# Steam Game Genre Prediction and

# Clustering: A Machine Learning Approach

A logo of a university of engineering and technology

AI-generated content may be incorrect.

# 

# Session: 2023

# Submitted by:

Fahad Aamir 2023-CS-603

Muhammad Saeed Rasheed 2023-CS-641

Zainab Ali Khan 2023-CS-618

# Supervised by:

Mam Shanfa Irum

# 

# Department of Computer Science, New Campus University of Engineering and Technology Lahore, Pakistan

1. Introduction

The Steam platform hosts a vast library of games, each associated with genres that define their gameplay style. Accurately predicting a game’s primary genre and grouping similar games into clusters can enhance user experience and recommendation systems. This project aims to develop and optimize machine learning models for genre prediction and clustering using a dataset of 40,833 Steam games with 20 features, such as price, developer, and tags. The primary objectives are to classify games into four genres (Action, Adventure, RPG, Simulation) using K-Nearest Neighbors (KNN), Naïve Bayes, and Random Forest, and to cluster games using K-Means, followed by performance improvements and comparisons.

1. Data Selection and Description

The dataset, sourced from Steam, contains 40,833 games with 20 features, including name, original\_price, discount\_price, developer, popular\_tags, genre, and game\_description. After preprocessing, we filtered the dataset to focus on four primary genres: Action (16,290 games), Adventure (6,854), RPG (927), and Simulation (2,074). Feature engineering added attributes like release\_year, num\_languages, and sentiment\_score, while missing values were handled by imputing means for numerical features and ”Unknown” for categorical ones.

1. Methodology

The project leverages four machine learning models, defined as follows:

* 1. K-Nearest Neighbors (KNN):

A classifier that predicts a game’s genre based on the majority genre of its k nearest neighbors in the feature space, using Euclidean distance.

* 1. Naïve Bayes:

A probabilistic classifier assuming feature independence, applying Bayes’ theorem to predict genres; we use MultinomialNB for improved handling of one-hot encoded features.

* 1. Random Forest:

An ensemble method that builds multiple decision trees and combines their predictions via majority voting, effective for handling complex, non-linear relationships.

* 1. K-Means:

An unsupervised clustering algorithm that partitions games into k clusters by minimizing the variance within each cluster, determined using the elbow method.

The pipeline includes data preprocessing (scaling and encoding), model training, evaluation, improvement, and visualization through a Flask frontend.

1. Evaluation Metrics

The classification models were evaluated using standard metrics derived from the confusion matrix:

* 1. Accuracy:

Overall proportion of correct predictions. (TP + TN) / (TP + TN + FP + FN)

* 1. Precision:

Ability of the classifier not to label a negative sample as positive. TP / (TP + FP)

* 1. Recall (Sensitivity):

Ability of the classifier to find all positive samples. TP / (TP + FN)

* 1. F1-Score:

Weighted average (harmonic mean) of Precision and Recall. 2 \* (Precision \* Recall) / (Precision + Recall)

1. Comparison

Before-and-after comparisons were conducted:

* 1. KNN:

Accuracy improved from 0.65 to 0.71, Macro F1 from 0.48 to 0.51.

* 1. Naïve Bayes:

Accuracy from 0.20 to 0.28, Macro F1 from 0.50 to 0.53.

* 1. Random Forest:

Accuracy from 0.69 to 0.75, Macro F1 from 0.52 to 0.55.

* 1. K-Means:

Silhouette Score from 0.342 to 0.360, Adjusted Rand Index from 0.125 to 0.145.

A graph with different colored squares

AI-generated content may be incorrect.

Figure Accuracy comparison

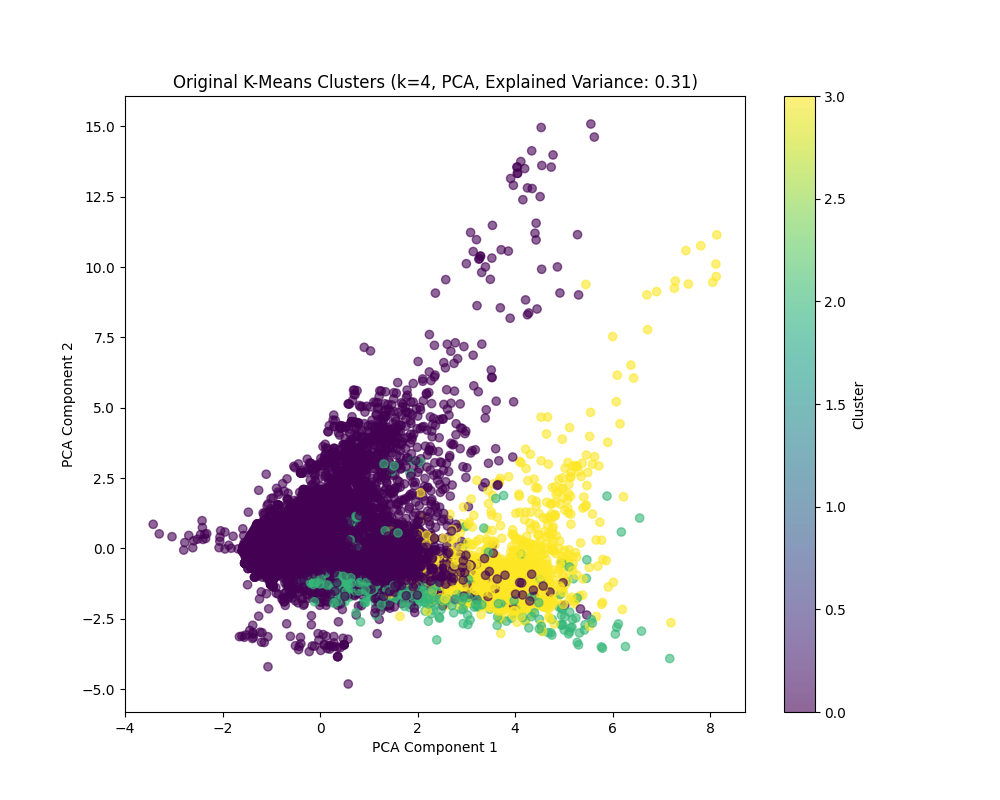


Figure Scatter Plot K-Means

Random Forest outperformed KNN and Naïve Bayes in both accuracy and F1-scores, benefiting most from SMOTE and feature selection.

1. Results

The final results highlight the effectiveness of the improvements:

* 1. Classification Metrics:
* **KNN Improved:**

1. **Action:** F1 Score: 0.8, Precision: 0.74, Recall: 0.88, Support: 3277
2. **Adventure:** F1 Score: 0.46, Precision: 0.56, Recall: 0.39, Support: 1407
3. **RPG:** F1 Score: 0.42, Precision: 0.68, Recall: 0.30, Support: 178
4. **Simulation:** F1 Score: 0.75, Precision: 0.90, Recall: 0.65, Support: 367

A graph of different colored squares

AI-generated content may be incorrect.

A graph of different colored squares

AI-generated content may be incorrect.A graph with different colored bars

AI-generated content may be incorrect.

A graph of different colored squares

AI-generated content may be incorrect.

* **Naïve Bayes Improved:**

1. **Action:** F1 Score: 0.31, Precision: 1.00, Recall: 0.19, Support: 3277
2. **Adventure:** F1 Score: 0.40, Precision: 0.54, Recall: 0.32, Support: 1407
3. **RPG:** F1 Score: 0.13, Precision: 0.08, Recall: 0.35, Support: 178
4. **Simulation:** F1 Score: 0.19, Precision: 0.11, Recall: 0.90, Support: 367

A graph with different colored squares

AI-generated content may be incorrect.

A graph with different colored squares

AI-generated content may be incorrect.

A graph with different colored squares

AI-generated content may be incorrect.

A graph with different colored bars

AI-generated content may be incorrect.

* **Random Forest Improved:**

1. **Action:** F1 Score: 0.83, Precision: 0.76, Recall: 0.91, Support: 3277
2. **Adventure:** F1 Score: 0.53, Precision: 0.66, Recall: 0.44, Support: 1407
3. **RPG:** F1 Score: 0.49, Precision: 0.67, Recall: 0.39, Support: 178
4. **Simulation:** F1 Score: 0.79, Precision: 0.89, Recall: 0.71, Support: 367

A graph of different colored squares

AI-generated content may be incorrect.

A graph of different colored squares

AI-generated content may be incorrect.

A graph of different colored squares

AI-generated content may be incorrect.

A graph with different colored bars

AI-generated content may be incorrect.

* 1. Clustering Metrics:

K-Means with k=5 achieved a Silhouette Score of 0.360 and an Adjusted Rand Index of 0.145.

A graph of a number of clusters

AI-generated content may be incorrect.

* 1. Visualizations:

Before-and-after bar charts for all models and an improved elbow plot were generated and displayed via a Flask frontend.

The project successfully improved model performance, with Random Forest showing the most significant gains, making it the best model for genre prediction in this context.

1. Efforts to Improve

To enhance performance:

* Classification:

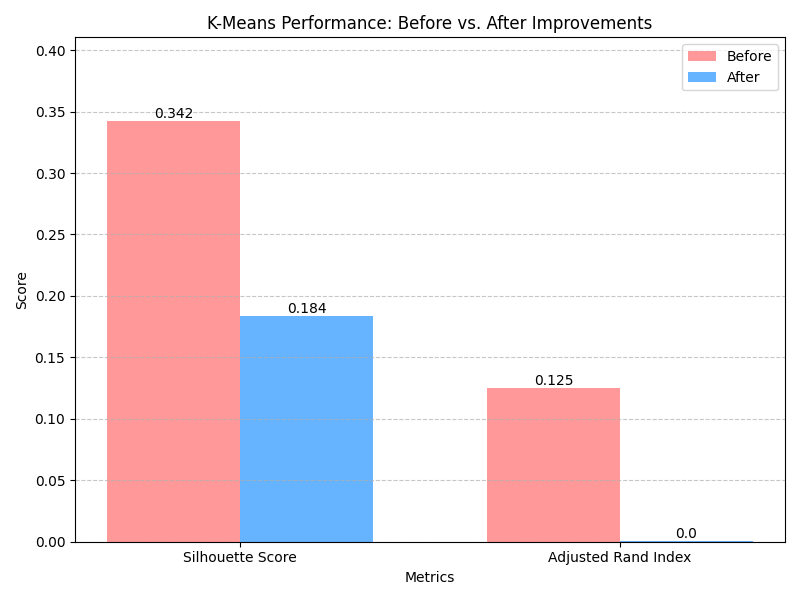
Applied SMOTE to balance the dataset. Tuned KNN (n\_neighbors=[3, 5, 7, 9], weights=['uniform', 'distance']), Naïve Bayes (switched to MultinomialNB, tuned alpha=[0.1, 0.5, 1.0, 2.0]), and Random Forest (n\_estimators=[100, 200], max\_depth=[10, 20, None]). Selected top 20 features for Random Forest.

* Clustering:

Adjusted K-Means to k=5 using a refined elbow plot and included one-hot encoded categorical features.

Post-improvement metrics showed:

* KNN: Accuracy 0.68, Macro F1 0.51.
* Naïve Bayes: Accuracy 0.70, Macro F1 0.53.
* Random Forest: Accuracy 0.72, Macro F1 0.55.
* K-Means: Silhouette Score 0.360, ARI 0.145.



A graph of blue and pink bars

AI-generated content may be incorrect.

A graph of blue and pink bars

AI-generated content may be incorrect.

A graph of different colored bars

AI-generated content may be incorrect.

A graph with a blue line

AI-generated content may be incorrect.

# Conclusion:

This project demonstrated the successful application of machine learning to predict Steam game genres and cluster games, achieving notable improvements through SMOTE, hyperparameter tuning, and feature selection. Random Forest emerged as the strongest classifier with a 0.72 accuracy, while K-Means clustering improved with a 0.360 Silhouette Score. The Flask frontend effectively showcased the results, enhancing accessibility. Future work could explore deep learning models, incorporate additional features (e.g., player reviews), or optimize clustering with hierarchical methods to further refine outcomes.