Phase 4 Report

Marco Rossi, Rylan Mahany, Gavin Harrold, Gabe Frahm

Ruthless Frogs (Group 9)

Dataset: https://www.kaggle.com/datasets/arevel/chess-games

What Information are we expecting to find

Openings will be more varied in higher ELO. Experienced players know more openings and Black's win rate will be lower at lower ELO than it is at higher ELO because lower ELO players struggle to play defensively.

Data Cleaning

A column that needed data cleaning was 'Result'. Initially, the dataset represented '1-0' as a Win for white, '0-1' as a win for black, and '½-½' as a stalemate. This is difficult for us to categorize properly, considering some analysis platforms interpreted these as dates, so we needed to figure out a way to represent it more expressively. To fix this issue we decided to make the change from '1-0' to 'W', representing a victory for White, the change from '0-1' to 'B', representing a victory for Black, and the change from '½-½' to 'S', representing a Stalemate. This way the result data can be easily categorized and used in later graphical displays or analysis.

Before:

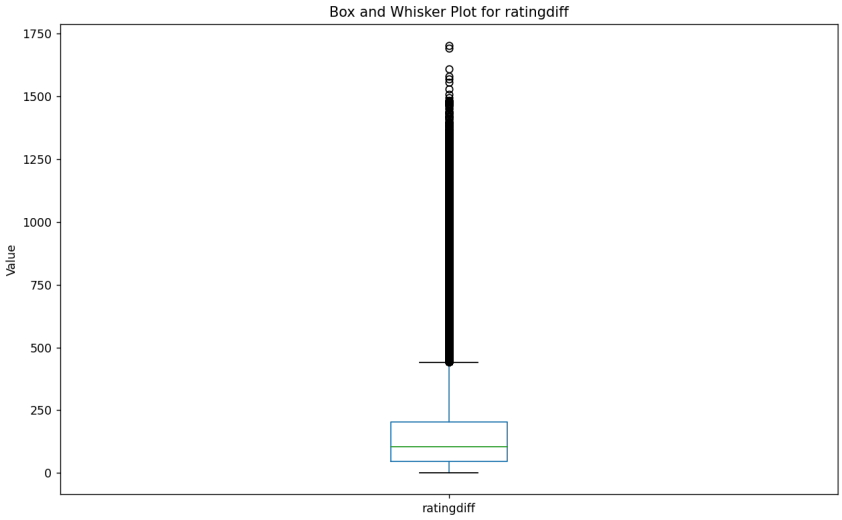
	event	white	black	result	utcdate	utctime
1	Classical	eisaaaa	HAMID449	1-0	2016.06.30	22:00:01
2	Blitz	go4jas	Sergei1973	0-1	2016.06.30	22:00:01
3	Blitz tournament	Evangelistaizac	kafune	1-0	2016.06.30	22:00:02
4	Correspondence	Jvayne	Wsjvayne	1-0	2016.06.30	22:00:02
5	Blitz tournament	kyoday	BrettDale	0-1	2016.06.30	22:00:02
6	Blitz tournament	lucaseixasouza	diguim	0-1	2016.06.30	22:00:02
7	Blitz tournament	RENZZO77	HeadlessChicken	0-1	2016.06.30	22:00:02
8	Blitz tournament	ipero	Bayern123	1-0	2016.06.30	22:00:02
9	Blitz tournament	Loginov19510410	Kereshu	0-1	2016.06.30	22:00:02
10	Blitz tournament	Shambobala	cernunnoss	1-0	2016.06.30	22:00:02
11	Classical	DARDELU	chess4life54	0-1	2016.06.30	22:00:01
12	Classical	Yaqyaqs	S888888N	0-1	2016.06.30	22:03:40
13	Classical	fabikim	sereno	1-0	2016.06.30	22:00:02
14	Blitz tournament	IZDenisZI	BoBo93	1-0	2016.06.30	22:00:02
15	Blitz tournament	lasha-fero	ildivinojohnny	1-0	2016.06.30	22:00:02

After:

	event	white	black	result	utcdate	utctime
1	Bullet	mule50	speedator	B	2016.07.29	09:29:46
2	Bullet	Gorbenko	QueenLover	W	2016.07.29	09:30:27
3	Classical	Frimos	pakha	B	2016.07.29	09:30:46
4	Blitz	armitagefr59	medion3000	W	2016.07.29	09:31:47
5	Blitz	Arya27	ant117	B	2016.07.29	09:32:00
6	Bullet	speedator	mule50	B	2016.07.29	09:32:41
7	Bullet	tomdecuigniere	jalec	W	2016.07.29	09:33:05
8	Blitz	christopheh	jbus	B	2016.07.29	09:34:46
9	Blitz	ashish1990	alex_234	W	2016.07.29	09:36:37
10	Bullet tournament	Philipp_Stuttgart	TheSettler	W	2016.07.29	09:40:09
11	Bullet tournament	CrusherDesu	Obanta	W	2016.07.29	09:41:22
12	Bullet tournament	HectorDaniel	liklak	W	2016.07.29	09:41:38
13	Bullet tournament	pohyi	bernes	W	2016.07.29	09:43:13
14	Bullet	c48	adamamrich	W	2016.07.29	09:44:22
15	Bullet	zobi	MySonKing	W	2016.07.29	09:46:49

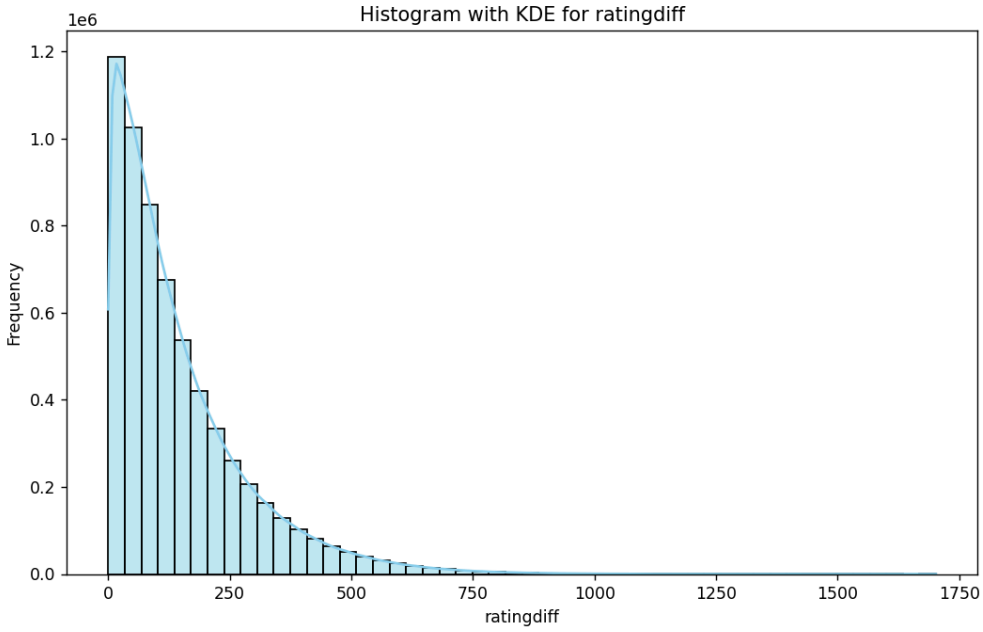
Outliers

We needed to create a new column called ‘ratingdiff’ to find outliers in this dataset. This column will show the total difference in elo between two players if the absolute value is taken. The reason this column is important is so we can show if any players were incorrectly matched up against each other, this would mean that their difference in elo is extremely large compared to other games played. Take this box and whisker plot of that column for example:



Q1: 46.0 | Q2: 106.0 | Q3: 204.0 | IQR: 158.0 | Lower Bound: -191.0 | Upper Bound: 441.0

From this boxplot, you can see evidence of uneven matchups in games. The median difference in elo is a rating of 106 with the upper and lower bounds being [-191, 441]. Since there are no negative total differences between games nothing can go below the lower bound, but as for the upper bound anything that exceeds a total difference of 441, would be considered an outlier. In this case, there are many, and even some extreme cases that exceed even a difference of 1500. The worst case recorded was a player rated 1200 going up against a player with a rating of 2902. This is the equivalent of your below-average chess player going up against an extremely high-ranking grandmaster. Records like this one are unfair matchups resulting in a skewed dataset. This can be shown more clearly with a histogram representation:



From this histogram, you can see that the majority of games played are below the upper bound of 441. Due to the extreme outliers, this histogram becomes right-skewed which could make taking the average total difference less accurate.

Handling Missing Data

Our data set had a handful of missing data that needed to be addressed. The most prominent being the ‘WhiteRatingDiff’ and ‘BlackRatingDiff’ columns having empty values. These columns represented the difference in elo for black and white according to the result of the match (How much is gained/lost from a game. This is different from ratingdiff which is the difference between both player's elo). To fix this issue we used a method of data imputation to replace these blank values with a more suitable one. First, each game or record has a column called ‘gameelo’ which represents the average elo for that match. Then using that column to group, we wrote a query that finds the average rating difference for a black win, black loss, black stalemate, white win, white loss, and white stalemate. This way we have a list for all cases of a specific game elo. For example, if we were missing data for a game that had a game elo of 1000, we would look at the average black/white difference rating for other games with a game elo of 1000, and based on the result, we would update its respective value. So if black lost that game it would be updated with the average amount lost for black from other games with the same game elo, and vice versa.

Another part of the dataset that had missing values was the result column containing a ‘*’ character. This was seemingly due to server issues causing both players to abandon, thus terminating the match. This did not affect many columns so records containing this value were removed.

In another instance, the ‘timecontrol’ column contained a ‘-’ character due to the game type of correspondence not having any official form of time control in play. The time control is defined as the following, time of the game for each player in seconds plus the number of seconds before the player's clock starts ticking in each turn. The game of correspondence chess tends to

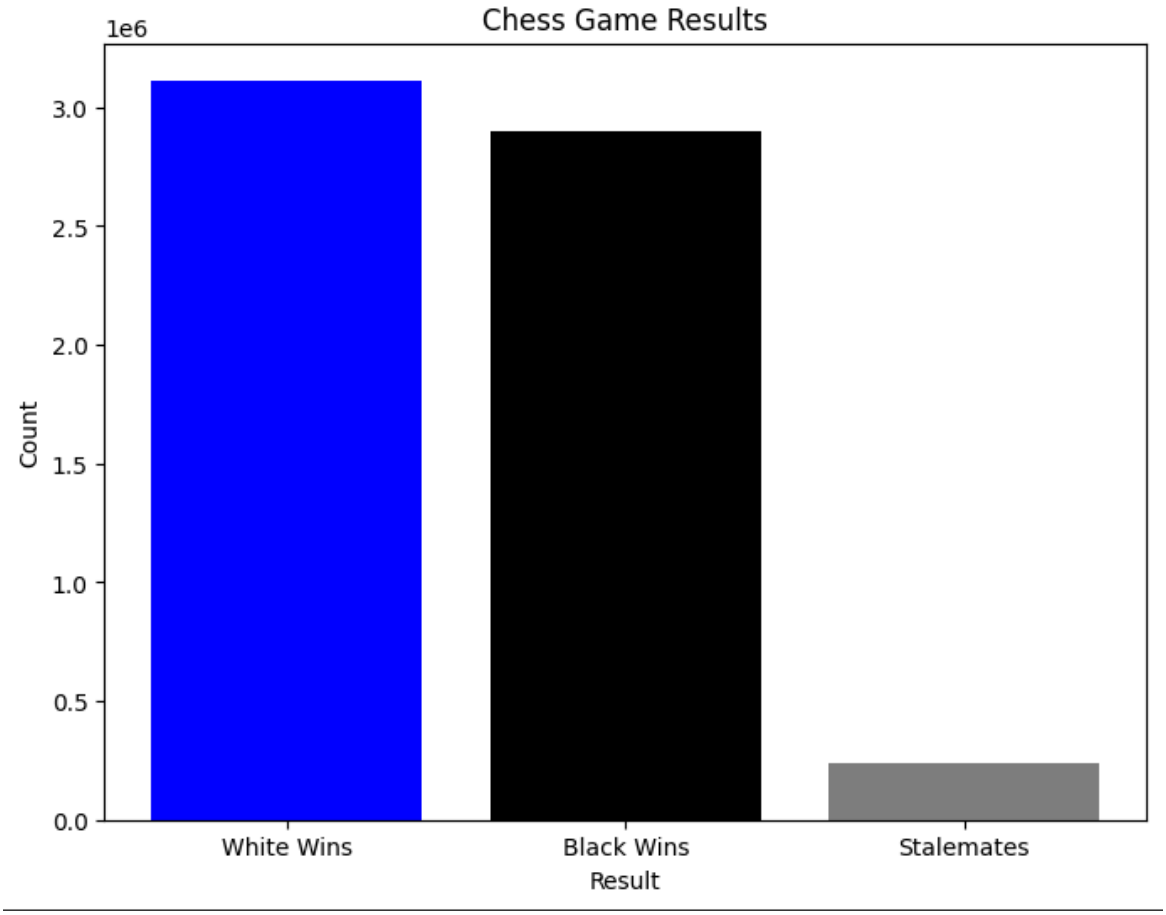
be played around one move per day so time control cannot be applied here. Therefore we decided to use the null value to represent the time control in a game of correspondence chess.

Removed fields

We chose to remove the ‘utcdate’ field from our dataset. We chose to remove this for two main reasons. The first reason is that we likely were not going to need this to answer the questions we were asking. The date shouldn’t affect the outcome of the games. Secondly, it was a problematic field because there was no way to know the accuracy of it since players could be playing from different time zones. For example, an Australian player and an American player would be playing on two completely different days, and there was no way to determine how the date was recorded. Since there is no indication in the data of a particular player’s timezone, this column was largely useless.

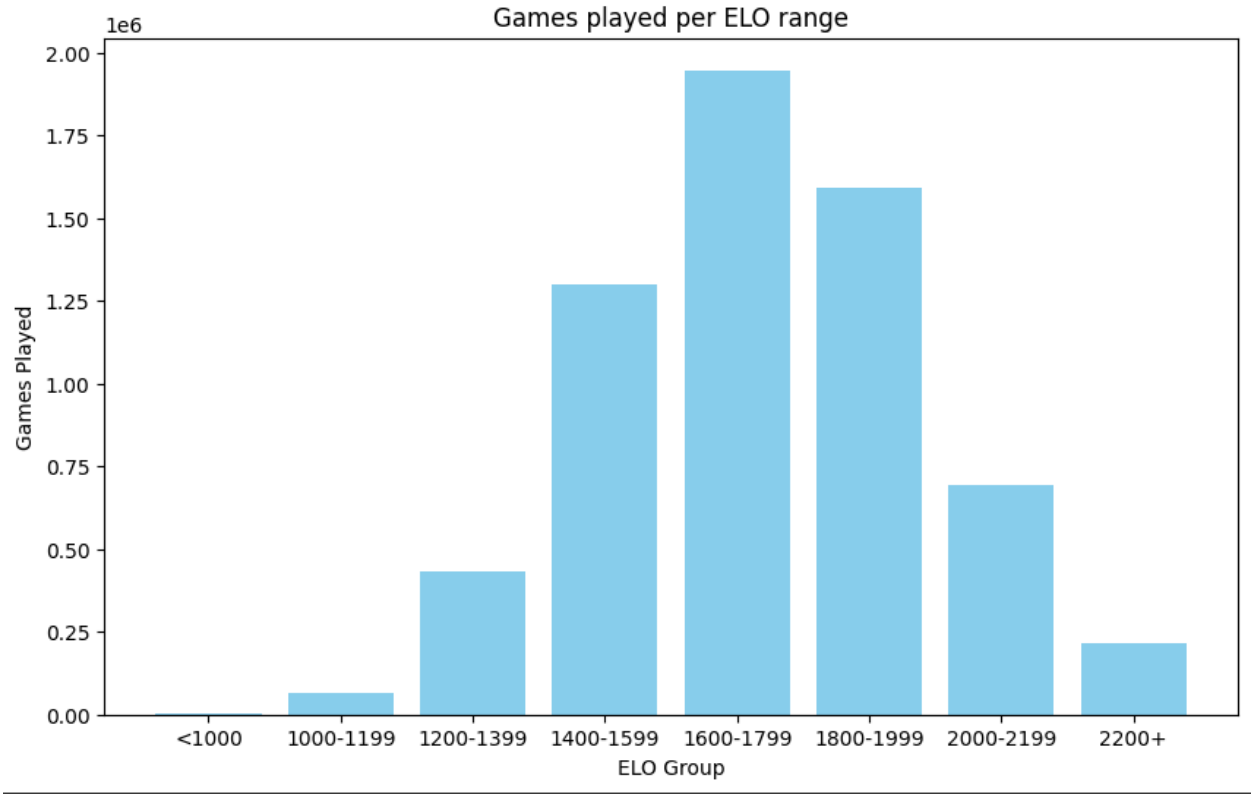
EDA

Game Results across all ELOs



This chart represents the number of games won (by millions) across all ELOs. As expected, White wins more often as they have the first move, which grants them the ability to dictate how the rest of the game will be played based on their opening, as well as always being one move ahead of their opponent.

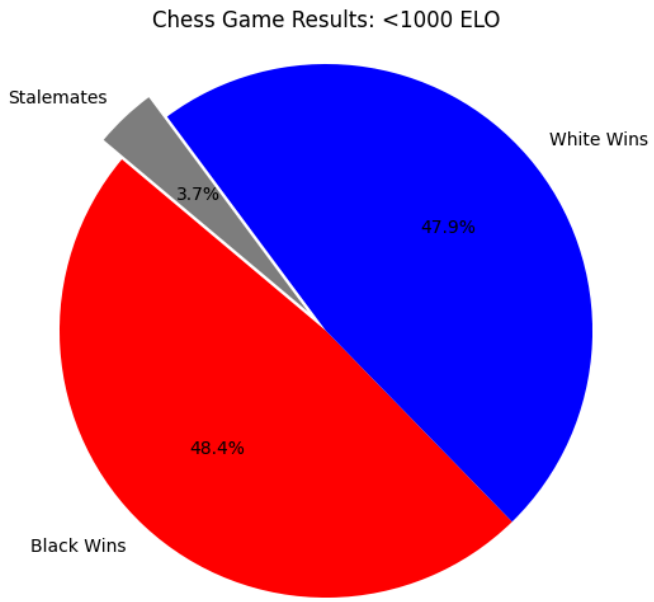
Games Played Per ELO Group

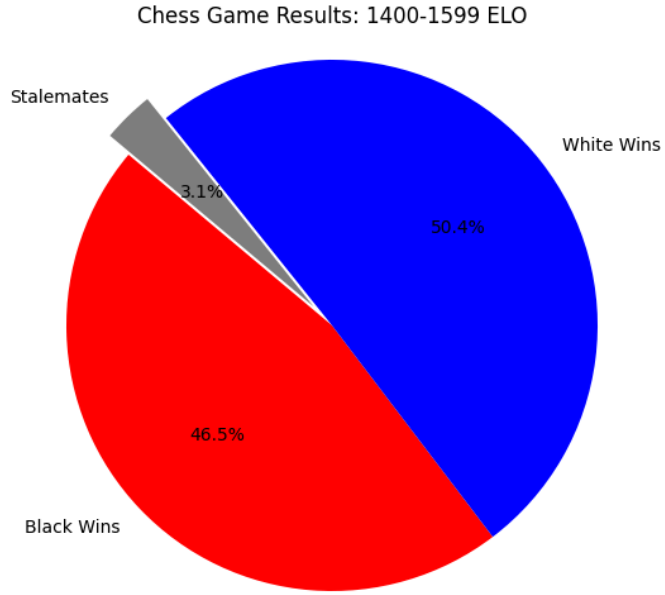


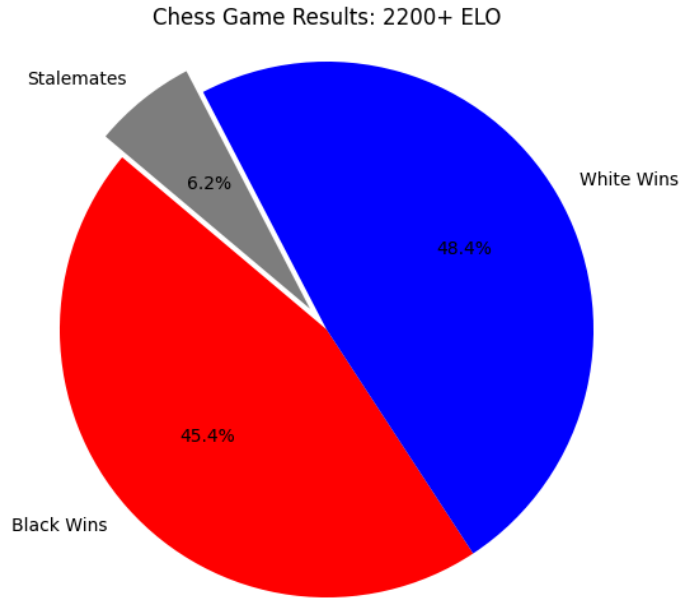
This chart represents the number of games played (by millions) per ELO group. This gives us a good idea of the distribution of players' ELO as well as how many games they are playing. We can see that a majority of games played, and by extent player ELOs fit into the 1600-1799 category, while the low and high ELO groups fall significantly. It makes sense that the highest ELO group has more games played than the lowest ELO group, as they need to play a lot to achieve and maintain that ELO. They likely also have a stronger passion for the game and want to play more since they took all the time to achieve that skill level.

Game result percentage, by specific ELO groups

(low vs low, medium vs medium, high vs high)

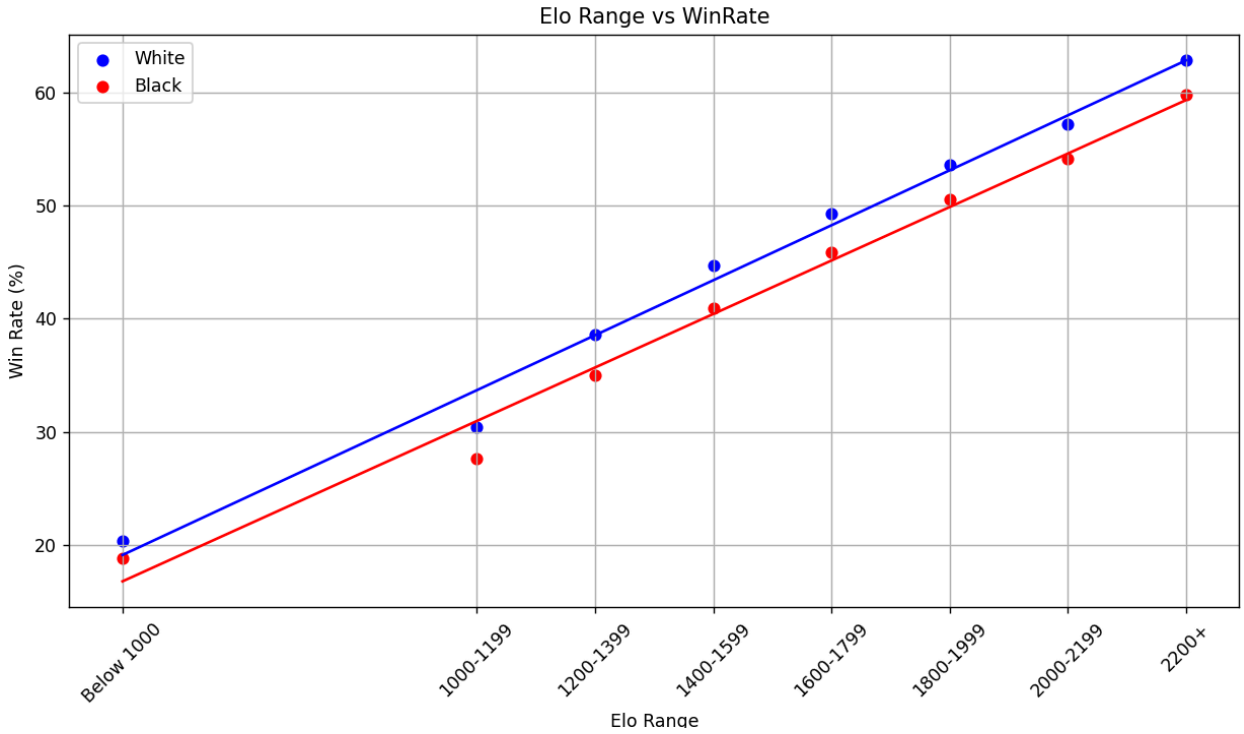






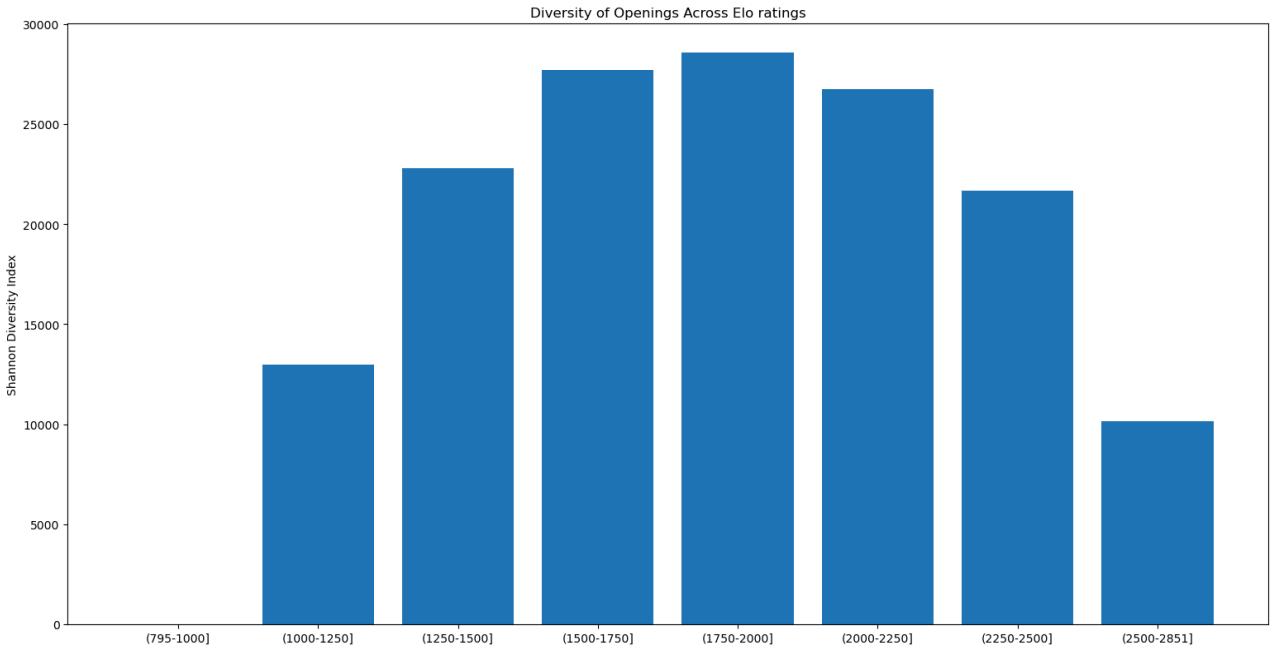
Surprisingly, In low ELO, black tends to win slightly more than white (.5%). Then in the middle ELO, White has a significant advantage over Black (3.9%). In high ELO games, White is favored 3%, but many more games end in stalemates compared to low and medium ELO games.

ELO Range vs Win Rate



This graph represents the Win rate by ELO group per side. As shown, lower ELO players are likely to win in about 20% of games vs all players (not just the ones in their skill group, as seen in the pie charts above) losing or stalemating in about 80% of the games. Higher ELO players are likely to win in about 60% of their games (same situation). This shows that there is a direct relationship between ELO and win rate, better players tend to win more. This also supports previous findings that players in a higher ELO range need to win more to maintain their high ELO. The regression lines shown as blue for White and red for Black show that at all skill levels, White is most likely to win, regardless of the ELO group.

Diversity of Openings across ELO Groups



The above graph shows the diversity of openings used across various binned elo ratings. We used the Shannon Diversity Index to measure the diversity, which is typically used in ecology, but the principles are still applicable here. This graph could be influenced by the fact that there are many more games in the middle ranks, leading to more played openings since there are over 1,000 possible openings. This is a mostly normal distribution, with a possible explanation of low-ranked players attempting to play by "the book," mid-ranged players trying out new openings to broaden their repertoire (although the most common openings are still standard openings), and high-ranked players finding a better opening that may be more difficult to utilize. These claims are also supported by the below table, based on data used in the histogram.

Elo Range	Opening	Number of Occurrences	Proportion
(1000-1250]	Van't Kruijs Opening	5,200	4.27%
(1250-1500]	Van't Kruijs Opening	28,017	3.04%
(1500-1750]	Van't Kruijs Opening	53,664	2.42%
(1750-2000]	Modern Defense	38,606	1.85%
(2000-2250]	Modern Defense	13,377	1.75%
(2250-2500]	Old Benoni Defense	2,196	1.67%
(2500-2851]	Old Benoni Defense	196	1.96%

A possible explanation for the proportion continuously going down, yet the diversity also decreasing could be that although the most common opening was used less, there were fewer openings used in total.

Phase 2 Conclusion

In conclusion, we were able to discover multiple interesting relationships in our data, with many coming from relating players of different ELOs to one another. We observed a change in win rate across both Black and White pieces as we shifted from low ELO to high ELO and visualized the distribution of ranks by graphing the number of games played in various clusters of ELO. While in our hypothesis we stated that Black would have a higher win rate in higher

ELO rather than lower ELO due to lower-rated players struggling to play defense, the results of our data surprisingly did not demonstrate this. Another important part of our hypothesis was the diversity of openers at different ELOs. For example, we predicted that higher ELOs would have a higher diversity of openings, as they are more knowledgeable of the game and therefore openings. The data does not necessarily support this claim. The highest diversity of openers was in the 1750-2000 ELO range: where most players fall. The very highest-ranked players in the dataset had less diversity in their openings, as did the very lowest-ranked players. Overall, performing Exploratory Data Analysis on our dataset resulted in interesting and both expected and unexpected results. These results will allow us to evaluate the effectiveness of our hypothesis, as well as challenge our way of thinking for next time.

Phase 3 Begins

Technique:

Neural Network (Supervised)

Why:

We picked this technique because we wanted to see how much of an impact the opening move made on the outcome of a chess match. Using the ‘Eco’ or ‘OpeningMove’ column, as indicator variables, we can compare their effectiveness to other variables concerning the predictions of the Neural Network using sensitivity analysis. Note that the ‘Eco’ column is a more generalized encoding of the most common chess openings, whereas the ‘OpeningMove’ column is the first move played in each game.

Process:

To perform a neural network on our dataset we first needed to standardize and normalize all data that we deemed could be properly normalized and were important to the dataset. The numeric variables we chose were ‘WhiteElo’, ‘BlackElo’, ‘WhiteRatingDiff’, ‘BlackRatingDiff’, ‘Gameelo’, ‘RatingDiff’, ‘timestamp’, ‘starttime’, and ‘incrementtime’. We performed Min-Max normalization to put all these values within the range of [0, 1]. Next, we also needed to assess the available indicator variables and put them in the range of [0, 1]. The indicator variables and their respective values are listed below:

ECO (Chess opening move encoding):

- A00-A49 → 0
- A50-A99 → 0.1
- B00-B49 → 0.2
- B50-B99 → 0.3
- C00-C49 → 0.4
- C50-C99 → 0.5
- D00-D49 → 0.7
- D50-D99 → 0.8
- E00-E49 → 0.9
- E49-E99 → 1

Event (Game type):

- Classical → 1
- Classical Tournament → 0.9

- Bullet → 0.7
- Bullet Tournament → 0.6
- Blitz → 0.4
- Blitz Tournament → 0.3
- Correspondence → 0

Termination (How the game ends):

- Normal → 1
- Abandoned → 0
- Rule Infraction → 0.3
- Time Forfeit → 0.7

OpeningMove (First move made)

- A3 → 0
- A4 → 0.05
- B3 → 0.1
- B4 → 0.15
- C3 → 0.2
- C4 → 0.25
- D3 → 0.3
- D4 → 0.35
- E3 → 0.4
- E4 → 0.45

- F3 → 0.5
- F4 → 0.55
- G3 → 0.6
- G4 → 0.65
- H3 → 0.7
- H4 → 0.75
- Na3 → 0.85
- Nc3 → 0.9
- Nf3 → 0.95
- Nh3 → 1

Result (Target Variable):

- W → 1
- B → 0
- S → 0.5

After having all the variables standardized properly, we picked the target variable as the 'Result' column to see if the Neural Network can predict the outcome of chess games. Then we ran the preprocessed dataset through a neural network and used sensitivity analysis to discover which variables had the highest effect on the prediction.

Result:

Below are the last five epochs from the Neural Network, and the result of the sensitivity analysis:

Epoch 45/50

19547/19547 ━━━━━━━━━━━━━━━━━━━━ 12s 615us/step - **accuracy: 0.6306 - loss: 0.6157**

Epoch 46/50

19547/19547 ━━━━━━━━━━━━━━━━━━━━ 12s 615us/step - **accuracy: 0.6313 - loss: 0.6151**

Epoch 47/50

19547/19547 ━━━━━━━━━━━━━━━━━━━━ 12s 612us/step - **accuracy: 0.6310 - loss: 0.6148**

Epoch 48/50

19547/19547 ━━━━━━━━━━━━━━━━━━━━ 13s 683us/step - **accuracy: 0.6296 - loss: 0.6152**

Epoch 49/50

19547/19547 ━━━━━━━━━━━━━━━━━━━━ 13s 647us/step - **accuracy: 0.6312 - loss: 0.6143**

Epoch 50/50

19547/19547 ━━━━━━━━━━━━━━━━━━━━ 13s 671us/step - **accuracy: 0.6301 - loss: 0.6150**

Feature sensitivities (In order of importance):

- blackelo
- whiteelo
- starttime
- openingmove
- incrementtime
- gameevent
- timestamp
- gameelo
- ratingdiff

- eco
- termination

Initially, we used the columns ‘WhiteRatingDiff’ and ‘BlackRatingDiff’ in the Neural Network, but upon investigating the results, we decided that those columns were too trivial. To further explain, the rating difference is the amount of elo lost or gained at the end of a match. It would be extremely trivial to see which side gained or lost points, and the result could be easily determined. Running this would provide perfect accuracy and almost no loss, so this was considered an anomaly and did not allow us to interpret meaningful information about other variables. These columns were removed for the actual run of the Neural Network which produced an accuracy and loss of about 63% and 61.5% respectively. This means that the model could predict the outcome of the chess game 63% of the time, but had around a 61.5% margin of error in those predictions.

Regarding the sensitivity analysis, there were some interesting results. Firstly the elo should be the highest due to a higher elo player winning against a lower elo which it is. The next most important variable was the starting time of the game (total length of the game at the start) followed by the opening move. The length of the game and the first initial move had the largest impact out of all the other variables. If we were to look at how the encoding performed, its effect on the neural network was almost negligible and provided little support during its predictions.

Technique:

K-Means Clustering (Unsupervised)

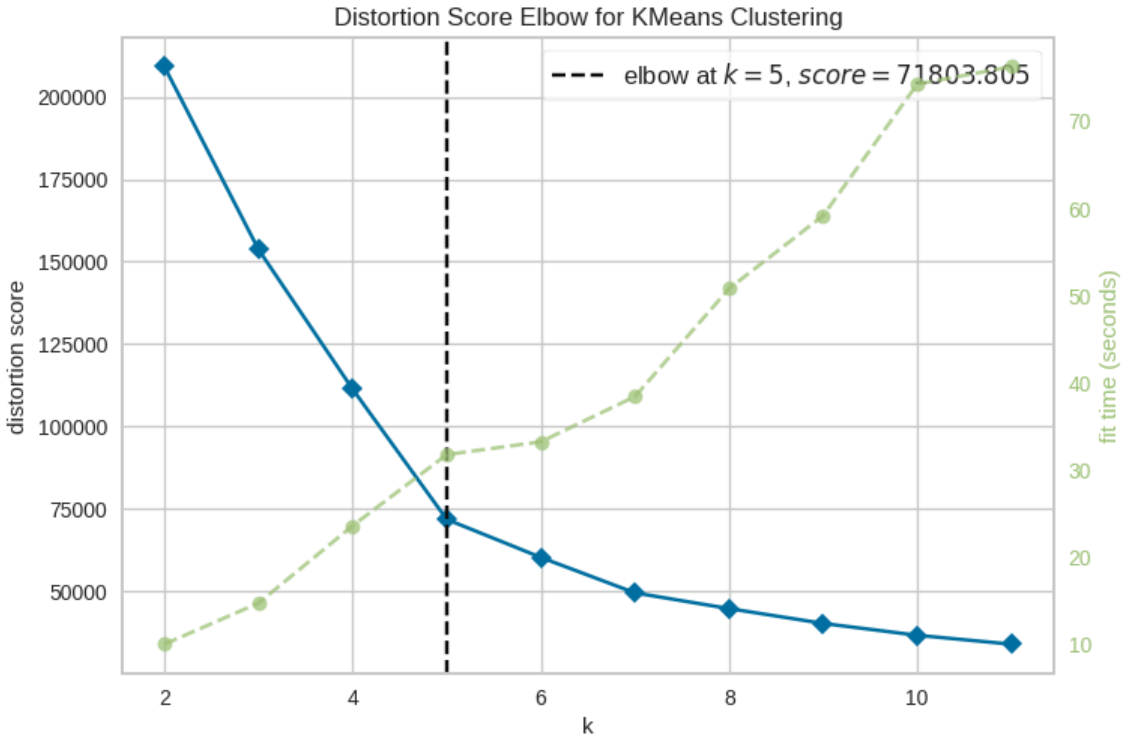
Why:

We chose K-means because it clusters similar data together. Namely, we wanted to cluster similarly ranked matches with the result of the game.

Process:

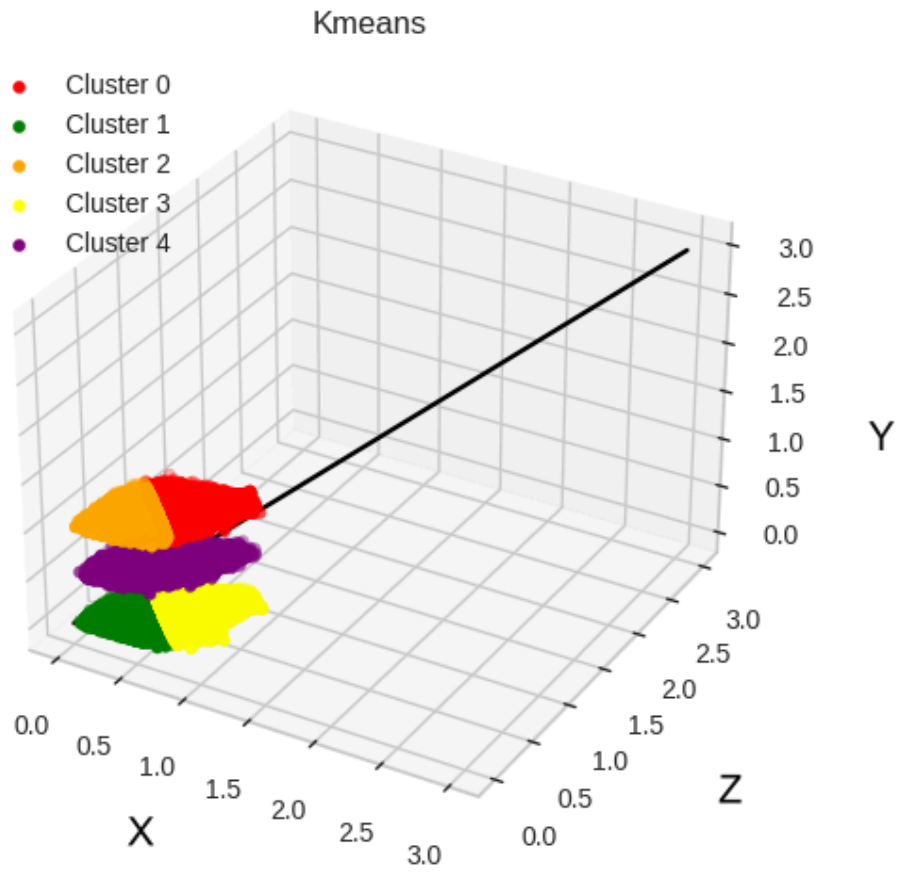
The first step in the K-means Clustering was to normalize the data, specifically the blackelo, whiteelo, and result columns. A blackelo/whiteelo of 1 would represent the highest-ranking players, while a blackelo/whiteelo of 0 would represent the lowest-ranked players, the median falling at 0.5. As for the results, A win for white would be represented as 1, a stalemate as 0.5, and a win for black as a 0.

The next step would be to perform the elbow method to select the best K-value (number of clusters)



From this graph, we can see the “elbow” is at k=5, so we continue with 5 clusters.

Result:



In this graph, we can see 3 ‘layers’ on the Y-axis at 0, 0.5, and 1. These represent the game result, 1 being a white win, 0.5 being a stalemate, and 0 being a black win.

The X-axis represents blackelo, with 0 being the lowest ranked, and 1 being the highest ranked players. The Z-axis represents whiteelo, with 0 being the lowest ranked, and 1 being the highest ranked players.

With this information, we can draw some conclusions:

Cluster 0 represents white wins, where both players are above average skill level. The way that the shape closes off as it approaches (1, 1, 1) shows that fewer games are being played at the highest skill level compared to the average skill level (0.5, 1, 0.5).

Cluster 1 represents a black win where both players are at a lower skill level, much like cluster 0, the shape closes off towards (0,0,0), meaning fewer games are played at a lower level than at the average level.

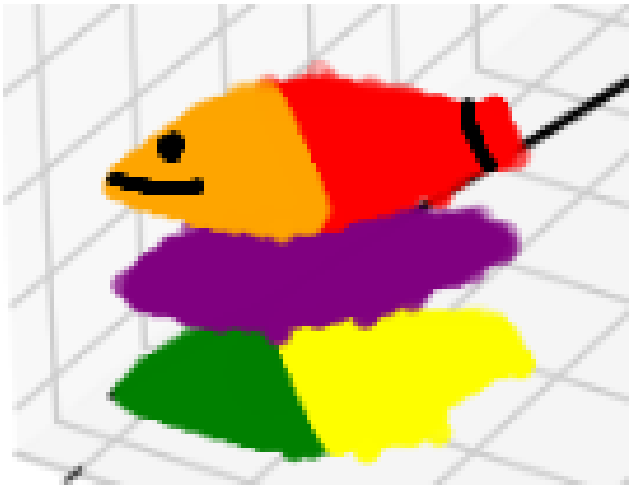
Cluster 2 represents a win for white at a lower skill level. Similar to other clusters, the shape is smaller towards the extreme and larger in the center.

Cluster 3 represents a win for black in higher skill levels. Similar to other clusters, the shape is smaller towards the extreme and larger in the center.

Cluster 4 represents a tie between the two players. Similar to other clusters, the shape is smaller towards the extreme and larger in the center. However, this cluster is slightly larger towards the higher skill levels, meaning that ties happen more frequently at higher levels.

Combining clusters 0&2 and 1&3, they have a sort of “fish" shape (see below). Where the mouth of the fish is toward (0,1,0) and (0,0,0), and the tail points toward (1,1,1) and (1,0,1). Since the tail is wider than the mouth of the fish, it can be inferred that players at higher skill levels often have to play a wider variety of skill levels than people at lower skill levels. This is

likely due to there being so few high-skilled players, and they play a lot more often than low-skilled players to achieve a high ranking. Since they play so often, there may not always be someone available at their skill level when they are looking for a game.



Technique:

Decision Tree (Supervised)

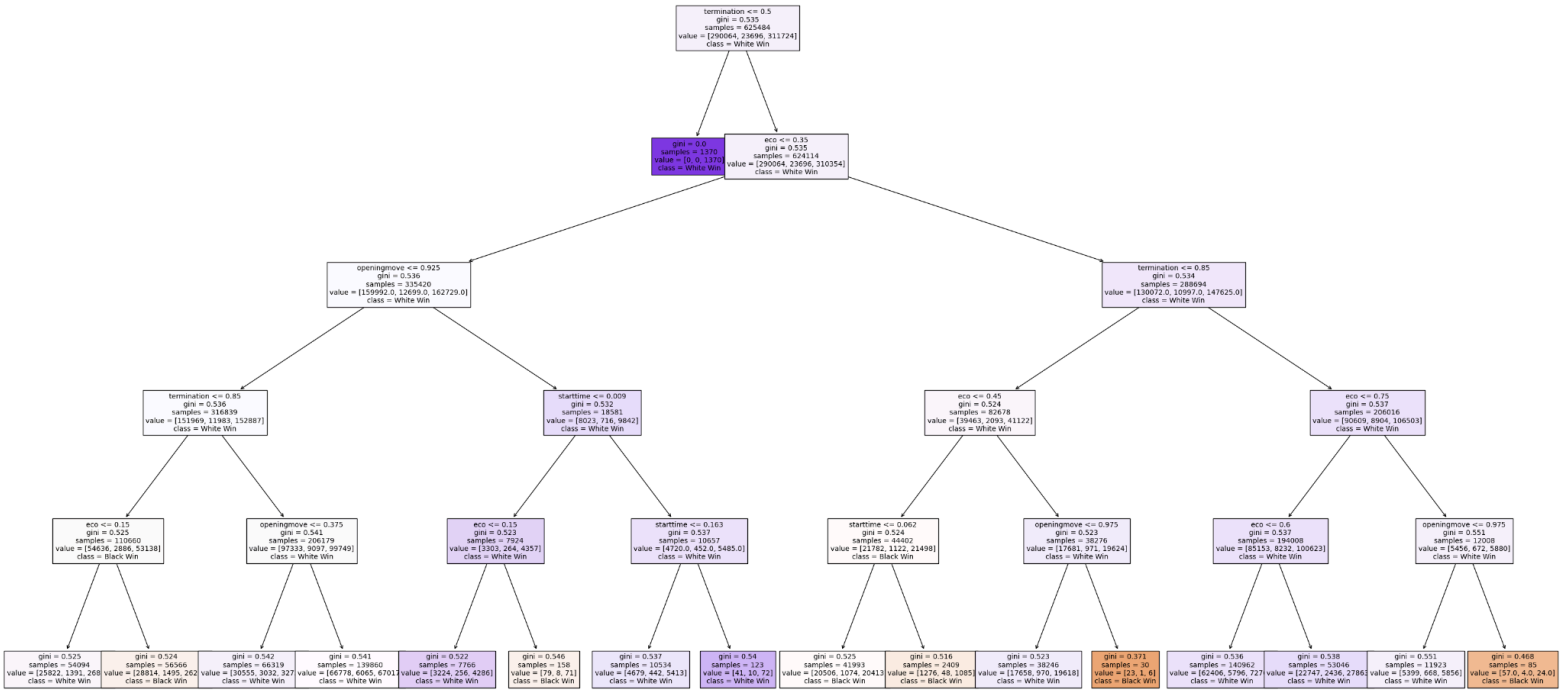
Why:

We picked this technique because of its strong ability to be easily visualized, the ability to capture nonlinear relationships between features, and its ability to ignore irrelevant features. We were interested to see if the opening move, along with the other features available in the data set could make some sort of accurate prediction of the outcome of the game.

Process:

To create the decision tree we first wanted to find out which variables we wanted to test against. Initially including elo columns, resulted in a decision tree based solely on the players' elo and disregarded other aspects of the game. These were removed and the variables we used are 'eco', 'event', 'termination', 'openingmove', 'starttime', and 'result' as the target variable.

We then split the data into training and test sets which were then used to train the decision tree classifier. The target variable ‘result’ had three available outcomes: ‘Black Win’, ‘Stalemate’, or ‘White Win’. The decision tree is listed below and also attached separately for viewing purposes.



Result:

The strongest result is the first layer. This layer sections off games that were terminated unusually. It classifies termination values less than or equal to 0.5 as a win for the white player entirely accurately, which is expected behavior. This includes terminations of black abandoning (not making a first move in time), or a rules infraction, which always awards a win to white. The

next split is based on the first move in eco encoding. Where all openings with the A-B classifications go to one side, and C-E to the other. This, along with most layers below it, were not very strong indicators. The Gini score for each branch sits very close to 0.5 indicating that the split at each point is near even and not very good at helping towards the confidence of the eventual classification. The strongest path of the tree is normal termination or time forfeit, an opening with eco encoding C-E, only time forfeits, and an opening move of Nf3 (white knight to f3), which resulted in Black Wins most of the time. It is unclear why this combination would result in a higher-than-average number of black victories, but with a Gini score of 0.371, out of any possible ending aside from the trivial first layer, this combination led to a more decisive conclusion than any other path. Better results may emerge from an increased depth. Additionally, no draws were classified. This may also be improved with an increased depth, but with something that can only be possible through a very long game, it makes sense that no strong indications can be made from just the first move of the move-set. Start time was rarely used to help classify results, which isn't surprising, but we wanted to see if there was some unusual pattern that could emerge.

Phase 3 Conclusion

In this phase, we utilized a neural network, K-means, and a decision tree to visualize and explain how the result of a given chess match is determined based on several features. The neural network performed with an accuracy of 63%, which could be explained by the results of chess games being more complex to calculate than either our data or model can account for. We used

K-means to cluster each game based on black and white ELO compared to the result of the game. This visualization allowed us to learn more about the results of high versus low-skill players. For example, we learned that ties tend to happen more frequently at higher skill levels. Finally, a decision tree was created to visualize how the result of the game changes based on a subset of features. All three of these methods provided insight into our hypothesis. We predicted that openings would vary more with an increase in elo and that black's win rate would decrease as elo decreased. The data mining showed that the opening move was an important factor in the result of a game, falling behind black and white elo in terms of sensitivity from the neural network. However, our decision tree suggests that the openings don't have as large an impact on the result, but this could emerge as the depth of the tree increases. The clusters reveal that a game with a result of black win tends to have more games with higher ELO winning as black than with lower ELO. This supports our hypothesis that black's win rate would be lower in lower skill levels but doesn't tell us much about how the openings affect these win rates.

Phase 4 Begins

Outline initial proposal of the expected outcome

Our expected outcome of this project revolved around our hypothesis, that openings will be more varied in higher ELO. Experienced players know more openings and Black's win rate will be lower at lower ELO than it is at higher ELO because lower ELO players struggle to play defensively. We expected to find data supporting our claims, especially surrounding the openings and how the result is determined from a given opening. Our hypothesis was based on the fact that while playing as Black in chess, you are forced to play in response to how your opponent plays. Therefore it followed that higher ELO players, who typically win more to maintain their high

ELO, were more likely to have a higher winrate on Black due to this ability to adapt, as well as utilizing a wider range of openings as they studied openings and responses as their ELO increased.

Describe the observed outcome

Many of our methods of data mining and analysis proved useful in not only providing evidence to support our hypothesis but also helping us further understand our dataset. One observation that came out of this process was an increased understanding of our problem domain. This experiment has supported the concept that chess games are complex matches dictated by several factors beyond just player ELO and opening moves. How a player conducts themselves throughout a match is more likely to affect the outcome of the game. One player could be rated much higher than the other and, therefore more likely to win. However, if the higher ELO player constantly leaves pieces undefended due to either a poor game or underestimating their opponent, this would be an example of a "favored" player losing potentially convincingly. The complexity of a chess game was shown when our neural network output struggled to reach 65%. The model wasn't able to consistently predict the result of a game because there is more that goes into the result of a chess game than the ELO, opening move, or the game rules. Despite the complex nature of these games, we were still able to uncover useful patterns from basic analysis of the dataset. For example, in our graphing of win rate and ELO for both Black and White, we saw that Black overall had a lower win rate than White, but players utilizing Black in high ELO had a higher win rate than in lower ELOs. This partially supports our hypothesis, although doesn't properly cite the openings as being the reason for the difference in win rate. When examining the diversity of openings in different binned ELO, we see that the highest diversity of openings occurred in the middle of the ELO distribution, with the least

diversity occurring at the highest level. This directly goes against what we thought in our hypothesis, however still makes sense as once high-level players find a subset of openings that are considered to be the “best”, there isn’t much need to switch it up outside of that. Overall, while our hypothesis may not have been 100% valid, the experiments and analysis performed on the data provided shed light on how we can improve our hypothesis to better match the expectations of the problem domain.

Explain possible reasons for differences and details of results

A reason for differences in the results may be due to a skew in the rating difference for our dataset. This is referencing the portion of our project where we identified outliers in our data. More specifically the rating difference variable, which represented the difference in elo between both players. Our outlier identification revealed that there was a right skew in that column meaning that many games were unfair with an extremely high Elo player versus a low Elo player. These games were still accounted for in our analysis, so it is most likely the reason for any differences in our results.

Explain process/techniques used to analyze data as well as how good the techniques were

For a large portion of our exploratory data analysis, we utilized binning to get a better understanding of how different groups of ELO compared along various metrics (win rate, opening move, etc.) This meant that we could easily visualize these comparisons without worrying about each individual ELO point. We also utilized K-means, a decision tree and a neural network to shed light on our data in our data mining phase. As mentioned before, the neural network showed that our data might not be as cut and dry as we thought, as more factors

go into predicting the result of a chess game than we had access to in our dataset. K-means allowed us the opportunity to see clusters of players and explain their behavior based on ELO, color played as well as the range of ELO a specific group might play against. This method of analysis provided more insight into our dataset and the general trends of the players rather than predicting the result of the games. We also utilized a decision tree to visualize which factors were most important when considering the outcome of a chess game. This proved to be less than successful, with the majority of outcomes resulting in a white win. This could be due to limitations in how long or for how many layers were run on the tree, which eventually became an issue of time when it came to executing the code. This also provides further evidence that our data wasn't quite prepared for highly accurate results prediction.

Describe overall project progress (struggles, areas of success, learning, etc.)

For the most part, our project progressed smoothly throughout the semester with minimal issues. To start, our group communicated heavily with one another, and everyone put in a strong effort for each phase of the project. We primarily spoke with one another via Discord, and everyone in this group was responsive and reliable. This contributed to a hard-working and dedicated environment. Initially, all of us were not dead set on what type of data set to use, but we all collectively loved the idea of using chess, as many of us are familiar with the game, and even play it recreationally. We were organized and always took the initiative to make sure that we completed every task at hand in a non-stressful manner. As for issues, one problem that came up was that our dataset either did not have a lot of useful columns, or the column itself was too complicated to use. To overcome this, we created new variables by performing operations on others, binning larger categories together, or splitting strings to represent two different columns.

This way we had more specific information to work with, and it was something that could be encoded when it was required in any Python scripts. This project allowed us to get hands-on experience with data mining, by being able to modify and interpret our own database. The skills we used such as univariate analysis, multivariate analysis, neural networks, decision trees, and clustering algorithms, are all techniques that can be applied to all other forms of data given that we preprocess them respectively. Each of these techniques allowed us to look at the data from a new angle and gave us a different perspective on how our data was interpreted. These were all valuable and provided us with more useful tools for performing any data mining in the future.