DM Report

Di Viesti Sofia, Perodi Martina
University of Pisa

Contents

1 Introduction **2**

2 Data Understanding and Preparation **2**
2.1 Overall features semantics . . . 2
2.2 Data Quality Assessment . . . 2
2.2.1 Duplicate Data, Missing Values and Noise . . . 2
2.2.2 Semantic Inconsistencies . . . 3
2.2.3 Zeros and Negative Values . . . 3
2.2.4 Correlation Analysis . . . 3
2.2.5 Outlier Detection, Skewness Analysis, and Transformations . . . 3
2.3 Features Engineering . . . 4

3 Clustering **6**
3.1 Center-based methods . . . 6
3.1.1 K-Means . . . 6
3.2 Density-based clustering . . . 8
3.2.1 DBSCAN . . . 8
3.2.2 OPTICS . . . 9
3.2.3 HDBSCAN . . . 10
3.3 Hierarchical clustering . . . 12
3.4 Discussion . . . 12
3.5 Cluster Profiling and Archetype Interpretation . . . 13

4 Classification **14**
4.1 KNN . . . 14
4.2 Naive Bayes . . . 15
4.3 Decision Tree . . . 15
4.4 Discussion . . . 17

5 Regression **18**
5.1 Simple linear regression . . . 18
5.2 Multiple linear regression . . . 18
5.3 Non-linear regression . . . 18
5.4 Multi-output regression . . . 18

6 Pattern mining **19**

7 Conclusion **21**

1 Introduction

This study utilizes a dataset retrieved from BoardGameGeek (BGG) in February 2021, comprising approximately 20,000 ranked board games. To ensure statistical reliability and mitigate noise from insufficient data, the analysis exclusively considers titles with at least 30 user votes, consistent with BGG's official ranking methodology.

The research aims to support strategic decision-making in game design and publishing by addressing the following question:

> *"Which characteristics and configurations of a board game maximize its success and popularity within the gaming community?"*

By identifying correlations between technical attributes (e.g., mechanics, complexity, and duration) and performance metrics, this report seeks to define an evidence-based profile to guide designers in reducing market uncertainty.

2 Data Understanding and Preparation

This section first introduces the semantic meaning and main characteristics of the features. It then presents a data quality assessment, addressing missing values, outliers, semantic inconsistencies, and feature correlations. Finally, we briefly discuss feature engineering strategies adopted to enrich the dataset.

2.1 Overall features semantics

In *Table 1*, we provide an overview of all features, specifying their types and a brief description for each.

Attribute	Type	Description
BGGId	Integer	Unique identifier from BoardGameGeek
YearPublished	Integer	Year of publication
MinPlayers	Integer	Minimum number of players
MaxPlayers	Integer	Maximum number of players
ComAgeRec	Integer	Suggested age by the community
BestPlayers	Integer	Optimal number of players (community)
NumOwned	Integer	Number of users who own the game
NumWant	Integer	Number of users interested in the game
NumWish	Integer	Number of users wishlisting the game
NumWeightVotes	Integer	Number of votes for game complexity
MfgPlaytime	Integer	Playtime in minutes stated by the publisher
ComMinPlaytime	Integer	Minimum playtime estimated by the community
ComMaxPlaytime	Integer	Maximum playtime estimated by the community
MfgAgeRec	Integer	Suggested age by the publisher
NumUserRatings	Integer	Number of user ratings
NumComments	Integer	Number of user comments
NumAlternates	Integer	Number of alternative versions
NumExpansions	Integer	Number of expansions
NumImplementations	Integer	Number of implementations
IsReimplementation	Binary	Indicates whether the game is a reimplementation
Kickstarted	Binary	Whether the game was funded via Kickstarter
Cat:...	Binary	Category membership flag
Rank:...	Ordinal	Ranking position across game types
Rating	Categorical	Game rating category (low, medium, high)
Name	Categorical	Name of the game
Description	Categorical	Textual description of the game
ImagePath	Categorical	Path to the image file
Family	Categorical	Game family or group
GoodPlayers	Categorical	Typical acceptable player range (e.g., 2–4)
GameWeight	Float	Game complexity provided by the publisher
ComWeight	Float	Game complexity provided by the community
LanguageEase	Float	Degree to which gameplay depends on language

Table 1: Description and type of features in the board game dataset

2.2 Data Quality Assessment

In the following, we describe the procedure adopted to address potential issues with the dataset provided.

2.2.1 Duplicate Data, Missing Values and Noise

No duplicate rows were found, confirming that each record represents a unique board game entry. Regarding missing values, we note that the *Description* attribute contained a single missing entry, which was manually retrieved. The attributes *ComAgeRec* and *LanguageEase* presented missing values that were imputed using the

median. All entries of the *NumComments* attribute were missing; therefore, the attribute was removed from the dataset. The *Family* attribute exhibited a very high proportion of missing values (69.61%) and was therefore removed from the dataset. Although *ImagePath* contained only a small number of missing values, it was also dropped, as it was not relevant for the objectives of this study.

2.2.2 Semantic Inconsistencies

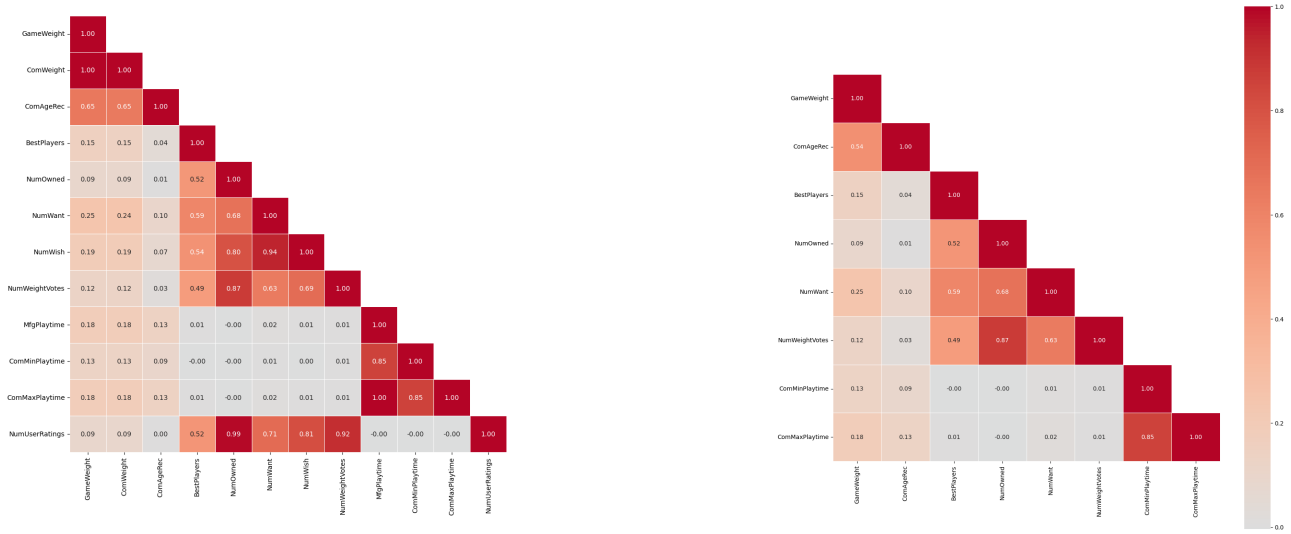
Several checks were performed to verify the semantic coherence of the data. In particular, inconsistencies were found between minimum and maximum values for both the number of players and the playtime. When *MinPlayers* exceeded *MaxPlayers*, the maximum number of players was replaced with its median value. Similarly, cases where *ComMinPlaytime* was greater than *ComMaxPlaytime* were resolved by adjusting the inconsistent value using the corresponding median.

2.2.3 Zeros and Negative Values

Initially, zero values were retained within the dataset, as they were hypothesized to convey potentially meaningful information, such as a complete absence of user votes or community interactions. However, subsequent analysis revealed that these observations lacked significant explanatory power and primarily represented noise. Consequently, a winsorization approach was adopted to handle these outliers, as detailed in *Section 2.2.5*. Negative values were observed exclusively in the *YearPublished* attribute, where they correctly represent years before Christ.

2.2.4 Correlation Analysis

To further refine the dataset, a correlation matrix was computed for the numerical attributes. In order to reduce redundancy, attributes exhibiting a Pearson correlation coefficient greater than 0.90 were removed. Specifically, the following variables were dropped: *ComWeight*, *NumWish*, *NumUserRatings*, and *MfgPlaytime*, as they present high correlation with *GameWeight*, *NumWant*, *NumOwned* and *NumWeightVotes*, and *ComMaxPlaytime*, respectively. The remaining attributes preserve the majority of the information content while improving the robustness and interpretability of subsequent analyses.



(a) Matrix before dropping values

(b) Matrix after dropping values

Figure 1: Correlation Matrices

2.2.5 Outlier Detection, Skewness Analysis, and Transformations

An in-depth analysis of numerical variables was conducted to assess the presence of outliers, distributional asymmetry, and the need for transformations prior to modeling.

As a preliminary step, the skewness of each numerical attribute was evaluated to determine the necessity of non-linear transformations. Following standard exploratory data analysis practices, we initially attempted to mitigate asymmetry by applying square root transformations for moderately skewed variables ($0.5 \leq |\text{skew}| < 1$)

and `log1p` or inversion techniques for highly skewed distributions ($|\text{skew}| \geq 1$). However, these standard approaches generally failed to yield significant improvements in terms of distribution normality. Consequently, the initial transformation strategy was discarded in favor of a more robust treatment for the majority of the dataset. The *LanguageEase* attribute represented the sole exception: as it exhibited a strong and persistent positive skewness, a logarithmic transformation was successfully applied to stabilize its variance and mitigate distributional asymmetry. For all other variables, an alternative methodology was adopted, as detailed in the following.

Following the skewness analysis, an explicit outlier treatment was performed. Exploratory analysis supported by boxplot visualizations revealed the presence of extreme values in several attributes, most notably *MaxPlayers*, *ComMinPlaytime*, and *ComMaxPlaytime*.

A detailed inspection of the boxplots highlighted three distinct patterns among extreme observations. First, a non-negligible proportion of upper-bound values appeared clustered beyond the upper whiskers, suggesting systematic behavior rather than random noise and therefore carrying meaningful information. Second, a limited number of isolated observations exhibited characteristics typical of extreme outliers, which can be interpreted as "outliers of outliers." Third, zero values were present in some distributions.

Retaining the above mentioned values unchanged or replacing them with the median would have excessively distorted the distributions. Hence, to address these issues while preserving the informative structure of the data, a Winsorization strategy based on percentiles was adopted. Specifically, values below the 5th percentile and above the 95th percentile were capped at the corresponding thresholds. Zero values were shifted toward the lower bound by replacing them with the 5th percentile, thus limiting their influence while maintaining distributional coherence.

Although Winsorization is often defined using thresholds derived from the interquartile range (e.g., median $\pm z\times$ IQR), a percentile-based approach was preferred in this context. After evaluating multiple percentile configurations, the 5th–95th percentile range emerged as the most effective compromise between mitigating the impact of true extreme outliers and retaining upper-tail observations that convey meaningful information for the analysis.

The boxplots in *Fig 2* visually summarize the results of the described transformations.

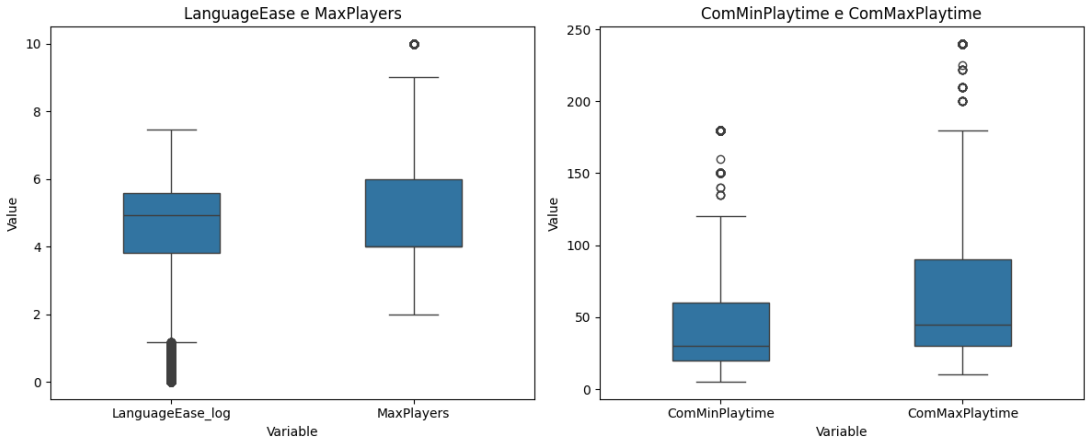


Figure 2: Boxplots of *LanguageEase_log, MaxPlayers, ComMinPlaytime* and *ComMaxPlaytime* after described transformations.

2.3 Features Engineering

A preliminary analysis of the dataset revealed significant limitations regarding the original ranking features, such as *Rank:strategygames*, *Rank:familygames*, and *Rank:abstract*. These variables, intended to represent the standing of a board game within specific categories, suffer from extreme **sparsity**.

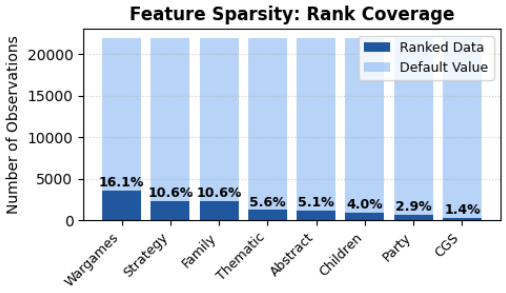


Figure 3: Sparsity analysis of ranking features.

Data inspection reveals that most records default to $21,926$ ($N+1$), where N is the total number of records of the dataset, acting as a placeholder for unranked entries. As shown in Figure 3, the distribution of valid ranks is remarkably sparse; for instance, *family games* covers only 10.6% of the dataset. Given this sparsity and the nearly null intersection between categories, these features are too fragmented for robust predictive modeling in their raw state.

To overcome the limitations of the original features and provide a unified measure of a game's prestige and market presence, two synthetic variables were engineered:

RankCluster A K-Means clustering algorithm (with $k = 5$) was applied to a subset of engagement and complexity features, reaching approximately analogous performance to the K-Means results discussed in Section 3.1.1. Specifically, we considered *NumOwned*, *NumWant*, *GameWeight*, and *PlaytimeRange* as reference features. This approach categorizes games into five distinct tiers, effectively creating a discrete rank based on the intrinsic distribution of the data. This allows for the identification of "high-tier" versus "low-tier" games, regardless of their specific category.

RankScore A scoring measure was developed to capture the popularity and complexity profile of each entry. The score is calculated using the following weighted linear combination:

$$\begin{aligned} Score =& (0.4 \cdot NumOwned) + (0.2 \cdot NumWant) + \\ & (0.2 \cdot NumWeightVotes) + (0.1 \cdot Weight) - (0.1 \cdot PlaytimeRange) \end{aligned}$$

The weighting scheme prioritizes **user ownership** (40%) and **desirability** (20%) as primary proxies for game success. Community engagement, captured by the number of weighted votes, is also incorporated into the model. A small negative weight is assigned to the *PlaytimeRange* feature to penalize excessive variability in expected session length, thereby favoring titles associated with more consistent play experiences. These weights were determined through our evaluation of the factors we deemed most relevant for a game ranking system. Given the exploratory nature of our project, this approach was considered sufficient for capturing the relative importance of the selected features.

RangePlayers Regarding player availability, the variable *GoodPlayers* initially presented a mix of specific counts and recommended ranges, leading to high granularity and noise. To standardize this information, we introduced *RangePlayers*, calculated as the difference between *MaxPlayers* and *MinPlayers*.

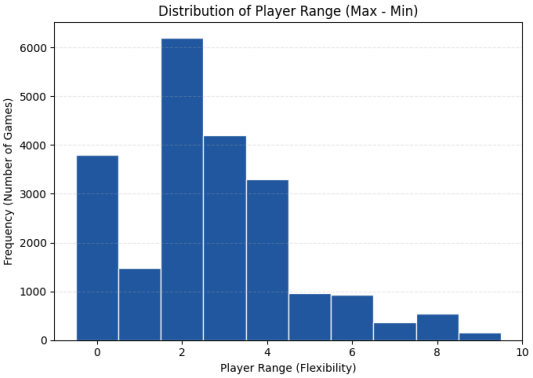


Figure 4: Distribution of game flexibility (*RangePlayers*).

This synthetic feature quantifies the game's flexibility; a higher range suggests a highly scalable title, whereas a range of zero indicates a game strictly designed for a fixed number of participants. This transformation converts a complex categorical attribute into a continuous numerical feature suitable for correlation analysis and clustering. The *RangePlayers* variable ($Max - Min$) was engineered to standardize the heterogeneous community recommendations from the *GoodPlayers* feature.

As illustrated in Figure 4, the distribution is heavily skewed toward low values, indicating that most titles are designed for rigid or specific group sizes. The presence of a long tail highlights a smaller subset of highly versatile games, such as party games. This transformation successfully reduces categorical noise into a continuous 'flexibility' measure, ideal for quantitative analysis.

To capture the complex dynamics of the board game dataset and to answer the research question, several other synthetic features were engineered. These variables are categorized as follows:

Player & Playtime Dynamics

- `IsSoloPlayable` / `IsPartySize`: Binary flag for solo ($Min = 1$) games.
- `IsPartySize`: Binary flag for large-scale ($Max \geq 8$) games.
- `PlaytimeRange`: ($ComMaxPlaytime - ComMinPlaytime$) quantifies session consistency.

Popularity & Market Demand

- `DemandRatio`: ($NumWant/(NumOwned + 1)$) highlights trending potential.
- `OwnershipRatio`: ($NumOwned/(NumOwned + NumWant + 1)$) measures market penetration.

Categorical & Ranking Synthesis

- `NumCategories`: Sum of active thematic tags as a proxy for game complexity.
- `BestRank`: Minimum value across available sub-rankings to identify niche peak performance.

3 Clustering

To evaluate the impact of different variables on cluster formation, several feature subsets were defined for the experimental phase. Table 2 provides a comprehensive legend of these subsets, assigned with unique identifiers (FS1–FS5) and nicknames to facilitate their reference throughout the remainder of this report.

For the feature subsets FS1 to FS4, we selected structural game-related variables, creating variants that include either minimum values (*MinPlayers*, *ComMinPlaytime*) or maximum values (*MaxPlayers*, *ComMaxPlaytime*) in the cases of FS1 and FS2. The *Rating* variable was subsequently introduced in FS3 and FS4. Finally, synthetic features were added by selecting them through a *SelectKBest* procedure using *Rating* as the target variable, and then incorporating the selected features into the corresponding subsets.

ID	Nickname	Feature Set
FS1	Max No Rate	MaxPlayers, ComMaxPlaytime, GameWeight, ComAgeRec
FS2	Min No Rate	MinPlayers, GameWeight, ComMinPlaytime, ComAgeRec
FS3	Min with Rate	MinPlayers, GameWeight, ComMinPlaytime, ComAgeRec, Rating
FS4	Max with Rate	MaxPlayers, GameWeight, ComMaxPlaytime, ComAgeRec, Rating
FS5	Base + Synthetic	GameWeight, ComAgeRec, BestPlayers, NumWant, ComMaxPlaytime, PlaytimeRange, OwnershipRatio, Rating

Table 2: Legend of the Feature Sets used in clustering experiments

3.1 Center-based methods

For center-based clustering methods, we evaluated the standard K-Means clustering algorithm.

3.1.1 K-Means

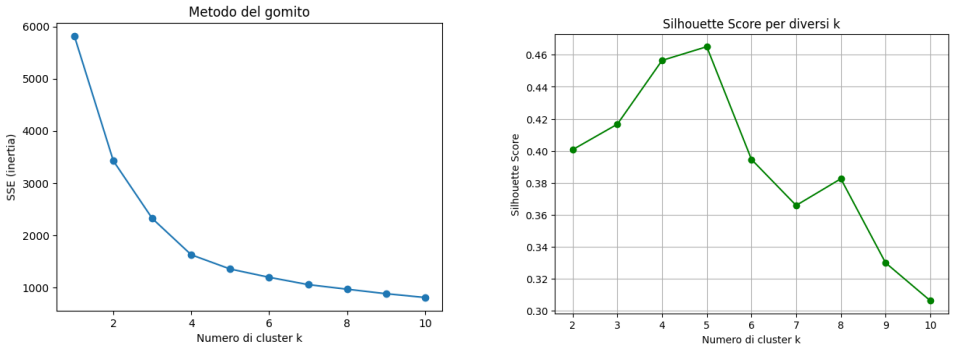
We applied standard K-Means clustering algorithm on our dataset exploring different feature configurations to identify the most meaningful grouping of games. Before applying the algorithm, a *MinMaxScaling* was applied. The same approach was maintained for all clustering approaches described in this report. For each feature set, the number of clusters was determined using both the Elbow Method and the Silhouette Score, employing 10 different centroids initializations, testing values on k ranging from 2 to 10 and selecting the best resulting one for each run.

The summary of all experiments is reported in Table 3.

Feature Set	k	Silhouette	SSE
FS1	5	0.317	1286.765
FS2	3	0.350	**941.045**
FS3	5	**0.465**	1353.411
FS4	6	0.361	2156.968
FS5	5	0.440	1645.574

Table 3: Summary of K-Means clustering experiments

Overall, FS3 (Min with Rate) produced the highest **Silhouette Score (0.465)** and the third best **SSE (1353.411)** at $\mathbf{k = 5}$ (as shown in Figure 5), reaching the best balance in terms of both measures, indicating that incorporating user ratings alongside gameplay characteristics enhances cluster separability.



(a) Elbow method (b) Silhouette method

Figure 5: Determining K using Elbow and Silhouette methods

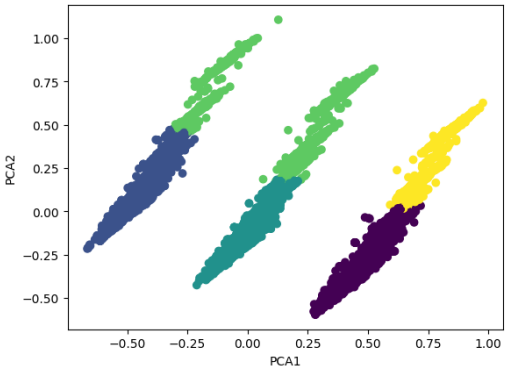


Figure 6: PCA projection of the K-means clustering results obtained using FS3. Same colors denote same clusters.

These results suggest that FS3 provides the most effective and meaningful partitioning of the dataset (Figure 6), capturing relevant variability in both game mechanics and perceived user ratings.
The distribution of cluster sizes in the best feature set is reasonably balanced, ensuring that no single cluster dominates. Specifically, **Cluster 0** contains 3,756 games (17.1%), **Cluster 1** contains 6,628 games (30.2%), **Cluster 2** contains 7,931 games (36.2%), **Cluster 3** contains 2,330 games (10.6%), and **Cluster 4** contains 1,280 games (5.8%).

This distribution supports the interpretability of the results and indicates that FS3 effectively captures meaningful groupings of games based on user ratings and gameplay characteristics.

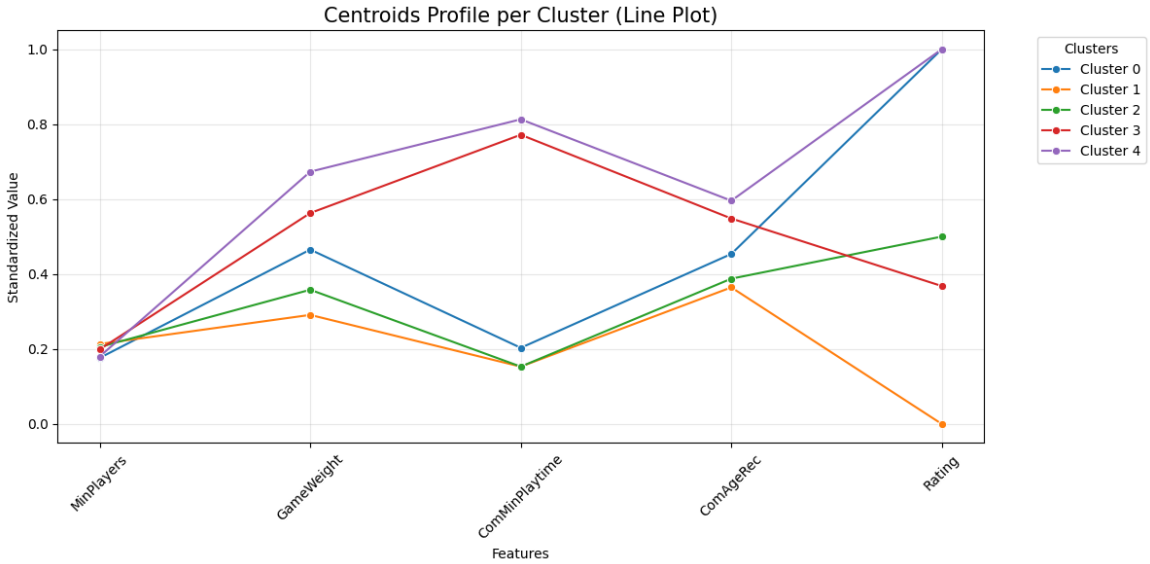


Figure 7: Cluster Profile Analysis via Centroid Coordinates

We complement the K-Means analysis with a discussion of centroid values. Figure 7 shows a line plot in which each line represents a cluster centroid and the dots indicate the corresponding feature values. The figure clearly highlights the distinction between **Cluster 4** and **Cluster 3**, which are characterized by high *ComMinPlaytime* and *GameWeight*, versus **Clusters 0**, **1**, and **2**, which exhibit the opposite pattern. All cluster centroids look close in terms of *MinPlayers* and *ComAgeRec*. Additionally, there is a clear difference in *Rating* between **Cluster 1**, representing games with low *Rating*, and **Clusters 0** and **2**, which have high *Rating*.

3.2 Density-based clustering

As density-based approaches, we focused on DBSCAN and its refinements, namely OPTICS and HDBSCAN.

3.2.1 DBSCAN

For each feature subset, DBSCAN was applied following a systematic procedure. The k-th distance plot was used to identify a plausible range for ε. The parameter space was explored as follows:

- $\varepsilon \in [0.05, 0.20]$, step 0.01;
- MinPts (`min_samples`) $\in \{4, 8, 16, 32, 64\}$.

DBSCAN was executed on MinMax-scaled data for all parameter combinations. For each run, we recorded the number of clusters (excluding noise), the number of noise points, and the silhouette score, computed only when at least two clusters were present. This procedure allowed comparison across runs and identification of configurations yielding stable and well-separated clusters. Results indicate a trade-off between cluster granularity and quality: loose constraints tend to produce a single dominant cluster with artificially high silhouette values, whereas restrictive parameters generate many micro-clusters with limited separation. For $\varepsilon \geq 0.20$, clustering often degenerates into a near-monocluster with high but misleading silhouette scores.

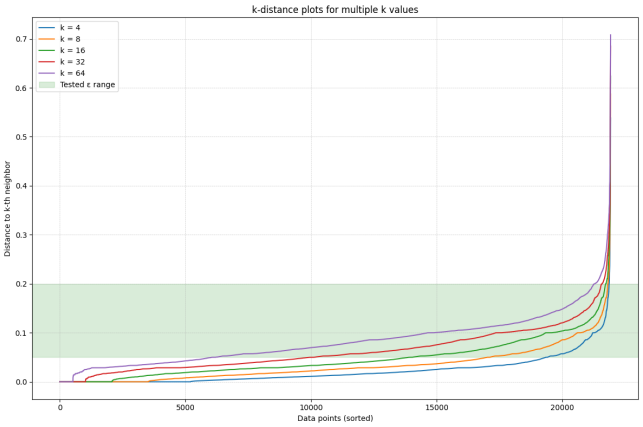


Figure 8: DBSCAN: k-th distance plot for multiple k values. Green area indicates the parameter space explored.

Including the *Rating* feature improves cluster coherence and reduces sensitivity to parameter choice. Among the tested subsets, FS3 represents again the best compromise between cluster quality and segmentation richness, with a **Silhoutte Score of 0.261**. For the sake of completeness, we report in Table 4 the worst and best Silhouette Score values observed for different explored ranges of ϵ w.r.t. different `min_samples` values.

min_samples	**ε range**	**Silhouette range**
64	0.09 – 0.20	0.174 – 0.307
32	0.06 – 0.20	0.159 – 0.312
16	0.05 – 0.20	0.114 – 0.252
8	0.11 – 0.23	0.139 – 0.299
4	0.12 – 0.16	0.129 – 0.234

Table 4: Ranges of ε and corresponding silhouette scores for each value of *min_samples*, reporting the minimum and maximum values observed across all feature sets.

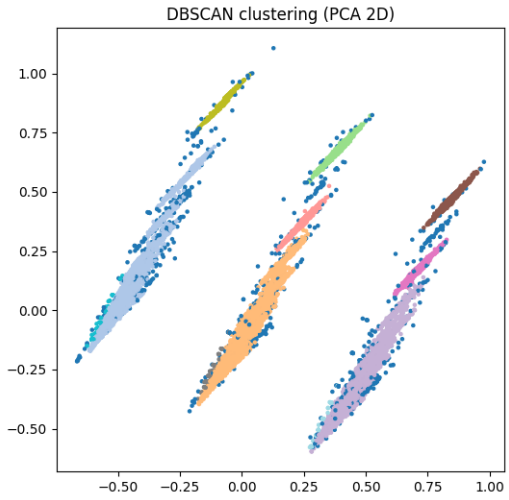


Figure 9: Visualization with PCA of best result. Colors indicate different clusters, while blue points indicate noise points.

Intermediate values of **min_samples** yield multiple meaningful clusters with a limited proportion of noise. The final configuration selected for FS3 is **min_samples = 32**, $\varepsilon = 0.11$ and **11 clusters** as result, with the following distribution, in descending order of size: Cluster 1 (36.31%), Cluster 0 (29.97%), Cluster 4 (16.53%), Cluster 2 (3.29%), Cluster 5 (2.87%), Cluster 3 (2.38%), Cluster 6 (1.85%), Cluster 8 (0.99%), Cluster 9 (0.78%), Cluster 7 (0.73%), and Cluster 10 (0.40%), while noise accounts for only 3.91% of the total population, representing a balanced trade-off between cluster separation and size distribution.

3.2.2 OPTICS

The clustering experiments using the OPTICS algorithm were conducted following a systematic multi-parameter exploration. For each feature subset, the following parameters were investigated:

- **min_samples:** [4, 8, 16]. Controls the minimum density required to form a cluster and explores clusters of varying granularity.
- **xi:** [0.03, 0.05, 0.07]. Determines the minimum steepness of the reachability plot to identify significant cluster boundaries.
- **eps:** [0.07, 0.10, 0.14]. Applied in a DBSCAN-like extraction on the reachability plot to obtain a fixed radius clustering.
- **min_cluster_size:** 5% of the dataset (or an absolute number). The minimum number of points to consider a cluster significant.

For each combination of `min_samples` and `xi`, the OPTICS model was fitted on the MinMax-scaled data. The following measures were computed: number of clusters detected (excluding noise), percentage of points labeled as noise, silhouette score, calculated only for non-noise points when at least two clusters were present, average reachability distance.

In addition, a DBSCAN-like cluster extraction was performed for each `eps` value on the reachability plot, and the same measures were recorded. Reachability plots were generated for each configuration to visually assess the cluster structure. This procedure allowed the comparison of different parameter combinations and the identification of stable and well-separated clusters. The systematic exploration highlights the trade-off between cluster granularity, noise levels, and silhouette quality, providing a basis to select the most informative configurations for further analysis.

A composite score was computed to rank OPTICS configurations, combining normalized silhouette, noise fraction, mean reachability, and deviation from an ideal number of clusters. The weighted score balances cluster quality, noise reduction, and structural coherence, supporting the selection of the most stable and interpretable configuration.

The analysis of multiple feature subsets using OPTICS revealed consistent patterns regarding cluster quality and noise presence. Low-dimensional subsets (FS1 and FS2) generally produced clusters with low Silhouette scores, occasional negative values, or a high proportion of noise, indicating limited separability and unstable structures. Including the *Rating* feature, as in FS3, improved cluster coherence, yielding **Silhouette scores up to 0.292**, though ϵ-based extractions reduced cohesion, highlighting sensitivity to global thresholds. Although

FS5 produced a higher Silhouette score in the pure OPTICS solution, in our experiments, the corresponding reachability plot shown only three main clusters, with poorly defined valleys and weak separation between them, indicating that the clustering captures less meaningful structure despite the high Silhouette.

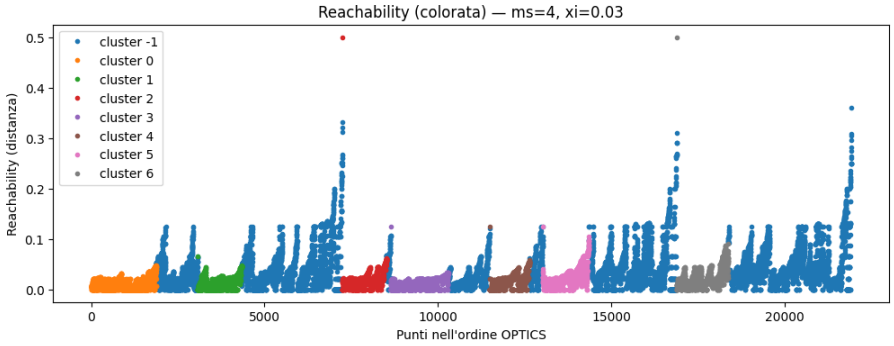


Figure 10: OPTICS reachability diagram for the best-performing configuration obtained with FS4.

In contrast, FS4 forms 7 well-separated clusters with clearer density valleys and a more interpretable structure. Despite a slightly lower Silhouette score, **FS4 (*Max with Rate*)** was selected as the optimal configuration with a **Silhouette score of 0.317**.

This selection is further validated by an average reachability distance of 0.0288. Such a low value indicates that the identified clusters possess an extremely high internal density and strong local cohesion, effectively separating dense nuclei from the surrounding sparser data. The extracted clusters represent a significant but compact portion of the dataset, with individual sizes ranging approximately between 5% and 8.2% of the total population. Although approximately 54.09% of points were labeled as noise, this reflects the intrinsic multi-scale density structure of the dataset. The decision to accept this level of noise is supported by the low reachability of the core points, which preserves the interpretability and stability of the clusters. This provides a balanced compromise between internal cohesion, structural coherence, and a meaningful number of clusters for subsequent analyses.

Feature Set	min_samples	ε range	Silhouette range
FS1	8	0.07 – 0.14	-0.081 – 0.148
FS2	4	0.07 – 0.14	-0.178 – 0.165
FS3	4	0.07 – 0.14	-0.116 – 0.292
FS4	4	0.07 – 0.14	-0.071 – 0.317
FS5	16	0.03 – 0.07	0.199 – 0.388

Table 5: Ranges of ε and silhouette scores for OPTICS across all feature sets. Silhouette scores are computed excluding noise points and only when at least two clusters are present.

3.2.3 HDBSCAN

We implemented a two-phase HDBSCAN exploration framework. First, all features were scaled using Min-Max normalization. We then defined a set of utility functions to calculate key clustering quality indicators, including the number of clusters, the proportion of noise points, the silhouette score (computed on non-noise points only), and the mean cluster persistence. Finally, HDBSCAN was run with the specified hyperparameters, and these measures were collected for subsequent evaluation.

The experimental procedure consists of two stages. In Phase 1 (coarse grid search), we explored a broad range of values for **min_cluster_size** (200, 500, 1000), a single distance metric (**Euclidean**), and cluster selection epsilon (**CSE**). Poor configurations, defined as those producing more than 60% noise or a single cluster, were automatically discarded. The two best-performing configurations, based on high silhouette scores and low noise, were retained for further exploration.

Phase 2 (fine search) refined the search around the best Phase 1 configurations by varying additional parameters, including the distance metric (**Euclidean** and **Manhattan**), cluster selection epsilon (0.0, 0.02, 0.05), and the cluster selection method (**eom** and **leaf**). This two-stage approach allows a balance between computational efficiency and thorough exploration of the hyperparameter space, facilitating the identification of robust and interpretable clustering solutions.

The application of HDBSCAN to the different feature sets highlights how the resulting cluster structures are primarily driven by feature composition, with algorithmic parameters playing a secondary but non-negligible role. Feature sets including the *Rating* variable (*low*, *medium*, *high*), consistently yield three clusters, suggesting that rating acts as a dominant structuring axis in the feature space. In these configurations, HDBSCAN identifies three well-separated density regions corresponding to rating levels, while the remaining variables mainly contribute to intra-cluster variation rather than to further segmentation. Although these solutions exhibit relatively good silhouette values and limited noise, they remain weak from an exploratory perspective, as the presence of a low-cardinality ordinal variable constrains the emergence of additional stable substructures.

In contrast, feature sets excluding the rating show more variable and less constrained clustering outcomes. The *max no rate* configuration, combined with the **leaf** cluster selection method, produces a higher number of clusters, but at the cost of increased noise, lower silhouette scores, and reduced persistence, indicating a tendency toward fragmented and unstable solutions driven by local density fluctuations. The most balanced result is obtained with the ***base + synthetic*** feature set (we consider this the **best result with a silhouette score of 0.486**), where the rating variable is complemented by behavioral and aggregate features. In this case, the use of the **eom** method allows the identification of 5 clusters with moderate noise and satisfactory separation, suggesting that the influence of rating is attenuated and integrated into a richer, more interpretable structure.

The quantitative distribution of the identified clusters reveals a highly polarized density structure within the feature space. The dataset is primarily partitioned into two large macro-segments, Cluster 2 (37.8%) and Cluster 0 (31.6%), which together encompass over two-thirds of the clusterized observations. This concentration suggests the presence of two predominant density peaks, reflecting the most frequent configurations of game mechanics and engagement metrics. In contrast, Cluster 3 represents a significant but smaller grouping (11.7%), while Clusters 1 and 4 appear as distinct, low-cardinality niches, each accounting for approximately 1.9% of the total. The identification of these smaller, highly-dense clusters highlights the effectiveness of the HDBSCAN algorithm in isolating local density peaks that would typically be absorbed into larger groups by center-based methods.

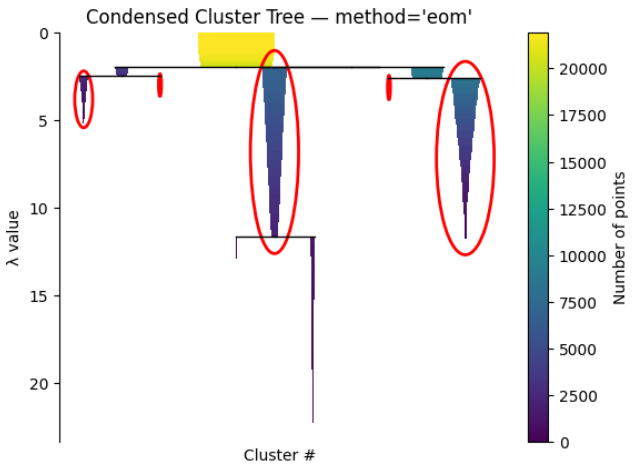


Figure 11: HDBSCAN condensed cluster tree for the best-performing configuration.

Finally, the noise rate of 15.2% indicates a portion of the dataset that does not conform to the primary structural patterns, further justifying the use of a density-based approach to ensure the cohesion of the resulting segments.

Regarding parameter settings, the results indicate that the *eom* method favors more stable and semantically coherent clusters, whereas *leaf* increases granularity at the expense of robustness. The Manhattan distance appears better suited to heterogeneous feature spaces, supporting the identification of multiple density-based clusters, while the Euclidean distance tends to produce more compact but less articulated solutions. Finally, higher values of *min_cluster_size* promote stability but limit the number of detected clusters, whereas intermediate values enable a more informative trade-off between robustness and structural complexity when combined with sufficiently expressive feature sets.

Analyzing the results in Table 6, a clear trade-off between granularity and persistence emerges. The *Min no rate* configuration achieves the highest silhouette score (0.628) but shows relatively low persistence (0.195), suggesting that while the two clusters are well-separated, they may be susceptible to minor data perturbations. Conversely, the *Base + synthetic* set maintains a balanced persistence of 0.115 despite identifying five clusters, indicating that behavioral features enable a more complex yet structurally sound segmentation. Practically, the lower persistence in fragmented solutions (such as *Max no rate*, 0.105) confirms that the *leaf* method captures transient density peaks lacking the long-term stability required for reliable board game categorization.

Feature set	Metric	Method	MinClSize	Clusters	Noise	Silhouette	Persistence
Max con rate	Manhattan	eom	1000	3	0.309	0.447	0.213
Min con rate	Manhattan	eom	1000	3	0.157	0.502	0.307
Max no rate	Manhattan	leaf	200	8	0.583	0.375	0.105
Min no rate	Euclidean	eom	500	2	0.128	0.628	0.195
Base + synthetic	Manhattan	eom	200	5	0.151	0.486	0.115

Table 6: Comparison of HDBSCAN results across different feature sets

3.3 Hierarchical clustering

Hierarchical clustering was evaluated across multiple feature sets and linkage strategies (Ward, Complete, and Average). The detailed results of all experiments, including the selected number of clusters and silhouette scores, are summarized in Table 7.

Feature Set	Linkage	k	Silhouette
FS1	Ward	10	0.211
	Complete	5	0.137
	Average	3	0.371
FS2	Ward	6	0.267
	Complete	9	0.186
	Average	3	0.562
FS3	Ward	10	0.196
	Complete	6	0.193
	Average	6	0.359
FS4	Ward	4	0.218
	Complete	8	0.118
	Average	5	0.308
FS5	Ward	6	0.231
	Complete	4	**0.680**
	Average	3	0.662

Table 7: Summary of hierarchical clustering experiments

Empirical observations suggest that Hierarchical Clustering, particularly when employing *average linkage* strategies, often yields dendrograms that appear structurally intuitive and well-partitioned. However, this apparent visual coherence frequently fails to translate into high internal validation metrics; such models often exhibit significantly low Silhouette coefficients. This discrepancy typically arises because average linkage tends to minimize the distance between the clusters' collective means, potentially ignoring local density or boundary overlaps that the Silhouette is sensitive to. Consequently, while the resulting hierarchy may seem cleaner from a taxonomic perspective, the quantitative distinctness of the individual clusters remains statistically weak.

Overall, the configuration that achieved the best performance combines the extended feature set, **FS5**, with Complete linkage, yielding the highest **Silhouette Score (0.6803)**. Average linkage applied to the same feature set also produced a comparably strong result, with a silhouette value of 0.6624. However, despite these favorable scores, the resulting clustering is characterized by a severe imbalance in cluster sizes across all linkage methods. Specifically, one cluster accounts for approximately 99.5% of the data points, while the remaining clusters represent only marginal portions of the dataset, with sizes of about 0.43%, 0.05%, and 0.02%, respectively. This pronounced dominance of a single cluster indicates that the data distribution is heavily skewed, which may reduce the interpretability and practical relevance of the clustering solution.

3.4 Discussion

The comparative analysis of the clustering experiments reveals a fundamental trade-off between the mathematical purity of the clusters and their practical interpretability within the board game market context. While each algorithm provided unique insights into the dataset's topography, the structural characteristics of the results varied significantly across the different methodologies.

Density-based algorithms, such as **DBSCAN** and **OPTICS**, demonstrated a superior ability to filter noise and identify high-density nuclei. However, this precision came at a substantial cost in terms of data coverage. In the case of OPTICS, the necessity to label over 54% of the observations as noise to achieve a stable structure significantly limits its utility for a comprehensive market analysis. Although these methods effectively isolated the most dense and cohesive groups, they failed to provide a holistic view of the dataset, leaving more than half of the games unclassified and thus unavailable for a global profiling.

Similarly, the **hierarchical clustering** results presented a paradoxical outcome. Despite achieving the highest silhouette scores among all tested methods (up to 0.68), the resulting partitions were characterized by a severe

dimensional imbalance. The emergence of a "monocluster" containing 99.5% of the points renders the model practically useless for segmentation, as it fails to distinguish between the diverse archetypes present in the data. This phenomenon suggests that while the hierarchical approach can identify a singular dense core, it lacks the sensitivity to partition the broader, more continuous regions of the board game feature space.

In contrast, **HDBSCAN** offered a more sophisticated middle ground, particularly with the *Base + Synthetic* feature set. By identifying five clusters with a balanced persistence, it proved that behavioral and synthetic features could attenuate the dominance of the rating variable. Nevertheless, the resulting distribution remained highly polarized, with two macro-segments absorbing nearly 70% of the classified points. While this provided excellent theoretical insights into the hierarchical stability of the data, it offered less granularity than required for a detailed profiling of specific market niches.

Consequently, the **K-Means** algorithm applied to **Feature Set 3 (FS3)** was identified as the most robust and actionable solution for the final profiling phase. Despite its theoretical simplicity, K-Means produced the most balanced and representative distribution of the entire population, with five clusters ranging in size from 5.8% to 36.2%. This configuration ensures that every segment of the market, from mainstream titles to specific expert niches, is adequately represented.

The silhouette score of 0.465, combined with the clear separation visible in the PCA projection, confirms that incorporating user ratings alongside gameplay characteristics creates a partition that is sia statistically sound and semantically rich. By assigning every record to a well-defined prototype, K-Means provides the most exhaustive foundation for characterizing the diverse board game archetypes in the subsequent analysis.

Table 8: Summary of Clustering Performance and Selection Rationale

Algorithm	**Separation**	**Distribution**
K-Means (FS3)	High (0.465)	Balanced
DBSCAN	Low (0.261)	Fragmented
OPTICS	Moderate (0.317)	Extreme Noise
HDBSCAN	High (0.486)	Polarized
Hierarchical	**Max (0.680)**	Degenerate (99%)

3.5 Cluster Profiling and Archetype Interpretation

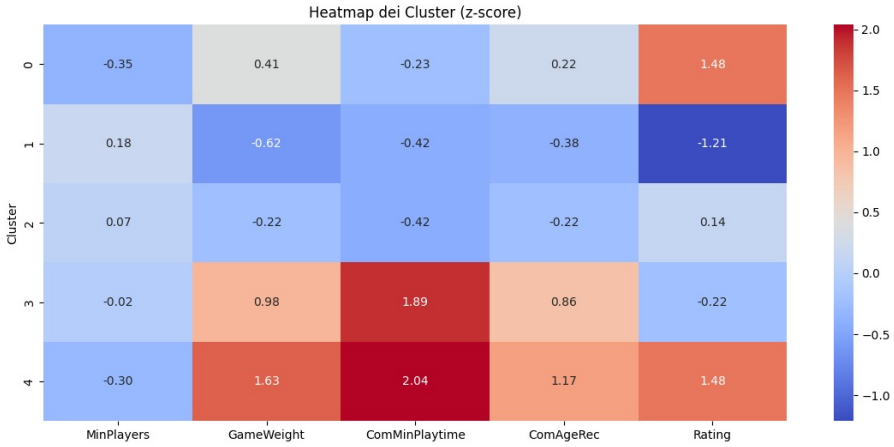


Figure 12: Heatmap of standardized feature means (z-scores) for each cluster obtained with K-means on Feature Set 3.

To interpret the final clustering solution, a cluster profiling analysis was performed on the results obtained with K-means, selected as the final algorithm for its superior stability, interpretability, and clustering quality. The analysis focuses on the solution derived from FS3 (*MinPlayers*, *GameWeight*, *ComMinPlaytime*, *ComAgeRec*, *Rating*), with all features standardized using z-scores computed over the entire dataset to ensure inter-cluster comparability. Standardized feature means were examined to derive interpretable archetypes, revealing clear differentiation primarily along the dimensions of game complexity and time investment, while *Rating* acts as a discriminating but non-dominant factor, indicating that community appreciation emerges from the interaction between complexity, accessibility, and audience targeting rather than from complexity alone.

Cluster 0 is primarily distinguished by a high average *Rating* ($z = 1.48$), while the remaining features remain close to the dataset mean. *GameWeight* is slightly above average ($z = 0.41$), whereas *ComMinPlaytime* and *ComAgeRec* are mildly below or near the mean. This profile suggests a group of games that achieve strong community approval while maintaining moderate complexity and accessibility, representing well-balanced and broadly appealing titles.

Cluster 1 groups games with below-average complexity (*GameWeight* $z = -0.62$) and significantly lower *Rating* ($z = -1.21$). Despite their accessibility in terms of playtime and age recommendation, these titles appear to be less appreciated by the community, indicating that simplicity alone is insufficient to guarantee positive reception.

Cluster 2 represents a baseline or mainstream segment, with all features close to the dataset mean and a *Rating* near zero ($z = 0.14$). These games exhibit no strong distinguishing characteristics and can be interpreted as standard or average titles that neither strongly attract nor repel players.

Cluster 3 is characterized by high complexity (*GameWeight* $z = 0.98$) and long playtimes (*ComMinPlaytime* $z = 1.89$), combined with a higher recommended age ($z = 0.86$), while the *Rating* remains close to the mean ($z = -0.22$). This suggests games aimed at experienced players, where increased depth does not necessarily translate into broader appreciation.

Cluster 4 represents the most extreme profile, combining very high complexity (*GameWeight* $z = 1.63$), very long playtimes (*ComMinPlaytime* $z = 2.04$), and higher age recommendations ($z = 1.17$) with a strongly positive *Rating* ($z = 1.48$). This cluster captures high-quality, expert-oriented games where the substantial cognitive and temporal investment is consistently rewarded by strong community approval.

Overall, the cluster profiling confirms that the selected feature set and the K-means solution effectively capture meaningful and interpretable segments within the dataset. The resulting archetypes highlight distinct relationships between complexity, accessibility, and perceived quality, providing a solid foundation for subsequent qualitative analysis and discussion.

4 Classification

To ensure a consistent evaluation across different models, two primary feature configurations were defined and systematically applied to the KNN, Naive Bayes, and Decision Tree classifiers. This standardized approach allows for a direct comparison of how different algorithmic architectures leverage raw gameplay mechanics versus engineered engagement metrics.

Configuration	Features Included
Baseline	GameWeight, ComAgeRec, NumWant, ComMaxPlaytime
Extended	GameWeight, ComAgeRec, NumWant, ComMaxPlaytime, BestPlayers, PlaytimeRange, OwnershipRatio

Table 9: Feature Set Configurations for Classification Models

4.1 KNN

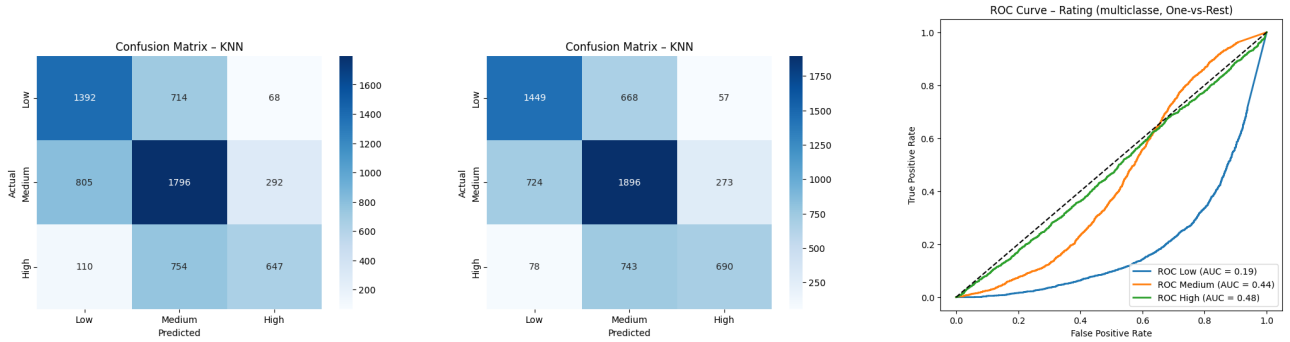
Feature selection using SelectKBest identified four primary features—*GameWeight*, *ComAgeRec*, *NumWant*, and *ComMaxPlaytime*—as the most informative for predicting the target variable Rating. The dataset was split into 70/30 training/test sets, and hyperparameters were optimized via RandomizedSearchCV over 40 random combinations using 5-fold cross-validation.

Two KNN models were evaluated: a baseline model using the original features, and an extended model incorporating synthetic features. The set of features used in the extended model included *GameWeight*, *ComAgeRec*, *BestPlayers*, *NumWant*, *ComMaxPlaytime*, *PlaytimeRange*, and *OwnershipRatio*, capturing both gameplay characteristics and player engagement. The optimized configurations and overall performance metrics are summarized in Table 10.

Model	Neighbors	Metric	Weighting	Test Accuracy	Weighted F1
Baseline (original features)	40	cityblock	uniform	0.583	0.58
Extended (synthetic features)	47	cityblock	distance	0.613	0.61

Table 10: KNN Classification Results (Test Set)

The improvement of the extended model can be further appreciated by comparing the confusion matrices of the two experiments and the ROC curve of the synthetic features configurations in Figure 13. The extended feature set increases correct predictions across all classes.



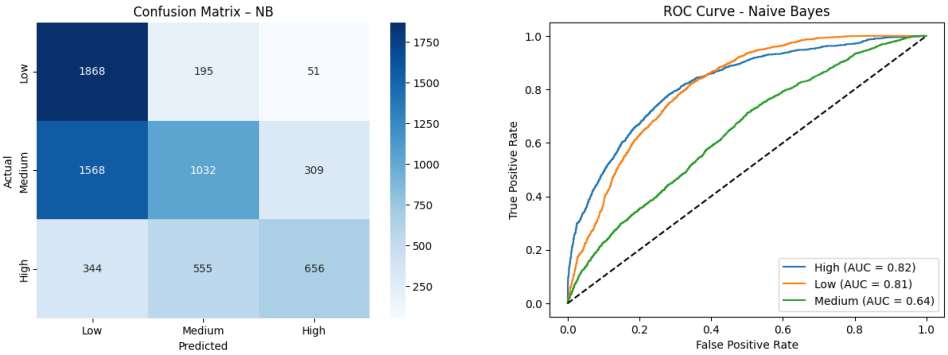
(a) KNN - Base features (b) KNN - Synthetic features (c) ROC - Synthetic Features

Figure 13: Confusion matrices for different KNN feature configurations and ROC curve for KNN Synthetic Features.

Overall, the inclusion of synthetic features captures additional gameplay and engagement patterns, which improves the KNN model's ability to predict user ratings.

4.2 Naive Bayes

The Naive Bayes classifier was first trained using the features selected for KNN via SelectKBest. These features included *GameWeight*, *ComAgeRec*, *NumWant*, and *ComMaxPlaytime* capturing key aspects of gameplay complexity and player engagement. The dataset was split into a 70/30 training-test ratio, and the model achieved an overall accuracy of 0.541 on the test set. The High rating class reached a precision of 0.65 but a lower recall of 0.42, indicating some difficulty in identifying all highly rated games. The Low class exhibited high recall (0.88) but lower precision (0.49), suggesting a tendency to over-predict this class. The Medium class achieved a precision of 0.58 and a recall of 0.35. Overall, the weighted F1-score was 0.52, reflecting moderate predictive performance and an imbalance in class identification.



(a) Naive Bayes – Base features (b) Naive Bayes – ROC curve

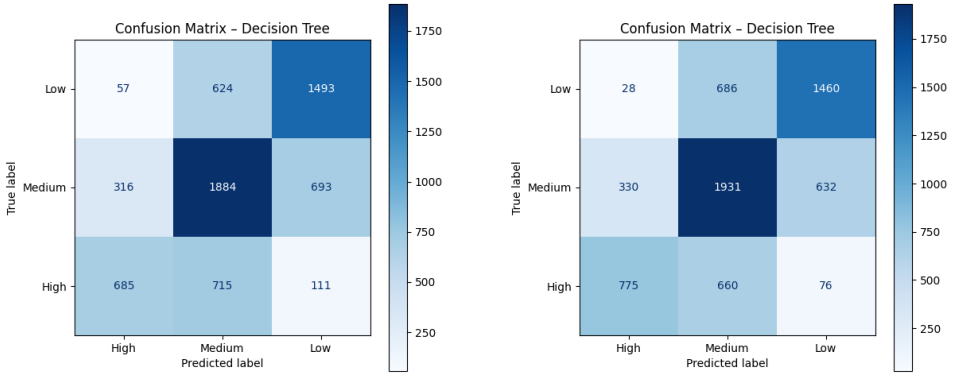
Figure 14: Confusion matrix and ROC curve for base features.

To evaluate the impact of synthetic features, the model was retrained including *PlaytimeRange*, and *OwnershipRatio* alongside the original four features. With this extended feature set, the overall accuracy slightly decreased to 0.524, indicating that the inclusion of synthetic features did not improve performance and slightly exacerbated the imbalance across classes. This suggests that Naive Bayes' assumption of feature independence limits its ability to fully leverage correlated synthetic variables in multi-class rating prediction.

4.3 Decision Tree

In line with the previous experiments, the Decision Tree was trained on both the original SelectKBest features and the extended synthetic set, adhering to the same data distribution and 70/30 splitting criteria. This ap-

proach ensures that performance variations can be attributed to the model's architecture rather than differences in the input data.



(a) Decision Tree – Base features (b) Decision Tree – Synthetic features

Figure 15: Confusion matrices for base and synthetic features.

The optimal baseline configuration, identified via randomized search (30 iterations), utilized **entropy** as the splitting criterion, a `max_depth` of 15, and constraints of 100 samples per leaf and 20 per internal split. The primary discriminant is *NumWant*, which holds a dominant feature importance of **0.68** and serves as the root node. The second level relies on both *NumWant* and *GameWeight* (importance: **0.27**), indicating that community interest and game complexity are the main rating predictors. *NumWant* remains highly influential further down the hierarchy, accounting for three out of four splits at the third level. However, the model exhibited significant overfitting, with accuracy dropping from **0.87** (train) to **0.62** (test). This is particularly evident for the *High* class, which recorded the lowest recall (**0.45**). The confusion matrix (*15a*) shows a systematic misclassification of extreme ratings as *Medium*; notably, 1,493 *Low* instances were incorrectly labeled as *Medium*.

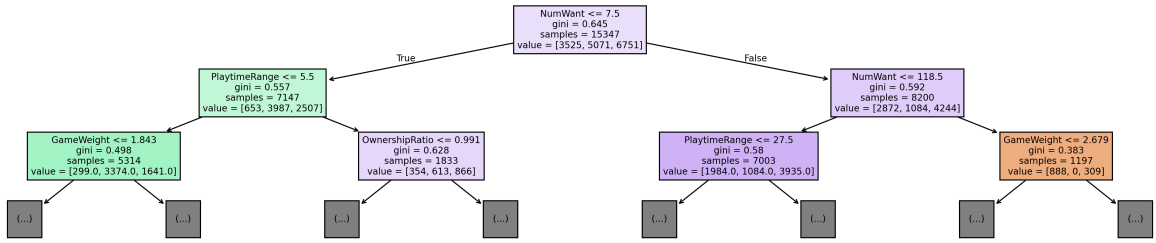


Figure 16: Decision Tree with Extended subset

The inclusion of synthetic features in the extended dataset fundamentally altered the model's architecture. The best configuration prioritized a shallower (`max_depth`: 7) yet denser tree using the **Gini** index and stricter regularization (`min_samples_leaf`: 30). *Figure 16* highlights the impact of engineered variables: while *NumWant* remains the root (importance: **0.55**), the subsequent levels are driven by new features such as *PlaytimeRange* (**0.18**) and *OwnershipRatio* (**0.12**). This integration enhanced generalization, as test accuracy and macro-averaged F1-score both rose to **0.63**. Specifically, the *High* class recall increased to **0.51**, and its ROC AUC reached **0.70** (up from 0.66), suggesting that synthetic features better capture success dynamics. Despite these improvements, substantial overfitting persists (0.93 train vs 0.63 test).

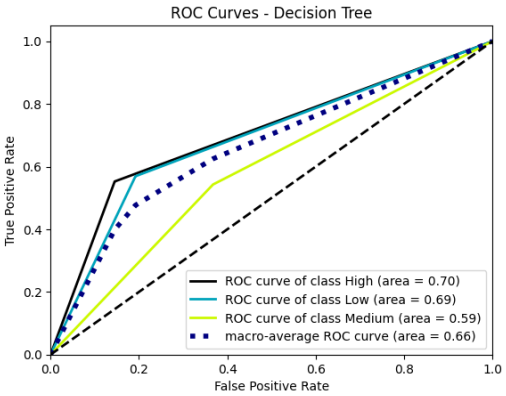


Figure 17: Decision Tree - ROC Curve

The Extended confusion matrix (*15b*) confirms that while total errors decreased, significant overlap remains between *Medium* and other classes (AUC: **0.59**). In conclusion, although features like *OwnershipRatio* capture community behaviors absent in the baseline, the persistent gap between training and test performance suggests that structural complexity alone does not account for the model's limitations.

To further investigate this behavior and attempt to mitigate overfitting, the impact of the α parameter and post-pruning techniques were evaluated. As α increases, the number of nodes decreases progressively, yet no "sweet spot" was identified. Both training and testing accuracy showed a simultaneous decline even for marginal values of α, indicating that pruning does not recover performance on unseen data. These results suggest that overfitting is not merely a consequence of excessive tree depth, but rather stems from intrinsic noise or significant class overlap (particularly between *Medium* and *Low* ratings) that hierarchical simplification cannot resolve.

Consequently, the configuration obtained through randomized Search was retained, as forced post-pruning would degrade overall predictive power without addressing the underlying challenges in class separation.

Class	**Precision**		**Recall**		**F1-score**	
	Base	*Ext.*	*Base*	*Ext.*	*Base*	*Ext.*
High	0.65	0.68	0.45	0.51	0.53	0.59
Low	0.65	0.67	0.69	0.67	0.67	0.67
Medium	0.58	0.59	0.65	0.67	0.62	0.63
Macro Avg	0.63	0.65	0.60	0.62	0.61	**0.63**
Accuracy			—		0.62	**0.63**

Table 11: Decision Tree - detailed classification metrics: Baseline vs. Extended Features

4.4 Discussion

The comparative analysis of KNN, Naive Bayes, and Decision Tree models highlights the decisive impact of feature engineering on predicting game ratings. A consistent performance uplift was observed across most models when incorporating the **extended feature set**. In the KNN classifier, synthetic metrics improved test accuracy to 0.613, suggesting that engagement-based synthetic features capture latent success patterns better than raw gameplay mechanics alone. Similarly, the Decision Tree benefited from these variables, shifting its reliance from *NumWant* to a more balanced architecture that improved the *High* class recall and ROC AUC, confirming that feature augmentation is vital for modeling complex community dynamics.

However, the models' varying responses to these features reveal significant theoretical insights. Unlike the other classifiers, Naive Bayes experienced a slight performance degradation (0.541 to 0.524), likely due to the violation of its **conditional independence assumption** caused by the inherent correlation between synthetic and original features. Meanwhile, the Decision Tree exhibited severe overfitting that remained unresolved by α-pruning, indicating that the error stems from **intrinsic noise** and substantial class overlap rather than mere structural complexity.

Ultimately, the persistent difficulty in distinguishing the *Medium* class across all models suggests that game ratings exist on a continuum rather than in discrete clusters. While the distance-weighted KNN emerged as the most robust approach, the persistent gap between training and testing performance indicates a clear predictive ceiling for these algorithms.

Model	Feature Set	Accuracy	Macro F1	Key Strength
KNN	Extended	0.613	0.610	Robustness to outliers
Naive Bayes	Baseline	0.541	0.520	Fast baseline
Decision Tree	Extended	**0.630**	**0.630**	Best class separation

Table 12: Comparative summary of best classification results for each algorithm

As summarized in Table 12, the **Decision Tree with Extended features** achieved the highest predictive performance, reaching an accuracy and Macro F1-score of 0.63. While KNN demonstrated a high level of stability and Naive Bayes served as a useful baseline for independence-based modeling, the Decision Tree proved most capable of leveraging engineered variables like *OwnershipRatio* to improve the identification of the most critical classes (High and Low). Despite the persistent challenge of class overlap in the Medium range, this comparison confirms that the inclusion of synthetic features, combined with a tree-based hierarchical split, provides the most effective framework for predicting user ratings in this dataset.

5 Regression

5.1 Simple linear regression

The regression task was performed with *GameWeight* as the target variable. A simple linear regression model was first trained using the single feature showing the highest correlation with the target. Based on the correlation analysis, *ComMaxPlaytime* was selected as the independent variable.

The model achieved an R^2 of 0.424, with an RMSE of 0.647 and a MAE of 0.503. These results indicate that the maximum recommended playtime alone is able to explain a relevant portion of the variability in game complexity. However, the magnitude of the error metrics suggests that additional factors are needed to capture the full complexity of the target variable.

5.2 Multiple linear regression

Multiple linear regression was then applied using the most informative features identified through correlation analysis. For *GameWeight*, the selected features were *ComMaxPlaytime*, *ComMinPlaytime*, *ComAgeRec*, *NumCategories*, and *PlaytimeRange*.

This configuration significantly improved model performance, achieving an R^2 of 0.507, with an RMSE of 0.599 and a MAE of 0.455. Compared to the simple linear regression, the increase in explained variance and the reduction in error values confirm the relevance of combining multiple gameplay and structural characteristics to better explain game complexity.

5.3 Non-linear regression

To further investigate possible non-linear relationships, Decision Tree Regression and K-Nearest Neighbors (KNN) Regression were applied using the same feature set adopted for the multiple linear regression model.

For *GameWeight*, Decision Tree Regression obtained an R^2 of 0.482, with an RMSE of 0.613 and a MAE of 0.453, while KNN Regression achieved an R^2 of 0.500, with an RMSE of 0.603 and a MAE of 0.446. These results are comparable to those of multiple linear regression, suggesting that although some non-linear patterns are present, the linear model already provides a strong approximation of the relationship between the selected features and game complexity.

Overall, the best predictive performance for *GameWeight* was achieved by multiple linear regression and KNN regression, both providing R^2 values close to 0.50. Among these approaches, multiple linear regression offers a favorable trade-off between predictive accuracy and interpretability, making it a suitable choice for modeling game complexity.

5.4 Multi-output regression

To simultaneously predict *GameWeight* and *ComAgeRec*, a multi-output Decision Tree Regressor was trained using the same feature set. The model achieved an aggregated R^2 of 0.421, with an RMSE of 1.672 and a MAE of 1.098 across both targets.

Evaluating the targets separately, the model obtained for *GameWeight* an R^2 of 0.505, with an RMSE of 0.600 and a MAE of 0.455, indicating a reasonable fit and low error. For *ComAgeRec*, the performance was lower, with an R^2 of 0.338, an RMSE of 2.287, and a MAE of 1.741, reflecting the higher variability and complexity of this target.

These results highlight that the multi-output approach can effectively model both targets within a unified framework, while also showing that predictive performance may vary across variables.

Table 13 summarizes the performance of all experiments.

Model	R^2	RMSE	MAE
GameWeight			
Simple Linear Regression	0.424	0.647	0.503
Multiple Linear Regression	0.507	0.599	0.455
Decision Tree Regression	0.482	0.613	0.453
KNN Regression	0.500	0.603	0.446
Multi-output Decision Tree	0.505	0.600	0.455
ComAgeRec			
Multi-output Decision Tree	0.338	2.287	1.741

Table 13: Performance comparison of all regression models. *GameWeight* results include single- and multi-output models; *ComAgeRec* results are reported only for the multi-output Decision Tree.

6 Pattern mining

The pattern mining workflow began with a discretization phase (as shown in Table 14) aimed at transforming continuous numerical variables into categorical representations suitable for association analysis. Discretization was performed using a combination of semantic binning, logarithmic transformation followed by quartile-based discretization, and predefined categorical scales. This preprocessing step enabled the application of frequent pattern mining techniques and improved both the interpretability and support stability of the extracted patterns. The final set of discretized features included *YearPublished_disc*, *GameWeight_disc*, *MinPlayers_disc*, *MaxPlayers_disc*, *ComAgeRec_disc*, *NumOwned_disc*, *NumWant_disc*, *ComMinPlaytime_disc*, *ComMaxPlaytime_disc*, *NumWeightVotes_disc*, and *Rating_disc*.

Original variable	Discretization method	Resulting categories
YearPublished	Semantic binning (decades)	<1990; 1990s; 2000s; 2010s; 2020s
GameWeight	Semantic thresholds	Light; Medium; Heavy
MinPlayers	Semantic binning	Solo (1); Few (2); Medium (3–4); Many (>4)
MaxPlayers	Semantic binning	Small (≤4); Medium (5–6); Large (7–10); VeryLarge (>10)
ComAgeRec	Age range binning	Child (0–6); Teen (7–12); YoungAdult (13–18); Adult (>18)
NumOwned	Log-transform + quartiles	Low; Medium; High; VeryHigh
NumWant	Log-transform + quartiles	Low; Medium; High; VeryHigh
NumWeightVotes	Log-transform + quartiles	Low; Medium; High; VeryHigh
ComMinPlaytime	Quartiles	VeryShort; Short; Medium; Long
ComMaxPlaytime	Quartiles	VeryShort; Short; Medium; Long
Rating	Predefined categorical scale	Low; Medium; High

Table 14: Discretization of numerical variables for pattern mining

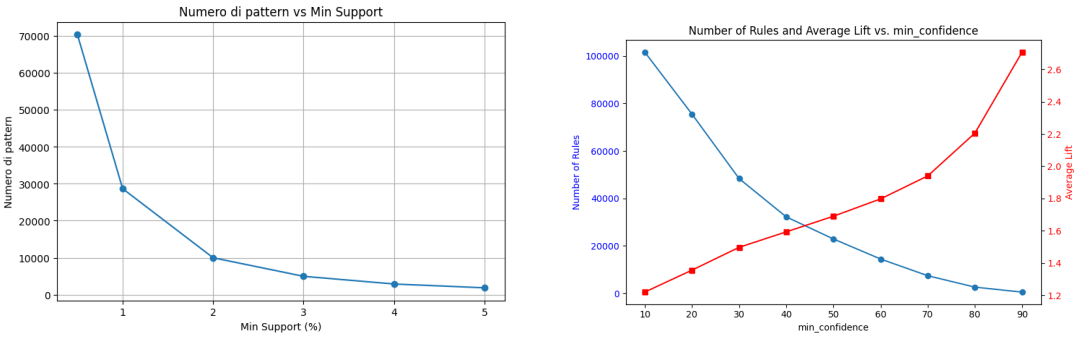
Frequent itemsets were subsequently extracted using the Apriori algorithm after evaluating multiple minimum support thresholds. Support values between 0.5% and 5% were tested, highlighting the trade-off between the number of extracted patterns and their generality (Figure 18a). A minimum support of 3% was selected as a balanced configuration, providing a sufficiently rich yet manageable set of frequent itemsets.

The top 3 frequent itemsets extracted for a minimum support of 3% are reported in Table 15. These patterns highlight common combinations of player numbers and recommended age ranges, showing strong associations in the dataset.

Itemset	Support (%)
MinPlayers_disc = Few, ComAgeRec_disc = Teen	49.24
MaxPlayers_disc = Small, MinPlayers_disc = Few	41.77
MaxPlayers_disc = Small, ComAgeRec_disc = Teen	37.86

Table 15: Top 3 frequent itemsets for a minimum support of 3%

On this basis, association rules were generated by varying the minimum confidence threshold. Confidence values ranging from 10% to 90% were explored (Figure 18b), revealing a decreasing number of rules paired with increasing average lift.



(a) Variation of minimum support values

(b) Variation of confidence values

Figure 18: Exploring minimum support and confidence values

A minimum confidence of 50% was chosen as an effective compromise between rule reliability and coverage. For all extracted rules, confidence and lift were computed and summarized in Figure 19 to assess their strength and relevance.

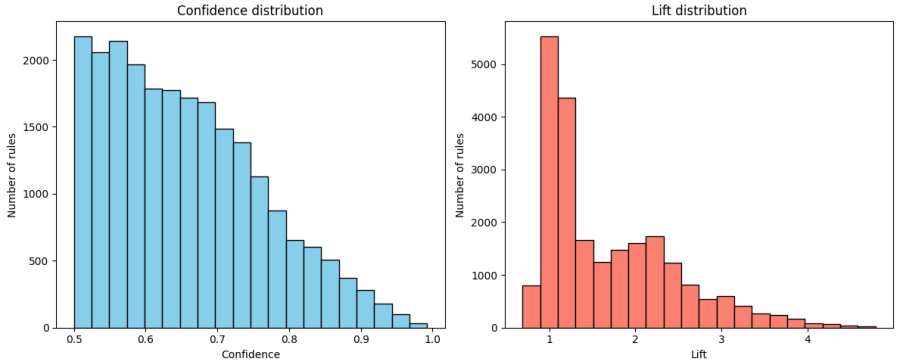


Figure 19: Confidence and lift

From the resulting rule set, five representative association rules were selected (Table 16) to illustrate the most meaningful and interpretable patterns identified in the data.

Antecedent	**Consequent**		**Sup. (%)**	**Conf.**	**Lift**
YearPublished_disc = 1990s, ComMaxPlaytime_disc = Long	ComMinPlaytime_disc = Long	=	2.90	0.97	4.79
ComAgeRec_disc = YoungAdult, ComMinPlaytime_disc = Long, Rating_disc = High, MaxPlayers_disc = Small	ComMaxPlaytime_disc = Long	=	2.87	0.94	4.78
ComAgeRec_disc = YoungAdult, ComMinPlaytime_disc = Long, Rating_disc = High, GameWeight_disc = Heavy	ComMaxPlaytime_disc = Long	=	3.55	0.93	4.74
ComAgeRec_disc = YoungAdult, ComMinPlaytime_disc = Long, Rating_disc = High	ComMaxPlaytime_disc = Long	=	3.88	0.93	4.73
YearPublished_disc < 1990, ComMaxPlaytime_disc = Long, MaxPlayers_disc = Small	ComMinPlaytime_disc = Long	=	2.90	0.95	4.71

Table 16: Representative Association Rules

The analysis of the extracted association rules highlights a strong internal consistency among variables related to game duration. In particular, games characterized by long *ComMaxPlaytime* and *ComMinPlaytime* are frequently associated with high *Ratings*, higher *GameWeight*, and a *ComAgeRec* of young adults. The high lift values indicate that these relationships are not random but reflect significant dependencies between extended playtime and specific structural features, such as a limited number of players or the publication period. Overall,

the rules outline a profile of demanding games designed for long sessions and a mature audience, showing stable characteristics across different publication eras.

These rules were then leveraged for a downstream classification task, with the discretized rating used as the target variable.

The rule-based classifier achieved an accuracy of 0.597, indicating moderate predictive performance and demonstrating that the extracted association rules capture non-trivial relationships between game characteristics and their corresponding ratings.

After analyzing the association rules derived from the original features, we extended the pattern mining analysis by including a set of synthetic attributes. The analysis highlights a consistent pattern that links the characteristics of the game with the target age group *YoungAdult* (13–18). In particular, games that are highly rated (*Rating_disc = High*), exhibit high complexity (*GameWeight_disc = Heavy*), and belong to a single category (*NumCategories_disc = One*) are frequently associated with a restricted number of players (*RangePlayers_disc = VeryLow*) and are not designed for party settings (*IsPartySize_disc = NotParty*).

These synthetic features collectively identify games that are specialized, strategically demanding, and intended for small groups, which aligns with the behavioral patterns of young adult players.

Overall, the inclusion of synthetic features allows for a more nuanced understanding of the game profiles preferred by the *YoungAdult* audience, reinforcing and extending the insights obtained from the base-feature analysis.

7 Conclusion

The extensive analysis conducted on the BoardGameGeek dataset provides a multi-faceted answer to the research question regarding the characteristics and configurations that maximize a board game's success and popularity. By integrating clustering, predictive modeling, and pattern mining, we have identified that success is not a monolithic concept but emerges from the strategic alignment of complexity, accessibility, and audience targeting.

The **cluster profiling phase** provided the most significant archetypal insights. Our analysis of the K-Means archetypes (Figure 12) revealed that high community appreciation is concentrated in two distinct poles. *Cluster 0* represents the "Optimal Balance" archetype: titles that achieve high ratings ($z = 1.48$) by maintaining moderate complexity and high accessibility. Conversely, *Cluster 4* represents the "Expert Excellence" archetype: extreme complexity and long playtimes that are "rewarded" by the community with equally high ratings. Crucially, the profiling of *Cluster 1* demonstrated that simplicity alone (low *GameWeight*) is insufficient for success, as it often correlates with lower community interest if not supported by engaging mechanics.

The **classification and regression tasks** further refined this picture. The transition to the Extended feature set (Table 11) proved that community-driven metrics, such as *OwnershipRatio* and *NumWant*, are the strongest predictors of success. The Decision Tree models highlighted that once a game reaches a certain threshold of "desirability" its technical configuration becomes secondary to its market momentum. Furthermore, **pattern mining** confirmed that for the "Expert" segment, a strong internal consistency (long playtimes associated with higher age recommendations and niche player counts) is a fundamental requirement for stability and high ratings.

In conclusion, to maximize success and popularity, a board game should be designed following one of two evidence-based strategic profiles:

1. **The Refined Core Path (Cluster 0):** Focusing on moderate complexity ($z \approx 0.4$) and high accessibility in terms of playtime and age, aiming for a broad but highly satisfied audience.
2. **The High-Fidelity Expert Path (Cluster 4):** Targeting a mature audience with high cognitive demand (*Weight* > 3.5) and long sessions, where the temporal investment is perceived by the community as a proxy for depth and quality.

Ultimately, while gameplay mechanics set the baseline, the interaction between complexity and perceived quality is what determines a title's standing. These findings provide board game designers and publishers with a robust framework to align their design choices with the specific expectations of different market segments, effectively reducing uncertainty in the publishing process.