and added to the match location table. (Table 1)

Table 1: Data table of match locations (Extract)

|  |  |  |
| --- | --- | --- |
| Team | City | Website path |
| MÁLAGA | Málaga, Spain | málaga/LEMG |
| SEVILLA FC | Seville, Spain | seville/LEZL |
| GRANADA | Granada, Spain | málaga/LEMG |
| ALMERÍA | Almería, Spain | almería/LEAM |
| EIBAR | Municipality of Eibar, Spain | eibar/LEBB |
| BARCELONA | Barcelona, Spain | barcelona/LEBL |

#### Weather data

Microsoft’s Power Automate was then deployed to web scrape the weather data from Weather Underground. About 70,000 rows of data were downloaded for the 26 cities and 7 league years.

Power Automate was preferred to the popular web scraping tool Selenium because of the former’s low-code environment and compatibility with Excel, which promises a swift deployment. As the scope of work was finite, speed was not the biggest concern.

Graphical user interface, application

Description automatically generated

Figure 3: Web scraping with Power Query

With the match locations and match dates assigned to each match, the daily weather data can be looked up accordingly with Power Query in Excel.

Figure 4 shows the high-level data flow. After the data cleansing, transforming and merging, these two columns were added to the original dataset:

1. Temperature (in Fahrenheit)
2. Precipitation

Finally, a CSV (comma-separated values) file was generated in Excel and fed into the model for analysis.

Diagram

Description automatically generated

Figure 4: Data flow of weather data

The following section depicts the data preprocessing performed.

## Data preprocessing

The data preprocessing consists of two parts: La Liga dataset and scraped data.

### La Liga Dataset

The original dataset from the source was clean. It did not consist of any missing values.

#### Column Manipulation

The ‘Score’ column intuitively contained the scores of each match. The column was originally of the object datatype, which was problematic for us to conduct any exploratory data analysis and to use it as a feature in our model. It is an important feature; therefore, we converted it into the “goal\_diff” column; which as the name suggests is the goal difference of the two teams. Particularly, it holds the value of the difference between the Home Team Score and the Away Team Score (Home Team Score – Away Team Score”).

As mentioned earlier, our objective is to predict the outcome of the game, whether it may be a Win, Loss or a Draw. We did not have a particular column that depicted this and therefore we created the “Winning Team” column. As the name suggests, it contains the data of the winner of each match where, “1” indicates that the Home Team was the winner of the match, “-1” indicates that the Away Team was the winner of the match, and “0” indicates that the match ended in a draw. The values in the “Winning Team” column were calculated using a simple IF function that is mentioned in our code. The “Winning Team” column was used as our target variable.

#### Dropping Redundant Columns

The “Home Team Second Yellow Card” and the “Away Team Second Yellow Card” columns were dropped immediately from our dataset. This is because, in football, if a player receives two yellow cards, they essentially receive a red card. Therefore, the “Home Team Red Card” and the “Away Team Red Card” columns are equivalent to the “Home Team Second Yellow Card” and the “Away Team Second Yellow Card” columns as they hold the same information and hence, it was an intuitive decision to drop the redundant columns.

#### One-Hot Encoding

We applied the One-Hot Encoding method to transform the following categorial features into numerical features for our model to process them:

1. Home Team
2. Away Team
3. Location[[5]](#footnote-6)

### Scraped Data

Unlike the clean data in the dataset, the scraped data is far from perfection and requires cleansing.

#### Match date

The team names in the location table do not match with those in the dataset. The original dataset makes use of capitalized letters as well as abbreviated team names. Moreover, it preserves the Spanish accent whereas, the scraped data uses full team names in pure English. An example of this can be found in Table 2.

Table 2: Examples of mismatched team names

| Problems | Dataset | Scraped data |
| --- | --- | --- |
| Abbreviation | HUESCA | SD Huesca |
|  | DEPORTIVO | Deportivo La Coruna |
| Accent in Spanish | CÁDIZ CF | Cadiz |
|  | ALAVÉS | Alaves |

Since there were a very limited number of discrepancies, these differences were manually corrected. We did not employ more sophisticated methods such as calculating the Levenshtein distance between two team names since the dataset is finite and the number of mismatched names was limited to less than a dozen. It was more efficient to align the names manually.

Another issue we faced while scraping the data was that three matches were missing from the Sky Sports website, which means three match dates were missing. Fortunately, all of the matches are sorted chronologically in the dataset, therefore, it was not difficult to deduce the match date based on the dates of the neighboring matches.

#### Weather data

Three cities[[6]](#footnote-7) were found to have no historic data on the Weather Underground site. We replaced them with the closest city based on Google Map (Figure 5), assuming that the weather is similar. They are usually within 2 hours’ drive from the actual venue.

Map

Description automatically generated

Figure 5: The closest city from Granada: Málaga (Source: Google Map)

With the cleansed data, we performed Exploratory Data Analysis (EDA), which is depicted in the following section.

# Exploratory data analysis

## Heat Map

图形用户界面

描述已自动生成

Figure 6: The correlation heat map between all features

After crunching all the data, we produced a heat map of all the remaining features. The heat map calculates the correlation between each pair of features. As shown in Figure 6, we can see that when two features have a strong positive correlation, the corresponding position of the heat map tends to be white. When the two features have a strong negative correlation, it tends to be black. When the correlation is low, it tends to be red.

By referring to the heat map, we selected a group of intuitionistic correlations and a group of non-intuitionistic correlations to analyze them.

## Intuitionistic correlations

图表, 散点图

描述已自动生成

Figure 7: Correlation between feature Team Rating and Goal difference

It can be seen from Figure 7 that there is a strong positive correlation between a team's rating and the goal difference (up to 0.892, very close to 1), which is in line with intuition. Since if a team has a higher rating, they are more likely to score more goals than their opponents and therefore, the goal difference would be greater.

## Non-intuitionistic correlations

图表, 散点图

描述已自动生成

Figure 8: The correlation between "Home Team Rating" and "goal difference"

In football, passing offense is an essential means. Therefore, intuitively, the higher a team's pass success rate, the higher their scoring should be (with a positive correlation). However, as seen in Figure 8, there is no correlation between the two (the correlation is only 0.12, close to 0). After our analysis, two reasons may lead to this result: 1. The data set is biased, and it happens that the data in the dataset does not reflect such a relationship. 2. The pass success rate does not reflect the number of times a team passes the ball. A team may pass the ball very few times in a game, and a high success rate cannot help scoring goals.

## Correlation of the latent variables with dependent variable

After scraping and cleaning the precipitation and temperature data for the matches, we tested to see the correlation between the 2 against the dependent variable “Winning Team” to see if they are correlated, and here are the results.

Graphical user interface, text, application

Description automatically generated

Figure 9: Correlation of Rain/Temperature vs Winning Team

As we can see from the scatter plot, there is no strong pattern found between “Rain” or “Temperature” and “Winning Team”, and the Pearson’s Correlation Coefficient of the “Rain” variable (-6.860e-5) or the “Temperature” variable (-0.04900) being almost 0 further backs our claim.

In order to further investigate if temperature is actually related to the outcome of the match, we plotted a box plot to analyze the temperature’s distribution.

Chart, box and whisker chart

Description automatically generated

Figure 10: Box-plot of Temperature on Match Outcome

From the boxplot, despite the random outliers when the Home or Away team won, the distribution looks almost identical, and the mean is roughly the same. Therefore, we conclude that the outcome of the match is not correlated with the temperature that day.

## Importance of the latent variables on prediction

In order to validate whether the latent variables are helping with our prediction, we fitted the data on a simple logistic regression model. However, since the logistic model only outputs 2 values but the actual outcome consists of 3 values (win, draw or lose), we had to change the problem from predicting the outcome of the match to predicting whether the Home team won or not.

By making use of the logistic regression model, we can now conduct the partial F test on the 2 latent variables to see if they actually contributed to making a more accurate prediction.

Graphical user interface, text

Description automatically generated

Figure 11: Partial F test on the 2 latent variables in baseline model

With a p value of 0.6649, we have little to no evidence against H0, therefore we conclude that the 2 latent variables are not contributing significantly to the prediction.

## Insights of the baseline model

The baseline model, albeit not as powerful as the other models, does share some insights that are worth mentioning.

With a pseudo-R-square [1] value of 0.4378 (figure 12), we can state that our model has a relatively good fit on the data, which led us to using the ensemble algorithms later.

Text

Description automatically generated with low confidence

Figure 12: Summary of baseline model (1)

The summary of the baseline model performs the partial F test on every variable individually, and from the test results we can conclude that the following features: “Possession %”, “On Target Shots”, “Clearance”, “Fouls”, “Total Shots”, “Number of Yellow/Red Cards Received” for the Home and Away teams contribute significantly to the prediction, this is done after dropping the variables that have multicollinearity issue which is mentioned in the next subsection.

Table

Description automatically generated

Figure 13: Summary of baseline model (2)

## Multicollinearity

One of the most important assumptions for regression problems is the non-multicollinearity assumption [2], where we assume that there is no correlation between the variables (features) we consider. This is an important assumption to check from a statistical point of view, because in order to carry out valid statistical tests on the model and the data, we would like to test the model’s performance on a variable just by itself, however, if the variable is correlated with another, then there is a third factor in our equation, which may impact the accuracy and validity of the test.

We used the Variance Inflation Factor (VIF), which is an empirical measure of the correlation between every variable with every other variable, to investigate the multicollinearity issue within our data, and we dropped a few variables so all of the variables’ VIF are under 5, which is considered free of multicollinearity from the other variables. It is worth noting that multicollinearity is a problem in inferencing, but not in prediction, which implies our boosting/bagging model is not affected by the multicollinearity issue. The reason why we wish to investigate it is to conduct the following tests in order to understand the underlying structure of our data.

A picture containing timeline

Description automatically generated

Figure 14: VIF on the data variables (VIF <= 5 implies no multilinearity)

# Models and evaluation

## Candidate Models

Making use of an ensembled model has many advantages. Since it reduces the spread of the predictions, it is more robust. It generally has lower variance due to the bootstrapping process during training and therefore it can avoid overfitting. Moreover, the trained model predicts only by comparing input data with the criteria in the nodes. Therefore, we adopted a few ensemble models as our candidate models, including AdaBoost, GradientBoost and Random Forest.

## Model Evaluation

Accuracy can be used when the class distribution is similar, however it is better to make use of the F1-score when there are imbalanced classes. In the test-set, the difference in the number of samples with label 1 and label 0 leads to some imbalance. Therefore, we selected the F1-score to be the metric for evaluating the model. Below is the evaluation of the 3 candidate models.

Table 3: GradientBoost's Classification Report on Test-Set

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| GradientBoost | precision | | recall | f1-score | | support |
| -1 | 0.87 | | 0.96 | 0.91 | | 113 |
| 0 | 0.88 | | 0.64 | 0.74 | | 109 |
| 1 | 0.87 | | 0.97 | 0.92 | | 158 |
| accuracy |  | |  | 0.87 | | 380 |
| macro avg | 0.87 | | 0.86 | 0.86 | | 380 |
| weighted avg | 0.87 | | 0.87 | 0.86 | | 380 |
| weighted avg | 0.84 | 0.84 | | 0.84 | 380 | |

Table 4 Random Forest's Classification Report on Test-Set

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Random Forest | precision | recall | f1-score | support |
| -1 | 0.83 | 0.94 | 0.88 | 113 |
| 0 | 0.81 | 0.56 | 0.66 | 109 |
| 1 | 0.85 | 0.96 | 0.90 | 158 |
| accuracy |  |  | 0.84 | 380 |
| macro avg | 0.83 | 0.82 | 0.82 | 380 |
| weighted avg | 0.83 | 0.84 | 0.83 | 380 |

Table 5 AdaBoost's Classification Report on Test-Set

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| AdaBoost | precision | recall | f1-score | support |
| -1 | 0.86 | 0.83 | 0.85 | 113 |
| 0 | 0.75 | 0.68 | 0.71 | 109 |
| 1 | 0.88 | 0.96 | 0.92 | 158 |
| accuracy |  |  | 0.84 | 380 |
| macro avg | 0.83 | 0.82 | 0.83 | 380 |
| weighted avg | 0.84 | 0.84 | 0.84 | 380 |

## Finalized Model

According to the model evaluation, GradientBoost performed the best on our dataset, hence we selected it as the finalized model. Below are the top 10 most important variables with their feature importance.

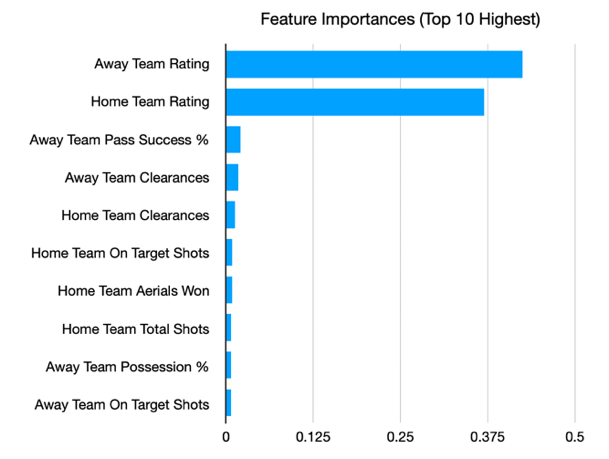


Figure 15: 10 Most Important Variables

According to the features’ importance, our model treats the rating of the two teams as the variables with the greatest impact on prediction.

# Demo program

In this demo, we aim to provide a prediction of the outcome of a match before the actual match. However, the majority of the match data is not available until the match is completed. Hence, we opted to create a dataset of each team's historical performance data over the past 6 years (from 2014 to 2019).

## Construction of Input Vector

To make a prediction using the developed demo application, the user is only required to input the Home team and Away team names, as well as the date of when the match is going to be played and the temperature of the date. The program automatically maps the historical data into the entries of the input vector from the historical performance dataset.

## Example

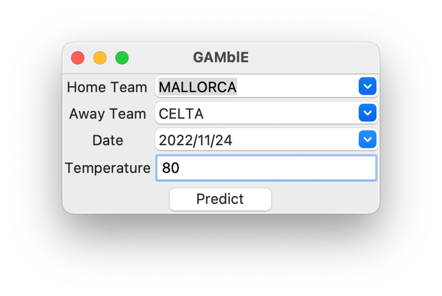


Figure 16: Demo Program UI(a)

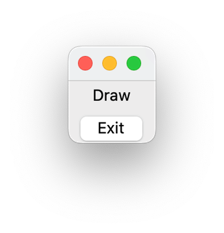


Figure 17: Demo Program UI (b)

After clicking the *Predict* button, the program will pop up a message containing the prediction for the outcome of the match.

# Conclusion

In this project, the La Liga dataset acquired from Kaggle was used to train a neural network. The goal was to implement a gambling application for predicting the outcome of a match. Some external factors, such as weather and temperature, were not included in the dataset. However, it was intuitively assumed that these factors usually tend to influence the outcome of the competition. Therefore, Power Automate was used to obtain the external factors of most matches in the dataset, including weather, location, etc. After analysis, unfortunately, these external factors have little effect on the match's outcome in this dataset. After this, data preprocessing was conducted on the entire dataset to make it cleaner and more suitable for training. After compiling the dataset, EDA (Exploratory Data Analysis) was performed, and several exciting relationships were visualized. Finally, the dataset was trained with various models, and by comparing the performance of each model, the demo application adopted gradient boost as the backbone model to make a prediction.

While we as a team were able to achieve a substantial result, there remains future improvement work that can be done. First, the current model only uses fixed datasets to predict outcomes. In other words, the model is not capable of continuous learning. Therefore, as a future improvement, we can provide real-time data to the model for each game and train the model with the latest data to make predictions in real time.

Secondly, as a future improvement, the external factors on the day of each match can be added automatically, to make the prediction faster and more convenient. As analyzed earlier, the external factors had little effect on the results. However, this may have something to do with the particularity of the dataset and the fact that there is very little data on rainy days, which makes their impact insignificant. However, if these external factors are added in the future, and the model continues to learn, their influence could grow, and the model could become more powerful. In addition, football is a sport with many players who tend to have a more significant influence on the outcome of the match results. Thus, as a future improvement, we could introduce player-specific analysis; for example, some players may be unable to play in some competitions due to injury, which should be considered.

Finally, because the gambling application is still preliminary, the UI is still relatively simple, and the functionality is relatively low. As a future improvement, we can beautify the UI interface and add more functions to the application.

# Acknowledgment

We would like to take the chance to thank Dr. Chan for her guidance given to this project. Her suggestion of adding external factors to our model led us to explore the works of data collection and data cleansing, which are seemingly simple but in fact tedious and daunting. Though the weather was later found to have little relevance to the match result, the learning process was fruitful and insightful.

The project could not be possible without the seamless cooperation among the five team members. All members are involved in the discussion of building the model and interpreting the results, and preparation of the presentation and the final report.

Here is our division of work for other tasks:

Harsh Sunil LAKHANI Data preprocessing, modeling, editing this report

Ho Pun CHENG Data collection

Yau Wai YIM EDA, conclusion, and future work

Ye HUANG EDA, modeling

Yueyin TAN Modeling, demo program

# References

|  |  |
| --- | --- |
| [1] | P. Allison, "Statistical Horizon," 13 Feburary 2013. [Online]. Available: https://statisticalhorizons.com/r2logistic/. |
| [2] | S. Wu, "Multicollinearity in Regression," 19 May 2020. [Online]. Available: https://towardsdatascience.com/multi-collinearity-in-regression-fe7a2c1467ea. |

1. https://www.kaggle.com/datasets/sanjeetsinghnaik/la-liga-match-data [↑](#footnote-ref-2)
2. A league year usually spans from August to May in the subsequent year. [↑](#footnote-ref-3)
3. https://www.wunderground.com [↑](#footnote-ref-4)
4. https://www.skysports.com/la-liga-results/ [↑](#footnote-ref-5)
5. This is a feature from the scraped data. [↑](#footnote-ref-6)
6. Granada, Pamplona and Vitoria-Gasteiz [↑](#footnote-ref-7)