Uncovering Hidden Patterns in Online Chess Games: A Multi-Faceted Data Science Exploration (Github)

Team member info: Yehuda Frist, Asaf Miron, Alon Rozenstein

Problem Description:

This project aims to explore, model, and understand patterns in online chess games through a combination of statistical analysis, prediction, community detection, clustering, and sequential behavior modeling. We set out to investigate questions such as: What statistical trends govern online chess? Can we predict game outcomes? Do distinct communities or playstyles emerge in the network of games and players? Are there natural clusters of games based on metadata or move dynamics? And how does time control affect player strategy and game evolution?

Data:

The data for this project comes from a publicly available Lichess dataset that contains detailed information on 20,000+ games (Link). Each row represents a single game and includes features such as game ID, number of moves, victory status, winner, time control, player ratings, full move history, and detailed opening information . The dataset is stored in CSV format and is approximately 7.5 MB in size. After cleaning and preprocessing, we extract both numerical and categorical features for analysis. No web scraping or additional data collection is required.

Table of Contents -

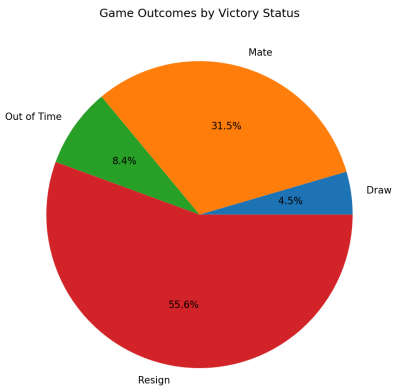
Statistical Analysis
Prediction
Community Detection
Clustering
Sequential Behavior Modeling
Future Work
Conclusion

Statistical Analysis

The initial phase of our project focused on descriptive statistics and data exploration to uncover basic patterns in online chess matches. By summarizing key variables and visualizing distributions, we aimed to provide a solid empirical foundation for prediction, identifying online chess communities, unsupervised clustering, and sequential behavior modeling.

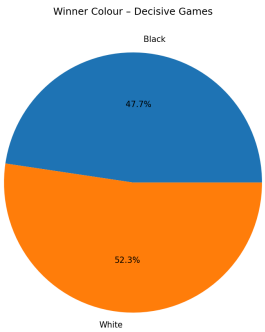
Victory Status Analysis - We analyzed how games ended, categorizing them into resignation, checkmate, timeout, and draw ***(victory_status.png***). The vast majority of games ended in resignation, at approximately 55.5% of matches. Next came checkmate at around 31.5%, followed by timeouts at about 8.5%. Draws were relatively rare, accounting for only 4.5%.

victory_status.png



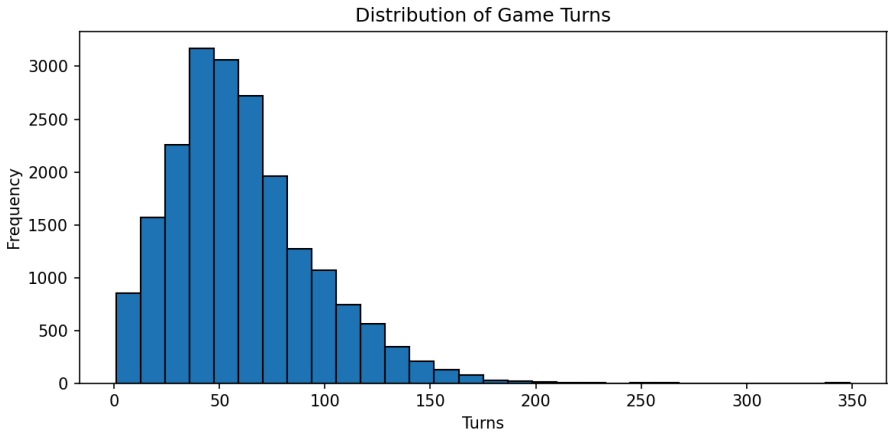
Winner Color Distribution - We assessed the proportion of games won by White versus Black (***winner_colour.png***). The analysis showed that White wins slightly more often (52.3%), which aligns with the common assumption that having the first move provides a modest advantage (It's worth noting that 4.5% of games ended in a draw, so there was no clear winner).

winner_colour.png



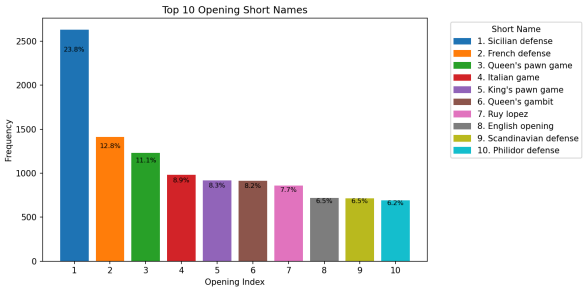
Game Length Analysis - We examined the distribution of game lengths based on the number of turns (***games_length_hist.png***). The histogram shows a long-tailed distribution: the majority of games (50%) fall within the range of 38 to 79 turns, while about 25% end in 37 moves or fewer, and 25% extend to 80 moves or more. Although the 4th quartile captures the longest 25% of games, nearly 75% of those still finish by 119 turns. Only about 6.7% go beyond 140 moves, and a vanishingly small 0.04% exceed 340 moves. This steep drop-off shows that truly long games are extremely rare.

games_length_hist.png

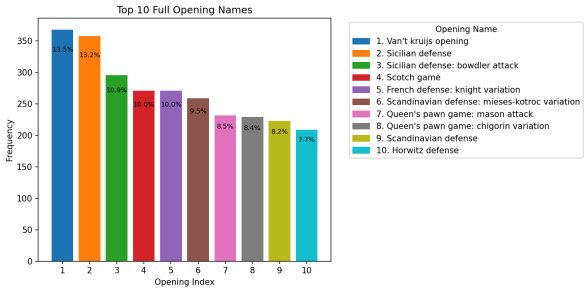


Opening Popularity - We analyzed how frequently different openings occur, using both short and full descriptive names (***opening_shortname_dist.png*** and ***opening_fullname_dist.png***). Among short names, the most common opening is the “Sicilian Defense,” accounting for 23.8% of the top 10 most common openings. It’s followed by the “French Defense” at 12.8% and the “Queen’s Pawn Game” at 11.1%, while the rest of the top 10 each have under 10%. For full opening names, the most common is the “Van’t Kruij” at 13.5% of the top 10, followed by the classic “Sicilian Defense” at 13.2% and its “Bowdler Attack” variation at 10.9%.

opening_shortname_dist.png

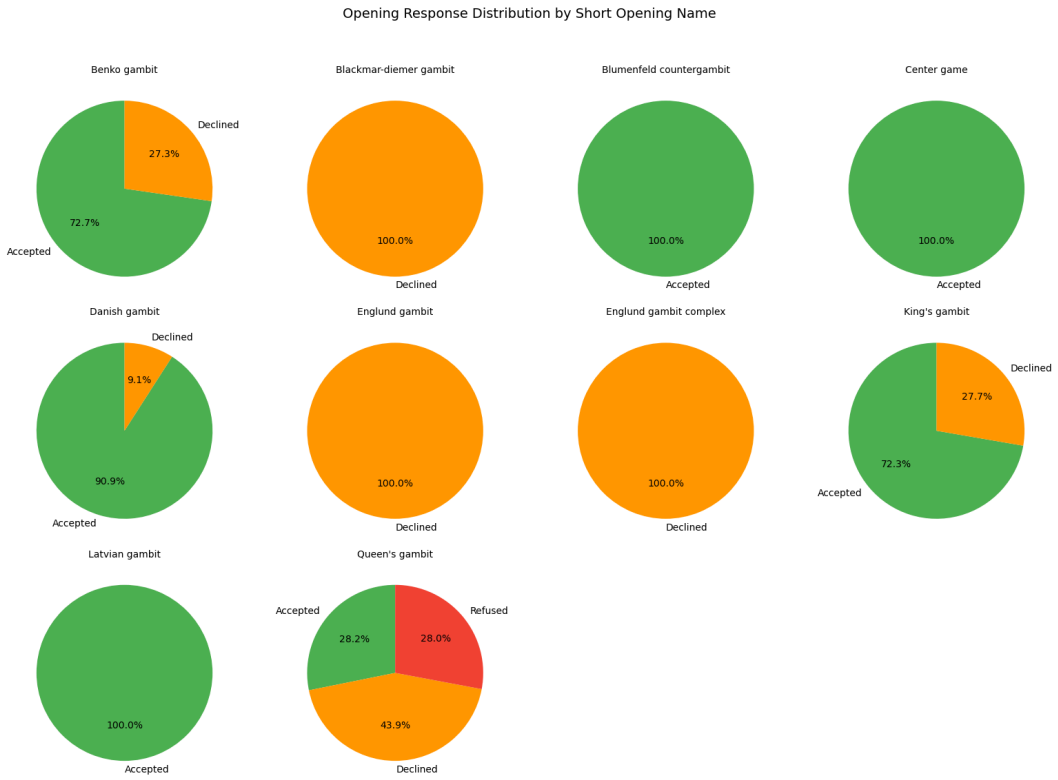


opening_fullname_dist.png



Opening Response Frequency - Not all openings lead to a clear “Accepted,” “Declined,” or “Refused” label, but we collected those that do (based on their short name) and analyzed how players typically respond (***opening_responses.png***). Some openings, like the “Blumenfeld Counter-Gambit” and the “Center Game,” were always accepted. Others, such as the “Englund Gambit” and the “Englund Gambit Complex,” were always declined. The “Benko Gambit” was accepted 72.7% of the time and declined 27.3%. The “Queen’s Gambit” had a more even distribution: 43.9% declined, 28.2% accepted, and 28% refused.

opening_responses.png



Prediction

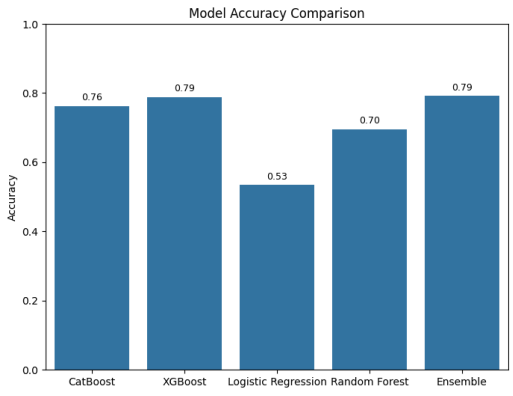
To predict the outcome of chess games (White win, Black win, or Draw), we framed the task as a multi-class classification problem. We trained five supervised machine learning models: **CatBoost**, **XGBoost**, **Random Forest**, **Logistic Regression**, and an **Ensemble Model** that averages the predicted probabilities of the base models using weights determined by Dirichlet sampling. The data was first cleaned and filtered to include only relevant features, such as player ratings, rating difference, opening name, and time control. Categorical features (such as opening name) were encoded appropriately for tree-based models and one-hot encoded for Logistic Regression.

Model performance was assessed using multiple criteria. Accuracy served as a coarse measure of overall correctness, but we supplemented it with precision, recall, and F1-score to account for class imbalances - particularly the rarity of draws. We also computed micro-averaged AUC scores from ROC curves, which quantify discrimination ability across all classes. To better understand error patterns, we included confusion matrices and per-class metrics for selected models.

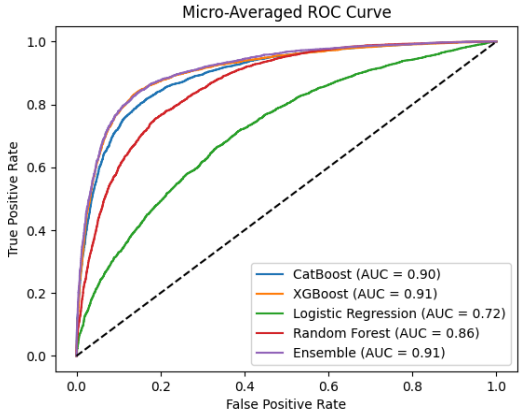
As shown in ***accuracy_comparison.png***, CatBoost, XGBoost, and the ensemble model all achieved strong performance, with accuracies around 0.79. Logistic Regression, by contrast, performed poorly with only 0.53 accuracy, likely due to its inability to capture non-linear interactions in the data. The ensemble model matched XGBoost in accuracy but showed improved stability across all classes.

ROC curve comparisons in ***roc_comparison.png*** further reinforce these conclusions. XGBoost, CatBoost, and the ensemble achieved AUC values of 0.90–0.91, while Logistic Regression lagged behind at 0.72. This indicates that even when Logistic Regression made correct predictions, it did so with less confidence and consistency.

accuracy_comparison.png

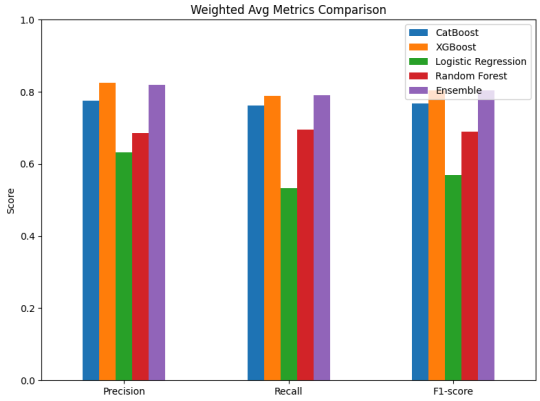


roc_comparison.png



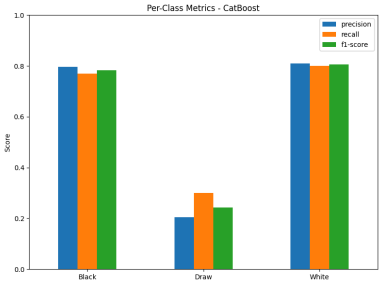
To evaluate class-level behavior, we present macro metrics in ***weighted_metrics_comparison.png***. XGBoost and the ensemble again lead in weighted precision, recall, and F1-score, while Logistic Regression's F1-score dipped below 0.60. The most notable challenge across all models was the draw class.

weighted_metrics_comparison.png

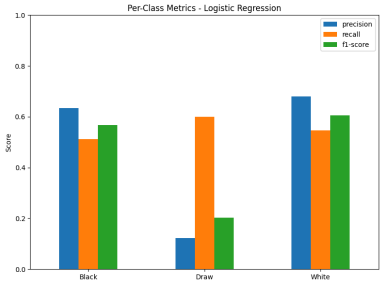


As seen in ***class_metrics_CatBoost.png*** (CatBoost), ***class_metrics_Logistic_Regression.png*** (Logistic Regression), and ***class_metrics_Random_Forest.png*** (Random Forest), the draw class is consistently underrepresented and misclassified. For CatBoost, for instance, the recall for draws barely exceeded 0.3, despite high performance for the win/loss classes.

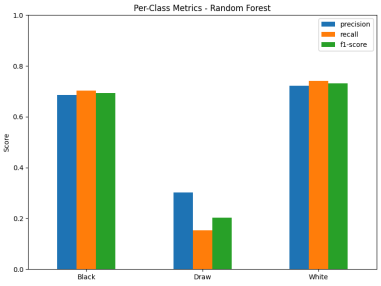
class_metrics_CatBoost.png



class_metrics_Logistic_Regression.png

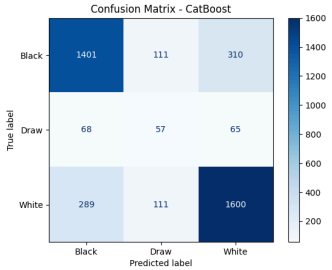


class_metrics_Random_Forest.png

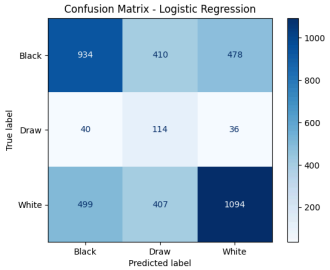


The confusion matrices offer additional insight. ***confusion_matrix_CatBoost.png*** (CatBoost) shows that most misclassifications involve confusing draws with wins or losses, highlighting the class imbalance issue. Logistic Regression's confusion matrix (***confusion_matrix_Logistic_Regression.png***) reveals widespread misclassifications, while Random Forest (***confusion_matrix_Random_Forest.png***) performs reasonably well but still struggles to separate draws from decisive outcomes.

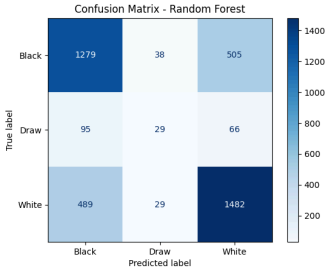
confusion_matrix_CatBoost.png



confusion_matrix_Logistic_Regression.png



confusion_matrix_Random_Forest.png



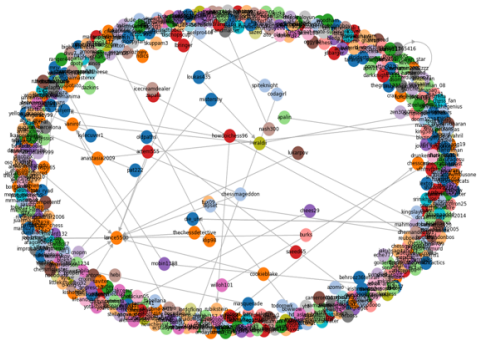
Despite careful tuning, class imbalance remained the primary impediment throughout this phase. While oversampling and class-weight adjustment were considered, they led to negligible improvements and occasional overfitting. The use of probabilistic ensembling helped mitigate overconfidence in specific classes and improved average F1-score, but the draw class remained a persistent weakness. Nonetheless, the model pipeline demonstrated strong predictive ability for the more frequent outcomes, providing a robust foundation for downstream tasks.

Community Detection

In this stage, we use network analysis techniques to examine the structure of the chess player community, identifying key subgroups, rivalries, and the relationship between player skill, influence, and opening strategies.

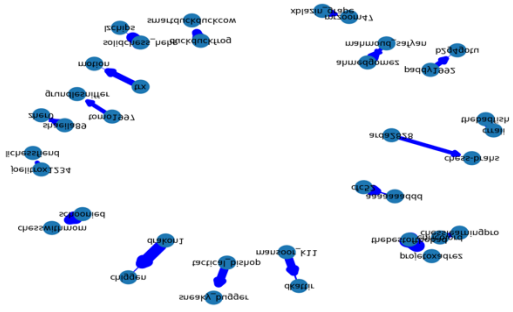
To uncover structural patterns in the chess player ecosystem, we constructed and analyzed a directed network where each node represents a player, and edges point from the loser to the winner in each game. Community detection was performed using label propagation on the largest component, with results visualized in ***top_player_network_communities.png***. The network exhibited clear modularity, with most central players concentrated within well-defined subgroups. Cross-community links were relatively rare, suggesting play is largely confined within stable clusters, possibly reflecting real-world affiliations or platform-based communities.

top_player_network_communities.png



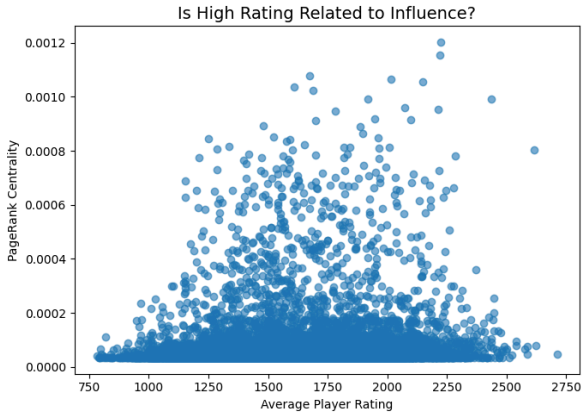
To quantify competitive intensity, we assigned edge weights based on the number of games played between each pair of players (***top_rivalries.png***). By extracting the most frequent rivalry edges, we identified a handful of high-intensity rivalries, highlighting the presence of repeated, competitive pairings at the center of the network. The rivalry structure is dominated by a few players with multiple frequent adversaries, indicating a small but highly active competitive core.

top_rivalries.png

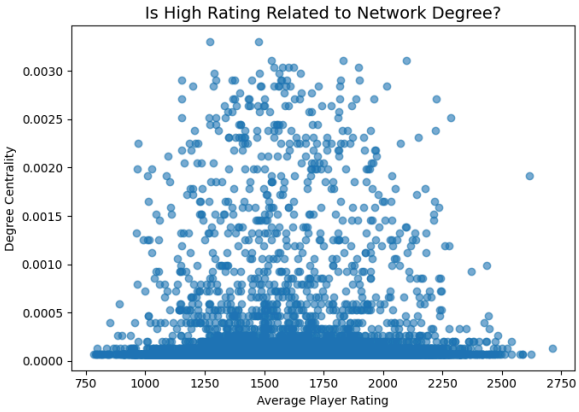


We then assessed the relationship between network centrality and player skill by computing PageRank and degree centrality and comparing these to average rating. The results (***pagerank_vs_rating.png*** and ***degree_centrality_vs_rating.png***) revealed a positive but non-deterministic correlation: most highly rated players were also central in the network, but not all central figures had the highest ratings. This suggests that network influence is driven by both activity and skill.

pagerank_vs_rating.png

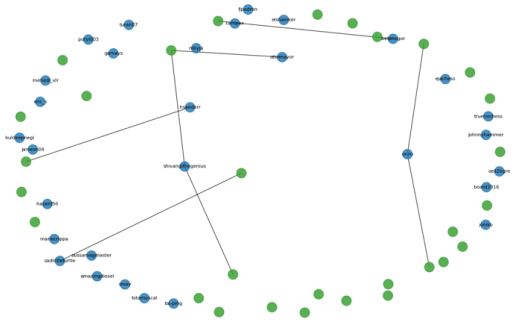


degree_centrality_vs_rating.png



To explore the connection between player communities and opening choice, we constructed a player-opening bipartite network (***player_opening_bipartite.png***). The resulting clusters show that most players share openings with a defined subset of peers, while a few popular openings serve as hubs linking otherwise distant groups. This points to both specialization and convergence in opening preferences across the network.

player_opening_bipartite.png



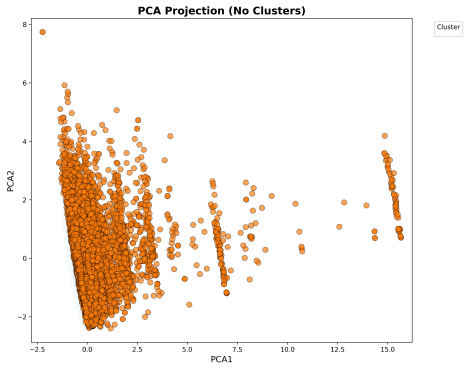
Together, these analyses provide strong evidence for the existence of stable communities, a concentrated rivalry structure, and a moderate relationship between chess skill and network influence. The visualizations illustrate both the global organization of the network and specific behavioral patterns among central players and communities.

Clustering

To uncover underlying patterns in chess gameplay, we applied unsupervised clustering techniques to group similar games based on their metadata. Our objective was to identify distinct “game types” that reflect variations in strategy, pacing, or player dynamics, independent of the game outcome. We focused on two algorithms: **K-Means** and **Agglomerative (Hierarchical) Clustering**, and compared their performance in terms of structure, coherence, and interpretability.

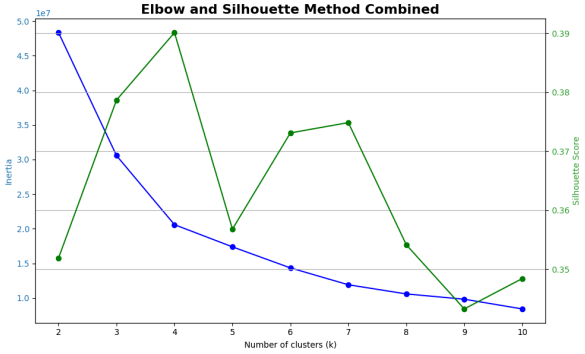
The dataset was first cleaned and transformed into a suitable feature space including time control, player ratings, material changes, and number of moves. Features were standardized, and dimensionality was reduced using **PCA** followed by **t-SNE** to enable effective visualization. As shown in ***pca_preclustering.png***, the pre-clustering PCA projection revealed no clear structure, justifying the need for clustering.

pca_preclustering.png

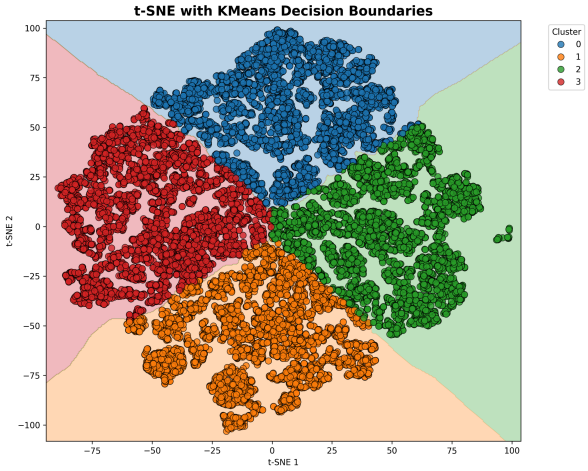


To select the number of clusters, we combined the **elbow method** and **silhouette scores**, displayed in ***combined_k_selection.png***. Both metrics suggested an optimal k=4, which we used for both clustering algorithms. K-Means was first applied to the t-SNE-transformed data, producing four distinct clusters shown in ***kmeans_clusters_annotated.png***, with decision boundaries overlaid in ***tsne_KMeans_boundaries.png***. The clusters were interpretable and captured meaningful game types, such as “Quick Decisive Blitzes” or “Balanced, Well-Developed Games”.

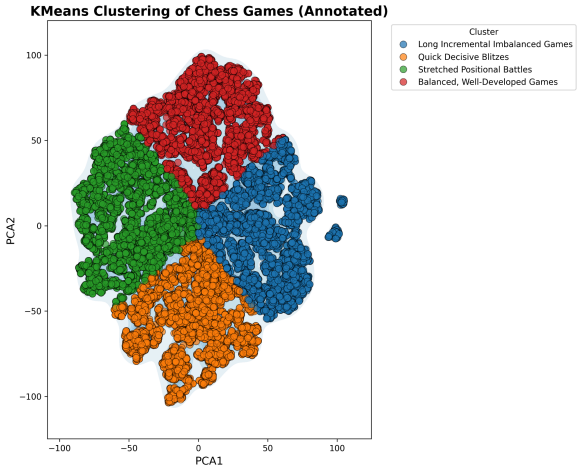
combined_k_selection.png



tsne_KMeans_boundaries.png

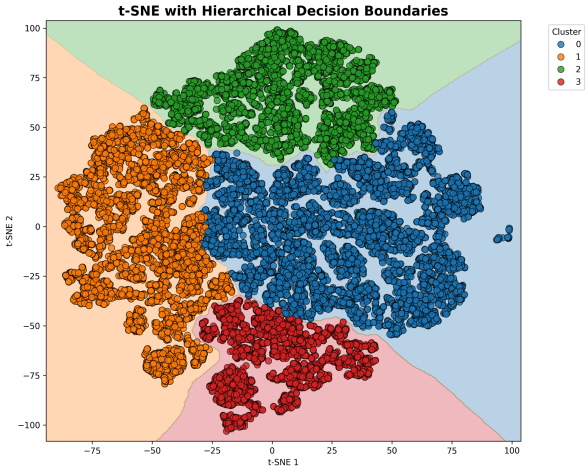


kmeans_clusters_annotated.png

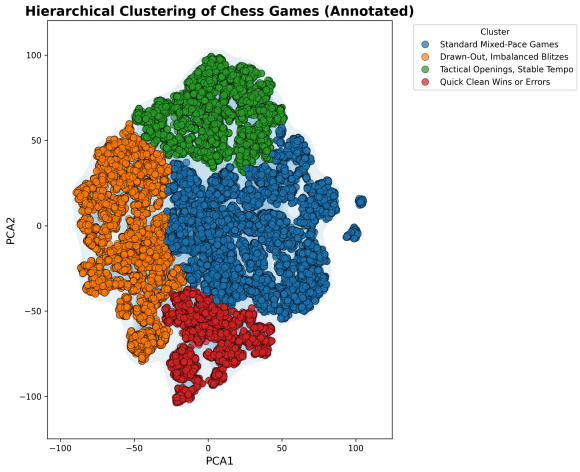


To validate these results, we repeated the process using **Agglomerative Clustering**. The resulting clusters (***tsne_Hierarchical_boundaries.png*** and ***hierarchical_clusters_annotated.png***) showed alternative groupings that remained coherent and interpretable, including “Drawn-Out, Imbalanced Blitzes” and “Tactical Openings with Stable Tempo.” Both methods yielded consistent separability and thematic coherence.

tsne_Hierarchical_boundaries.png

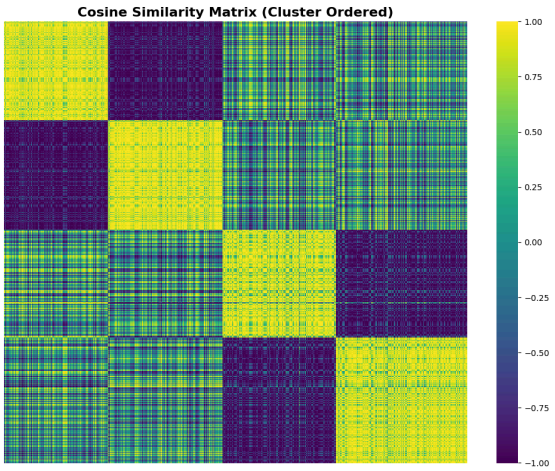


Hierarchical_clusters_annotated.png

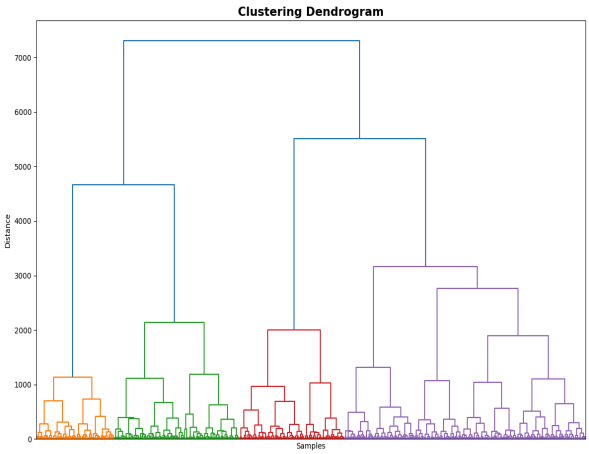


To further assess clustering quality, we computed a **cosine similarity matrix** reordered by cluster assignment (***similarity_matrix_ordered.png***). The clear block-diagonal structure confirmed high intra-cluster similarity and justified our cluster boundaries. A **dendrogram** from the hierarchical clustering (***dendrogram.png***) provided additional insight into the nested structure of the data.

similarity_matrix_ordered.png



dendrogram.png



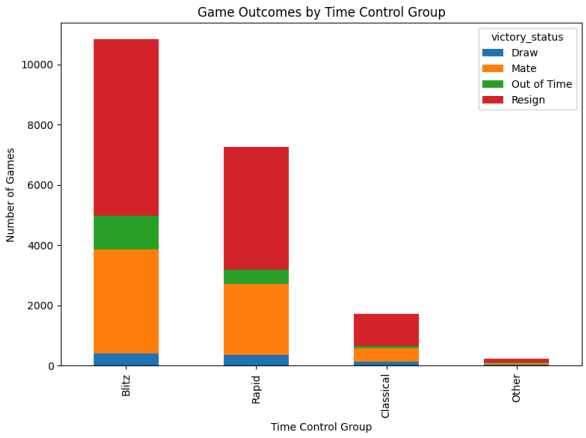
Evaluation combined internal validation (silhouette scores, inertia, similarity matrices) with qualitative inspection of cluster composition. While no ground-truth clustering was available, both algorithms produced clusters aligned with meaningful gameplay patterns. Challenges included the high-dimensional nature of the data and the need to ensure that visual artifacts from t-SNE did not mislead interpretation. Cross-referencing cluster assignments with metadata distributions helped ensure interpretability and robustness.

Sequential Behavior Modeling

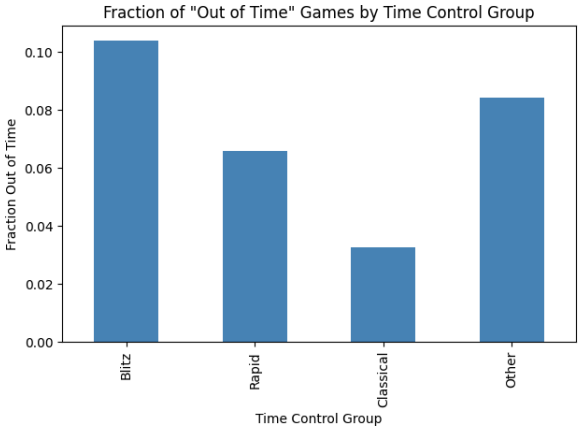
To understand how temporal constraints and early-game patterns influence chess outcomes, we conducted a comprehensive two-part analysis focusing on time control dynamics and move sequence similarity. Our dataset was first grouped by time increment, mapping each game into standard categories (Bullet, Blitz, Rapid, Classical, Other) based on the total allotted time. Outcomes were aggregated by group to assess the relationship between time control and result type (Mate, Resign, Draw, Out of Time).

The analysis began with a series of visualizations. Grouped outcome rates (***time_control_grouped_outcomes.png***) revealed that shorter controls (Bullet, Blitz) are strongly associated with a higher fraction of “Out of Time” results, while longer formats like Rapid and Classical are more likely to conclude via mate or resignation. This pattern is reinforced in (***out_of_time_fraction_grouped.png***), where the fraction of games ending on time is highest in fast controls.

time_control_grouped_outcomes.png

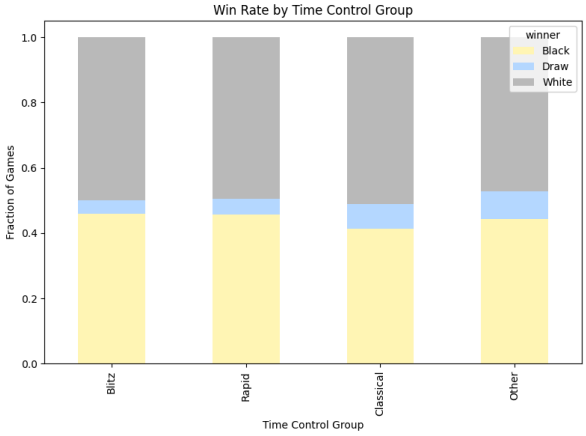


out_of_time_fraction_grouped.png

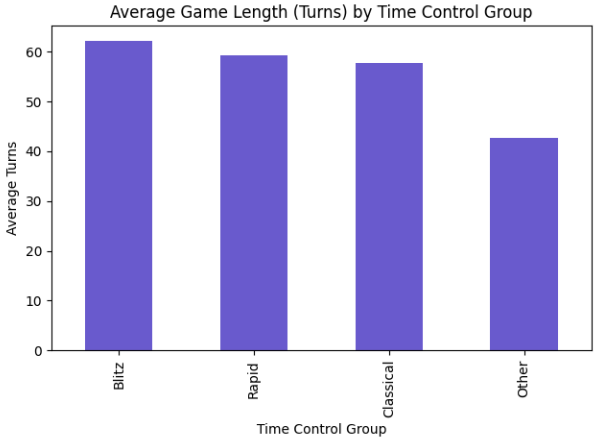


Win rate analysis (***win_rate_by_time_control_group.png***) demonstrated that the overall balance between White, Black, and Draw outcomes is generally stable across groups, though draws are modestly more frequent in slower games, which makes sense, as more time usually leads to better decision making and less mistakes. Average game length, summarized in (***avg_game_length_by_time_control_group.png***), decreases with time control, with Classical games typically having the least amount of moves, however by a short margin.

win_rate_by_time_control_group.png

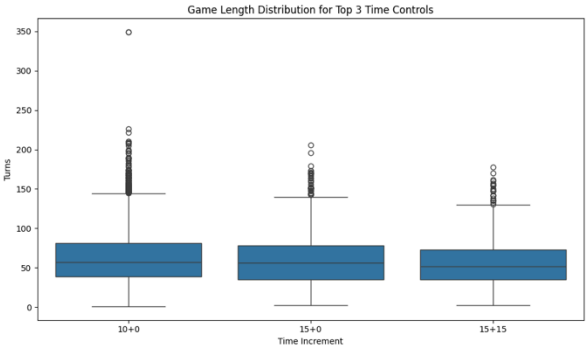


avg_game_length_by_time_control_group.png

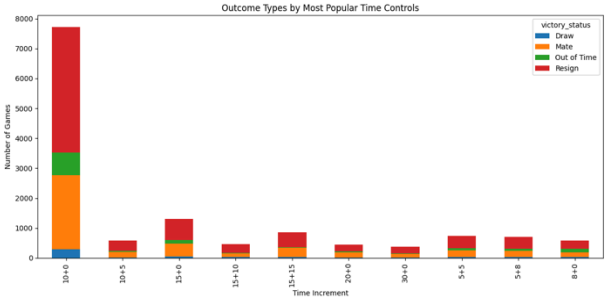


To further characterize time control effects, we compared the most popular time increments. As shown in (***game_length_distribution_top3.png***), game length distributions for the top three controls are broadly similar, with the majority of games finishing in 50–70 turns but some extreme outliers. Stacked bar charts of outcome types by specific increments (***outcome_types_top10_timecontrols.png***) confirm that “Out of Time” is a common result only in the fastest formats.

game_length_distribution_top3.png

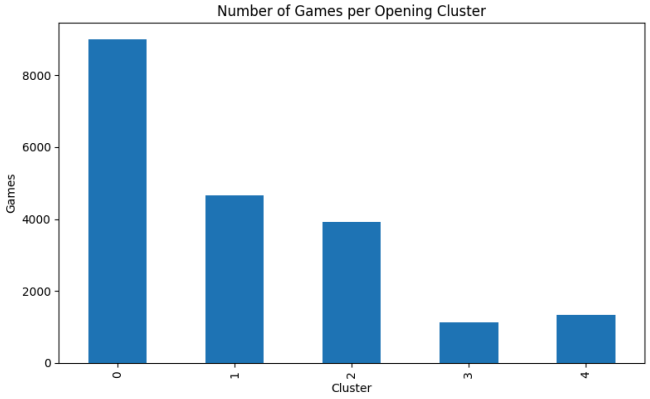


outcome_types_top10_timecontrols.png



To complement the statistical overview, we analyzed move sequence similarity using TF-IDF vectorization on the first eight plies of each game, followed by dimensionality reduction and MiniBatch KMeans clustering. The resulting clusters (***opening_cluster_counts.png***) highlight the prevalence of several dominant opening patterns. These clusters underscore the concentration of chess play around a handful of popular opening lines.

opening_cluster_counts.png



Taken together, this stage demonstrates that time controls not only impact the mechanism by which games conclude but also shape both game length and opening repertoire. Fast controls increase the likelihood of time-based outcomes and favor a narrower set of openings, while slower formats encourage more “classical” chess. These findings illuminate how temporal structure molds competitive play at every level.

Future Work

Future work could focus on improving draw prediction using advanced sampling or cost-sensitive methods. Incorporating move-level data into clustering and community detection may reveal deeper strategic patterns. Additionally, analyzing how players and communities evolve over time would offer a dynamic perspective on rivalries and skill progression.

Conclusion

This project demonstrated the power of data science methods in uncovering hidden structure within online chess games. Through descriptive statistics, predictive modeling, network analysis, clustering, and sequence-based exploration, we revealed consistent patterns in player behavior, game dynamics, and strategic diversity. The results highlight how computational tools can enrich our understanding of complex domains like chess, offering both theoretical insights and practical foundations for future applications.