PH125.9X Capstone Project - Predicting The Results Of English Premier League Soccer Matches

Mike Woodward

2021-02-13

Contents

1 Executive summary

In this project, I aim to forecast English Premier League (EPL) soccer match results. To do this, I studied several potential machine learning modeling features, developed a large number of models and examined their properties, and I discuss how the project could be extended for greater accuracy.

The features I selected for modeling were: team transfer value, foreign player count, team mean player age, season week, mean prior goal difference, mean prior points difference, and mean prior red and yellow card difference. In the text, I explain and define these features.

The nine models I developer were: naive means, generalized linear, generalized linear net, support vector machines (radial version), k-nearest neighbors, neural nets, XGB linear, XGB tree, and random forest.

The best models were only a 8% improvement over naive means for RMSE, and I explain why this is so in the conclusion. For home team predictions, the best model was XGB Tree, for away team predictions, the best model was a neural net. These results suggest the underlying system may be non-linear.

2 Introduction

2.1 Project goals

The aim of this project was to predict English Premier League soccer match results with an accuracy better than using a naive mean results model alone. The prediction target in this project was the number of goals scored in each match by the home team and the away team.

2.2 Soccer and the EPL: background and modeling features

To properly model soccer matches, we need to understand the properties of soccer and the EPL in particular.

The origins of the EPL have been extensively described elsewhere (see for example [Robinson]), so I won't repeat them here, however, I will describe some features relevant for modeling. The league was founded by a group of clubs who wanted a larger share of TV rights money and bigger TV deals [Butler], with the first matches in the 1992-1993 season. This ruthless commercial focus has continued over the last twenty-five years, with the league becoming one of the most commercially successful sports organizations in the world [Robinson]. A key part of the league's success has been its ability to attract overseas talent, in fact, the EPL is known for the very high number of foreign players. This suggests two areas for investigation:

- Do financially larger clubs score more?
- Does having a higher number of foreign-born players lead to more goals?

In common with nearly all soccer leagues worldwide, the EPL operates as the top-tier league, with a system of promotion and relegation from the league below it (now called the Championship League). Each season, several clubs are promoted and relegated, with the rules for promotion and relegation changing over time (e.g at the end of the 1994-1995 season, the league dropped from 22 teams to 20 via relegation). Finishing top or bottom of the league has substantial financial and reputational implications. The team that finishes top of the league are the league champions, and the top few teams qualify for European competition. European competition is very lucrative, both from match attendance and from TV rights. The bottom teams may be relegated, which means a very large drop in revenue and may cause players to leave the club. Therefore towards the end of the season, teams near the top or bottom of the league may have stronger motivations to win. This suggests another modeling feature:

- Are there more wins and goals as the season progresses?

Home field advantage has been extensively discussed in the literature (e.g. [Pollard], [Leard], [Thomas]). If there were no home team advantage, we would expect the number of home wins to be about the same as the number of away wins.

- Is there evidence of a home team advantage?

Each of the teams in a match comes to the match with a track record of previous games won, lost or drawn. We might expect teams higher up the EPL table to beat teams lower down, and we might expect teams with a good track record of goal scoring to beat teams with a poor record.

- Is there evidence that a team's prior performance influences the outcome of its current match?

On-field fair-play has been an important issue for the EPL, and for English soccer as a whole. Players receive a yellow card as a warning, with a red card for dangerous play or serious rule-breaking. A player who receives a red card, or two yellow cards in the same match, is sent off (can't play in the rest of the match) and can't

be substituted. His team has to play with one less player, a substantial disadvantage. However, yellow and red cards might also be associated with the kind of risk-taking that wins matches.

- Do the number of red cards and yellow cards affect a team's goal-scoring ability?

It's well-known that soccer is a low scoring game with a high element of randomness [Bunker], which causes issues for modeling as we'll see. Currently, the EPL has 20 teams who play each other twice per season (once at home and once away). This gives a total of 380 games per season and there have been 29 seasons, which is not a large data set to work with and may limit the accuracy of machine learning.

Like all sports leagues, the EPL has been affected by COVID. The 2019-2020 season was extended for several months and the 2020-2021 season is being played without fans in stadiums. This causes issues for modeling as we'll see.

- Is the 2020-2021 season different from other seasons due to COVID?

2.3 Prior work

Mathematicians have studied gambling for hundreds of years; in fact, the whole discipline of probability theory was largely created to understand gambling [Epstein]. Unlike other areas of math, those who are successful at analyzing gambling may chose not to publish, instead becoming wealthy themselves [Mezrich, Meloche]! Despite the the disincentive to publish, researchers have released a large number of studies analyzing soccer matches.

- Home field advantage has been extensively studied (e.g. [Leard], [Pollard], [Thomas], [Vergina]) and has been found to exist in many sports, including the EPL [Allen].
- Dawson *et al* [Dawson] studied consistency of red and yellow cards. They found that referees penalized away teams more (which may contribute to the home team effect). Oberstone [Oberstone] found a weak link between yellow cards and team performance.
- Several researchers ([Plumley], [Barros]) have examined the link between financial performance and on-field performance, but financial performance has been measured using company financial reports (e.g. incomes statements), not the transfer value of the team.
- Surprisingly little has been written about seasonal effects. Allen *et al* [Allen] found a relationship between the size of the home advantage effect and final league position.
- The effect of COVID has been reported in the press and in academic studies [McCarrick], specifically, the disappearance of home field advantage.

2.4 Key steps in this project

I broke the project into four key steps.

1. Downloading data. This is where I download three data sets from two websites and ingest a file of team abbreviations I created by hand. To keep things simple, I perform minimal cleaning. This step is described in the 'Data download' section below.
2. Cleaning data. This is where I cleaned the data and merged data sources to create one data frame with the fields I need. I describe this step in more detail in the 'Data cleaning' section.
3. Exploratory data analysis. I create different ways to analyze the data and visualize the results. The results of this work I describe in the 'Methods and analysis' section.
4. Modeling. This is where I ran all machine learning models. The work is described in the 'Methods and analysis' and 'Results' section.

The code for this project (EPL-Project.R) follows this breakdown exactly.

2.5 Running the software

The entire code is in the file EPL-Project.R. This file can be run from any folder and it will create all folders (using relative paths) and data it needs. It will download data, clean it, analyze it, and model it.

The key code is at the end of the file (shown here). Each of these functions performs one of the key steps as described above. In some cases, the functions can be run in isolation (for example, by commenting out the other functions). All told, the file takes close to 24 hours on a high-spec multi-core machine.

```
# Housekeeping. Loads the relevant libraries - it must *always* be run.
housekeeping()

# Download downloads the files from the internet. It can take 25 minutes to run.
# Only call this function if you haven't already downloaded the data.
download()

# Data cleaning. Can be run without running download if the data has already
# been downloaded. Writes results to an rda file.
clean()

# Data analysis. Can be run on its own, provided download and clean have been
# run previously. Reads in cleaned data from an rda file. Outputs charts
# and variables for use in the Rmd report.
dataanalysis()

# Model. Can be run on its own if clean has been run previously. Reads in
# cleaned data. Note: does not depend on dataanalysis having been run.
model()
```

Note the software will install all necessary packages if they haven't already been installed. Here's a code snippet to show loading just a few libraries.

```
  # Snippet starts...
  if(!require(doParallel)) install.packages(
    "doParallel", repos = "http://cran.us.r-project.org")
  if(!require(elasticnet)) install.packages(
    "elasticnet", repos = "http://cran.us.r-project.org")
  if(!require(xgboost)) install.packages(
    "xgboost", repos = "http://cran.us.r-project.org")

  library(tidyverse)
  library(caret)
  library(data.table)
  # ...Snippet ends
```

3 Methods and analysis

3.1 Data download

Compared to other sports, most notably baseball, detailed data on soccer matches has only recently become available [Rein]. Very detailed data is now available, but at a very high price from dedicated sports data vendors; the price put this data beyond my reach. For this project, I used two free data sources:

- Football-data. This site has match results in csv form going back to the start of the EPL and is frequently updated. The match data varies by season (more data is available for later seasons) and includes match score and red and yellow cards.

- TransferMarkt. This site has data on the notional transfer values of teams, the number of foreign players, squad size, and mean team age. The data is updated twice a month but is only available for the 2011-2012 season onwards.
  - The transfer value for a club is the transfer value of its players. For example, if a team buys a new player for £200mn, then the transfer value of the team goes up by £200mn. TransferMarkt update this value twice a month.
  - The squad size is the number of players a team has available. A soccer team fields 11 players in a match, but substitutions are allowed and of course players get sick or may have to miss games. The EPL now sets an upper limit on squad size. TransferMarkt update squad size at the start of the season.
  - 'Foreign' players here means any player born outside of England. TransferMarkt update this number at the start of the season.
  - Mean age is the mean age of the squad. TransferMarkt update this number at the start of the season.

I scrapped the TransferMarkt data from its website using the rvest r package. However, due the way TransferMarkt presents HTML tables, I had to use some complex processing to extract data.

3.2 Data cleaning

English soccer teams are often known by several names, for example, Manchester United is also known as:

- Man Utd
- Man United
- Manchester United
- Manchester United FC
- MUFC

and various derivatives and combinations. To join data from different sources, I needed a consistent naming convention. I used the EPL codes for teams and mapped name variations to the code, for example, I mapped 'Man United' and 'Manchester United' to the code MUN. I created this file by hand and added it to the Github page page for this project. I download this file from Github in the download function of my code.

TransferMarkt calculates team values and team sizes twice a month, but matches are held many times a month on different days. To map team value (and team size) to matches, I used r's fill function to interpolate team values for the day of the match.

Using a simple join on team and season, I used the foreign player count at the start of the season for all matches in the season.

The soccer season starts in August and typically runs until May of the next year. I calculated a season week using date offsets.

The clean function in my r code performs all these functions and a few more (e.g. consistent date formats). It merges all the data into one data frame, called match_results, which it saves to an rda file that both the datanalysis and model functions use.

3.3 Data exploration and visualization

3.3.1 Distribution of scores

Are certain scores more likely than others? Using the entire data set, I calculated how many times each score occurred. The heatmap below shows the results.

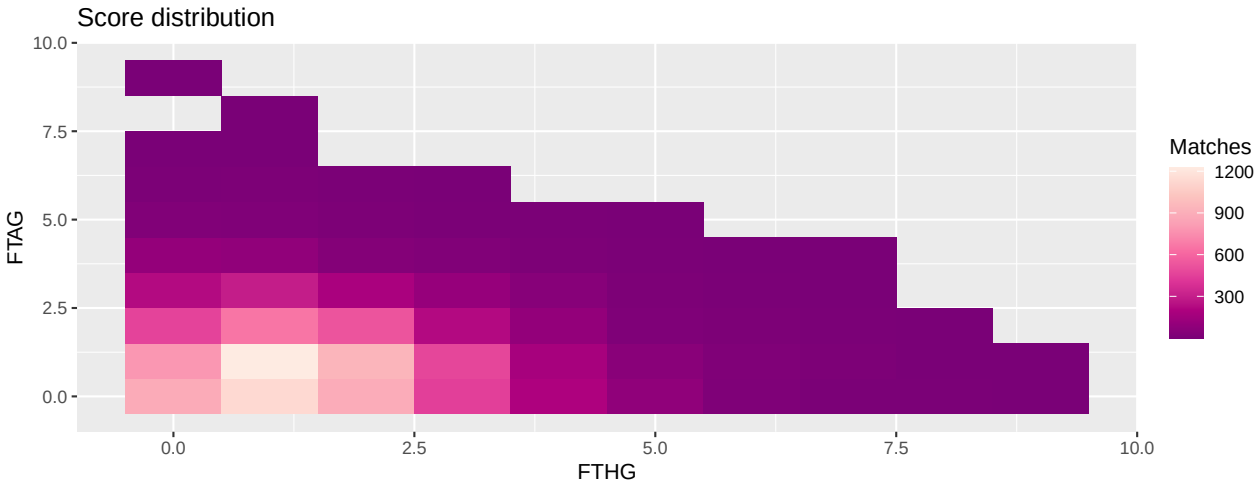


Figure 1: Scores heatmap. FTAG=Full Time Away Goals, FTHG=Full Time Home Goals

The most likely result is a 1-1 draw, followed by 1-0. Note how the probability falls off rapidly and how some scores don't occur, for example, in almost thirty years worth of data, a score of 7-7 has never happened.

3.3.2 Home field advantage

If there were no home field advantage, we would expect the number of home and away wins to be roughly equal. More formally, we might expect:

$$\frac{count\ of\ home\ wins}{count\ of\ home\ wins + count\ of\ away\ wins} \approx 0.5$$

and the test for equivalence would be a z-test or a t-test as appropriate.

In Figure 2a, I show this proportion by season with the error bars representing the 95% confidence interval.

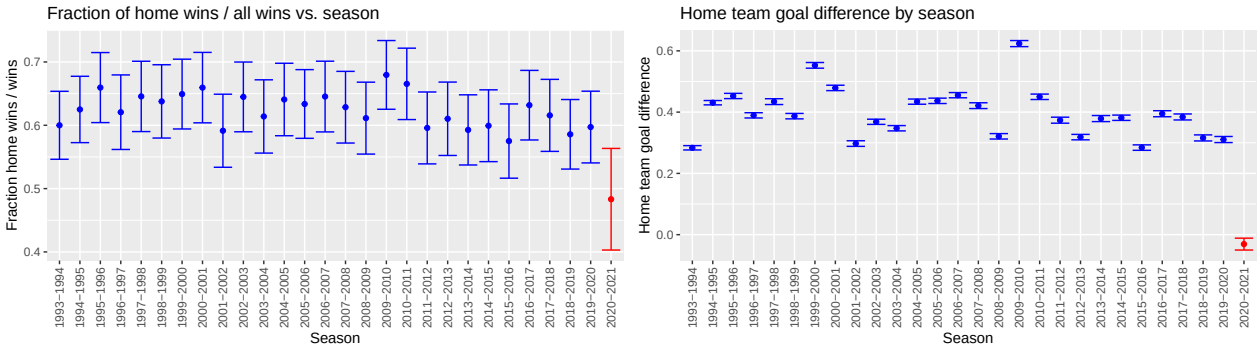
Interestingly, the 2020-2021 results suggest a mechanism for home field advantage. Due to COVID-19, this season is running entirely without spectators; teams are playing in empty stadiums. For 2020-2021, the fraction of home wins, 0.483, is close to 0.5 (the larger error bars are because the season is only part way through at the time of analysis). It seems like then that the home field advantage may be due to the presence of home spectators.

The home effect is also apparent if we look at goal difference. Goal difference is the difference between the number of goals scored by the home team and the away team. In Figure 2b, I've plotted the mean goal difference (over all games in the season) against the season. Clearly, home team advantage is worth about 0.35 goals, except for 2020-2021 (more evidence of a COVID effect).

3.3.3 Team value advantage

EPL teams are well-known for spending very large sums of money buying players. Is having a more valuable team than your opponents an advantage? As a reminder, team value in this project is the notional transfer value of the team as reported by TransferMarkt.

For each match in each season, I calculated a value difference and a goal difference (value difference is the difference in transfer value of the two teams and arranged so it's $most\ expensive\ team\ -\ cheaper\ team$, the goal difference is $most\ expensive\ team's\ goals - cheapest\ team's\ goals$). Figure 3 shows the result, each point is a match (with an alpha of 0.3 to show where matches overlap), the (red) straight line is a linear fit, with the light red zone a 95% confidence interval. The chart clearly shows the aggregate effect of a value difference between teams, with a £700mn difference worth about 2 goals for the more valuable team.



(a) Home win proportion

(b) Goal difference

Figure 2: Home field advantage

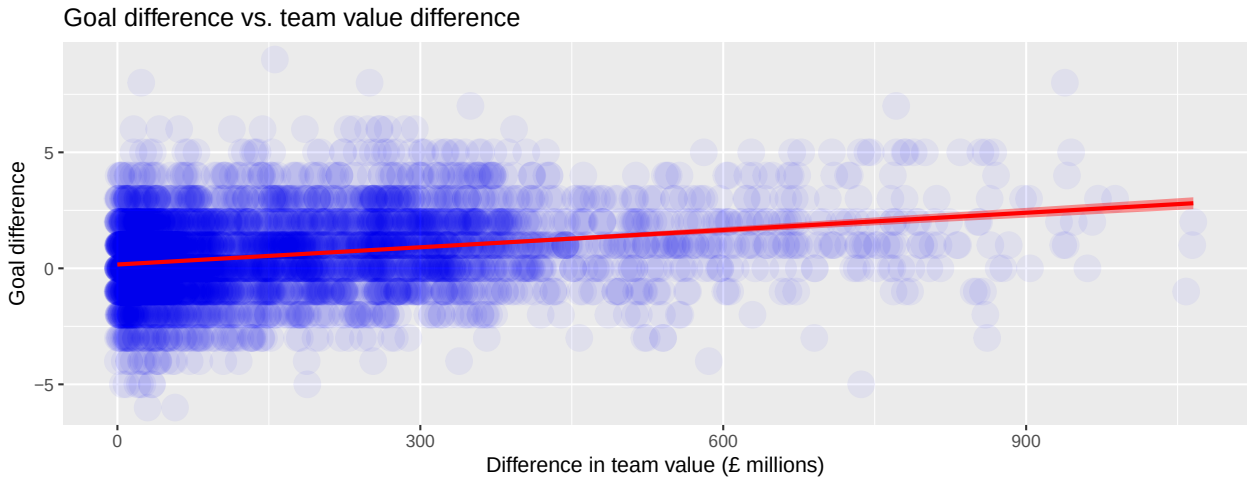


Figure 3: Goal difference vs. value difference

3.3.4 Foreign players

Perhaps as a consequence of their large expenditure on players, EPL teams are famous for having large numbers of foreign players, TransferMarkt notes that for the 2020-2021 season, about *63%* of players in the league are foreign born. The obvious question is, does having more foreign born players give a team an advantage?

For each match in each season, I plotted goal difference vs. the difference in foreign player count (data arranged so the foreign player difference is always positive). Figure 4 shows there's a small effect.

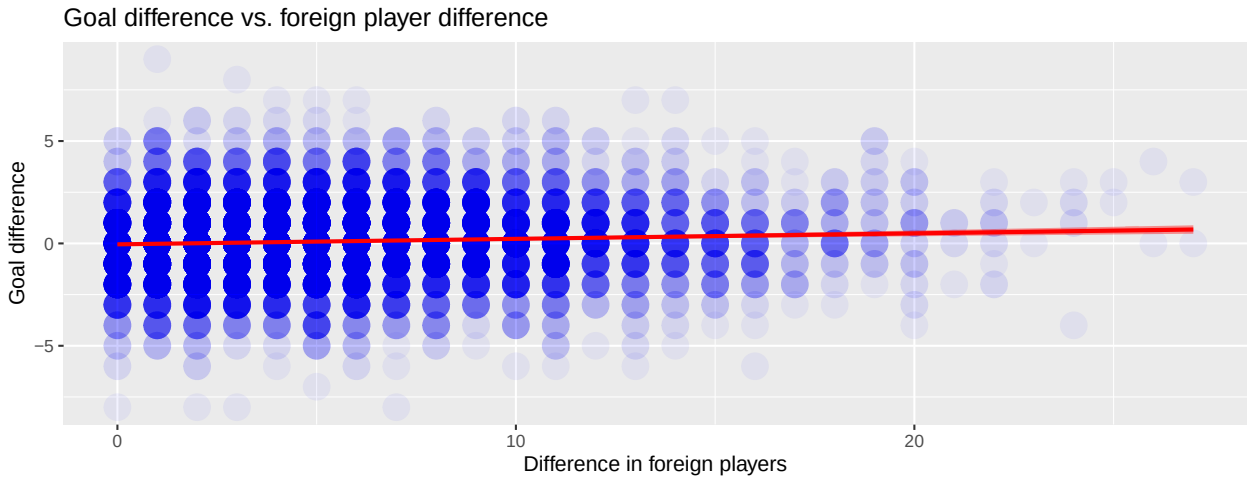


Figure 4: Goal difference vs. foreign player difference

3.3.5 Mean age

It may be true that younger players have more energy and older players have more experience, but what about at the team level? Does the mean age of the team make a difference? For each match, I plotted the goal difference vs the difference in mean age for the teams (Figure 5). There is an effect, worth about a goal for a 5 year difference. To put it simply, older teams appear to be at a disadvantage.

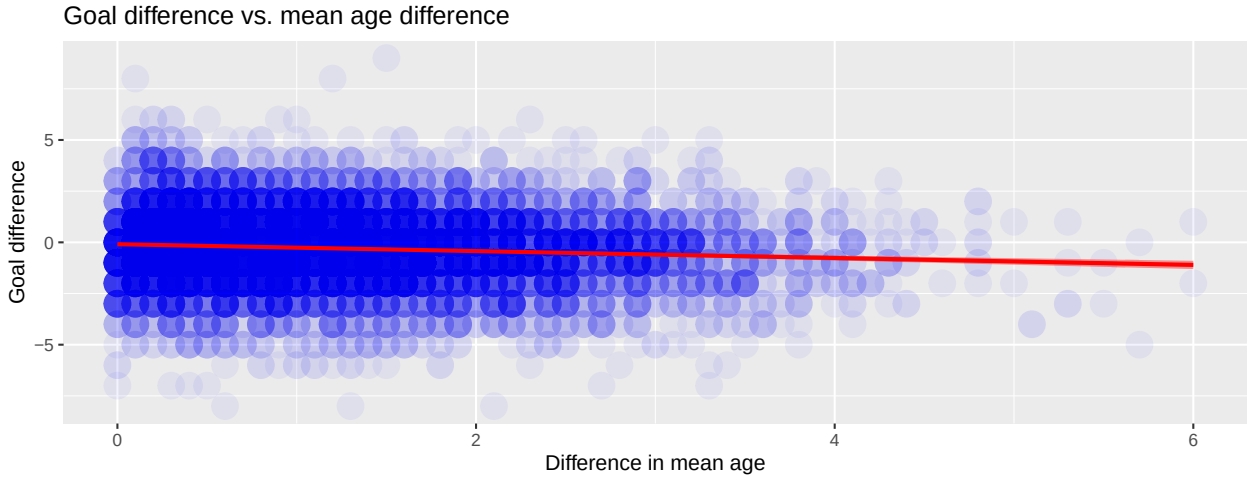


Figure 5: Goal difference vs. difference in mean age

3.3.6 Squad size

There are league rules on the maximum size of squads, which is currently 25 players. For the 2020-2021 season, every team has a squad of 25 players, but that hasn't always been the case and there was much more variability in the past. Is squad size a useful feature? I plotted goal difference against squad size difference for every match in Figure 6.

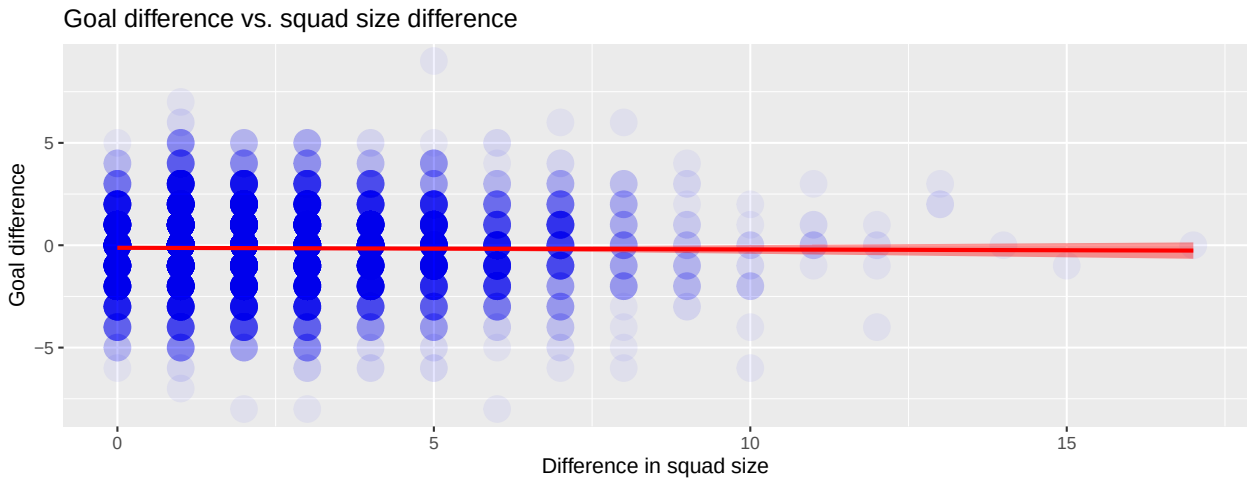


Figure 6: Goal difference vs. difference in squad size

If there is an effect, it's very small. As squad sizes are now limited to 25 and every team has a squad of 25, I've excluded squad size as a modeling feature.

3.3.7 Season effects

As I explained earlier, at the end of the season, some teams may have additional incentives to win. If we look at seasons on a weekly basis, we should see the proportion of matches that ended in a draw go down as the season progresses.

For each season week, I calculated the proportion of matches that were draws and plotted them in Figure 7a. There's a slight trend downward as the season progresses.

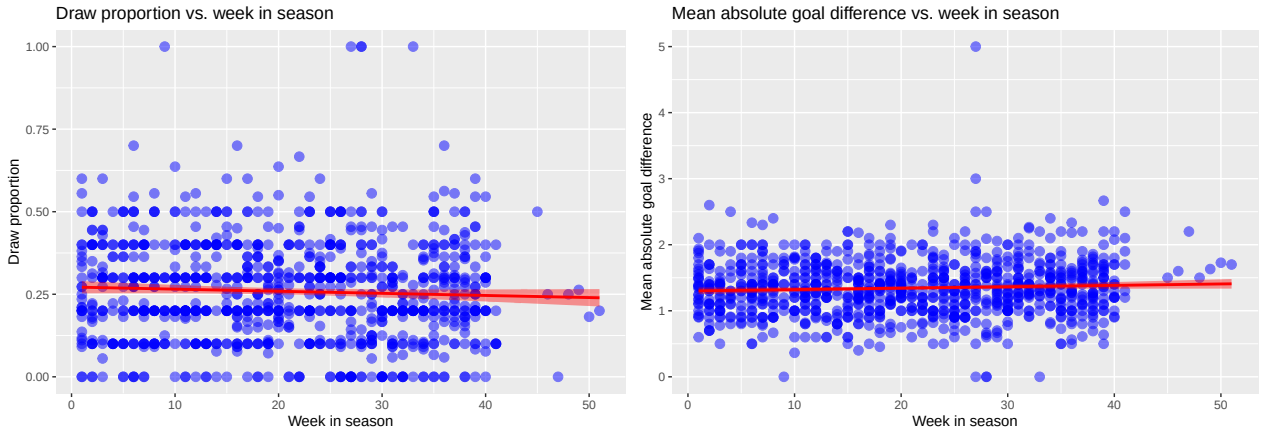
This effect is also apparent in the mean absolute goal difference as the season progresses, meaning games are won by a slightly larger goal margin (Figure 7b). Interestingly, the number of goals scored per match doesn't change very much during the season (not shown in this report), suggesting it's the split of goals between the teams that changes.

(Figures 7a and 7b also show a COVID effect. The 2019-2020 season was paused by COVID and as a result, the season ended much later than usual.)

3.3.8 Discipline, goals, and points

As the season progresses, teams receive more points, goals, and red cards/yellow cards. To remove length of season effects, we need to use the rolling means of these quantities. We want the mean prior to the current game. I'll used the mean number of red cards to illustrate the math. If team A and team B are playing, and this is the ith match of the season for team A and the jth match of the season for team B, and the red cards team A received in match g are $(red\ cards)_g$, then the difference in mean red cards is:

$$\left(\frac{1}{i-1}\sum_{g=1}^{i-1}(red\ cards)_g\right)_{TeamA} - \left(\frac{1}{j-1}\sum_{g=1}^{j-1}(red\ cards)_g\right)_{TeamB}$$



(a) Draw proportion vs. week in season

(b) Mean absolute goal difference vs. week in season

Figure 7: Week in season effects

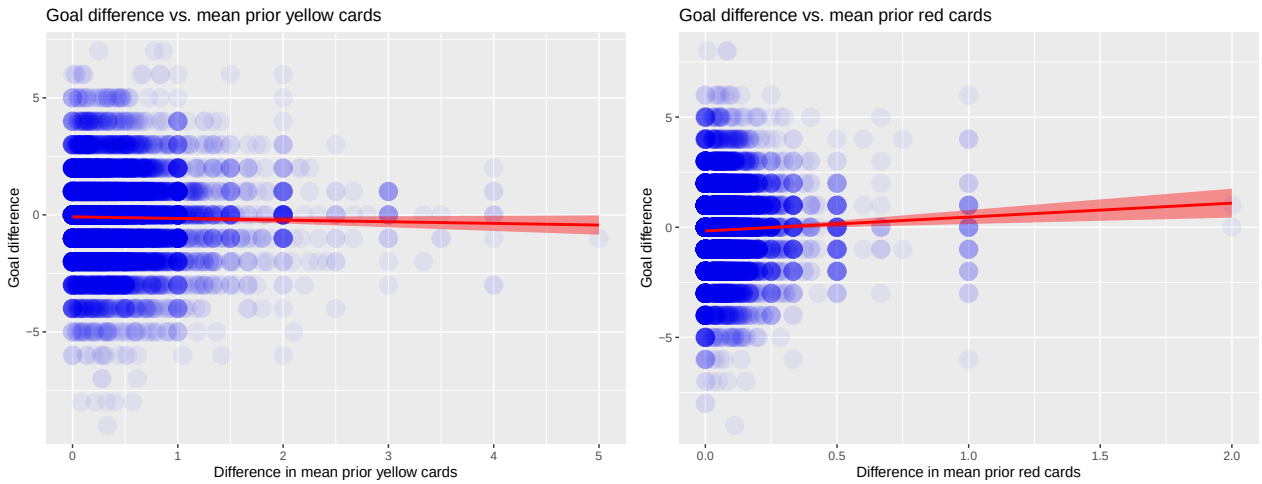
where team A is the team with the higher number of red cards.

3.3.8.1 Red and yellow cards Here's the code to calculate the mean prior red and yellow cards cards.

```
mutate(r=rowid(TeamAbbreviation),
       MPCR=ifelse(r>1, (cumsum(R) - R)/(r-1), 0),
       MPCY=ifelse(r>1, (cumsum(Y) - Y)/(r-1), 0))
```

The code "MPCR=ifelse(r>1, (cumsum(R) - R)/(r-1), 0)" is an exact implementation of $\frac{1}{i-1}\sum_{g=1}^{i-1}(red\ cards)_g$

The plot below shows a small effect for yellow cards (Figure 8a), but it is there.



(a) Goal difference vs. mean prior yellow card difference

(b) Goal difference vs. mean prior red card difference

Figure 8: Discipline effects

The red card effect (Figure 8b) is small, and oddly, it's a positive effect (more red cards, more goals).

3.3.8.2 Points Like most soccer leagues, the EPL awards 3 points for win, 1 for a draw, and none for a loss. Each team will have a number of points *before* a match. Points encode a team's track record, the more successful a team is, the more points it will have.

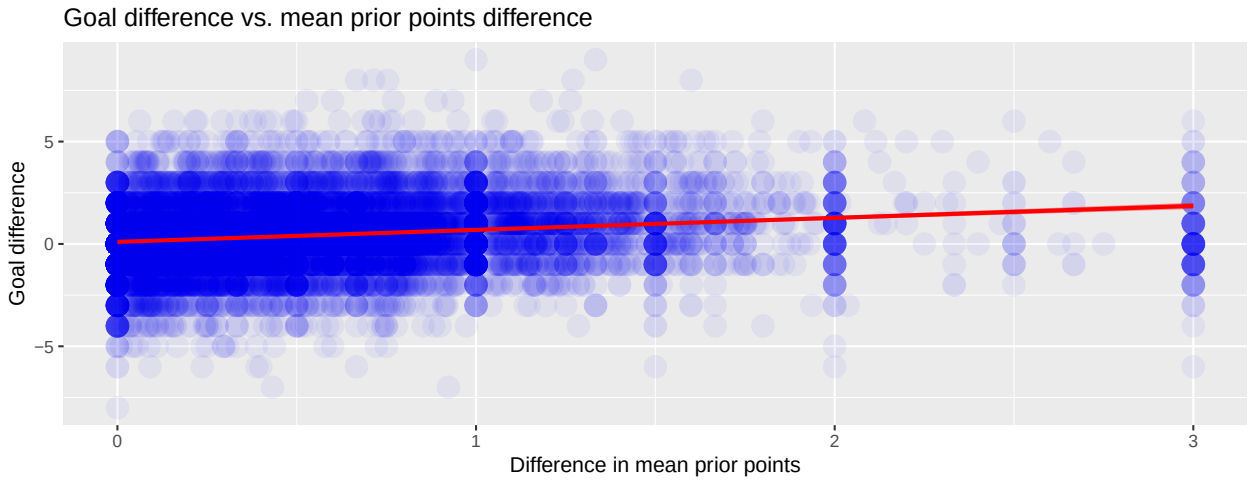


Figure 9: Goal difference vs. mean prior point difference

Figure 9 shows there's a strong effect. To put it simply, teams with a stronger points record tend to score more goals against teams with a weaker points record.

3.3.8.3 Goals If we're forecasting future goals, the best indicator is probably past goals.

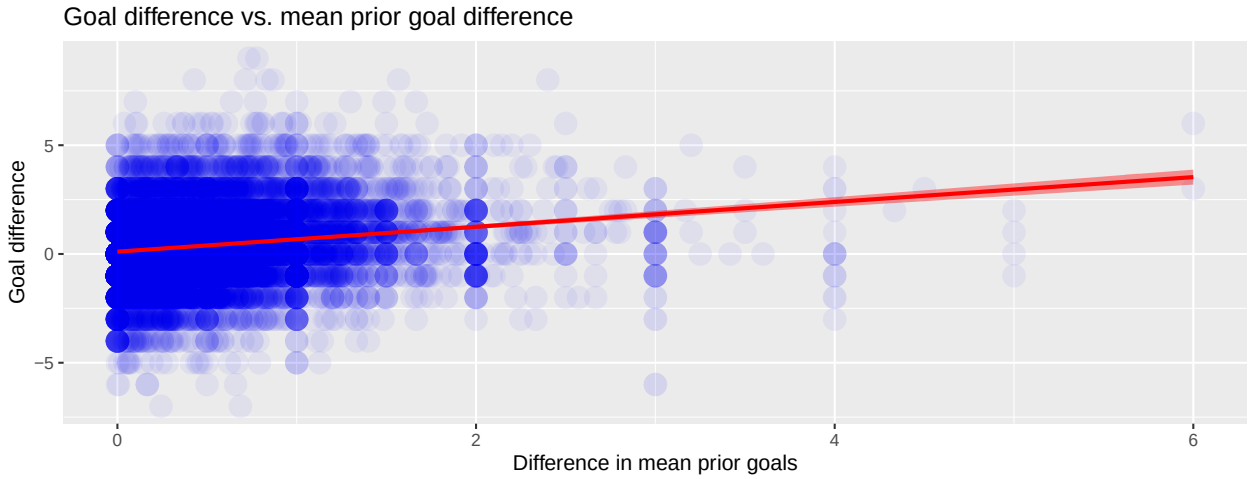


Figure 10: Goal difference vs. mean prior goal difference

As expected, Figure 10 shows a very clear relationship: the more goals you've scored in the past, the more goals you're likely to score in your next match.

3.3.9 Insights - feature selection

My analysis has shown the following features are worth including in machine learning modeling:

- Home field advantage
- Team value difference
- Foreign player count difference between teams
- Mean age difference between teams
- Week in season

- Mean prior goal difference
- Mean prior points difference
- Mean prior yellow card difference
- Mean prior red card difference

But the following feature is not worth including:

- Team size

In initial experimentation, I included the season as a modeling feature, however this increased the RMSE values for different models. This may be because it split the data into groups too small to make an accurate fit to the data.

Because COVID-19 is such a disruptive event, I will exclude the 2020-2021 season from my modeling work. A similar argument could apply to the 2019-2020 season, but for now, I'll include it to avoid reducing the size of my data set.

Not all data is available for all seasons. The team value data, for example, is only available from 2010-2011 onwards. This limits my modeling data set to just the seasons 2010-2011 to 2019-2020.

3.4 Modeling approaches

A soccer match result is a two-dimensional variable, it consists of a home team score and an away team score. This type of problem is known as multi-dimensional target regression [Borchani] and the techniques involved go well beyond the scope of PH125. To transform this into a one-dimensional problem, there are at least two approaches:

- Goal difference between the teams. Predict the goal difference between the teams rather than match score. The easiest way to do this is to calculate $home\ goals - away\ goals$. The approach has the advantage of clearly including a home team advantage field. However, this is approach is less satisfying from a prediction viewpoint because it doesn't predict match scores.
- Split games. In this approach, every game is split in two, with a 'Home' version and an 'Away' version. One model is trained to predict the Home team score and the other model trained to predict the Away team score. Unfortunately, this approach does not explicitly include home advantage, it's implicitly encoded into the other features. On the plus size, this approach makes predictions of match scores, which is more satisfying.

I chose to use the split games approach in this work.

As we're trying to predict match scores and not win/lose/draw outcomes, the problem is a regression problem. This indicates that our loss function should be the RMSE, defined as:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i}(\hat{y}_i - y_i)^2}$$

where:

- N is the number of matches,
- $\hat{y}_i$ is the predicted number of goals for the ith game
- y_i is the actual number of goals for the ith game

Plainly, a wide range of factors influence EPL matches. It's not clear how they vary and interact. This suggests it may be worth trying a variety of different modeling approaches. I used several different regression model types, all using the caret package [Kuhn], except the naive mean model. All of models are supervised learning models. Here are the approaches I chose:

- Naive mean score model. A simple model that's just the mean scores of the (training) data set.
- Generalized Linear Model. A form of ordinary linear regression.
- Glmnet. Fits lasso and elastic-net regularized generalized linear models.
- SVM. Support Vector Machines - boundary based regression. After some experimentation (not shown here), I selected the svmRadial form of SVM, which uses a non-linear kernel function.
- KNN. K-nearest neighbors. Given that EPL scores are all in close proximity to one another, we might expect this model to return good results.
- Neural nets.
- XGB Linear. This is linear modeling with extreme gradient boosting. Extreme gradient boosting has gathered a lot of attention over the last few years and may be one of the most used machine learning models today.
- XGB Tree. This is a decision tree model with extreme gradient boosting.
- Random Forest.

3.4.1 Training and holdout data sets

Because a number of features are cumulative (for example, mean prior goals), it makes sense to use a whole season as a holdout season, so I selected 2019-2020 as my holdout. All other seasons I used for training (2010-2011 to 2018-2019). The holdout data set I called holdout and the training data set I called epl.

3.4.2 Seeding

For repeatability, I set the random number seed to 72 at the start of the modeling process.

3.4.3 Coding approaches

This project proved to be computationally very expensive. To speed up execution, I used the doParallel package to run parallel processing across multiple cores. The caret package will use multiple cores once this code is run.

```
cores <- detectCores() - 1
cl <- makePSOCKcluster(cores)
registerDoParallel(cl)
```

Because of the large number of models, I want to keep models consistent in both what they're modeling and in terms of train control for regularization. I used a standard variable for the form argument in the caret train models. Here's the definition of the 'Away' version:

```
train_formula_away <- as.formula(FTAG ~ WeekNumber +
                                 AwayTeamValue + HomeTeamValue +
                                 AwayTeamForeignPlayers +
                                 HomeTeamForeignPlayers +
                                 AwayTeamMeanAge + HomeTeamMeanAge +
                                 AwayMPP + HomeMPP +
                                 AwayMPCR + HomeMPCR +
                                 AwayMPCY + HomeMPCY +
                                 AwayMPAG + AwayMPFG +
                                 HomeMPAG + HomeMPFG)
```

Here's the standard train control term. Note the use of cross-validation, allowing parallel processing, and constraining the prediction results to lie in the range 0 to 9. As Figure 1 showed, the highest number of goals scored in an EPL match in the last 30 years was 9, so this seems like a reasonable choice for an upper bound.

```
train_control <- trainControl(method = "repeatedcv",
                              number = 10,
                              repeats = 3,
                              p = 0.9,
                              allowParallel = TRUE,
                              predictionBounds = c(0, 9))
```

For model implementation, the code looks like this (SVM model example). The variable formula_ is either the 'Home' or 'Away' (train_formula_away) version of the training formula.

```
fit_svm <- train(form=formula_,
                 method = "svmRadial",
                 data = epl,
                 metric='RMSE',
                 maximize=FALSE,
                 trControl = train_control,
                 preProcess = c("center", "scale", "nzv"),
                 tuneGrid = expand.grid(
                   C = seq(from=0.2, to=4, by=0.2),
                   sigma = seq(0.001, 0.01, 0.001)))
```

In the caret package, random forest training accepts the number of trees as a parameter (ntree), but does not tune for it. To find the optimum number of trees, I searched through a range of ntree values. However, although there's regularization in random forest models using the train control parameter, because ntree is not a tuning parameter, I could overtrain the model. For finding ntree in my random forest model, I split the epl data into a test and train set: the test set was the season 2018-2019, and the training set was all other seasons (except 2019-2020). Once I found an optimal ntree value, I used the epl and holdout dataset for training and evaluation.

4 Results

4.1 Naive means

I used the epl training set to calculate a mean home team score and an away team score. I evaluated this model on the holdout data set (2019-2020 season). Here are the results.

Method	RMSE.away	RMSE.home
naive mean	1.198747	1.247664

Notice the away RMSE is lower than the home RMSE, which might suggest there's more variability in home games.

4.2 GLM

Again, I used the epl data set for training and the holdout set for evaluation. This model ran extremely quickly.

Method	RMSE.away	RMSE.home
glm	1.147625	1.200173

The results are an improvement over the naive means and again note the higher RMSE for home games.

4.3 GLMnet

This is a slightly more complex model than the GLM and the results are a very slight improvement on the GLM.

Method	RMSE.away	RMSE.home
glmnet	1.144463	1.19012

4.4 SVM

In the caret package, the svmRadial model has two tuning parameters, C and sigma. I tuned these parameter ranges by hand (the code I used for SVM is shown above). Notably, this step is computationally expensive.

The charts for the home and away RMSE results are similar, so I'll only show the home result for illustration. The optimal C and sigma values were in the middle of the range of values I tested. It's possible I might minimally improve RMSE by tweaking the tuning grid, but this will be expensive.

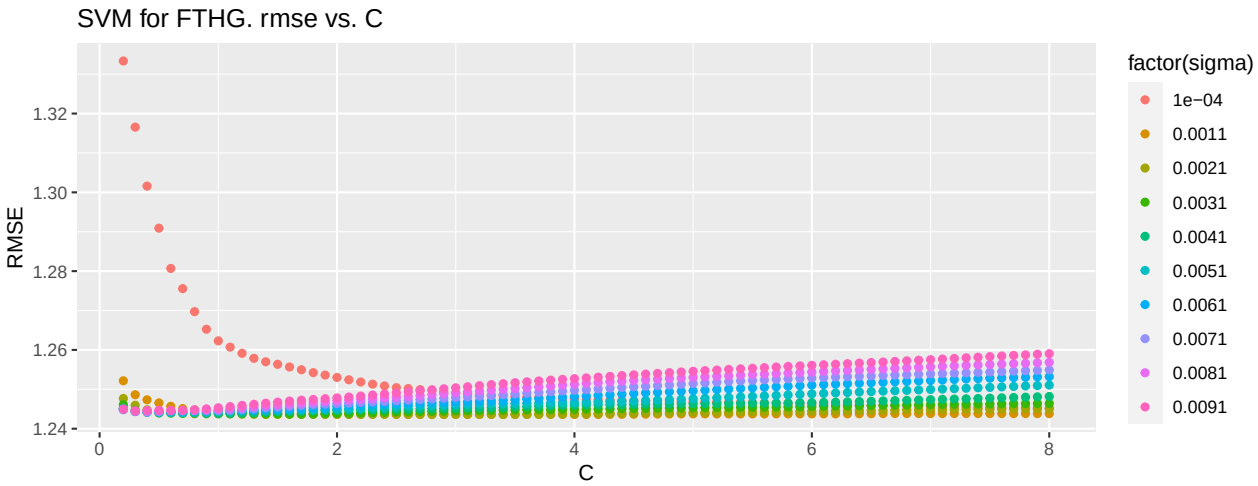


Figure 11: RMSE vs. C and sigma for home model

Method	RMSE.away	RMSE.home
svm	1.135394	1.158612

4.5 KNN

As shown in this code snippet, I varied the number of nearest neighbor nodes from 1 to 40.

```
fit_knn <- train(form=formula_,
                 method = "knn",
                 data = epl,
                 metric='RMSE',
                 maximize=FALSE,
                 trControl = train_control,
                 preProcess = c("center", "scale", "nzv"),
                 tuneGrid   = expand.grid(k = 1:40))
```

The curve of RMSE vs k (the number of nearest neighbors) for the away team shows the result we would expect (Figure 12), the home model is very similar. I could have achieved a lower RMSE by increasing the maximum number of neighbors, however the improvement will be very small and the price will be more computational time.

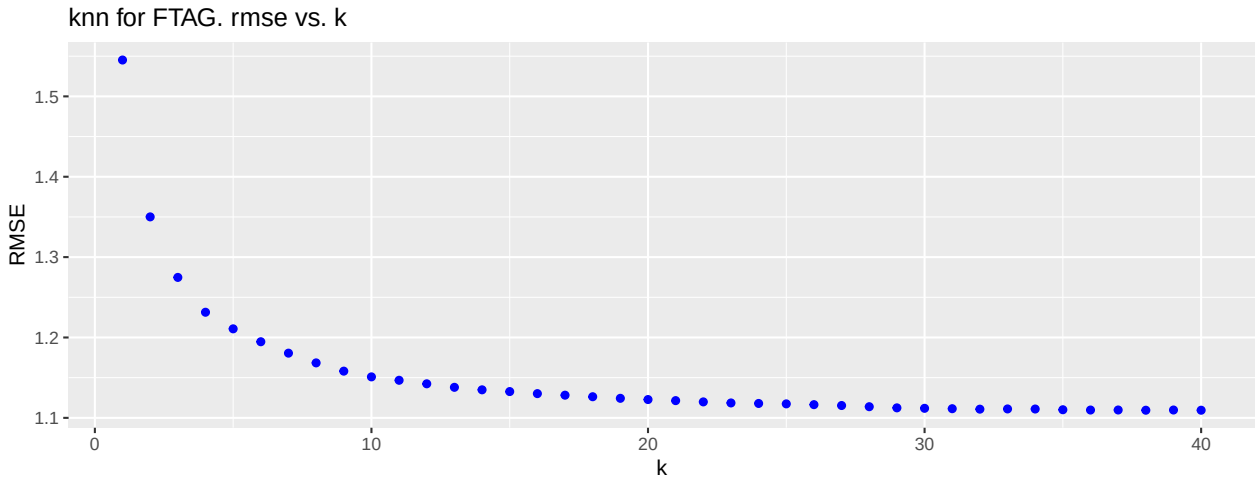


Figure 12: RMSE vs. k for away model

Method	RMSE.away	RMSE.home
knn	1.12715	1.16726

4.6 XGB linear and tree

There are a number of tuning parameters for both model types, so many that searching for optimal tuning parameters takes time and memory. Because of the high memory consumption of these model types, I had several crashes and had to restrict the tuning space.

Here's a code snippet for the xgbTree model that shows the tuning parameters. I had to restrict min_child_weight, subsample, and gamma. The other parameters I chose by trial and error, making sure that the final values were not at the ends of the ranges.

```
xgbTuningGrid <- expand.grid(nrounds = seq(from=1, to=250, by=20),
                             eta = c(0.01, 0.025, 0.05, 0.1, 0.3),
                             max_depth = c(1, 2, 3, 4, 5),
                             gamma = c(0, 1),
                             colsample_bytree = seq(0.5, 0.9, 0.1),
                             min_child_weight = 1,
                             subsample = 1)

fit_xgbTree <- train(form=formula_,
                     method = "xgbTree",
                     data = epl,
                     metric='RMSE',
                     maximize=FALSE,
                     trControl=train_control,
                     preProcess = c("center", "scale", "nzv"),
                     tuneGrid=xgbTuningGrid)
```

Here are the results. The xgbTree modelis the only model that returns a lower RMSE for home matches than away matches.

Method	RMSE.away	RMSE.home
xgbLinear	1.151539	1.183693
xgbTree	1.154226	1.152681

4.7 Random forest

The first step with this model was finding an ntree value for the home and away variants. To avoid overtraining, I split the epl (training) data set into two data sets for this work.

```
epl_train <- match_results %>% filter(Season!='2018-2019')
epl_test <- match_results %>% filter(Season=='2018-2019')
```

I kept with the same idea of using a whole season (2018-2019) as a holdout.

The chart of RMSE vs ntree followed the same pattern for the home and away models. For brevity, I'll just show the away plot. Finding ntree was extremely computationally intensive, even using multiple cores, it takes close to 24 hours to calculate an optimal ntree for the home and away variants.

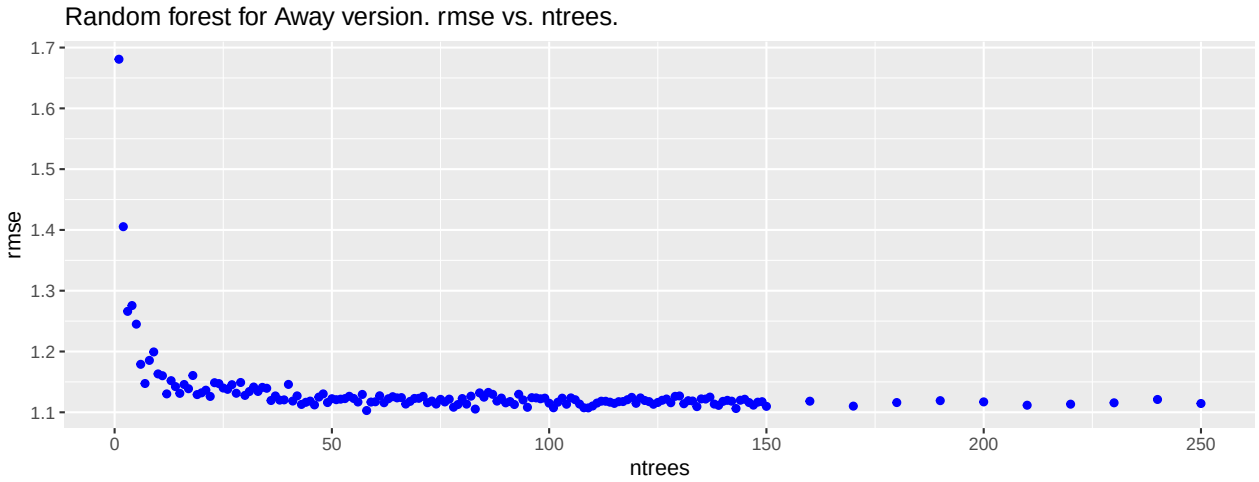


Figure 13: RMSE vs. ntree

Unsurprisingly, the difference in optimal ntree values for home and away teams was small. The optimal ntree value for home teams was 49 and for away teams 58.

Using these ntree values, I trained a random forest model (see code below).

```
fit_rf <- train(form=formula_,
                method = "rf",
                data = epl,
                metric='RMSE',
                maximize=FALSE,
                trControl = train_control,
                preProcess = c("center", "scale", "nzv"),
                tuneGrid = expand.grid(.mtry=seq(1:20)),
                ntree=ntree)

predict_rf <- predict(fit_rf, newdata=holdout)
RMSE_rf <- RMSE(holdout[,as.character(formula_[[2]])], predict_rf)
```

Method	RMSE.away	RMSE.home
random forest	1.15054	1.174885

With all of my models, I examined variable importance to gain a sense of how the different features were contributing to the model. The results were very similar model to model, so I'll just show the random forest results as an example (see Table 1).

Table 1: Variable importance for random forest for away and home

	Overall - away
AwayTeamValue	100.000000
AwayMPP	87.973032
AwayMPFG	85.196455
HomeTeamValue	78.288656
HomeMPP	68.699015
AwayMPAG	59.445793
HomeMPAG	59.438867
HomeMPFG	54.166503
HomeMPCY	41.791950
AwayMPCY	36.827514
AwayTeamMeanAge	35.388584
HomeTeamMeanAge	27.387274
WeekNumber	23.661186
HomeTeamForeignPlayers	12.124271
AwayMPCR	6.209250
AwayTeamForeignPlayers	2.949038
HomeMPCR	0.000000

	Overall - home
HomeTeamValue	100.000000
HomeMPFG	58.384806
HomeMPP	57.217557
HomeMPAG	55.578454
AwayTeamValue	44.105951
AwayMPFG	43.108103
AwayMPP	42.322332
AwayMPAG	38.477137
HomeTeamMeanAge	36.152006
AwayMPCY	23.195629
WeekNumber	22.079702
HomeMPCY	19.632789
AwayTeamMeanAge	17.020487
AwayTeamForeignPlayers	9.101966
HomeTeamForeignPlayers	3.768251
AwayMPCR	1.247732
HomeMPCR	0.000000

Table 2: Overall RMSE results

Method	RMSE.away	RMSE.home
naive mean	1.198747	1.247664
glm	1.147625	1.200173
glmnet	1.144463	1.190120
xgbLinear	1.151539	1.183693
xgbTree	1.154226	1.152681
svm	1.135394	1.158612
knn	1.127150	1.167260
nnet	1.121113	1.174135
random forest	1.150540	1.174885

The home team mean prior red card data doesn't contribute to either the home or away model, but the away team red card result does. This is consistent with the literature that suggests referees penalize away teams more and consider a team's prior discipline record [Dawson]. In future work,I might consider removing mean prior red cards as a modeling features.

The slightly different ordering of features suggests that different dynamics are at work for home and away teams, which is unsurprising given the existence of home field advantage.

4.8 Overall results

In table 2, I present a summary of all RMSE results. A couple of points stand out:

- None of the modeling approaches reduced the RMSE below 1. I'll explore this further in the 'Conclusion' section.
- For all models except XGB Tree, the RMSE is worse for home teams than away teams. This suggests there are features relevant for home team modeling that I haven't captured.
- The best model for away team prediction is neural networks, for home team prediction the best model is XGB Tree. Both of these models are non-linear.

5 Conclusion

The most important step to improve the RMSE results is to understand where the model results are incorrect. If we can understand where models are wrong, we can correct them or create an ensemble model, where one model corrects for the weakness of others. I'm using a single season as my holdout data set, which is only 380 games, a small enough data set to investigate by hand. Here are the areas I think are most worth investigating.

- Start of season effects. I'm using several cumulative quantities as model features, for example, mean prior goals, mean prior points, and mean prior red cards. Each of these is reset to zero at the start of the season, reducing their predictive effectiveness for the first games of the season. If this is a problem, one solution is to carry over last season's results for the next season (with some special allowance for newly promoted teams).
- Season effects. I did not use season as a modeling features because it increased the RMSE. However, there is evidence that from other authors of changes over time in the behavior of the EPL (see for example [Bradley]). It may be possible to model these changes in a way that does not add more noise.
- Prediction distribution. Figure 1 shows the actual distribution of results. The forecast distribution should be compared to the real distribution to check for discrepancies, for example, the model may predict 2-2 games more often than they occur.

This project suffered from a lack of data in two areas: too few seasons to model and too few fields. The EPL started in the 1992-1993 season, but prior to that, *all* of the teams were in the old First Division and there were identical relegation rules. One way forward is to use the old First Division results, which go back over a hundred years. The availability of transfer data restricted modeling to 2010-2011 onward, but transfer fees were publicly known for a long time, even before the start of the EPL. Similar arguments apply to foreign player and other data. Finding new free data sources may be a significant effort, and may involve digitizing old records. Some free sources do have some detailed match level data, for example, the number of corners taken in each match, the players in the match, etc., but the data will require significant effort to scrape. Other new field data that may be easier to obtain includes:

- match attendance - if spectators have an effect on performance, then more spectators ought to have a larger effect. This data is available from sources other than the two I used for this project.
- weather - it's possible that some weather conditions favor some teams. This data is available, but not easily.
- injuries - if 'star' players are injured and can't play, that may well impact performance. We could infer this through the starting lineup of players in the match. Taking this idea a stage further, we could use attempt a calculation of the team value based on the players on the pitch.
- league-within-a-league - several authors, for example Porter [Porter] have pointed out that the EPL seems to have a league-within-a-league of several clubs. These clubs tend to be wealthier than the others and tend to cluster at the top of the league table season after season. It might be worth adding a model feature for these teams (possibly as a hot one encoded 'superleague' field).

The EPL is just one league. There are four fully-professional leagues in England, and obviously many more throughout the world. One solution to the lack of data is to broaden the scope of the model to more leagues and more countries, however, we may again be hampered by a lack of data and data inconsistencies across leagues.

The best performing home model was XGB Tree, a non-linear model. The best away model was a neural network model, which can have non-linear behavior. It may well be that soccer is a very non-linear game. If this is the case, using other non-linear modeling methods may yield better results. However, this may require moving beyond what's available in the caret package.

As I stated earlier, a soccer match should be modeled as a multi-target regression model. It may be that the simple models I've used here are too simple. A good case in point is modeling home field advantage. The simple split model can't include a home field hot one encoded field (it would always be 1 for the home team

and always 0 for the away team, meaning modeling home and away teams separately would ignore it). It's noticeable that model performance was worse for predicting home results.

None of these models individually brought the RMSE down under a goal and the improvement on the naive model was only of the order of 8% or so. Other authors have used machine learning models for score prediction in the EPL [Baboota, Bilek, Bunker], with similar limited success. In these cases, they used a classification approach to predict win, loss, or draw. These models had an accuracy of about 0.56 at best.

The top team of the 2015-2016 Premier League Season was Leicester City, a surprising result given the relative cheapness of the team and their unimpressive previous track record. At the start of the season, bookmakers were quoting odds of 5,000-1 for Leicester City to win. Leicester's win points out that soccer is not deterministic and surprising results do happen, even over the course of an entire season. This suggests there may well be limits to the predictive power of any machine learning model. In other words, there may be an RMSE floor for EPL predictions.

6 References

[Allen] Mark S. Allen, Marc V. Jones, The home advantage over the first 20 seasons of the English Premier League: Effects of shirt colour, team ability and time trends, International Journal of Sport and Exercise Psychology. Vol. 12, No. 1, 10–18

[Baboota] Rahul Baboota, Harleen Kaur, Predictive analysis and modelling football results using machine learning approach for English Premier League, International Journal of Forecasting, Volume 35, Issue 2, 2019, Pages 741-755,

[Barros] Barros CP, Leach S. Analyzing the Performance of the English F.A. Premier League With an Econometric Frontier Model. Journal of Sports Economics. 2006;7(4):391-407

[Bilek] Gunal Bilek and Ulas Efehan, "Predicting match outcome according to the quality of opponent in the English premier league using situational variables and team performance indicators." International Journal of Performance Analysis in Sport 19, no. 6 (2019): 930-941.

[Borchani] Borchani, H., Varando, G., Bielza, C. and Larrañaga, P. (2015), A survey on multi-output regression. WIREs Data Mining Knowl Discov, 5: 216-233,

[Bradley] Paul S. Bradley, David T. Archer, Bob Hogg, Gabor Schuth, Michael Bush, Chris Carling, and Chris Barnes. "Tier-specific evolution of match performance characteristics in the English Premier League: it's getting tougher at the top." Journal of sports sciences 34, no. 10 (2016): 980-987.

[Bunker] Rory Bunkera, Teo Susnjakb, The Application of Machine Learning Techniques for Predicting Results in Team Sport: A Review, arXiv preprint arXiv:1912.11762.

[Butler] Robert Butler, Patrick Massey, Has Competition in the Market for Subscription Sports Broadcasting Benefited Consumers? The Case of the English Premier League, Journal of Sports Economics

[Cutler] Adele Cutler, Leo Breiman, Archtype analysis, Technometrics, 36(4), pp3380347

[Dawson] Peter Dawson, Stephen Dobson, John Goddard, John Wilson, Are football referees really biased and inconsistent? Evidence on the incidence of disciplinary sanction in the English Premier League, Journal of the Royal Statistical Society: Series A (Statistics in Society), 170: 231-250

[Epstein] Richard A. Epstein, The Theory of Gambling and Statistical Logic, Academic Press, 2nd Edition, 2009

[Kuhn] Max Kuhn, Building Predictive Models in R Using the caret Package, Journal of Statistical Software November 2008, Volume 28, Issue 5

[Leard] Leard B, Doyle JM. The Effect of Home Advantage, Momentum, and Fighting on Winning in the National Hockey League. Journal of Sports Economics. 2011;12(5):538-560.

[McCarrick] McCarrick, Dane, Merim Bilalic, Nicholas Neave, and Sandy Wolfson, Home Advantage during the COVID-19 Pandemic in European football (2020).

[Mezrich] Ben Mezrich, Bringing down the house, Atria Books, 2002

[Meloche] Renee Meloche, The High-Tech Gambler: The True Story of Keith Taft & His Astonishing Machines, 2012

[Oberstone] Joel Oberstone, Differentiating the Top English Premier League Football Clubs from the Rest of the Pack: Identifying the Keys to Success, Journal of Quantitative Analysis in Sports, Volume 5, Issue 3, 2009, Article 10

[Plumley] Daniel James Plumley, Robert Wilson and Simon Shibli, A holistic performance assessment of English Premier League football clubs 1992-2013. Journal of Applied Sport Management, 9 (1)

[Pollard] Richard Pollard and Gregory Pollard, Home advantage in soccer: a review of its existence and causes, International Journal of Soccer and Science Journal Vol. 3 No 1 2005, pp28-44

[Porter] Chris Porter, (2019) The Social, Cultural and Political Shaping of English Football. In: Supporter Ownership in English Football. Football Research in an Enlarged Europe

[Rein] Rein, Robert, Memmert, Daneiel, Big data and tactical analysis in elite soccer: future challenges and opportunities for sports science. SpringerPlus 5, 1410 (2016).

[Robinson] Joshua Robinson, Jonathan Clegg, The Club: How the English Premier League Became the Wildest, Richest, Most Disruptive Force in Sports, Mariner Books, 2019

[Tang] Cheng Tang, Damien Garreau, Ulrike von Luxburg, When do random forests fail?, 32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montreal, Canada

[Thomas] Thomas S, Reeves C, Bell A. Home Advantage in the Six Nations Rugby Union Tournament. Perceptual and Motor Skills. 2008;106(1):113-116

[Vergina] Roger C.Vergina, John J.Sosika, No place like home: an examination of the home field advantage in gambling strategies in NFL football, Journal of Economics and Business Volume 51, Issue 1, January–February 1999, Pages 21-31