Understanding Soccer Scores

Abheek Basu, Christine Huang, Ege Sagduyu

May 2, 2016

Abstract

In this paper, we analyze soccer data from 1888 to 2014 in order to discover trends and develop a match outcome classifier. This objectives can be broken down into two main questions: how did soccer's evolution as a sport impact scoring patterns, and would a Classifier based on the Elo rating system produce reliable, above-average results?

1 Background

1.1 Motivating Ideas/ Hypothesis

The Elo rating system was developed in the 1950s to provide an objective measure of ability in chess. It was different from other rating systems at the time because of its statistical grounding and has recently been applied to other sports. At the same time FIFA (the international governing body of soccer) determines ranking (not to be confused with game scores, which record the particular outcome of a single game) with an arbitrary system with many unjustified adjustments and assumptions. We thus explore whether the Elo rating system could be used to create a good predictor of soccer match outcomes.

1.2 Overview of Football Ranking

The Fédération Internationale de Football Association (FIFA) is an association founded in 1904 and based in Zurich. It has 209 member associations and its goal is the constant improvement of football. As one of the most prominent organizations, their ranking system is widely regarded, despite not being the most mathematically rigorous.

FIFA's ranking algorithm is as follows, where P is the total number of points that determine the final ranking:

P = M x I x T x C x 100

M: Was the match won, or did it end in a draw?
I: How important was the match (friendly match to FIFA World Cup$^{\text{TM}}$)?
T & C: How strong was the opponent in terms of its ranking position and the confederation to which it belongs?

Each variable corresponds to different scale (i.e., Match is a 3-level variable, Status of match depends on 'importance' from a scale of 1 to 4, and strength of opponent/confederation both rely on recursive calculation of previous Match and Status). Additionally, FIFA arbitrarily selects a k-factor for all point calculations. This is problematic as a higher-than-optimal k-factor will overfit on recent events, while a lower-than-optimal k-factor will not be sensitive enough to changes. Having the right amount of sensitivity is important due to how teams and players evolve: generally slowly, but with potential "black swans" in terms of performance improvement.

As is manifest above, there is quite a bit of room for ambiguity and subjectivity, especially where the variable I is concerned. Since the ambiguity of I calls the validity of the ranking into question, variables T & C are also subject to question due to lack of nuance.

1.3 Creating A Rating System - Theory

As described above, FIFA's rating system itself is quite susceptible to variable influences unrelated to actual performance, and there are ways in which we can make the measure more quantitatively based. To begin with, the Elo rating assumes normality in random variable distribution, and that performances of teams only change slowly and over time. In other words, according to the Elo rating scheme, a team's true skill is more like the mean of that team's performance random variable.

The Elo rating system is designed for two player (or team) games and models two teams, A and B with rankings R_a and R_b. A win is equivalent to a score of 1, a loss to 0 and a draw to 0.5. The expected score (E_A) of team A is given by $1/1+e^{(R_b-R_a)}$ and the the expected score (E_b) of team B is $1/1+e^{(R_a-R_b)}$. The sum of the expected scores of A and B add up to 1.

1.3.1 Update Rule

After a match the ranking of A is updated according to the update rule:

$R'_a = R_a + K(S_a - E_a)$

and the ranking of B is updated according to the update rule:

$R'_b = R_b + K(S_b - E_b)$

where S_a and S_b are the scores of team A and team B respectively. K is a factor that determines the score volatility and can be thought of as a form of memory. Small K factors, mean any one match will not affect a teams ranking as much, whereas higher K factors give more weighting to more recent games.

This eliminates the arbitrary nature of having some 'important games' being weighted disproportionately, and provides a more neutral perspective on team capabilities. Additionally, we will also be testing for different K factors (5, 10, 15, 25, 50) in order to account for both short term and long term change in performance levels. It is important to note that K factors and team scores are relative, and meaningless when considered on there own. For example K factor of 10 is very high if the starting score of a team is 60.

2 Summary of Data

All of the data was obtained from a source for an article on the website FiveThirtyEight.[1]Data dates from 1888 to 2014, recording every soccer match in the top four leagues in the professional English soccer system. There are 190,096 matches recorded with 12 variables noted in each match, i.e. $n = 190,096, m = 12$.

2.1 Variables

The variables we work with in this project include:

Date – Date of match
Season – Season of match
home – Which team is playing in the home stadium
visitor – Which team is playing as the visitor (or away)
FT - Scoring outcome (home goals - away goals)
hgoal - Total home team scores
vgoal – Total visitor team scores
division – Division of the team
tier – Tier of the team
totgoal – Total goals scored
goaldif – Difference in goals scored between teams
result – Outcome of game: "A"= Away Win, "D" = Draw, "H" = Home Win

Tier and division are for the most part the same, though for various historical reasons, division 3 was split into two divisions for a short period of time, thus we use tier as a very accurate proxy for the league teams played in.

3 Data Exploration

The data from the source was already well formatted and clean. Even when we checked for missing values, NAs and badly formed input there were none. Interesting points to note include the fact that there were only 95 different score outcomes over 200 years, with an interesting distribution of frequencies (described in part later).

For our initial data exploration, we wanted to look at how many teams played across different years and different leagues. After running exploratory scripts in `R` to output which years and across which leagues a given team played in, we then recorded the `R` outputs as text files for further analysis. We decided to use `Python` for this part due to easier parsing and data manipulation capabilities. To give a short summary of what we did, using the information we had on years

[1] Original code can be found on this GitHub repository.

and leagues our teams were active in, we created binary responses to mark the years/leagues a given team was present in (i.e. it would equal 1 if the team played in a given year/league and 0 otherwise). This enabled us to create the following three plots.[2]

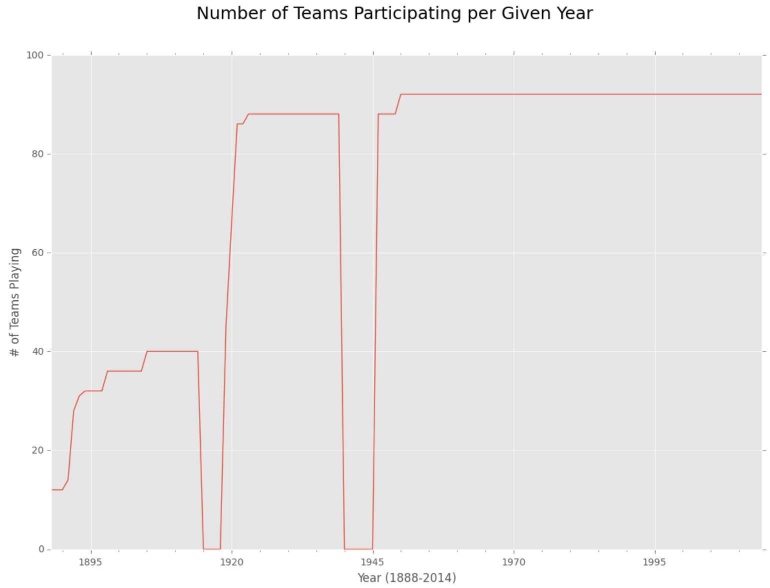


Figure 1: Plot of number of teams playing in any league at a given year from 1888 to 2014

First, we looked at how many teams were active in a given year to see how the total numbers changed over the years, which is given in Figure 1. An immediate realization from this plot is that there are 2 periods in which the activity drops to zero. Looking at the years, it is clear that these are the years that correspond to World War I and World War II. This makes intuitive sense, since most of the players should be of prime drafting age for the military at the time. Overall though, the number of teams show a steady increase right until the end of WWII, where it peaks at 92 teams. As the years following Second Industrial Revolution increased people's ability to spend money and afford leisurely activities over time, it logically follows that the demand and participation for a competitive past-time activity such as soccer also increased with it.

[2] This entire script for cleaning, parsing and plotting the three plots is provided in the file textcleaning.py.

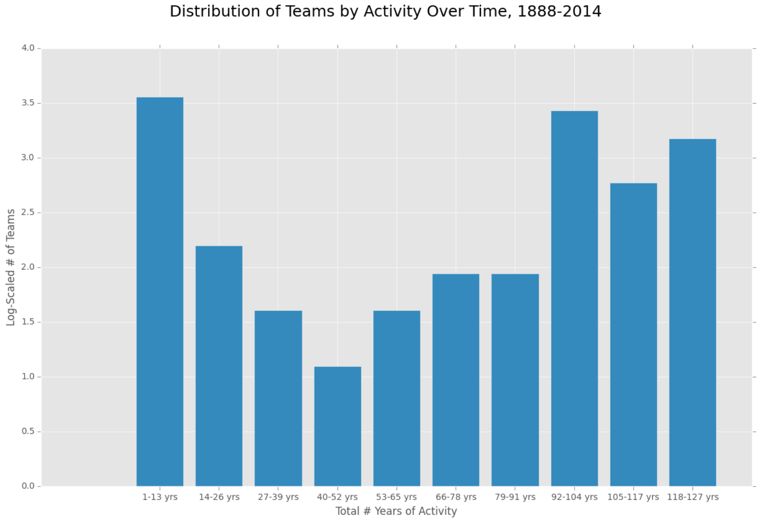


Figure 2: Number of Teams by Activity Periods

As a follow-up to the previous data exploration, we also looked at total years of activity across teams over time, which is depicted in Figure 2. We decided to split the total active years into 10 buckets for easier presentation. We also used a log-scale for total number of teams in a given bucket in order to make the distribution across the bar charts much more uniform. Looking at the results above, we can see that there is an accumulation at the tail ends of our bucketing. A significant portion of the teams are not active for more than 13 years, and the ones that are active for a long time tend to remain active (especially after the 92-years-total mark). If a researcher is not aware of the existence of this fact and choose the sample without accounting for this, they might be susceptible to survivorship bias, where they might include too many of the long-time winners into the model and not account for the variability added by other teams. Hence, looking at the distribution across buckets is a very useful preliminary step before data analysis.

The last related bar chart we have constructed is given in Figure 3, which aggregates the teams into the total number of leagues they have participated in over time. The English football league system is a complicated one[3], so for our purposes, we only collected participation data for the top 4 levels of the English football league system, which are English Premier League, Football League Championship, Football League One, and Football League Two. Looking at the plot, we can see that most of the teams have experienced relegation (dropping down one league) across leagues at least once. Also, we should note that moving within the Football League itself is much more common than moving to Premier League, which might explain the slight drop in the number of teams who played across all 4 leagues.

[3] For more detailed information about the English football league system, visit the associated Wikipedia page by clicking here.

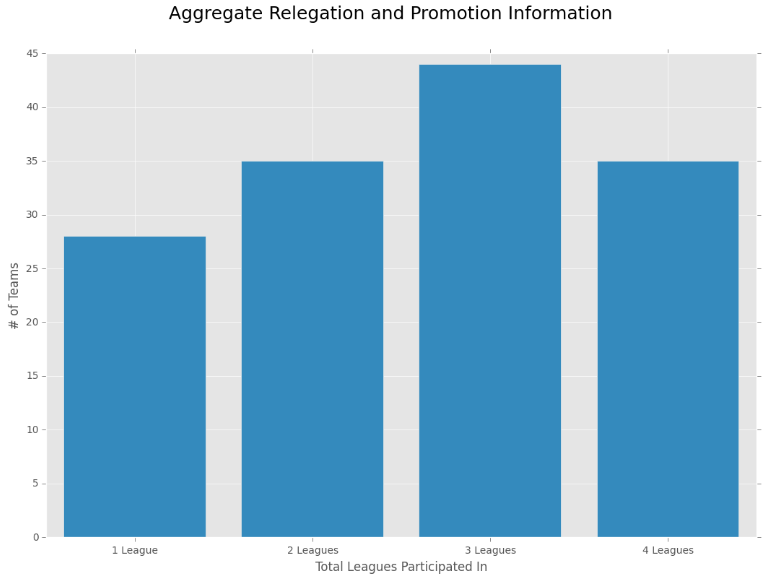


Figure 3: Bar chart of number of teams for the total number of leagues competed in over time

As mentioned earlier, over the nearly 150 years of soccer data only 95 score outcomes where observed. We present a plot 4 below showing the how often "0-x", "1-x" and "2-x" scores occurred, where the y in "y-x" represents the home score and the x represents the away score. We show "0-x" "1-x" and "2-x" because they encompass some of the most frequent scores that occurred

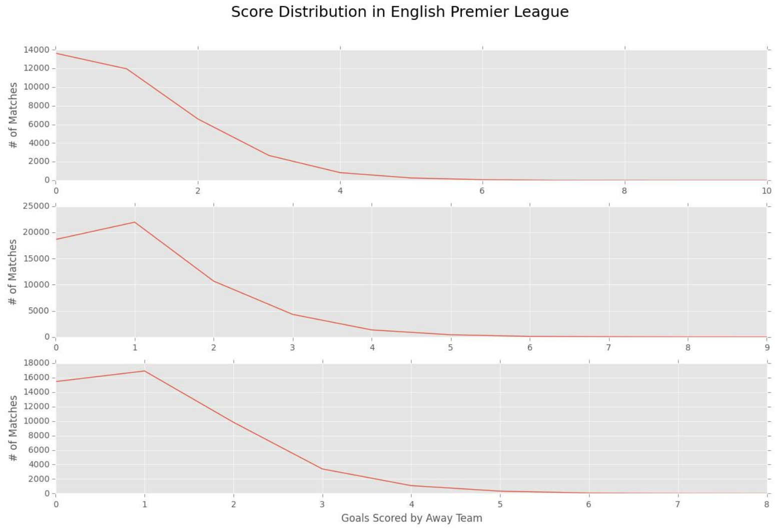


Figure 4: Bar chart of number of teams for the total number of leagues competed in over time

The most 5 common scores from 1960 to 2014 in order were "1-1" , "1-0", "2-1", "2-0" and "0-0".

3.1 Changing Win-rate

The most important thing that the exploratory analyses showed us would be how win percentage patterns changed over time. **We define win-rate as the number of matches that result in a definitive result, i.e. not a draw**. To determine if win-rate had changed dramatically over time, we calculated the win rate in every 1,5,10,20, and 50 year window from 1889 to 2014 for every league as well as across all leagues. We created a time series of the winrates from the different windows. This process was computationally expensive taking $O(n^3)$ time, however resulted in interesting results.

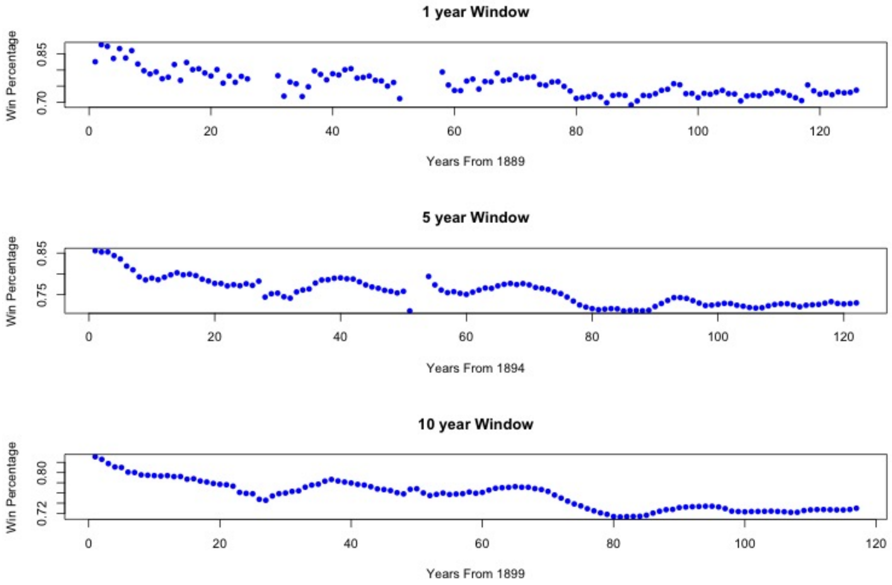


Figure 5: Win-rate Calculated Across All Leagues Over 1,5 and 10 year Windows

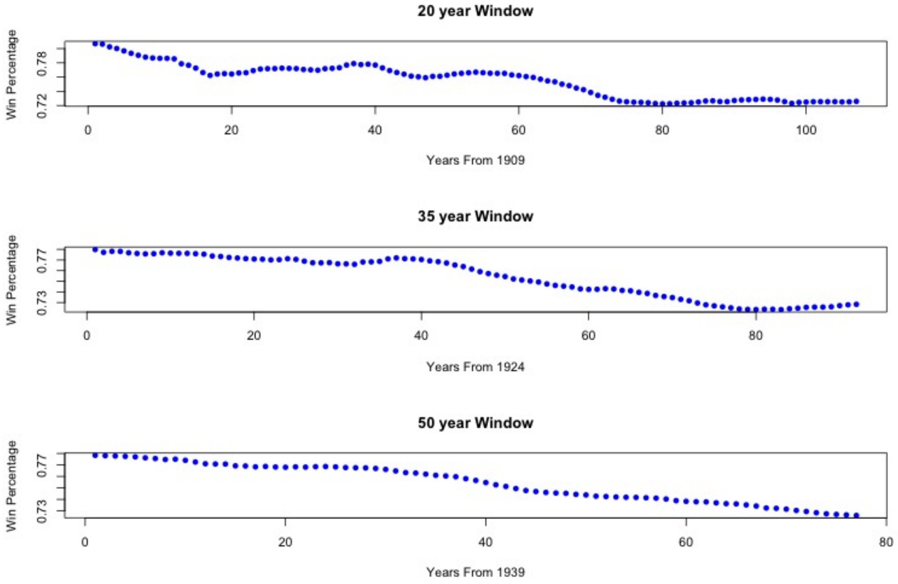


Figure 6: Win-rate Calculated Across All Leagues Over 20,35 and 50 year Windows

As can be seen in 5 and 6, the yearly win percentages decrease from 1889 quite dramatically from upwards of 80% in the early days of soccer to around 73% today. Additionally the 5, 10 and 20 year window win-rates seem to suggest the win-rates have permanently reduced, especially after the 1960's. We suggest two possible reasons for this. The ball used in soccer has undergone significant changes over time and only became stable in its bounce during the 1960s with the introduction of a more spherical ball. An alternative hypothesis is that teams have become more evenly matched over time, though this hypothesis does not explain the specific decline in winrates during the 1960s.

Given the pattern of winning percentages, we decided that it would make most sense to analyze data starting from 1960 to present day, as previous data would most likely disrupt any predictions due to the significant difference in environment.

4 Simulation and Derived Features

This project made extensive use of simulation using the raw data to create our own predictor variables, namely pre-match Elo ratings for each team involved. For every match 1960 to 2014 we created 18 new features: pre-match Elo scores calculated with k-factors of 5, 10, 15, 25, and 50 (accounting for 12/18 of the features) and the differences in Elo scores between the teams (accounting for the remaining 6/18). We ran two main simulations.

4.1 Simulation 1

In the first simulation, all teams across all tiers started of with the same Elo score of 100 in 1960. We then ordered the dataset chronologically, and looped through the observations so that Elo scores were updated correctly. We feared that in this simulation, teams that consistently perform very well in their leagues but do not get promoted receive inflated Elo scores when they do eventually get promoted (and vice versa). On the other hand given that so many teams move up and down leagues and that the we ran the simulation with so many different k factors, we thought if the simulation ran long enough (i.e. over many matches) these adverse effects would be neutralized.

4.2 Simulation 2

In the second simulation all teams in 1960 in tier 1 were equally ranked with 100 points, all teams in tier 2 were ranked with 80 points, all teams in tier 3 were ranked with 60 points, and all teams in tier 4 were ranked with 40 points. We incorporated this difference to adjust for relative differences in strength between leagues and to avoid the problem mentioned in the section above: inflated and deflated scores of consistent strong or weak performers who are then promoted (or relegated). Once again we ordered the dataset chronologically, and looped through the observations so that Elo scores were updated correctly.

5 Analysis

After building the Elo rating system under the two different simulations, we tested how predictive our Elo rating variables were by running different multi-class classifiers over 3 time-frames (2010-2014, 2000-2009, 1980-1999) and looking at the resulting confusion matrices. We aimed to ascertain the efficacy of our models, and to understand any time-varying effects, and whether Simulation 1 created a better Elo rating system or Simulation 2 created a better Elo rating system

We used 2 multi-class classification algorithms as our main methods: Random Forests and Boosting. We did not use the lasso multi-class algorithm because we did not want to use any variable reduction technique in our analysis, given the small number of variables in use. We believed random forests would work well given that they should be able to select the best Elo k-factor at each split. We believed boosting would be a strong choice of classifier because it is often described as the best "out of the box" machine learning technique.

To serve as a baseline, Figure 7 is the actual distribution of the three types of outcomes, A (Away Score), D(Draw), and H(Home Score) across all matches from 1960 onwards across all leagues:

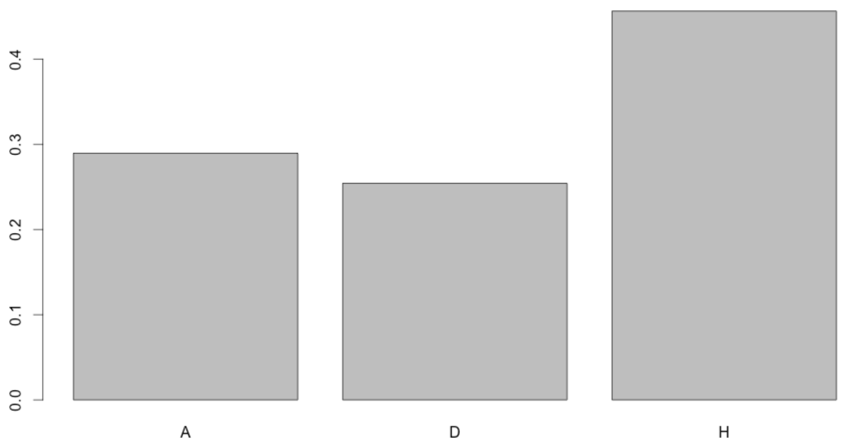


Figure 7: Proportion of Match Outcomes

Given Figure 7 and the well-known fact that home team has a large advantage, our goal with the classifier was to create one that performs at or above the level of a classifier that only uses information about the home team to make a prediction.

Thus to compare how useful our Elo variables, we used the same methodology on all three time periods, comparing 6 models for each time period under each simulation. The process for each time frame, for each simulation is broken down below for clarity.

5.1 General Methodology

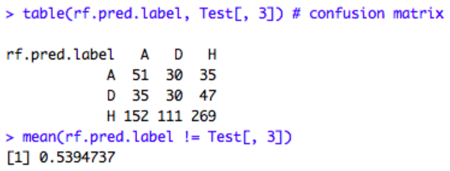
1. Subset training data to contain only Elo features and train Random Forest on this data set.
2. Subset training data to contain only team information, such as team names and home team advantage and train Random Forest on this data set.
3. Subset training data to contain both team information, such as team names and home team advantage, and Elo Features and train Random Forest on this data set.
4. Subset training data to contain only Elo features and use Boosting on this data set.
5. Subset training data to contain only team information, such as team names and home team advantage, and use Boosting on this data set.
6. Subset training data to contain both team information, such as team names and home team advantage, and Elo Features and use Boosting on this data set.
7. Understand which model is best.

5.2 2010 to 2014 Under Simulation 1

For the first set of models, all data from 2010 to 2012 was training data and data from 2013-2014 was testing data.

5.2.1 Model 1

The Random Forest algorithm trained only on the Elo features yielded the following confusion matrix (on the test data) and had an test accuracy of 46.6%.



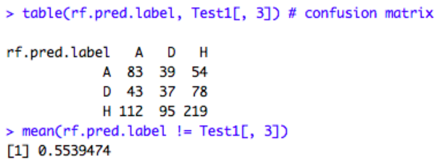
```
> table(rf.pred.label, Test[, 3]) # confusion matrix

rf.pred.label   A   D   H
            A  51  30  35
            D  35  30  47
            H 152 111 269
> mean(rf.pred.label != Test[, 3])
[1] 0.5394737
```

The accuracy is better than random guessing, which would have 33% accuracy and outperforms a classifier that would label everything the majority class in the training set (which would have an accuracy of 45%)

5.2.2 Model 2

We compared the random forest trained on Elo features against a random forest trained only on the team based variables. This model yielded a test accuracy of 44 %:

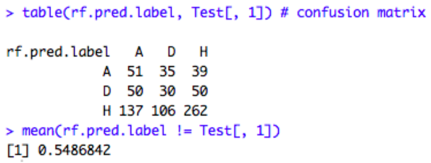


```
> table(rf.pred.label, Test1[, 3]) # confusion matrix

rf.pred.label   A   D   H
            A  83  39  54
            D  43  37  78
            H 112  95 219
> mean(rf.pred.label != Test1[, 3])
[1] 0.5539474
```

5.2.3 Model 3

Finally we wanted to see if a classifier trained on both team factors, such as home team information, and Elo rankings would be best. While the model was better than model 2 it was (surprisingly) worse than just on Elo scores, with an accuracy of 45%



```
> table(rf.pred.label, Test[, 1]) # confusion matrix

rf.pred.label   A   D   H
            A  51  35  39
            D  50  30  50
            H 137 106 262
> mean(rf.pred.label != Test[, 1])
[1] 0.5486842
```

We also trained 3 gradient boosting algorithms, one on just Elo features, another on team factor and home team advantage and third on both team factors and Elo features. The third Boosting algorithm had the best test set accuracy with an accuracy of 46.5%.

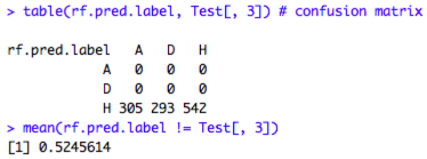
5.3 2010 to 2014 Under Simulation 2

We compared the best model under simulation 1 during the time frame 2010 to 2014 to the best model under simulation 2 from 2010 to 2014. The best model under simulation 2 was also gradient boosting trained on both Elo features, and team factors which had a test set accuracy of 46%, slightly worse than under simulation 1. This was somewhat surprising given the belief that Simulation 2 is a more robust method of creating Elo ratings, with fewer exaggerated Elo ratings.

5.4 2000 to 2009 Under Simulation 1

We tested our models during the 2000 to 2009 time frame and created training data from 2000-2006 and testing data from 2007 to 2010 and compared boosting algorithms and random forests separately on derived Elo Features, Team Factors, and all features (as mentioned in the general methodology section).

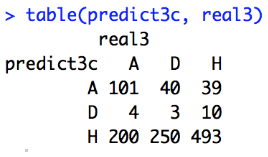
The best random forest was trained on just Elo ratings and outperformed a random forest trained on team factor (team names, home team and visitor team) by 2% in this time frame with an accuracy of 47.5%!



```
> table(rf.pred.label, Test[, 3]) # confusion matrix

rf.pred.label   A   D   H
            A   0   0   0
            D   0   0   0
            H 305 293 542
> mean(rf.pred.label != Test[, 3])
[1] 0.5245614
```

The best overall model was gradient boosting on both Elo features and team factors, which had a test accuracy of 52% shown below.



```
> table(predict3c, real3)
         real3
predict3c   A   D   H
        A 101  40  39
        D   4   3  10
        H 200 250 493
```

5.5 2000 to 2009 Under Simulation 2

Comparing the best models from Simulation 1 to Simulation 2 again, we found that we had better test accuracy under Simulation 1.

5.6 1980-1999 Under Simulations 1 and 2

We tested both random forests and boosting on the 1980-1999 time frame under both Simulation 1 and Simulation 2. We used the same variables as before and found that the best model in this case was boosting trained on Elo scores, and team factors under Simulation 1 (again!). The algorithm had a test accuracy of 51%.

6 Results and Interpretation

Elo scores on their own can provide predictive power similar to home team advantage, though the best classifiers combined team information with Elo scores.

Additionally, 12/18 of the gradient boosting algorithms found that Elo scores created with a K factor of 20 was the most important Elo variable. This has implications for online Elo soccer rating systems that arbitrarily select k factors, as it seems to be that the right ratio between k factor and starting score is 1/5 (20/100).

Surprisingly, Elo rankings derived from Simulation 1 ,where all teams began the simulation in 1960 with the same start rank, were more useful than as Elo rankings derived from Simulation 2 where the 1960 Elo rankings were adjusted for starting league.

This is probably due to the fact Elo score differences in the second simulation all tended to be lower than Elo score differences from the first simulation. As a result it is possible that the classifier was less confident of its predictions there.

Most interestingly it seems as if Elo scores have become less predictive over time. In the 1980-1999 window and the 2000-2000 window our classifiers had above 50% test accuracy but on the 2010-2014 data, it drops 2-3%. This is most likely explained by the fact that Elo scores tended to become more similar over time. Further back in time there were clearer delineations in rankings between teams.