Data Science Methods for Economics and Finance 871 Final Project: Predicting Chess Outcomes

Wesley Williams[a]

[a] *Stellenbosch University, South Africa*

Abstract

Chess is one of the most complicated games in the world. Humans on the other hand are not so complicated and we are the ones playing the game. This made me ask the question: "Is it possible to predict a game of chess based on just on data on the players before the game has started and just the openings used?" I employ an extreme gradient boosting model to answer this question and find that I can only predict the outcome with just over 60% accuracy, which is double the chances of just guessing.

Table of Contents

Email address: 21691126@sun.ac.za (Wesley Williams)
You can find my code on my Github page: https://github.com/wjwilliams/MLPROJ/tree/main/WriteUp

1. Introduction

Chess, often considered one of the most popular games globally, has experienced a surge in popularity in recent years, partly fueled by its portrayal in popular media such as the acclaimed TV show “Queen’s Gambit” and the movie “Pawn.” These portrayals, although fictionalized to varying degrees, shed light on the remarkable life and achievements of renowned chess player Bobby Fischer. The story of Bobby Fischer is both inspirational and incredibly sad. During the Cold War, the United States (US) and the Soviet Union (USSR) engaged in a fierce competition to assert their global dominance. This rivalry extended beyond conventional arenas like the space race and nuclear arms race, encompassing intellectual pursuits such as chess. Chess was revered as the game of the intellectual elite, and both nations sought to establish themselves as the preeminent force in this domain, symbolizing their intellectual superiority. The USSR dominated until Fischer’s victory against world champion Boris Spassky, symbolizing a small triumph for the US in the Cold War context. This illustrates that chess extends beyond a mere board game, carrying significant cultural and geopolitical implications. While the stakes are not as high in the modern era, chess is still seen as the epitome of intellect.

Chess can be broken down into three stages of the game: the opening, the middlegame and the endgame. The opening represents the different strategies of getting all of your pieces into the most optimal positions on the board to both attack your opponents pieces and defend your own. This is the key part of a chess game that will be investigated in the paper. I want to determine if the outcome of a chess game can be predicted using only the information available about the game and players before the game begins and the openings employed by the players. The use of machine learning is crucial in this analysis due to the complexity of the game. I use an extreme gradient boosting model (XGBOOST) with three potential outcomes (white winning, black winning or a draw). The outcome is not binary therefore I cannot use a normal ordinary least squares as comparison and so I use the probability of guessing as the baseline. I also subset the data by ELO rating to assess whether it is easier to predict weaker or stronger players. I find that the opening is slightly more important for weaker players but the model is not as accurate as when using the entire sample. The rest of the paper is structured as follows, in section 2 I describe the data section 3 explores and visualizes the data, section 4 explains the methodology, section 5 presents the results and section 6 concludes.

2. Data

The data was obtained from Kaggle and it includes over 20,000 games played on the online chess platform "lichess". The dataset includes 16 variables but there are only seven variables of interest: 1) the winner given as white, black or draw; 2) time and increment code which details the time control as the base time and the additional time per move; 3) white's ELO rating; 4) black's ELO rating; 5) moves which are all the moves in the game given in chess notation; 6) opening eco which is a code that represents the opening that was played and 7) opening ply which represents the number of moves played in the opening that corresponds to chess theory. An additional variable of rating difference was engineered from the perspective of white, which is just white's rating minus black's rating.

2.1. Factor engineering

Some of the variables need to be wrangled to be used in the model. All of the numerical variables are sufficient for the model but the character variables need to be engineered to be included. Firstly, I am only interested in the first five moves of the game, I therefore expand the moves variable so that I have a variable for each of the first five moves played by each player and disregard the rest. The moves are then converted from chess notation to a unique numerical factor for each piece moving to each co-ordinate on the board. Secondly, I separate the increment variable into the base time and the increment for each move and ensure both are numerical factors. Lastly, I assign a unique numerical value to each of the unique openeing codes to ensure comparability between the training as testings samples[1]. All the variables are therefore sufficiently engineered to be used in the model.

3. Exploratory Data Analysis

In this section I attempt to explore and visualize the data to gain insights into the patterns that emerge with respect all the features and the the target.

[1] I did attempt one-hot encoding but I had issues with the differences in lengths between the training and testing samples

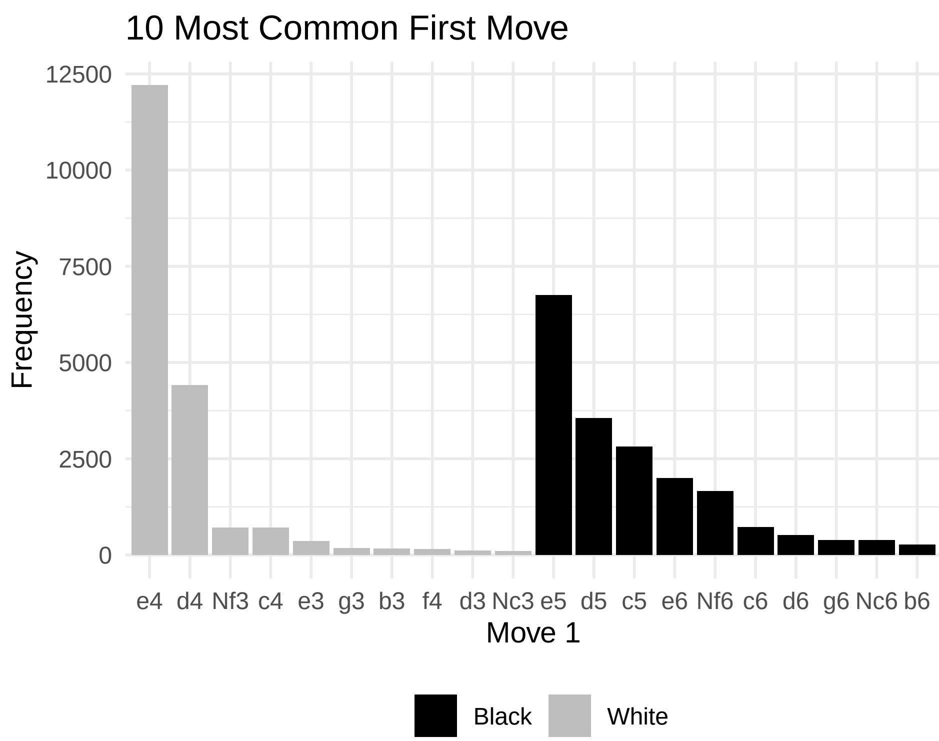

Figure 3.1: Most Common Opening Moves by Colour

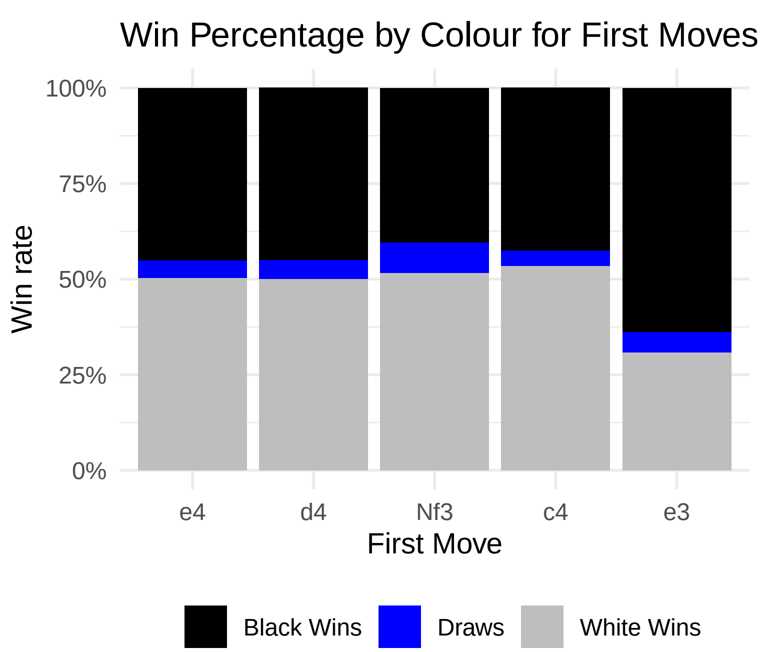

Figure 3.2: Proportions of Outcomes by White First Move

Figure 3.1 shows the most common First move for each colour. It is no surprise that central pawn moves are the most common as controlling the center of the board is instrumental in the opening phase of a chess game. Figure 3.2 shows the outcome proportions for white's first move and it shows that if white claims the center they gain an advantage and black needs to respond. The move e3 instead allows black to claim the center and white loses its first mover advantage. This highlights that mistakes early on in the openings have consequences that last throughout the entire game.

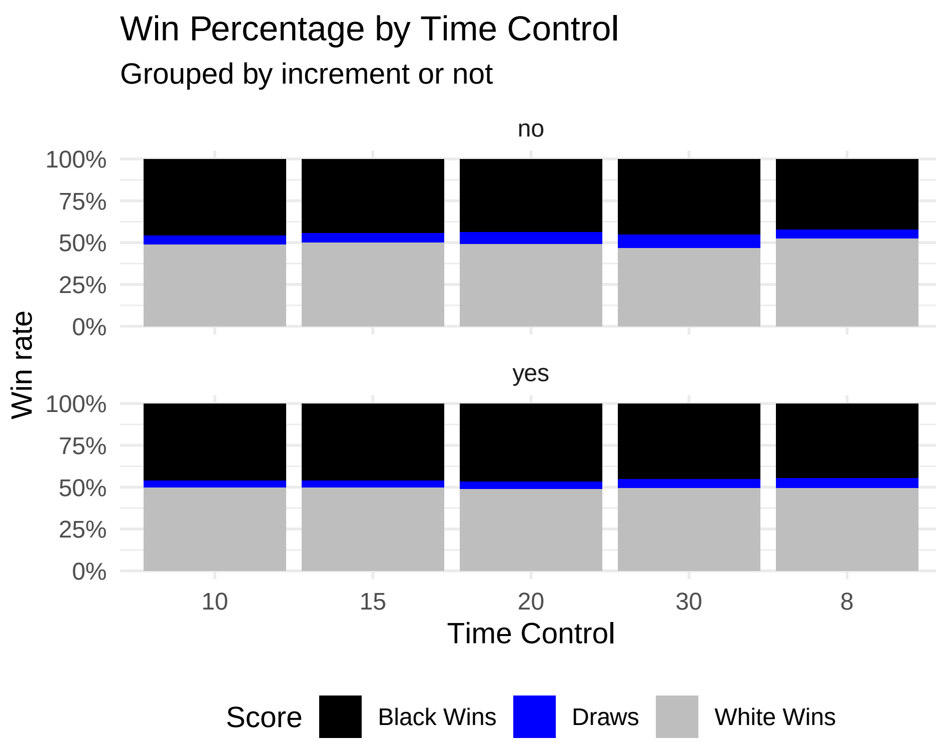

Figure 3.3: Outcomes by Time Controls

Figure 3.3 shows the differences in outcome according to different time controls. It presents the top 5 time controls that are played with and without time increments. There are no significant differences in wins between whether a game has time increments or not. The only real difference is that a draw is more likely with larger time controls and no time increments.

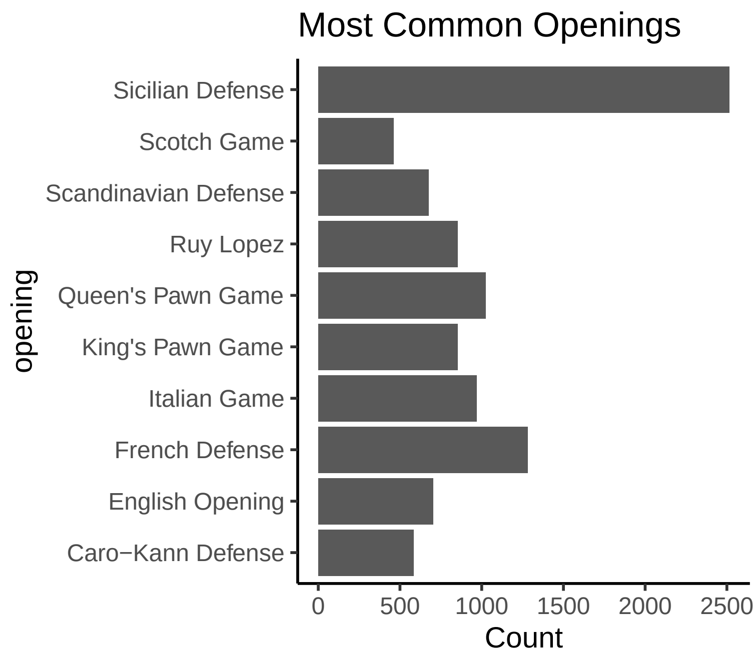

Figure 3.4: Most Popular Openings

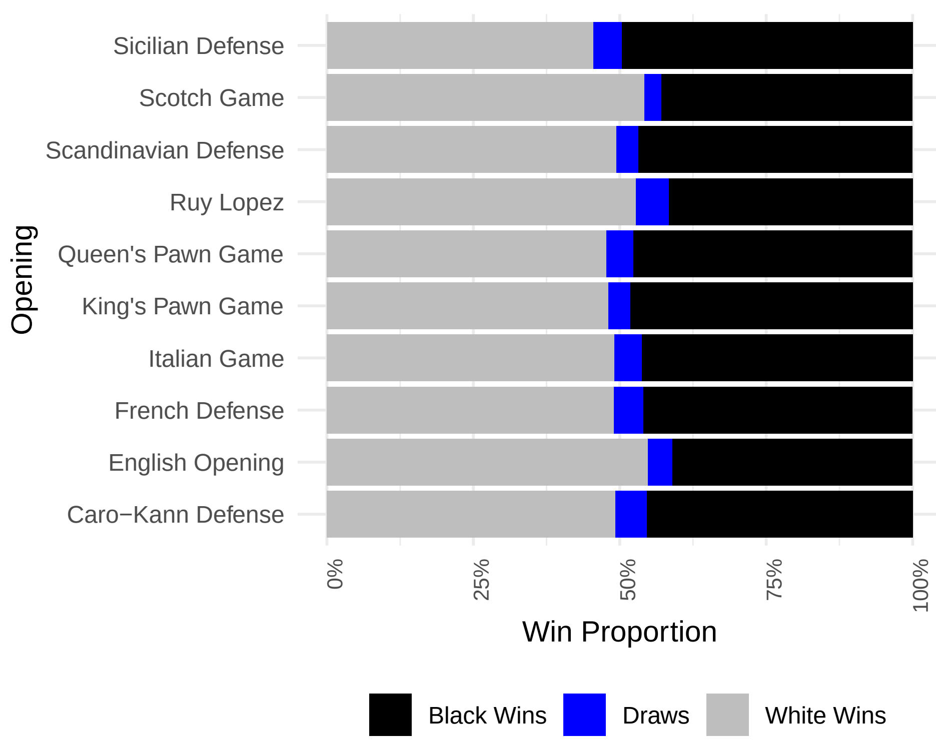

Figure 3.5: Outcome by Popular Openings

Figures 3.4 and 3.5 show the most played openings and the win proportions of those openings respectively. The opening names were given in the dataset with their variations (e.g. Queen's Gambit: Declined) where I only want the base opening name. I therefore separated the names and then dropped the variation. The Sicilian defense being the most popular is interesting as it is initiated by black as a response to white's first move of e4 who then may need to change their strategy in the opening. The opening is usually initiated by white and black has to adjust their strategy. The power of the Sicilian defense is show in figure 3.5 with the largest proportion of black winning out of all the common openings at 50%. The reason for this is that it partially eliminates white's first mover advantage as they have to respond in a way that may not have been their plan. These figures not only highlight the importance of the opening as one can gain a significant advantage that can carry through the game but also show that some openings favour a colour.

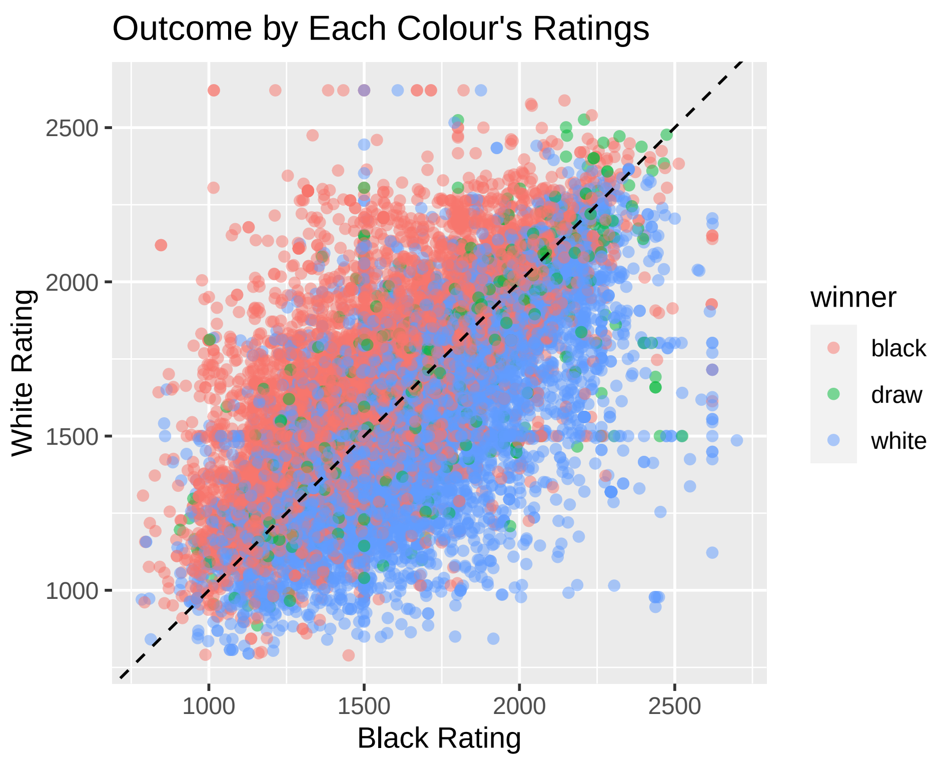

Figure 3.6: Outcome and Ratings

Figure 3.5 shows the influence of the respective players ratings on the outcome. As expected players with the higher rating will win but around the 45 degree line there are some exceptions. One would assume that there would be more white wins at the same or similar rating but at different levels this seems to change. Between 1000 and 1200 white seems to win more than black but between 1300 and 1700 it appears that black wins more than white and above 1700 it appears to be even with the majority of draws occurring above 2000. This highlights that the determinants of the outcomes of games may change at different ratings. Figure 3.6 shows the distribution of the ratings. Both colours have nearly identical distrbutions which makes sense as a player will have to play with both colours. The median of black is 1562 and the median of white is 1567. This distinction is used later to assess whether it is easier to predict lower or higher rated games. This section has provided some key insights of the variables of interest and show that they can have an influence on the outcome.

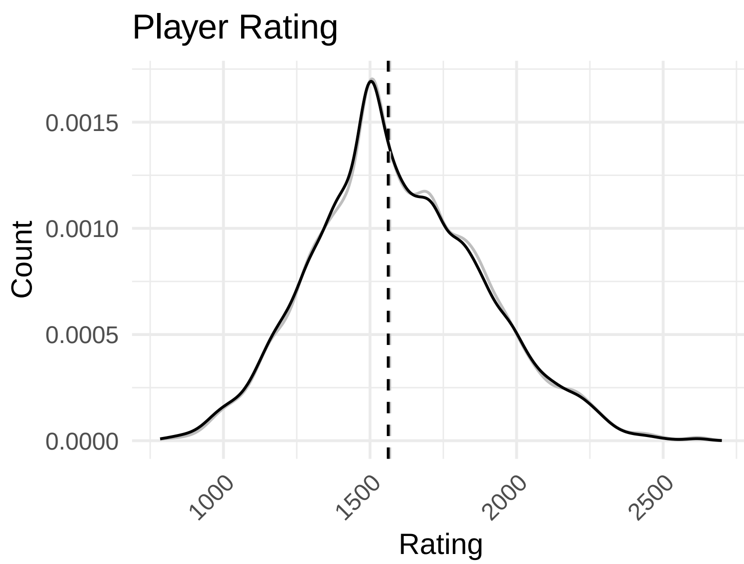

Figure 3.7: Distribution of Ratings

4. Methodology

5. Results

5.1. Full Sample

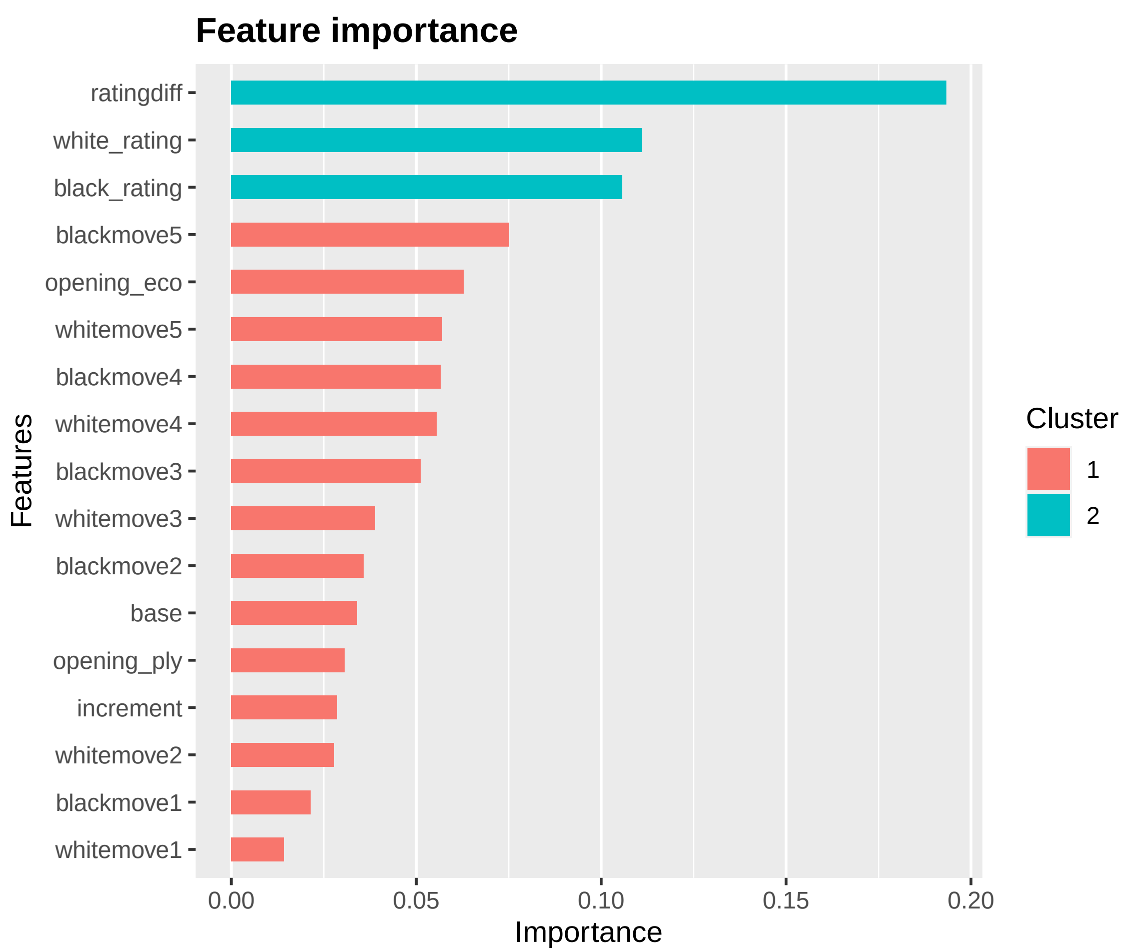

Figure 5.1: Importance of Features for Untuned Model: Full Sample

	eta	depth	weight	subsample	colsample	gamma	lambda	alpha	rmse	trees
1	0.01	3.00	3.00	0.50	0.50	0.00	1.00	0.00	0.80	874.00

Table 5.1: Hypergrid Full Sample

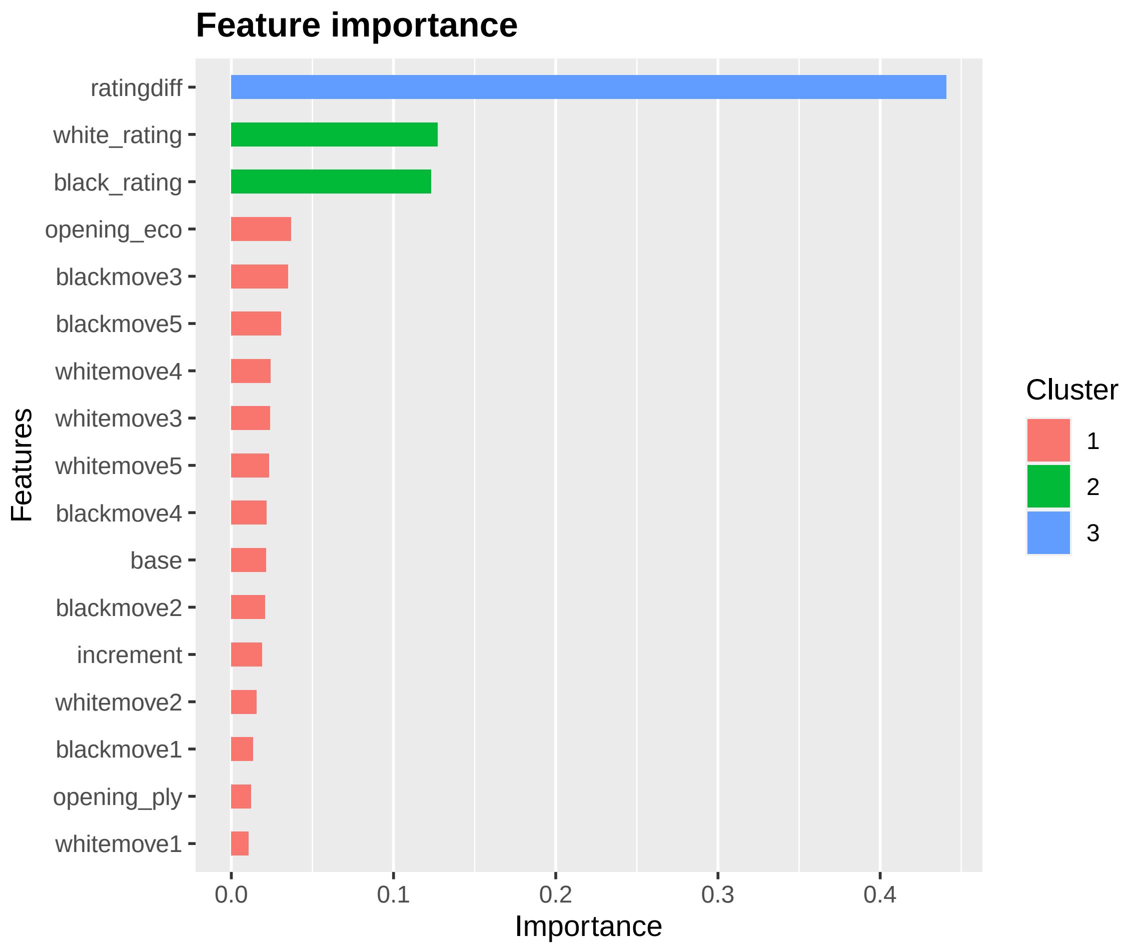

Figure 5.2: Importance of Features for Tuned Model: Full Sample

The initial accuracy after tuning is 62% on the test sample and 100% for the training sample

	Sensitivity	Specificity
Class: black	0.61	0.66
Class: draw	0.12	1.00
Class: white	0.67	0.61

Table 5.2: Accuracy Full Sample: Tuned Model

After tuning the model has an accuracy of 62%

5.2. Sub-samble by ELO

5.2.1. Bottom Half

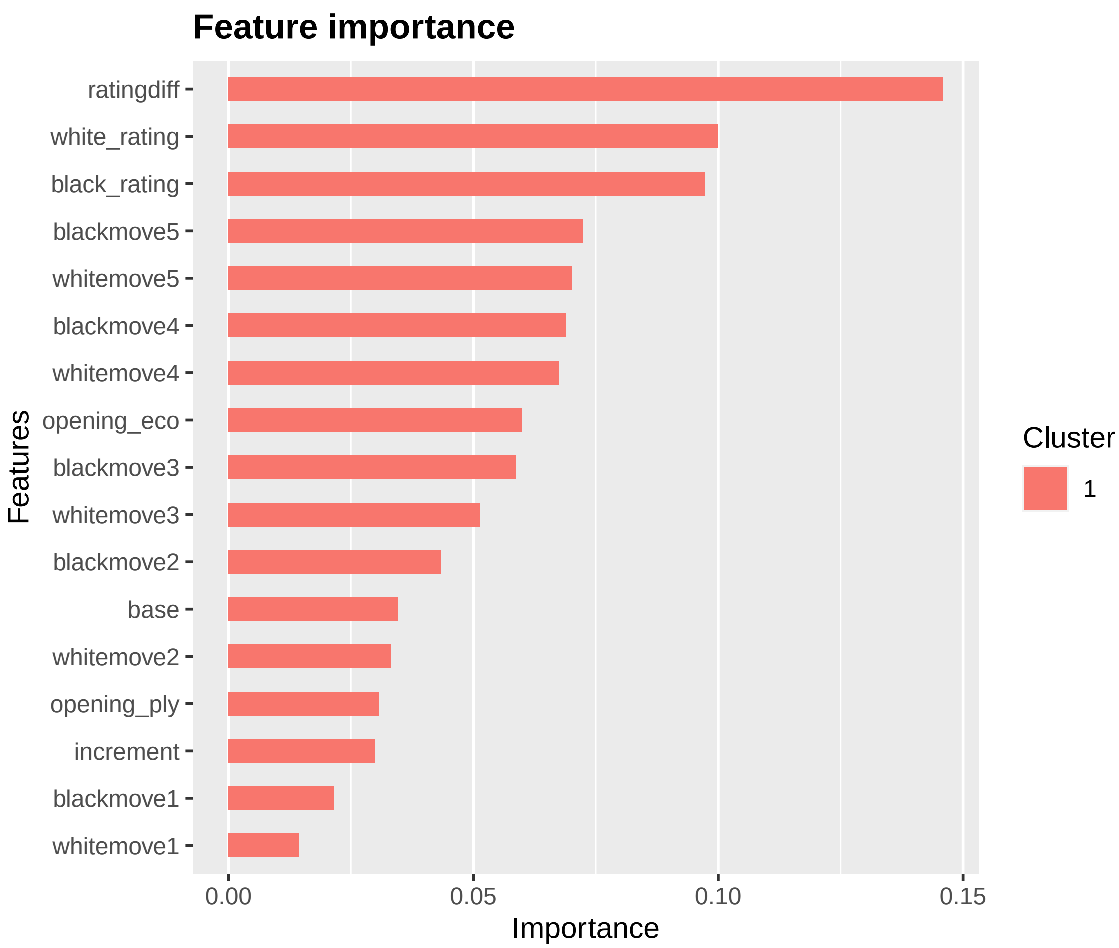

Figure 5.3: Feature Importance Untuned Model: Bottom Half

The initial accuracy before tuning is 59% on the test sample and 100% for the training sample

	eta	depth	weight	subsample	colsample	gamma	lambda	alpha	rmse	trees
1	0.01	3.00	3.00	0.50	0.50	1.00	1.00	0.00	0.80	603.00

Table 5.3: Hypergrid Bottom Half

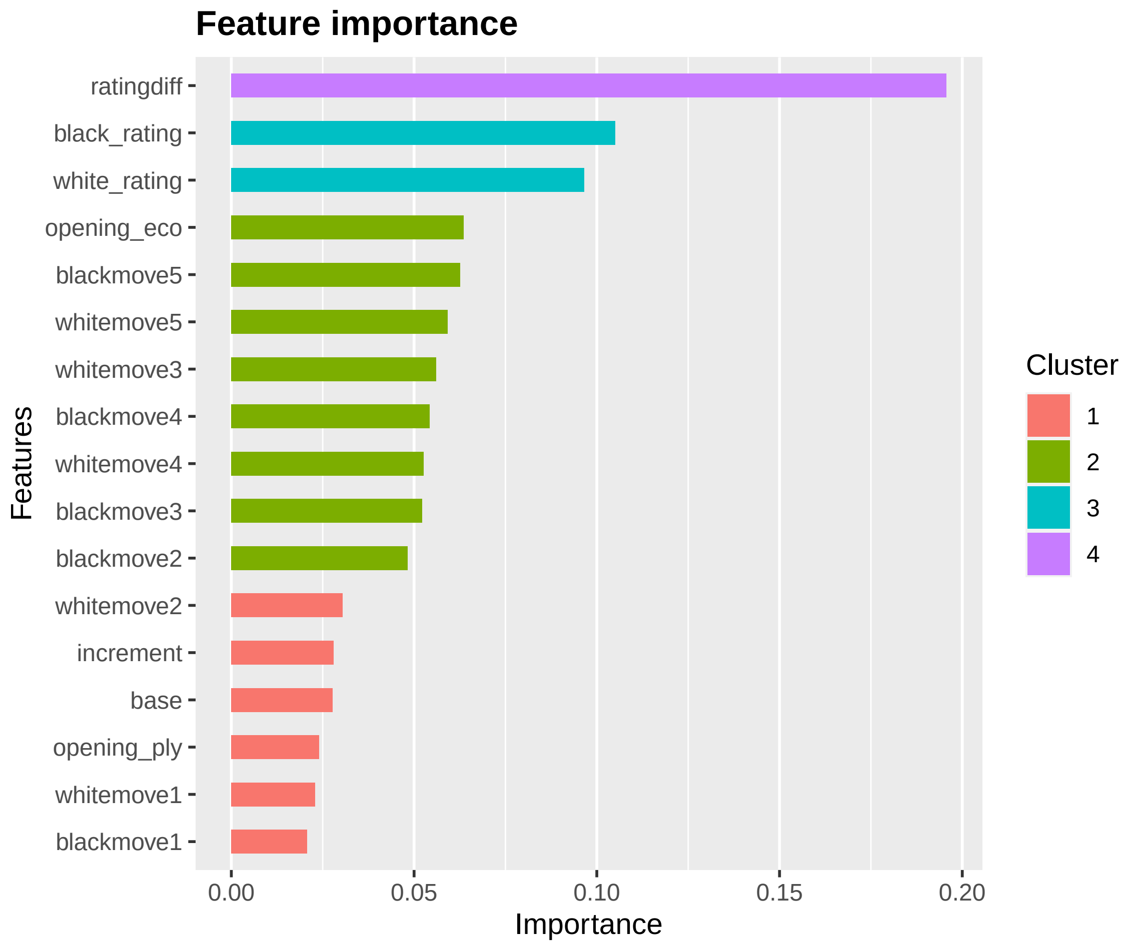

Figure 5.4: Feature Importance Tuned Model: Bottom Half

	Sensitivity	Specificity
Class: black	0.55	0.65
Class: draw	0.10	1.00
Class: white	0.65	0.56

Table 5.4: Accuracy Bottom Sample: Tuned Model

The initial accuracy after tuning is 59% on the test sample and 100% for the training sample

5.2.2. Top Half

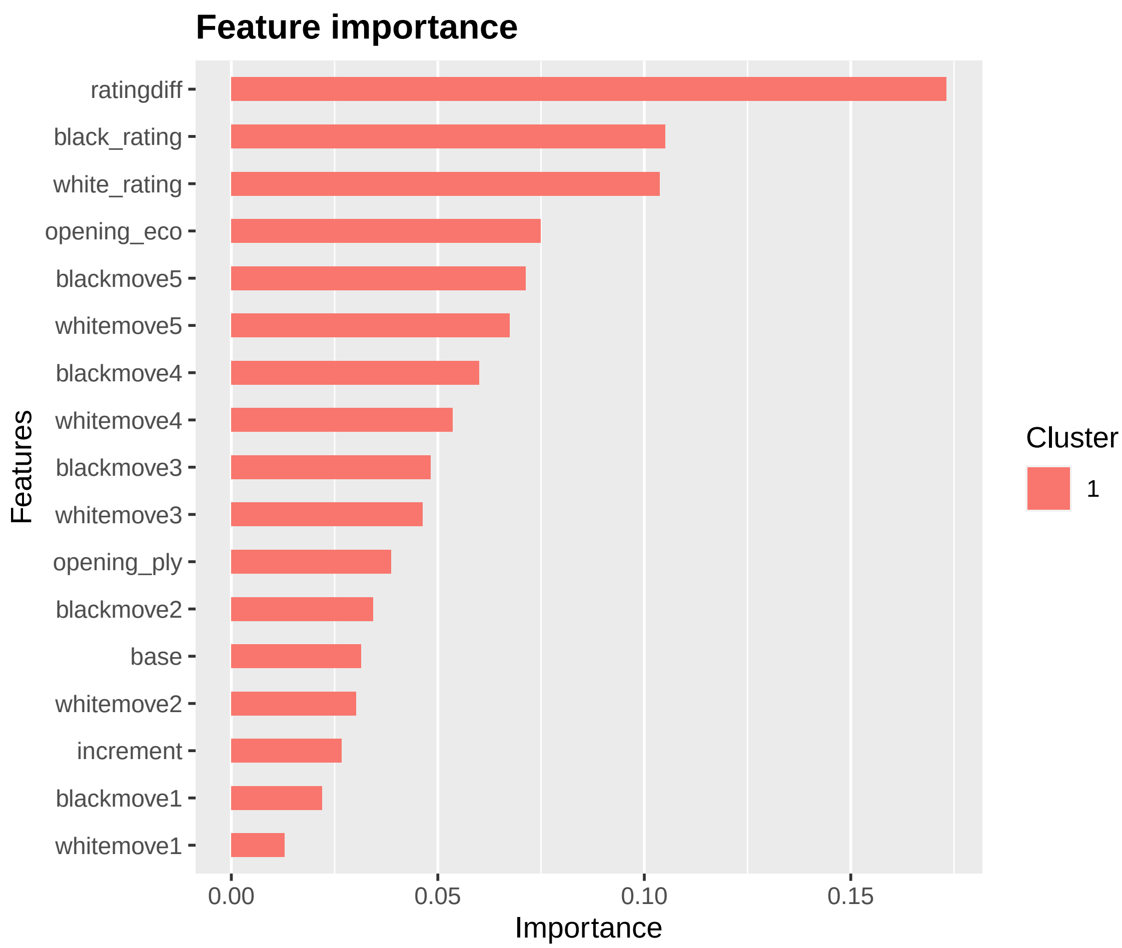

Figure 5.5: Feature Importance Untuned Model: Top Half

The initial accuracy before tuning is 58% on the test sample and 100% for the training sample

	eta	depth	weight	subsample	colsample	gamma	lambda	alpha	rmse	trees
1	0.01	3.00	3.00	0.50	0.50	1.00	0.00	0.00	0.81	676.00

Table 5.5: Hypergrid Top Half

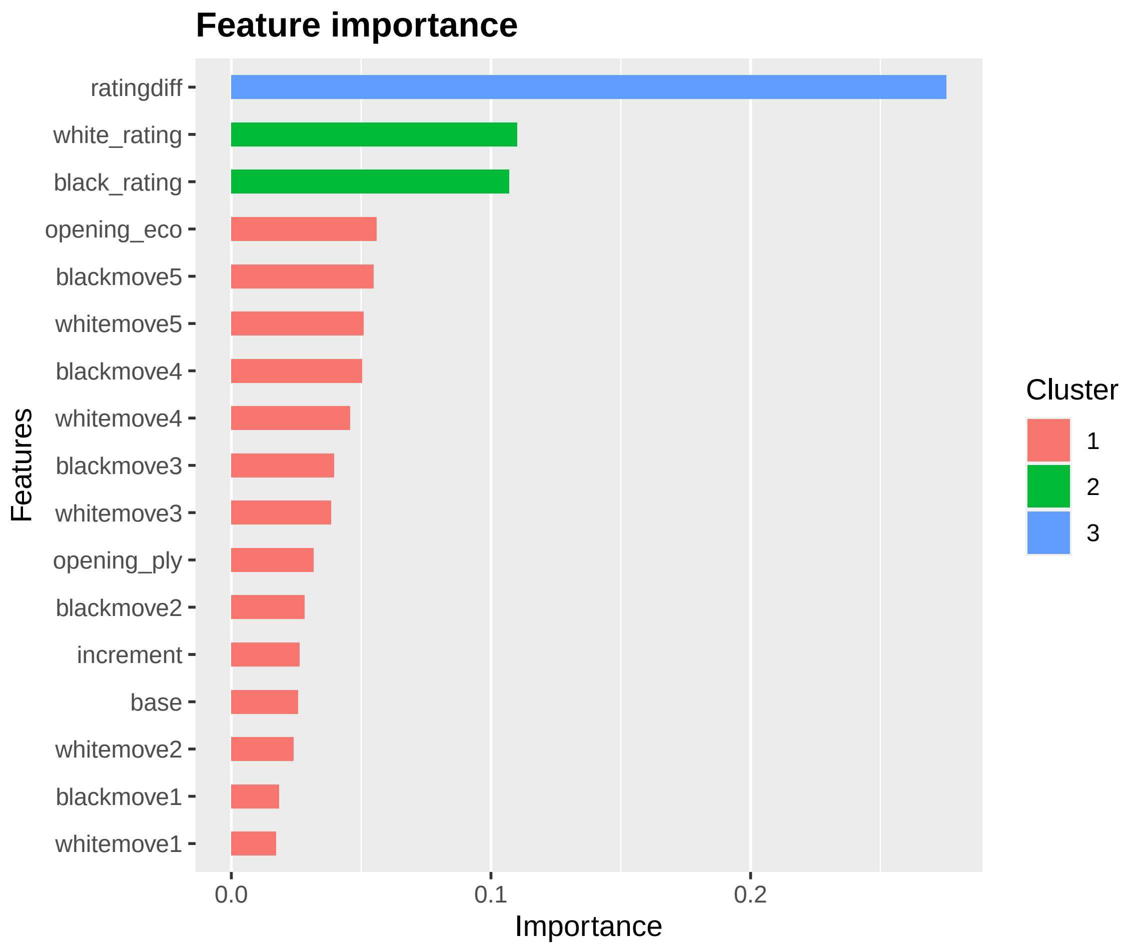

Figure 5.6: Feature Importance Tuned Model: Top Half

	Sensitivity	Specificity
Class: black	0.58	0.64
Class: draw	0.11	0.99
Class: white	0.64	0.57

Table 5.6: Accuracy Top Sample: Tuned Model

The initial accuracy after tuning is 58% on the test sample and 100% for the training sample

6. Conclusion

7. References