Department of Computer Science

Master programme in Data Science and Business Informatics

Data Mining: Foundations

Project report

Board Games Dataset

Eraldo Bushpepa

A.Y. 2025/2026

Contents

Data Understanding and Preparation

1.1 Dataset

The *Board Games Dataset* contains information regarding more than 20k board games rated by an online board game community. The information provided range from year of publication, game difficulty, number of players, information about games rank in specific categories and game characteristics/themes. The variables are specified in the table below:

Name	Description	Type
BGGId	Game Id	int64
Name	Name of game	object
Description	Description of the game	object
YearPublished	Year in which the game was published	int64
GameWeight	Game complexity from 1 to 5	float64
ComWeight	Community recommended game complexity from 1 to 5	float64
MinPlayers	Minimum number of players	int64
MaxPlayers	Maximum number of players	int64
ComAgeRec	Community's recommended age minimum	float64
LanguageEase	Language requirement	float64
BestPlayers	Community voted best player count	int64
GoodPlayers	List of community voted good player counts	object
NumOwned	Number of users who own this game	int64
NumWant	Number of users who want this game	int64
NumWish	Number of users who wishlisted this game	int64
NumWeightVotes	Number of votes for the weight category received by users	int64
MfgPlaytime	Manufacturer Stated Play Time	int64
ComMinPlaytime	Community minimum play time	int64
ComMaxPlaytime	Community maximum play time	int64
MfgAgeRec	Manufacturer Age	int64
NumUserRatings	Number of user ratings	int64
NumComments	Number of user comments	int64
NumAlternates	Number of alternate versions	int64
NumExpansions	Number of expansions	int64
NumImplementations	Number of implementations	int64
IsReimplementation	Is this game presenting a reimplementation?	int64
Family	Game family the game belongs to	object
Kickstarted	Is this game from a kickstarter (crowdfunding campaign) project?	int64
ImagePath	Image http:// path	object
Rank:strategygames	Rank in strategy games	int64
Rank:abstracts	Rank in abstracts	int64
Rank:familygames	Rank in family games	int64
Rank:thematic	Rank in thematic	int64

Name	Description	Type
`Rank:cgs`	Rank in card games	`int64`
`Rank:wargames`	Rank in war games	`int64`
`Rank:partygames`	Rank in party games	`int64`
`Rank:childrensgames`	Rank in children's games	`int64`
`Cat:Thematic`	Binary is in Thematic category	`int64`
`Cat:Strategy`	Binary is in Strategy category	`int64`
`Cat:War`	Binary is in War category	`int64`
`Cat:Family`	Binary is in Family category	`int64`
`Cat:CGS`	Binary is in Card Games category	`int64`
`Cat:Abstract`	Binary is in Abstract category	`int64`
`Cat:Party`	Binary is in Party category	`int64`
`Cat:Childrens`	Binary is in Childrens category	`int64`
`Rating`	Game rating (low, medium, high)	`object`

This report contains the summary of the analysis performed on the dataset in two stages: Data Understanding and Preparation, and Clustering. From initial exploration we have:
Total number of records: 21925
Total number of attributes: 46
In our dataset, we found 0 duplicate rows.

1.2 Distribution of variables

An automatic classification of the variables based on their `dtype` and cardinality was performed. This analysis is crucial for determining the correct preparation strategy (e.g., scaling, transformation).

- **Categorical (18 columns):** Object types, IDs, and binary flags.

  BGGId, Name, Description, GoodPlayers, NumComments, IsReimplementation, Family, Kickstarted, ImagePath, Cat:Thematic, Cat:Strategy, Cat:War, Cat:Family, Cat:CGS, Cat:Abstract, Cat:Party, Cat:Childrens, Rating

- **Continuous (4 columns):** All `float64` types.
  GameWeight, ComWeight, ComAgeRec, LanguageEase

- **Discrete (24 columns):** All high-cardinality `int64` types (counts, ranks, etc.).
  YearPublished, MinPlayers, MaxPlayers, BestPlayers, NumOwned, NumWant, NumWish, NumWeightVotes, MfgPlaytime, ComMinPlaytime, ComMaxPlaytime, MfgAgeRec, NumUserRatings, NumAlternates, NumExpansions, NumImplementations, Rank:strategygames, Rank:abstracts, Rank:familygames, Rank:thematic, Rank:cgs, Rank:wargames, Rank:partygames, Rank:childrensgames

A skewness and kurtosis analysis (Table 1.2) was performed on all numeric features to test for normality.

Table 1.2: Skewness and Kurtosis of Numeric Features. High positive values indicate a heavy right-skew and non-normal distributions.

Feature	**Skewness**	**Kurtosis**
GameWeight	0.395861	0.053182
ComWeight	0.302567	0.209106
ComAgeRec	0.143862	-0.381596
LanguageEase	1.671916	4.431087
YearPublished	-11.324235	152.995691
MinPlayers	1.704234	10.722395
MaxPlayers	42.387696	2647.275467
BestPlayers	3.733134	15.745030
NumOwned	12.517373	238.628210
NumWant	6.956857	65.837595
NumWish	9.350407	124.392344
NumWeightVotes	15.317043	366.316525
MfgPlaytime	74.739212	7730.566352
ComMinPlaytime	116.207073	15289.798471
ComMaxPlaytime	74.739212	7730.566352
MfgAgeRec	-0.838558	0.947437
NumUserRatings	12.586978	231.561002
NumAlternates	52.601012	3822.922060
NumExpansions	24.951409	1186.177390
NumImplementations	12.157622	342.326796
Rank:strategygames	-2.569618	4.615674
Rank:abstracts	-4.090590	14.739232
Rank:familygames	-2.571919	4.627397
Rank:thematic	-3.871507	12.995307
Rank:cgs	-8.329871	67.394094
Rank:wargames	-1.857211	1.471660
Rank:partygames	-5.594612	29.305022
Rank:childrensgames	-4.684208	19.947404

The results in Table 1.2 clearly show that most count-based columns (like `NumOwned` and `MaxPlayers`) are extremely right-skewed and not normally distributed. This finding justifies our decision to use a log-transform and a non-parametric scaler (`RobustScaler`) during preparation.

1.3 Outliers detection

To identify outliers, both Z-Score (Table 1.3) and the Interquartile Range (IQR) method (Table 1.4) were used. Given the non-normal, skewed distribution of our data (seen in Section 1.2), the IQR method is considered more reliable.

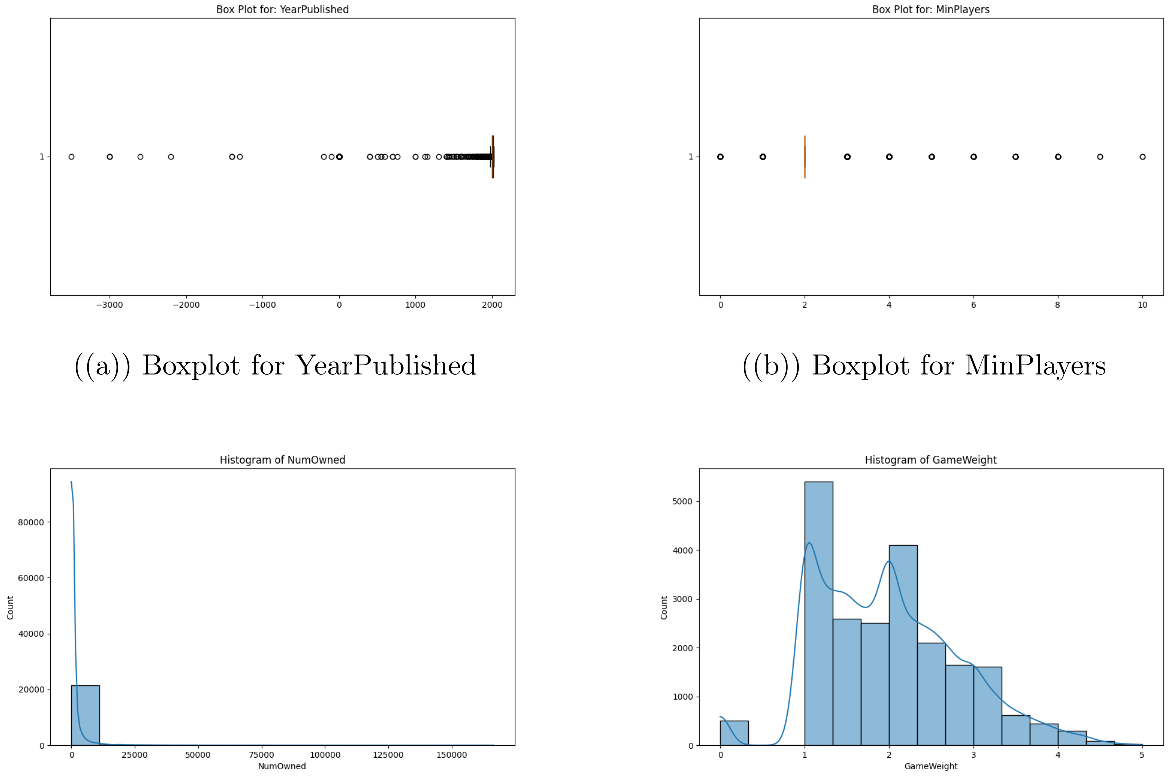
Table 1.3: Potential Outliers (Z-Score > 3)

Feature	**Outlier Count (Z-Score)**
YearPublished	218
GameWeight	51
ComWeight	38
MinPlayers	118
MaxPlayers	190
ComAgeRec	28
LanguageEase	206
BestPlayers	1048
NumOwned	307
NumWant	433
NumWish	353
NumWeightVotes	279
MfgPlaytime	67
ComMinPlaytime	21
ComMaxPlaytime	67
MfgAgeRec	13
NumUserRatings	304
NumAlternates	78
NumExpansions	232
NumImplementations	350
Rank:strategygames	561
Rank:abstracts	1115
Rank:familygames	571
Rank:thematic	1224
Rank:cgs	303
Rank:partygames	640
Rank:childrensgames	881
Cat:Thematic	1224
Cat:CGS	303
Cat:Abstract	1115
Cat:Party	640
Cat:Childrens	881

Table 1.4: Potential Outliers (IQR Method). This method is more robust for our skewed data.

Feature	**Outlier Count (IQR)**
MinPlayers	6886
NumImplementations	4873
Rank:wargames	3530
Cat:War	3530
NumAlternates	3477
Kickstarted	3362
NumUserRatings	3110
NumWish	3030
NumWeightVotes	2938
NumWant	2910
NumOwned	2845
IsReimplementation	2560
Rank:strategygames	2319
Cat:Strategy	2319
Cat:Family	2316
Rank:familygames	2316
NumExpansions	2183
BestPlayers	1981
ComMinPlaytime	1711
MfgPlaytime	1463
ComMaxPlaytime	1463
MaxPlayers	1340
MfgAgeRec	1339
Cat:Thematic	1224
Rank:thematic	1224
YearPublished	**1143**
Rank:abstracts	1115
Cat:Abstract	1115
Cat:Childrens	881
Rank:childrensgames	881
Rank:partygames	640
Cat:Party	640
Cat:CGS	303
Rank:cgs	303
LanguageEase	257
GameWeight	134
ComWeight	100
ComAgeRec	41

The IQR analysis (Table 1.4) confirmed the presence of a large number of outliers. The most critical finding was in `YearPublished`, which showed 1,143 outliers. This was caused by some ancient games (e.g., a minimum value of -3500) and justified our decision to clip this feature before imputation. The plots in Figure 1.1 visualize these distributions.



((a)) Boxplot for YearPublished

((b)) Boxplot for MinPlayers

((c)) Histogram for NumOwned (Highly Skewed)

((d)) Histogram for GameWeight (Near-Normal)

Figure 1.1: Boxplots (top) visualizing outliers and Histograms (bottom) visualizing data skewness.

1.4 Handling missing values

The dataset was checked for missing values, with the results summarized in Table 1.5 and visualized in Figure 1.2.

Table 1.5: Columns with Missing Values

Column	**Missing Count**	**Missing %**
Family	15262	69.61%
LanguageEase	5891	26.87%
ComAgeRec	5530	25.22%
ImagePath	17	0.08%
Description	1	0.00%

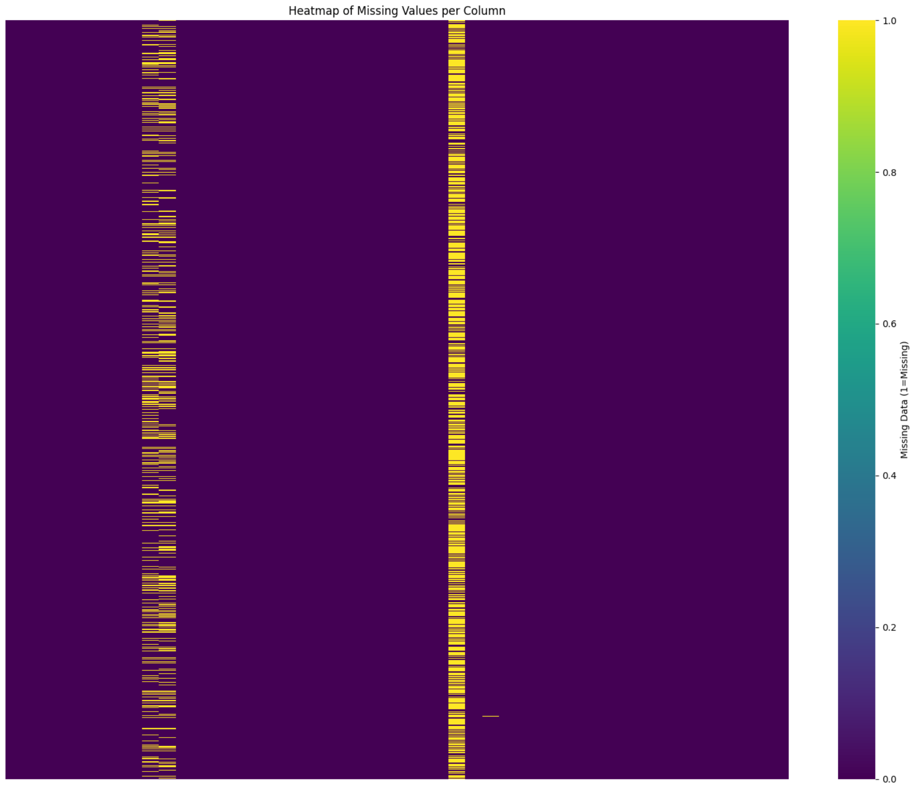


Figure 1.2: Heatmap visualizing missing data. A yellow line indicates a missing value.

Based on this analysis, the following preparation strategy was applied:

- **Family:** This column was dropped, as it was 69.61% empty and thus contained little usable information.
- **LanguageEase & `ComAgeRec`:** These columns were kept. Dropping them would discard over 25% of the data. Instead, the missing values were imputed using their respective **median** value.
- **Other columns:** The few remaining missing values (e.g., in `ImagePath`) were also imputed.

1.5 Dependencies and correlations

Finally, the correlation between all numeric features was calculated to identify redundant data.

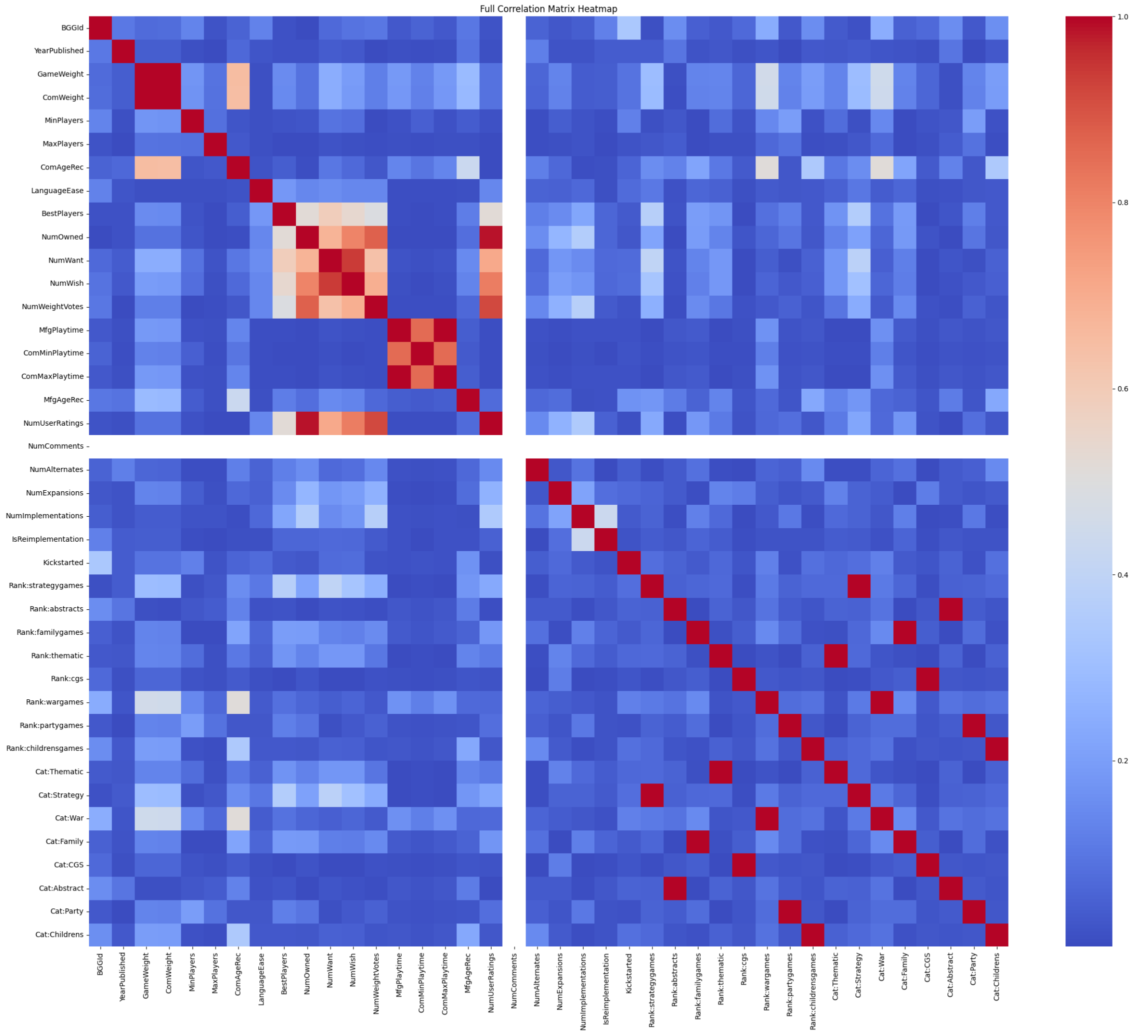


Figure 1.3: Full Correlation Matrix Heatmap.

Table 1.6: Highly Correlated Pairs (Threshold > 0.8)

Feature 1	**Feature 2**	**Correlation**
`MfgPlaytime`	`ComMaxPlaytime`	1.000000
`Rank:cgs`	`Cat:CGS`	0.999992
`Rank:partygames`	`Cat:Party`	0.999962
`Rank:childrensgames`	`Cat:Childrens`	0.999927
`Rank:abstracts`	`Cat:Abstract`	0.999880
`Rank:thematic`	`Cat:Thematic`	0.999854
`Rank:familygames`	`Cat:Family`	0.999421
`Rank:strategygames`	`Cat:Strategy`	0.999415
`Rank:wargames`	`Cat:War`	0.998480
`GameWeight`	`ComWeight`	0.997268
`NumOwned`	`NumUserRatings`	0.985474
`NumWant`	`NumWish`	0.939758
`NumWeightVotes`	`NumUserRatings`	0.917185
`NumOwned`	`NumWeightVotes`	0.874876
`MfgPlaytime`	`ComMinPlaytime`	0.854679
`ComMinPlaytime`	`ComMaxPlaytime`	0.854679
`NumWish`	`NumUserRatings`	0.814348

The correlation matrix (Figure 1.3) and table (Table 1.6) revealed significant redundancy. All `Rank:*` columns were nearly identical to their `Cat:*` counterparts. Furthermore, `ComWeight` was 99.7% correlated with `GameWeight`. Based on these findings, all redundant columns (all `Rank:*` columns, `ComWeight`, `NumWant`, etc.) were dropped.

Final Preparation Strategy

With all analysis complete, a final preparation script (`task_2_analysis.py`) was created to perform the following steps:

1. **Clip Outliers:** The extreme negative values in `YearPublished` were clipped.

2. **Drop Columns:** All redundant, high-missing, and text-based columns were dropped.

3. **Impute Data:** Missing values for `YearPublished`, `ComAgeRec`, and `LanguageEase` were filled using their medians.

4. **Log-Transform:** An automatic skew-detection (threshold > 1.0) was run. All 26 identified skewed columns were transformed using `np.log1p` to normalize their distributions.

5. **Scale Data:** Finally, the data was scaled using `RobustScaler`. This scaler was chosen over `StandardScaler` because our analysis in Section 1.2 proved the data is not normally distributed and `RobustScaler` is not sensitive to the outliers identified in Section 1.3.

This process resulted in the final `dm1_prepared_dataset.csv` file used for clustering.

Clustering

Before starting the cluster analysis, we used the `dm1_prepared_dataset.csv` file from the previous preparation step. This dataset is fully cleaned, imputed, log-transformed, and scaled using `RobustScaler`.

Our clustering analysis followed the project guidelines by testing all three mandatory methods. We performed an experiment by comparing a "Full Feature" model (using all 32 prepared features) against a "Selected Feature" model (using only 3 core features) to find the best result.

2.1 Centroid-based clustering

2.1.1 Choice of k

We applied the following techniques for choosing the optimal value of k.

The Elbow Method

This method involves running the k-means algorithm for a chosen range of values of k. For each value of k, the Sum of Squared Errors (SSE) is calculated. The “elbow” in the plot of SSE versus k is considered as an indicator of the appropriate number of clusters.

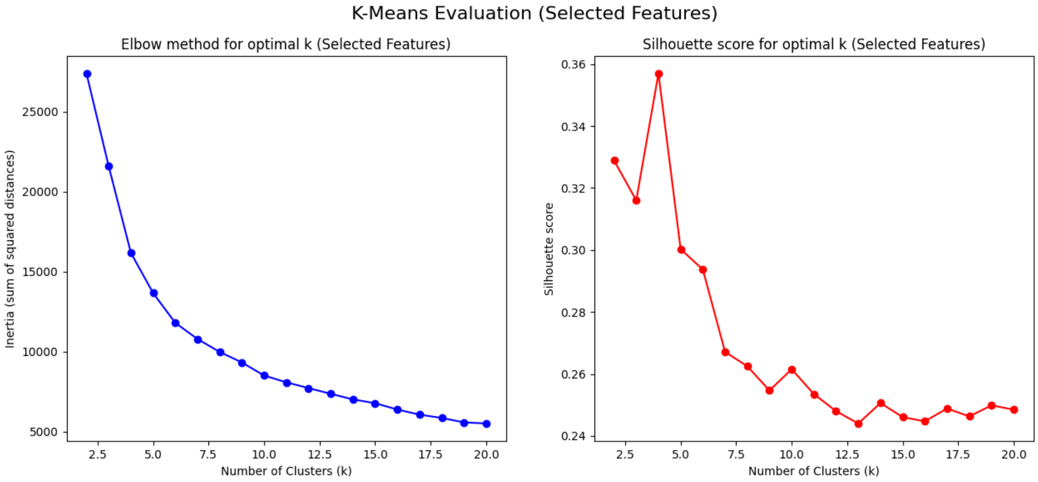


Figure 2.1: Elbow and Silhouette plots for the winning 3-Feature set.

The Silhouette Method

This method measures (range -1 to 1) how similar an object is to its own cluster compared to other clusters. A high value indicates that the object is well matched to its own cluster. We use the highest average silhouette score to select the best k.

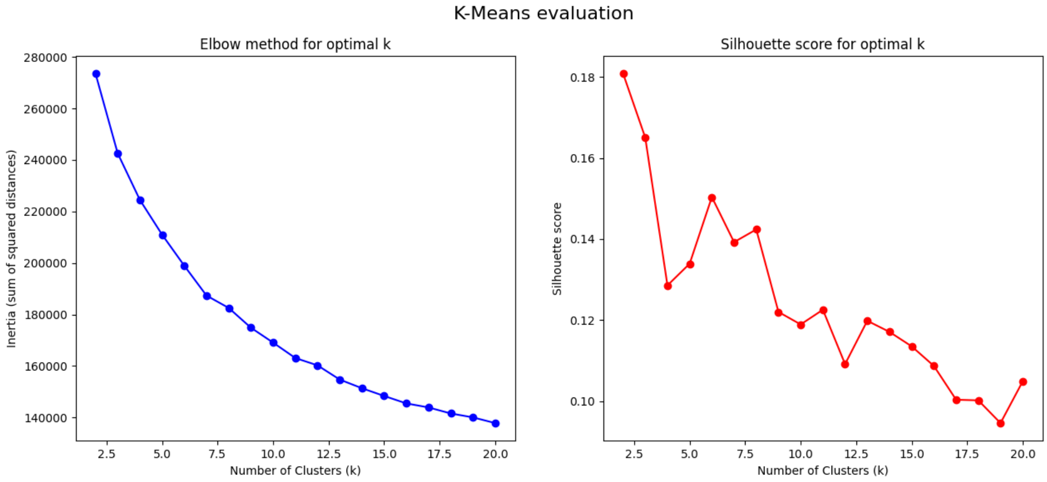


Figure 2.2: Elbow and Silhouette plots for the baseline 32-Feature set.

2.1.2 K-Means

K-Means clustering was performed twice to test the effect of feature selection.

Baseline Run: Full Features (32)

First, a baseline model was run using all 32 prepared features.

- **Best k:** 2
- **Silhouette Score:** 0.180

This score is very low, indicating that the clusters are poorly defined and overlap significantly. This is likely due to the "Curse of Dimensionality" caused by including noisy or binary features.

Experiment: Selected Features (3)

Next, we ran an experiment using only 3 core features identified through analysis: `['GameWeight', 'MfgPlaytime', 'NumOwned']`.

- **Best k:** 4
- **Silhouette Score:** 0.357

This result is a 98% improvement over the baseline, proving that feature selection was critical. The model identified 4 distinct, interpretable clusters, as seen in the centroid plot (Figure 2.3) and 3D scatter plot (Figure 2.4).

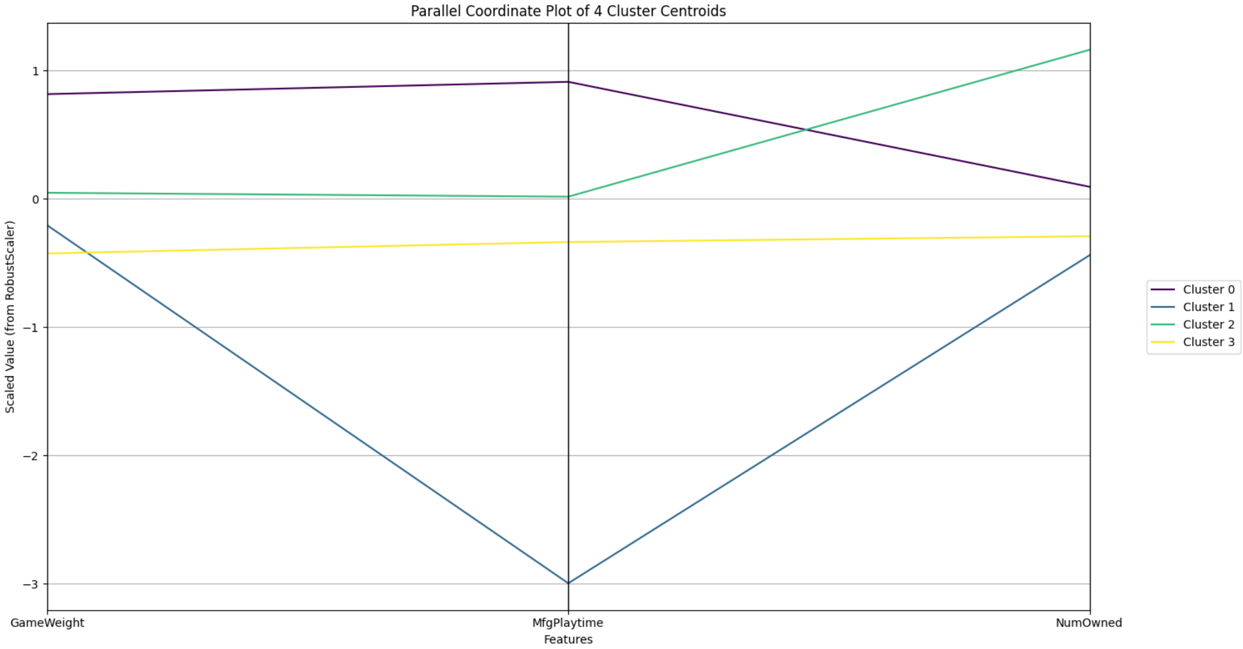


Figure 2.3: Parallel coordinate plot of our final K-Means centroids ($k = 4$).

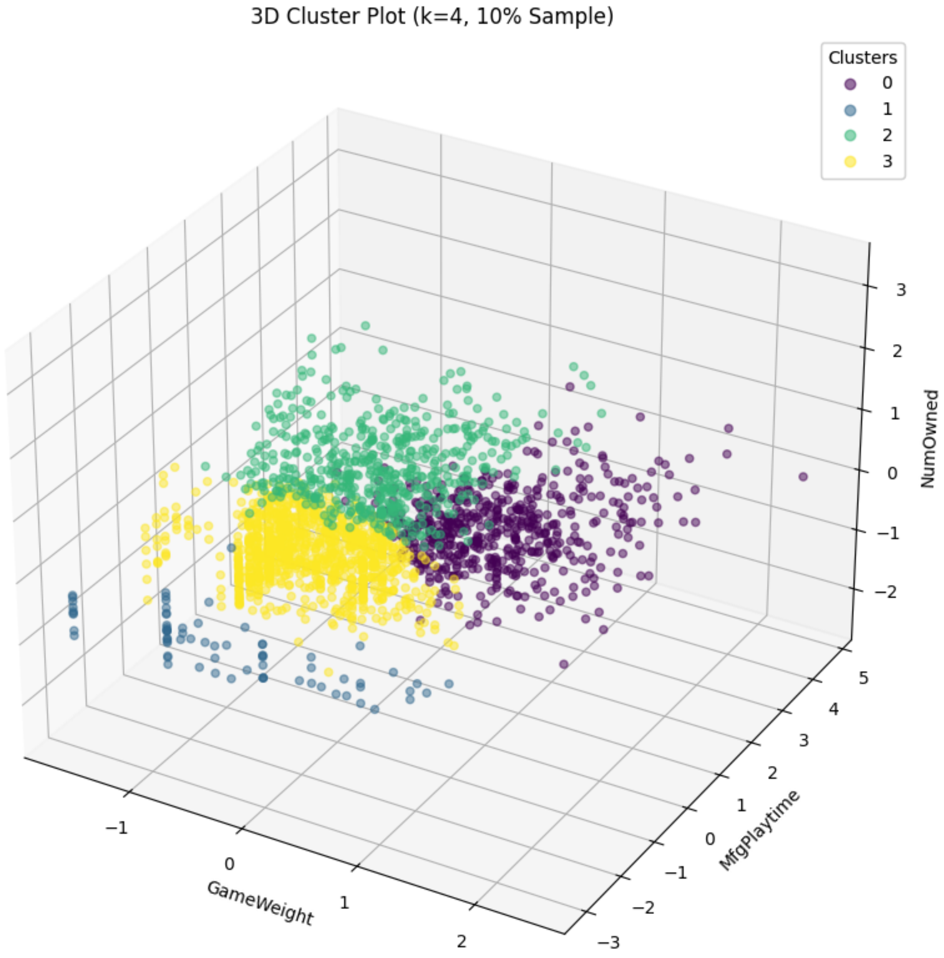


Figure 2.4: 3D Scatter Plot of the 4 clusters (on a 10% sample).

Based on the centroid plot, we can interpret our 4 clusters:

- **Cluster 0 (Heavy/Long Games):** This cluster (5674 games) is defined by a high `GameWeight` (0.81) and a high `MfgPlaytime` (0.91). This group represents the **’Heavy/Long Games’**—complex, strategic games that take a long time to play. Their `NumOwned` (0.09) is near the median, suggesting they are a large but specialized part of the market.

- **Cluster 1 (Quick Play Games):** This is a small, specialized cluster (836 games) defined by an extremely low `MfgPlaytime` (-2.99). This clearly represents the **’Quick**

Play' or 'Filler' Games. These are games that play very fast, and they have below-average complexity and ownership.

- **Cluster 2 (Popular Games):** This cluster (5034 games) is defined almost entirely by its very high `NumOwned` (1.16). Its `GameWeight` (0.04) and `MfgPlaytime` (0.01) are almost exactly at the median. This represents the **'Popular & Mainstream Games'**—games that are widely owned regardless of their complexity or length.

- **Cluster 3 (Light/Standard Games):** This is the largest cluster (10381 games) and represents the baseline **'Light/Standard Games'**. All its features are below the median: low `GameWeight` (-0.42), low `MfgPlaytime` (-0.33), and low `NumOwned` (-0.29). This group consists of the vast number of simpler, faster, and less-owned games in the dataset.

2.2 Density-based clustering

2.2.1 DBSCAN

We ran DBSCAN on the same 3-feature set to provide a fair comparison. To find the best parameters, we first generated a k-distance plot (Figure 2.5) to find the "elbow".

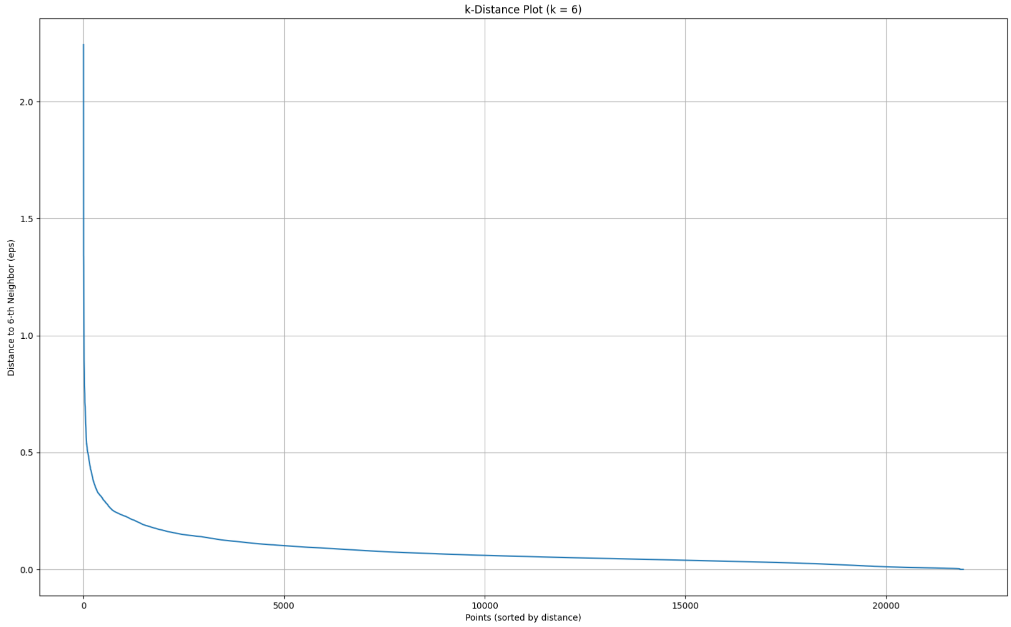


Figure 2.5: k-Distance plot ($k = 6$). The "elbow" was automatically found at $eps = 0.1456$.

We used the identified elbow value of `eps` $= 0.1456$ and `min_samples` $= 6$. The model failed completely:

- **Total Noise Points:** 1840 (8.39% of the data)
- **Silhouette Score (excl. noise):** -0.3221

The model produced 40 tiny micro-clusters and one giant "blob" cluster containing 17,961 points. The negative silhouette score confirms that the resulting clusters are worse than random. This definitively proves that our dataset does not have a density-based structure. Here we ran the script once to find what was the optimal eps=0.1456 and we rerun it.

2.3 Hierarchical clustering

Finally, we ran Agglomerative Hierarchical Clustering on our 3-feature set, setting `n_clusters`=4 for a direct comparison with K-Means.

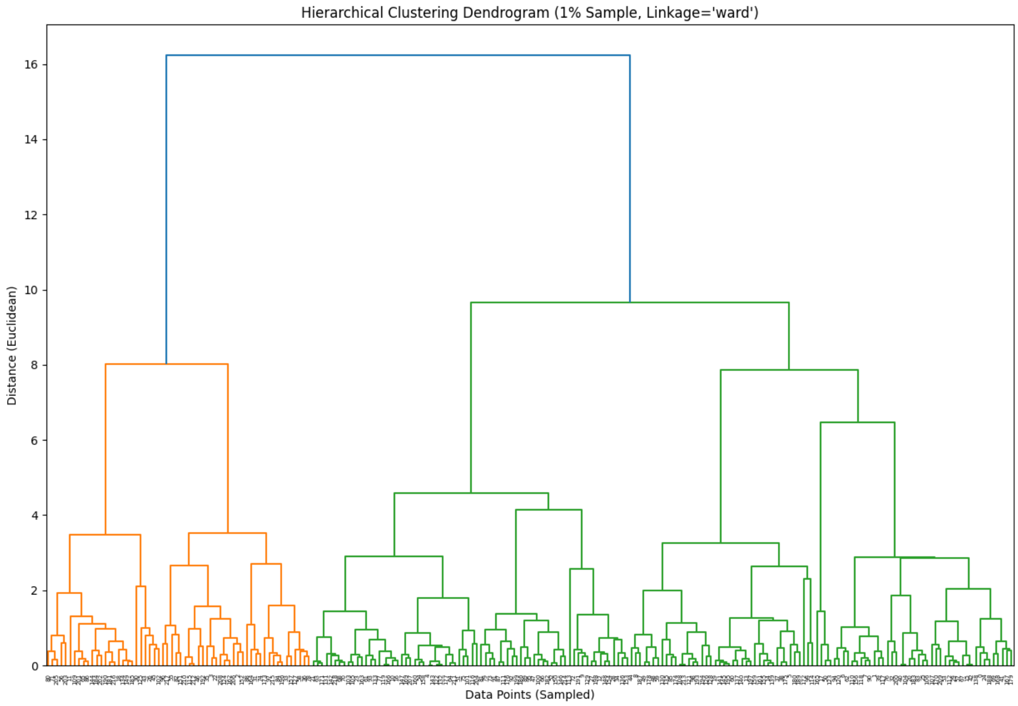


Figure 2.6: Dendrogram for Hierarchical Clustering (on a 1% sample). The tree is unbalanced and does not show 4 clear, distinct clusters.

The model produced a Silhouette Score of 0.3367. While this is a good score (and an 87% improvement on the baseline), it is still lower than our K-Means result but acceptable.

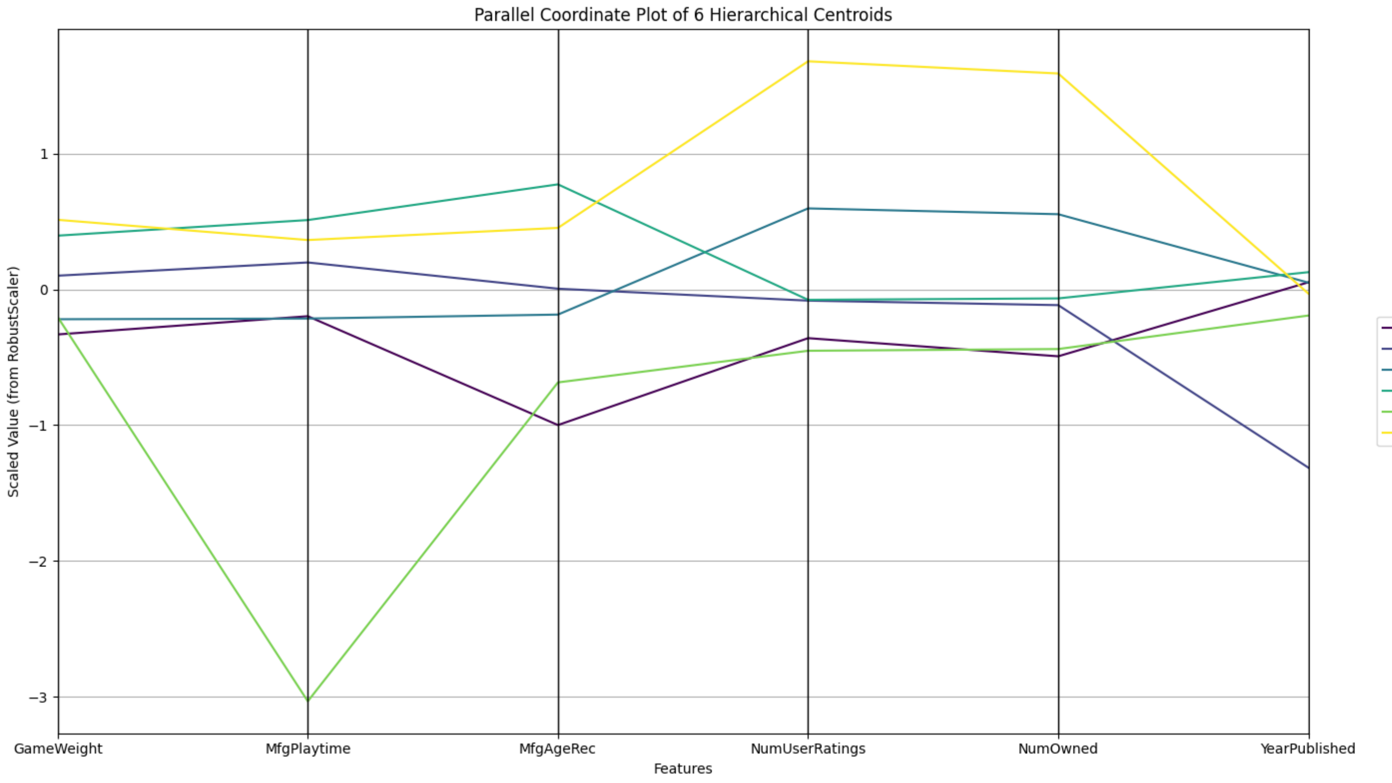


Figure 2.7: Centroid plot for the Hierarchical model. It attempts to find the same 4 groups as K-Means, but the cluster sizes are less balanced.

The cluster analysis (Figure 2.7) shows it found the same 4 archetypes as K-Means, but with very different sizes (e.g., the "Light Games" cluster has 12,348 points, and the "Quick Play" cluster only 791). This confirms that forcing a hierarchical structure onto the data is a worse fit than K-Means.

2.4 Final discussion

This analysis followed the three mandatory clustering methods. Our experiments provided a clear and definitive winner.

- **DBSCAN** was a total failure (Silhouette Score: -0.322), proving the data is not density-based.
- **Hierarchical Clustering** was a good runner-up (Silhouette Score: 0.337), but was ultimately outperformed.
- **K-Means** was the clear winner. By performing feature selection and reducing 32 noisy features to 3 core features, we improved our model quality by 98%, achieving a final Silhouette Score of 0.357.

We conclude that the best model for this dataset is K-Means with $k = 4$ applied to the `GameWeight`, `MfgPlaytime`, and `NumOwned` features.