English Premier League Game Predictions

CA683 Data Analytics and Data Mining

Group – 15

Anup Bolli 18210820
Arun Ramakrishna 18210416
Ashish Vijay Lakamale 18210227
Sabyasachi Modak 18210389
Saiyyed Faizan Ahmed 18210956
Somsubhra Mukherjee 18210308

Table of Content

1. Introduction

There are a number of countries in the world which have their own football/soccer leagues. The duration of these tournaments covers an entire season which is nearly a year. Usually each league play with 20 odd teams selected from the tier divisions and they compete against each other. They play both at home and away destinations. The matches can have the following result – Home Win, Away Win, Draw.

English Premier League (EPL) is statistically and historically the most followed and widely speculated football league. It has fan following across the world from the States to the Eastern Asian counties with fans clubs of multiple clubs across the world. The EPL is considered to be the most competitive league around the world. The competition is so high that in the last 10 years there has not been any single winning team which has been able to defend its title in consecutive seasons.

Given such a huge fan base and craze around the tournament, the betting agencies and masters of the game always analyses each game. The analysis of the game statistics is done by the betting companies for profits, by the teams themselves to have better team formation and player ratings and also for the transfer windows. Most pundits would follow the statistics and discuss them on highly followed sport shows.

1.1 Motivation

The introduction of statistics into sports has been a long followed schema. The introduction of analysis and predictive capacities into the formula is a new addition. The motivation for this particular genre of the game comes with the focus on team building. Most teams and managers would love to know their opposition's strength and weaknesses beforehand; also they would want to know the best possible team lineup to play on a given day. It certainly helps them to target potential players in the transfer market. Similarly the game experts and betting companies also spends a huge amount of money on the statistics to make things interesting and profitable for the fans. At the end and perhaps the biggest motivation is for the fans of the game who would just love to know the stats fast and easy before the game.

1.2 Question

The primary questions are:

- Can we can predict the outcome of EPL matches?
- If we can do the above, how accurate our results are?

1.3 Data Set

The data set that we have taken over here for our assignment is one of the most trusted sources of football data for the EPL (http://www.football-data.co.uk/). The dataset contains the statistics of the premier league matches from Season 2000-2001 to 2017-2018 in CSV format. It has around 6500 x 45 (R x C) statistical data of 17 seasons. It is a comprehensive data set with different attributes of the game included throughout all these years.

C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
HomeTear	AwayTean	FTHG	FTAG	FTR	HTHG	HTAG	HTR	Attendanc	Referee	HS	AS	HST	AST	HHW
Charlton	Man City	4	0	H	2	0	H	20043	Rob Harris	17	8	14	4	2
Chelsea	West Ham	4	2	H	1	0	H	34914	Graham Ba	17	12	10	5	1
Coventry	Middlesbr	1	3	A	1	1	D	20624	Barry Knigl	6	16	3	9	0
Derby	Southamp	2	2	D	1	2	A	27223	Andy D'Urs	6	13	4	6	0
Leeds	Everton	2	0	H	2	0	H	40010	Dermot Ga	17	12	8	6	0
Leicester	Aston Villa	0	0	D	0	0	D	21455	Mike Riley	5	5	4	3	0
Liverpool	Bradford	1	0	H	0	0	D	44183	Paul Durkir	16	3	10	2	0

The dataset is divided into two kinds of data:

- Numerical data – (FTHG, FTAG, Attendance, HS, AS, HST, AST, AY, AR, etc.)
- Categorical data – (HomeTeam, AwayTeam, Referee, HTR, FTR)

Data Dictionary – The full form of the entire dataset abbreviations can be found at - http://football-data.co.uk/notes.txt. It contains the various column description that have been used in the later stages for the project.

Target Column – FTR (Full Time Result – Home Win, Away Win, Draw) i.e. the column gives us the final outcome of the match itself.

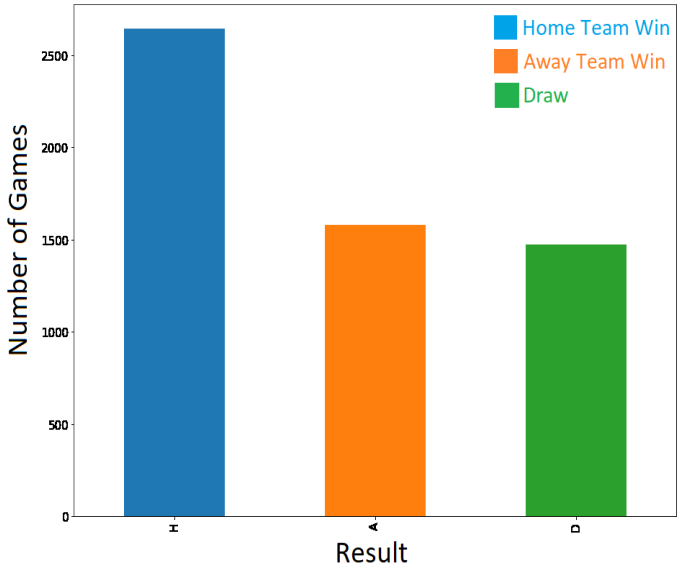


Fig – Target Column Statistics

2. Data Pre-Processing

2.1 Handling Missing Values

2.1.1 Dropping Records

We found that for around 20 – 25 games the FTR and some other key attributes were missing from the original datasets combined. As the number was very small, we decided to drop those records from our data frame.

2.1.2 Dropping Columns:

We did not want external factors affecting the predictions, like the betting odds. The betting data from several organizations have also been included in the datasets. Most of the data in those columns were incomplete. Hence we some columns like:

- B365H – Bet 365 Home Win Odds
- B365A – Bet 365 Away Win Odds
- BSH = Blue Square home win odds
- BSD = Blue Square draw odds
- B365>2.5 = Bet365 over 2.5 goals
- B365<2.5 = Bet365 under 2.5 goals

2.1.2 Feature Engineering

The data set has several features (more than 26) which has been recorded for all the previous seasons. In order to better understand and modify our parameters further we have included some features based on some research into football game statistics.

- **Aggregated Features –** How Strong is a team in attacking and defense?

For this particular category we separated the table into groups of Home Team and Away Team. Then we took the average number of goals scored by a team per home game (Total Home game - 19) divided by the Average number of goals at home.

For example,

Attacking strength at home (HAS) = (Goals scored at home / 19) / Average Number of goals at home

In the above way we came across the following four aggregated features:

- HAS – Home Attacking Strength
- HDS – Home Defensive Strength
- AAS – Away Attacking Strength
- ADS – Away Defensive Strength

- **Dynamically generated Features –** How good or bad is a team's form coming in a match?

In this category we wanted to evaluate the team's progress during individual seasons and as the matches progressed the value of the attributes also changing depending on the performance of the teams in the past games.

For example,
DiffPts – Difference in Aggregated points gives the difference between a home team and away team depending upon the past win (+3), draw (+1) or loss (0).
In the above way we generated the following new features to be included:

- ATP – Away team point
- HTP - Home team goal point
- ATGD – Away team goal difference
- HTGD – Home team goal difference
- DiffPts – Difference in Aggregated Points
- DiffFromPts – Difference in Form

2.1.3 Feature Elimination

- After creating the new features we integrated the new features with the existing features to create a feature table.
- The feature table has all numerical data. The FTR column was converted from alphabetical to numerical form (H : +1, A : -1, D : 0).
- **Recursive Feature Elimination with Cross Validation** was applied to the new feature table in order to select the best features among all of them.
- In order to further fine tune our features we have applied two other techniques for the table. **Extra Trees Classifier** and **Random Forest Classifier** has been applied. After taking in consideration all the three methods the following features have been identified as the best attributes for the models.
    - HAS – Home Attacking Strength
    - HDS – Home Defensive Strength
    - AAS – Away Attacking Strength
    - ADS – Away Defensive Strength
    - DiffPts – Difference in Aggregated Points
    - DiffFromPts – Difference in Form
    - AY – Away Red
    - HR – Home Red
    - AR – Away Red

- Correlation Matrix for the nine features obtained can be seen as below. The correlation is very low among most of the features and hence it is perfect for the models to be used.

	HAS	HDS	AAS	ADS	DiffPts	DiffFormPts	AY	HR	AR
HAS	1	-0.82	-0.045	0.041	0.46	0.36	0.045	-0.036	0.03
HDS	-0.82	1	0.041	-0.039	-0.42	-0.33	-0.045	0.02	-0.024
AAS	-0.045	0.041	1	-0.81	-0.45	-0.34	0.00084	0.044	0.017
ADS	0.041	-0.039	-0.81	1	0.44	0.34	0.013	-0.037	-0.015
DiffPts	0.46	-0.42	-0.45	0.44	1	0.66	0.035	-0.047	0.0092
DiffFormPts	0.36	-0.33	-0.34	0.34	0.66	1	0.039	-0.046	0.022
AY	0.045	-0.045	0.00084	0.013	0.035	0.039	1	0.084	0.12
HR	-0.036	0.02	0.044	-0.037	-0.047	-0.046	0.084	1	0.081
AR	0.03	-0.024	0.017	-0.015	0.0092	0.022	0.12	0.081	1

Fig – Feature Correlation Matrix

3. Modeling

We have included the following models or our predictive analysis in our project using the features selected from the feature elimination process.

3.1 XG Boosting

Python along with 'Jupyter Notebook' was chosen as a modelling tool for implementing statistical Models and analyzing data. All the analysis done above is implemented with ease with the help of keras and respective functional API. The two reasons to use XGBoost are also the aim of this project:

1. Execution Speed
2. Model Performance

Model Breakdown

1. Learning Rate (learning rate = 0.1): XGBoost has a very useful function called as "cv" which performs cross-validation at each boosting iteration and thus returns the optimum number of trees required. Determining the optimum number of trees for this learning rate.
2. Max-Depth (max depth = 5): It's the maximum value to which a tree will grow. Controls the overfitting as higher depth will allow model to learn relations very specific to a sample. We tuned it with the help of Cross Validation.

3. Estimators (n_estimators = 10): Scikit-learn was used to perform a grid search of the n_estimators model parameter, evaluating a series of values from 50 to 350 with a step size of 50 (50, 150, 200, 250, 300, 350).
4. Colsample Bytree (colsample_bytree = 0.3): It denotes the fraction of columns to be randomly samples for each tree. Typical values: 0.5-1.
5. Objective (objective='reg:linear'): This defines the loss function to be minimized. Multiclass classification using the linear objective, returns predicted class not probabilities. Also set an additional num_class (number of classes) parameter defining the number of unique classes.

Feature Statistics

Trained XGBosst model is used to extract the feature importance in the model. Below is the plot of the 8 features with respective importance.

Results are calculated using F 1 score, in statistical analysis of binary classification, the F 1 score (also F-score or F-measure) is a measure of a test's accuracy. It considers both the precision p and the recall r of the test to compute the score.

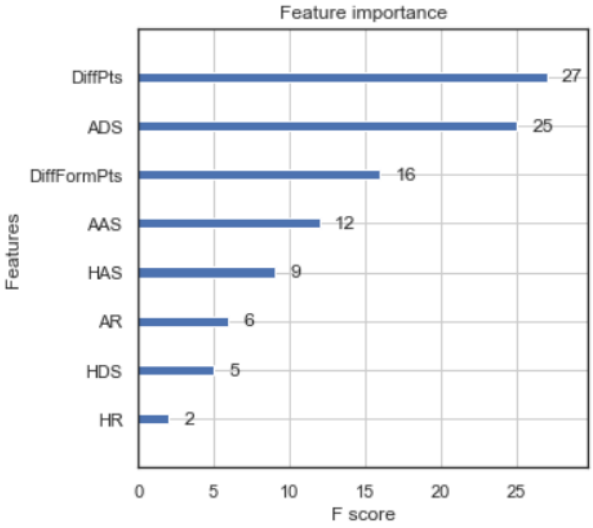


Fig – Feature Statistics

3.2 Support Vector Machine

Predicting a premier league is a supervised multi-classification problem, which makes Support Vector Machine (SVM) a suitable model to be tried on this problem. Initially we have built a basic SVM model based on the features selected in feature engineering process, with dataset split into 70% test and 30 % train and a basic kernel (Kernel = Linear, Gamma = Auto, Regularization Parameter C = 100). This model gave an accuracy of 51%, f1-score of 0.43, with 85 % correctly predicted Home win, 2% correctly predicted Draws and 40% correctly predicted Away wins.

Multicollinearity between independent features is considered to have instable effect on final estimate. The instability increases the variance of estimate, which means a small change in a feature causes large changes in estimate. As we had few features that were highly correlated such as Diffpts and DifformPts, we have eliminated these features from our lists used to build final SVM model.

To get the best parameters suited for SVM model that will gives better predictions we have used GridSearchCV (Grid Search Cross Validation) function. GridSearchCV function has two parts – Cross Validation and Parameter tuning. Cross Validation trains the model with one set of data and test with a different set of data. Hyper Parameter tuning is the process of selecting the values that maximizes the accuracy of the model. GridSearchCV was given with following parameters – Kernels: Linear and Radial Basis function, Regularization parameter: 1, 10, 100, 1000 and Gamma values ranging from 0.001 to 0.0001. Best parameters selected by GridSearchCV were: Kernel = RBF, Gamma = 0.001, Regularization parameter C = 1000.

	Accuracy
Overall	56 %
Target = -1 (Away Win)	54 %
Target = 0 (Draw)	3 %
Target = 1 (Home Win)	87 %

3.3 Logistic Regression

A logistic regression model has been included as our third model of predictions. As the correlation between the selected features is very less we have included all the 9 features into the model. As this is a classification problem with multinomial result strategy the model is applied to the target. We have used 'Step-AIC' to fine tune the dimensions available to us. We have a training set and test set split of 70% and 30% on our available final dataset. A threshold is chosen in such a way so as to give the least FTR (False Positive Rate) of 3%, i.e. a specificity of 70%. The results obtained after the final model testing can be viewed in table shown.

	Accuracy
Overall	55 %
Target = -1 (Away Win)	53 %
Target = 0 (Draw)	1 %
Target = 1 (Home Win)	87 %

4. Evaluation

After running the models we obtained the following data from the three models. The confusion Matrix for the models Logistic Regression and SVM are summarized as below.

Confusion Matrix:

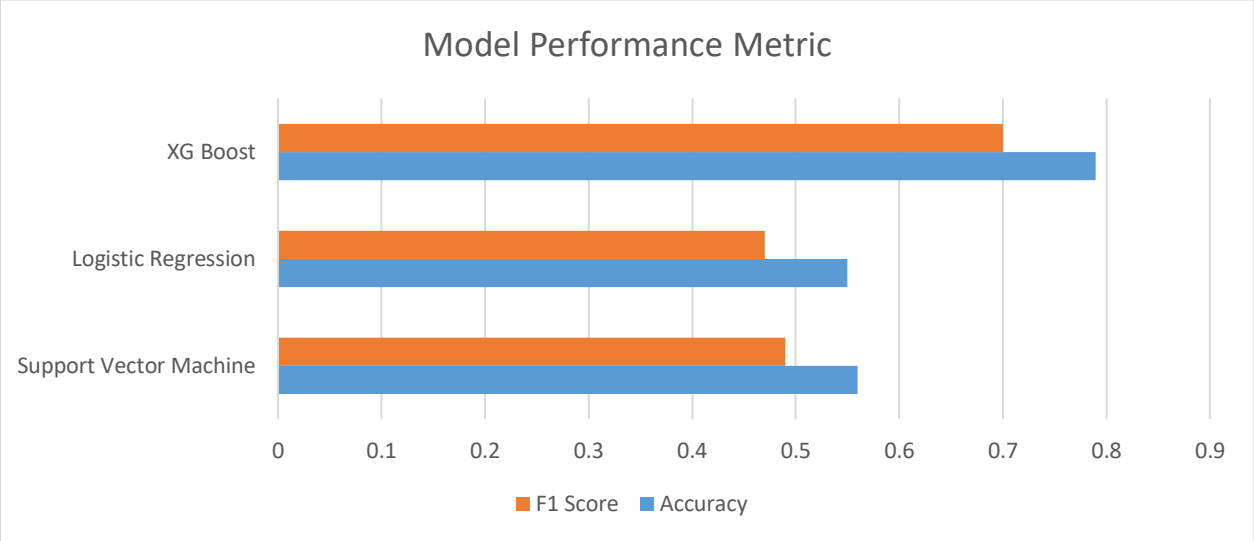
	Prediction			
Truth		-1	0	1
	-1	315	10	254
	0	152	14	358
	1	112	8	829

Support Vector Machine

	Prediction			
Truth		-1	0	1
	-1	252	3	217
	0	123	6	312
	1	101	4	685

Logistic Regression

F – Score and Accuracy Comparison: In order to better understand the difference between the prediction capacities of all the models we compared the F- Scores of all the models vs their accuracies.



5. Conclusion & Scope

In our analysis we find that the XGBoost performs the best among all the models implemented. The better accuracy is certainly worth noting although the other two models have good performances as per the confusion matrix. Other deep learning models could not be applied in this project as it would require a lot more data set to give sustainable outputs. Other than the

game statistics, player position statistics, weather data and certain other data sets could be clubbed with the project as well in future. The result of adding external factors into game statistics as of now has not been carried out here.

6. References

1. Bunker, Rory & Thabtah, Fadi. (2017). A Machine Learning Framework for Sport Result Prediction.
2. Stylianos Kampakis, University College London. (2015). Using Machine Learning to Predict the Outcome of English County twenty over Cricket Matches.
3. Siraj Raval, (2017). Predicting the Winning Team Using Machine Learning
4. Albina Yezus, (2014). Predicting outcome of soccer matches using machine learning.
5. Mariam Sulakian - https://mariamsulakian.com/2018/02/01/machine-learning-predicting-the-2018-epl-matches/
6. Tuan Doan Nguyen, (2018) - The Beautiful Game: Predicting the Premier League with a random model - https://towardsdatascience.com/o-jogo-bonito-predicting-the-premier-league-with-a-random-model-1b02fa3a7e5a