Tableau Coursera course project writeup

Project Proposal

With this visualization and data exploration project, I hope to show that there isn't a significant difference in chess performance on Lichess.org between weekdays (Mo-Th) and weekends (Fr-Su). The goal is to discourage time-specific playing of chess in order to inflate Lichess ratings. Additionally, I wish to see whether such an effect exists in my personal games, as well.

The audience that this project is aimed at is mostly Tizian Bonus, but also myself. I'm curious to know if there is an actual statistical difference in his performance on weekends versus on weekdays, which he has an impression that there is ($P(\mathrm{win}|\mathrm{rating}, \mathrm{weekday}) > P(\mathrm{win}|\mathrm{rating}, \mathrm{weekend})$). Tizian's hypothesis is that the population of players who play on the weekend is different from the one that plays on weekdays, and thus, that there are different rating equivalencies in each population (e.g.: rating of 1500 on the weekend equates to rating of 1600 on weekdays).

The currently available data is only data based on Tizian Bonus' and my own games. However, I also have easy access to approximately a billion games from the Lichess database: However, I would need to parse dozens of large (~75Gb) CSV files in order to acquire this. For the initial iteration, I'll settle for just using the currently available data, but if this does not prove enough, I'll also parse the Lichess database files in an attempt to find a stronger experimental power. The currently available data was gathered from the Lichess API through my own chess pipeline, while the extended data would be from the Lichess game database. This data can be trusted - I've collected it myself from Lichess, and Lichess itself has a good track record for its data quality.

For any data analysis that cannot be done exclusively, or simply, in SQL, Python will be used. This includes the parsing of PGN game files from the Lichess database, or creating any metrics that do not currently exist in the data.

The main challenge I foresee in this undertaking is in determining the actual performance difference between weekend and weekday days. If I need the different player populations, I will actually need the extended dataset - which also means I'll require resources to parse the files. I suspect I will be able to use NYU's High Performance Computing cluster for this, but this is not a guarantee. However, I can attempt to avoid this by using Stockfish evaluations of moves, along with some form of "time management" metric, to estimate the performance of opponents in both situations.

The final presentation of results will be a single-frame visualization, together with explanatory text and tables, as well as links to the analysis itself.

On a personal level, I aim to practice a statistical approach to problems, as well as learn how to develop good player/game metrics (and where these can be used).

Tools Used

For collecting the data, Python, and specifically my chess pipeline linked above, were used. For storing the data, I used a PostgreSQL server, with a database named `chess_db`. For visualizing data, I used Redash, a free alternative to Tableau, which I self-host.

The Data

The data can mostly be found in my Raspberry Pi Zero W, in the `chess_db` database. The main table for this is `chess_games`, but individual game evaluations by Stockfish can be found in `game_evals`, and individual game clocks can be found in `game_clocks`. As mentioned, the data was gathered from the Lichess API, and for games where the evaluation was not present in the Lichess database, I ran Stockfish over the game myself in order to gather the evaluations.

There should be few missing values, but I'll examine this. This is mostly dependent on how well my chess pipeline code ran, as well as when it was that I added certain elements to the chess pipeline. For instance, there was a period of about 3 weeks in September 2019 during which the pipeline was not functional; this data was mostly re-gathered, so it should be intact. However, the position evaluations were only added in August 2019, while the pipeline itself has existed since May 2019. This means a lot of games prior to August 2019 will have the evaluations missing. I can fill these in after the fact, but it will require work. It might be easier to re-run the entire pipeline starting from early 2018 in order to gather the data in a more consistent manner.

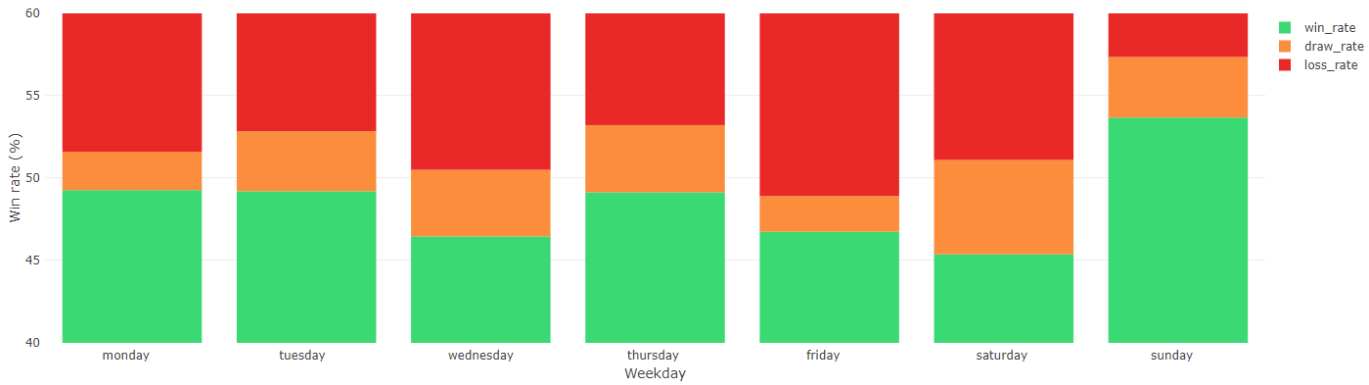
Importing the data into Redash proves simple, since Redash connects directly to PostgreSQL databases.

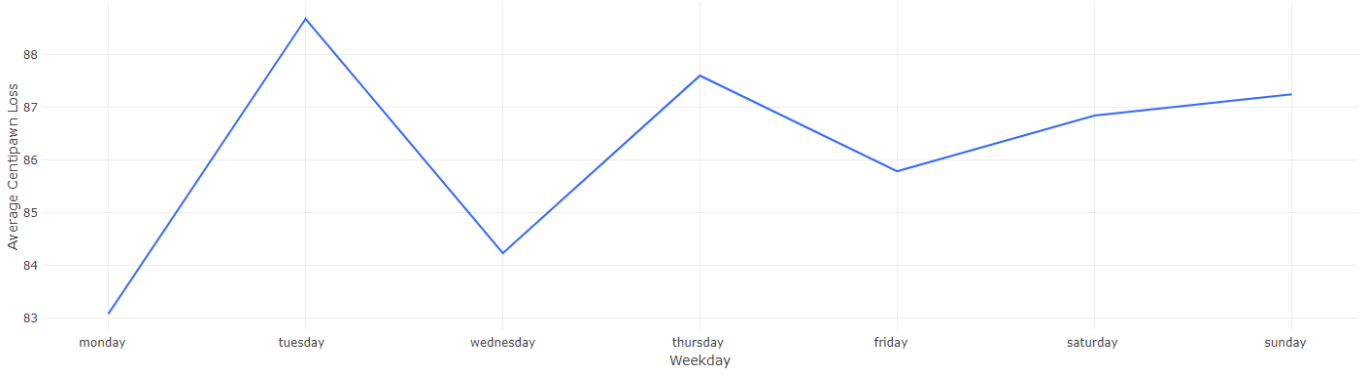
Assessing the data quality itself, it is apparent that there are several games for which the evaluations are missing. This was done by counting how many games are present in the database overall, grouped by player, as well as how many games are present in the `game_evals` and `game_clocks` tables, also grouped by player. For Grahtbo, approximately 70% of the games have evaluations and clock times (4199 games); for siddhartha13, only about 8.5% (864 games) do. For the initial evaluation for Grahtbo, this amount of data should suffice, but for evaluating my own games this might not be enough data. In general terms, the column types match what they should be - I wrote these myself.

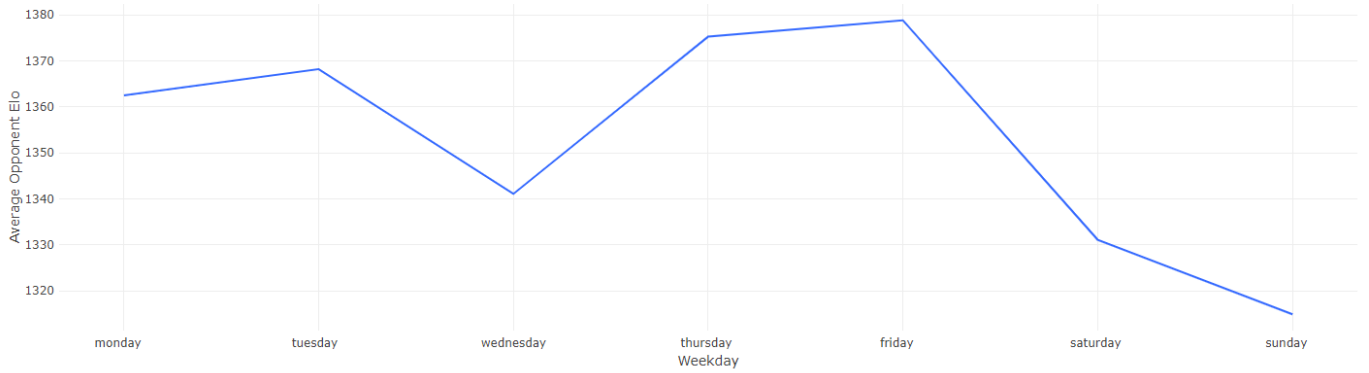
If I end up using the extended data, I will later mention how that data has been prepared.

Exploratory Data Analysis

Here are some visualizations I created:







The data used for these came from **Grahtbo's blitz games, from 02/01/2019 to 09/01/2019**. The centipawn loss values were backfilled using Stockfish, exploring to a depth of 15. The visualizations themselves can be found on my Redash, under this link. Here is the data in tabular form:

Day of week	Average Opponent Elo	Average Centipawn loss	Sample size	Win Rate	Draw Rate	Loss Rate
Monday	1,362.50	83.08	343	49.27	2.33	48.40
Tuesday	1,368.22	88.68	439	49.20	3.64	47.15
Wednesday	1,341.10	84.23	297	46.46	4.04	49.49
Thursday	1,375.32	87.60	466	49.14	4.08	46.78

Day of week	Average Opponent Elo	Average Centipawn loss	Sample size	Win Rate	Draw Rate	Loss Rate
Friday	1,378.84	85.79	599	46.74	2.17	51.09
Saturday	1,331.09	86.85	227	45.37	5.73	48.90
Sunday	1,314.93	87.25	136	53.68	3.68	42.65

Here we assume that there is enough gameplay between Tizian's opponents and other players for their Elo to be an accurate estimate across the entire player pool. With that assumption, it's clear from these graphs that Tizian's average opponents on weekdays were of a different caliber than those on weekends: The weekday opponents were approximately at 1368.06 Elo, while the weekend opponents were at 1325.04 Elo. Overall, his opponents were of 1361.82 Elo. This shows that, in fact, Tizian's opposition was a lot easier to play against on weekends, which could explain his subjective understanding that it was a lot easier to win games on weekends. Examining the win/draw/loss rates more closely, it's clear that Sundays were the best day for Tizian to win games; However, his opposition was also the weakest then. Weighing it out, it seems like Wednesdays were the worst days for Tizian, where his average opponent Elo was relatively low and his loss rate was relatively high. On the other hand, Thursdays were good days: His average opponent Elo was high, but he was able to win his fair share of games.

The question remains whether this Elo discrepancy in weekend vs. weekday is enough to explain his much higher win rate for Sundays. If I calculate Tizian's "expected Elo" from his win/draw/loss rates against opponents, I get the following distribution:

Day of week	Expected Elo
Monday	1365.62
Tuesday	1375.55
Wednesday	1330.23
Thursday	1383.80
Friday	1363.25
Saturday	1318.45
Sunday	1354.60

It's apparent that, in fact, Tizian played worse on the weekends than over the weekdays.

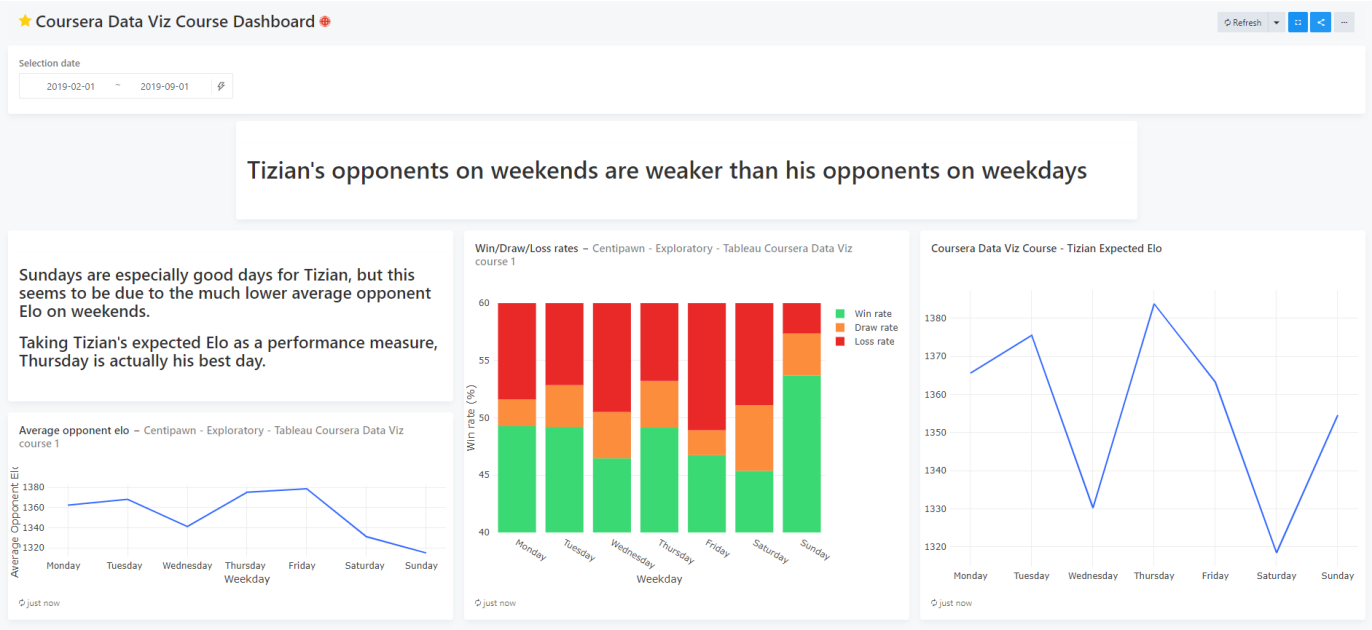
On the topic of centipawn loss, there is no apparent correlation between Tizian's average centipawn loss and his performance. This indicates that centipawn loss might not be the correct measure to use here in order to estimate performance. Given the context of what centipawn loss is - a comparison between the computer engine's best move and the game's actual move - it also makes sense that there's no correlation: Some players play better in aggressive positions, while others might take advantage of opening traps or expect blunders in responses to gambits. Indeed, at the level Tizian was playing at during that time period, his opponents could be expected to not know deep opening theory or fall for simple traps.

Visualizations

It's best to employ the two graphs above that show average opponent Elo and win/draw/loss rate by weekday. These will be enough to demonstrate that weekend players aren't substantially stronger than weekday players; in fact, Tizian's impression is exactly the opposite of the truth.

It would also be helpful to have a visualization that shows Tizian's "expected Elo" by weekday, in order to further demonstrate how his play was superior on certain weekdays compared to others.

Deliverable



[Link to the dashboard on Redash]

The deliverable for this project is above.

I chose the visualizations I did because they convey the information concisely, without any extraneous information. The win/draw/loss rates are required to be able to understand why Tizian thought his play was different on weekends and weekdays; the average opponent Elo line graph shows us the reason for the differing performance. Finally, the expected Elo line graph shows us a "summary" that is essentially a performance graph, revealing on what weekdays Tizian is on top of his game.

The colors for the win/draw/loss graph were chosen appropriately, according to intuition: wins are green, losses are red, and draws would be yellow, but that color was not available on Redash v8, so orange was chosen instead. In Redash v9, a custom color picker will be available, so this can be improved.

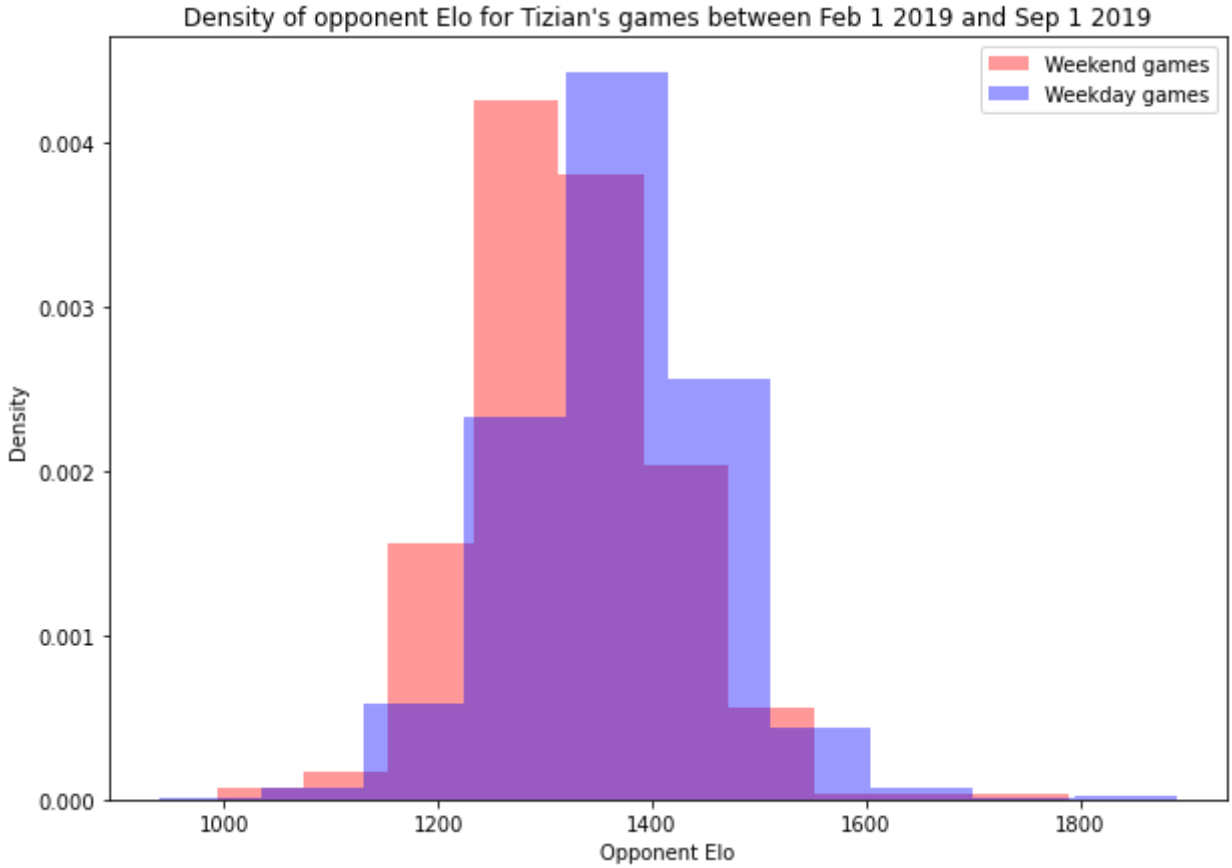
The win/draw/loss graph was chosen as a stacked bar chart because each of the weekdays must add up to 100%, and the most important element - the amount of wins - is easily comparable across the different weekdays. The other two graphs were made line graphs because they show a pattern across the week - the datapoints aren't completely disconnected (so not a scatter plot), and they represent a state, not a "count" (so not a bar chart).

The deliverable is meant to be presented. During the presentation, attention must be brought to the following points:

- Sundays are especially good days for Tizian
- The average opponent Elo is much lower on weekends
- The performance chart is the expected Elo given average opponent Elo and win/draw/loss rates
- The performance chart reveals Thursday is Tizian's actual best day

Examining the opponent Elo distributions

In order to examine the opponent Elo distributions, I ran a Kolmogorov-Smirnov test between them. This analysis was done in Python. The p-value for this analysis was 4e-15, which reveals that indeed they are two very different distributions. The opponent Elos, graphed out, look like this:



It's pretty clear that both distributions are approximately normal with about the same standard deviation, but different means.

Examining the central assumption

In order to examine the central assumption of this analysis, we would have to verify that players play at approximately the level of their approximate Elo. Unfortunately, this involves a lot of data munging; but thankfully, due to Lichess being open-source, it's easily verifiable that players' Elo is adjusted once they win or lose a game, and this dynamic readjustment is scaled accordingly to the players' Elo differences. Due to the very large player pool, it's also unlikely that there are two distinct player pools - one on weekdays and one on weekends - that do not interact at all. There's also anecdotal evidence that there are interactions between these groups in the form of titled players/chess streamers playing during both those time periods. This should eventually lead to Elo stability across the player pool, which validates our assumption.