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. You can find my code on my Github page: https://github.com/wjwilliams/MLPROJ/tree/main/WriteUp

Table of Contents

Email address: 21691126@sun.ac.za (Wesley Williams)

Special Thanks to Ruan Geldenhuys for his help.

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

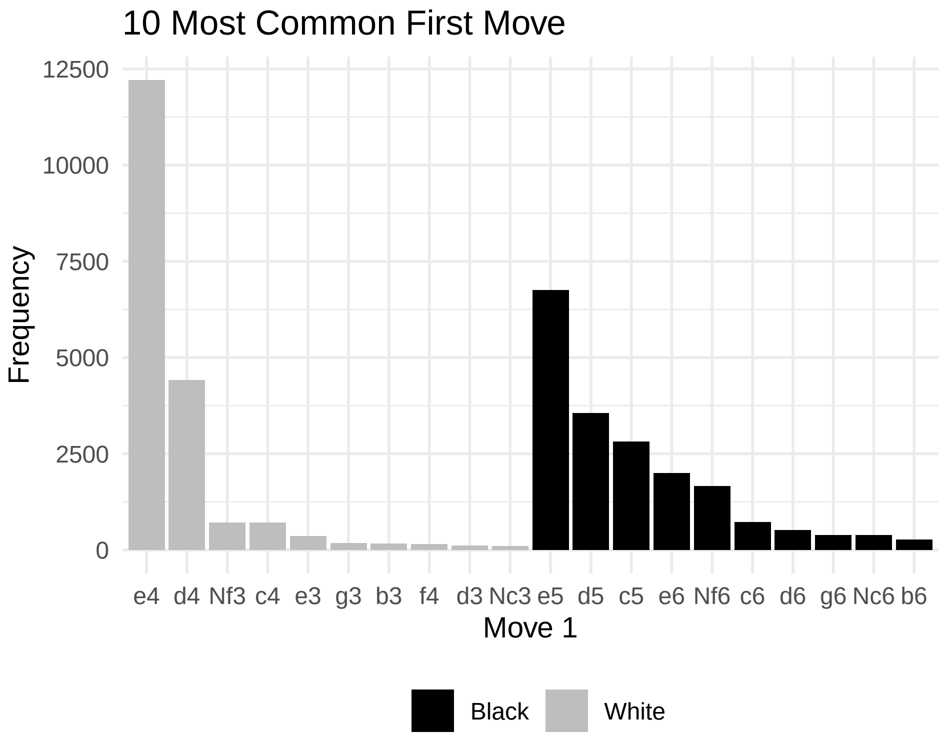

Figure 3.1: Most Common Opening Moves by Colour

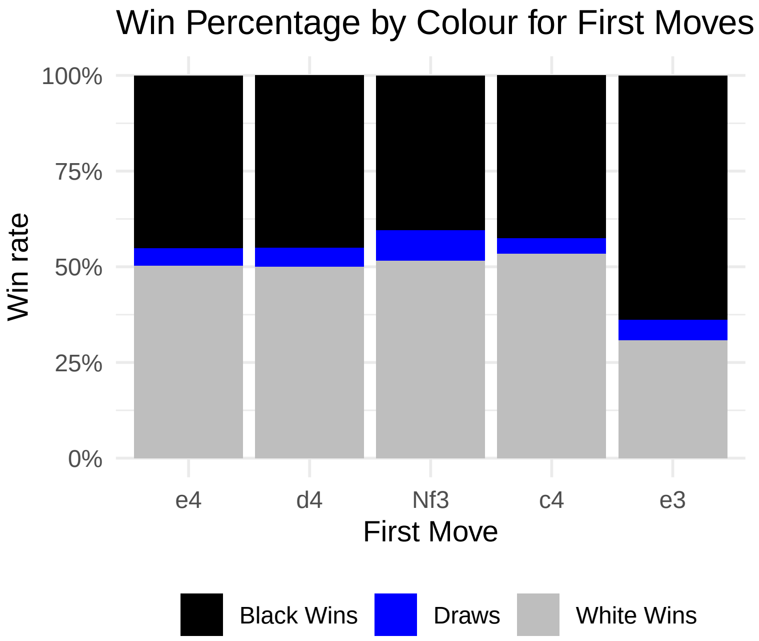

Figure 3.2: Proportions of Outcomes by White First Move

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.

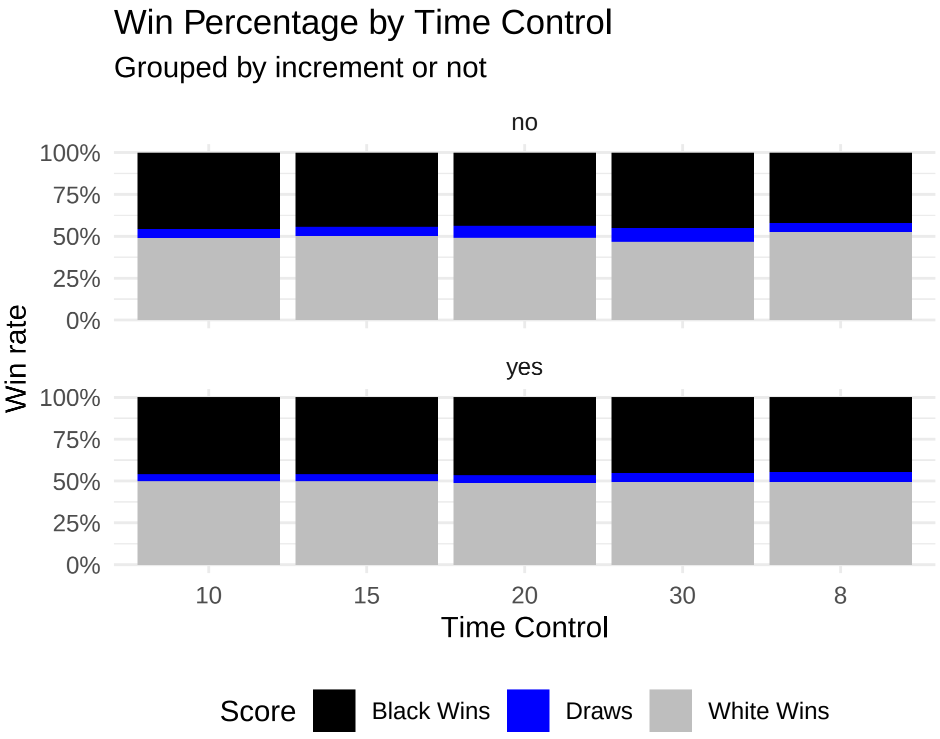

Figure 3.3: Outcomes by Time Controls

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.

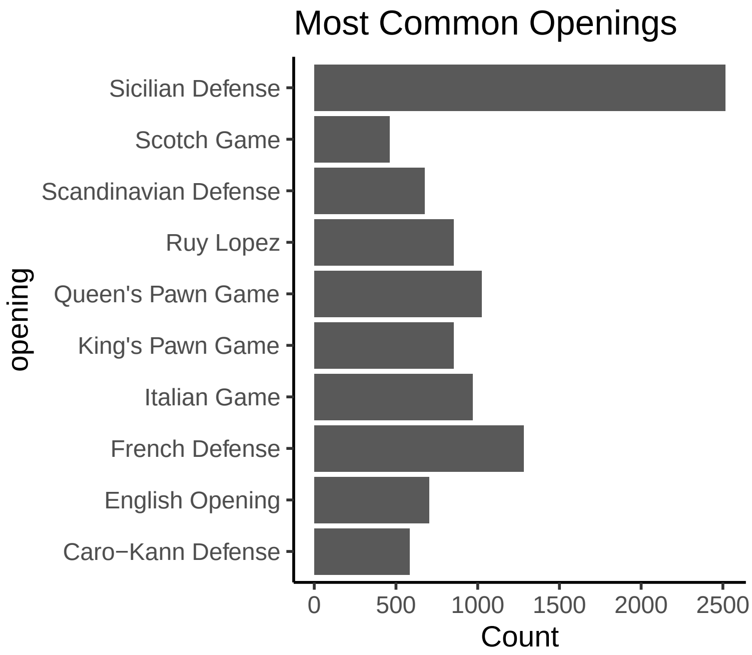

Figure 3.4: Most Popular Openings

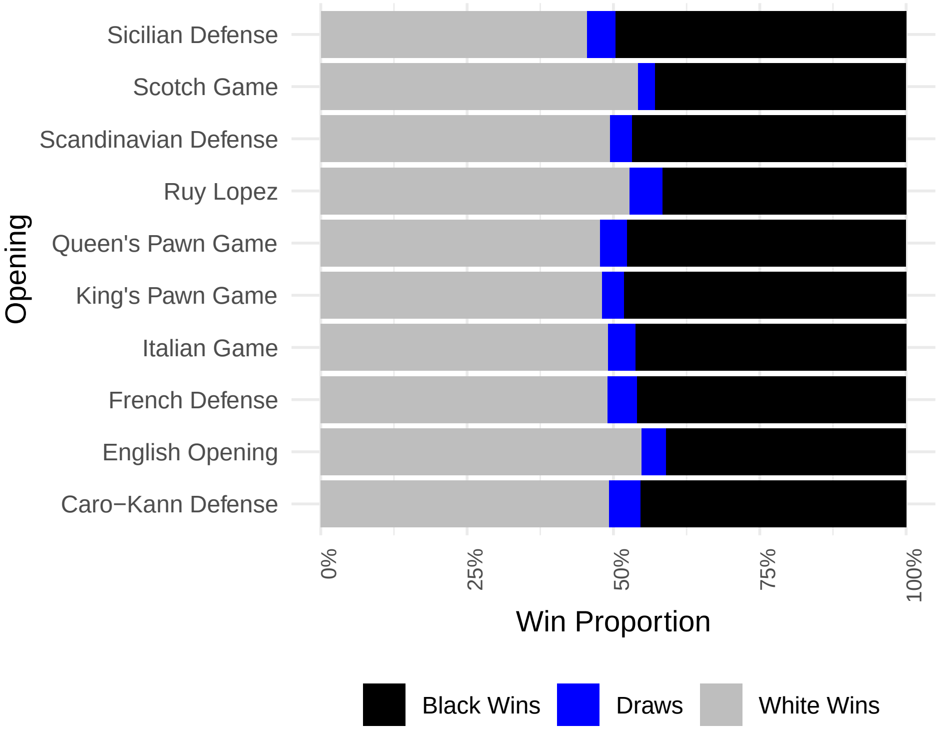

Figure 3.5: Outcome by Popular Openings

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.

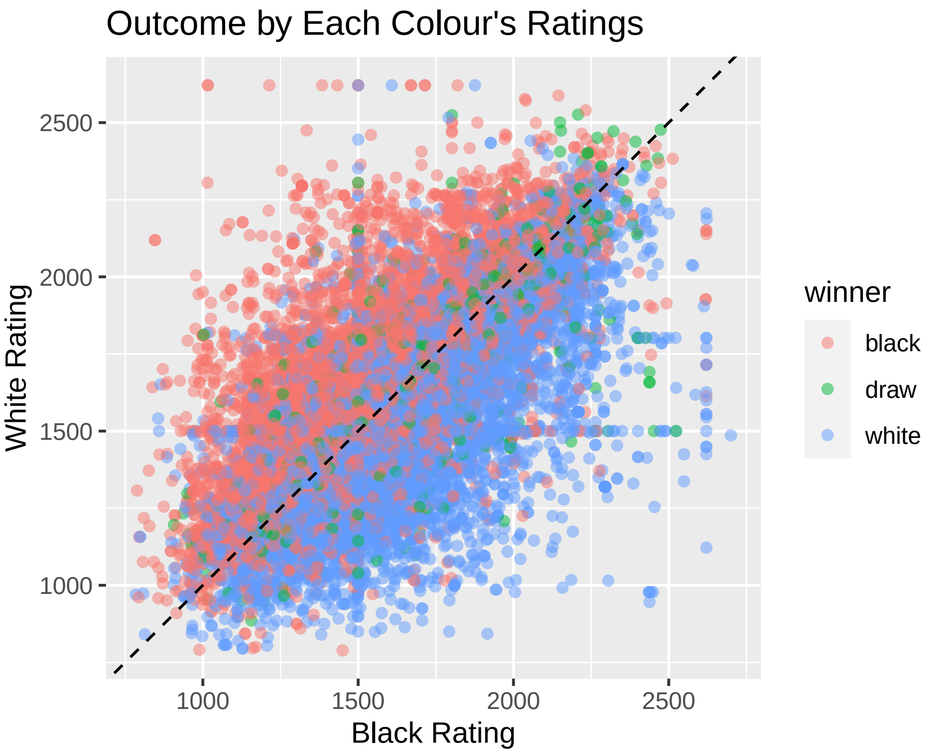

Figure 3.6: Outcome and Ratings

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.

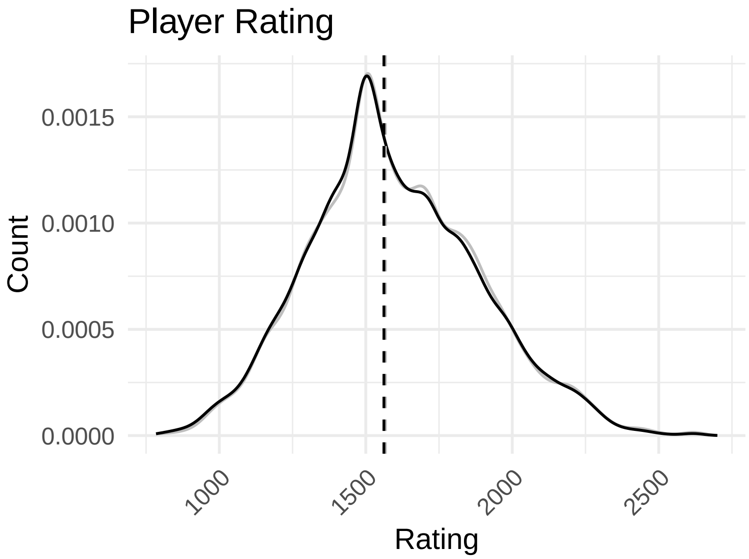

Figure 3.7: Distribution of Ratings

4. Methodology

I use a gradient boosting model with the winner of the game as the target and the features being both players ratings, their rating difference, the opening used, the number of moves played in the opening, the time controls and the first five moves of each player. I stratified the sampling by the opening to ensure that as many openings were included in both the training and test split. This also means that the moves were stratified as the moves correspond to the opening. Gradient boosting employs a sequence of shallow trees that learn from the previous tree and makes improvements. I used the XGBOOST package and due to the target variable not being binary and because I had more than 53 categories for the opening feature which the random forrest package in R could not handle. I used the “multi:softmax” objective function to be able to predict three types of outcomes and the evaluation metric was a multiclass log loss which evaluates the logarithm of the predicted probabilities for each class and aggregates them across all instances and is calculated as follows.

$$\text{mlogloss} = -\frac{1}{n}\sum\left[y \cdot \log(p) + (1-y)\cdot\log(1-p)\right] \tag{4.1}$$

The methods used follow Boehmke & Greenwell (2019). I first trained and tested the model with default parameters before employing a grid search that uses cross validation to determine the optimal value of the learning rate (η), the minimum loss reduction(γ), the L2 regularization term (λ) and the number of trees. I then used the results of the grid search to hyper parameter tune the model. I then used the importance of each factor and a confusion matrix to interpret and assess the model. I then subset the data using the median ratings and used same method and steps to assess if it is easier or more difficult to predict stronger or weaker players.

5. Results

This section presents the results of all six models. I present the importance of the features of the models with the default parameters to see how the importance changes after hyperparameter tuning. For the tuned models I also present the sensitivity and specificity calculated through a confusion matrix.

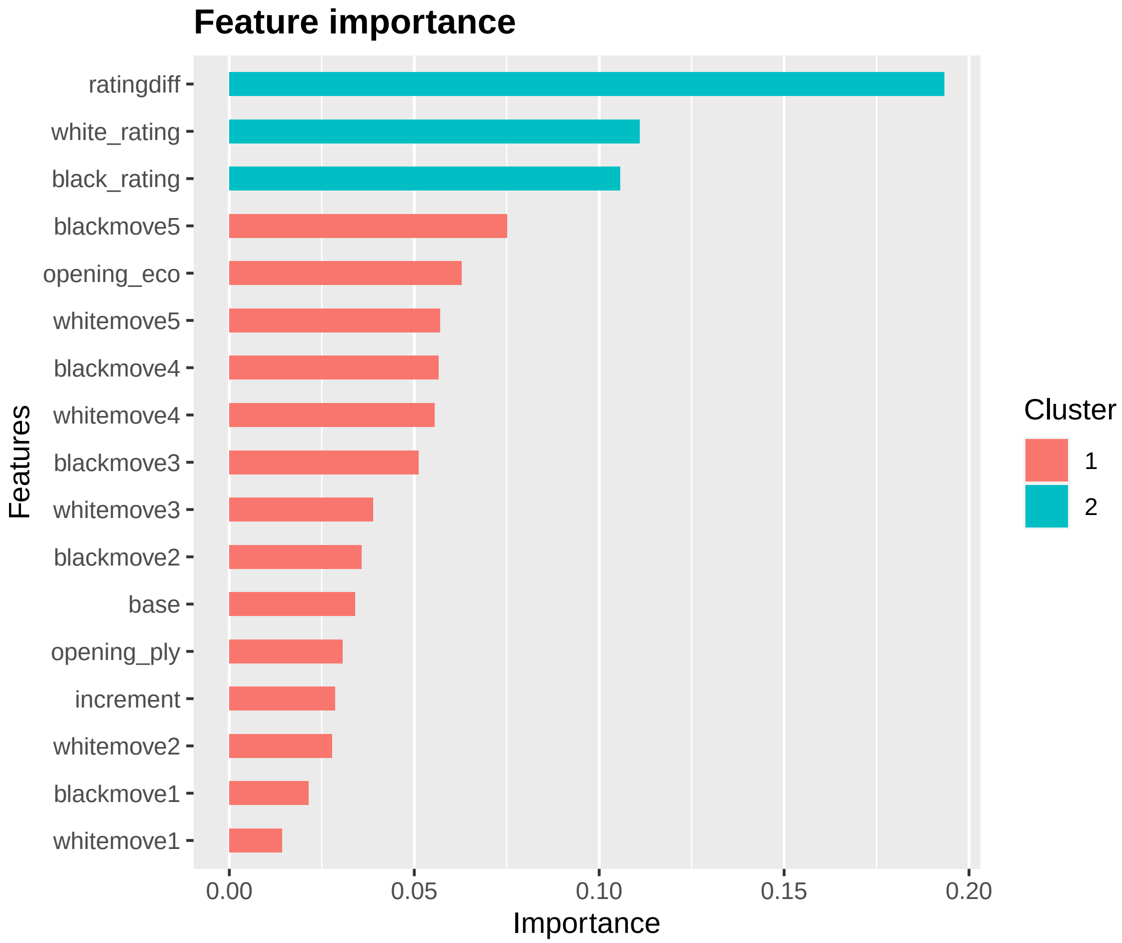

Figure 5.1: Importance of Features for Untuned Model: Full Sample

Figure 5.1 shows that, as expected, ratings features are the most important features, combined they account for 41% and are the three most important features of the model. The opening that is played is only the fifth most important factor at only 6%. The results also show that the moves increase in importance with both white and black's first moves being the least important features and every subsequent move becomes more important. Black's moves are also shown to be more important to the model than white's moves with black's fifth move being the fourth most important feature at 8% which is interesting as it means it is more important than the opening. The model has an accuracy of 100% on the training set and just 60% on the testing set so it may be over fitted. I then used hyper parameter tuning to attempt to address this issue. Table 5.1 below presents the most optimal parameters provided by the grid search. The results suggest the learning rate (η) should be reduced from 0.1 to 0.01. The parameters α, γ

and λ do not change and are still zero. The tree depth should be reduced from 6 to 3, which should also help solve the issue of over fitting. These parameters are used to obtain the final model and the results are presented below.

	η	Depth	Weight	Subsample	Colsample	γ	λ	α	RMSE	Trees
1	0.01	3.00	3.00	0.50	0.50	0.00	0.00	0.00	0.80	874.00

Table 5.1: Hypergrid Full Sample

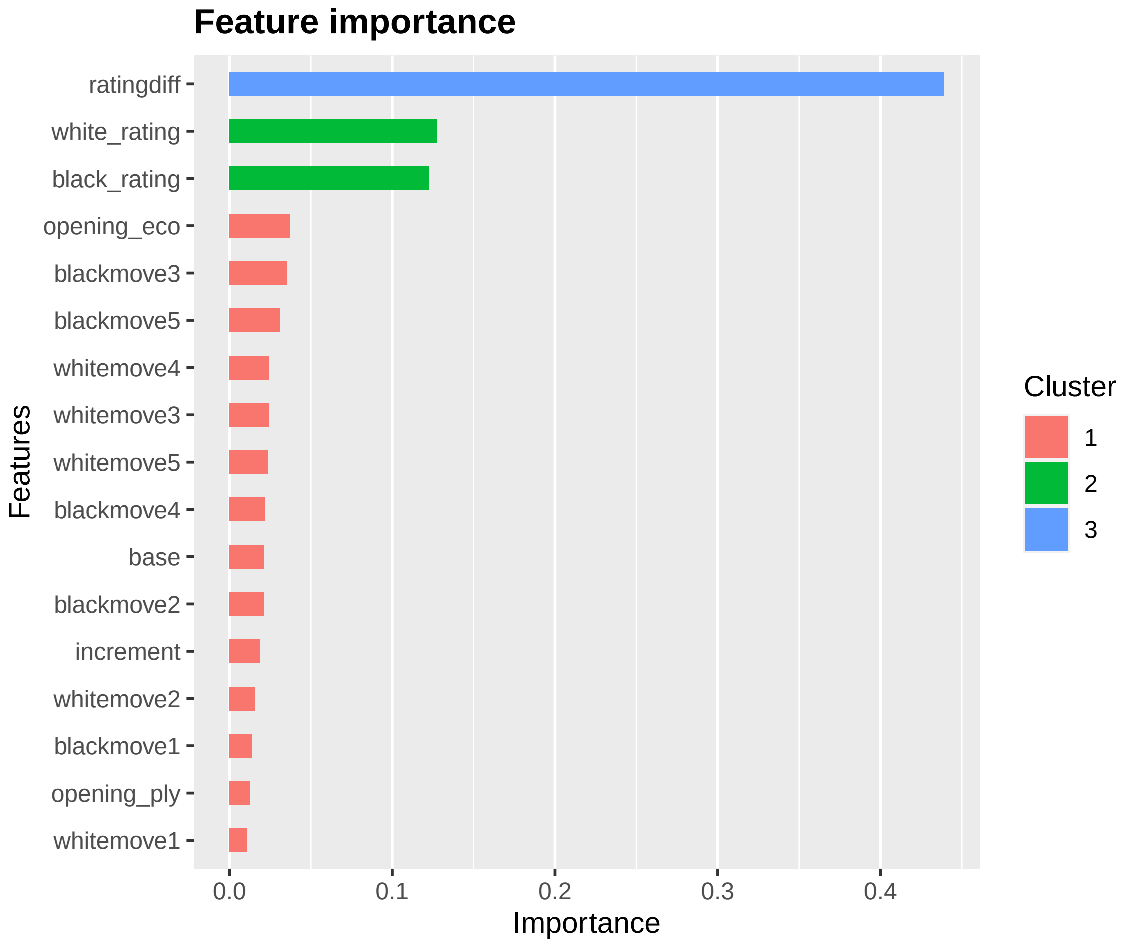

Figure 5.2: Importance of Features for Tuned Model: Full Sample

	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

Figure 5.2 and table 5.2 present the results of the tuned model. The tuning of parameters has improved the accuracy on the testing set to 61.66% but again the accuracy on the training set is still 99.98% which means that even after tuning the model still suffers from over fitting. The importance of the rating features increased to 68.92%. The lower learning rate results in better accuracy but it seems that the increased accuracy is driven by the ratings features rather than moves. The opening now becomes the most important feature after ratings which shows that it can influence the game but its importance decreases to just 3.73%. The time controls also become relatively more important after tuning but similarly to the opening it percentage of importance also decreases.

Table 5.2 is an extract of the model's confusion matrix where sensitivity is the percentage of white wins accurately predicted and specificity is the percentage of white not winning accurately predicted. The table shows that the model predicts white the best with a sensitivity and specificity of 66.67% and 60.92% respectively. It predicts black almost as well with a sensitivity and specificity of 61% and 66.13% respectively, unlike for white the specificity is larger than the sensitivity meaning it predicts black not winning more accurately than black winning. The model struggles the most to predict a draw with a sensitivity of just 11.74%. The accuracy is still nearly double that of just guessing which would be 33%.

5.2. Sub-samble by ELO

The sample was then subset using the medians of ratings calculated in the exploratory data analysis section. I use the same methodology as above for both subsets of the sample.

5.2.1. Bottom Half

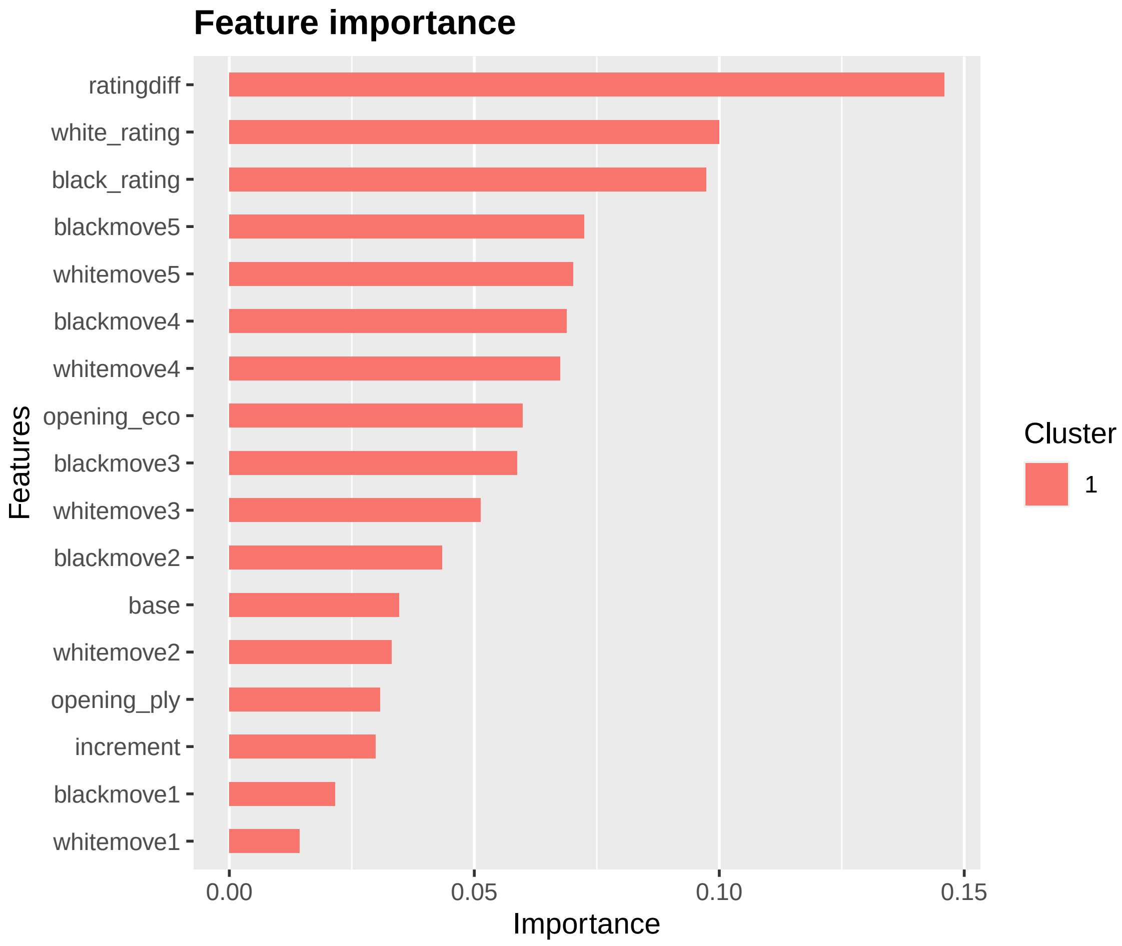

Figure 5.3: Feature Importance Untuned Model: Bottom Half

Figure 5.3 shows the feature importance of the untuned model which has an accuracy of 58% which is 2% lower than the full sample. Again ratings feasures are the most important and combined contribute 34.33% to the model. The relative importance of openings is much lower than in the full sample as it is only the eighth most important factor now with both white and black's fourth and fifth moves contributing more to the model. The results of the grid search are presented below in table 5.3 and the results suggest the same parameters as for the full sample with the exception of λ and γ which are now both 1. The results of the hyper parameter tuned model are presented below.

	η	Depth	Weight	Subsample	Colsample	γ	λ	α	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

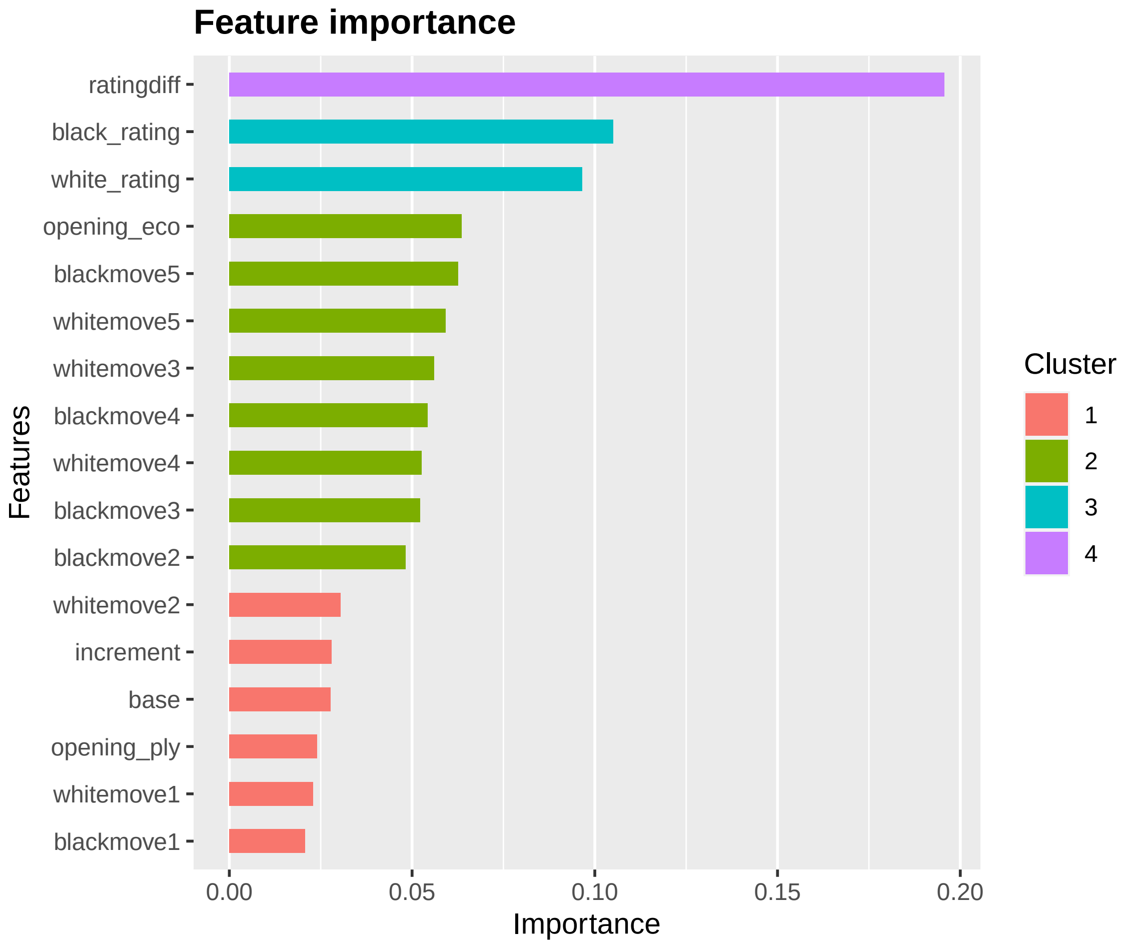

Figure 5.4: Feature Importance Tuned Model: Bottom Half

	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 tuned model still performs worse than the full sample even after tuning with an accuracy of only 58.61%. This makes sense as weaker players will not stick to rigid openings as they do not know much chess theory. This makes them more unpredictable and therefore it is expected that the model performs worse. The model still seems to be over fitted as the accuracy on the training set is still 99.98%.

Figure 5.4 presents the importance of features in the tuned model. The decrease in the learning rate parameter does not increase the importance of the ratings features as much as the tuned model for the full sample as it only increases to 39.72%. The type of opening used now increases in both relative and absolute importance. It is now the fourth most important feature and its importance increased to 6.36% from 6%. The decrease in the learning rate helped to identify that even for weaker players the opening used is the most important feature after rating features.

Table 5.4 breaks down the models accuracy and again it best predicts white, then black and still struggles with predicting a draw. All values of sensitivity and specificity have decreased compared to the full sample so it can be concluded isolating the weaker players does not improve the models accuracy.

5.2.2. Top Half

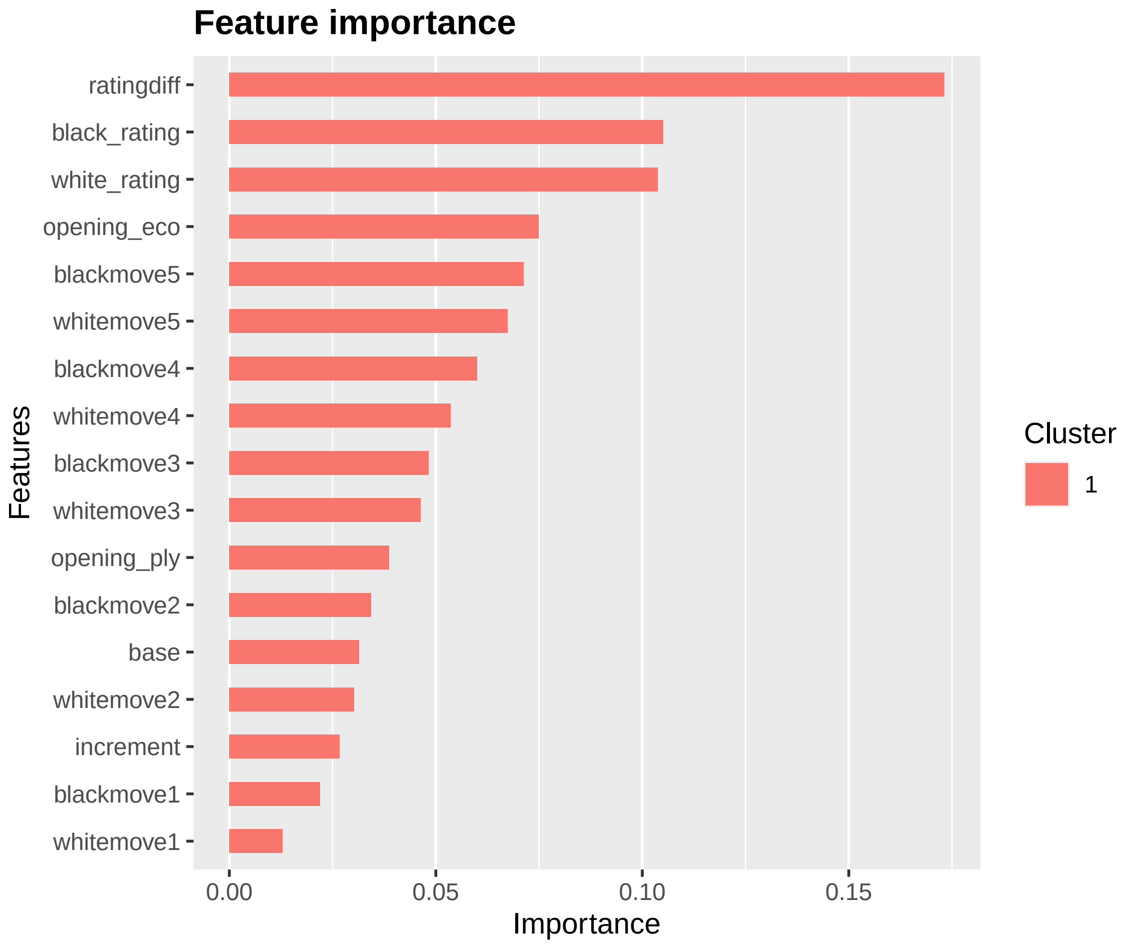

Figure 5.5: Feature Importance Untuned Model: Top Half

Figure 5.3 shows the feature importance of the untuned model which has an accuracy of 57.68% which is 2% lower than the full sample. This is interesting as I would have expected the stronger players to be easier to predict as they play more rigid chess and follow chess principles more stringently. Again ratings features are the most important and combined contribute 38.19% to the model. The importance of the opening for the stronger players is apparent already as it is the fourth most important factor at 7.49%. The ordering of the importance of the rest of the does not change compared to both other samples. Table 5.5 shows the results of the grid search and the parameter values are the same as for the bottom sample with the exception of λ which is 0, similar to the full sample. The parameters from the grid search are then used to hyper parameter tune the model and the results are presented below.

	η	Depth	Weight	Subsample	Colsample	γ	λ	α	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

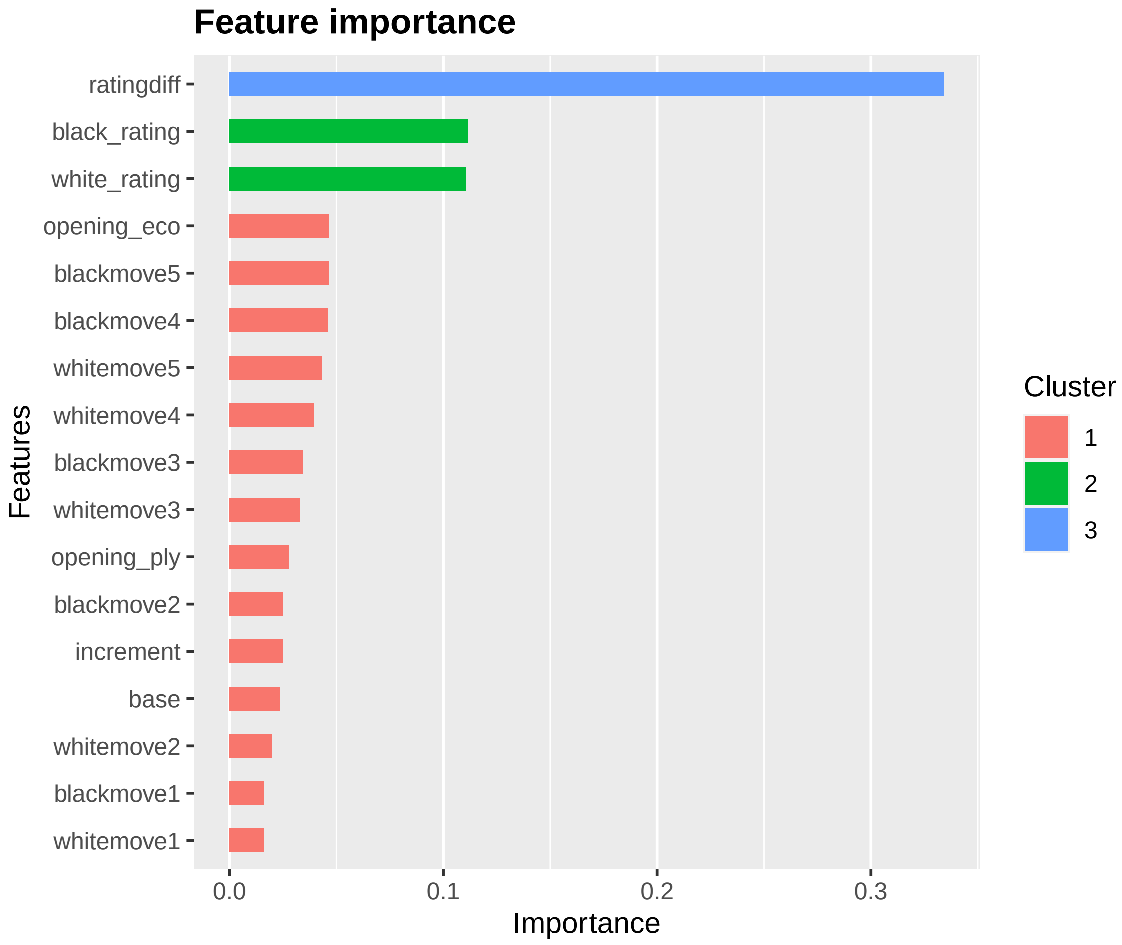

Figure 5.6: Feature Importance Tuned Model: Top Half

	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 hyper parameter tuning does not increase the overall accuracy by much and it is still

57.68% on the test set and 99.98% for the training set. This is puzzling as not only is there no improvement, it also worse than for the bottom half sample. Over fitting is still an issue, the hyper parameter tuning does not seem to solve. Figure 5.6 shows the importance of the features and the importance of the ratings features increased to 55.67% which is much larger than for the bottom sample. Simalarly the importance of the opening for stronger players is only 4.66%.

Table 5.6 presents the results from the confusion matrix. It shows that the model predicts black winning and losing more accurately than for the weaker players but predicts white winning and losing less accurately than for the weaker players. Its accuracy and predicting the game to be a draw also decreases. While my assumption going in was that it would be easier to predict the outcome for stronger players was incorrect it highlights how complex the game of chess is and that stronger players seem to rely less on the openings than weaker players.

6. Conclusion

Chess is a complicated game and trying to predict the outcome based just on the opening does not work. My results suffer from over fitting which is a large limitation of the investigation. The results show that the type of opening used is a determinant of the outcome but is minuscule compared to the ratings features. The importance of the opening also seems to decrease for players that are stronger, indicating that they rely less on specific opening strategies and may adapt their gameplay based on the opponent's moves and overall game dynamics. Subsetting the data based on player ratings, either by considering the bottom or top half, resulted in models that were less accurate compared to the analysis on the full sample. This suggests that having a diverse range of player ratings in the dataset improves the model's accuracy, likely due to the increased variability and complexity introduced by players of different skill levels.

In conclusion, while the opening moves do play a role in determining the outcome of a chess game, they are overshadowed by the ratings features of the players. To build a more accurate prediction model, it is important to consider a wide range of player ratings and incorporate other relevant features beyond just the opening moves. Chess remains a challenging game to predict, highlighting the intricate nature of the game and the need for comprehensive models that encompass various aspects of player skill and strategy.

References

Boehmke, B. & Greenwell, B.M. 2019. *Hands-on machine learning with r*. CRC press.