Chess Analytics

Stephanie Tanasia Saputra
40199284

I. Introduction

Chess is a classic, timeless board game whose origin is blurry and enjoyed by many nowadays. A game called *chaturanga* originated in India around the 6th century CE and has an uncanny resemblance to it [1]. The Sanskrit word *chaturanga* refers to the ancient army divisions of infantry, cavalry, elephantry, and chariotry. The pieces, rules, and objective are similar to modern chess; White moves first and the player needs to checkmate or reduce the opposition to just the Raja (king) of the opponent.

The game was played on an 8×8 uncheckered board. After becoming popular in India, the simple yet challenging game infiltrated Iran where it became a tool for teaching young princes of Persia [2]. Chess then gained popularity among the Muslims, which later expanded to Spain and the rest of Southern Europe. By around 1500 CE, the game had largely taken on its modern form [3].

The modern chess we play today came about in the 19th century. It shortly turned competitive after a series of tournaments was first held in 1834 between La Bourdonnais and McDonnell [3]. The first official World Chess Championship was held in 1886 followed by a chess *Golden Age*- births of many chess giants in the 1920s. Nowadays, chess has a wider player-base than ever, to online gaming platforms.

The best chess players in the world can earn huge amounts of money by winning chess tournaments. Magnus Carlsen, a grandmaster and current world champion, earns an estimated of $750,000 per year from winning chess competitions [4]. Of course, this motivates amateurs to train intensively for the prize incentives. In addition to the fiscal incentives, pop culture also influences the interest in chess. Play Magnus Group, which runs some of the leading online chess platforms mentioned that the interest in chess skyrocketed by 300% following the screening of Netflix's *The Queen's Gambit* in 2020 [5].

Online chess platforms are undoubtedly the easiest way for one to learn chess nowadays. In addition to playing against an opponent in real-time, the users are able to analyze all of the moves that they did in a game **after** the match has ended. Usually, the moves are ranked in a categorical group: brilliant, great, best, excellent, good, mistake, blunder, missed win, etc [6]. In other words, these online platforms are equipped with AI tools that enable users to learn from their past matches easily.

Using such tools to improve performance is precisely the purpose of sports analytics. The author found an archive of online chess games, Lichess that was scraped by Jolly in 2017 [7]. In joining to the chess craze, we would like to dive deeper into the opening moves of chess and see how it affects the game. In addition, the author would like to create a metric to analyze the winning probability throughout the game so that players can learn more from their moves.

The objective of this project is to:

- Present the descriptive statistics of the online chess games.
- Predict the winner based on the opening moves with logistic regression, k-NN classifier, and decision tree.
- Perform goodness-of-fit test with AUC and ROC curve to compare the results obtained from each method.
- Create expected win rate based on game history using *Stockfish*.

II. Game Rules and Standard Notation

Chess is played on an 8x8 checkered board, with 16 white pieces and 16 black pieces as seen below. The columns are called 'files' and the rows are referred to as 'ranks'. A straight line of squares of the same color, running from one edge of the board to an adjacent edge, is called a 'diagonal' [8]. White pieces are **always** located in rows 1 and 2 while Black is in rows 7 and 8.

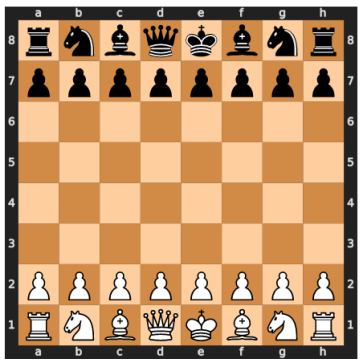

Figure 1: Starting Position of Chess Board

Pieces	Moves
King ♔, K	Any adjoining square; may also castle.
Queen ♕, Q	Any square along the file, the rank/diagonal on which it stands.
Rook ♖, R	Any square along the file or the rank on which it stands.
Bishop ♗, B	Any square along a diagonal on which it stands.
Knight ♘, N	Move to one of the squares nearest to that on which it stands but not on the same rank, file or diagonal.
Pawn ♙,	Move forward to the square immediately in front of it on the same file. On its first move, it may advance two squares and it captures pieces diagonally.

Table I: Chess Pieces

The pieces and their moves are shown in Table I [8]. The moves in a match are written down in standard algebraic notation using the board and pieces notation. For example, d5 is a move for a pawn to d5 and Nc3 is a move for a knight to c3. When a piece is captured, denote "attacking piece" x "captured piece's location". For instance, Qxe5 means the queen captured the piece on e5.

Chess pieces have values, but these do not determine who wins the game. The value of pieces simply act as a guide for the player to determine what piece should be traded for another. Q is worth 9 points, R is 5 points, B and N are 3 points, and the pawn is a point. As an example: If all other

things are equal, capturing three pawns at the cost of one's bishop should be considered a fair trade for either player. The king doesn't have any value because capturing it is the one and only win condition.[1]

Chess is all about board positions. Claude Shannon showed that there are about 10^{120} possible games and 10^{43} possible number of positions in his 1950 paper "Programming a Computer for Playing Chess"[9]. There are 2 pivotal positions in chess: check and checkmate. During a check, the opponent's king is threatened by a player's piece. With checkmate, the game is over since the opponent's king cannot escape the check. There is also the possibility of a draw where neither player can possibly checkmate the opponent's king. There are countless checkmate and check patterns in chess, making it complex and difficult to master.

III. Literature Review

Several past works have been concerned with chess game prediction using statistical analyses and machine learning.

Fan et al. suggested a system for predicting the outcome of chess matches [10]. They trained a classifier with the World Chess Federation (FIDE) Elo rating system, using a recent 11-year period game database as their training data. After building the classifier, they were able to achieve a success rate of 55.64% in predicting the chess matches of the same players in the following half year.

Another similar project by Thabtah et. al was published in 2020 where they investigated the impact of the Elo rating system on chess game results and its potential to be used as a predictor for predicting winners [11]. They also compared different features besides Elo and found that it was a very significant predictor for their dataset. In their study, they used decision tree, neural network, and Naïve Bayes to achieve the result.

An advanced study was done by Masud et. al in 2015 to predict the winner of a chess match while the game is still ongoing [12]. They used a dynamic ensemble-based classification technique where a classifier from previous chess games is created to predict a new ongoing game. In the end, they achieved 66% prediction accuracy and most of them were made with ≥ 9 moves before the game ended.

IV. Data Description

One of the most well-known chess websites in the world is Lichess, which is completely free to use. It is an open-source online server founded in 2010 which focuses on real-time game-play [13]. More than 5 million games are played daily by users, giving Lichess access to a mountain of chess game data.

Since Lichess is open-source, Jolly was able to scrape some data using the Lichess API five years ago, and the data is now accessible on Kaggle [7] [14]. The data set is available in .csv format which contains 20,058 rows and 16 columns.

A. Derivatives

Based on the objective, the author would need to create a first move for white and black pieces, by extracting it from the all moves variable. Next, the author would like to make use of the time increment since it affects the game and categorize it based on Lichess definition:

Name	Time Increment
Bullet	1+0, 2+1, 3+0
Blitz	3+2, 5+0, 5+3
Rapid	10+0, 10+5, 15+10
Classical	30+0, 30+20
Custom	Other than above

Table II: Time Increment Grouping

The time increment shown in Table II has a general notation: minutes + seconds. When a player is on a bullet 2+1 game, they are playing with a 2 minutes timer and after making a move, they will get 1 second added to their total time. This is why players can run out of time in chess and hence lose the game.

Lastly, it would be more convenient to analyze the game result based on the opening moves in terms of probability. By looking at the winning rate when the player used a certain move as their opening, we can assign the conditional probability to each opening move. Thus, instead of having the moves as a string, we will work with a numerical variable: the probability of winning with a certain opening move.

After cleaning and removing unused variables, the data set is ready to be explored.

V. Methodology

The author is interested in applying several statistical methods to this data set. Considering the nature of the response variable: win/lose (the draw was excluded), classification-based methods are considered. The predictors to be included in this project are the number of turns, White rating, Black rating, opening ply, time increment, and probabilities of winning based on the opening move.

Before performing any of the statistical methods, the author divided the clean data into training and test set. We chose the *Pareto Principle* [15], a very popular guiding principle. With this principle, 80% of the data goes into the training set and 20% goes into the test set. This leaves us with 15,272 observations to build the model and 3,818 observations to test the model.

A. Logistic Regression

The logistic regression uses regression methods to perform classification. First, assign:

$$Y = \begin{cases} 1, & \text{to all winners data points} \\ 0, & \text{to all losers data points.} \end{cases}$$

This provides numerical responses, which enables us to treat this as a regression. Given the predictors X_i, assume the responses Y_i is Bernoulli distributed [16].

To solve this, we need GLM regression model with logistic function $\eta(z) = \frac{e^z}{1+e^z}$ as the inverse link function [16]. Thus,

$$\begin{aligned} \mathbb{P}(Y_i = 1 | X_i = x_i) &= E[Y_i | X_i = x_i] \\ &= \eta(\tilde{x_i}^T \beta^*) \\ &= \frac{exp(\beta_0 + \sum_{j=1}^{p} \beta_j x_{i,j})}{1 + exp(\beta_0 + \sum_{j=1}^{p} \beta_j x_{i,j})} \end{aligned} \tag{1}$$

where $\tilde{x_i} = [1 \quad x_{i,1} \quad \ldots \quad x_{i,p}]$ and $\beta^* = [\beta_0 \quad \beta_1 \quad \ldots \quad \beta_p]$ [16] [17].

[1] In truth, one does not capture the king to win. In order to win in chess, one must threaten to capture the opponent's king while simultaneously leaving no means for the opponent to move their king out of harm's way on their upcoming turn. This is called "checkmate".

B. k-Nearest Neighbors (k-NN) Classifier

Given a positive integer K, the k-NN classifier first identifies K points in the training data whose predictors X are closest to x_0 (our observation), then denote the set of those points as $\mathcal{N}_0$ [16].

The conditional probability that Y belongs to the class c is estimated as the fraction of points in $\mathcal{N}_0$ whose class is c [16] [17]:

$$\hat{\mathbb{P}}(Y = c|X = x_0) = \frac{1}{K} \sum_{i \in \mathcal{N}_0} \mathbb{1}_{y_i = c} \tag{2}$$

In this project, c will be win/lose and similar to logistic regression, the output will be the probability of win/lose based on the values of predictors, x_0.

C. Decision Trees

Decision trees involve stratifying or segmenting the predictor space into a number of simple regions. The general steps of constructing decision trees are [16]:

1) Divide the predictor space into J distinct non-overlapping regions $R_1, R_2, ..., R_J$
2) For every observation that falls into region R_J make the same prediction. For classification problem, use the mode. The predicted class for R_J would be the one that minimizes objective function.

The common choices for the objective functions are Gini index and entropy since they provide better interpretability [17]. For this project, we will use the Gini index. Denoted by G_r, it measures the purity of a region by calculating the variances:

$$G_r = \sum_{c \in C} \hat{p}_{rc}(1 - \hat{p}_{rc}),$$

where p_{rc} is the proportion of observations, whose predictor fall in region r and response is c. The perfect Gini index would be 0 which means all data points were correctly classified.

D. AUC and ROC Curve

The Receiver Operating Characteristic (ROC) curve is a function $R : [0,1] \longrightarrow [0,1]$ which plots the True Positive Rate (TPR) (on the y axis) as a function of the False Positive Rate (FPR) (on the x axis) when is varied from 0 to infinity [16].

$$\text{TPR} = \frac{\text{TP}}{\text{TP+FN}} \tag{3}$$

$$\text{FPR} = \frac{\text{FP}}{\text{FP+TN}} \tag{4}$$

A measure often used to assess the performance of a classifier is the area under ROC curve (AUC). If the classifier is perfect, the AUC would be 1 and for classifiers with no predictive ability, the AUC would be close to 0.5. The higher the AUC, the better the model performance.

E. Stockfish

A chess engine is a program that analyzes chess positions and returns the best moves to tackle the position. *Stockfish* (developed by Romstad, Costalba, Kiiski, et. al) is one of the free, leading, open-source chess engines that is available for various platforms [18]. As of October 2022, it has won the Top Chess Engine Championship 13 times.

Motivated by the fact that chess has enormously large position possibilities, the author utilized Stockfish as a way to create a win metric. Since the engine returns the best moves for a certain position and our data recorded all of the moves played in a game, the author was able to keep track of the board position in the game. Thus, by "feeding"the position to the engine, it will tell us n top moves to play with a certain depth. The default setting for depth is 15, meaning that the engine is able to see 15 plies ahead in the game. The greater the depth, the more accurate the score, and the prediction.

The author decided to use $n = 3$ and the default depth due to limited computing power and some prior testing by comparing it with $n = 5$. The metric is then defined to be

$$\text{Win Rate} = \frac{\text{\# of moves similar to Stockfish}}{\text{\# of moves played in the game}} \tag{5}$$

If White's win rate is bigger than Black's, then White is the winner and vice versa. If their rate is equal, then it's a draw.

VI. Results

A. Exploratory Data Analyses

Since we're trying to predict the winner of the game, the author would like to see if the winning distribution is uniform for both White and Black.

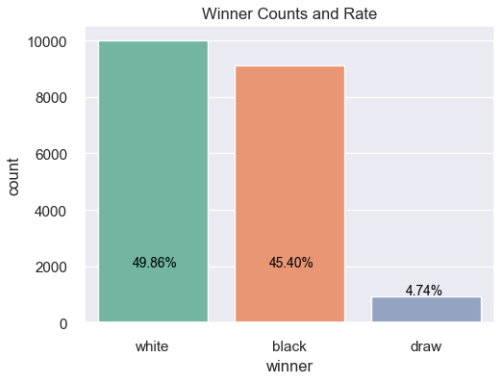

Figure 2: Winning Proportions from Lichess Data

From Figure 2, there is a quite noticeable gap; almost 5% between White and Black's winning rate. This raises a suspicion: was there a huge gap between the player's ratings?

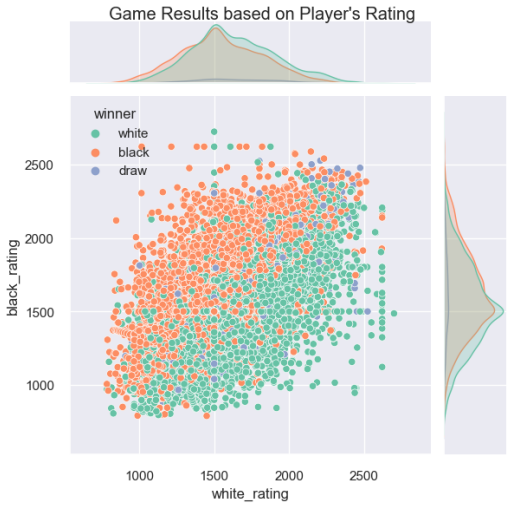

Figure 3: Results based on Player's Ratings

As seen in Figure 3, the game was mostly fair between the players where they are assigned to an opponent whose rating is similar to theirs. There are still some outliers where one player has a significantly higher rating but this happened in both cases. In addition, both of the player's ratings distribution looks fairly similar and approaching normality. From this, we can conclude that there is a strong evidence from our data that supports first-move advantage in chess (which is indeed a well documented phenomenon, explained by what chess players would call the theory of "tempo").

B. Predictive Analyses

Using logistic regression, the author obtained an accuracy of 65.2%. The author then applied k-NN Classifier and Decision Trees, which were not better than logistic regression at 62.5% and 62% respectively.

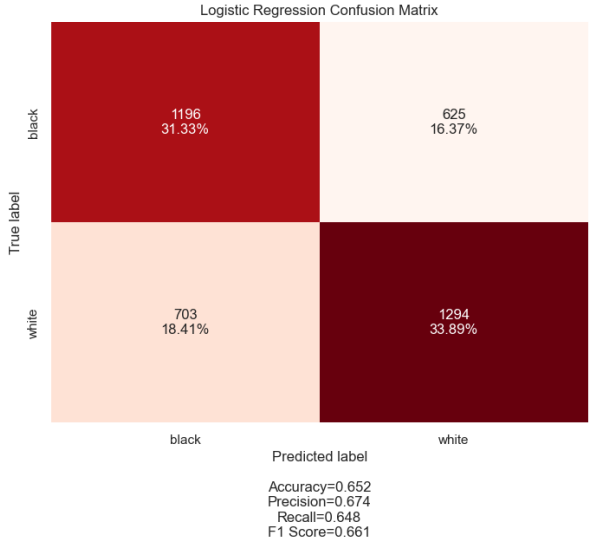

Figure 4: Confusion Matrix for Logistic Regression

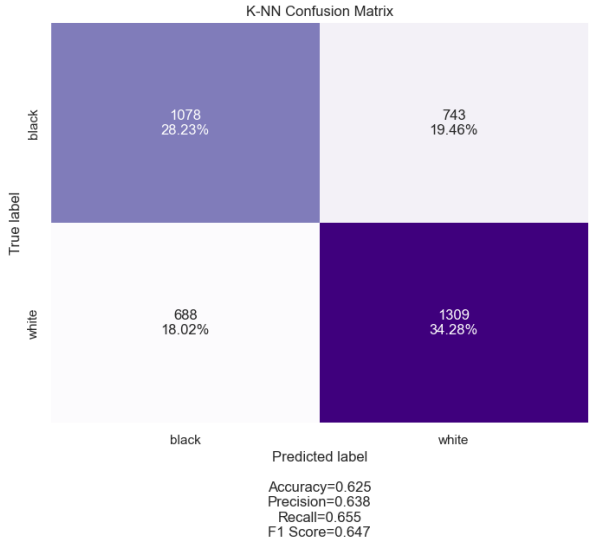

Figure 5: Confusion Matrix for k-NN Classifier

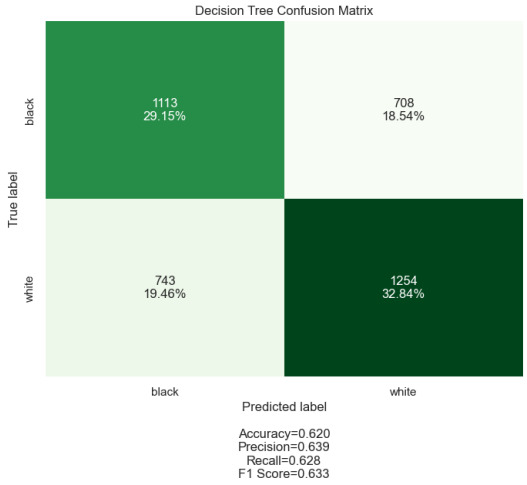

Figure 6: Confusion Matrix for Decision Tree

From the confusion matrices, we observe that all three methods performed better in predicting the White pieces. To analyze the goodness-of-fit, the area under ROC curve were calculated for all 3 methods and it was found that the logistic regression has the biggest area, 0.72, followed by k-NN and Decision Tree which can be seen below.

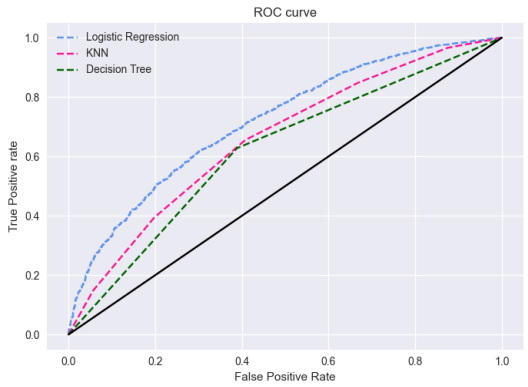

Figure 7: Area Under ROC Curve (AUROC) for All Methods

C. Stockfish Win Metric

In hopes to improve the prediction accuracy, the author tested the metric based on Equation 5 on 100 observations and found the accuracy to be 61.6%.

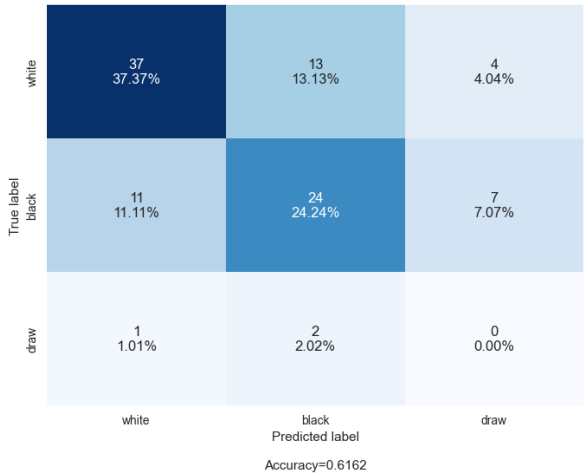

Figure 8: Confusion Matrix for The Metric

From Figure 8, our metric performed extremely poor in predicting draws, with 0% accuracy.

VII. Discussion

The results obtained from all three classification methods were above average, but were not exceedingly impressive. No method surpasses 65%. This is in line with some of the literature mentioned in Section III where Fan et al. achieved 55.64% accuracy as well as Masud et. al with 66% prediction accuracy on their dynamic classification technique.

The first flaw of the author's model is that we excluded draw from the game outcome. This was done due to the fact that the author wanted to build a simple binary classifier. For further research, it might be better to include draw so the model will be able to predict all outcomes.

As mentioned in Section II, chess is a sophisticated game with outrageously large board position possibilities. It is very difficult to predict where the game is going with only the opening moves. A player might be doing better at the end of the game and our model can't capture that. Therefore, it might be better to use a dynamic modeling approach similar to Masud et. al, where the prediction is made while the game is still ongoing. That will greatly help with any "curveballs"thrown by players during the game.

Another possibility is to model this with a reward-based approach with reinforcement learning. Since the pieces have values, the model can use that as a reward if a player captures the pieces and give out a penalty if their pieces are being captured.

Furthermore, the author realizes that statistical methods may not be the best approach in this case due to the huge number of board position possibilities and that is why Stockfish was used to create the metric.

The metric performed above average and it can still be improved. Due to limited computing power, the author was not able to use the whole data set (20,040 observations) and instead only used 100 to see how the metric performed. With more data points, we can get higher confidence about the metric performance.

This metric did not perform well in predicting draws. This might be due to the fact that if their metric rate is equal, we set the outcome to be draw. However, some high-rated players might outplay their opponent by playing non-textbook moves that are not included in the Stockfish's top moves.

The author only looked at the top 3 moves by Stockfish with a depth of 15 moves ahead. These are "free" parameters and the author was limited by computing power. If we increase the number of moves and depth, we might get a better performance since the Stockfish would be able to foresee more than 15 moves ahead.

VIII. Conclusion

In this final project, the author modeled three different approaches and created a win metric for predicting a chess winner. Logistic regression, k-NN Classifier, and Decision Tree were applied to an online chess dataset where logistic regression has the highest accuracy prediction of 65.2%, followed by k-NN at 62.5% and Decision Tree at 62%. Using AUROC, the author also confirmed that Logistic Regression is the best out of the three since it has the biggest area of 0.72.

Stockfish chess engine was used to create a metric and it was found to have a prediction accuracy of 61.6%. The models built in this project failed to predict draw accurately due to the limitations mentioned. To conclude, all attempts in this project have an above-average performance for binary prediction outcomes for chess, but it is still open for improvements.

References

[1] H. J. R. Murray, *A history of chess*. Northampton, Mass. : Benjamin Press, 1913.

[2] Y. Averbakh, *A History of Chess: From Chaturanga to the Present Day*. Russell Enterprises, Inc, 2012.

[3] D. Shenk, *The Immortal Game: A History of Chess*. New York: Knopf Doubleday, 2007.

[4] G. Berkeley, "Magnus carlsen net worth," 2021, last accessed on 15 November 2022. [Online]. Available: https://www.insidethegames.biz/articles/1103159/carlsen-highest-earning-esports-player

[5] J. Melvin, "Interest in chess rises over 300% following netflix's 'the queen's gambit'," 2020, last accessed on 15 November 2022. [Online]. Available: https://techround.co.uk/news/interest-in-chess-surges-after-the-queens-gambit-release

[6] chess.com, "Analysis," 2022, last accessed on 15 November 2022. [Online]. Available: https://www.chess.com/analysis

[7] M. Jolly, "Chess game dataset (lichess)," 2017, last accessed on 14 November 2022. [Online]. Available: https://www.kaggle.com/datasets/datasnaek/chess

[8] I. C. Federation, "Fide handbook: Basic rules of play," 2022, last accessed on 15 November 2022. [Online]. Available: https://handbook.fide.com/chapter/E012018

[9] C. E. Shannon, "XXII. programming a computer for playing chess," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 41, no. 314, pp. 256–275, Mar. 1950. [Online]. Available: https://doi.org/10.1080/14786445008521796

[10] Z. Fan, Y. Kuang, and X. Lin, "Chess game result prediction system," *Machine Learning Project Report CS*, vol. 229, 2013.

[11] F. Thabtah, A. J. Padmavathy, and A. Pritchard, "Chess results analysis using elo measure with machine learning," *Journal of Information &amp Knowledge Management*, vol. 19, no. 02, p. 2050006, Mar. 2020. [Online]. Available: https://doi.org/10.1142/s0219649220500069

[12] M. M. Masud, A. Al-Shehhi, E. Al-Shamsi, S. Al-Hassani, A. Al-Hamoudi, and L. Khan, "Online prediction of chess match result," in *Advances in Knowledge Discovery and Data Mining*, T. Cao, E.-P. Lim, Z.-H. Zhou, T.-B. Ho, D. Cheung, and H. Motoda, Eds. Cham: Springer International Publishing, 2015, pp. 525–537.

[13] Lichess, "About lichess.org," 2022, last accessed on 14 November 2022. [Online]. Available: https://lichess.org/about

[14] ——, "Lila," 2022, last accessed on 14 November 2022. [Online]. Available: https://github.com/lichess-org/lila

[15] T. Blanchard, "The pareto principle — spending time and energy effectively as a data scientist," 2021, last accessed 10 November 2022. [Online]. Available: https://towardsdatascience.com/the-pareto-principle-spending-time-and-energy-effectively-as-a-data-scientist-8d48f7cc2b0f

[16] D. W. T. H. James, Gareth and R. Tibshirani, *An Introduction to Statistical Learning with Applications in R*. Springer, 2021.

[17] R. T. Trevor Hastie and J. Friedman, *The Elements of Statistical Learning*. Springer New York, 2009.

[18] D. Yang, "About stockfish," 2022, last accessed on 25 November 2022. [Online]. Available: https://stockfishchess.org/about/