Finding Elo

Predicting FIDE Elo Ratings from A Single Game

Goonethilleke Amanda
Rutgers, The State University of New Jersey
New Brunswick
amanda.goonetilleke@rutgers.edu

Okubor Miracle
Rutgers, The State University of New Jersey
New Brunswick
miracle.okubor@rutgers.edu

ABSTRACT
This project for the Rutgers University Department of Computer Science Topics in Computer Science: Introduction to Data Science (01:198:443) course. Taught by Tina Eliassi-Rad for the Spring 2015 semester. This project is based on the Kaggle competition, *Finding Elo*. The aim is to determine the FIDE Elo score of a chess player based solely on the strategic moves made in the duration of a single game [3].

Categories and Subject Descriptors
I.5 [**Pattern Recognition**]: Models; I.5.2 [**Design Methodology**]: Classifier design and evaluation—*Feature evaluation and selection, Pattern Analysis*

General Terms
Chess, Features, Regression, Supervised Learning

Keywords
Kaggle, Elo rating, Linear Regression, Data Science

1. INTRODUCTION
The Elo rating system is a method for calculating the relative skill levels of players in competitor-versus-competitor games such as chess [2].The Elo rating is represented by a discrete value which increases or decreases based on the outcome of games with rated players. A first time players ranking is calculated using the average rating of his rated opponents during a tournament, the average result of all games played during the tournament [1].According to the online FIDE handbook, there are many criteria that must be met before an unrated players score can be calculated, a minimum number of games in a tournament, minimum number of rated players in a tournament, etc. Typically, a players rating gets more accurate after playing several games. Traditionally, Elo ratings are updated at the end of a tournament. The player gains points after defeating a player with a higher Elo rating and loses points after losing to a player with a lower Elo rating.

Intuitively, in any competitive game, the player that wins is believed to have more skill and deserves a better rating than the player that looses. According to the Kaggle competition description, recent work on chess analysis has focuses on intrinsic performance ratings, where one assesses skill bases on the quality of the decisions rather than the outcomes of games. In a strategy based game such as chess, a decision is defined as the moves a player makes. This intrinsic approach provides the rating system with more data because there are more moves on which to base a players rating than games. Following this approach, the competition challenges participants to determine a players FIDE Elo ranking based on the strategic decisions made in a single game.

2. RELATED WORKS
Previous works using the intrinsic approach have involved calculating a players Elo rating after a tournament or series of games based on the idea of quality of decision made rather than the outcome of contests.This is still preferred over the traditional FIDE Elo rating system because outcome depends on the skill of opponents and the unquantifiable ”luck”. The intrinsic approach also provides a much larger sample size for training prediction models.

3. BACKGROUND INFORMATION
Each chess game in our data set is in a Portable Game Notation format which is a plain text recording of chess games which contains information such as the outcome of the game, the name of the black or white player, the name of the tournament and most importantly the moves made during the game in either Universal Chess Interface format or Standard Algebraic Notation.

To implement our idea, we needed a way ascribe a value to each move made during a game. This is done by running a chess game through a chess engine which analyses piece positions and assigns a score to a players move based on the best possible moves calculated by the engine. We use the Stockfish chess engine to get the scores for our game moves [4].

4. PROPOSED APPROACH
Our problem is a supervised learning problem because have the corresponding target values (Elo Scores) in our data set. This is also a regression problem because we are creating a model that will output a real number - an Elo Score. We chose to model this problem using Linear Regression because

we believe there is a naturally linear relationship between higher Stockfish scores (good decisions) and high Elo scores.

5. EXPERIMENTS

5.1 Set-Up

The first set in our experimental process was to combine the data from our given Stockfish and PGN dataset into a single CSV file by coding a Python script so we could easily map and retrieve related values. Then we extracted features from each game. We chose to create two models, white and black because white is given a first move advantage in the game. Then within each game we partitioned it equally into three sections to mimic the opening, mid-game, and end-game in chess. From each partition we extracted features such as the min, max, and mean stockfish scores. Using these datasets and features we made our model using Linear Regression to predict Elo Scores. This model was generated using the Python package sklearn linear modelâĂŹs Linear Regression feature. We used ten fold cross validation to evaluate our model using the Python package sklearn cross validation, as well as sklearn metrics to calculate the Root Mean Square error. We used 10 fold cross validation to evaluate our model.

5.2 Datasets

The datasets in our experiment are a PGN dataset which includes Elo Scores and the moves in a game, as well as a Stockfish dataset which includes the corresponding Stockfish score for every move in a game - measured in centi-pawns. In our training dataset we had 25,000 games. In our test dataset we had 25,00 games. Our training and test dataset were both comprised of a PGN dataset and the Stockfish dataset. Because we had two models, we had two corresponding target variables - Black Elo and White Elo which are both real numbers. All of our features were real numbers - NumWhiteMoves, NumBlackMoves, WhiteStockfishMinScore,BlackStockfishMinScore, blackStockfishDeltaStd, blackStockfishMaxScore, blackStockfishMeanScore, whiteStockfishDeltaStd, whiteStockfishMaxScore, whiteStockfishmeanScore, WhiteWins, BlackWins, Draw - where features like BlackWins had the corresponding value of 1 if true, and Draw being .5. Each of these features were further divided by sections in the game based on the three partitions we established, for example blackStockfishMeanScore1, blackStockfishMeanScore2, and blackStockfishMeanScore3 for each partition in the game.

5.3 Results

In terms of cross validation results, the average white error over 10 splits was 193.894, and the average black error over 10 splits was 196.863. To get these results for every split we took the absolute value of White Error to be the absolute value of the test white Elo score minus the predicted White Elo Score, and once we added all of the absolute white errors together we divided it by the total number of games to get the error of 193.894. We did the corresponding calculations for black. To be clear, our error is not 193 percent, rather it is measured in Elo points. Based on these results we can tell our white model did marginally better than our black model. Our Root Mean Square error for white was 241.364726992 and for black it was 243.246266632. Once again supporting that our white model did a little better.

6. CONCLUSION

The findings of this model are important because they provide a more comprehensive ranking system for unrated players. This approach is especially beneficial to them because it ranks them based on the strength of their decisions and not on overall outcomes against seasoned, ranked players. A players moves may not reflect their absolute skill but as more games are ranked, we have a good way to measure a players improvement and growth.

The intrinsic approach gives a general idea of a players rational ability but our approach does not take into account possible external factors, not including the still unquantifiable "luck". Another limitation of this model is that while the moves from one game do provide sufficient data, the moves from multiple games would provide our model with even more data which could introduce features not considered in our current model, such as the average result over all games played by a player, which could strengthen the predictive capability of our model.

Improvements can always be made. A further step would be to come up with an algorithm to definitively separate a game into the beginning, middle and end sections. Also, using the intrinsic approach, we could train a model to determine if a player is cheating by performing moves above their skill level.

7. REFERENCES

[1] fide.com. Fide rating regulations effective from 1 july 2014. website, June 2014. https://www.fide.com/fide/handbook.html?id=172view=article.

[2] t. f. e. Wikipedia. Elo rating system. website. http://en.wikipedia.org/wiki/Elo$_{r}ating_{s}ystemPerformance_{r}ati$

[3] www.kaggle.com. Finding elo. website, October 2014 - 2015. https://www.kaggle.com/c/finding-elo/.

[4] www.stockfishchess.org. Stockfish. website, 2010 - 2015.