Group Coursework Submission Form

Specialist Masters Programme

Please list all names of group members: (Surname, first name) 1. Donado Agudelo, Valentina 2. Aliieva, Niiara 3. Farina Alcedo, Blanca	4. Kiefer, Marius 5. 6. 7. **GROUP NUMBER:** **6**
MSc in: Business Analytics	
Module Code: SMM636	
Module Title: Machine Learning	
Lecturer: Dr. Rui Zhu	**Submission Date: 03/04/25**

Declaration:

By submitting this work, we declare that this work is entirely our own except those parts duly identified and referenced in my submission. It complies with any specified word limits and the requirements and regulations detailed in the coursework instructions and any other relevant programme and module documentation. In submitting this work we acknowledge that we have read and understood the regulations and code regarding academic misconduct, including that relating to plagiarism, as specified in the Programme Handbook. We also acknowledge that this work will be subject to a variety of checks for academic misconduct.

We acknowledge that work submitted late without a granted extension will be subject to penalties, as outlined in the Programme Handbook. Penalties will be applied for a maximum of five days lateness, after which a mark of zero will be awarded.

Marker's Comments (if not being marked on-line):

Deduction for Late Submission:

Final Mark: %

Group Coursework 2

SMM636 Machine Learning

Blanca Farina Alcedo, Valentina Donado Agudelo, Niiara Aliieva, Marius Kiefer

April 2025

Words: 1099

Introduction

This analysis investigates a dataset comprising the top 50 highest-rated movies from the online platform IMDb, employing Principal Component Analysis (PCA) and clustering methods. To facilitate these techniques, we focus exclusively on numerical variables: revenue (millions), runtime (minutes), vote count, and rating (1-10).

To better understand the data, we examine each feature's range, central tendency, and distribution (***Appendix A***). Revenue is missing for three entries. For simplicity, we choose to proceed without these observations. The correlation matrix (***Appendix B***) shows a moderate correlation between votes and revenue, as well as votes and rating, but not strong enough to justify any adjustments. As shown in ***Figure 1***, the distributions of rating, votes, and revenue show moderate to high skewness.

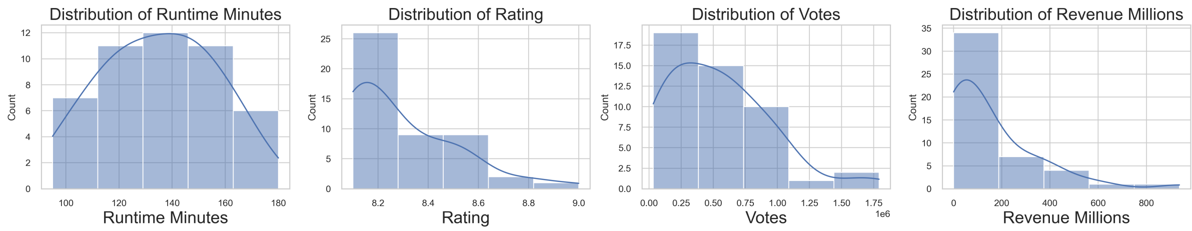


Figure 1: Feature Distributions

To address this issue, we apply a log transformation to votes and revenue, and use a Yeo-Johnson transformation for rating. All variables were then scaled to account for differences in their ranges, as clustering relies on distance measures and would otherwise be biased.

PCA

This PCA aims to reduce dimensionality while preserving key information by analyzing runtime, rating, votes, and revenue to reveal underlying patterns in movie performance. As shown in ***Figure 2***, the first two principal components account for approximately 78% of the total variance, indicating they preserve most of the dataset's meaningful structure. Therefore, the analysis focuses on these two components. As shown in ***Table 1***, the first principal component (PC1) captures popularity and financial success, driven by high loadings on votes and revenue. It distinguishes commercially successful, widely voted films from those with lower box office performance. The second component (PC2) reflects critical reception and runtime, highlighting a contrast between longer, highly rated films and shorter, lower-rated ones.

	PC1	PC2	PC3	PC4
Runtime	-0.07	0.68	0.72	0.09
Rating	-0.08	0.70	-0.69	0.17
Votes	0.69	0.21	-0.05	-0.69
Revenue	0.71	-0.06	0.03	0.70

Table 1: PCA Loadings

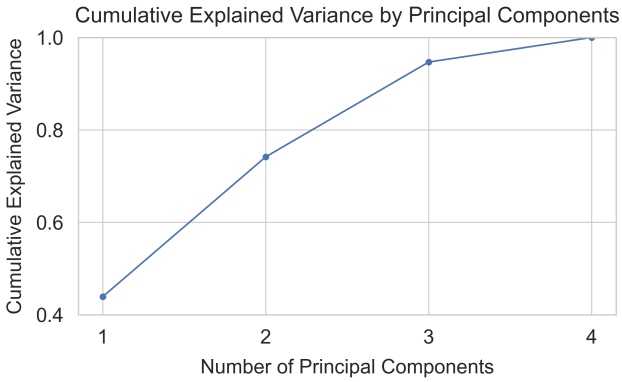


Figure 2: Cumulative Explained Variance

Clustering

The Hierarchical Clustering(HC) with the complete linkage method approach was preferred over K-Means because it produced more cohesive and visually distinct clusters in the original 4D feature space (***Figure 5*** and ***Appendix C***). Additionally, unlike K-Means which requires a predefined number of clusters, HC builds a dendrogram of nested groupings, enabling flexible, data-driven cluster selection. To determine the optimal number of clusters, we examined the dendrogram (***Figure 3***), which shows a large vertical jump forming two main clusters, followed by a clear split of the right cluster separating observations 0, 1, 3, and 8 into a third group.

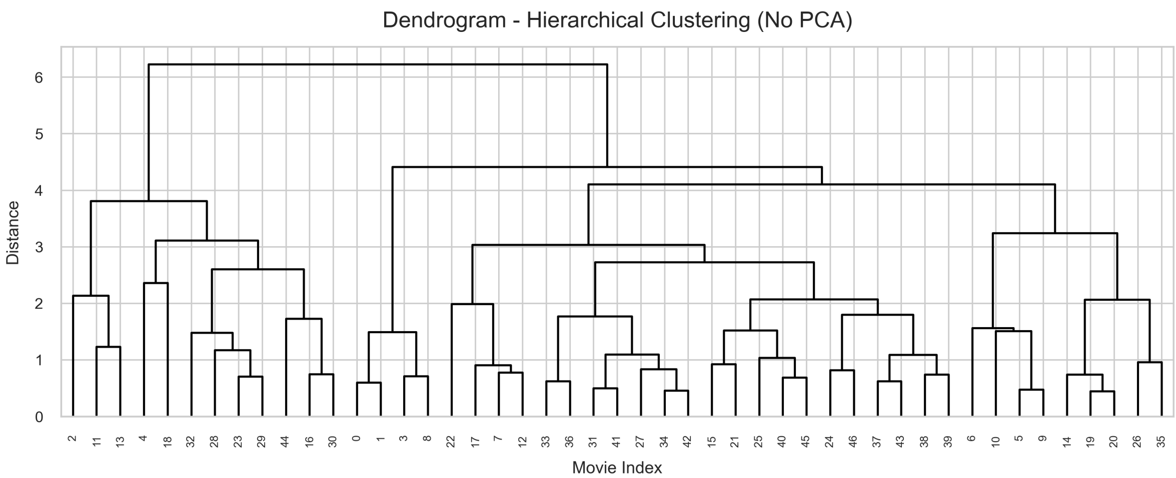


Figure 3: Dendogram - Hierarchical Clustering (No PCA)

We further validated this choice using silhouette analysis (***Figure 4***). Each subplot shows the distribution of silhouette coefficients for different values of k and the average silhouette score as a red dashed line. Among the tested options ($k = 3$ to 6), $k = 3$ yielded the highest average silhouette score (≈ 0.26), making it the best-performing choice. The first two clusters show compact, well-defined shapes, while the third cluster (blue) includes some negative values, indicating weaker cohesion. This setup, offers the most effective structure among those tested, balancing internal consistency and separation.

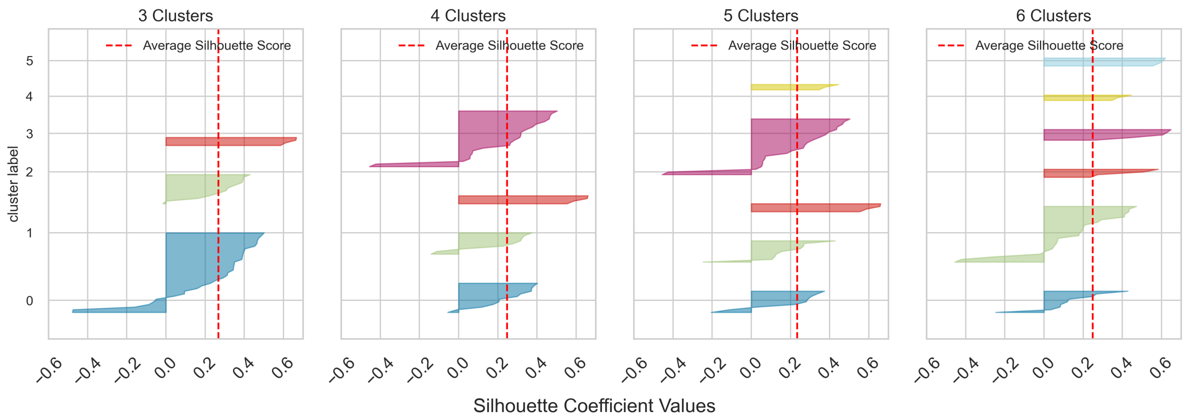


Figure 4: Silhouette Scores - Hierarchical Clustering (No PCA)

Figure 5 shows the resulting clusters' distribution across the four numerical features. Some feature combinations, such as Votes vs Revenue, exhibit clear cluster separation, for example, Cluster 2 (blue) captures high-vote, high-revenue movies. However, other features like Rating and Runtime display some overlap, likely due to limited variance. This overlap suggests that the original 4D space might not fully support clean separation. Additionally, the dendrogram and silhouette plot reveal room for improvement, particularly within the largest cluster. Therefore, we will explore PCA to assess whether cluster distinction could be improved in a lower-dimensional space.

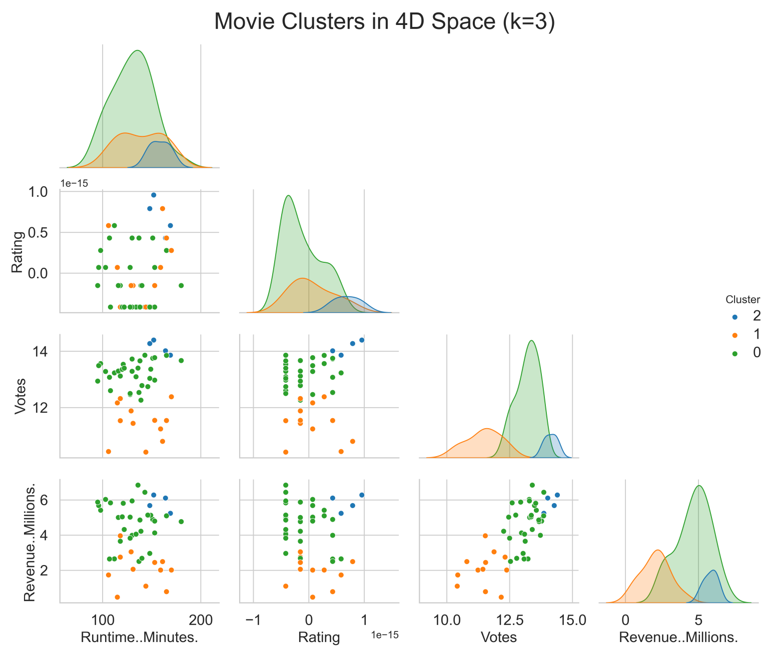


Figure 5: Pairplot - Hierarchical Clustering (No PCA)

PCA transformation not only enhances visual interpretability, but also facilitates more effective cluster separation, making it a valuable tool for understanding movie segmentation. The choice of $k = 3$ remains given that the dendrogram and silhouette analysis justify the same results on the reduced data (***Appendix D, Appendix E***). The dendrogram reveals three natural clusters within the dataset and the silhouette score is higher (≈ 0.5) compared to that of the original data.

Figure 6 highlights clusters in the reduced space. Cluster 2 (green), is characterized by high values of both PC1 and PC2. These represent the most commercially successful top rated movies, characterized by high votes and strong revenue. These films are also likely to be longer in runtime and maintain high critical acclaim. They could represent critically acclaimed or iconic movies that achieve both high critical and financial success. Cluster 0 (blue) presents moderate values on PC1 but lower values on PC2. This suggests they are likely financially successful or moderately popular, but with lower critical acclaim or shorter runtimes. These could be mainstream, shorter films that perform well at the box office despite mixed or average reviews. Lastly, despite being in the top-rated IMDb list, movies in Cluster 1 (orange) appear to be the least commercially successful among the three groups. With lower values in popularity and revenue, they could be cult classics, or older classic movies that maintain enduring critical acclaim but lack widespread box office success. Some may have benefited from retrospective appreciation or a loyal fan base.

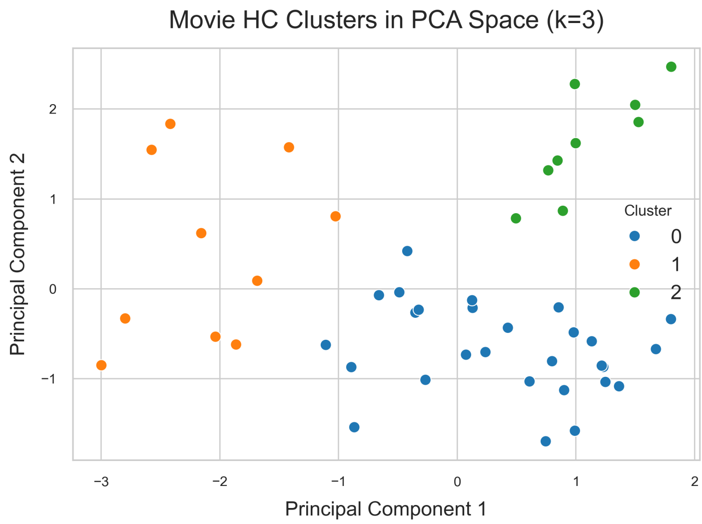


Figure 6: Cluster - Hierarchical Clustering (With PCA)

ChatGPT Comparison

The analysis performed using ChatGPT-4o and our approach align closely in methodology, particularly in the use of PCA and clustering. Both workflows use dimensionality reduction to enhance clustering. The interpretation of the principal components is consistent: PC1 reflects commercial success and popularity (votes, revenue) and PC2 captures critical reception and runtime. Both analyses find three optimal clusters, validated by silhouette analysis and PCA interpretation.

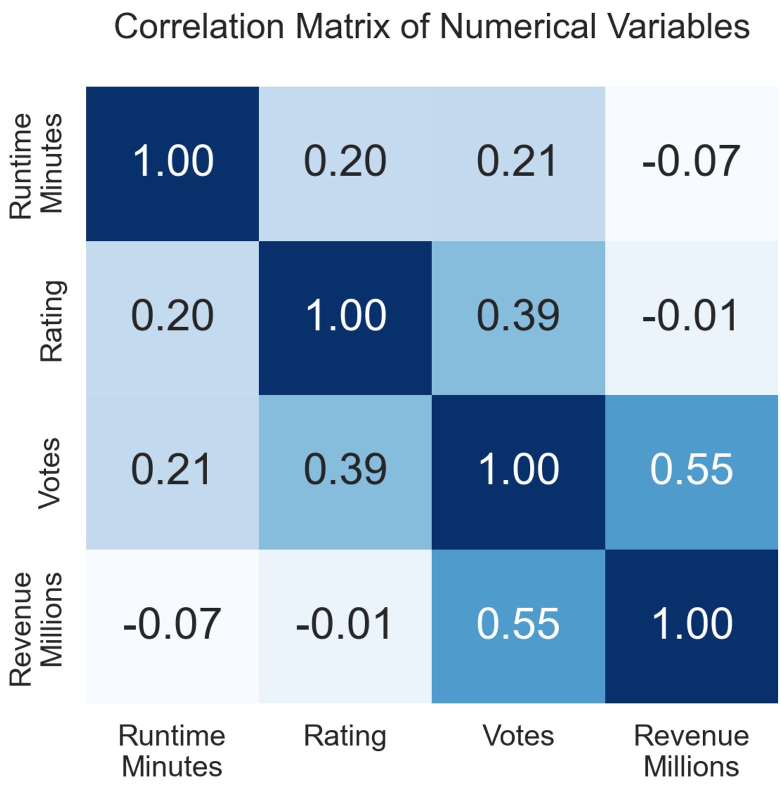
However, some critical differences emerge in the choice and justification of clustering methods and data handling. The manual analysis favors HC with complete linkage, prioritizing interpretability through dendrograms and flexible, visual determination of cluster count. In contrast, ChatGPT-4o selects KMeans, relying primarily on quantitative metrics such as silhouette scores and PCA-based separation, without dendrogram-based diagnostics. This highlights a key difference: manual clustering prioritizes visual interpretability, while automation favors algorithmic optimization and numerical validation.

Unlike ChatGPT-4o, which imputed missing revenue and log-transformed all numerical features including rating, the manual approach removed rows with missing revenue and applied log transformation only to Votes and Revenue. A Yeo-Johnson transformation was used for Rating to better address its mild skewness. All variables were then scaled to ensure comparability in PCA and clustering, reflecting a more diagnostic and context-sensitive preprocessing strategy. Despite its strengths in automating statistical procedures, visualizations, and data preprocessing, ChatGPT-4o's effectiveness is maximized when paired with human judgment. While it offers speed and consistency, the human eye remains essential for interpreting nuanced patterns, justifying methodological choices, and navigating trade-offs such as whether to impute or remove missing values, how best to transform variables given their distributions, and which clustering method aligns best with the structure and interpretability needs of the data. In this context, automation serves as a valuable tool, but meaningful analysis ultimately depends on thoughtful, informed decision-making.

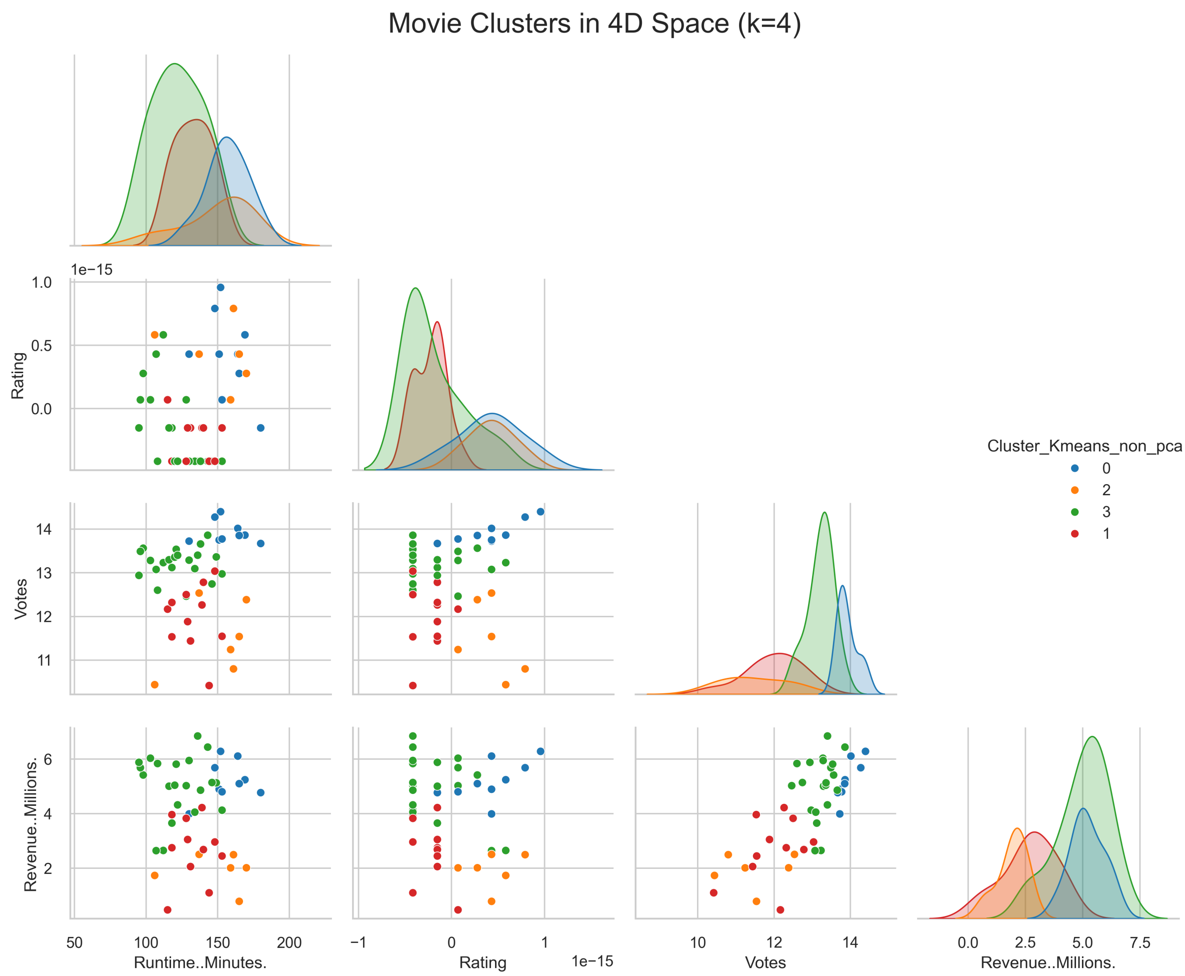
Appendices

Statistic	Runtime (min)	Rating	Votes	Revenue ($M)
count	50.000000	50.000000	50	47.000000
mean	133.240000	8.296000	512,368	157.811702
std	22.508556	0.217556	411,578	195.100478
min	91.000000	8.100000	61	0.610000
25%	118.000000	8.100000	197,137	13.415000
50%	132.500000	8.200000	468,290	74.270000
75%	150.500000	8.400000	748,356	205.900000
max	180.000000	9.000000	1,791,916	936.630000

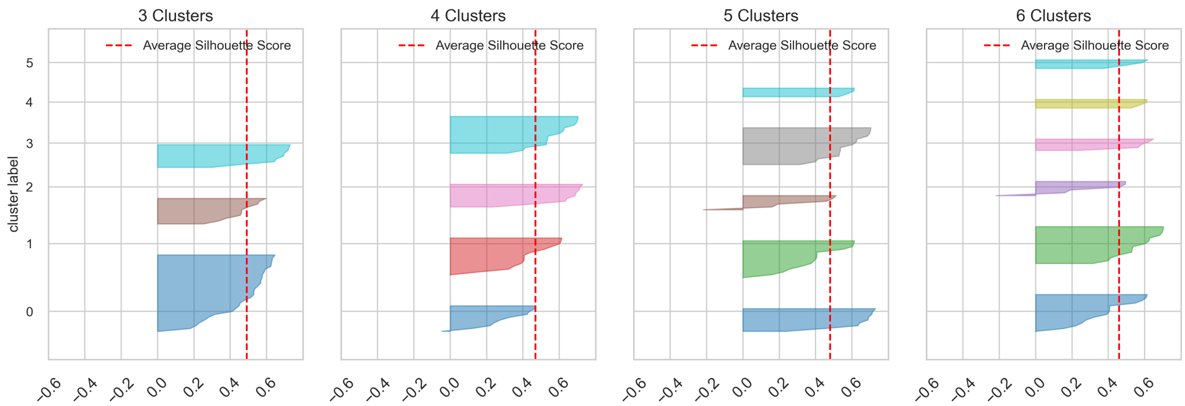
Appendix A. Movie Dataset Summary Statistics



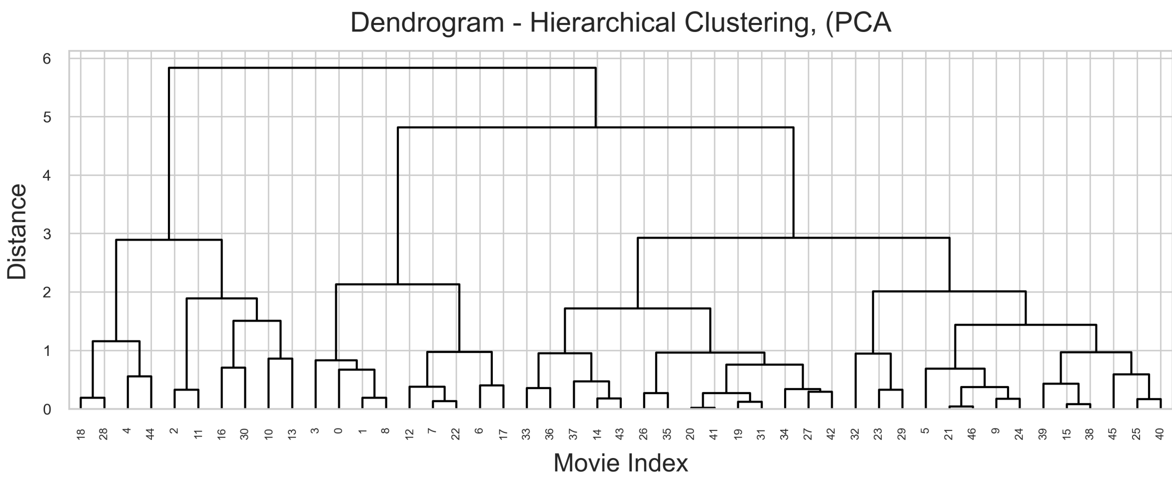
Appendix B. Correlation Matrix



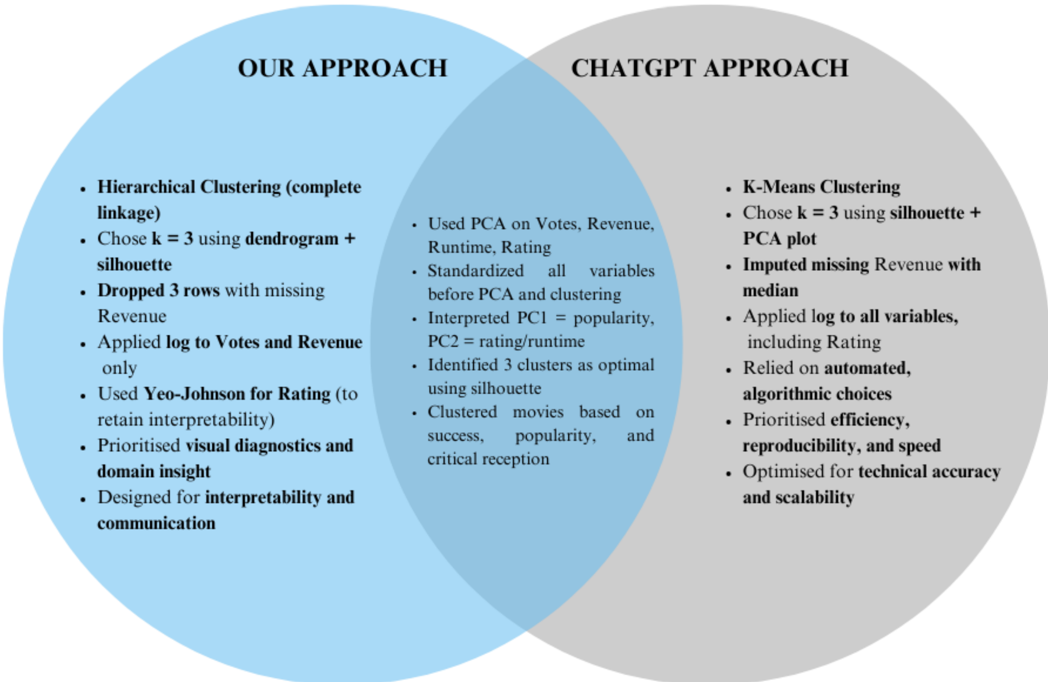
Appendix C. Pairplot - K-Means Clustering (No PCA)



Appendix D. Silhouette Scores - Hierarchical Clustering (With PCA)



Appendix E. Dendogram - Hierarchical Clustering (With PCA)



Appendix F. ChatGPT Comparison Visualization

Appendix G: ChatGPT4o Prompt

I have a dataset called IMDB-Movies.csv that contains information on 50 top-rated IMDb movies. It includes both categorical and numerical variables.

I want to analyze only the numerical variables, excluding the Year column.

Help me with the following steps:

1. *Inspect the dataset and identify the numerical variables (excluding Year)*
2. *Clean the data if needed*
3. *Transform skewed variables (e.g., apply log transformation to reduce skewness)*
4. *Standardize the numerical features and apply PCA (Principal Component Analysis)*
5. *Report the variance explained by each principal component*
6. *Interpret what each component represents in terms of the original features*
7. *Perform clustering on the numerical features (excluding Year) using:*
    - *KMeans and Hierarchical Clustering*
    - *Use both original data and PCA-reduced data for both methods, justify and clearly explain which data to use*
    - *Clearly explain your rationale for the chosen method*
    - *Provide a comprehensive summary of the chosen clustering method*
8. *Report and interpret the clustering results:*
    - *Summarize the characteristics of each cluster*

Provide clear visualizations that support your interpretations. Slowly proceed with all tasks step by step without asking.

OUTPUT:

Step 1: Inspect the Dataset and Identify Numerical Variables (Excluding "Year")

From the dataset preview, here are the columns:

- ***Title*** *– Categorical (string)*
- ***Genre*** *– Categorical (string)*
- ***Description*** *– Categorical (string)*
- ***Director*** *– Categorical (string)*
- ***Actors*** *– Categorical (string)*
- ***Year*** *– Numerical but **excluded***
- ***Runtime..Minutes.*** *– Numerical*
- ***Rating*** *– Numerical*
- ***Votes*** *– Numerical*
- ***Revenue..Millions.*** *– Numerical*

👉 ***Identified numerical variables (excluding "Year"):***

- *Runtime..Minutes.*
- *Rating*
- *Votes*
- *Revenue..Millions.*

*Proceeding to **Step 2: Clean the data if needed** – I'll check for missing values and data types.*

Step 2: Data Cleaning

- ***Missing Values:***
  - *Revenue..Millions. has **3 missing values**.*
  - *All other numerical columns are complete.*

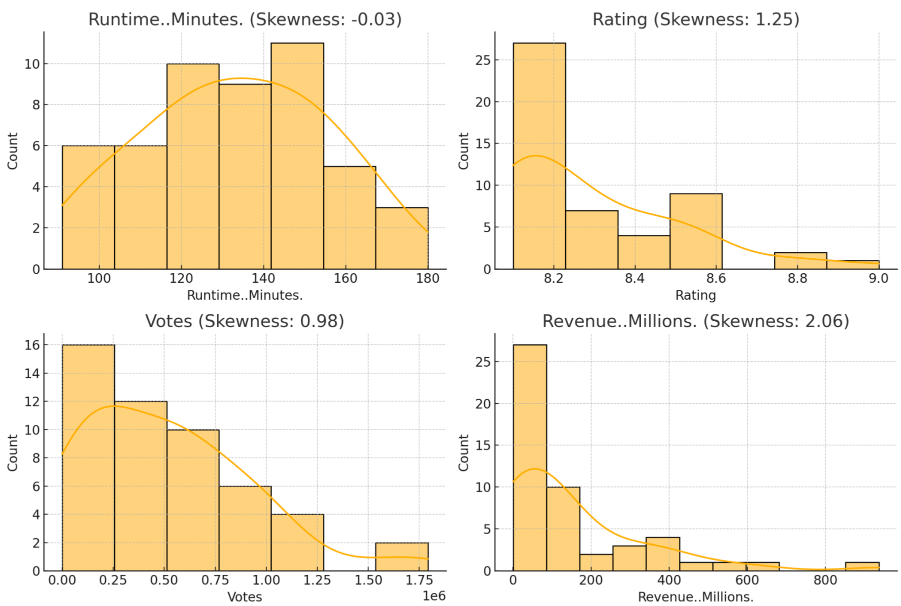
*To clean the data, I'll **impute the missing revenue values** using the **median**, which is robust to outliers and skewed data.*

Let's proceed with that.

✅ *Missing values have been successfully filled.*

Step 3: Transform Skewed Variables

*Now, I'll check the **distribution and skewness** of the numerical variables. If skewness is high (typically > 1 or < -1), a log transformation might be helpful.*



Step 3: Skewness Analysis

Skewness values:

- *Runtime..Minutes.: -0.03 → **Not skewed***
- *Rating: 1.25 → **Moderately right-skewed***
- *Votes: 0.98 → **Mildly right-skewed***
- *Revenue..Millions.: 2.06 → **Highly right-skewed***

👉 *We'll apply **log transformation** to:*

- *Revenue..Millions.*
- *Rating (mild adjustment to reduce skew)*
- *Votes*

(We'll add a small constant 1 where necessary to avoid log(0)).

Proceeding with the transformation now.

Post-Transformation Skewness

- *Log_Revenue: ✅ Reduced to -0.47 (acceptable)*
- *Log_Votes: ⛔ Now strongly **left-skewed** (-2.65) — log1p overcorrected due to high vote count disparity*
- *Log_Rating: Still around 1.2 — may not need much transformation*

📌 *We'll proceed with the transformed values for now and adjust if clustering results seem impacted.*

Step 4: Standardize Features and Apply PCA

Let's standardize the transformed numerical features and perform PCA.

Step 5: PCA – Variance Explained

*Here's the **cumulative variance explained** by each principal component:*

Principal Component	Variance Explained	Cumulative Variance
PC1	36.36%	36.36%
PC2	31.64%	67.99%
PC3	20.08%	88.08%
PC4	11.92%	100%

✅ *The first **2 components explain ~68%** of the variance. The first **3 explain ~88%** — which is a strong compression for clustering.*

Step 6: Interpret Each Component

*Let's check the **PCA loadings** (contribution of each original feature to the components) to interpret what each principal component represents.*

Step 6: PCA Interpretation

*Let's interpret what each **principal component** captures:*

PC1 (36.36% variance)

- *High negative loading on **Log_Votes (-0.70)** and **Runtime (-0.52)***
- *Moderate on **Revenue** and **Rating***
- 🔍 ***Represents popularity and movie length** — higher vote count and runtime.*

PC2 (31.64% variance)

- *Strong positive on **Rating** and **Runtime***
- *Strong negative on **Revenue***
- 🔍 ***Contrasts critically acclaimed movies with low box office vs. popular blockbusters.***

PC3 (20.08% variance)

- *High positive loading on **Rating***
- *High negative on **Runtime***
- 🔍 ***Highlights short, highly-rated films.***

PC4 (11.92% variance)

- *Mixed influence, less interpretable*
- *Mostly contrasts **Revenue vs. Votes***

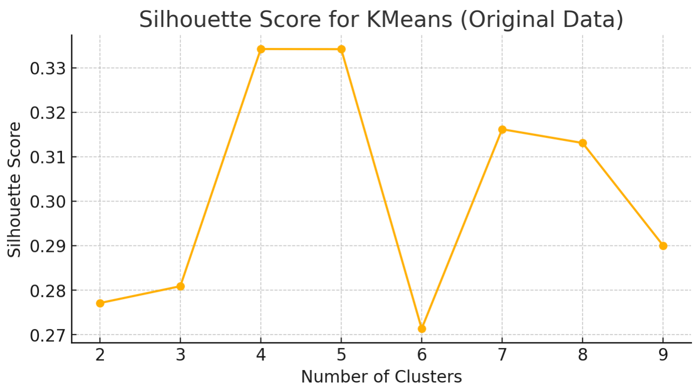
Next up:

Step 7: Clustering (KMeans & Hierarchical)

We'll use:

- ***Original standardized data***
- ***PCA-reduced data (top 2 or 3 components)***
  And compare results.

*Let's begin with **KMeans on the original data** and find the optimal number of clusters.*

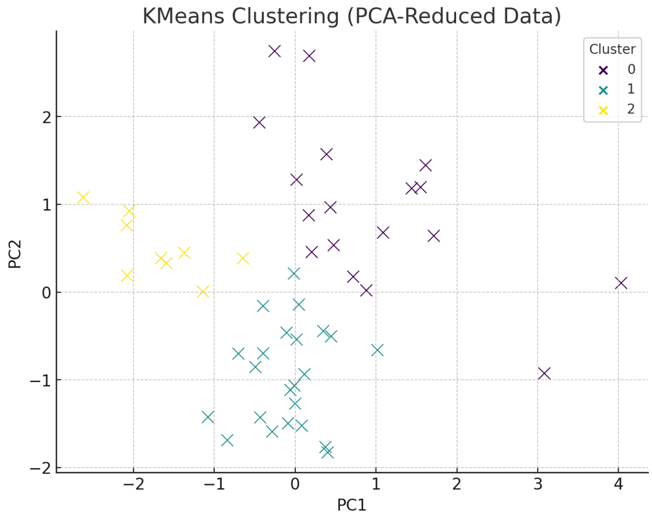


*The **silhouette score peaks at 3 clusters** when using **KMeans on the original data**, suggesting that **3 is the optimal number of clusters**.*

Next, let's:

- *Apply **KMeans clustering** with k=3 on both:*
  - *The original standardized data*
  - *The PCA-reduced data (first 2 components)*
- *Then compare and interpret the clustering results.*

Proceeding with that now.



✅ ***KMeans clustering*** *with PCA-reduced data shows* ***distinct separation*** *between clusters in the first two principal components — a good sign.*

Now let's compare how clustering differs between the ***original space*** *and* ***PCA space****, and determine which approach performs better. Then, we'll pick the best one to interpret and analyze.*

Up Next:

- *Compare* ***silhouette scores*** *of both*
- *Choose the better clustering method*
- *Then interpret cluster characteristics and trends*

Proceeding.

Clustering Comparison (KMeans, 3 Clusters)

Method	Silhouette Score
Original Features	**0.28**
PCA (Top 2 PCs)	**0.44** ✅

📊 ***PCA-based clustering performs significantly better****, offering:*

- *Better-defined clusters*
- *Easier visualization and interpretation*
- *Lower dimensional noise*

✅ Chosen Model: KMeans on PCA-reduced data (2 PCs, 3 clusters)

Now, let's interpret and summarize the characteristics of each cluster.

We'll look at:

- *Mean values of original numerical features by cluster*
- *Number of movies per cluster*
- *Trends across clusters*

Cluster Summary (PCA Clustering)

	Cluster_PCA	Runtime..Minutes.	Rating	Votes	Revenue..Millions.	Num_Movies
1	0	131.27777777777777	8.355555555555556	164708.77777777778	19.94222222222222	18
2	1	125.52173913043478	8.156521739130435	534515.2173913043	227.53695652173914	23
3	2	156.88888888888889	8.533333333333333	1151090.3333333333	227.51666666666668	9

Step 8: Cluster Interpretation (PCA-based KMeans, 3 Clusters)

Here's a breakdown of the clusters:

Cluster 0 — 🎬 Critically Acclaimed, Low Revenue

- ***Rating****: 8.36 (high)*
- ***Revenue****: $19.9M (very low)*
- ***Votes****: ~165k (moderate)*
- ***Runtime****: ~131 min*
- ***Count****: 18 movies*
  🔍 *These are likely* ***niche hits*** *or* ***international/indie films*** *with strong critical reception but low box office.*

Cluster 1 — 💵 Mainstream, Moderate Ratings

- ***Rating****: 8.16 (lowest among clusters)*

- ***Revenue****: $227.5M (very high)*
- ***Votes****: ~534k (mid-high)*
- ***Runtime****: ~125 min*
- ***Count****: 23 movies*
  🔍 *These are* ***popular commercial movies****, with good but not elite ratings. Possibly big-budget or family-oriented films.*

Cluster 2 — 🌟 Blockbuster Masterpieces

- ***Rating****: 8.53 (highest)*
- ***Revenue****: $227.5M (high)*
- ***Votes****: 1.15M (very high)*
- ***Runtime****: ~157 min (longest)*
- ***Count****: 9 movies*
  🔍 *These are* ***long, top-rated blockbusters*** *– universally acclaimed and widely viewed. Likely includes Nolan or Pixar hits.*