Project Proposal

- Basic Info

Title: Winners and Chess
Team Members: Divya Prabhakaran, Joshua Berger, Tatham Dees
Link to Project Repo: https://github.com/TathamDees/Data-Vis-Group-Project

- Background and Motivation:

As we first started thinking about our project, we came up with several qualifications for selecting a data set. First of all, the data set shouldn't be too small, as that could result in either an extremely large variance in the data, which would make it difficult to create a meaningful visualization, or otherwise in such a small and uniform set of data that no useful conclusions could be drawn. Neither could the dataset be too large, as this would make it unwieldy to analyze in a reasonable amount of time. In regards to the content, we wanted data that was relatively straightforward, both to make it simpler for us to comprehend when designing our visualization, and to allow those who view our project to understand without needing any specialized knowledge. Finally, we wanted a set of data that we all agreed was interesting, and that we had an actual desire to learn more about.

With these specifications in mind, we began searching the Internet for any suitable datasets. We were fortunate to swiftly come across the dataset we ended up choosing, a set of chess data recorded via the Lichess API, while browsing kaggle.com. We quickly decided that this dataset would be a good fit, as it was easy to understand (most people are at the very least aware the chess exists), and it had a large enough variety of statistics that we weren't constrained in terms of what visualizations we could create. Plus, it allowed us to attempt to answer an obvious, but still valuable question: what choices can players make to maximize their odds of winning a game of chess?

- Objectives:

Our project seeks to find measurements about players and play styles in order to find statistics about winning strategies and player matchups. We will also seek to find correlations between games and aspects of those games like the match length to draw conclusions about how these aspects affect player victory chances.

- Data:
    - Collection

We will be using a Kaggle dataset of approximately 20,000 chess games in a csv format. This data has been collected from users on Lichess, an open-source Internet chess server. Lichess is run off of a Python-sourced API that allows for a large-scale of users and teams to play chess through the server. Moreover, Lichess allows for the collection of any given users' game history. For each game, the Kaggle dataset contains the following data:

- Game ID; Rated (T/F);Start Time;End Time;Number of Turns;Game Status;Winner;Time Increment;White Player ID;White Player Rating;Black Player ID;Black Player Rating;All Moves in Standard Chess Notation;Opening Eco (Standardised Code for any given opening);Opening Name;Opening Ply (Number of moves in the opening phase)
    - Link to Dataset

https://www.kaggle.com/datasnaek/chess

- Data Processing:
    - Data Cleaning [eliminating possible problem outliers]

We have identified possible sources of problem outliers. As discussed earlier, this Kaggle dataset has sourced data from over 20,000 chess games on Lichess. While the Lichess API has access to every aspect of the game, this dataset focuses on specific aspects such moves made by each player with an emphasis on standardized code moves for each player and an emphasis on opening moves. Within these fields, we see possible issues arising such as moves that are under or overutilized that may act as problem outliers, in both the normal course of the game as well as during the opening phase. Without removing these outliers, it's possible the data may be skewed when calculating statistics on the role of particular moves and winning. Another issue that may arise is unnecessary complexity in the visualization with under or overutilized moves.

- Data Aggregation:

To get the opening moves of each player, we will use the first values of 'moves' separated by a space. At first, it seemed more optimal to take the values from 'opening_name', but these do not tell the opening for both players, so it is better to take the opening directly from the player moves list. The 'winner' field contains values for more than just black and white victory. For draw situations, the value will be added to buckets as either half black and half white victory, or not added at all. Adding a black and white victory would throw off the calculations by making one game count for more than a games worth in the statistics.

- <u>Must-Have Features</u>:

Our visualization must be able to produce information about the chances of winning based off of each player's move to help form predictions about winners based off of openings, and comparisons based off of ELO to determine how important the ELO gap is when determining chances of victor.

- <u>Optional Features</u>:

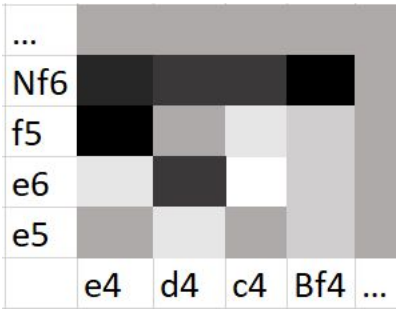
Optionally, our visualization would be able to pick out data for player specific matchups to help guide playstyle choices against specific opponents, but this is not critical. Statistics we could pull are frequencies of player opening usage to develop strategies against. Also, we could do comparisons between the length of a game, the winner, and the ELO to see how important the opening and endgame is for players of different skill levels.

- <u>Project Schedule</u>:

Week of November 4th: Data aggregation and analysis, code basic data visualizations such as bar charts, prepare for project update, work on process book
Week of November 18th: Complete data collection/cleaning, code data structures and have framework set for more complex data visualizations, continue working on process book - prepare for prototype
Week of November 25th: Complete coding of all visualizations, set up HTML
Week of December 2nd: Present

- Visualization Design:
    - **Opening move Heatmap:**

Our first visualization can be a heatmap of player openings. On the x axis will be all of the possible opening moves that the white side player can do. The y axis will be all of the possible opening moves that the black side player can do.The boxes produced will be shaded on a scale from black to white, with gray in the middle. The scale corresponds to the percentage of wins that each player achieves from their openings. For example, if the white player wins a majority of a specific opening combination, then the box will be lighter. Alternatively, a more even match would result in a neutral gray box and a black player advantage box would be darker. The data to be used will be the first few moves from the moves field to find out how each player chooses to start, and the winner of the side will go into a bucket for the opening pair to calculate percentage of wins.



Example conclusions:

Nf6 is a strong all around move for black player

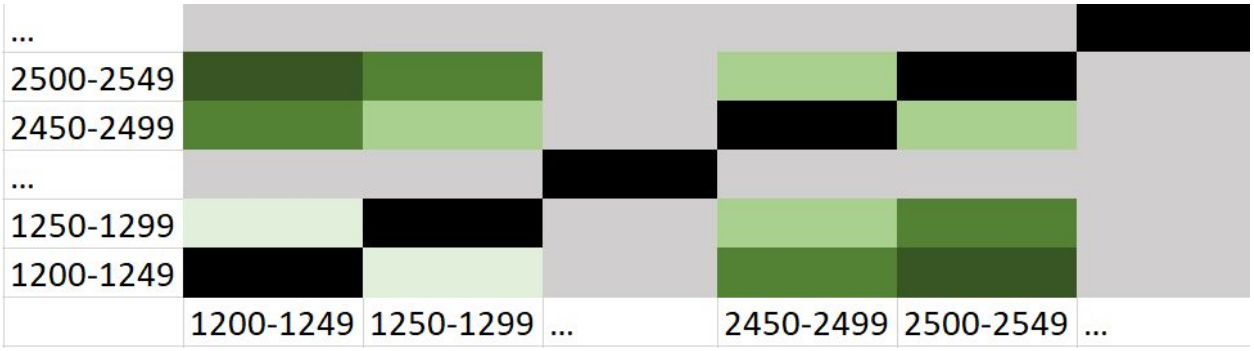
Bf4 good white player move, countered heavily by Nf6

c4 and e6 heavily white favored. If black player sees c4, don't choose e6

From this visualization, we will be able to determine any correlations between opening choice and victory rate. Players could use this to formulate plans to counter specific openings, and if there is little correlation, we will know that the opening move is not a very important factor in determining the outcome of a chess game. Optionally, the ability to select a specific move combination and develop a new heatmap for each next move would allow for percentages to be shown for each additional move, but the size of this dataset might not permit scaling for a level to determine patterns of moves. In addition, a move translator would help less experienced players decipher openings.

- **ELO Heatmap win rate:**

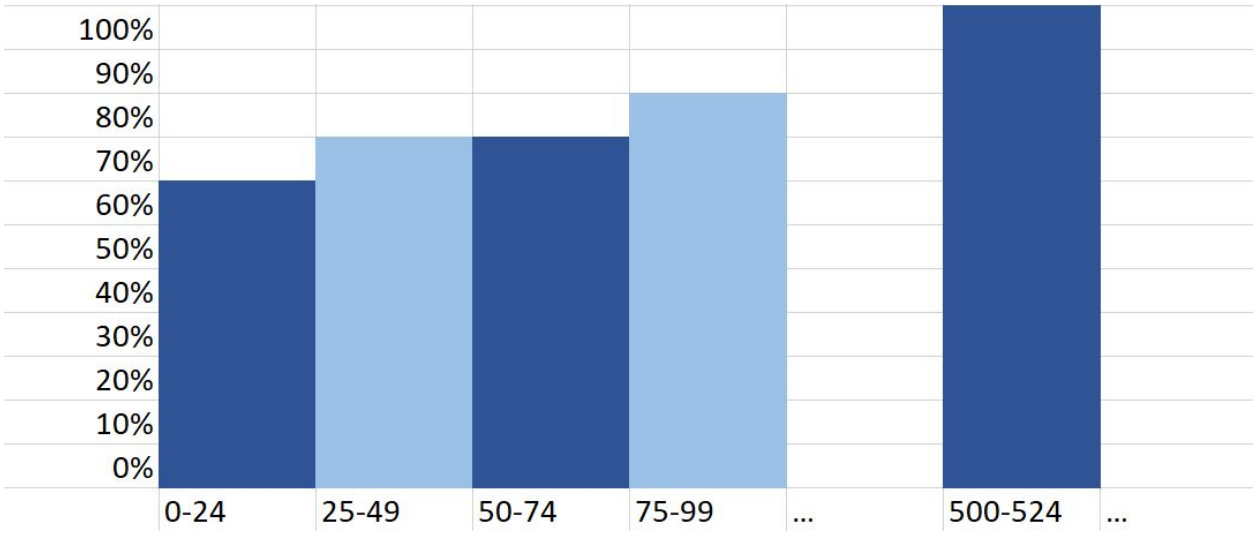
Another part of our visualization would need to be able to determine how strong ELO is at predicting win rates. To do this, we could build another heat map with ELO segmented in increments of 50 for both players. The cells could be shaded in green to represent the confidence that the higher player would win (darker = more certain).



This visualization would be able to tell how specific ELO players do against others, but would also have repetition of ELO matchups. As an example, in a box representing a 1200-1249 level player versus a 2450-2500 level player, there might be a dark green box depicting that it is almost 100% certain the higher ELO player would win. This visualization might be limited by the dataset if there are too many differences in ELO that some ELO ranges might not play each other.

- **Scaleband Elo differences:**

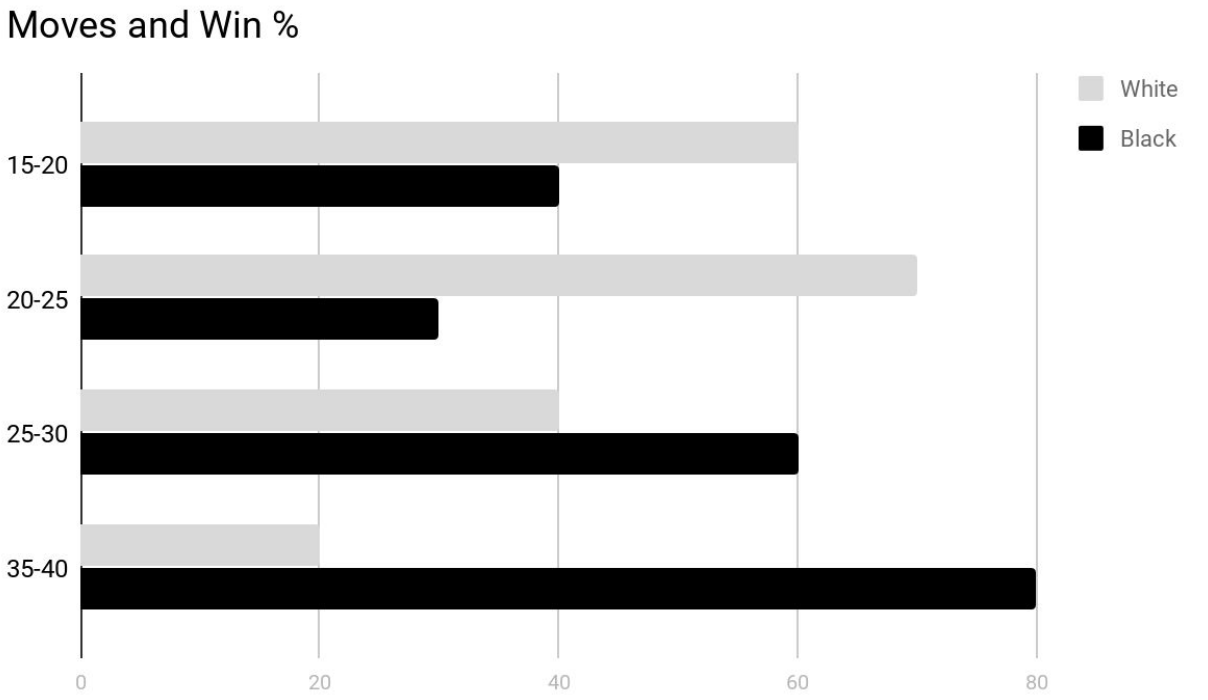
Another possible way to represent how ELO corresponds to win rates is to create an aligned bar chart with buckets for the differences in ELO compared to win/loss. The height of the bar is the % chance that the higher ELO player wins. By taking the highest and lowest ELO difference in the dataset, we can then decide how many sections to create for bars.



This method would not be able to separate by the specific ELO of each player, but would be much easier in depicting the general trends in confidence of ELO differences. The visual channel of position in height of bars is a lot better for understanding the comparisons between ELO differences and victory percentages. This can visually depict how hard it is to beat a player of higher skill rating, and can show the probability of an upset, giving context to how skillful a top level player is compared to lower level players.

- **Winning side by number of moves [20-25 moves, white player bar 60%]**

The next part of our visualization would focus on data from the winning outcomes. We could analyze data by looking at the data values for winner ID (whether white or black) and number of moves. Since there are thousands of different players in this dataset, our method would involve allowing the user to choose a range of moves to look at, through a selection method. Through using bar visualizations, we can look at we can look at the percentage of wins per side based on the range of moves. The primary data in this visualization would include the winner ID as well as the total moves. To calculate the visualization, we would select the winner ID and calculate the percentage of wins to create a bar graph.

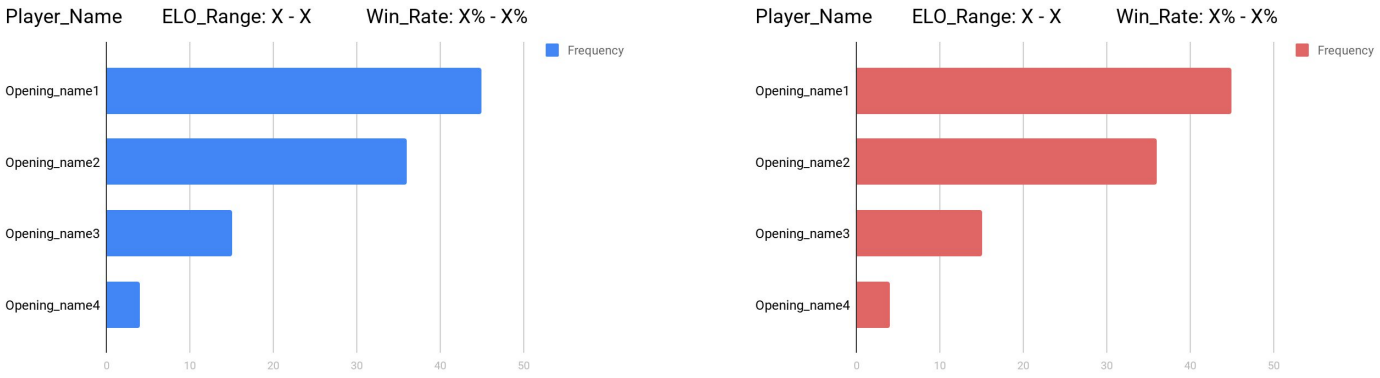


This visualization allows users to determine an optimal number of moves along with probabilities of winning.

- **Frequency of openings by player bar chart:**

A final option for our visualization would be to analyze the data on a player by player basis, looking at each player's preferences with regards to openings, and also including their ELO score and win rate to allow for more generalizable conclusions. Due to the thousands of different players in this dataset, it would be impractical to display the results for every player at once. Instead, we would have some method of choosing two players to compare (via a drop down or otherwise). Alternatively, instead of viewing stats for each individual player, we could allow the user to choose some range of players whose stats they wish to view in aggregate.

Two options for determining these ranges are win-rate or ELO ranking. The latter would integrate especially well with the ELO/win-rate heatmap, as the user would be able to select an individual square, and then view more extensive data for each ELO range that determines that square's coordinates (i.e if selecting the 1200-1249 vs 2450-2500 square, the user would see one chart for players with an ELO of 1200-1249, and a second for players with a ranking of 2450-2500).



The primary data displayed in this visualization would be the frequency at which each opening was used. For now, we expect to visualize this as two bar charts side by side for each of the two players (or player aggregates) the user wishes to compare. Other options would be to have both sets of data on the same chart (making it easier to compare), or displaying the data in another form, such as a pie chart or a treemap (a treemap in particular would allow us to account for very similar, yet technically different openings like `Alekhine Defense` and `Alekhine Defense #2`). This visualization would make it easy to see to what extent opening choice contributes to win-rate and ELO (if at all) and vice versa (for example, it would allow us to answer the question: do "better" players tend to use different openings than "worse"