BEAT THE BOOKIE

COMP0036 - GROUP ASSIGNMENT

Group F
Department of Computer Science
University College London
London, WC1E 6BT

January 3, 2023

1 Introduction

Betting on football games has been a large business for decades. Many bookies now employ machine learning algorithms and mathematical models to predict the outcome of games in the English Premier League using historical data. The goal is to match the accuracy of the predictions (around 53%) or even beat it.
The starting point is the given data to which more parameters have been added and then transformed to be simpler and easier to process. Afterward, different machine learning algorithms were trained using the training dataset to determine the most accurate one by validating them against the evaluation dataset. The best performing algorithm (i.e. the most accurate) will be employed to predict the outcome at Full Time of the games played on the weekend of the 14^{th} of January.
The chosen model's accuracy was over 54% which results in approximately 5 out of 10 correct match outcome predictions.

2 Data Transformation & Exploration

Data transformation is used to allow for data to be processed better in the notebook as well as to improve accuracy. Data was visualised mainly using scatter plots for a visual understanding of the correlation between points so further changes could be made to the method.

2.1 Data Transformation

A series of transformations are made to the provided and additional data. These include the digitisation of non-numerical variables to allow them to be better processed in the notebook. For most of the data, a rolling sum was found in order to represent a team's form and provide a more accurate and up-to-date result. Additionally, some data is combined into differences rather than separate points to have fewer parameters thus making the algorithm faster. The transformations' purpose is to align data with the processing method and create better accuracy.

2.1.1 Cleaning and Pre-Processing

Firstly, the data needed to be cleaned. Day, month, and year columns were added from the date given to obtain that set of data in singular form for usability. Half-time data was also removed from the data set as this was seen as unnecessary for a match predictor made before a game begins, with no knowledge of that particular game. The given data on shots taken and shots on target were used to find a percentage to show each team's shot accuracy. The yellow card data was also removed, and dimensionality reduction was conducted by converting the home and away goal data into a difference between them to reduce the number of parameters. The next processes followed were to remove the empty rows in the data, remove the redundant 'date' column, and reduce the 'year' data to start at 0 and end at 22. The removal of redundant data and fixing of the structure of the data will make it easier to use and manipulate, while increasing model accuracy by not feeding the algorithm unnecessary data.

2.1.2 Digitisation

Certain data points were required to be assigned numerical values to be able to analyse them, which increases the usability of the data. Team names were replaced with a unique identifier based on the number of teams that competed in the Premier League over the specified time frame, and this was then normalised with respect to the total count of each attribute.

In order to convert the referee and manager data to numeric values from strings, all of the data points were ordered by the full-time result. Then based on this order all of the unique referees were ranked, this gives a numeric value to the referees and introduces the relationship for which referees were more likely to play in a match that ends in a home win and vice versa. This ranking was then normalised to keep the data points between 0 and 1.

This process was repeated using all of the manager data with the same method, with the ordering and ranking via the full-time result. This leads to all attributes in the data being numeric and this can be processed by the machine learning algorithms.

Another solution would have been one-hot encoding. This allows to transform individual features, such as teams, into a singular binary value. This has the potential to greatly increase performance as it makes data easy to read and manipulate by the algorithms. However, the benefits come at the expense of the creation of an extremely dense array.

With over 40 teams, 100 referees, and 100 managers, we would be generating over 300 binary features within our data. The cost of this: time and computing power[1].

2.1.3 Rolling Mean

To get a metric of form for each team, a rolling mean was used. Various sources in Section 3.1.1 show the importance of current form in predicting the outcome of matches and thus the rolling mean was found for most of the numeric metrics. Initially, the rolling mean included the current match itself which is incorrect as it would mean that the predictions of games would need knowledge of the final outcome of the game it is attempting to predict. To counteract this the rolling mean was offset by one so it takes the past three matches instead, resulting in the mean leading up to the game.

Using a rolling mean aids in the analysis of the trends in the data and a longer rolling correlates to longer-term trends in data. Three matches for the mean was chosen as a value too high would include more outdated data that would have a smaller correlation to the team's current performance. A value too low would not show the full desired form/ trends.

The rolling mean of the Budget data was also processed with the rolling mean of the last three seasons of the team. This allows for a metric on how much money has been going in and out of the club on average in the past three years.

2.1.4 Feature Scaling

Feature scaling is an important preprocessing step that allows for the transformation of the features in the dataset so that they are all on the same scale. This is important as many of the algorithms used are sensitive to the scale of the input features. For example if two features are not on the same scale, one might dominate the other when comparing for similarity and finding the distance between data points.

There are two main methods of feature scaling which are normalisation and standardisation.

Normalisation involves scaling an input feature so that it always ranges from $0 \rightarrow 1$ and is done by subtracting the min value of the feature from every value. Then dividing by the total range of the feature.

Standardisation consists of scaling an input feature so that it has a mean of 0 and a standard deviation of 1. This is done by subtracting the mean value from all the values then dividing by the standard deviation. Many algorithms such as logistical regression and support vector machines assume the data to be normally distributed so could behave in an unwanted fashion if data is not standardised.

2.2 Data Exploration

Correlation vectors and matrices were used to verify whether the selected features have an impact on the full time result. This shows how much of a linear relationship each variable has with the full time result. For example, when the correlation is close to 0, there is no relationship. When it's close to 1, a positive linear relationship exists where a high value of the feature is highly linked with a home win. Similarly, when the correlation is close to -1, a negative linear relationship exists where the higher the parameter value, the more likely the home team will lose meaning an away win.
Looking at the correlation vectors in table 1, the feature with the highest positive correlation to FTR is the league position of the away team. This means that the lower the away team is placed in the league (placing 15th is lower than placing 5th), the more likely it is for the home team to win the game. In contrast, the variable with the highest negative correlation is the home team's league position which makes it more likely for the away team to win. A mean of the absolute values of each correlation vector was taken to then evaluate the overall feature correlation:

Day	-2.46%	ForHomeFoul	-2.60%
Month	-1.43%	AgainstHomeFoul	-6.27%
Year	-3.54%	HomeBudget	13.04%
HomeLeaguePos	-24.29%	AR	1.70%
AwayLeaguePos	25.05%	AS	-10.68%
HR	-2.21%	AST	-11.86%
HS	10.18%	ScoreA	-4.83%
HST	11.19%	ForAwayGoals	-8.98%
ScoreH	8.06%	AgainstAwayGoals	5.72%
ForHomeGoals	7.09%	ForAwayCorner	-6.00%
AgainstHomeGoals	-6.90%	AgainstAwayCorner	6.86%
ForHomeCorner	6.78%	ForAwayFoul	-6.00%
AgainstHomeCorner	6.27%	AagainstAwayFoul	-2.23%
AwayBudget	-15.27%	CorrelationMean	8.056%

Table 1: Features Correlation Results

The correlation of these features can be visualized using scatterplots showing gaussian distribution as seen in figure 1 below:

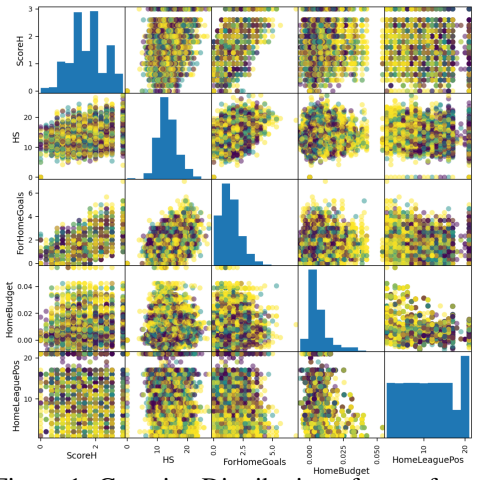

Figure 1: Gaussian Distribution of some features

Some data was simplified which increased the correlation average by 1.065%. Adding the manager information and normalizing the data increased the correlation average by a further 2.299%.

Month	-1.43%	HS	10.18%
Year	-3.54%	AS	-10.68%
HomeTeam	22.63%	HR	2.21%
AwayTeam	-20.32%	AR	1.70%
Referee	3.97%	HST	11.19%
GoalsPredDifference	11.21%	AST	-11.86%
CornerPredDifference	9.23%	HomeBudget	13.04%
FoulPredDifference	1.21%	AwayBudget	-15.27%
ScoreDiff	9.30%	HomeManager	25.87%
LeaguePosDiff	-34.01%	AwayManager	23.39%
HSTP	3.93%	ASTP	-5.05%
CorrelationMean	11.420%		

Table 2: Features Correlation Results post-Transformation

Some of the newly generated features have a much higher correlation with the FTR which makes the model more accurate. A Gaussian distribution can also be observed on scatterplots:

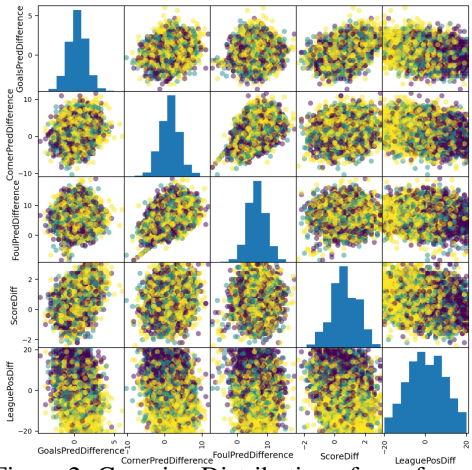

Figure 2: Gaussian Distribution of new features

3 Methodology Overview

The methodology followed started with a thorough literature review into the potential data sources that have a large influence on match results, and into general approaches used for similar algorithms to this topic. Research was also conducted into techniques for model training and evaluation. Decisions were then finalised on the best sources of data to add to the algorithm, what should be left out, and the optimum approaches to obtain the best performing algorithm.

3.1 Literature Review

First of all, a full literature review was undertaken to find the best performing models for similar projects, to understand the ideal data sources to be included that have the largest effect on performance, and to analyse which algorithm performs best for these sets of data.

3.1.1 Additional Data Review

A lot of research was undertaken in finding the most decisive factors that influence football matches, to gain an idea of the best additional data to add to the algorithm. This was then combined with group discussions to reach a final decision.

Firstly, a source that predicted German Bundesliga games [2] noticed a pattern where accuracy of the model is increased when the data used is closest to the date of the predicted game. This suggests an importance in the form of the certain teams, and that data from recent years may be more useful to the algorithm than data from many years prior. This influenced the use of the rolling mean data transformation used previously also, as this approach would be more focused on the recent form of the teams.

Next, a research paper that analysed premier league team performance [3] argued that data is only usable when the data is compared. Therefore, using differences of the statistics truly shows one teams superiority in comparison to another, for example using goals scored statistics to find the differences in number of goals scored by each team that is competing, as seen in the dimensionality reduction in Section 2.1.1. Another argument made is that there is intangible data that is not represented by numbers, such as difference in team performance caused by team usability. This infers that factors like the manager and owner of the teams may have a large impact, alongside form which was researched previously. Another source [4] further proves the importance of managerial data in particular, and details the importance of the managers and its effect on the form of the team, which it also details as an important factor that can have a large effect on the outcome of football matches. It describes that teams that have lost more in recent match weeks feel more pressured to win a game whereas those that have won more recently may have become more complacent in training and feel under less pressure to win. However, in contradicting terms, teams in a good run of form may be more likely to win due to increased confidence and momentum that the team brings into a game, where a team in bad form may lack confidence to take risks needed to win games. These patterns may differ for each team depending on the players and their mentality, so form data for each team is important.

A research paper that focuses on form at the start of the season in addition to financial budget [5] shows that form is a good metric once again. However, form can be influenced greatly by financial budget of the team. This is because a larger financial budget should lead to a more expensive, and therefore better team of players. A better team of players due to their quality and better mentality are therefore more likely to maintain a good run of form, and more likely to turn around a bad run of form. Budget therefore expresses the importance of form in different teams, and will also have a large effect on form over a long period of time.

Other sources were also briefly studied to judge the effect of different measurable stats. One source [6] judged that the use

of shots and goals could be limited due to their randomness and variation despite that a pattern inevitably prevails over time. Using a resource that could take into account the quality of shots and which team is actually more likely to score despite the difference in shots taken could be more valuable. Also, another source [7] states that the availability of key players in teams inevitably has a significant effect on the full time result of that teams game. The source proves this effect by assigning different players quality scores, and showed the effect of missing the higher quality scored players in games they were injured to have a direct correlation with worse match results. This approach could be seen as limited to some extent however, as there will be some bias and opinion involved in choosing a metric to judge a players quality rating.

3.1.2 Approaches Review

Research was also undertaken in determining the best approaches that are most likely to give the best results, based on various attempts at similar problems completed by others. Methods that are found to have worked well in similar cases can then be tested for this specific problem, and an optimum solution can then be found as a result.

Firstly, a research paper was analysed that also predicted EPL football matches using many machine learning approaches [8]. One interesting find of this paper was using a support vector machine method (in this case RBF and linear SVM) produced very good performance for home wins, good for away wins, but bad for draws. The algorithm had a recall value of 0.8 home wins, suggesting a very low false negative rate. This suggests potential of using different kernel SVM techniques with our data. Draws however had a very high false negative error which means that this algorithm on its own may not be an optimal solution.

Next, the research on predictions for German Bundesliga games [2] analysed previously makes use of a random forest algorithm. This shows good results in predictions of league results on a season basis, and therefore may have good use for individual match predictions. The random forest approach also helps to predict draws [8]. This source found it produced a balanced result across all three classes, and was able to overcome shortcomings in the SVM models relating to poor performances regarding draws.

Next, research was studied that involved predicting EPL matches in a previous season using logistic regression [9]. It describes the method as easy to implement and understand, while also providing an improved insight through the estimated coefficients. It showed a very high model accuracy, the best being 69.5%. This accuracy value is limited however, as it is based on the accuracy of the model towards the training data and not an unseen test set of data, so these results may not be reproducible in the case studied in this report.

Finally, a report detailing the use of the K-Nearest-Neighbours approach to predict football matches in comparison with using logistic regression [10] was studied. This found that logistic regression performed better for the particular study of the research, but that a KNN approach can be useful with a good model. In general these algorithms give high classification accuracy and speed, and should work well with a normalized form of the dataset produced. It may also predict draws better than other algorithms tested, so there are potential uses in this regard.

3.2 Overview of Model Training and Evaluation

Research into what methods and techniques are used in the field regarding model training and evaluation is carried out and summarised. Approaches to hyperparameter optimisation was also investigated.

3.2.1 Training

Model training involves using a subset of the finalised data set to learn the complex relationships between the various input features and the full time result. The goal is to find the optimal set of parameters for each of the machine learning models in order to have the highest accuracy in predicting the full time result of games and data it has never seen before.

3.2.2 Hyperparameter Optimisation

All of the different machine learning models used, have hyperparameters which are parameters that are not learnt by the data but are set in the code to determine the learning behaviour of the model. Hyperparameters need to have different values depending on the data set and the problem being solved. Thus hyperparameter optimisation is needed and there are a few methods such as manual search, grid search and random search [11].

Manual search involves manually going through and selecting a set of values for the hyperparameters and manually evaluating the performance of that model, then tweaking the values to improve the accuracy.

The grid search method consists of specifying a range of values for each hyperparameter and training the model for every single combination. Due to the fact that every single point in the hyperparameter space needs to be trained and evaluated this is the most systematic and thorough method with the highest computational cost.

Random search means to randomly sample combinations of the hyperparameters for training and evaluating the model. This method can be more efficient that grid search if the space is significantly large but might be less thorough.

Due to the relatively small data set being used, and taking into account the computational power accessible to us, grid search was used to ensure the best possible values were found.

3.2.3 Evaluation

This leads to the model evaluation where the remaining data that has never been seen before by the model is used to test the accuracy of the model. It is imperative that this data has not been seen by the model already to give a more realistic assessment of the model's performance.

This method of evaluating the model might not give a true indication of the performance of the model as the testing data given to the evaluator could be skewed and give a biased result. In order to counteract this, k-fold cross validation can be used to give a better, more robust evaluation of the model.

This involves dividing the data in k number of "folds" where each fold is used as a validation set in the training once, and the rest is used as training data. This is then repeated k times with all folds being used as validation data only once as seen in Figure 3. This method allows for the model to be trained and evaluated on different subsets of the data increasing the robustness of the evaluation, as well as aids in identifying overfitting or underfitting [12]. Too large of a k value could lead to a significantly slow evaluation time and a small k value would lead to a less robust performance metric.

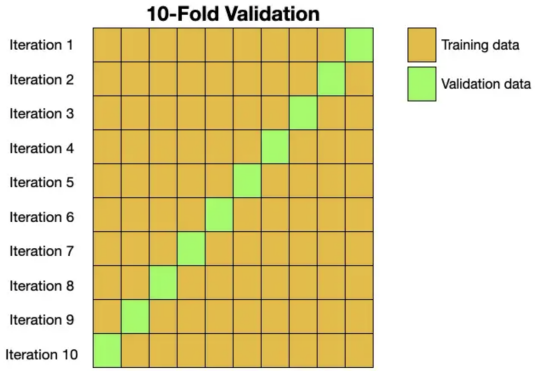

Figure 3: Example of a 10 fold cross validation, showing how different validation subsets are used in each iteration. (Lee, 2021)

In this study a value of $k = 10$ was chosen as a trade-off between computational time against the robustness for the training and evaluation.

3.3 Additional Data Sources

Additional data sources were collected based on group discussions and background research into the topic to determine important areas that could be looked into. Many areas were then explored, some tested within the algorithm and some disregarded because of limited data availability or limited benefits that it could provide.

3.3.1 Recent Data

The provided data set included match statistics for games up until 30^{th} October 2022, to more accurately represent live form and allow for further data to train from, match results and statistics up until 1^{st} January 2023 were found and included. This aligns with the literature that states more recent data may be more useful than data from many years prior.

3.3.2 Budget

One additional data source used was the budget of each team per season, due to the relationship found between budget and form, which has a direct correlation with team performance. Transfermarkt provides data on the purchases and sales of each team for each year from 2000-2022 [13]. The budget was defined as the net spend of the teams per season (including the summer and winter transfer windows), which is the difference between the cost of the incoming transfers and the cost of the outgoing transfers. This data should provide a good insight into the strength of the teams and a pattern should be observed where a higher net spend correlates to better performance. If a team is selling fewer players and buying more players also, then the squad should be strengthened. There are potential limitations to this, as some teams have lower-skilled scouts and sporting directors to identify the best value players available. Also, many lower-placed teams may make a large sale and use the money to strengthen in many areas, and this is unable to take into account the potential benefits that may bring. Overall, the data should provide an improvement in the performance of the algorithm despite the anomalies.

3.3.3 Manager / Coach

Another additional data source used was the manager of each team for each game, as it was implied to have a large effect from the conducted research. There are many examples of teams improving and declining vastly due to the appointment of different managers. For example, it can be observed that when Manchester United changed their manager in 2013 after a very successful run from Sir Alex Ferguson, they went from consistent finishes inside the top 3 of the league to a 7^{th} place finish the first season after he left. Many managers included also managed a number of the teams within the data set, therefore providing good training data. The manager of every team for each month since January 2000 was easily accessible [14] [15] and added to the data.

3.3.4 League Position

Another additional data source used was the league positions of each team for each season in the data set. This data source is a very easily obtained source that has several potential benefits [16]. One potential benefit could be to aid predictions for teams who outperform expectations in a certain season. For example, Leicester City in the 2015/16 season won the league, despite managing no higher than a 13^{th} placed finish beforehand, and even not managing to compete in the premier league in many seasons. League positions aid to understand the performance of a team in a certain season, and is a long-term indicator of the team's form, which is known to be an important factor in determining results from research, especially when used alongside the budget data found. The specific data collected is the league position that the team finished in the previous season, so the 2022 results can be used for the predictions of the weekend of the 14^{th} January 2023 that is required.

3.3.5 Other Considerations

Other considerations of data that could be added were made also. Firstly, a new football stat called expected goals (xG) was considered due to its wide use in modern football analysis. xG measures the quality of the shot based on several factors, like the type of shot, the angle it is taken from, and the distance from the goal to name a few. This will therefore give a measure of the probability of a shot being converted to a goal, where an xG of 0 suggests no chance of scoring and an xG of 1 suggests a certain goal. As a result, this data would give a much more accurate and fair representation of how many goals a team should have scored or conceded in comparison to shots taken, which doesn't consider any context. Like shots taken, it is likely however that if a random game is picked, a team is

more or less clinical than their xG predicts, but throughout a long period of time this usually balances out where a teams goals scored or conceded will be very similar to their xG for or against. As the algorithm is being trained based on over 20 years of data, this would be a good use. However, because xG is a new statistic, the data for premier league matches prior to 2014 does not exist. Therefore, because it is limited to a short period within the training data, a decision was made to disregard this data despite its potential.

Another parameter considered to have a large effect on some teams is the current owners of the football clubs, which is another form of data that is not represented by numbers which were researched and found to have an effect on results. For example, a large improvement in the performance can be observed by Manchester City and Newcastle in the years after the respective takeovers by richer owners. On the whole, these two are an anomaly, however, and unlike managers, owners tend to be consistent and do not swap between football clubs due to the complexity of a takeover of an expensive football club. The results of these large takeovers should also be reflected in the budget data used as the main effect a new owner will have is to provide extra funds to the team. As a result, this data was seen as limited and not used for the model.

Injuries were also studied to determine the effect that they have on performance. More players getting injured for a certain team should lead to a performance drop for that team. In theory, there should also be a pattern where the teams who have the lowest budget would suffer more from injuries due to their lack of squad depth as a result in comparison to other teams. Despite this, injury data per team dating back enough years is very difficult to obtain, as statistic websites usually keep track of when a player gets injured but will not then add this to a tally that shows respective teams' injuries in a period of time. This was therefore disregarded.

3.4 Approaches

A number of different approaches were attempted to find the optimal approach. The optimal will be the algorithm that performs best on the test set of data used. The approaches used will be supervised learning algorithms as it will use a labelled set of data to learn features, so it will then be able to predict results based off of this data.

3.4.1 Logistic Regression

The first machine learning approach used was logistic regression, in an aim to try to replicate the model accuracy obtained in other studies. Regression analysis is a method to model a relationship and predict a value of a dependent variable, Y, based on values of independent variables, X. The algorithm finds a regression line or curve that best describes the relationship between the variables. Logistic regression follows the logistic function in equation 1:

$$G(s) = \frac{1}{1 + e^{-x}} \tag{1}$$

Where x is the input to the function. In this case, the problem is a multinomial logistic regression problem, because it has two or more discrete outcomes. The dependent variable will be the Full Time Result (FTR), with possible outputs of home win, draw, or away win.

3.4.2 K-Nearest Neighbour (KNN)

The next approach used was a K-Nearest Neighbour (KNN) algorithm due to its applicability for the data in use with normalised values, and to verify whether the performance of this algorithm is actually lower than the other methods, especially regarding draw prediction. The algorithm arranges known data into a space defined by selected features. Then, when new data is supplied to the algorithm, it will be arranged depending on the classes of the 'k' closest data points to determine the class of the new data. A value of 'k' will therefore need to be chosen, which will be the number of nearest neighbours for the algorithm to consider when a prediction is made. The distance between the neighbours and the data points in this case will be calculated using Euclidean distance. The algorithm is then able to make a prediction of the result of a match based on the class that the data point is defined to be in [17].

3.4.3 Polynomial Support Vector Machine (SVM)

The next approach used was a polynomial support vector machine (SVM) algorithm, the first kernel method used to test if the good performance can be reproduced and if limitations in draw predictions appear. A SVM finds a hyper-plane (a boundary that divides the plane, one dimension lower than the space being considered) which then creates a boundary between the types of data. This type of problem involves non-linearly separable data, so kernelized SVM is used, where a kernel is a measure of the similarity between data points. In this approach, a polynomial kernel will be used, which takes a degree parameter that controls the complexity of the model and the computational cost of the transformation. The equation that this algorithm is based off is shown in equation 2.

$$K(x, y) = (xy + C)^d \tag{2}$$

Where x and y are the input data points, C is a constant, and d is the degree of the polynomial.

3.4.4 Radial Basis Function Support Vector Machine (RBF SVM)

The next approach used was a Radial Basis Function (RBF) SVM algorithm, the second kernel method used to test if the performance can be reproduced and if draw limitations similarly occur again. This is the same method as the one used in the studied literature. This method of kernelized SVM uses an RBF kernel, which measures the similarity between the data points by assigning a high value to data points close together and a low value to data points far apart, both in terms of Euclidean distance. This kernel allows the SVM to learn complex nonlinear relationships in the data. The equation that this algorithm is based off is shown in equation 3.

$$K(x, y) = e^{-\gamma||x-y||^2} \tag{3}$$

Where x and y represent the input data points, γ is a parameter that determines the width of the kernel, and ||x-y|| is the Euclidean distance between x and y.

3.4.5 Random Forest

The final approach used was a random forest algorithm, to test if overall high performance and usefulness in predicting draws can be replicated. This algorithm trains multiple decision trees, which are individual models, on different subsets of the data provided, and then takes an average of the predictions that the decision trees find to then make a final prediction. This method is useful to avoid overfitting, a potential problem when there are so many variables and statistics that can have an effect on the result of a football match, so a lot of different data will be incorporated in the model. It avoids this by reducing the variance in the predictions, which can then lead to a performance improvement when tested against unseen data.

4 Model Training & Validation

Using the finalised data, various machine-learning approaches were used to find the algorithm with the highest performance. The EPL-training data set was split into a training set and a testing set. Hyperparameter optimisation and K-fold cross-validation was carried out on the training data to find the best combination of hyperparameters and to increase the robustness and decrease the probability of testing on biased data.

4.1 Model Training

For the training of the models, the whole data set of 8486 games is split into a training and testing data set with an 80/20 percent split. Where almost 6800 games are used to fit the model and adjust the parameters of the various machine learning algorithms.

When splitting the data, the algorithm was split in a stratified manner so that the proportion of target data ie Full-Time Result data was the same in both subsets.

4.2 Hyperparameter Optimisation

In this study, grid search was used due to it being more systematic and efficient than manual search, despite the computational cost and time taken to search the possible large parameter space.

For each machine learning model that was investigated, all of the possible hyperparameters were found in the Scikit Learn documentation. Then using the grid search all of these combinations were searched using a range of exponentially increasing values for numeric hyperparameters such as 'C' (the inverse of regularization strength). The numpy function of `logspace` was used to help in determining the appropriate order of magnitude of the numeric hyperparameters. Once the grid search had finished finding the most optimal value of the large range, the range was then decreased in size, resulting in finding a more optimal hyperparameter.

4.3 Model Validation

In this study, k-fold cross validation was used for the hyperparameter optimisation with a $k = 10$. The accuracy of each model and the `classification report` function was used to evaluate the performance of the various algorithms as seen in Section 5. This leads to following four metrics [18]:

1. Precision: Percentage of correct positive predictions relative to total positive predictions.
2. Recall: Percentage of correct positive predictions relative to total actual positives.
3. F1 Score: A weighted harmonic mean of precision and recall. The closer to 1, the better the model.
4. Accuracy: The percentage of correct predictions.

5 Results

5.1 Results Analysis

Using the `classification report` function, the accuracy of each metric was displayed in tables for evaluation:

Random Forest	Precision	Recall	F1-score
Away Win	0.51	0.50	0.51
Draw	0.41	0.07	0.12
Home Win	0.56	0.82	0.67
Accuracy			0.5330
Macro Average	0.49	0.46	0.43
Weighted Average	0.51	0.54	0.48

Table 3: Random Forest Classification Report

Poly SVM	Precision	Recall	F1-score
Away Win	0.55	0.42	0.47
Draw	0.22	0.03	0.05
Home Win	0.54	0.87	0.66
Accuracy			0.5365
Macro Average	0.44	0.44	0.39
Weighted Average	0.46	0.53	0.45

Table 4: Polynomial SVM Classification Report

RBF SVM	Precision	Recall	F1-score
Away Win	0.50	0.49	0.49
Draw	0.00	0.00	0.00
Home Win	0.55	0.85	0.66
Accuracy			0.53
Macro Average	0.35	0.45	0.38
Weighted Average	0.40	0.53	0.45

Table 5: RBF SVM Classification Report

Logistic Regression	Precision	Recall	F1-score
Away Win	0.50	0.54	0.52
Draw	0.40	0.01	0.03
Home Win	0.56	0.83	0.47
Accuracy			0.5412
Macro Average	0.49	0.46	0.41
Weighted Average	0.50	0.54	0.47

Table 6: Logistic Regression Classification Report

Looking at the F1 score of each model, it can be seen that the Logisitc Regression discussed in 3.4.1 has the highest accuracy at 54.12%. However, the downfall of SVM is that the model does not predict any draws as seen on the second row of table 4. Therefore, the draws were predicted using the Random Forest method discussed in 3.4.5. This said, the Polynomial Support Vector Machine and the Random Forest algorithms could be combined to generate the most accurate results, which aligns with the literature in regard to random forest being able to

overcome the shortcomings in SVM models regarding draw predictions.

Forest & Poly SVM	Precision	Recall	F1-score
Away Win	0.53	0.45	0.49
Draw	0.39	0.08	0.13
Home Win	0.55	0.84	0.67
Accuracy			0.5353
Macro Average	0.49	0.46	0.43
Weighted Average	0.51	0.54	0.48

Table 7: Classification report of combining Random Forests with Polynomial SVM

This method sacrifices some of the accuracy in predicting home and away wins as seen in the recall percentage difference between tables 4 and 7 which is about $2-3\%$. However, this gives a better estimate for predicting draws. Yet it isn't an improvement compared to the best predictor.

5.2 Further Model Development

As witnessed in the previous data the idea of merging algorithms has the potential to yield better results in some areas, as can be witnessed in Table 6. However, a quick interpretation of data and a fast solution will never be able to replace a well-constructed algorithm.

Thus is born the voting classifier. By comparing predictions of multiple, pre-optimized, algorithms, the downfalls that they all have encountered can be overruled by gaining from the benefits of other algorithms. Requiring a high level of data processing, it is the job of a secondary algorithm to sort out the selection. The method of voting chosen was "Hard" which only takes into consideration the classes given and not the probability related to this prediction, compared to "soft" where the probability is the main source of the voting method.

SK Learn's voting classifier was thus used combining our Linear Regression, and Random Forest algorithms with a 3rd Linear Regression algorithm with a balanced weight distribution. With a final accuracy of 54.53 % which beats our previous best by 0.41%. Furthermore, the F1 score increased by 8.3% on average leading to believe that the increased complexity of the model has indeed lead to a better outcome.

Forest & Poly SVM	Precision	Recall	F1-score
Away Win	0.50	0.57	0.53
Draw	0.45	0.04	0.07
Home Win	0.57	0.81	0.67
Accuracy			0.5453
Macro Average	0.51	0.47	0.42
Weighted Average	0.52	0.55	0.48

Table 8: Classification report of the Voting Estimator

Figure 4 highlights the different areas where the model is most and least accurate. For example, the algorithm predicted a certain number of games to be home wins of which 81% were correct predictions. Meaning that there is a high level of certainty when home wins are predicted and nearly 60% (57) of away wins correctly predicted which is more promising that the average 52% which has been seen in the models (with a maxima of 55%).

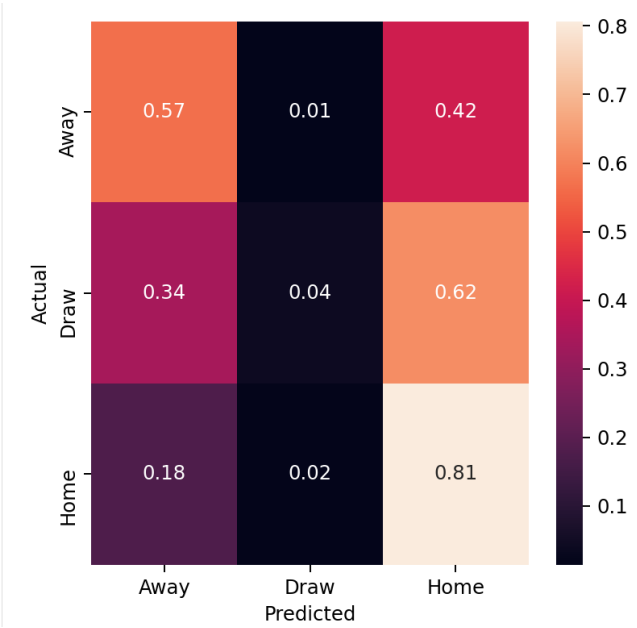

Figure 4: Final model accuracy

6 Final Predictions on Test Set

Using the combined method of Random Forest and Polynomial Support Vector machines, a prediction was made for all 10 games of matchday 20 of the English Premier League taking place on the weekend of January 14^{th}.
To perform these predictions, the data regarding current managers, referees, and budgets was added to get the most accurate prediction possible. The budget was important since Liverpool just signed in-form Cody Gakpo which increases their budget post-2023 by £45 million [19].
The same transformations performed in section 2.1 were carried out to match the training format. The outcome of these predictions is displayed in table 9:

Date	HomeTeam	AwayTeam	FTR
14/01/2023	Aston Villa	Leeds	A
14/01/2023	Brentford	Bournemouth	D
14/01/2023	Brighton	Liverpool	A
14/01/2023	Chelsea	Crystal Palace	D
14/01/2023	Everton	Southampton	A
14/01/2023	Man United	Man City	D
14/01/2023	Newcastle	Fulham	A
14/01/2023	Nott'm Forest	Leicester	A
14/01/2023	Tottenham	Arsenal	A
14/01/2023	Wolves	West Ham	A

Table 9: Final Predictions on week of the 14^{th} of January

7 Conclusion

7.1 Considerations of Findings

To conclude, in comparison to the target 53%, which is the 'gold standard' that the bookies are able to predict games at, the model managed to get an accuracy of 54.53% for the training data. This is therefore higher than the bookies for this particular set, and suggests a very good model.

It is however not flawless. Indeed, it constantly fails to predict draws, and when it does, the certainty of success is quite low (around 4%).

The number of draws that were lost is not negligible yet, with the data set that has been described in this paper, it seems improbable that this will change. Indeed, if we look at more proficient methods of predictions, only complex random forests are able to predict draws with a 30% accuracy and achieve an overall accuracy over 66% [20]. The limit is small, yet it is the only thing stopping this model from breaching the 60 percent mark.

7.2 Future Developments

Many avenues can be explored for future research that can further improve or validate the model created.

One potential area of future work is to test the inclusion of parameters that were disregarded. For example, the use of xG would likely be very beneficial in modern predictions despite that it has limited availability to only recent years. According to the literature review in Section 3.1.1, using data closest to the date of the predicted game increases accuracy, so using a method to only add this data for later years could even prove to be beneficial to improving performance.

Another obvious area would be to research and test more factors that could affect the outcome of matches. For example, more statistics outside of match statistics which were not considered in this report could show correlations to having an impact on a team's performance, like the distance travelled by the away team to their away game. In modern football, due to transport advances and larger budgets for overnight stays, a pattern may be discovered where a minimum occurs for performance at a certain distance away that will be too close to travel by plane and stay overnight, but far enough a way to be a long, exhausting drive. Other data similar to this could aid in improving the algorithm further. For match statistics, other statistics could be tested also, for example analysing teams head to head records to see the effect of certain matches. Derby matches are usually much more contested and close due to the pressure of the occasion, so these matches in particular could be predicted better by modifying the model in this way.

Another future avenue is by utilising some form of a neural network model to learn the data rather than the methods used in this report, which may yield better performance results, and potentially fix the issue with draw predictions. For example, Keras or TensorFlow could be exploited, and this could validate whether our model choices were correct or if deep learning techniques could be a more optimal solution.

Another potential task in the future would be to test the algorithm long-term and see if it can beat the odds given by the bookies. A method for doing this is by simply using the algorithm for many weeks of the EPL season and calculating the gain or loss that would be made by betting using odds given by various betting apps. Keeping track of this thoroughly could validate the model well, and show if it may only perform well in low odd games so there is no profit margin, be less accurate so that it is in a loss, or if the model has officially been able to 'Beat the Bookie'.

References

[1] Seger, C. (n.d.). An investigation of categorical variable encoding techniques in machine learning: binary versus one-hot and feature hashing. [online] Available at: https://www.diva-portal.org/smash/get/diva2:1259073/FULLTEXT01.pdf.

[2] Goller et al. (2018). Predicting Match Outcomes in Football by an Ordered Forest Estimator. [online] ResearchGate. Available at: https:+//www.researchgate.net/publication/328486514_Predicting_Match_Outcomes_in_Football_by_an_Ordered_Forest_Estimator

[3] Carmichael, F., Thomas, D., and Ward, R. (2000). Team Performance: The Case of English Premiership Football. Managerial and Decision Economics, 21(1), 31–45. [online] Available at: https://www.jstor.org/stable/3108117

[4] Audas, R., Dobson, S. and Goddard, J. (2002). The impact of managerial change on team performance in professional sports. Journal of Economics and Business, [online] 54(6), pp.633–650. doi:10.1016/s0148-6195(02)00120-0.

[5] Journal of Sports Sciences. (2015). Just how important is a good season start? Overall team performance and financial budget of elite soccer clubs. [online] Available at: https://www.tandfonline.com/doi/abs/10.1080/02640414.2014.986184?journalCode=rjsp20

[6] Rathke, A. (2017). An examination of expected goals and shot efficiency in soccer. Journal of Human Sport and Exercise, 12(Proc2). doi:10.14198/jhse.2017.12.proc2.05.

[7] Chinwe Igiri, Enock Nwachukwu. (2014). An Improved Prediction System for Football a Match Result. [online] ResearchGate. Available at: https://www.researchgate.net/publication/273164409_An_Improved_Prediction_System_for_Football_a_Match_Result

[8] Baboota, R. and Kaur, H. (2019). Predictive analysis and modelling football results using machine learning approach for English Premier League. International Journal of Forecasting, [online] 35(2), pp.741–755. doi:10.1016/j.ijforecast.2018.01.003.

[9] Prasetio, D. and Harlili, Dra. (2016). Predicting football match results with logistic regression. 2016 International Conference On Advanced Informatics: Con-

cepts, Theory And Application (ICAICTA). [online] doi:10.1109/icaicta.2016.7803111.

[10] Rudin, P. (2016). Football result prediction using simple classification algorithms, a comparison between k-Nearest Neighbor and Linear Regression. [online] DIVA. Available at: http://kth.diva-portal.org/smash/record.jsf?pid=diva2%3A930960&dswid=-9068

[11] Kumar, S. (2021). 7 Hyperparameter Optimization Techniques Every Data Scientist Should Know. [online] Medium. Available at: https://towardsdatascience.com/7-hyperparameter-optimization-techniques-every-data-scientist-should-know-12cdebe713da

[12] Lee, W.-M. (2021). Tuning the Hyperparameters of your Machine Learning Model using GridSearchCV. [online] Medium. Available at: https://towardsdatascience.com/tuning-the-hyperparameters-of-your-machine-learning-model-using-gridsearchcv-7fc2bb76ff27

[13] Transfermarkt.co.uk. (2022). Premier League - Transfers 22/23. [online] Available at: https://www.transfermarkt.co.uk/premier-league/transfers/wettbewerb/GB1

[14] Soccerbase.com. (2023). Football Betting | Place Your Football Bet Today | Soccer Base. [online] Available at: https://www.soccerbase.com/

[15] Leaguemanagers.com. (2022). League Managers Association - HOME. [online] Available at: https://leaguemanagers.com/

[16] Premierleague.com. (2022). Premier League Table, Form Guide & Season Archives. [online] Available at: https://www.premierleague.com/tables

[17] Malek et al. (2019). Bioimage Informatics. Encyclopedia of Bioinformatics and Computational Biology, [online] pp.993–1010. doi:10.1016/b978-0-12-809633-8.20308-7.

[18] Zach (2022). How to Interpret the Classification Report in sklearn (With Example) - Statology. [online] Statology. Available at: https://www.statology.org/sklearn-classification-report/

[19] Blow, T. (2023). Liverpool predicted line-up vs Brentford as Cody Gakpo sweating on debut. [online] mirror. Available at: https://www.mirror.co.uk/sport/football/news/liverpool-brentford-team-news-gakpo-28853164

[20] Rodrigues, F. and Pinto, Â. (2022). Prediction of football match results with Machine Learning. Procedia Computer Science, [online] 204, pp.463–470. doi:10.1016/j.procs.2022.08.057.