Predicting the winner of chess games

Ben Purdy

Theodore Davis

We are using the Chess Game Dataset (Lichess) by Mitchell J available at:
https://www.kaggle.com/datasnaek/chess

What are you trying to do?

Using a data set found on kaggle, we will try to predict the winner given the players chess ratings, who has the first move, and the opening position. Our data set has 20,050 games collected from the site Lichess.org. This data contains a range of players from different skill levels ranging from the low 700s to 2700. This creates a bell cursive and should make outlier data less of an issue in our training data.

The author also tells us how to collect more data from Lichess.org through their api.

How is it done today and what are the limits of current practice?

Currently in place are the Elo System and the Glicko System. These systems are in place to give players a ranking to see how they match up with other players. The Elo system allows a player to calculate their percent chance of winning against the other player. However, there can be uncertainty in this prediction. The Glicko prediction accounts for uncertainty by adding a RD, rating Deviation, which tells the algorithm how confident that a player is between two different rankings.[1.] To improve on this prediction of who will win we plan to also account for the first move of both players and possibly several moves afterwards.

What is new in your approach and why do you think it will be successful?

The game of chess is primarily a game of patterns and data science has benefited from this aspect since Deep Blue versus Garry Kasparov. Using this knowledge classification is a starting point given the nature of 2 players and the data provided. To find patterns within our data set I suspect a combination of tools can be used together to predict higher accuracy then solely an classification approach. A combinationation of logistic regression to obtain a classification set and then randomforest to build out a probability tree might be a successful first attempt. The second approach we'd like to try is to implement a deep learning/supervised learning model to compare the results.

If you are successful, what difference does it make?

If we are successful then we'll be able to more easily define chess games. We'll also be able to classify opening moves based on user skill and compare them to other opening moves to see if one is comparably better or a counter to another.

What are the risks?

There are no personal risks involved with this project. However, if a player were to use the prediction of our project, and it was accurate. Then he'd be able to decide his next move based on the prediction of the project and not his own skill. Thus him cheating in the game.

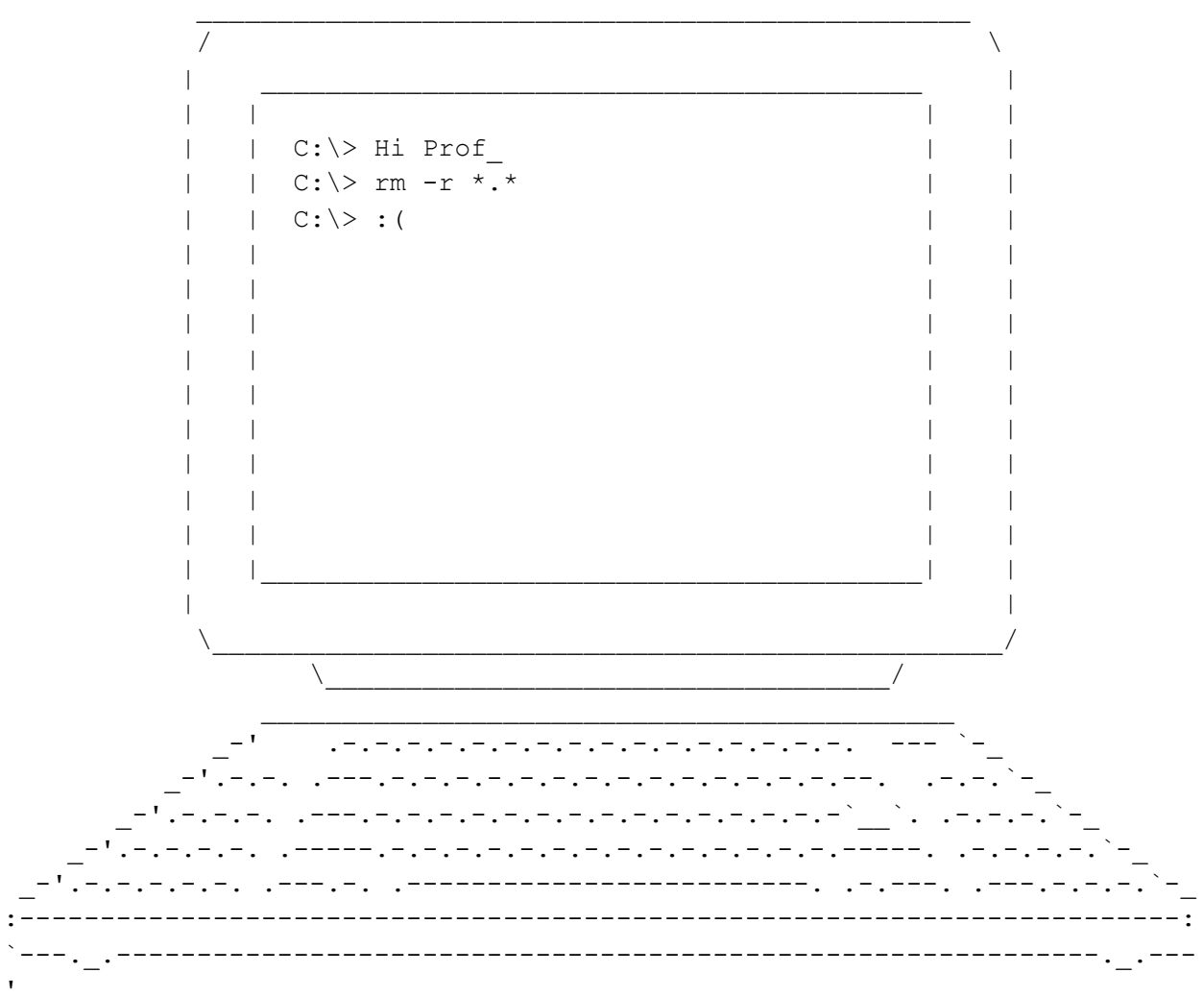
How long will it take?

Since the data is relatively small, only 20,000 rows, a surface level analysis would take less than a day. We plan on starting simple and expand the depth of our analysis. Overall I predict the project to take a couple of weeks of consistent work from both of us to complete the project successfully.

What are the midterm and final "exams" to check for success?

A midterm exam for our project would see if we can accurately predict the winner of the chess game based off of the first move. Further along in the project we can expand the exam to include more moves after the first.

1. https://www.chess.com/article/view/chess-ratings---how-they-work


C:\> Hi Prof_
C:\> rm -r *.*
C:\> :(