Analysis on Chess Openings and their Optimal Counters

Arden Chaing

Summary of Research Questions:

1) *What is the optimal first move for white?*

Chess is a turn-based strategy game consisting of an 8x8 square board and 32 pieces; 16 white and 16 black. The objective for this two-player game is to "checkmate" your opponent's king while preventing your own from being "checkmated." Since there are six specific types of pieces, each with their own unique movements, after just two moves there are 400 possible board setups (1). And after 10 moves, each player moving five times, there are 69,352,859,712,417 possible games that could have been played (2). However, there are only 20 potential opening moves for white. Thus, by analyzing the first-moved-played and its winner, this program can produce statistically best opening move.

Answer: White moves their knight on g1 to f3 otherwise known as Nf3.

2) *What is more useful the bishop or the knight?*

In chess, each piece is given a respective point value; pawn one, queen 9, king infinity, etc. However, the bishop and knight both have the same point value. It has been a debate over which one is more valuable with the world champion Bobby Fisher stating that a bishop should be worth 3.25. However, this program will analyze thousands of chess games and will see which one was utilized more the bishop or the knight.

Answer: Overall the bishop has more meaningful use than the knight.

3) *What is the optimal opening to play depending on your opponent's Elo rating and starting color?*

Despite previously stating that there are billions of possible games after the first couple of moves in chess, most of them are not likely to occur because of the countless number of resources on chess theory; specifically, chess openings. Most players will play standard openings since they have been tried and tested. But users will most likely not know their opponent's favorite chess opening. However, a user's ELO typically showcases how talented they are at the game and one can infer their quality of play. Therefore, this program will analyze thousands of chess games and their respective openings to learn what is the optimal opening or response for any skill level of their opponent.

Answer: Able to predict the optimal opening line with at least 11% accuracy.

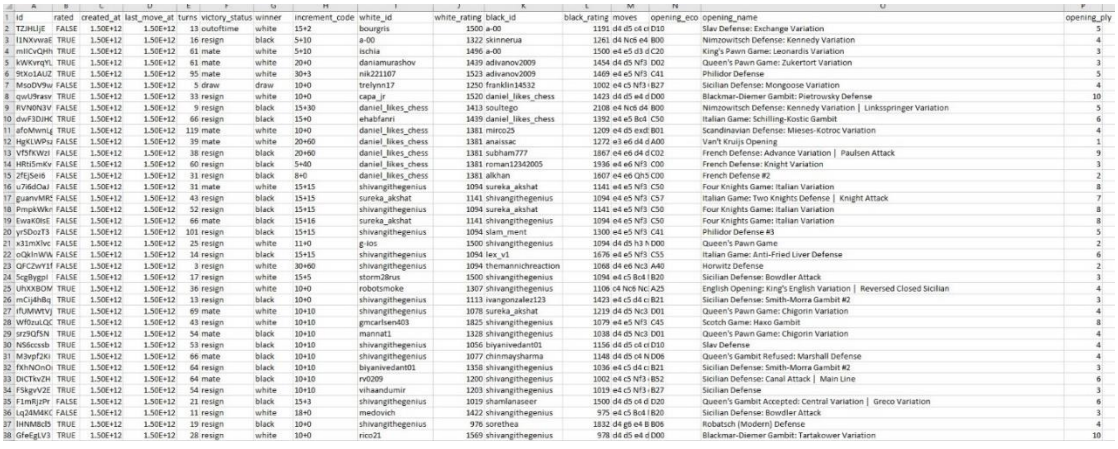
Motivation and Background:

I learned how to play chess in 3rd grade from my mother. Because she was not an adamant player, I soon began to crave higher competition and joined my elementary school's chess club. After receiving local lessons and competing at several regional competitions my chess career reached its peak when I challenged my social studies teacher in 7th grade. Known for his strong passion at chess and prestigious academic studies at Yale, he systematically countered my flashy moves with his calculative openings, most dangerous of all his Queen's Gambit. After two scornful defeats, I began to realize that despite my aggressive flare in the middle game and tactical moves in the end game, I would never defeat my teacher unless my early game strengthened dramatically.

Like my 7th grade self, many others in the chess community - especially beginners - can be intimidated by the sheer magnitude of openings. Lacking preparation compared to more seasoned players; new players typically rely on mechanical fundamentals rather than theory. Even though mastering fundamentals is key to enjoying the sport of chess, no one likes losing. Games can be easily decided by the strength of one's opening playstyle as it heavily influences the flow and momentum of the middle and eventually end game. In hopes to retain more bright-eyed beginners, this analysis aims to provide users with information on optimal openings and piece usefulness. By providing a data driven guide on chess openings, chess players can continue to sharpen their fundamentals and ease their way into learning many different chess openings while maintaining a positive spirit about the sport of chess. And in time they too can defeat their own social studies teacher.

Dataset:

The main dataset comes from 20,000 chess games played on lichess.org. This dataset was uploaded three years ago and contains information about individual chess matches. Columns: 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, opening name, and opening ply (number of moves in the opening phase). Link to the dataset can be found here.



	id	rated	created_at	last_move_at	turns	victory_status	winner	increment_code	white_id	white_rating	black_id	black_rating	moves	opening_eco	opening_name	opening_ply
2	TZHtLJjE	FALSE	1.50E+12	1.50E+12	13	outoftime	white	15+2	bourgris	1500	a-00	1191	d4 d5 c4 e	D10	Slav Defense: Exchange Variation	5
3	l1NXvwaE	TRUE	1.50E+12	1.50E+12	16	resign	black	5+10	a-00	1322	skinnerua	1261	d4 Nc6 e4	B00	Nimzowitsch Defense: Kennedy Variation	4
4	mIICvQHh	TRUE	1.50E+12	1.50E+12	61	mate	white	5+10	ischia	1496	a-00	1500	e4 e5 d3 d	C20	King's Pawn Game: Leonardis Variation	3
5	kWKvrqYL	TRUE	1.50E+12	1.50E+12	61	mate	white	20+0	daniamurashov	1439	adivanov2009	1454	d4 d5 Nf3	D02	Queen's Pawn Game: Zukertort Variation	3
6	9tXo1AUZ	TRUE	1.50E+12	1.50E+12	95	mate	white	30+3	nik221107	1523	adivanov2009	1469	e4 e5 Nf3	C41	Philidor Defense	5
7	MsoDV9w	FALSE	1.50E+12	1.50E+12	5	draw	draw	10+0	trelynn17	1250	franklin14532	1002	e4 c5 Nf3	B27	Sicilian Defense: Mongoose Variation	4
8	qwUr9rasv	TRUE	1.50E+12	1.50E+12	33	resign	white	10+0	capa_jr	1520	daniel_likes_chess	1423	d4 d5 e4 d	D00	Blackmar-Diemer Gambit: Pietrowsky Defense	10
9	RVN0N3V	FALSE	1.50E+12	1.50E+12	9	resign	black	15+30	daniel_likes_chess	1413	soultego	2108	e4 Nc6 d4	B00	Nimzowitsch Defense: Kennedy Variation | Linksspringer Variation	5
10	dwF3DJHC	TRUE	1.50E+12	1.50E+12	66	resign	black	15+0	ehabfanri	1439	daniel_likes_chess	1392	e4 e5 Bc4	C50	Italian Game: Schilling-Kostic Gambit	6
11	afoNwmLq	TRUE	1.50E+12	1.50E+12	119	mate	white	10+0	daniel_likes_chess	1381	mirco25	1209	e4 d5 exd5	B01	Scandinavian Defense: Mieses-Kotroc Variation	4
12	HgKLWPcs	FALSE	1.50E+12	1.50E+12	39	mate	white	20+60	daniel_likes_chess	1381	anaissac	1272	e3 e6 d4 d	A00	Van't Kruijs Opening	1
13	VTSfKW2I	FALSE	1.50E+12	1.50E+12	38	resign	black	20+60	daniel_likes_chess	1381	subham777	1867	e4 e6 d4 d	C02	French Defense: Advance Variation | Paulsen Attack	9
14	HRti5mKv	FALSE	1.50E+12	1.50E+12	60	resign	black	5+40	daniel_likes_chess	1381	roman12342005	1936	e4 e6 Nf3	C00	French Defense: Knight Variation	3
15	zEjSei6	FALSE	1.50E+12	1.50E+12	31	resign	black	8+0	daniel_likes_chess	1381	alkhan	1607	e4 e6 Qh5	C00	French Defense #2	2
16	u7i6dOaJ	FALSE	1.50E+12	1.50E+12	31	mate	white	15+15	shivangithegenius	1094	sureka_akshat	1141	e4 e5 Nf3	C50	Four Knights Game: Italian Variation	8
17	guanvMRf	FALSE	1.50E+12	1.50E+12	43	resign	black	15+15	sureka_akshat	1141	shivangithegenius	1094	e4 e5 Nf3	C57	Italian Game: Two Knights Defense | Knight Attack	7
18	PmpkVKn	FALSE	1.50E+12	1.50E+12	52	resign	black	15+15	shivangithegenius	1094	sureka_akshat	1141	e4 e5 Nf3	C50	Four Knights Game: Italian Variation	8
19	EwaK0lsT	FALSE	1.50E+12	1.50E+12	66	mate	black	15+16	sureka_akshat	1141	shivangithegenius	1094	e4 e5 Nf3	C50	Four Knights Game: Italian Variation	8
20	yr5Doz73	FALSE	1.50E+12	1.50E+12	101	resign	black	15+15	shivangithegenius	1094	slam_ment	1300	e4 e5 Nf3	C41	Philidor Defense #3	5
21	x31mXlvc	FALSE	1.50E+12	1.50E+12	25	resign	white	11+0	g-ios	1500	shivangithegenius	1094	d4 d5 h3 h	D00	Queen's Pawn Game	2
22	oQkInWlN	FALSE	1.50E+12	1.50E+12	14	resign	black	15+15	shivangithegenius	1094	lex_v1	1676	e4 e5 Nf3	C55	Italian Game: Anti-Fried Liver Defense	6
23	QFCZwY1f	FALSE	1.50E+12	1.50E+12	3	resign	white	30+90	shivangithegenius	1094	themannichreaction	1068	d4 e6 Nc3	A40	Horwitz Defense	2
24	5cgBygpI	FALSE	1.50E+12	1.50E+12	17	resign	white	15+5	storm28rus	1500	shivangithegenius	1094	e4 c5 Bc4	B20	Sicilian Defense: Bowdler Attack	3
25	UhXXBOMV	TRUE	1.50E+12	1.50E+12	36	resign	white	10+0	robotsmoke	1307	shivangithegenius	1106	c4 Nc6 Nc	A25	English Opening: King's English Variation | Reversed Closed Sicilian	4
26	mCjJhBq	TRUE	1.50E+12	1.50E+12	13	resign	black	10+0	shivangithegenius	1113	ivangonzalez123	1423	e4 c5 d4 c	B21	Sicilian Defense: Smith-Morra Gambit #2	3
27	rfUMWfVj	TRUE	1.50E+12	1.50E+12	69	mate	white	10+10	shivangithegenius	1078	sureka_akshat	1219	d4 d5 Nc3	D01	Queen's Pawn Game: Chigorin Variation	4
28	WfRzuLQC	TRUE	1.50E+12	1.50E+12	43	resign	white	10+10	gmcarlsen403	1825	shivangithegenius	1079	e4 e5 Nf3	C45	Scotch Game: Haxo Gambit	8
29	srz9Qf5N	TRUE	1.50E+12	1.50E+12	54	mate	black	10+10	mannat1	1328	shivangithegenius	1038	d4 d5 Nc3	D01	Queen's Pawn Game: Chigorin Variation	4
30	NS6ccssb	TRUE	1.50E+12	1.50E+12	53	resign	black	10+10	shivangithegenius	1056	biyanivedant01	1156	d4 d5 c4 c	D10	Slav Defense	4
31	M3vpf2KI	TRUE	1.50E+12	1.50E+12	66	mate	black	10+10	shivangithegenius	1077	chinmaysharma	1148	d4 d5 c4 N	D06	Queen's Gambit Refused: Marshall Defense	4
32	fXhNOnOi	TRUE	1.50E+12	1.50E+12	64	resign	black	10+10	biyanivedant01	1358	shivangithegenius	1036	e4 c5 d4 c	B21	Sicilian Defense: Smith-Morra Gambit #2	3
33	DiCTEvZH	TRUE	1.50E+12	1.50E+12	64	mate	black	10+10	rv0209	1200	shivangithegenius	1002	e4 c5 Nf3	B52	Sicilian Defense: Canal Attack | Main Line	6
34	FSkgvV2E	TRUE	1.50E+12	1.50E+12	54	resign	white	10+10	vihaandumir	1203	shivangithegenius	1019	e4 c5 Nf3	B27	Sicilian Defense	3
35	F1mRjzPr	FALSE	1.50E+12	1.50E+12	21	resign	black	15+3	shivangithegenius	1019	shamlanaseer	1500	d4 d5 c4 d	D20	Queen's Gambit Accepted: Central Variation | Greco Variation	6
36	Lq24M4KC	FALSE	1.50E+12	1.50E+12	11	resign	white	18+0	medovich	1422	shivangithegenius	975	e4 c5 Bc4	B20	Sicilian Defense: Bowdler Attack	3
37	IHNM8cB5	TRUE	1.50E+12	1.50E+12	19	resign	black	10+0	shivangithegenius	976	sorethea	1832	d4 g6 e4 B	B06	Robatsch (Modern) Defense	4
38	GfeEgLV3	TRUE	1.50E+12	1.50E+12	28	resign	white	10+0	rico21	1569	shivangithegenius	978	d4 d5 e4 d	D00	Blackmar-Diemer Gambit: Tartakower Variation	10

Methodology:

Step 1:

Read and store the dataset. Clean the data so therefore it will contain rated (T/F), number of turns, game status, winner, white player rating, black player rating, all moves in standard chess notation, and opening name. Also eliminate rows that do not have rated games. This eliminates casual games in which players might not always play the optimal move since there is less at stake. Also eliminate rows where both players are not above a 1200 ELO. Typically, a beginner has an ELO rating of 800, a mid-level player around 1600, and a professional around 2400. To eliminate "sub-optimal" opening moves, we are only considering games played by knowledgeable players (anything below a 1200 is considered a novice).

Step 2:

To answer, what is the best starting first move for white, create a new dataset in which a white player won with a mate or resignation and the number of turns is greater than 18. That value was selected because Gary Kasparov resigned in just 19 moves in Game 6 of the 1997 rematch against Deep Blue. After creating a new data set, then parse through the moves of the entire game and keep track of the first move of every single game and which of those games white wins. Lastly, calculate each opening move's respective winning percentage and the one that is the highest is the best move. Only consider moves where at least 3% of the people play it. This value was chosen because it encompassed the four most played moves.

Step 3:

To answer, which is more useful the bishop or the knight, perform the following for each game (only considering rated games where both players have ELOs of 2200 of higher, meaning they are at least a Candidate Master).

Receiving only the moves for every game, evaluate the starting position score of the chess board from white's POV utilizing stockfish for analysis. Convert stockfish analysis into meaningful systemized score integer. Scores received from stockfish as either Cp (centi-pawns) or Mate. Example of order can be seen as Mate(-0) < Mate(-2) < Cp(-50) < Cp(2000) < Mate(12) < Mate(0). Conversion used for Mate to Cp is 5000 – mate score. Example being Mate(5) = Cp(4995). Discover what move is going to be played and what piece type is being move. If castling, then consider both the rook and king as moving. Update the board with the move and analyze the current position score from white's POV. Convert stockfish analysis into meaningful systemized score integer as described above. Store/update the overall score for the respective piece type moved as the change in positional score integer after that piece was moved. Do above for every single move and then every single game.

Step 4:

To answer, what is the optimal opening to play depending on the specific opponent, clean the dataset so it only contains the winner, white_rating, black_rating, and the opening_name. Since we are solely looking at positional advantages from an opening, I believe it would be best

to eliminate all other columns as they do not provide much importance to the question. Because ratings cover a vast range, change both players ratings to reflect their respective FIDE title such as Class A or Grand Master. Also, since there are many opening variations, change the openings names so they only reflect the main line.

Split the data with 60:20:20 for training, developing, and testing, respectively. Now discover the optimal hyperparameters using the training and developing datasets. I chose to explore max_depth, min_samples_split, and min_samples_leaf. Then create a decision tree classifier with the newly discovered optimal hyperparameters and calculate the test accuracy.

Receive information from the user to discover the optimal opening line for them. This includes their opponent's rating and starting color and their own rating. Construct a hypothetical "game" with that information and predict the opening in which the user wins using the trained model.

Results:

1) *What is the optimal first move for white?*

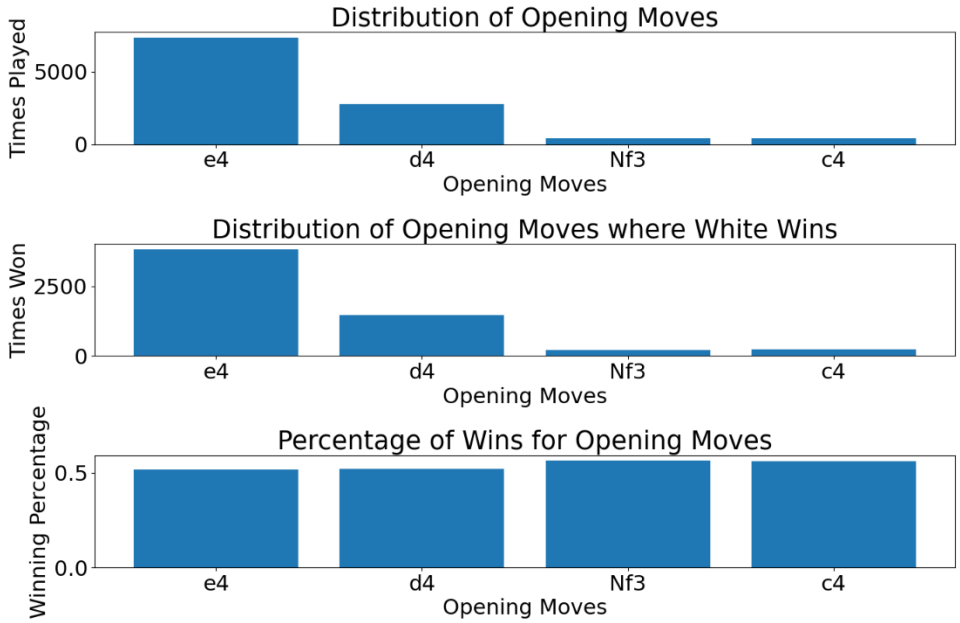
Initially my approach for this was to discover the opening move that had the highest total amount of wins. However, upon analyzing the data clearly, I realized a lot of players will open with e4 (the King pawn). Coined by American Grandmaster Bobby Fisher as the "best by test" opening. Any chess player can see that it is a strong opening move for white as it controls the center with the king pawn and opens lines of movement for both the light square bishop and the queen. Of the valid 11,679 chess games from Lichess, 7,374 opened with e4, 2nd most being d4 (the Queen pawn) with 2,800 plays. After that there is a clear drop between the 2nd most played and the 3rd and 4th most played which were c4 with 416 plays and Nf3 with 391 plays.

However, despite being played the most and winning the greatest number of games as a byproduct, e4 did not have the highest win ratio among the opening moves. Considering only the top four most played openings to ensure a large enough sample size for each opening move (utilizing the law of large numbers), the following showcases the openings in order of highest win ratio to lowest. Nf3 with 56.53%, c4 with 56.25%, d4 with 52.18%, and e4 with 52%. Even though the top four opening move sets does not surprise me, I was a bit astonished to see that Nf3 and c4 had a higher winning percentage than e4.

We do need to keep into consideration how many times Nf3 and c4 were played, as its many factors lower than d4 and e4. Therefore, those winning percentages for Nf3 and c4 may not be as accurate as statistics for e4 and d4 because it does not have as many "trials." Nevertheless, Nf3 and c4 are valid opening moves as they form the basis of some very popular openings, including the King's Indian Attack or the English Opening (3). A great number of players in the history of the game have favored Nf3 because unlike e4 and d4 it does not commit white to a particular strategy, but can transpose into several other openings, which can throw off a black player who commits too soon to a particular

response (3). In conclusion, any four of these openings give an advantage to white as they have a winning percentage greater than 50%, but Nf3 does seem to produce the best chance of winning.

```
Results from opening move method
Total games analyzed: 11679
First moves times played: {'e4': 7374, 'd4': 2800, 'Nf3': 391, 'c4': 416, 'a4': 8, 'g3': 94, 'e3': 190, 'f4': 79, 'b4': 46,
'Nc3': 43, 'b3': 94, 'h3': 6, 'd3': 53, 'h4': 6, 'c3': 34, 'a3': 18, 'Na3': 1, 'g4': 12, 'Nh3': 3, 'f3': 11}
First moves where White wins: {'e4': 3832, 'd4': 1461, 'Nf3': 221, 'c4': 234, 'e3': 56, 'b4': 27, 'Nc3': 24, 'b3': 49, 'h3':
 4, 'f4': 38, 'd3': 20, 'g3': 37, 'h4': 4, 'c3': 22, 'g4': 4, 'a4': 4, 'a3': 14, 'Nh3': 1, 'f3': 5}
Win ratios: {'e4': 0.5196636832112829, 'd4': 0.5217857142857143, 'Nf3': 0.5652173913043478, 'c4': 0.5625}
Best opening: Nf3
```



2) *What is more useful the bishop or the knight?*

I tried to analyze whether two pieces with differing functions impact the game equally. After computing the positional analysis aggregate for the piece types for every move of the 132 games (number of rated games where both players have ELO ratings of 2200+), we can see below that bishops produced a more advantageous. However, there is some caveats to this conclusion.

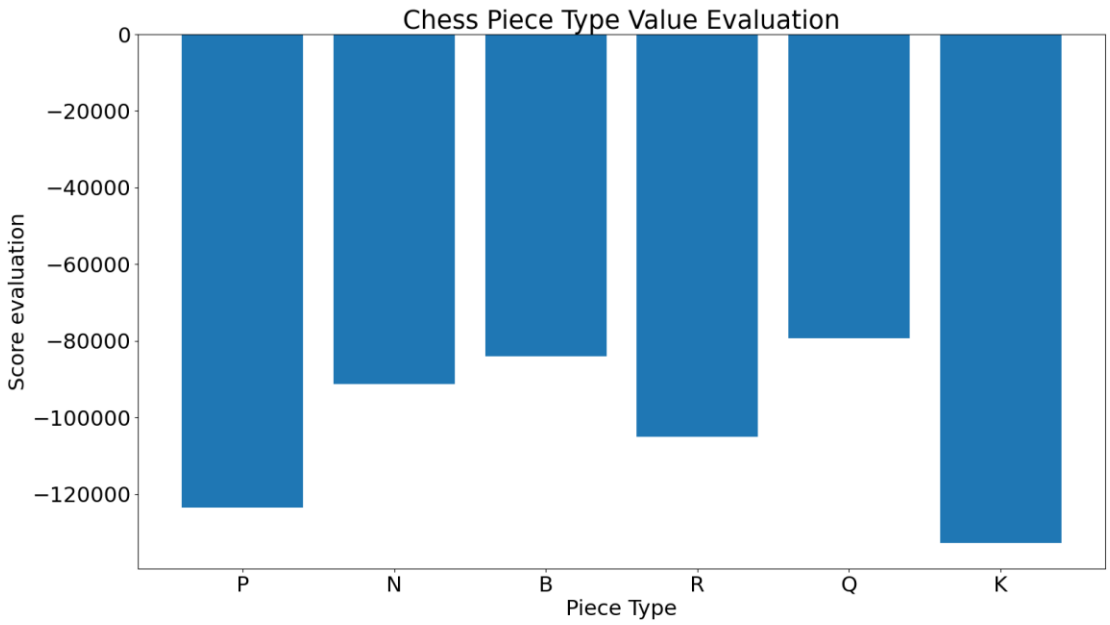
One thing I found particularly interesting was that despite analyzing games played by players of CM level or higher, no piece type had a positive overall score. Meaning all the pieces had more bad moves than good moves. This makes sense as even masters are not perfect and therefore would not always play the optimal move.

Interpreting the evaluations, we can see that the pawn had the second-most negative positional score. One explanation could be that since many see them as the most expendable piece type because they are the weakest on their own, players do not value them as highly as others and as a byproduct produce poor positional scores. The third-worst score was with the rooks which was surprising. It's the second highest piece type with a value of 5 and its usage skyrockets during the endgame where it has long ranks

and files to dominate. Possibly due to the sheer number of possible moves for rooks, especially during the endgame, players are less likely to play the best move resulting in such a low overall positional score aggregate. The king has the worst score. I was expecting this to have the worst or second worst score because the game hinges on the protection of one's own king. And when a king must move it is usually because it is in check. Which in turn, means its move options are already limited from the check and it is in danger, therefore any move afterwards is likely not a very favorable one because it is already in danger. And of course, the Queen has the best score evaluation of all the piece types. This can be explained due to its flexibility and strength in a game of chess. I am intrigued that it is not 9x more useful than a pawn or 3x more useful than a bishop/knight (going off the standard piece values). Possibly these piece values are based of idealized concepts of the pieces where they are utilized to their maximum, something humans cannot replicate on a chess board due to our imperfections.

Lastly, we can see that bishops have a slightly higher aggregate positional score than knights. These scores of course are highly dependent on positioning, which is especially true for bishops and knights. In completely open positions without pawns, the bishop if superior to the knight. Conversely, the knight is superior to the bishop in closed positions, on the one had because the pawns are in the bishop's' way, and on the other hand because the pawns form points of support for the knight (4). As the comparative strength of each piece type depends entirely on the circumstance of the board, it is difficult to explicitly which piece type will be more important in a single game. However, after analyzing 132 master-level chess games, bishops were utilized better than knights to achieve a more advantageous position.

```
Results from piece_value method
Piece evaluation: {'P': -123445, 'N': -91217, 'B': -84082, 'R': -105096, 'Q': -79409, 'K': -132830}
```



3) *What is the optimal opening to play depending on your opponent's Elo rating and starting color?*

Most of this time split between training/testing the machine learning model and creating a replicate "game" to predict the optimal opening line. And after testing three different hyperparameters (min_depth, min_samples_split, and min_samples_leaf), I discovered their optimal values were 5, 2, and 14 respectively since it yielded the highest test accuracy.

Initially shocked that the testing accuracy was always around 12%, I was concerned my model was not trained well enough or the hyperparameters were not optimal. However, I specifically tested all the hyperparameters within a scientifically backed range and realized that my model will almost never reach an ideal test accuracy since there are so many more intangibles to the game of chess than just the opening and ELO ratings, such as playstyle and time per move to name a few (5). Nevertheless, my algorithm still provides a suitable option for an opening line if user is stumped, therefore still proving useful. An example can be shown below where it suggested the Sicilian Defense if the user was playing as black, a widely used counter to white's e4.

```
Results from opening_model method
Parameters used:
max_depth = 5
min_samples_split = 2
min_samples_leaf = 14
Test accuracy: 0.1241486068111455

Opponent's ELO rating:1500
Opponent's starting color: white
Your ELO rating:1450

Results from best_opening model
Best opening line to follow: ['Sicilian Defense']
```

Challenge Goals:

New Library:

I utilized the chess library to assist in deciphering which was more beneficial the knight or the bishop. Because in chess notation it only indicates the piece moving and the location moved to (including information about checks and captures), it can be difficult to decern a piece's value. There are many available methods and classes within the chess library that proved useful. Including but not limited to; a way to replicate an entire chess game from standard algebraic notation, check which piece type is intending to move, check if a move is considering castling, check if a move is considering a promotion, send a board position to a chess engine (stockfish) for evaluation, and calculating positional scores for each respective side. The stockfish analysis proved a more systematic way of evaluating each move that an initial approach I was considering. Without the chess library my question would have been magnitude harder to answer confidently.

Machine Learning:

In my third question, I wanted to make inferences about what opening to play based on my opponent's statistics. Therefore, I created a machine learning algorithm (decision tree classifier) and trained several models with different hyperparameters to discover which model produced the highest win rate against each respective opening across the entire data set. Choosing to manipulate the max_depth, min_samples_split, and min_samples_leaf, I was able to find the optimal set of hyperparameters that yielded the highest test accuracy for my decision tree classifier model. Another challenge within this was ensuring proper features to train since there are so many variables to consider within a game of chess. I decided that the winner and both player's FIDE title was the most important. And that the labels would be best utilized if they were just the mainline opening name to remove conflicts and complications.

Work Plan Evaluation:

Import and Clean Data | estimate 0.5 hours | actual 2 hours:

- Convert the datafiles (csv) into a dataframe.
- Clean the data so it only contains the columns listed above in methodology step 1.
- Eliminate casual games.
- Eliminate novice players.
- Perform proper testing to ensure validation.

Feedback: Took longer than expected because I forgot to include time for testing/validating the import and cleaning of the data worked.

Best Opening Move | estimate 2 hours | actual 8 hours:

- Create a new dataset so that the winner is white and won with a mate or resignation and the turns are greater than 18.
- Parse through the move column to store the first move of every single game and whether keep track if white won with that move or not for that game.
- Considering only opening moves where at least 3% of players used it in the valid dataset, find the percentage of games where white would win with each opening.
- Find the opening move with the highest percentage of winning for white.
- Perform proper testing to ensure validation.
- For each of the opening moves where at least 3% of the players used it in the valid dataset, plot the following:
    - Distribution of opening moves
    - Distribution of opening moves where white wins
    - Percentage of wins for each opening move

Feedback: Took longer than expected because I changed my approach to calculating the optimal opening move after completing my first approach, which include testing/validation and plotting. Realizing its flaws, I tried this new approach which added more time. Also, I forgot to include time for testing/validation and plotting in my estimate.

Bishop or the Knight | estimate 15 hours | actual 20 hours:

- Clean data so it only contains the columns listed above in methodology step 1.
- Eliminate casual games.
- Eliminate non-master layers.
- Import chess, chess.engine, and stockfish correctly.
- Simulate one chess game correctly from the moves provided in standard algebraic notation.
- Perform positional analysis for the starting position in the game.
- Perform positional analysis for every move in the game.
- Find the change in positional analysis score after every move in the game.
- Discover/record the piece type that corresponds to the change in positional analysis.
- Update the overall aggregate positional score for that specific piece type.
- Perform proper testing to ensure validation that algorithm works for a single game.
- Modify code so it works for multiple games.
- Perform proper testing to ensure validation that algorithm works for multiple games.
- Plot the outcome of the overall piece evaluation.

Feedback: Took longer than expected because I changed my approach to calculating the value of each piece. Learning a new library was more time consuming than first expected. Also did not expect the runtime to be this long. It took 20 minutes to analyze 132 games and initially I wanted to analyze 20k games. Finding proper testing/validation was initially difficult because stockfish does not produce the same exact analysis score every single time. Also, I forgot to include time for test/validation and plotting in my estimate.

Optimal Opening depending on Opponent | estimated 15 hours | actual 15 hours:

- Convert both players ratings into their respective FIDE titles.
- Convert the opening into its mainline title name.
- Test both conversions stated above.
- Split into 60:20:20 for training, developing, and testing.
- Discover optimal hyperparameters.
- Test against the testing dataset.
- Ask user for critical information about their hypothetical game.
- Convert given information into a “game” for the model to read.
- Predict the best opening line for the user to win that “game”.

Feedback: I experienced difficulties choosing the ranges for the hyperparameters I chose. I found a specific article helpful and used those ranges (5). Most of the time was spent creating a proper “game’ to run the model on. This was difficult because of all the dummy variables

created since the columns held non-numeric values and I needed to create a game with the same number of dummy variables.

Testing:

1) *Import and Clean Data:*
   I tested both the original data file and a smaller data file using asserts. Since I know initially how many games and columns are in each both data files, I can check to see if the data was imported correctly. For the original data file, I know how many columns I should have after cleaning and that the number of rows should be less than what it was initially. For the smaller data file, I know exactly how many columns and rows I should output because I manipulated it as such.

2) *Best Opening Move:*
   I tested this method on a smaller data file using asserts. I manipulated the moves of each game so therefore I would know the outcome of the following asserts: the breakdown of all the opening moves, the breakdown of all the opening moves in which white wins, and the winning percentages of each opening move.

3) *Bishop or Knight:*
   I tested this method on a smaller data file using asserts and print statements. Since this is a hard method to test explicitly because stockfish does not give the same score analysis every time. I utilized print statements to check that my algorithm was performing as needed. More specification for each print statement can be seen in the comments of the code. In addition, using a smaller data file where I know there's only certain piece types that are moving, it was easier to explicitly check if the outcome was expected with asserts. For example, I have 30 games where only pawns and knights moved, therefore I checked to see that only pawns and knights had non-zero scores.

4) *Optimal Opening depending on Opponent:*

   I tested its helper methods using smaller data files using asserts. I manipulated the ratings of each game so therefore I would know the outcome of the asserts, which was the rating of each player for all 30 games and the mainline opening name. And to test whether I created the user's "game" correctly I utilized some print statements to see if it was producing the correct output.

Collaboration/References:

This project's sole contributor is Arden Chaing.

Chess library documentation.

Stockfish download for chess engine analysis.

[1]: https://bit.ly/3s4bTGE

[2]: https://bit.ly/3qLT5Mh

[3]: https://bit.ly/3cpssX9

[4]: https://bit.ly/2OfBrT7

[5]: https://bit.ly/3lgjhfS