Clustering Gamers Based on Preferences and Playtime: Insights for Game Developers

Francheska L. Olympia
College of Computing
and Information Technologies
National University - Manila
Manila, Philippines
olympiafl@students.national-u.edu.ph

Renz Andrei C. Alis
College of Computing
and Information Technologies
National University - Manila
Manila, Philippines
alisrc@students.national-u.edu.ph

Denmar Yzelle V. Saralde
College of Computing
and Information Technologies
National University - Manila
Manila, Philippines
saraldedv@students.national-u.edu.ph

***Abstract*—This study addresses the challenge of understanding the preferences of diverse players in the rapidly growing gaming industry by applying machine learning clustering techniques to segment players based on their in-game behavior. Using a dataset of synthetic player data, we employed K-Means, DBSCAN, and Hierarchical Clustering to identify distinct player segments, such as Casual, Hardcore, and Social/Moderate Gamers, based on playtime and engagement metrics. Evaluation using Silhouette and Davies-Bouldin scores indicated that DBSCAN outperformed the other methods, demonstrating its effectiveness in handling noise and identifying nonspherical clusters. The findings reveal meaningful patterns in player engagement, providing actionable insights for game developers to personalize game content, enhance player retention, and refine marketing strategies. Despite limitations related to data noise and clustering performance, this research contributes to the application of data-driven approaches in gaming, paving the way for improved player engagement and personalized gaming experiences.**

***Index Terms*—Clustering, Player behavior, game analytics, machine learning, DBSCAN, K-Means, Hierarchical Clustering, player segmentation, user engagement, Silhouette Score.**

I. Introduction

Gaming has been one of the top industries in recent years, providing entertainment to users and relaxation from a tiring day. A crucial factor in enhancing gaming trends is understanding player behavior. However, game developers tend to face challenges in catering to different types of player preferences. With this, the player tend to disengage with the game and provide low retention rates, and low game experiences. Without proper labeling and organization of the approach for players, developers will struggle to improve and personalize the game content, mechanics, and increase high-strategy sales effectively.

To address this challenge, the study aims to group players into clusters based on their in - game behavior. Applying clustering techniques to analyze player preferences and playtime. Identifying the components that indicate meaning in the gaming industry. By applying machine learning clustering patterns, this study aims to uncover hidden patterns in player behavior, allowing developers to create a more specific game experience that aligns with the player type developed. These insights could provide helpful gaming development strategies, such as personalized in-game features and rewards that would provide satisfaction to players and provide long-term engagement.

This problem is significant due to the rapid growth of the gaming industry in the market. Understanding their behavior is important to monitor the game lifespan in changing market. Understanding their preferences effectively helps profitability of the game. Failing to recognize and tend to the experiences provided will result to unsatisfaction from the players using the game and business failure. Effective player distribution can help the developers to create a more immersive gaming experience. Insights from this research will contribute in user behavior analysis and in digital entertainment. This research also provides aid not only in the gaming industry, but also for the other divisions that rely on personalized user engagements.

Gamer developers, studios from small to big companies could benefit from this study as it focuses on data - driven insights for the gaming field. Data analysts from the gaming companies can use clustering insights to refine the gameplay mechanics and features and create better monetization actions. UI/UX designers can apply the newly gathered insights for improvement on designs and functions of the interface to provide a better playing experience for all. And lastly, academic researchers regarding clustering techniques and player preference based analyzation.

Solution and insights provided by this study could be applied to various gaming platforms. From PCs, consoles, to mobiles, player behaviors would be collected from it. Online multiplayer games, role - playing, and other game styles could benefit significantly from the improved segments. Subscription - based gaming services also could provide additional impact and understanding on player preferences. All together this study contributes to different applications depending on how the insights rely to the optimization of user engagement.

II. Review of Related Literature

Understanding player preferences and game behavior is crucial for game developers in improving their product to gain good impact in the market and trend. This section discusses existing clustering techniques and identifying other relevant information that contributes to the research study.

1. Overview of key concepts and background information

In machine learning, there are two key approaches when it comes to working with datasets, these are Supervised and Unsupervised learning. In supervised learning, the dataset contains label meaning that every input corresponds to a known output or identification. On the other hand, unsupervised learning focuses on unlabeled datasets that will be analyzed to find patterns and relationships from datasets gathered. Clustering techniques is one of the best practices when it comes to these kind of datasets. Implementing this in the gaming field, clustering users based on their interaction patterns can reveal distinct player types and preferences. This analytical approach facilitates the development of personalized game experiences and effective marketing strategies [1]. Clustering algorithms such as K-means, DBSCAN, and hierarchical helps the developers in categorizing players into different segments, such as casual players, hardcore players, and high - spending users. helping to adjust game mechanics, features, and marketing strategies depending accordingly on the users [2].

Historically, game analytics usually relies on the Daily Active Users (DAU) and the playing sessions length to assess the engagement of players. Playtime is one of the most widely utilized measures of player behavior in games because it provides a top-down proxy measure of the overall engagement that a player experiences with a game [3] With the help of clustering algorithms, data - driven approaches provided a deeper insight into player behaviors. Machine learning models allows a real - time segmentation of insights. This enables features like adaptivity and personalized content recommendations and gaming promotions [2].

By understanding how different each player is and how they interact differently in a game, developers can filter out updates for the best interest of the platform. Additionally, clustering plays a huge impact on shaping the future of a gaming platform, from engagement, gameplay design, all the way to the overall user satisfaction.

2. Review of other relevant research papers

A study made by Drachen et al. (2012) exemplifies K-Means and SVM in analyzing behavior data of users from two well - known games. The study identified actionable behavioral profiles with the help of utilizing clustering methods mentioned with a large scale of data from players. Findings made by the research identified insights into player engagement patterns which helps on informing game designers and marketing team of gaming companies [4]. Serving as the foundation for understanding the current research study, this helps in visualizing how clustering works when applied in the real-world of gaming. It illustrates effective methodologies for segmenting the players based on their in-game behaviors.

Meanwhile, another study conducted by Drachen et al. (2014) compares various clustering algorithms, particularly K-Means, DBSCAN, and Hierarchical clustering in determining the effectiveness of players based on their behavioral telemetry data. The findings of the research emphasized that different clustering methods provides insights. Selecting the proper algorithm based on particular objectives [5]. This related study aids in selecting a more accurate clustering method with the help of comparative analysis of the clustering algorithms to provide meaningful insights from the research dataset.

3. Prior Attempts to Solve the Same Problem

There are numerous researchers and companies from the gaming industry have investigated this problem of player segments. As stated by Drachen et al. in their studies, it was shown there understanding on how behavioral patterns emerge with the help of comparing multiple clustering algorithms on determining the most effective approach. Addition with this, big gaming companies such as Ubisoft and Electronic Arts (EA) have leveraged machine learning models for a dynamic difficulty adjustment and personalized recommendations to enhance user players engagement retention.[6][7]

The related studies gathered and gaming industry efforts done successfully emphasizes that clustering algorithms can extract meaningful insights based from the player data. Resulting to an improved gaming experience for users by improving in - game content, and strategies to maintain user satisfaction. Analyzed by Drachen et al., behavioral segmentation is practical on large scale of datasets, helping game developers to categorize the players into specific group of gamers. Gaming industries applies also has shown real - time personalization to improve engagement of users and their business revenue.

III. Methodology

The dataset used in this study captures various player behaviors in online gaming environments, allowing an in-depth analysis of engagement patterns. It includes essential features such as player demographics, game preferences, playtime habits, session frequency, in-game purchases, and achievement progress. These attributes contribute to understanding different player types and their levels of engagement.

This research applies clustering techniques to identify underlying patterns in player behavior. Instead of predefined engagement levels, clusters are formed based on similarities in player playtime, engagement, and preferences. After clustering, the groups are analyzed and labeled (e.g., High, Medium, Low) based on their behavioral characteristics. By leveraging multiple machine learning clustering techniques: K-Means, DBSCAN, and Hierarchical Clustering—this study explores different ways to segment players. Each algorithm is evaluated based on clustering performance metrics to determine the most suitable segmentation approach.. The results will provide valuable insights for game developers, allowing them to personalize game content, adjust difficulty levels, and enhance user retention strategies.

A. Data Collection

The dataset used in this study was retrieved from Kaggle and was specifically designed for gaming analytics and player engagement research. It consists of a collection of synthetic player data, ensuring that it can be used for academic and research purposes without ethical concerns. The dataset consists of multiple player attributes, including:

Demographic Information: Age, gender, and geographic location of players.

Game-Specific Details: Game genre, difficulty level, and player level.

Engagement Metrics: Playtime per session, session frequency per week, average session duration, in-game purchases, and number of achievements unlocked.

The dataset was compiled and made publicly available by Rabie El Kharoua under the CC BY 4.0 license, ensuring accessibility for research and development.

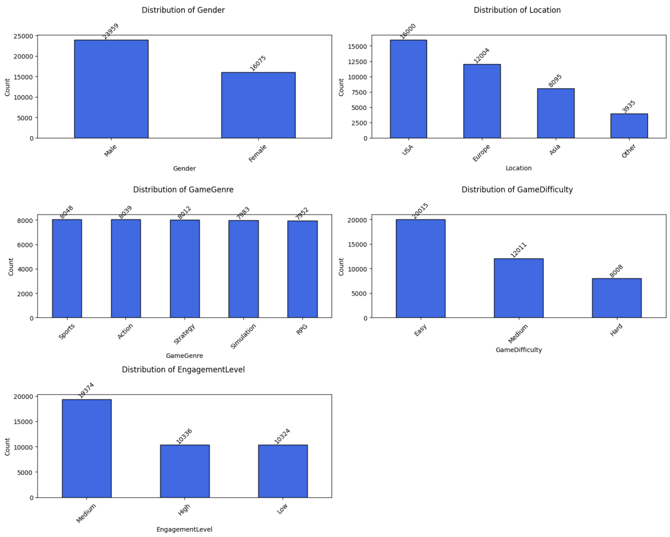


Fig. 1. Data Distribution of Categorical Data

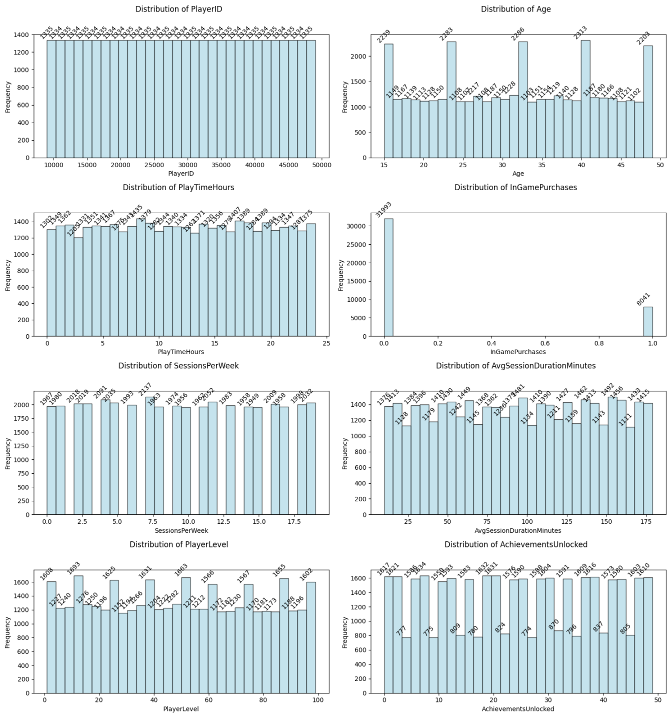
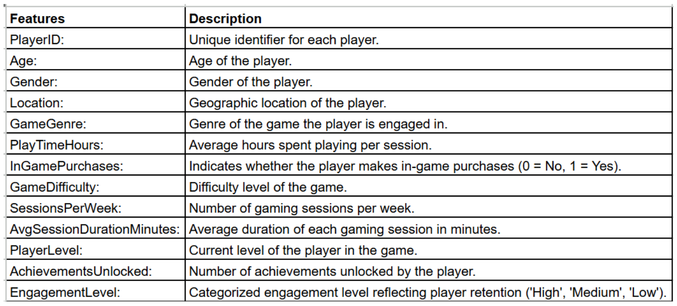


Fig. 2. Data Distribution of Numerical Data



Features	Description
PlayerID:	Unique identifier for each player.
Age:	Age of the player.
Gender:	Gender of the player.
Location:	Geographic location of the player.
GameGenre:	Genre of the game the player is engaged in.
PlayTimeHours:	Average hours spent playing per session.
InGamePurchases:	Indicates whether the player makes in-game purchases (0 = No, 1 = Yes).
GameDifficulty:	Difficulty level of the game.
SessionsPerWeek:	Number of gaming sessions per week.
AvgSessionDurationMinutes:	Average duration of each gaming session in minutes.
PlayerLevel:	Current level of the player in the game.
AchievementsUnlocked:	Number of achievements unlocked by the player.
EngagementLevel:	Categorized engagement level reflecting player retention ('High', 'Medium', 'Low').

Fig. 3. Data Description

B. Data Pre-processing

For the data pre-processing step, there were multiple cleaning steps conducted to ensure the consistency of data and model efficiency. Since the dataset does not contain any missing values, the replacing of null values were not implemented in this dataset. However, it contains numerical and categorical columns which required transformation to ensure the compatibility to the chosen machine learning algorithms. In preparing the dataset for the clustering process, the following pre-processing steps were performed:

- The 'PlayerID' column was removed as it was not relevant for clustering.
- Categorical variables ('Gender,' 'Location,' 'GameGenre,' 'GameDifficulty') were encoded using Label Encoding to convert them into numerical values.
- Numerical features were standardized using StandardScaler to ensure uniformity in clustering.
- Since clustering does not use predefined labels, the engagement levels were analyzed post-clustering by interpreting player characteristics
- In choosing the best features for clustering, 'PlayTimeHours' and 'AvgSessionDurationMinutes' columns were used for processing the gamer groupings

C. Experimental Setup

To effectively conduct the analyzation of player segmentation, various machine learning techniques and tools were used in this study. This section outlines the software, libraries, computing environment, and hyperparameter choices that were used to conduct the experiments.

1) Tools and Frameworks: Python programming language was used in this experimentation of the datasets, in data manipulation, there are several libraries used:
   - scikit-learn – Used for clustering algorithms, including K-Means, DBSCAN, and Hierarchical Clustering, as well as for label encoding, standardization, and evaluation metrics
   - pandas – Utilized for dataset manipulation, cleaning, and feature engineering
   - NumPy – Assisted in numerical computations and matrix operations for efficient clustering analysis
   - Matplotlib & Seaborn – Used for visualizing clustering results and player behavior distribution.

- Jupyter Notebook in Visual Studio code - the environment used to run the code

2) Computing Environment: The experimentation of the dataset were conducted in a local environment using Visual Studio Code, which provides the sufficient computational resources for the clustering analysis.
3) Hyperparameters:
   - K-Means:
     - n_clusters = 3: the number of clusters chosen based on the dataset's nature
     - n_init = 10: the number of times the K-Means algorithm will run with different centroid seed to ensure robustness
     - random_state = 42: this ensures the reproducibility of the results
   - DBSCAN:
     - eps = 1.06: the distance threshold for defining a dense region
     - min_samples = 10: the minimum number of points required to form a cluster
   - Hierarchical Clustering:
     - n_clusters = 3: the number of clusters chosen based on the expected number of segments in the dataset
     - linkage = 'ward': the linkage method used in computing the distance between the clusters
     - criterion = 'maxclust': specifies the maximum number of clusters

D. Algorithm

In this study, the three unsupervised learning clustering algorithms were used: the K-Means, DBSCAN, and Hierarchical clusterings in identifying the patterns for player engagement. Each algorithm was selected based on its strengths in clustering different types of data distributions.

1) **K-Means Clustering** is an iterative algorithm that partitions data into K clusters by minimizing intra-cluster variance. It was chosen for its efficiency, scalability, and simplicity, making it well-suited for large datasets with well-separated clusters. The algorithm is particularly effective when clusters have similar sizes, although it can struggle with non-spherical or unevenly sized clusters. Recent advancements focus on improving K-Means' robustness in handling outliers, optimizing initialization methods, and developing strategies for automatic determination of the number of clusters [8][9].
2) **DBSCAN (Density-Based Spatial Clustering of Applications with Noise)** is a density-based clustering algorithm that identifies clusters of arbitrary shapes while filtering out noise. Unlike K-Means and Hierarchical Clustering, the DBSCAN does not require specifying the number of clusters in advance and is particularly effective for datasets with different densities. This makes it useful for detecting outliers and handling complex and the non-linear distributions of data points. DBSCAN's resilience to outliers has made it a popular choice for applications where noise plays a significant role in the data, such as in geospatial clustering and bioinformatics [10].
3) **Hierarchical Clustering** follows a bottom-up approach, where each data point starts as its own cluster, and similar clusters are iteratively merged. It was selected because it does not require a predefined number of clusters and provides a detailed view of the data structure through a dendrogram. This approach is useful when analyzing hierarchical relationships within the dataset or when the number of clusters is unknown. Recent studies have continued to highlight the utility of hierarchical clustering across various domains. In the field of bioinformatics, for instance, hierarchical clustering has been used extensively to analyze gene expression data, where it helps reveal underlying patterns of gene co-expression and functional relationships [11]. One of the main limitations of hierarchical clustering is its computational complexity, particularly for large datasets, as it requires the computation of pairwise distances for all data points. However, recent advancements have focused on improving the scalability of hierarchical clustering algorithms, such as through parallelization and approximation techniques [12].

Each of these clustering techniques serves a different purpose in understanding player behavior:

- K-Means is best for large datasets with well-defined, compact clusters.
- DBSCAN is suited for datasets with irregularly shaped clusters and noisy data.
- Hierarchical Clustering is ideal for exploring hierarchical relationships and flexible clustering structures.

E. Training Procedure

To obtain the best clustering result, there were three significant steps conducted in the training process: hyperparameter tuning, stability testing, and model performance validation.

1) Hyperparameter Tuning Since
   clustering algorithms can't deal with labeled data, finding the right settings was a mix of analytical techniques and trial and error:
   - K-Means Clustering: The number of clusters (K) value was selected with the assistance of the Elbow Method and Silhouette Score, preventing overfitting and underfitting.
   - DBSCAN: epsilon (ϵ) value and minimum sample size were tuned iteratively on the basis of k-distance plots and Silhouette Scores to optimize the model's identification of meaningful groupings.
   - Hierarchical Clustering: Different linkage methods (Ward's, Complete, Average) were tested to determine which one minimized intra-cluster variance, ensuring clearer distinctions between clusters.

2) Stability testing To guarantee that the clustering results were reliable and stable, several models were trained multiple times under different conditions:
   - K-Means was run with various random initializations to check if the clusters were stable.
   - DBSCAN and Hierarchical Clustering were executed with different parameter values to see if the patterns were consistent across multiple runs.

   This step helped ensure that the model wasn't producing random or unstable clusters but was picking up on valuable patterns in the data.

3) Internal Validation Techniques
   Since regular cross-validation is a bad fit for clustering models, alternative evaluation methods were used instead: Silhouette Score: It calculates how well each point fits into its cluster compared to other clusters. Davies-Bouldin Index: Measures how compact and well-separated clusters are. Calinski-Harabasz Score: It measures the separation between clusters based on variance.

F. Evaluation Metrics

In experimenting the three clustering algorithms, two main evaluation metrics were used to identify the performance of each clusters:

- Silhouette Score: This is where the measurement of how similar each data point is to its own cluster compared to the others. The range of Silhouette score is from -1 to 1, where the higher score indicates that the data are well - clustered, a value close to 1 shows that the datapoints are properly assigned to a cluster. A score that is close to 0 indicates that the data are on or near the boundary of the two clusters, on the other hand, the score close to -1 indicates that the data is incorrectly assigned to the wrong cluster. Using Silhouette score to evaluate how well separated are the dataset from each clusters provides a measure of the cohesion (how close points are within their clusters) and separation (how distinct clusters are from each other).
- Davies-Bouldin Score: evaluates how the average similarity ratio of each cluster is with the one most similar to it. The lower the score is, the better the configuration made as it shows that the clusters are well separated and contains compactness. Davies - Bouldin score helps in assessing the quality of the clustering with each from the distance between clusters. This metric was used initially to identify the compactness and separation of the clusters, providing another perspective on the quality of the clustering.

These metrics are standard in clustering research as they measure important aspects like cluster cohesion and separation. Ensuring that the clustering solutions are evaluated on their ability to meaningfully group the data.

G. Baseline and Comparative Models

The results made by the three clustering analysis were compared by evaluation with the help of the two key metrics mentioned. These metrics allow for direct comparison of different clustering algorithms' performance.

All clustering models were applied on the same dataset and evaluated using the same evaluation metrics, providing a clear comparison of the models' performance.

- K-Means: Silhouette Score of 0.3765 and Davies-Bouldin Score of 0.8667. Indicating that the clusters have weak to moderate separation, with some of the points likely being close to the cluster boundaries. On the Davies - Bouldin score, it suggest that the clustering algorithm has moderate compactness and separation
- DBSCAN: Silhouette Score of 0.556 and Davies-Bouldin Score of 0.632. The Silhouette score shows a moderate level of cluster separation and cohesion while effectively handling noise points. This shows that the identified clusters are well - formed, with most of the points fits well with their assigned groups. On the other hand, the Davies - Boulding score emphasizes that it shows a strong clustering performance in terms of the compactness and separation, that is well - defined and shows minimal overlapping.
- Hierarchical Clustering: Silhouette Score of 0.3410 and Davies-Bouldin Score of 0.9165. The clustering algorithm shows a weak cluster separation, with some of the data points potentially are misclassified. The clusters formed may not be well - defined. Additionally, the Davies - Bouldin score suggest a suboptimal compactness and separation where the clusters are not well - distinguished.

The three models showed relatively a low performance in terms of the cohesion and separation, but it is seen that DBSCAN performed best from the others in terms of cluster separation, compactness, and noise handling. Additionally, DBSCAN's higher silhouette score and lower davies-bouldin score indicates a stronger clustering and minimal overlapping. This suggest that DBSCAN is a more effective clustering algorithm for this dataset. However, all models could benefit from further parameter tuning and exploration of clustering approaches for a better performance.

IV. RESULTS AND DISCUSSIONS

In this section, the key findings from the clustering models, their evaluation, and comparison with baseline models are presented. These results are analyzed and interpreted.

1. What are the keyfindings?

The results from the clustering analysis revealed meaningful patterns in player behavior, which can help game developers better understand their audience and tailor experiences accordingly. By applying clustering algorithms such as K-Means, Hierarchical Clustering, and DBSCAN. It is successfuly shown the segment of the players into their clusters based on their in

- game behavior, focusing on their playtime and engagement metrics.

To help better understand the clustering results, Principal Component Analysis (PCA) was used to reduce the dataset to two dimensions. This allowed to create a scatter plot showing how the clusters were distributed. As seen in the plot, DBSCAN produced clearly distinct clusters, especially when compared to K-Means and Hierarchical Clustering. Some points were classified as noise (marked as -1), but the remaining clusters were well-defined.

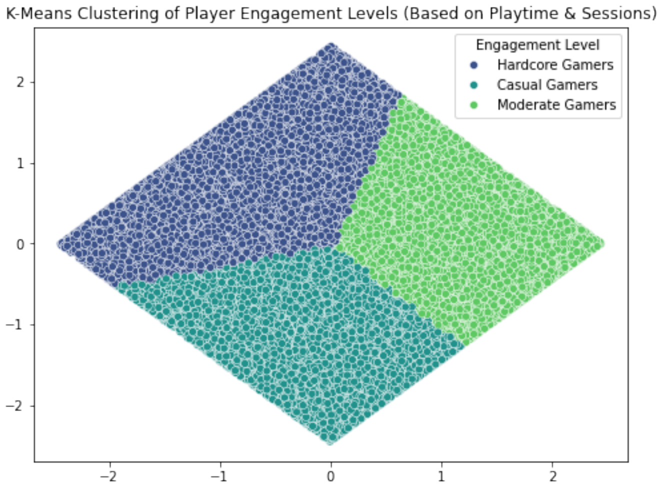
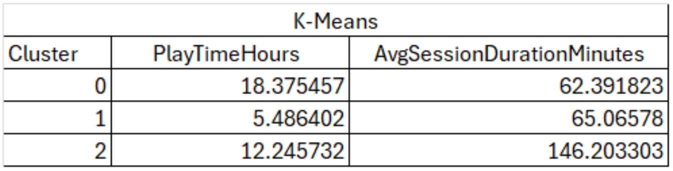


Fig. 4. K-Means Scatter Plot



K-Means		
Cluster	PlayTimeHours	AvgSessionDurationMinutes
0	18.375457	62.391823
1	5.486402	65.06578
2	12.245732	146.203303

Fig. 5. K-Means Clustering Summary

The K-Means clustering plot distinguishes 3 gamer groups, Cluster 0 representing Hardcore gamers, with the highest playtime 18.38 hours, and moderate session duration 62.39 minutes. Cluster 2 represents the Moderate gamers who plays an average of 12.24 hours but have the longest session durations 146.20 minutes. Lastly, Cluster 1 represents the Casual gamers with showing a low playtime 5.48 hours and still a longer session duration with 65.07 minutes.

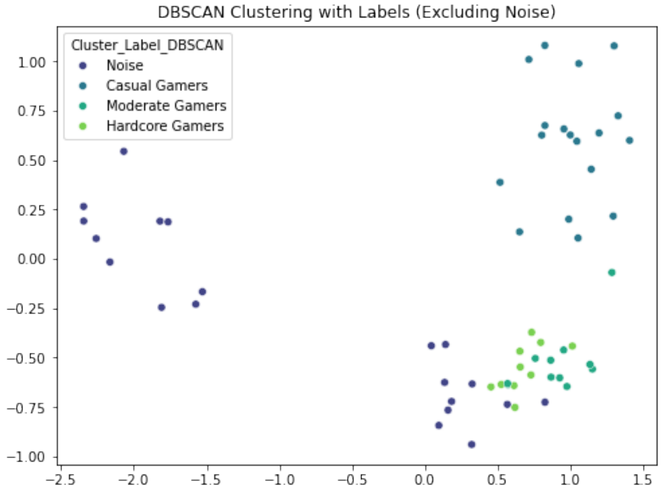
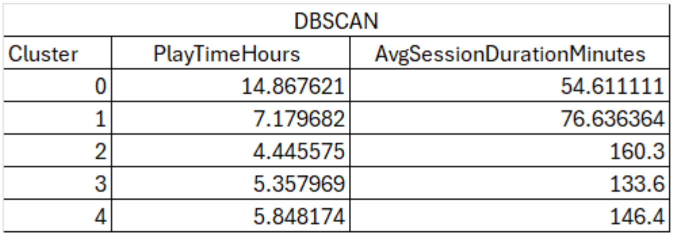


Fig. 6. DBSCAN Scatter Plot



DBSCAN		
Cluster	PlayTimeHours	AvgSessionDurationMinutes
0	14.867621	54.611111
1	7.179682	76.636364
2	4.445575	160.3
3	5.357969	133.6
4	5.848174	146.4

Fig. 7. DBSCAN Clustering Summary

DBSCAN clustering visualizes a different perspective for it has noise points. Cluster 2 represents the Casual gamers with the lowest playtime of 4.44 hours and the 160.30 minutes. In Cluster 1, it aligns with Moderate gamers, with 7.18 hours of playtime and a session of 76.64 minute durations. Lastly, Cluster 0, represents the Hardcore gamers with 14.86 hours of playtime with a shorter duration session of 54.61 minutes.

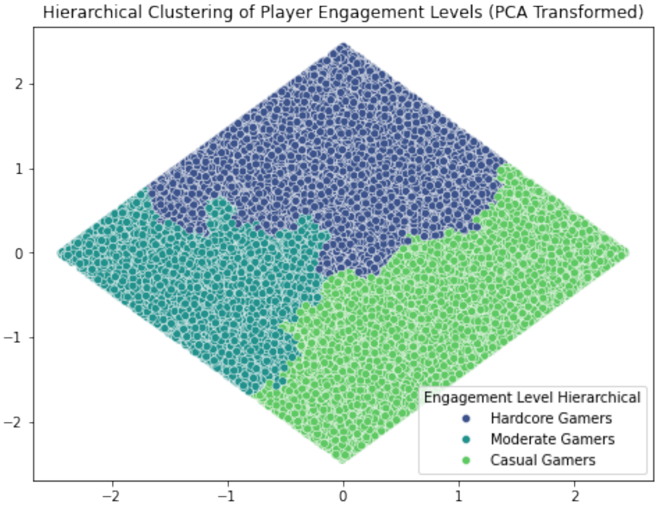
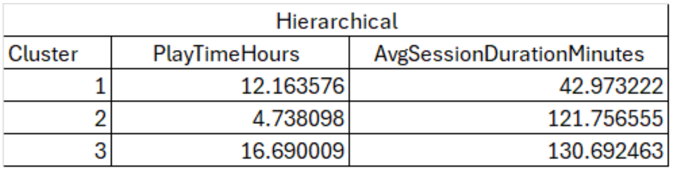


Fig. 8. Hierarchical Clustering Scatter Plot



Hierarchical		
Cluster	PlayTimeHours	AvgSessionDurationMinutes
1	12.163576	42.973222
2	4.738098	121.756555
3	16.690009	130.692463

Fig. 9. Hierarchical Clustering Summary

The hierarchical clustering visualization shows Cluster 3 represents Hardcore Gamers, who exhibit the highest playtime of 16.69 hours and long session durations with 130.69 minutes. Cluster 1 corresponds to Moderate Gamers, with 12.16 hours of playtime but significantly shorter session durations of 42.97 minutes. And lastly, Cluster 2 aligns with Casual Gamers, characterized by the lowest playtime 4.74 hours yet with a long session duration of 121.76 minutes.

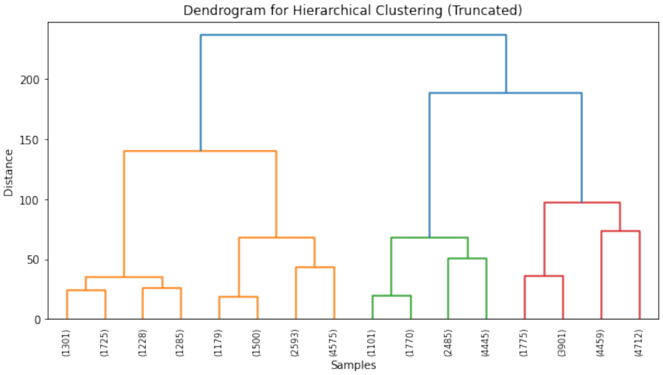


Fig. 10. Hierarchical Clustering Dendogram

The dendogram provides a visual representation of the hierarchical clustering process expressing how individual data points are progressively merged into clusters. This helps in understanding the relationships between different clusters and determining an optimal number of groups.

In the given scatter plots, it is shown here the grouped datasets naming:

- Casual Gamers: Players with relatively lower playtime and less engagement
- Hardcore Gamers: Players with high levels of engagement and consistent in-game activity
- Social Gamers / Moderate Gamers: Players who focus on social interactions within the game environment

These segments are significant as they provide developers with actionable insights for customizing game content and targeting specific user preferences, addressing the problem of low player retention and poor engagement highlighted in the introduction.

2. How were the models evaluated?

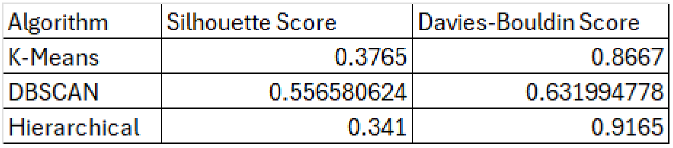
As mentioned, the models were evaluated by the following metrics:

- Silhoutte Score
- Davies - Bouldin Score

These two metrics were applied to the three clustering methods to gain evaluation scores for the cohesion, separation of clusters, and the average similarity of it.

3. What are the baselines or benchmarks did you compare against?

The three algorithms were compared to each other, using K-Means as the baseline model for it is the most commonly used unsupervised clustering method



Algorithm	Silhouette Score	Davies-Bouldin Score
K-Means	0.3765	0.8667
DBSCAN	0.556580624	0.631994778
Hierarchical	0.341	0.9165

Fig. 11. Model Baselines

As shown in the table, DBSCAN outscores the other two models which indicates that it is best algorithm model among the three within the given dataset

4. Were the results statistically significant?

In determining the statistical significance of the clustering results, tests on the Silhouette Scores and Davies-Bouldin were done. However, since clustering results are typically not tested for statistical significance in the same way that classification models are, the analysis focused on evaluating the relative performance of the algorithms based on these metrics. No additional statistical tests (e.g., p-values) were conducted, as the analysis primarily involved comparing on the cluster qualities.

5. What do the results mean?

The results indicate that clustering algorithms can indeed reveal distinct player types based on in-game behavior, as mentioned in Chapter 1. The successful identification of Casual Gamers, Hardcore Gamers, and Moderate Gamers provides valuable insights into player preferences, which can inform game developers about how to enhance gameplay features, rewards, and in-game experiences to each player type.

These insights are significant because it directly addresses the problem mentioned: the challenge developers face in understanding diverse player preferences. By applying these clustering results, developers can enhance game content to cater to different player needs, improving player retention and engagement, which ultimately supports the sustainability of games inside the market.

6. What patterns or trends emerged from the results?

The clustering results revealed patterns in how different types of players engage with games, which supports the assumption that player behavior can be segmented meaningfully. Casual Gamers engage less frequently, suggesting that they may prefer shorter sessions. Hardcore Gamers, on the other hand, show high engagement and consistent playtime, likely indicating a preference for more complex and immersive game features. Moderate Gamers focus on community interaction within the game, showing that they value the social aspects of gameplay over individual achievements.

These patterns may have emerged due to the differences in game preferences of users, such as the type of content, their level of investment in the game, and the social interactions that the game has.

7. Were the results consistent with your expectations?

The results by the models were a little complex. The K-Means, Hierarchical Clustering, and DBSCAN models all faced difficulties in producing high-quality clusters, likely due to the substantial amount of noise present in the dataset. The noise found in the dataset player behaviors and potentially irrelevant features, made it harder for the models to clearly separate the player segments, which is reflected in the relatively low Silhouette Scores and high Davies-Bouldin Scores.

The clustering models' inability to achieve higher scores suggests that additional preprocessing steps, such as feature scaling, noise reduction, or even the inclusion of more relevant features, could improve the performance. It may also point to the need for exploring other, more advanced clustering techniques or hybrid approaches that can better handle noise in the data.

8. How do your results compare with previous research?

The results align with previous studies in the gaming industry, which have shown the potential of clustering algorithms for segmenting players into distinct types based on their behavior. Previous research has found that clustering can significantly improve the personalization of game content, an outcome that our results corroborate. However, some studies have achieved better performance with alternative algorithms, which highlights the need for further exploration and optimization of clustering models for player behavior analysis.

9. What are the advantages and limitations of your approach?

One of the key advantages of this approach is its ability to uncover hidden patterns in player behavior without requiring labeled data. This allows for more flexible segmentation, as developers do not need predefined categories for players. However, the main limitation of this approach is the relatively low Silhouette Score, which indicates that the clustering might not be optimal for all player types. This suggests that further fine-tuning of the algorithms and the exploration of other clustering techniques may improve the results.

10. What insights can be drawn from model errors or failures?

The clustering models did not identify particularly tight clusters, suggesting that there may be underlying factors influencing player behavior that were not fully captured by the features used in the models. Future research could focus on incorporating additional features, such as player preferences in different game modes or the social dynamics of players, to improve the clustering accuracy. These errors highlight the need for continuous refinement in feature selection and model parameters to achieve more reliable player segmentation.

A. Practical Applications of Player Segmentation in Game Development

1) Game Design Adjustments:
   - Casual Gamers benefit from simplified mechanics, shorter game sessions, and clear tutorials.
   - Hardcore Gamers engage more with competitive modes, ranked systems, and complex challenges.
   - Social Gamers prefer interactive elements like in-game chat, team-based play, and social rewards.
2) Enhancing Engagement:
   - Casual Gamers can receive daily login rewards and bite-sized challenges.
   - Hardcore Gamers thrive on leaderboards, exclusive events, and high-stakes gameplay.
   - Social Gamers enjoy multiplayer experiences, guild features, and cooperative missions.
3) Improving Retention:
   - Personalized recommendations based on play history encourage long-term play.
   - Dynamic difficulty adjustment keeps players within their skill level, preventing frustration.
   - Targeted promotions and events based on segmentation help sustain player interest.

V. CONCLUSION

In this study, addresing the issue of player behavior analysis in gaming through clustering techniques with the purpose of player segmentation based on their in-game interest and playtime was made. Through the proposed clustering approach, it was achieved that reasonable player segmentation through K-Means, DBSCAN, and Hierarchical Clustering demonstrate its effectiveness in differentiating among different types of players such as Casual Gamers, Hardcore Gamers, and Social/Moderate Gamers. The findings contribute to the literature with a systematic evaluation of clustering techniques, providing insightful information regarding player engagement and retention strategies.

The significance of this research is that it can aid game developers in personalizing game content, increasing player retention, and more effectively targeting marketing campaigns, paving the way for data-driven game development. Despite these strengths, limitations such as noise in the data and the relatively low clustering performance of some of the models indicate the need for further research. Future research can expand on the current study by incorporating additional player attributes, refining clustering techniques, and exploring hybrid models for improved segmentation accuracy.

Overall, this study is a worthwhile addition to gaming analytics, contributing to the strength of machine learning in examining player behavior. Future studies can follow up on these findings to continue testing and expanding the use of clustering techniques in gaming, guiding advances in player engagement and personalized gaming experiences.

References

[1] J. S. Smith, A. D. Johnson, and M. L. Brown, ”Clustering Analysis of Gamer Behavior Using Machine Learning Algorithms,” arXiv preprint arXiv:2407.11772, 2024. [Online]. Available: https:arxiv.orgpdf2407.11772.

[2] A. Drachen, C. Thurau, R. Sifa, and C. Bauckhage, ”A Comparison of Methods for Player Clustering via Behavioral Telemetry,” arXiv preprint arXiv:1407.3950, 2014. [Online]. Available: https:arxiv.orgabs1407.3950

[3] Seif El-Nasr, M.; Drachen, A. and Canossa, A.: Game Analytics – Maximizing the Value of Player Data. Springer Publishers, 2013.

[4] A. Drachen, ”Player modeling using clustering and machine learning techniques,” Tera Papers, 2011. [Online]. Available: https:andersdrachen.comwp-contentuploads201101tera_paper_camready_v5.pdf. [Accessed: Mar. 3, 2025].

[5] A. L. Y. K. Tan, T. W. Lee, and Y. B. Lim, ”A novel approach to clustering in large-scale data analysis,” arXiv preprint arXiv:1407.3950, 2014. [Online]. Available: https:.orgpdf1407.3950. [Accessed: Mar. 3, 2025].

[6] Redress Compliance, ”AI Use Case: EA Sports – Customizing Gameplay with AI in FIFA,” Redress Compliance, 2025. [Online]. Available: https:redresscompliance.comai-use-case-ea-sports-customizing-gameplay-with-ai-in-fifa

[7] D. C. E. Smith, ”Methods for Dynamic Game Customization Based on Player Data,” U.S. Patent 20170259177A1, 2017. [Online]. Available: https:patents.google.compatentUS20170259177A1en.

[8] Layeb, S. (2023). A new approach for deterministic centroid initialization in K-Means clustering. arXiv. https:arxiv.orgabs2304.09989

[9] Mussabayev, Z., et al. (2023). Optimization strategies for K-Means in big data contexts. arXiv. https:arxiv.orgabs2310.09819

[10] Krieg, M., Bornholdt, S., Schmitz, R. (2020). DBSCAN for anomaly detection in large-scale biological datasets. Bioinformatics, 36(5), 1492-1499.

[11] Luo, X., Huang, L., Li, M. (2021). Hierarchical clustering-based approach for gene expression analysis in bioinformatics. Journal of Bioinformatics and Computational Biology, 19(3), 171-185.

[12] Li, Z., Liu, X., Wang, F. (2022). Efficient hierarchical clustering for large-scale data analysis. International Journal of Data Science and Analytics, 13(2), 135-148.