Data Analytics MSc Dissertation MTHM038, 2023/24

Predicting Evaluations of Chess Endgames

Using data analysis and machine learning algorithms

David Andrews, ID 230807534

Supervisor: Dr. Hong Qi

A thesis presented for the degree of
Master of Science in *Data Analytics*

School of Mathematical Sciences
Queen Mary University of London

Declaration of original work

This declaration is made on August 30, 2024.

Student's Declaration: I David Andrews hereby declare that the work in this thesis is my original work. I have not copied from any other students' work, work of mine submitted elsewhere, or from any other sources except where due reference or acknowledgement is made explicitly in the text. Furthermore, no part of this dissertation has been written for me by another person, by generative artificial intelligence (AI), or by AI-assisted technologies.

Referenced text has been flagged by:

1. Using italic fonts,
2. using quotation marks "...", **and**
3. explicitly mentioning the source in the text.

Abstract

This thesis explores the approach of using machine learning algorithms to predict the evaluation of chess endgame positions. This prediction will be based on various features such as material count and the number of passed pawns. The primary objective is to implement an accurate model that reflects the complexity of any given chess position. By using a variety of different prediction models we find that the polynomial model is the most accurate, being able to predict which player is winning with 92% accuracy. Furthermore, this thesis will analyse certain piece combinations to discover which pieces perform better than others in an endgame. These compelling results can assist players in their own chess games by highlighting the key factors that can lead to victory. In order to get a better understanding of the analysis we will upload a large amount of different chess positions from `https://lichess.org/`. This is the only site that will be used when showing examples of different chess positions.

Contents

Chapter 1

Introduction

Chess as we know it today, dates back to “6th century India” [**1**] where it was known by only a small handful of people in the world. The influence of the game spread through to “Europe in the 9/10th century” [**2**] and now in the 21st century that we live in today, it is played, studied and loved by hundreds of millions. The obvious question one might ask is why? Why are so many children and adults from every corner of the earth spending their free time indulging in this 8x8 grid board game? Former World champion Anatoly Karpov said “Chess is everything: art, science, and sport” [**3**]. As we will see throughout this project, chess is a difficult, complex and also a very competitive game. Humans thrive on this competitive nature and it fuels them to constantly play and improve. In a world of technological distractions, chess is a great alternative as studies have shown it improves creativity, problem-solving skills and even mental health. A single game of chess can be split into 3 sections: the opening, the middle game and the endgame. In this thesis we will explore the endgame, the last section of the game to identify what factors of a position will help determine which player is in the winning situation. As well, we will use modern machine learning algorithms to make prediction models for a given position. By doing so we can help players identify key positions and get a stronger idea on whether or not they are winning or losing in a given chess position. Before going into further detail, we need to firstly explain the game of chess itself.

1.1 Basic Rules of Chess

1.1.1 Objective

The game of chess is played between two players, one with the white pieces and the other with the black pieces. The primary objective of the game is to checkmate the opponent's king. Each player, starting with the player who has the white pieces, takes it in turns to make one move at a time to each try and complete a checkmate before the opponent does. A checkmate occurs when the opponent's king is in a position to be captured and there are no legal moves to prevent this from happening.

1.1.2 The Chess Board

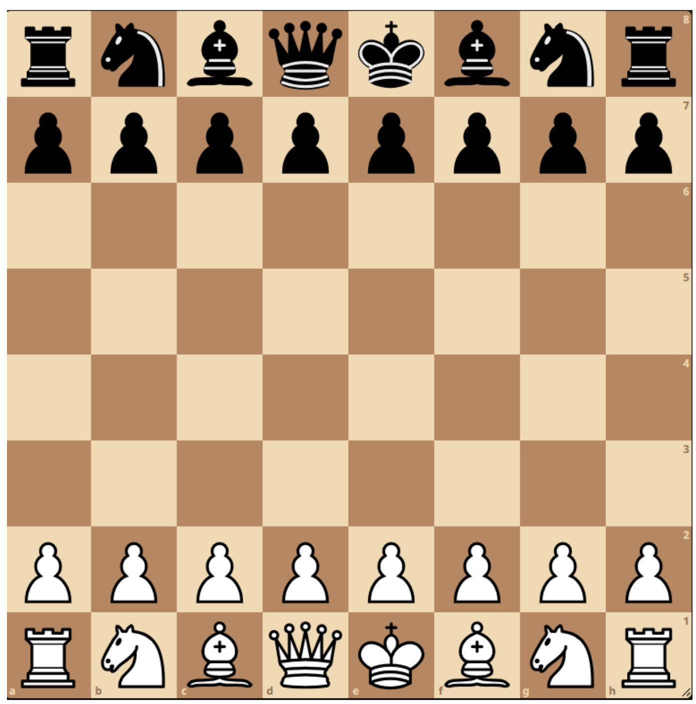

Figure 1.1: The starting position of a chess board

Figure 1.1 is the starting position of a chess board. It is an 8x8 grid with squares alternating between light and dark. The rows are given a rank from **1-8** and the columns **a-h**. For instance, we can say the white king is currently on the square **e1**. Before any moves are made each player has the same number of pieces but on opposite sides of the board to one another. Only one piece can be on a square at any given time.

1.1.3 The Pieces

Each player begins with 16 pieces: one king, one queen, two rooks, two bishops, two knights, and eight pawns.

- **King**: Moves one square in any direction. The game is lost if the king is checkmated.
- **Queen**: Moves any number of squares in any direction (vertically, horizontally, or diagonally).
- **Rook**: Moves any number of squares vertically or horizontally.
- **Bishop**: Moves any number of squares diagonally.
- **Knight**: Moves in an "L" shape: two squares in one direction and then one square perpendicular, or one square in one direction and then two squares perpendicular. Knights can jump over other pieces.
- **Pawn**: Moves forward one square but captures diagonally. On its first move, a pawn can move forward two squares. If a pawn reaches the farthest row from its starting position, it can be promoted to any other piece (except a king or pawn).

As we can see, each piece is different with some pieces being much more powerful than others. For example, the queen is the most powerful piece as it can move in the direction of the rook and bishop combined (vertically, horizontally, or diagonally) while the pawn has very limited options/distances in which it can travel. To measure the difference of value between each piece, chess players have assigned a point system for each piece. The best way to think about it is that a pawn is the weakest piece worth 1 point. A queen is the strongest worth 9 and is essentially 9 times as powerful as a pawn. Figure 1.2 not only shows what each piece looks like but also the material points assigned to each piece which is a very useful set of numbers to use when evaluating chess positions.

Figure 1.2: Chess pieces values and shapes [5]

As we can see from Figure 1.2, the queen is the most valuable piece representing 9 points of material. The rooks are the second most valuable with 5 points for each piece. The bishops and knights are equal with 3 and then the least valuable piece is a pawn with 1 point of material. The kings points is said to be infinite as the objective is to capture the king hence it is technically the single most important piece but in regards to the total sum of each players points we won't include the kings. At the start of the game each player has a total of 16 pieces on the board summing to 39 points of material.

1.1.4 Capturing

An inevitable part of a game to help you defeat an opponent is capturing their pieces. By removing their pieces you can create an imbalance, usually in your favour. The example below will show this:

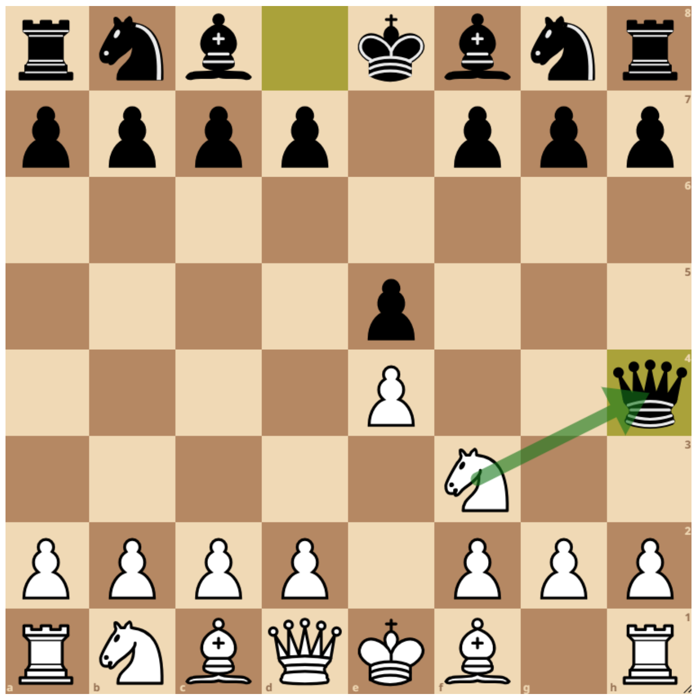

Figure 1.3: A knight capturing a Queen

We can see that if it is white's move then using the move rules for each piece we can simply move the knight to the black queen's square. By doing so this removes the black queen and 9 points of material giving white a huge advantage. It is important to take advantage of opportunities like this however at a higher level of chess you are unlikely to get such an easy capture like we see in Figure 1.3.

1.1.5 End of the Game

As previously stated the game is won when the opponent's king is in a checkmate position. However, if all the pieces are removed from the board except each players king then the game will end in a draw because it is impossible for a king to checkmate another king. We can see a simple example of where a player is in checkmate. White has just moved its rook to the square **d8**. This move has attacked the black king and no matter what move black makes it can not defend itself hence black has been checkmated and white has won the game.

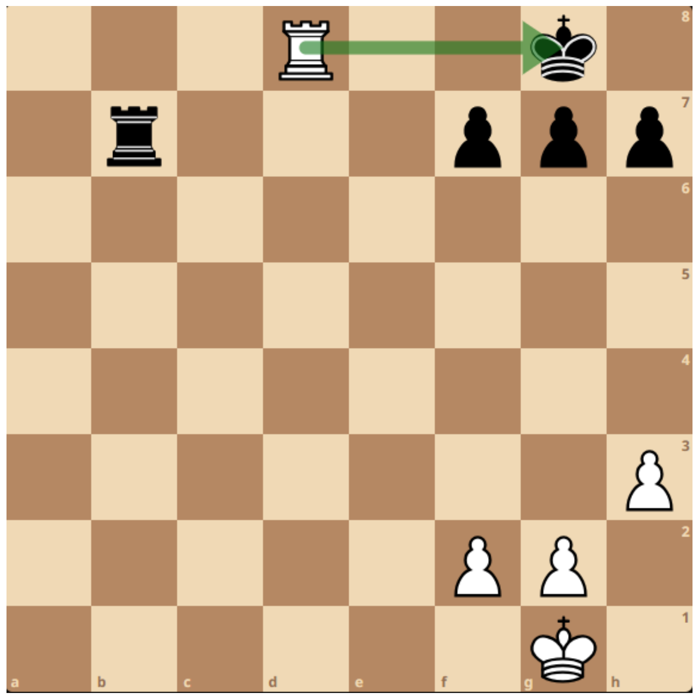

Figure 1.4: White checkmating black's king

By only slightly changing the position, we can see in Figure 1.5 that the only difference is that black's pawn on **h7** has moved forward by one square to **h6**. Black's king can now move to **h7** and is no longer in checkmate. The game will continue until one player wins or a draw occurs.

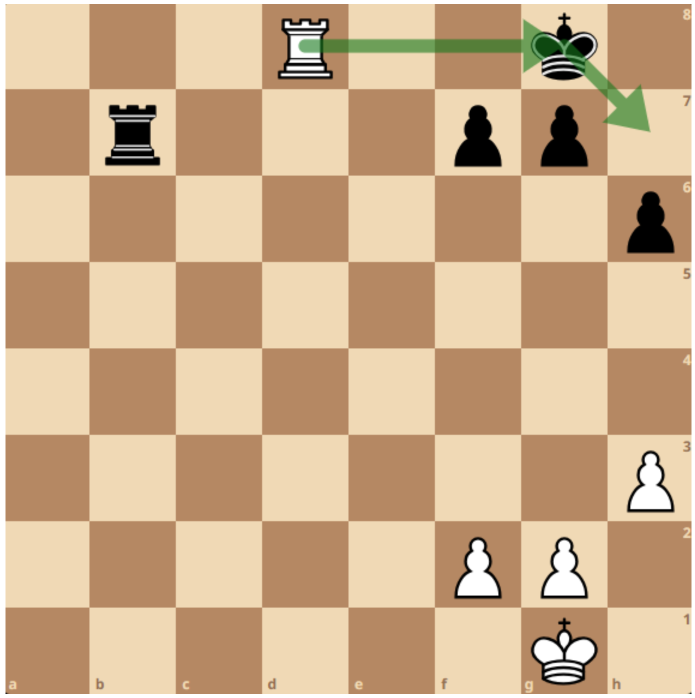

Figure 1.5: Black's king escaping a checkmate threat from white's rook

There are other rules in chess like en passant or castling plus many tactics that haven't been touched on however in regards to this task they won't be relevant to the data analysis performed and the results discovered to help improve ones chess ability in an endgame.

1.2 Background & Justification for Chess Endgame Analysis

The first chess engine to beat the best player in the world was "Deep Blue in 1997" [**6**]. Since then chess engines have continuously developed making it nearly impossible for any human to come close to defeating them. There are many different chess engines today in 2024 with each taking different approaches to try and become as accurate as possible. One might ask if chess can be solvable to the point where an engine cannot become anymore accurate then it already is. "There are 10^{78} to 10^{82} atoms in the observable universe and we can estimate the number of possible chess positions to be in between 10^{111} and 10^{123}!" [**7**]. The sheer amount of variation possible makes it a difficult task for chess engines to be perfected which is why millions of dollars are being spent yearly to get to as close to perfection as possible. The best chess engine to date is Stockfish which can analyse "100 million positions per second using rudimentary heuristics" [**8**]. Just based on this sheer number, we can see already why the gap between engine and human is so big as humans just don't have the same processing speeds and memory storage as computers do.

Stockfish gives chess positions an evaluation number to determine which player is in the better position. With a positive number giving the advantage to the white pieces, a negative number giving an advantage to the black pieces and 0 if the position is even which results in a draw if the best moves are all played out. From an article from Cornell University [**9**], "a unit of evaluation above 1.5 or below -1.5 means that a human chess master with the advantage will likely win. If the evaluation is above 4 or below -4 then the position is completely winning for the player with the advantage". In this analysis, I will only include games with evaluations between -4 to -1.5 or 1.5 to 4, where there is an advantage for one player but not a completely winning position.

In this report I will use machine learning algorithms to find a prediction model that will output its own evaluation, trying to get similar results to Stockfish's evaluation. I will use a number of different parameters and algorithms in order to get the model as accurate as possible. Instead of trying to get the evaluations from my model exactly the same, I will test to see if I can get the output the same sign as Stockfish's evaluation (positive or negative). If this is achieved this will be a successful result as the model can determine which player has the advantage. Afterwards, I will analyse different types of endgames to explore if certain pieces/combinations are more beneficial than others.

The endgame occurs in chess when there are few pieces on the board. There are many different debates from chess experts as to what exactly defines an endgame. “Minev characterizes endgames as positions having four or fewer pieces other than kings and pawns. Fine considers endgames to be positions without queens. Flear considers endgames to be positions where both players have at most one piece (other than kings and pawns)” [**12**]. As there is no real definition to when an endgame actually starts we will look at games where there are no queen’s on the board as well as there being 10 or less pieces in total. These constraints are a good idea of what an endgame is in terms of analysis.

The opening and middlegame are extremely complicated to analyse and the approach of machine learning would not be efficient as there are too many parameters to consider just on one random position alone. Due to the fact there are less pieces in an endgame, we can use ML algorithms more effectively to make prediction models.

The parameters/factors for our model will now be discussed. Later on, we find that the best model has an accuracy around 92%.

Chapter 2

Parameters

In this section we will go over the parameters used in our model to see why they are important into the chances of having a higher evaluation in a chess endgame. Our parameters are as follows:

- Material points white
- Material points black
- Passed pawns white
- Passed pawns black
- King activity white
- King activity black
- Player turn

An important thing to note is that due to the nature of the game, each player will have the same aims and tactics to beat the other opponent hence, we see the same factors for each player. In the following sections we will explain each factor and give examples as to how they are important into increasing a players winning chances.

2.1 Material Points

As we saw from Figure 1.2 each piece has a different material point value due to how powerful the piece is. One would be quick to assume that the higher ones material points over the other player, the more likely the evaluation would be in their favour. They'd be correct most of the time. Let's see a simple example below:

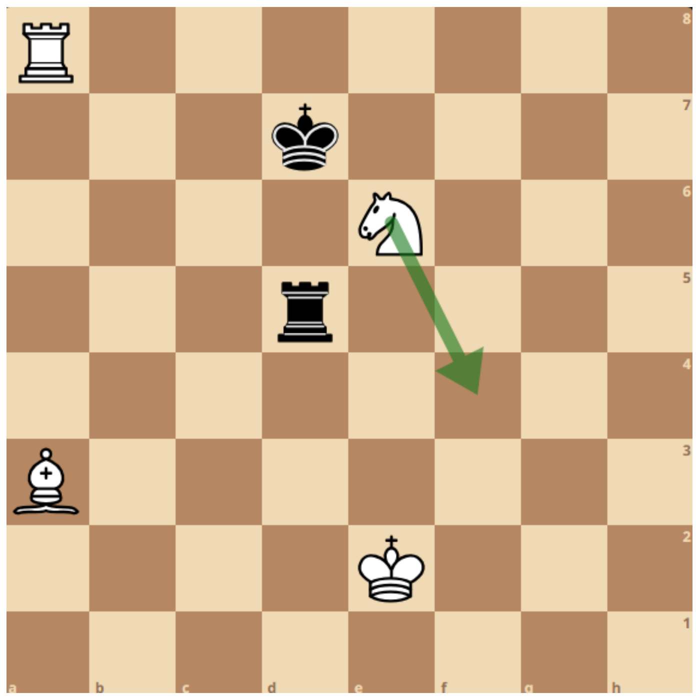

Figure 2.1: An unbalanced chess position with white having higher material points

We see that apart from the king black has one rook which is worth 5 points. White has one rook, bishop and knight totalling 11 points. This 6 point difference gives white a big advantage as it can control way more squares than black can. White's knight is under attack, however it is white's turn so the best option is to move the knight to the square **f4** where it cannot be taken by either of black's last two pieces. This position corresponds to a Stockfish evaluation of 3.45 hence showing how an advantage in material points can increase the chances of winning a game.

2.2 Passed Pawns

If a pawn reaches the end of the board from its starting position then it can promote to a piece of their choosing. Almost all of time except on extremely rare occasions this will be a queen. Assuming the new queen can't be easily captured then this will give the player a huge advantage. The pawn will go from a piece worth one point to a piece worth nine. However, it isn't an easy task to get a piece with less power like a pawn to the back of the enemy's line. In endgames it is a very common thing players need to consider and can cost/win you the game. "A passed pawn is a pawn that has advanced beyond all of the enemy pawns. In other words, a passed pawn has no enemy pawns in front of it or on adjacent files that can stop it from being promoted" [**13**].

If there was an enemy pawn in front or on adjacent files it would be easier to stop the pawn from promoting. In my code I have defined a passed pawn to be 3 or less moves from the last row of the board. This is because if the pawn was on its starting square, then it would take a lot of moves to successfully promote. A pawn closer to promoting would be a much bigger threat hence would contribute to a bigger factor in a players chances of winning.

As we can see from Figure 2.2 below, each player has a similar position, each with a rook. Not only does white have one extra pawn, it is also a passed one on the **a5** square. It posses a massive threat to black resulting in its rook and king distracted to it. While black has to deal with the pawn white's pieces can target black's undefended pawns on the other side of the board. This passed pawn for white gives white an advantage evident by its Stockfish evaluation of 2.00.

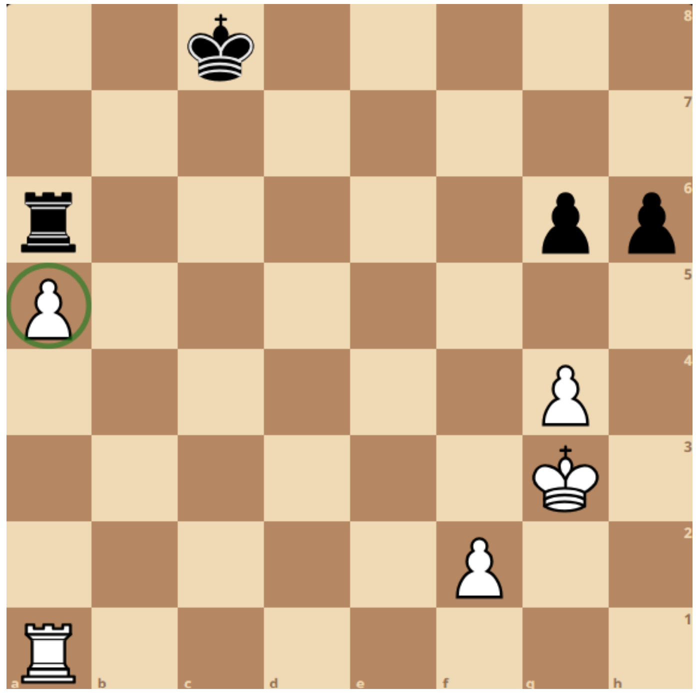

Figure 2.2: A position where white has a passed pawn

2.3 King Activity

At the start of the game it is important to get your king to safety. If not it can easily be attacked resulting in you losing the game. In the endgame this is a different case. The king can become activated and offer a huge amount of help in controlling the board. In a general case it is important to get your king as close to the centre 4 squares as possible. This is because the centre squares are very important and whoever controls them has a big advantage. We can show this by looking at Figure 2.3 below.

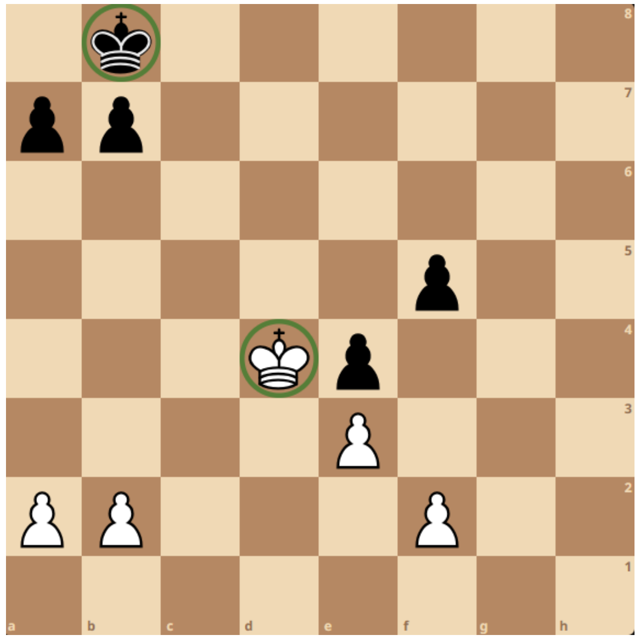

Figure 2.3: A position where white has a more active king than black

The material count for both players are the same and there are no passed pawns for either side. However, due to the fact that white's king on the **d4** square is more active, it gives white a huge advantage as black's king on **b8** won't be able to defend its pawns in the centre of the board.

We can use the Chebyshev distance to calculate the minimum distance the king needs to travel to reach one of the 4 centre squares. In the analysis, we represent the chessboard as a plane with coordinates (x_i, y_i), where $i \in [1, 8]$. To determine the minimum number of moves a king needs to move from one square to another, we use the following formula to calculate the distance between two points (x_1, y_1) and (x_2, y_2):

$$d = \max(|y_2 - y_1|, |x_2 - x_1|) \quad \textbf{[14]} \tag{2.1}$$

This formula reflects the king's ability to move one square in any direction (horizontally, vertically, or diagonally), thus the maximum of the horizontal and vertical distances gives the required number of moves.

2.4 Player's Turn

There are instances where which player's turn it is can have a big impact on the evaluation of a position. If we look back at Figure 2.1, we saw that not only did white have a 6 point material advantage, it was also white's turn giving white the opportunity to move its under attacked knight. If we use Stockfish's evaluation but instead it is black's move then the king can simply capture the knight moving the evaluation from 3.45 to 0 so if the best moves were to be played the game would end in a draw. Hence, there are scenarios in which the player's turn has a big impact however what we will see when computing the ML algorithms is that this parameter has less of an impact than the others we have gone over.

Now we have explained how each factor into ones chances of winning a chess endgame, we can start discussing which ML algorithms will be used to find an accurate prediction model.

Chapter 3

Methodologies

3.1 Standardisation

It is important to standardise the data before computing any algorithms. One reason to do so is because of each variable having different scales. For instance, the highest material count is taken with integers up to 12 while the other parameters can have inputs as low as 1. Machine learning models perform better when each variable is on a similar scale. We standardise the data which is in the format $\mathbf{X} = \left(\left(\mathbf{x}^{(1)}\right)^{\top}, \left(\mathbf{x}^{(2)}\right)^{\top}, \ldots, \left(\mathbf{x}^{(s)}\right)^{\top} \right) \in \mathbb{R}^{s \times d}$ using the formula below:

$$\hat{\mathbf{x}}_k^{(i)} = \frac{\mathbf{x}_k^{(i)} - \langle \mathbf{x}_k \rangle}{\left(\sigma_{\mathbf{x}}\right)_k}, \tag{3.1}$$

where $\langle \mathbf{x}_k \rangle = \frac{1}{s} \sum\limits_{j=1}^{s} \mathbf{x}_k^{(j)}$, and $\left(\sigma_{\mathbf{x}}\right)_k = \sqrt{\frac{1}{s} \sum\limits_{j=1}^{s} \left(\mathbf{x}_k^{(j)} - \langle \mathbf{x}_k \rangle \right)^2}$ are the mean and standard deviation of data vector $\mathbf{x}$.

3.2 Linear Model

The linear prediction model is given by: z=$\mathbf{\Phi}\left(\mathbf{X}\right)\hat{w}$ where $\hat{w}$ is a vector of the optimal weights for each variable and $\mathbf{\Phi}\left(\mathbf{X}\right)$ is the standardised data matrix such that:

$$\mathbf{\Phi}\left(\mathbf{X}\right) = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & \dots & x_d^{(1)} \\ x_1^{(2)} & x_2^{(2)} & \dots & x_d^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{(s)} & x_2^{(s)} & \dots & x_d^{(s)} \end{bmatrix}$$

In this data set the rows s correspond to the total number of chess positions. And the columns corresponds to each parameter. In total we have 7 different parameters hence d=7.

3.3 Polynomial Model

As well as a linear prediction model we can also use polynomial regression to fit our model. As before the model is of the form z=$\mathbf{\Phi}\left(\mathbf{X}\right)\hat{w}$. We will use the same methods to find optimal weights but for a polynomial regression the standardised data matrix is now of the form:

$$\Phi\left(\mathbf{X}\right) = \begin{bmatrix} \left(x^{(1)}\right)^1 & \left(x^{(1)}\right)^2 & \dots & \left(x^{(1)}\right)^d \\ \left(x^{(2)}\right)^1 & \left(x^{(2)}\right)^2 & \dots & \left(x^{(2)}\right)^d \\ \vdots & \vdots & \ddots & \vdots \\ \left(x^{(s)}\right)^1 & \left(x^{(s)}\right)^2 & \dots & \left(x^{(s)}\right)^d \end{bmatrix}$$

Here d is the degree of the polynomial model. Depending on the size of d the vector of optimal weights will increase with a higher degree. We will explore in the model selection section if the linear or polynomial model is better suited for the prediction model.

3.4 Bootstrap Regression

In addiction to the linear and polynomial regression models, we can use bootstrapping to estimate the variability in the model's predictions. This method involves repeatedly sampling from the data to generate multiple estimates for the parameters used in the model. Given the standardised data matrix $\mathbf{X}$ plus the corresponding outputs $\mathbf{y}$ the procedure is as follows:

1. **Sample Selection**: For each bootstrap iteration $j \in \{1, \ldots, M\}$, we generate a sample $(\mathbf{X}_j^*, \mathbf{y}_j^*)$ by randomly selecting s observations from $(\mathbf{X}, \mathbf{y})$ with replacement.

2. **Model Fitting**: We then fit a regression model to each bootstrap sample. Here we will use ridge regression:

$$\hat{\mathbf{w}}_j = \operatorname{argmin}_{\mathbf{w}} \left\{ \|\mathbf{y}_j^* - \mathbf{X}_j^* \mathbf{w}\|^2 + \alpha \|\mathbf{w}\|^2 \right\}$$

where α is the regularization parameter.

3. **Weight Storage**: The estimated weights $\hat{\mathbf{w}}_j$ from each iteration are stored. After M iterations, we obtain a set of weight vectors $\{\hat{\mathbf{w}}_j\}_{j=1}^{M}$ for each parameter, which allows us to assess the variability and robustness of our model's predictions.

When we come to computing this method we have selected $M = 2000$ iterations. The optimal value of the regularization parameter α was found by performing a grid search over a specified range of α values. For each α in the grid, the validation error was computed using K-fold cross-validation. The α that minimized the validation error was selected as the optimal $\hat{\alpha}$=16.0.

3.5 Random Forest Algorithm

We can use a built in random forest algorithm to rank the importance of each parameter in the model. This is done by calculating the decrease in prediction accuracy when a feature is randomly permuted, indicating how critical that feature is to the model. Later on we will see this being done with the parameters stated in Section 2.1. We will also compute this algorithm with the following parameters:

- White Pawns
- Black Pawns
- White Bishops
- Black Bishops
- White Knights
- Black Knights
- White Rooks
- Black Rooks

We will not include these parameters in the prediction model. This because it overlaps with the material points parameter as well as the fact we want to avoid over fitting the model. Even so it is important to get an idea of which pieces are more important than one another when analysing a chess endgame.

Chapter 4

Dataset Overview

4.1 Data Source & Structure

The data set that will be used for the analysis and ML algorithms is from `lichess.org` which is the 2nd most popular chess website in the world [**10**]. The 1st is `chess.com` however it requires payment for data collection while lichess is completely free to use. The data used can be found here: [**11**]. The data set contains over 13 million different chess positions with a number of different pieces of information. However, it is important to note that in these 13 million positions there are a large amount of positions that don't fit the constraints of what is an endgame as well as what is a winning but not completely winning position. Once we implement these constraints onto the data set we are left with a new data set only containing 74,489 positions. Here we see an example of what the data looks like for one of the positions:

```json
{
  "fen": "2bq1rk1/pr3ppn/1p2p3/7P/2pP1B1P/2P5/PPQ2PB1/R3R1K1 w - -",
  "evals": [
    {
      "pvs": [
        {
          "cp": 311,
          "line": "g2e4 f7f5 e4b7 c8b7 f2f3 b7f3 e1e6 d8h4 c2h2 h4g4"
        }
      ],
      "knodes": 206765,
      "depth": 36
    },
    {
      "pvs": [
        {
          "cp": 292,
          "line": "g2e4 f7f5 e4b7 c8b7 f2f3 b7f3 e1e6 d8h4 c2h2 h4g4"
        },
        {
          "cp": 277,
          "line": "f4g3 f7f5 e1e5 d8f6 a1e1 b7f7 g2c6 f8d8 d4d5 e6d5"
        }
      ],
      "knodes": 92958,
      "depth": 34
    },
    {
      "pvs": [
        {
          "cp": 190,
          "line": "h5h6 d8h4 h6g7 f8d8 f4g3 h4g4 c2e4 g4e4 g2e4 g8g7"
        },
        {
          "cp": 186,
          "line": "g2e4 f7f5 e4b7 c8b7 f2f3 b7f3 e1e6 d8h4 c2h2 h4g4"
        },
        {
          "cp": 176,
          "line": "f4g3 f7f5 e1e5 f5f4 g2e4 h7f6 e4b7 c8b7 g3f4 f6g4"
        }
      ],
      "knodes": 162122,
      "depth": 31
    }
  ]
}
```


Figure 4.1: An example of the data of a single chess position in the dataset

There are only two pieces of information needed for the analysis:

- FEN: A FEN (Forsyth-Edwards Notation) is a standard notation for describing a board, providing all necessary information to analyse a given position.
- First cp: The first cp for a given position is the most accurate Stockfish evaluation times by 100. For example the first cp for the position in Figure 4.1 is 311 which corresponds to an evaluation of 3.11. Under the constraints stated before for the evaluations, this value of 3.11 will be included in the analysis.

4.2 Data Extraction

Once the data is all uploaded we can run functions to extract the positions which are endgames under the definition stated before. After that we can extract the two meaningful pieces of information: the FEN and then the first cp which we can divide by 100 to obtain the evaluations. Once completed, we see that the first few rows of the data frame looks like this:

	FEN	Evaluation
0	R4k2/5br1/3N4/p3P3/8/5K2/8/8 b - -	3.91
1	1nb5/1p6/1kp5/8/8/6B1/8/3R3K w - -	2.25
2	8/8/4kpp1/3p1b2/p6P/2B5/6P1/6K1 b - -	-2.47
3	8/8/8/3N4/2KB4/5k2/8/8 b - -	2.45
4	8/8/3N1N1k/5K2/6p1/8/8/8 w - -	3.24

Figure 4.2: A table representing the required extracted data

We have now obtained the necessary data to start extracting factors that increase/decrease chances of winning an endgame in chess and are now ready to start computing the machine learning algorithms.

Chapter 5

Model Development & Analysis

In this chapter we will look at how each model stated in chapter 3 performs. To determine the accuracy of each model we will split the data into a training set and a validation set. The training set will contain 80% of the data while the validation set will contain the other 20%. We will use the training set to compute the prediction models and use the validation set to calculate how accurate each model is. Also, we will use the random forest algorithm to see which parameters are the most important to the models in order to get a better idea of which factors best increase one's chances of winning an endgame in chess. We will first start with the linear model. As stated before the parameters used for the models are as follows:

- Material points white
- Material points black
- Passed pawns white
- Passed pawns black
- King activity white
- King activity black
- Player turn

5.1 Developing the Linear Prediction Model

When computing the ML algorithm in python we get the following weights:

Parameter	Weight
Material Points White	1.494792
Material Points Black	-1.734059
White Passed Pawns	0.507920
Black Passed Pawns	-0.153358
King Activity White	0.016266
King Activity Black	0.132987
Player Turn	0.105077

Table 5.1: Linear Model Weights

We see some interesting results here. Firstly, in a perfect scenario we would expect each same factor to have the same weight but a different sign however due to the nature of the data this does not occur. We see that the material points for black is more weighted then it is for white. We see the opposite for the passed pawns where it is more weighted for white. We see that the further black's king is from the centre the bigger the advantage to white is. We should expect the weighting for white's king to be negative as the further away it is the bigger the advantage is for black. The weighting is 0.016 which is positive however this value is close to 0 resulting in less of an impact on the final predicted evaluation.

When plugging the validation set into this prediction model we obtain an accuracy of 79.27%. Although this score isn't terrible it can be improved. We will look at an example of where this linear model falls short:

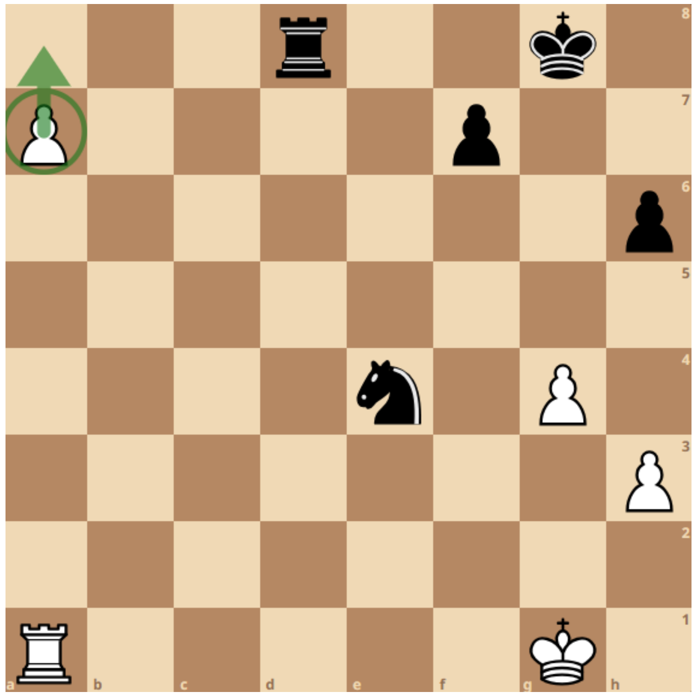

Figure 5.1: Example of where the linear prediction model fails

In this position we see white has a rook and 3 pawns totalling to white's material points equaling 8. Black has two pawns, a rook and a knight totalling to 10 material points. White has one passed pawn while black has none. White's and black's king are both 3 moves away from the centre. It is also white's turn. If we plug this information into our linear prediction model we get the predicted evaluation to be approximately equal to -4.32 predicting black to have a big advantage however the true evaluation for this position is 3.64. From the board above we can see why. As it is white's move the player can simply push the pawn on **a7** forward by one square to promote to a Queen. This one move instantly pushes white's material points from 8 to 16. The linear model could not grasp how important the fact white had a very strong passed pawn as well as the importance of it being white's turn. In order to resolve this, the model could include a feature that could determine how strong a passed pawn is as it can vary in strength. If we had a larger weight for this type of passed pawn the model may of predicted the evaluation correctly.

5.2 Developing the Polynomial Prediction Model

Similar to before we will compute the polynomial ML algorithm in python using the same parameters as before. For this model we will use a degree of 2. Once computed we find that the weights for the prediction model are as follows:

Polynomial Feature	Weight
Material Points White	1.891728
Material Points Black	-2.144717
White Passed Pawns	0.489212
Black Passed Pawns	-0.176228
King Activity White	-0.061275
King Activity Black	0.138152
Player Turn	0.150372
(Material Points White)^2	0.251254
(Material Points Black)^2	-1.053458
(White Passed Pawns)^2	-0.045576
(Black Passed Pawns)^2	-0.042383
(King Activity White)^2	0.012129
(King Activity Black)^2	0.040027
(Player Turn)^2	2.104219

Table 5.2: Polynomial Model Weights

Now we have the weights we can test the accuracy of the model by using the same training and validation data set as before. By doing so we find that the model has an accuracy of 91.14%. A big improvement from before! For simplicity we have only used a degree of 2 however if we chose a degree of 8 the accuracy slightly increases to 91.90%. This is likely due to the nature of the data but as it is only a small increase we will keep this optimal model at a degree of 2. Let's now see an example of where this model falls short.

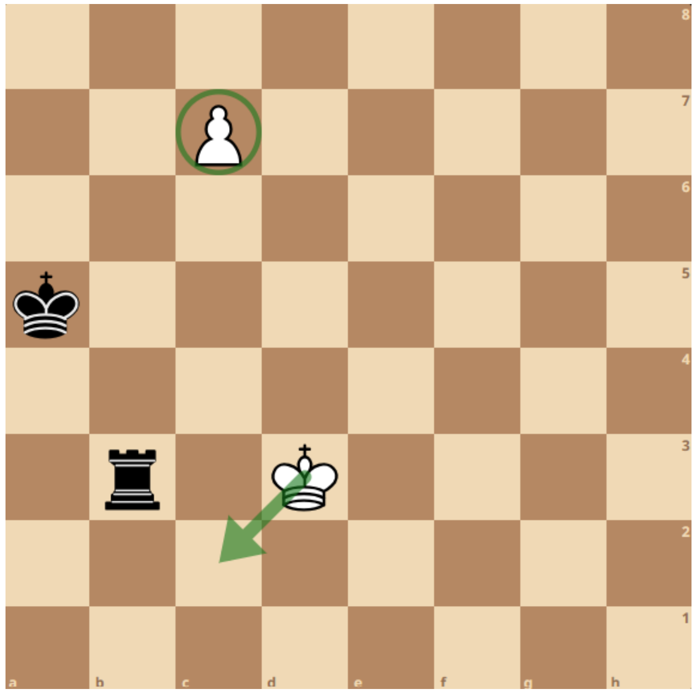

Figure 5.2: Example of where the polynomial prediction model fails

In this position black just has one rook making black's material points 5. White just has one passed pawn while its king is one square from the centre while black's king is 3. It is also white's turn which is evident because the king is in check. If we plug this into our polynomial prediction model we get an evaluation of less than -20, predicting a huge advantage for black. However the true evaluation is 2.37. We saw a similar situation in the linear model where white had a very strong passed pawn which couldn't be stopped from promoting. To get out of the check white's king can move to **c2**. Afterwards, it is impossible to stop white's pawn from promoting to a queen.

It appears both models fail to capture the importance of passed pawns. But why? There could be a few reasons for this. As mentioned before, a passed pawn can vary in strength with the ones we see in Figure 5.1 and 5.2 being much stronger than others in the data set. Another reason could be that there just aren't many positions in the data set where passed pawns are as important. It is a hard task to program a model which can determine how powerful a passed pawn is as chess is a very complex game where only the tiniest details can make a huge difference to the evaluation of a given position.

5.3 Implementing the Bootstrapping Method

Once we compute the bootstrap algorithm using the parameters used in the linear and polynomial prediction models, we get the following means for each parameter from the weight vectors:

Parameter	Mean of Weight Vector
Material (White)	1.492221
Material (Black)	-1.725153
Passed Pawns (White)	0.506282
Passed Pawns (Black)	-0.166653
King Activity (White)	0.012805
King Activity (Black)	0.136612
Player Turn	0.106257

Table 5.3: Mean Weights of Parameters

If we compare this to the weights of the linear prediction model we hardly see any difference at all. By plotting a box plot for each weight vector we can get a better idea of this. We see from Figure 5.3 that for each parameter there is a minor amount of variability for each weight. This uniformity in weight variability implies that the bootstrapping process has provided consistent estimates for the model's parameters across different features. As a result, the model's sensitivity to each parameter is comparable, and no single parameter appears to dominate the variability in the weight estimates. This stability can be advantageous for ensuring robust and reliable model performance.

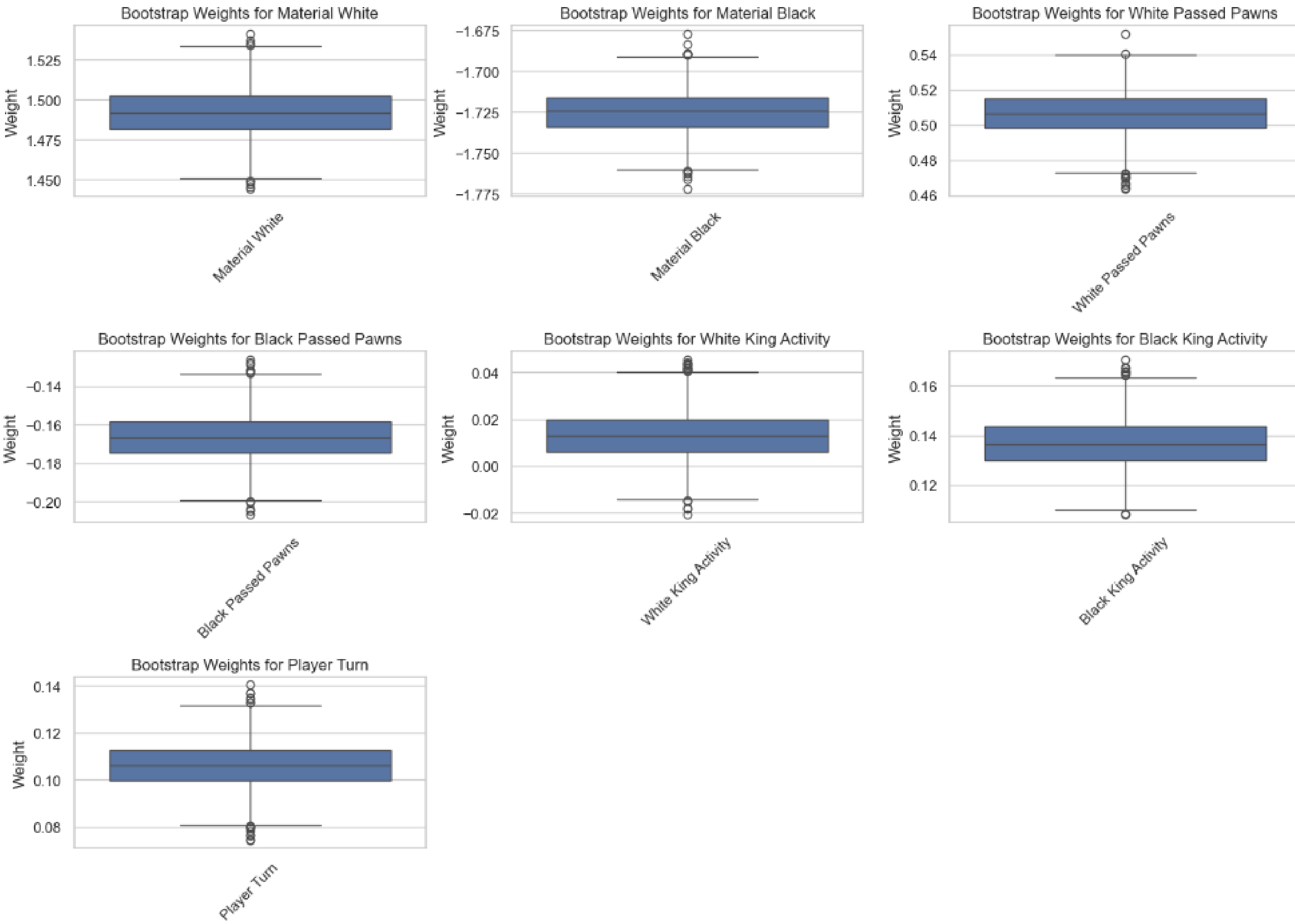

Figure 5.3: Box plots of each weight vector from bootstrapping

5.4 Feature Importance

We will now look at the results from the random forest algorithm to determine the importance of each parameter in the prediction models. As well as looking at each parameter we will also look at a model which contains a parameter for each piece on the chess board (except the king) to see how important each piece is compared to one another. We will first start with the parameters used in the prediction models.

5.4.1 Parameter Set 1

By using the same parameters as in the prediction models and computing the algorithm we get the following results.

Parameter	Importance
Material Points White	40.97%
Material Points Black	35.99%
White Passed Pawns	5.25%
Black Passed Pawns	5.56%
King Activity White	4.89%
King Activity Black	5.34%
Player Turn	2.00%

Table 5.4: Parameter Importance

As one might expect the material points are the most important parameter. This makes sense as it is simple to realise that the more pieces one has the better the chances of winning will be. We see slight variation between white and black's material points. This could be because there are more games in our database in which white has the advantage. In terms of the other parameters we see a much lower importance but less variation between white and black which is a good result as each passed pawn and king activity parameter should have a similar importance independent of what colour piece it is. The lower importance for passed pawns matches as to where the models fall short in being unable to capture the difference between a strong passed pawn or weak one.

5.4.2 Parameter Set 2

We will now look at the importance of all the chess pieces on the board. When computing the random forest algorithm we get the following results:

Piece	Importance
White Pawns	14.22%
Black Pawns	15.51%
White Knights	10.75%
Black Knights	8.52%
White Bishops	11.83%
Black Bishops	11.21%
White Rooks	13.59%
Black Rooks	14.36%

Table 5.5: Piece Importance

We see some interesting results here. First thing to note that except for the knights each of the same piece but different colour has a similar importance which is an expected result. The pawns are the most important in front of the rooks then the bishops and then the knights. It would make sense that the rook is more important then the bishop and knights as it is worth 5 material points instead of 3. There are more pawns at the start of a game hence indicating why they are the most important, especially due to the fact of a passed pawn.

5.5 A Case Study of Piece Combinations

In this final section of analysis, we will explore and analyse different types of chess endgames to see if certain pieces/combination of pieces fare better than others. This is important to consider as one player will have an advantage knowing if a certain position is in their favour or not before potentially reaching it. We will go over the following two types of endgames:

- knight vs bishop (no rook)
- knight + bishop vs rook (no extra pieces except pawns)

As we saw in the previous chapter the material points were the biggest factor in increasing/decreasing one's chances of winning a chess endgame thus in this analysis we will only look at positions in which the material points for white and black are equal in order to remove the effect of material points.

5.5.1 Knight vs Bishop

The knight and bishop are two pieces with very different functions. Not only is the shape they can move in completely different, the distances they can travel as well as the knight being able to jump over pieces also make each piece very different. They are both given a material point of 3 hence they are practically identical in strength if optimised. It is known that when chess positions are closed the knight can be very good at getting into tight spaces while the bishop can easily get stuck behind pawns. This type of position can occur in endgames.

In other chess endgames the position is very open and free with each piece having many directions to travel in. This benefits the bishop over the knight due to the fact of how far it can travel. We can see this from the position below:

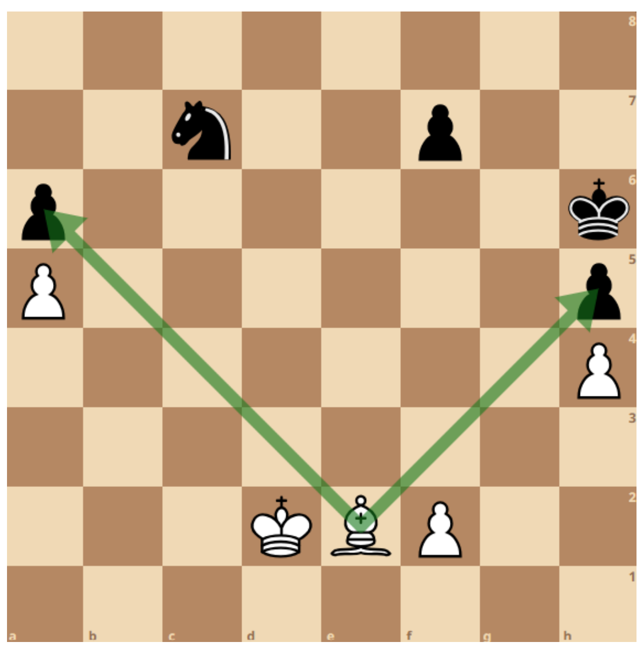

Figure 5.4: A bishop proving to be stronger than the opponent's knight

In this endgame we can see that the material points for each player are equal with the only difference being a bishop instead of a knight. The board is very open with the main thing to spot is the fact that the white bishop is attacking 2 pawns at the same time. The knight and king are unable to move without losing one of the pawns. Black's knight is also too far from any action and takes longer to reach it while the bishop can get from one side to the board to another in just one move. This highlights the strength of the bishop when we have an open endgame such as this one. The true evaluation of this position over +5 no matter whose turn it is proving how white's bishop is much stronger than black's knight.

In another example below we will see a different scenario where a knight can prove stronger than a bishop.

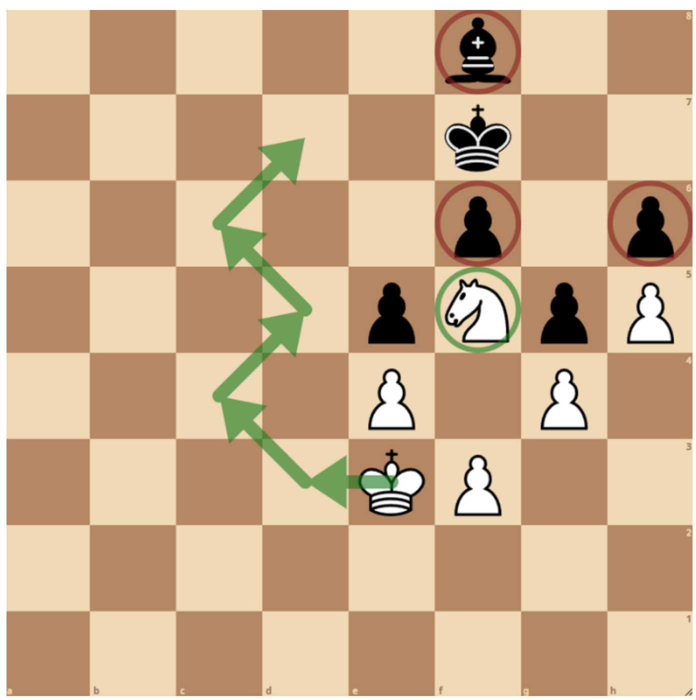

Figure 5.5: A knight proving to be stronger than the opponent's bishop

This example is a bit more challenging to visualise hence why the arrows are drawn. We see that black's bishop cannot attack any of white's pawns while white's knight can attack black's pawns. White's plan here is to march the king up the board to help the knight. Black's options are limited and eventually black's king won't be able to move forward due to white's king and will have to go back giving white the advantage. In simpler terms, white's knight opposes attacking threat while black's bishop has none. This example shows how a knight can be more powerful than a bishop in an endgame which is evident by an evaluation of over +5 if it is white's move. So which piece is better overall in an endgame? If we simply look at the dataset we can find the total percentage of games in which a bishop would be beating a knight.

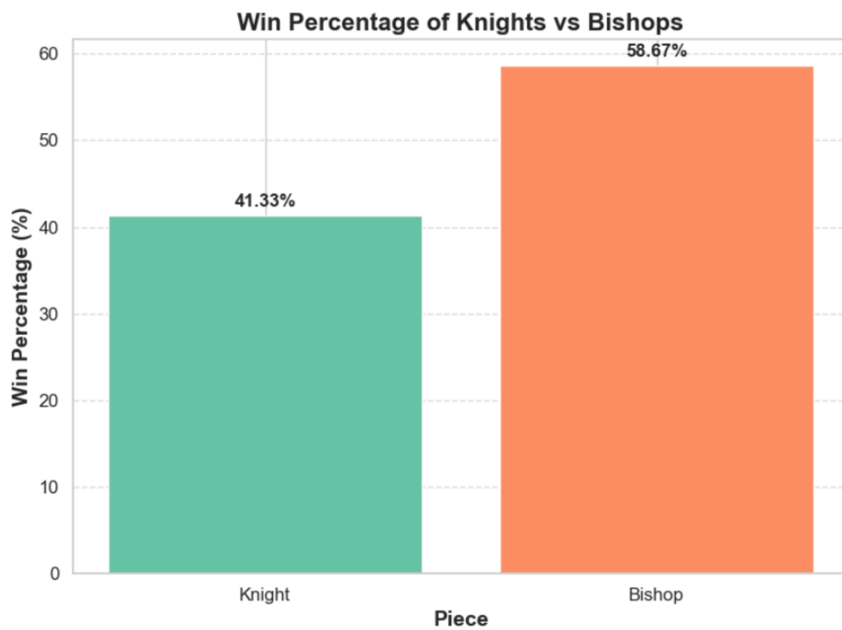

Figure 5.6: Comparison of endgames involving a knight vs a bishop

We see that over 58% of the games gives the bishop an advantage. In general we see that the bishop is actually more powerful than the knight in a chess endgame. It is important to know that there are a huge amount of random chess positions which definitely aren't all included in the database however based on the data provided we do see that in general, the bishop is stronger than the knight.

5.5.2 Knight and Bishop vs Rook

Another interesting combination one might come across in a chess endgame is a knight and bishop against a rook. An important thing to note is that a knight and bishop is worth 6 points of material while a rook is worth 5 so we will be looking at games in which the player with the rook has one extra pawn. There is a lot of debate as to which combination is better. The bishop or knight on its own is undoubtedly worse then a rook however as a team they put up more of a fight. As we are looking at two pieces this gives them the power to 'team-up' against the single rook. The power of being able to control a certain square with two pieces is a very strong tactic in a chess endgame. The rook on the other hand is a very strong piece even on its own. At the start of the game its options are limited being stuck in the corner however in an endgame it can become extremely strong with the board freeing up.

By extracting these endgames and plotting as before we find:

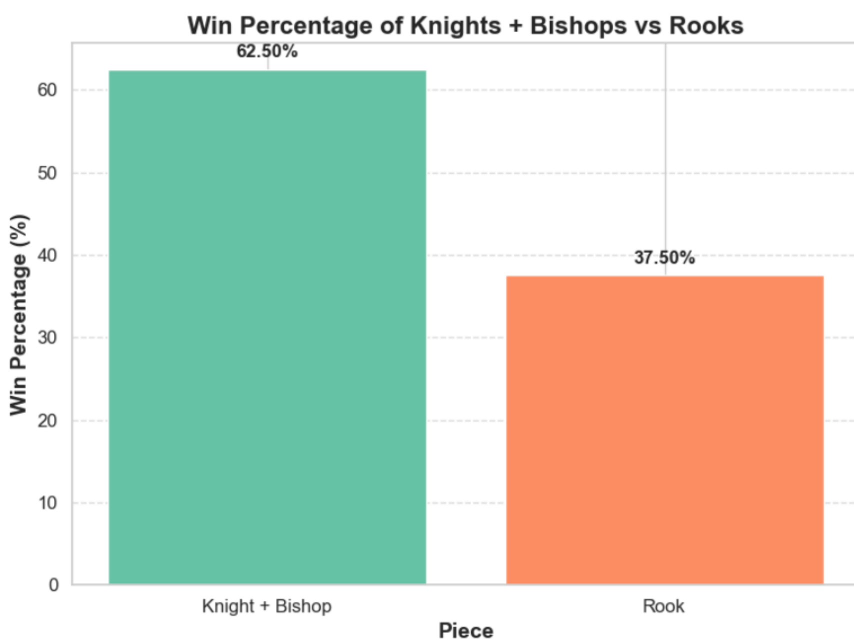

Figure 5.7: Comparison of endgames involving a knight & bishop vs a rook

In this analysis we see some interesting results. Although there is a lot of debate as to which combination is better it appears that the knight and bishop combination is much stronger than a rook with an extra pawn.

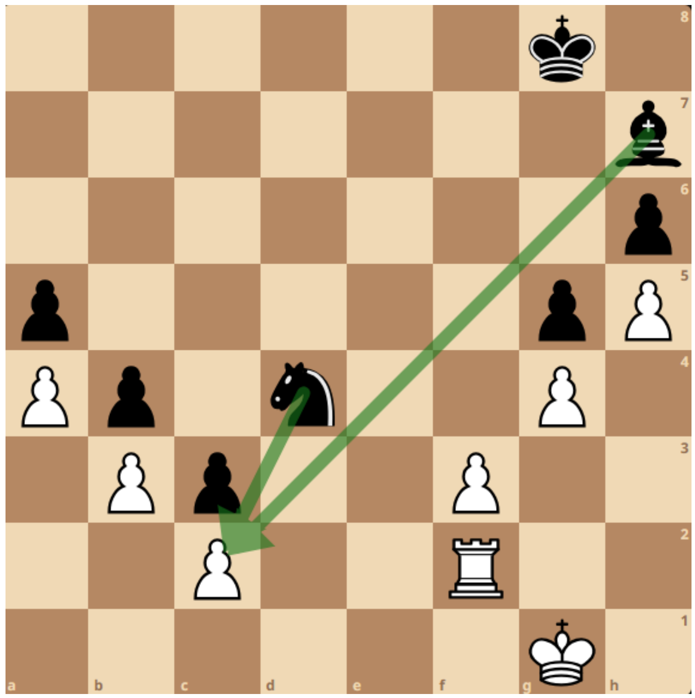

Figure 5.8: An example where a knight and bishop is stronger than the opponents rook

We see in the position above that black's knight and bishop and both attacking the same pawn together. White's rook is hopeless in being able to defend the pawn against the two pieces. Once the pawn falls, white's whole position will fall with it. This is matched with an evaluation of less than -6 no matter if it is black's or white's move. This example is one of many examples that highlights the strength of the bishop and knight combination over the single rook.

Chapter 6

Conclusion

Overall in this report we have been able to identify the key factors that increase ones chances of winning a chess game from the endgame. The highest being the material points however, we showed examples in which the number of passed pawns or the king's activity also helped increase ones chances. By combining all of these factors, we computed a variety of different prediction models in order to identify which player was winning in a given position. The most accurate model was the polynomial with an accuracy of around 92%. As touched on before this could be improved by defining the strengths of different passed pawns. Although using machine learning is not as accurate as the methods used by Stockfish, it is still a unique, interesting and overall successful approach into evaluating chess positions.

As well as that we looked into a case study of different piece combinations when the material points were equal for each player. We found compelling new results, one being that in general a bishop is stronger than a knight and another being that a knight and bishop together are stronger than a rook with an extra pawn. These results can prove to be useful to help a player figure out whether or not they are in a winning position. Let's look at one final example of a random chess position and use what we have discovered to see what the best plan for the player with the white pieces is.

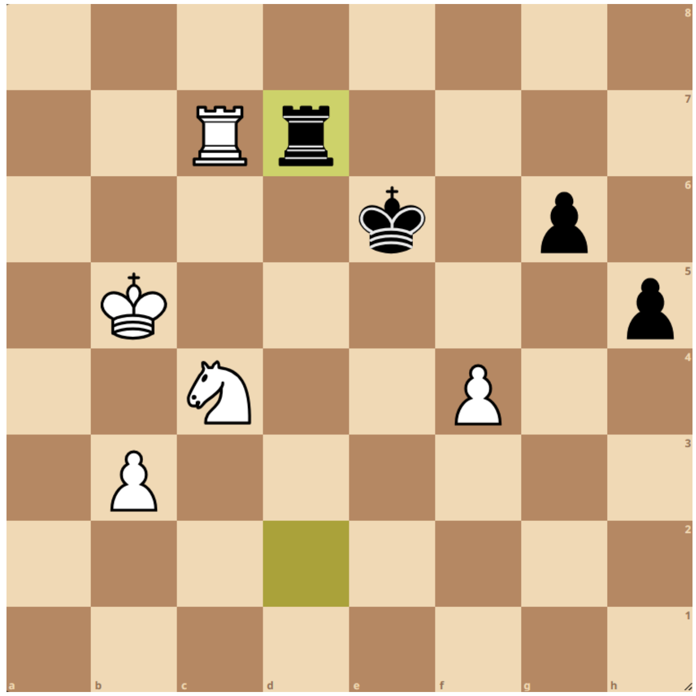

Figure 6.1: A random endgame position

So what can we see here? We see black has one pawn on the right hand side of the board which is a threat as it could promote. However, white has an extra 3 points of material due to the knight. White has an option as it is their turn. Will taking black's rook help increase the winning chances? The answer is yes. Once the rooks are off the board white's knight becomes extremely strong and although black has a passed pawn, white's knight can stop it and eventually white can use one of the last two pawns to push for promotion.

Ultimately, this research shows that an analytical approach to evaluating endgame positions can significantly increase a player's ability to identify winning opportunities to help make accurate strategic decisions.

Bibliography

[1] **Salar Jung Museum** (n.d.) A Game of Thrones: How Chess Conquered the World. https://artsandculture.google.com/story/a-game-of-thrones-how-chess-conquered-the-world-salar-jung-museum/fgUhNlxUQVZ2Kg?hl=en (Accessed: 8 August 2024).

[2] **Wikipedia** (2024) Chess in Europe. https://en.wikipedia.org/wiki/Chess_in_Europe (Accessed: 29 July 2024).

[3] **Goodreads** (n.d.) Chess is everything: art, science, and sport. https://www.goodreads.com/quotes/783943-chess-is-everything-art-science-and-sport (Accessed: 10 August 2024).

[4] **Lichess** (n.d.) Lichess: Free online chess. https://lichess.org/ (Accessed: 19 August 2024).

[5] **Shutterstock** (n.d.) Chess piece relative value chart (classic). https://www.shutterstock.com/image-vector/chess-piece-relative-value-chart-classic-1865156923 (Accessed: 20 August 2024).

[6] **Wikipedia** (2024) Deep Blue (chess computer). https://en.wikipedia.org/wiki/Deep_Blue_(chess_computer) (Accessed: 20 August 2024).

[7] **Kiernan, P.** (n.d.) Which is the greater number: the atoms in the universe or the number of chess

moves?. https://www.liverpoolmuseums.org.uk/stories/which-greater-number-of-atoms-universe-or-number-of-chess-moves (Accessed: 20 August 2024).

[8] **apgoucher** (2021) The neural network of the Stockfish chess engine. https://cp4space.hatsya.com/2021/01/08/the-neural-network-of-the-stockfish-chess-engine/ (Accessed: 20 August 2024).

[9] **Mohammad, H.** (2022) Game theory: How Stockfish mastered chess. https://blogs.cornell.edu/info2040/2022/09/30/game-theory-how-stockfish-mastered-chess/ (Accessed: 20 August 2024).

[10] **SimilarWeb** (n.d.) Chess.com competitors. https://www.similarweb.com/website/chess.com/competitors/(Accessed: 17 August 2024).

[11] **Lichess** (n.d.) Lichess database: Evaluations. https://database.lichess.org/#evals (Accessed: 17 August 2024).

[12] **Wikipedia** (2024) Chess endgame. https://en.wikipedia.org/wiki/Chess_endgame (Accessed: 18 August 2024).

[13] **Chess.com** (n.d.) Passed pawn. https://www.chess.com/terms/passed-pawn (Accessed: 12 August 2024).

[14] **Chris3606** (n.d.) Measuring distance: Grid components. https://chris3606.github.io/GoRogue/articles/grid_components/measuring-distance.html (Accessed: 16 August 2024).