Project Title: Strategic Checkmate - Predicting Chess Outcomes using Machine Learning
Author: Dominik Davidhi Chavez
Date: December 5, 2025

1. Introduction

Chess is often described as a game of perfect information, yet predicting the outcome of a match remains a complex challenge due to the human element involved. In this project, I analyzed a dataset of over 20,000 games played on the Lichess platform. The primary objective was to build a binary classification model capable of predicting whether the "White" or "Black" player would win based on pre-game attributes such as player ratings and time controls.

My analysis reveals that while player rating is the dominant predictor of success, the specific conditions of the match (such as time constraints) also play a role. The final Random Forest model achieved an accuracy of 65.54%, significantly outperforming the baseline.

2. Data Description

The dataset was sourced from the Lichess.org game library (via Kaggle/Google Drive). It originally contained approximately 20,058 rows and 16 columns, including game metadata (IDs, rated status), player information (ratings, IDs), and game specifics (moves, opening codes).

Data Cleaning & Preprocessing:

- Target Filtering: To focus on decisive outcomes, I removed matches ending in a "Draw," reducing the dataset to a strict binary classification problem (White Win vs. Black Win).
- Feature Selection: I excluded post-game features (such as turns or victory_status) to ensure the model only used data available before the first move was made.
- Feature Engineering:
    - Rating Difference: Created a new variable (white_rating - black_rating) to capture the relative skill gap.
    - Time Control: Parsed the increment_code (e.g., "15+2") into separate base_time and increment integer columns to analyze the impact of game speed.

3. Models and Methods

I approached this as a supervised classification problem. After splitting the data into a training set (80%) and a testing set (20%), I implemented a robust modeling pipeline:

1. Baseline Model: I established a "Dummy Classifier" to determine the accuracy of simply guessing the most frequent winner. This served as the floor for performance.
2. Logistic Regression: I applied this linear model to establish a benchmark and understand the direct linear relationships between ratings and victory probability. Data was standardized using StandardScaler to ensure optimal convergence.
3. Random Forest Classifier: I implemented this ensemble method to capture non-linear relationships and complex interactions between features.

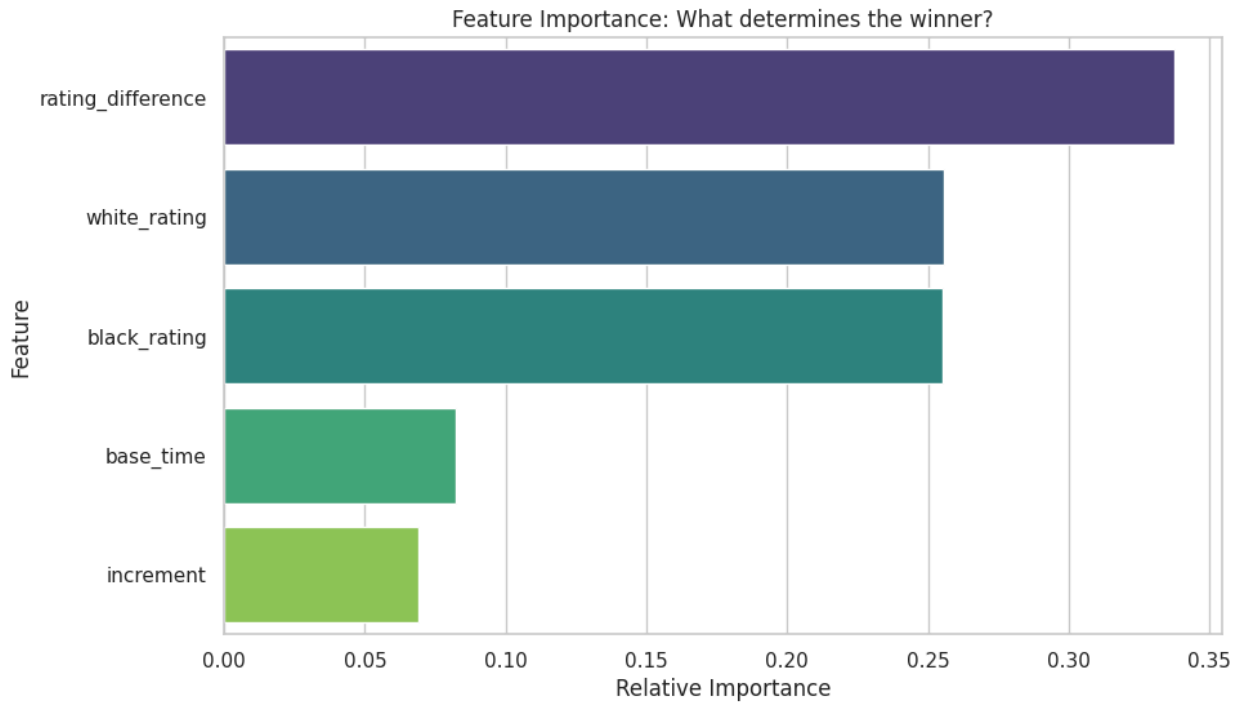
4. Results and Interpretation

The modeling process yielded the following performance metrics on the test set:

- Baseline Accuracy: 52.07% (The "floor" - guessing White every time).
- Logistic Regression: 65.02%
- Random Forest: 65.54%

Interpretation: The Random Forest model achieved the highest accuracy of 65.54%, providing a 13.47% improvement over the baseline. However, it only marginally outperformed the simpler Logistic Regression model (a difference of just 0.52%).

This suggests that the relationship between rating difference and victory is largely linear. While the Random Forest model captures some complex interactions, the sheer "Rating Difference" is the dominant factor.



5. Conclusion and Next Steps

This project successfully demonstrated that the outcome of online chess games can be predicted with reasonable accuracy using only pre-game metadata. The strong performance of the rating_difference feature highlights the reliability of the ELO rating system.

Next Steps for Further Analysis:

- Opening Theory: I would like to incorporate the opening_name or opening_eco into the model using One-Hot Encoding. This could reveal if certain openings provide a statistically significant advantage for White.
- Move Analysis: Future iterations could analyze the first 5-10 moves of the game (using the moves column) to update prediction probabilities in real-time as the game progresses.