Inferring Chess Skill from Chess Rating and Recent Win/Loss History is More Accurate than from Chess Rating Alone

Kathy Shi (kathy.shi@yale.edu)
Yale Department of Psychology, 100 College St.
New Haven, CT 06511 USA

Abstract

Humans are prone to interpreting a string of repeating values in a random set as non-random, or a "streak". Expected outcome of an event affects how much effort humans will expend. In order to investigate the effect of expecting a win or loss in the next game in chess on the player's chess playing skill in that game, I built a Bayesian inference model that infers player's chess skill from their recent chess win/loss record (recent history) and their current ELO rating. This model does better at predicting a game's outcome with the true recent record than it does with a randomized recent record and the best alternative no model algorithm of choosing the outcome based on whichever player's ELO rating is higher.

Keywords: Expectations; motivation; Bayesian modeling; chess

Introduction

Every day life is full of probability estimations. From deciding whether to play the lottery to deciding to further practice a basketball shot, the odds of winning something is integral to making choices about what to do. However, humans perform suboptimally at many forms of probability estimation. One such form of probability estimation is that of seeing "streaks". It has been shown previously that people believe that random events must be "random" even with a small sample size, e.g. a truly random coin flip will alternate between heads and tails frequently, termed by Tversky and Kahneman (1971) as the "belief in the law of small numbers" . This belief leads to the interpretation of a streak of head or tails on a coin flip as the coin is not random.

In a sports setting (such as basketball), a streak of what is believed to be above average performance is termed as having a "hot hand", which has been previously shown to be a fallacy in the same way that seeing a coin flip with a streak of heads and tails and believing it to be weighted is a fallacy (Gilovich, Vallone, & Tversky, 1985). However, what Gilovich et. al. does not consider is that basketball shooting and coin flips differ in a fundamental way, which is that basketball is a *skill* driven event while coin flips are *luck* based, or completely random. A basketball player's performance can be modulated by how much effort they put into their performance.

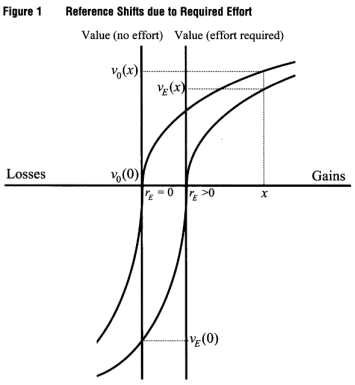

Figure 1: This figure from (Kivetz, 2003) clearly highlights how effort can change utility calculation. In a high effort state, outcomes have to be higher (right shifted) to achieve the same amount of utility.

In behavioral economics, prospect theory is a way to describe how outcomes in a risky decision making task can translate into utility, the economic measure of how much enjoyment an outcome brings a person (Kahneman & Tversky, 1979). There has been work to conceptualize a shift that different amounts of effort can shift the utility function curve, making outcomes produce different utilities based on how much effort was put in (Kivetz, 2003).

While this deals with how current effort changes how current outcomes are evaluated, it lends credence to the idea that expected outcomes can shift the amount of effort put into an action. For example, if one expects a bad outcome, they may want to shift their effort state from a highly effortful one to a less effortful one to minimize their utility loss. However, when this effort state is directly linked to their outcome, such as in a skill-based event, this downregulation of skill to minimize utility loss when expecting a loss can be a self-fulfilling prophecy.

To test whether someone's expectation of a poor

outcome can shift their effort negatively and thus bias them to fulfill their expectation of a poor outcome (and vice versa), I sought to investigate chess data. Chess is well-suited for this investigation primarily because it is (1) skill-based and (2) has a wealth of open-source data. Using these data and the hypothesis that streaks and the expectations they produce bias player's performance, I built two models, one with player's win/loss history as well as their current chess rating and one with purely their current chess rating. In comparing these two models, I hoped to find if including the recent player history, particularly streaks, can give predictive information above and beyond what their chess ratings alone predict.

Methods

Data

The chess playing data were derived from the open-source database ("https://database.lichess.org/") of the online chess playing website "lichess.org". The data were divided into games played in one month on the site.

In order to decrease amount of games in the file, I used the earliest month available, January 2013, which, after removing players with less than 100 games (and those games), had 204,575 games and 700 players. To further limit the amount of games, I only included players who played between 300 and 400 games in that month (an arbitrary selection), which limited the number of games to 24,634 games and 72 players.

To test the idea that it is *streaks* of wins or losses that contributed to the predictive power of recent history, I did not include any games where neither of the two players had an all win or all loss recent history. This left 498 games that were used in my final analysis.

From the games, I extracted the observed chess rating (*elo*) and recent *history* to feed into my model(s), as well as the win/loss record to compare to my model(s) and other possible algorithms.

Bayesian Foundation

Applying the classic Bayes' Rule ($p(A|B) = \frac{p(B|A)p(A)}{p(B)}$) to my problem of inferring chess skill from *elo* and the recent *history*:

$$p(skill|history, elo) = \frac{p(history, elo|skill)p(skill)}{p(history, elo)}$$
$$p(skill|history, elo) \propto p(history, elo|skill)p(skill)$$
$$p(skill|elo, history) \propto p(history|elo, skill)p(elo|skill)p(skill)$$

For simplicity, I assumed $p(skill) = 1$

$$\boxed{p(skill|elo, history) \propto p(history|elo, skill)p(elo|skill)}$$

Generative Model

To create the prior, I assumed that the *elo* was a normal distribution around the *skill* with standard deviation of 12. This was to contain 99.9% of data within $skill \pm 40$, which seemed like a reasonable cutoff for possible fluctuations of *elo* around *skill*. That is, it seemed unlikely that someone would display an elo that was more than 40 points above or below their actual skill ability, since the change in elo after any single game was, from visual inspection, usually around 0 to 30.

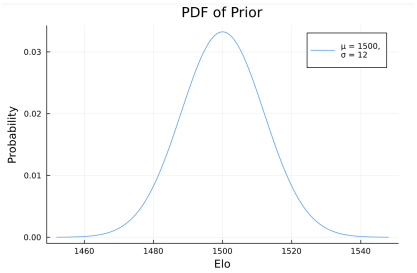

Figure 2: This is an example of what a prior PDF may look like. The prior is the observed elo given a true skill. It is a normal distribution with a mean of the true skill and a fixed standard deviation of 12.

To create the likelihood, I assumed that the *history* (or the fraction of wins over the number of games played) was a function of the difference between the *skill* and *elo* s.t. if $skill - elo > 0$, $history > 0.5$ and if $skill - elo < 0$, $history < 0.5$. Since history cannot be below 0 or above 1, I truncated the distribution to between 0 and 1.

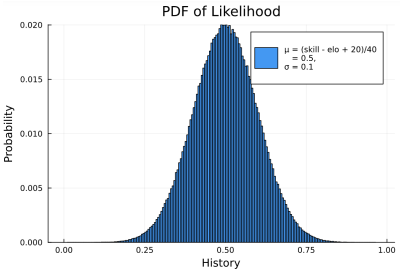

Figure 3: This is an example of what a likelihood PDF may look like. It is a transformed truncated normal distribution from a mean of skill minus elo to a mean of 0.5, which is now the history.

Inference Generation

To create the posterior distribution, I used the elo as the peak of a probability density function with exponentially decaying left and right sides. The steepness of the decay was determined by the recent history:

left side's parameter = $history * 0.03 + 0.03$

right side's parameter = $(1 - history) * 0.1 + 0.1$

so that (1) a string of losses would make the left side less steep and the right side steeper while a string of wins would make the right side less steep and the left side steeper and (2) the decay on the right side was always steeper.

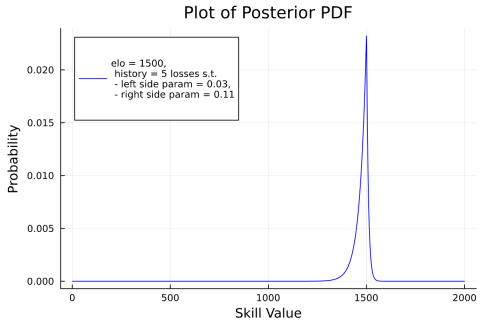

Figure 4: This is an example of what a posterior PDF may look like.

Inference Procedure

Full Model This model was implemented by feeding it the recent history for the past five games for each player and the skill of each player for every game where at least one of the players had played six or more games and at least one player had either all wins or all losses in the past five games. Then, I ran importance resampling to minimize the difference between the *elo* and *history* that my generative model generates based on inferred skill and what the true *elo* and *history* are. The importance resampling procedure had 50 iterations. I ran the procedure 10 times and drew one at random for the trace I used and stored the skills for each player.

History Lesion Model This model was implemented in the same way as the full model, except the *history* for each player was a random number drawn from a uniform distribution between 0 and 1.

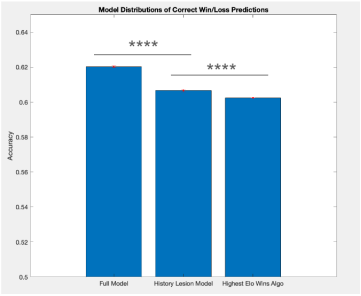

Figure 5: The full model outperformed the lesioned model by 1.36% ($p < 0.0001$) and the "Higher Elo Wins" algorithm by 1.78% ($p < 0.001$) with 100 runs of the models. The lesioned model slightly outperforms the the "Higher Elo Wins" algorithm.

Win/Loss Record Generation

Finally, the inferred *skill* was used to generate a win/loss record, which was then compared to the true win/loss record. This is done deterministically by having the player with the higher *skill* win.

Results

The Full Model Performed Best in Win/Loss Prediction Accuracy

The full model was about 1.36% better at predicting the true win/loss record than the History Lesion Model. This suggests that the history is important to the model, as without it, the model does worse. The model still does better than a simple algorithm of choosing the winner by whoever has the higher elo ("Higher Elo Wins"), suggesting that the architecture itself has some advantages over a simple use of *elo*. However, the difference between the History Lesion Model and the "Higher Elo Wins" algorithm is only 0.4%, suggesting that the bulk of the better performance of the model comes from the inclusion of history.

History Drove Changes Between Inferred Skill and Observed Elo

Looking at the difference between the inferred skill and the elo of each player, the full model appears to be bimodal, with one peak around -20 and another around +20. For the history lesioned model, the distribution appears to be more focused around 0 and is unimodal. These results are reasonable,

since the full model's history was drawn from a distribution centered around 0 and 1. This means that the difference between *skill* and *elo* should be pushed to either positive or negative, since a 0.5 history means the difference was 0. However, since the history distribution of the lesioned model was uniform, the distribution of difference between skill and elo reflects that, clustering around 0. It appears that this bimodal change in skill based on their history makes the prediction of win/loss slightly more accurate.

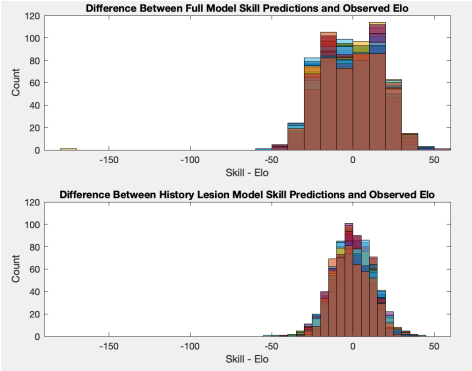

Figure 6: These are the histogram plots of all the skill-elo differences for each model. Each color on the histogram represents a different run of the 100 runs of the model. In general, the runs overlap and agree with each other

The Model Underperformed When History Was Not Limited to Only Streaks

This model did not beat the "Higher Elo Wins" algorithm if the data was not limited to streaks. I am unclear as to why this would be. However, the Full model still beats the History Lesion model, indicating that having history gives more information than not having history.

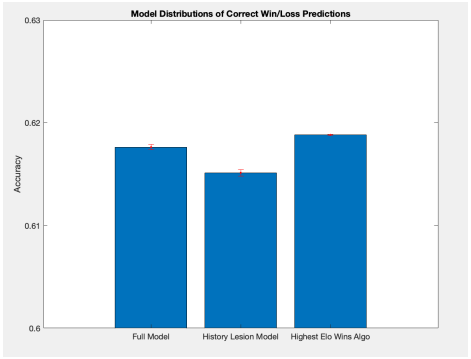

Figure 7: The full model outperformed the lesioned model by 0.3% ($p < 0.01$) but underperformed "Higher Elo Wins" algorithm by 0.1% ($p < 0.001$) with 100 runs of the models.

Discussion

Interpretations

Recent history of chess games may inform next outcome beyond information contained in elo ratings. This is a promising first step for my initial hypothesis, although many more steps must be taken. For one, assuming that the model truly performs better and it was not an artifact of this particular dataset, all it informs us is that recent streak history is informative. It does not illuminate what cognitive mechanisms lie behind it. Although I have given an account of what might be occurring, it does not necessarily need to be true, from this model's success, that players were *expecting* better or worse, which lead to them *performing* better or worse due to downregulation of skill. One piece of evidence is that when I did not exclude non-streak history, the performance of my model does worse. In fact, it does not beat the "Higher Elo Wins" algorithm. This implies there is something special about a streak of wins, or that this particular dataset works well for my model. Thus, I would like to continue testing my model to make sure it is not from another confounding factor.

This model could also be a tool to inform chess rating systems. The system used by Lichess is not the actual ELO system, which is a fairly simple algorithm, but the more rigorous Glicko system (specifically Glicko2), invented by Dr. Mark Glickman in 1995. This system builds in volatility,

but the volatility is not principled. Perhaps this is why my model with history improved on it by only a small amount, since Glicko2 already captures volatility (Glickman, 2022). Still, it may be useful to consider more directed volatility changes and perhaps finding the largest factors of this volatility can give us an even better tuned rating system.

Limitations

The main limitations to this project were time and computational resources. Due to the large size of the data, even after my attempts to limit it, the models would take a very long time to run and be difficult to make small changes. I also had trouble with the cluster. Since my model needed longer than 2 hours to run (OOD logs you out after that amount of time), I pulled them off the cluster and onto my personal computer. Given more time, I would have figured out how to run jobs using the terminal and slurm on the cluster to implement a better model.

Future Directions

Use More/Different Games The simplest change would be to include more games. 498 games is not a small number, but I have found that the amount of games and which games you include changes the accuracy quite a bit, although it usually does not change the relative accuracy. With such a large sample, it would be more convincing to see these results hold regardless of how many games are included. Testing on a different month can give surety about the robustness of this finding.

Another interesting direction may be to see a difference between different types of players, such as those with high elos vs low elos. It may be interesting to see if the "Higher Elo Wins" algorithm may work better in comparison to my model for those with higher elos, who may be less likely to be affected by a string of losses. I have seen that limiting the dataset to those with more than 1000 games in the month increases the "Higher Elo Wins" algorithm to about 68% accuracy. I have not ran my model against it due to time constraints but it may be interesting to see if the model still does better or if elo is sufficient when it has enough datapoints.

One last interesting change in use of games is to change the amount of games needed to win or lose in a row to be counted as a streak. I believe that five is a reasonable number, as if I consider winning or losing five times in a row, I would believe I am in a streak, but this number is arbitrary. Further tests or reading of the literature can be done to see what number may be optimal.

Run Longer Inference Procedure Another change would be to increase the number of iterations in the importance resampling from 50. In class, most of our use of importance resampling had us take about 1000 iterations and run them 100 or 1000 times to get to the final inference. Running more iterations should improve my inference by making sure the data (*elo* and *history*) are better recovered. However, that would rely on having a reasonable architecture for the generative model, as a wildly inaccurate one would have my model moving towards an incorrect space.

Refine Prior and Likelihood Distributions In the generative model, I assumed a normal prior and likelihood. This could be incorrect and perhaps further consideration of the problem may lead to a principled change in the distributions. I also chose the standard deviation of the prior arbitrarily, although it was reasonable. Analysis of the elo changes after a game may give better insights about what that standard deviation should be, because this elo change reflects a sort of uncertainty about the player's true skill.

Use More Data Within Each Game The only information I used from the data were the player's elos and who won the game. However, this is the full representation of what information is within one game:

Even for this question, there are many ways to use these data. What's most relevant is likely the time and date. If the effect is truly because of a streak expectation, then it should be stronger when the games are played back to back in time. Furthermore, I could run (or build) a model that considers the goodness of moves and evaluate how well they won or badly they lost to factor into where their true skill lies for that game and how they may fare in future games. The amount of data available is astonishing and could be valuable to answering many

Table 1: Chess Game Information

Category	Example
Event name	Rated Classical game
Website name	https://lichess.org/abcde123
White player name	Jane Doe
Black player name	HM
Result	[0-1]
UTC Time	2013.01.01
UTC Date	00:50:31
White Rating	1887
Black Rating	1623
White Rating Change	-21
Black Rating Change	+19
ECO	D00
Opening Name	Queen's Pawn Game
Time control	240 + 0
Type of Termination	Normal
All moves	1. e4 e5 2. Nf3 Nc6 ... 0-1

types of questions about skill, planning, and what affect these processes.

Build a New Model to Explore the Underlying Cognition Lastly, should this finding be verified, I would like to dive into exactly what the mechanisms behind this bias in players to lose more after a streak of losses and win more after a streak of wins. I would like to refine my hypothesis and build a model that can truly test the underlying cognitive processes.

Data and Code Availability

All code used can be accessed at:
`https://github.com/kathymshi/AotM_finproj`

Acknowledgments

Thank you to Ilker Yildirim for the conceptual direction and clarifications. They were incredibly helpful. Thank you also to Dongyu Gong and Zihan Wang for moral support.

References

Gilovich, T., Vallone, R., & Tversky, A. (1985). The hot hand in basketball: On the misperception of random sequences. *Cognitive Psychology*.

Glickman, M. (2022). *Example of the glicko-2 system*. Instruction manual, Data Science Initiative, Harvard University.

Kahneman, D., & Tversky, A. (1979). Prospect theory: An analysis of decision under risk. *Econometrica*.

Kivetz, R. (2003). The effects of effort and intrinsic motivation on risky choice. *Marketing Science*.

Tversky, A., & Kahneman, D. (1971). Belief in the law of small numbers. *Psychological Bulletin*.