**INFO 648**

**BUSINESS DATA ANALYTICS**

GROUP PROJECT

**PREDICTING SONG POPULARITY TO ENHANCE SPOTIFY’S RECOMMENDATION SYSTEM**

**Submitted By:**   **Submitted To:  
 GROUP 6** Dr. Ning Luo  
 Gauri V Nair   
 Jesin Kandathy Joy  
 Riddhi Rungta  
 Sarath Sabu

**Date of Submission**

6th May 2025

**Contributions**

Gauri V Nair – Data Preprocessing and Feature Engineering

Sarath Sabu – Q1 Models : Decision Tree, Logistic Regression, KNN

Jesin Kandathy Joy – Q2 : Profit Driven Model Evaluation

Riddhi Rungta – Q3 : Clustering and Valence Impact

**TABLE OF CONTENT**

[INTRODUCTION 2](#_kn5x6zeueiwl)

[DATA PREPROCESSING AND FEATURE ENGINEERING 3](#_g0ejkv2so166)

[PREDICTING POPULAR SONGS USING VARIOUS MODELS (Q1) 5](#_n8wm5ydnu1s)

[Feature Selection and Importance 7](#_usjkl1lb7873)

[Managerial Insights and Recommendations 7](#_289qzk30wkch)

[PROFIT DRIVEN MODEL EVALUATION (Q2) 9](#_qyarzsjksz68)

[Business-Oriented Evaluation Framework 9](#_arrr2yyr0iml)

[Profit-Based Model Selection 10](#_ra0avpdvdoqg)

[Strategic Implications: Accuracy vs. Profit 11](#_4ap847hxwcht)

[Managerial Insights and Recommendations 11](#_a6vrqwxqlvns)

[CLUSTERING AND VALENCE IMPACT (Q3) 13](#_xu08j84wsuaj)

[(1) Cluster Characteristics and Managerial Interpretations 13](#_18u5jm4316rb)

[Insights and Recommendations 15](#_mm5597xnq6)

[(2) Valence Impact Analysis 15](#_drl5c62gh4yw)

[Actionable Recommendations for Valence Modeling 16](#_1hczqqhg5sza)

[(3) Association Rule Mining & Hit-Predictive Feature Patterns 17](#_27myblb6bxv9)

[Final Recommendations 18](#_qaq7t513rdcs)

[APPENDIX 20](#_8btnkxgtllxt)

[A. Data Preprocessing & Exploratory Analysis 20](#_q65uj97239ng)

[B. Q1 – Predicting Popularity (Model Comparisons & Tuning) 21](#_76iihzy9yeon)

[C. Q3 – Clustering, Valence, and Association Mining 23](#_hgwq6tbf0py)

# **INTRODUCTION**

As competition intensifies among music streaming platforms, the ability to deliver personalized, timely, and engaging content has become essential. Spotify, as one of the world’s leading streaming services, relies heavily on its recommender system to suggest tracks that resonate with user preferences. This project aims to enhance Spotify's recommendation system through predictive analytics by identifying which songs are likely to become popular upon release.

The dataset used for this project (songs\_utf.csv) contains data from Spotify’s U.S. charts spanning the years 1998 to 2020. It includes various song-level features such as acoustic properties, tempo, loudness, danceability, and genre-based indicators. The primary objective is to predict whether a song will achieve a popularity score of 65 or higher, using these features. This binary classification can assist Spotify in prioritizing songs for promotion or inclusion in high-impact playlists.

This project was structured into three phases:

- Q1 – Predictive Modeling: Develop and evaluate multiple machine learning models to predict song popularity, comparing their performance based on evaluation metrics such as accuracy, precision, recall, and AUC.

- Q2 – Profit Optimization: Extend the analysis by incorporating cost and revenue information into model evaluation, thereby selecting the model that maximizes business value.

- Q3 – Pattern Discovery: Explore clustering and feature relationships to understand how specific song attributes, especially valence, influence success, and how these patterns can support playlist curation.

By integrating data science with business-driven objectives, this project provides actionable insights for improving Spotify’s recommendation pipeline.

# **DATA PREPROCESSING AND FEATURE ENGINEERING**

The dataset provided included 1,500 observations, each representing a unique song, and a total of 28 columns containing various song-level attributes. These features covered a broad range of characteristics, including audio properties (e.g., danceability, acousticness, valence), metadata (e.g., year of release, duration), and genre indicators (e.g., pop, rock, hip hop).

The first step in preprocessing involved handling missing data. Upon inspection, we found that all missing values were located at the bottom of the dataset, affecting a small number of rows. Given that the missingness appeared random and did not impact any specific feature disproportionately, we opted to remove all rows containing null values. This approach helped maintain the statistical integrity of the remaining data while simplifying the cleaning process.

Next, we created the binary target variable popular\_bin to define the modeling objective. Songs with a popularity score of 65 or higher were labeled as 1 (popular), and all others were labeled as 0 (not popular). This binary classification framework aligned with the project’s goal of helping Spotify distinguish between potential hits and lower-performing tracks.

To prepare the data for modeling, we categorized the dataset’s columns into numerical and categorical variables. Numerical features included values such as duration\_ms, energy, tempo, valence, and speechiness. Categorical variables included artist, song, genre, and explicit. Among the categorical features, explicit—which originally appeared as a Boolean variable (True or False)—was encoded to binary values (1 for True, 0 for False) to ensure compatibility with logistic regression and other modeling algorithms. Other high-cardinality textual features like artist and song were excluded from modeling due to their limited predictive value and the complexity they introduce.

To further refine the dataset, we performed multicollinearity checks using the Variance Inflation Factor (VIF). This step helped identify features that were highly correlated with each other and could potentially distort the model’s performance. Two columns, popularity and hot, were immediately removed as they were highly correlated with the target variable and posed a risk of data leakage. Additionally, the year column was dropped due to its extremely high VIF score of over 220, indicating strong multicollinearity with several other variables. We also retained the encoded explicit variable, which was not impacted by multicollinearity and had shown promise during preliminary analysis.

After completing VIF-based filtering, we further refined our feature set using Sequential Forward Selection (SFS). In this method, features are added one at a time based on their ability to improve the model’s AUC score through cross-validation. We implemented SFS using logistic regression as the base model, evaluating combinations with 5-fold cross-validation. The best-performing subset consisted of the following 12 features: explicit, song\_name\_len, energy, key, loudness, speechiness, acousticness, tempo, rock, hiphop, dance, and rnb. This subset achieved a cross-validated AUC of 0.6386, indicating a meaningful balance between model complexity and predictive power. These selected features were then used as inputs for subsequent model development and evaluation.***(Refer to Appendix Figures A1–A4 for the graph of top 10 songs, correlation matrix between features, ROC curve of the model used to evaluate the effectiveness of the features selected and the feature importance graph.)***

# **PREDICTING POPULAR SONGS USING VARIOUS MODELS (Q1)**

To enhance Spotify’s ability to recommend songs likely to succeed, we formulated the problem as a binary classification task, predicting whether a song's popularity score exceeds 64 (i.e., popular\_bin = 1 for popularity ≥ 65). We built and compared three different predictive models: K-Nearest Neighbors (KNN), Logistic Regression, and Decision Tree Classifier. All models were trained and evaluated using a stratified train-test split, and we applied 10-fold cross-validation to tune hyperparameters and reduce overfitting. Model performance was assessed using accuracy, precision, recall, and F1 score, with final selection based on test accuracy, in line with the project requirements.

Initial modeling revealed that logistic regression achieved a test accuracy of 0.582, with moderate precision and recall (0.578 and 0.514 respectively). The original decision tree slightly improved recall but showed clear signs of overfitting with a very high training accuracy (0.846) and test accuracy of only 0.575. To mitigate this, we fine-tuned the decision tree using grid search over max\_depth and min\_samples\_split, improving performance marginally to a test accuracy of 0.588 and an F1 score of 0.560. (***Refer to appendix B1 for detailed graphs and plots for evaluation metrics, decision tree visualizations, and learning curves.)***

Among all models, the tuned KNN classifier performed the best, achieving a test accuracy of 0.593, an improvement over both the logistic regression and decision tree models. Precision and F1 score were also highest for KNN, at 0.624 and 0.552, respectively. The recall remained steady across models at around 0.495. This indicates that the KNN model was better at balancing precision (reducing false positives) and overall predictive power.

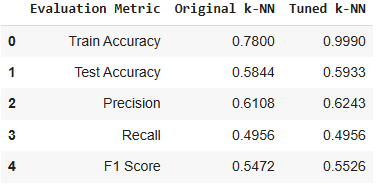


Fig 1:

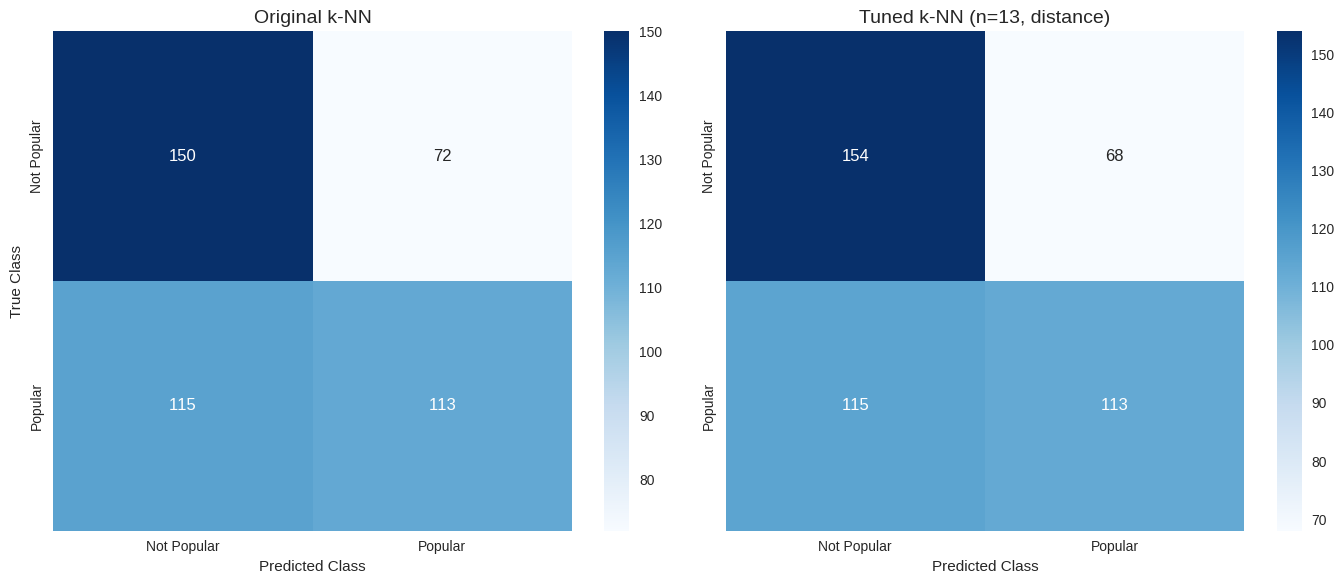


Fig 2:

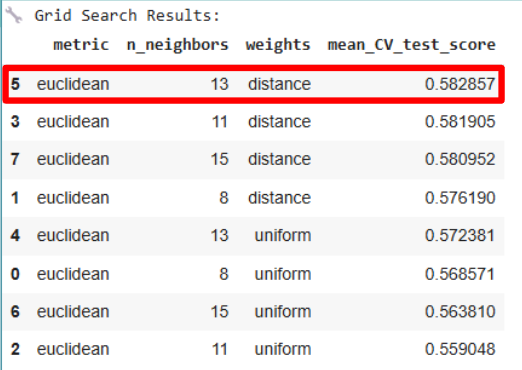
******

Fig 3:

We used a grid search to tune the KNN model, varying the number of neighbors (n\_neighbors = [8, 11, 13, 15]), weight functions (uniform, distance), and distance metric (euclidean). The best parameters were found at n\_neighbors = 13 with distance weighting, which accounts for proximity more effectively than a flat uniform approach. Cross-validation accuracy trends confirmed this setting minimized overfitting while improving generalizability. ***(Refer to appendix B2 and B3 for detailed graphs and plots)***

## **Feature Selection and Importance**

Feature selection was guided by both domain relevance and statistical analysis. We included audio features known to influence listener behavior, energy, danceability, valence, loudness, speechiness, tempo, and acousticness, as well as binary genre tags (hiphop, dance, rnb, etc.) and the explicit label. The derived feature song\_name\_len was also retained, based on the hypothesis that shorter, catchier song names might correlate with virality.

Feature importance plots from the decision tree models highlighted energy, acousticness, loudness, and speechiness as consistently strong predictors. Interestingly, genre indicators had lower importance, suggesting that musical characteristics often matter more than traditional genre labels in determining song success.

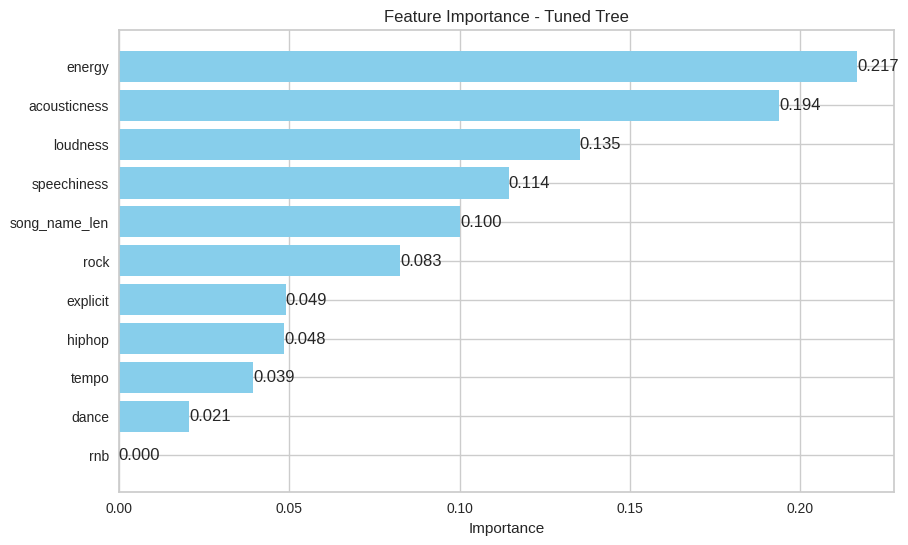


Fig 4:

## **Managerial Insights and Recommendations**

The findings offer valuable direction for Spotify's recommender system and playlist strategy:

1. Energetic and high-loudness tracks are central to song popularity. These features signal high listener engagement and should be prioritized in playlists designed for broad appeal.
2. Danceability and speechiness, traits common in hip-hop, spoken word, and upbeat pop, are also strongly predictive of success. Tracks with these features are good candidates for "Top Hits", "Viral 50", and high-rotation lists.
3. Contrary to intuition, explicit songs were frequently popular, suggesting that content restrictions shouldn’t overly limit what gets recommended, Spotify should remain inclusive of diverse lyrical content when generating playlists.
4. Genre diversity enhances popularity, but genre alone is not enough. The most successful tracks tend to combine multiple strong audio features, reinforcing the value of feature-based recommendation over genre-based filters.
5. The KNN model’s strong performance further supports a strategy of proximity-based recommendation, where songs similar to already-popular ones (in terms of energy, loudness, etc.) are surfaced earlier in release cycles or discovery feeds.

Ultimately, these insights suggest Spotify should lean into feature-rich, cross-genre songs, and maintain dynamic, feature-aware modeling like KNN to stay ahead of evolving music tastes. Updating models periodically and integrating listener behavior feedback will further improve hit prediction accuracy and user satisfaction.

# **PROFIT DRIVEN MODEL EVALUATION (Q2)**

To ensure that Spotify’s predictive modeling aligns with its financial objectives, we extended our evaluation framework to include a profit-based analysis. Rather than relying solely on predictive accuracy, this approach evaluates models based on their potential financial impact, incorporating both the revenue from correct recommendations and the costs of misclassifications.

## Business-Oriented Evaluation Framework

Each outcome in the confusion matrix was assigned a monetary value to reflect its real-world business implications:

* True Positive (TP): +$1,000 (Correctly recommending a popular song)
* False Positive (FP): –$700 (Incorrectly promoting a non-popular song)
* False Negative (FN): –$900 (Failing to promote a song that becomes popular)
* True Negative (TN): $0 (Correctly ignoring a non-popular song)

This framework enabled a comprehensive assessment of each model’s financial performance. The table below summarizes the results based on the confusion matrices from the tuned versions of each model.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Model** | **TP** | **FP** | **FN** | **TN** | **Net Profit/Loss** |
| Tuned Decision Tree | 119 | 77 | 109 | 145 | –$33,000 |
| Tuned K-Nearest Neighbors | 113 | 68 | 115 | 154 | –$38,100 |
| Logistic Regression (CV) | 367 | 267 | 347 | 519 | –$132,200 |

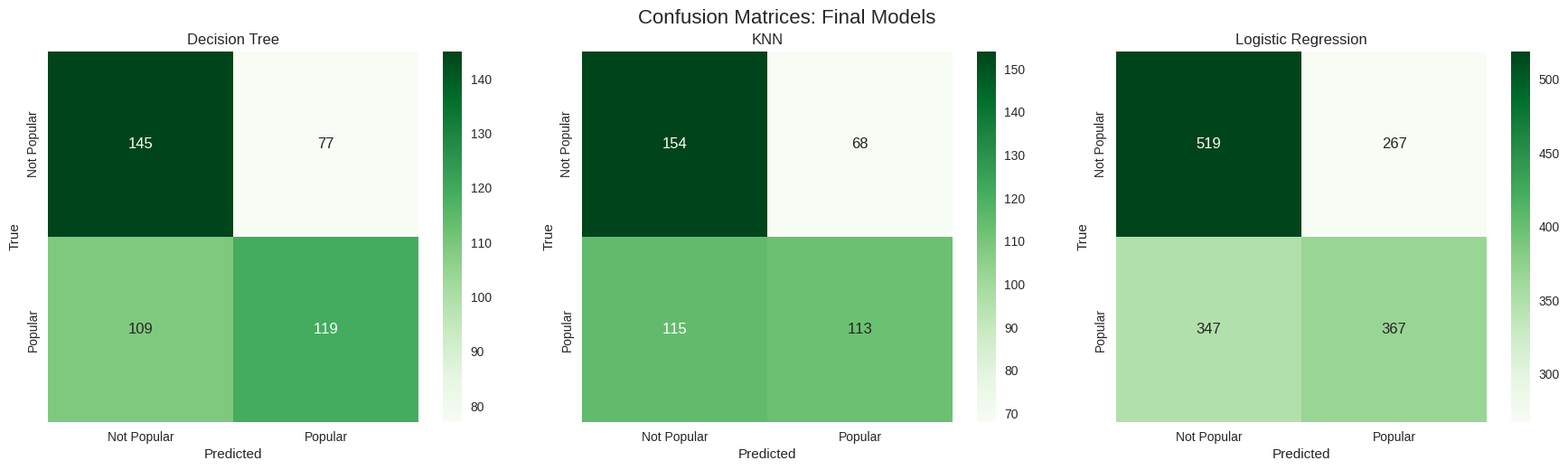


Fig 1: Confusion matrix- Decision Tree Model

## **Profit-Based Model Selection**

Among the evaluated models, the tuned Decision Tree demonstrated the best financial outcome, incurring the lowest overall loss of $33,000. It achieved this by generating more true positives and fewer false negatives than other models. While the false positive count was slightly higher than that of the KNN model, the Decision Tree compensated for this through a higher number of correct high-value predictions.

In contrast, the KNN model, despite being selected as the most accurate model in Q1, incurred higher financial losses due to its limited ability to capture true positives. The Logistic Regression model resulted in the greatest loss due to a combination of high false positives and false negatives, illustrating that even accurate models may perform poorly from a business perspective.

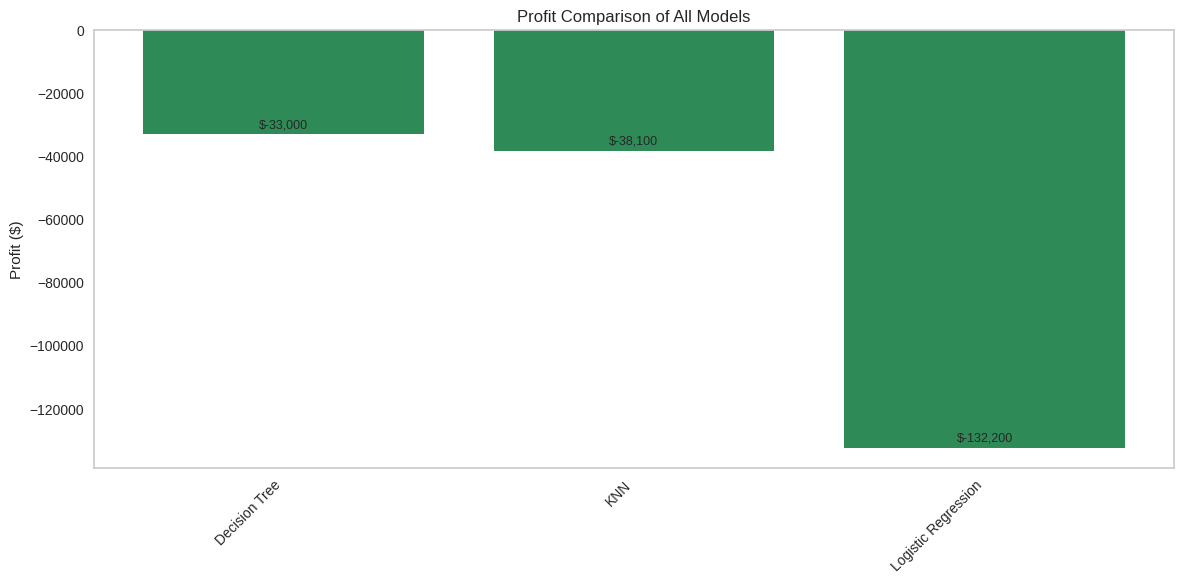


Fig 2: Net Profit/Loss for Each Model Based on Business Cost Structure

## **Strategic Implications: Accuracy vs. Profit**

The shift from Q1’s accuracy-focused selection to Q2’s profit-based evaluation revealed important trade-offs. While the KNN model offered the highest test accuracy (59.33%), the Decision Tree model, with slightly lower accuracy (58.67%), provided a superior balance of cost-effective predictions. This highlights the need to consider business-aligned evaluation metrics when deploying models in real-world recommendation systems.

## **Managerial Insights and Recommendations**

* Adopt Profit-Oriented Models: The Decision Tree model aligns closely with Spotify’s financial goals by minimizing the most costly errors (false negatives) and increasing profitable predictions (true positives). It is also interpretable and easier to integrate with operational systems.
* Implement Real-Time Monitoring: Spotify should incorporate financial impact metrics into its model monitoring dashboards. Tracking profit, misclassification costs, and prediction distributions can support timely interventions and continuous improvement.
* Cost-Sensitive Training: Future iterations of model development should explore cost-sensitive learning and threshold optimization to enhance business performance.
* Dynamic Model Updates: Given the evolving nature of music trends and listener behavior, periodic retraining and evaluation using updated data and cost parameters are essential.

In conclusion, while traditional metrics such as accuracy remain important, Spotify’s recommendation strategy should prioritize models that deliver the highest net business value. The tuned Decision Tree offers a practical, cost-efficient solution that balances predictive performance with profitability, supporting smarter recommendation outcomes and sustainable platform growth.

# **CLUSTERING AND VALENCE IMPACT (Q3)**

To generate deeper insights that could improve Spotify’s playlist recommendations, we conducted a clustering-based segmentation of songs based on their audio features, primarily danceability and energy, with optional genre dummies such as hip-hop, rock, dance, and rnb. These features were chosen due to their strong association with user engagement and musical vibrancy, both of which are critical for playlist performance. The overarching objective was to (1) identify clusters of similar songs, (2) examine how valence (musical positivity) impacts popularity within each group, and (3) discover frequent combinations of features linked to high or low popularity.

We applied three different unsupervised learning methods to the dataset: K-Means, Agglomerative Clustering, and DBSCAN. All three methods yielded meaningful groupings of songs with distinct energy and mood profiles. The K-Means method, using elbow and silhouette methods for optimization, identified an optimal k=6 clusters. Agglomerative clustering, using complete linkage and silhouette analysis, produced nine clusters with a silhouette score of 0.734, highest among the hierarchical options tested. Finally, DBSCAN, with eps=0.1 and min\_samples=9, identified seven clusters plus a group of outliers and showed robust separation with a silhouette score of 0.767. (***Refer to appendix C for detailed graphs and plots)***

## **Cluster Characteristics and Managerial Interpretations**

The clusters revealed clear distinctions in feature combinations and popularity profiles. Across all methods, clusters with high energy and danceability consistently made up the majority of popular tracks. For example, in K-Means, Clusters 0, 1, and 2 accounted for over 60% of the dataset and were dominated by energetic, mainstream tracks. These clusters frequently blended hip-hop, rnb, and dance genres, which correlated well with mass appeal and playlist rotation potential. Niche segments, such as Cluster 3 (dance/EDM-focused) and Cluster 4 (high-energy rock), were smaller but showed high engagement, making them ideal candidates for targeted or mood-specific playlists.

Agglomerative clustering confirmed these findings, with Cluster 2 and Cluster 4 emerging as the most robust for broad engagement. Meanwhile, DBSCAN provided additional nuance, distinguishing a high-energy, genre-fused cluster (Cluster 2) from a more niche, R&B-heavy group (Cluster 5), while also identifying outliers (8%) which can be used for experimental or discovery playlists.

**K-Means**

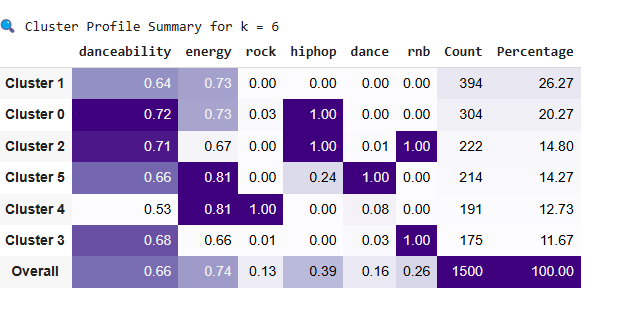


Fig 5:

**Agglomerative Clustering - Complete Linkage**

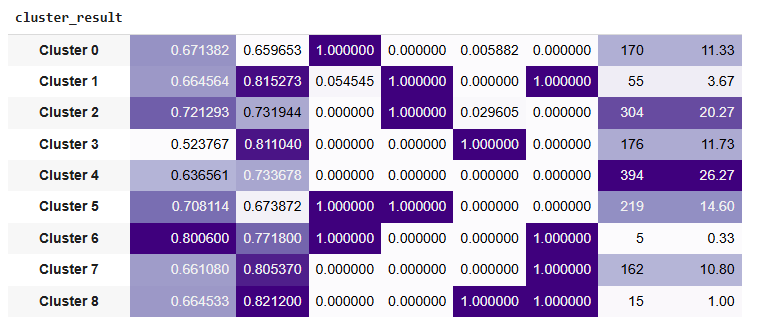


Fig 6:

**DBSCAN**

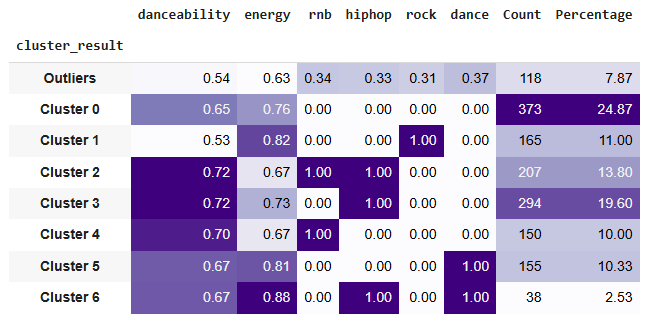
****

Fig 7:

### **Insights and Recommendations**

1. **Prioritize energetic, high-danceability clusters for flagship playlists**

Clusters containing songs with high energy and danceability, particularly those blending hip-hop, R&B, and dance genres made up the majority of popular tracks across all clustering methods. Spotify should focus on these clusters when curating mass-appeal playlists like “Today’s Top Hits” or “Viral 50”.

1. **Use niche clusters for targeted mood- or genre-specific playlists**

Smaller, more specialized clusters, such as those with strong dance, rock, or EDM identity, can power genre-specific playlists and deepen listener engagement. These are ideal for playlists like “Dance Rising”, “Rock Classics”, or “EDM Essentials”.

1. **Leverage outlier clusters to promote discovery and experimentation**

DBSCAN revealed 8% of songs as outliers, those that don’t belong to any dominant cluster. These tracks are excellent candidates for exploratory playlists like “Fresh Finds” or “Discover Weekly”, helping Spotify surface novel or under-the-radar content.

1. **Regularly refresh cluster models to adapt to changing music trends**

Listener preferences shift rapidly, so Spotify should re-run clustering on recent data at regular intervals. This will ensure playlists remain aligned with emerging subgenres and evolving user behavior across global markets.

1. **Use cluster membership for personalized recommendation strategies**

Assign users to preferred clusters based on their streaming history, and recommend songs aligned with their audio feature preferences (e.g., energy, genre, mood). This will enable smarter, data-driven personalization and stronger playlist retention.

## **(2) Valence Impact Analysis**

We conducted logistic regression analyses within each cluster to examine whether valence (i.e., how happy or positive a song sounds) influenced its popularity. The results revealed that valence does not have a uniform effect across clusters. For example, in cluster\_kmeans\_0 and dbscan\_5, higher valence positively correlated with popularity, suggesting upbeat songs resonated well with those audiences. In contrast, clusters like agnes\_5 and kmeans\_2 showed a negative relationship, where moodier songs (lower valence) were more popular.

Most coefficients, however, were not statistically significant (p > 0.05), indicating that valence alone is not a strong standalone predictor of song success in most segments. One exception was agnes\_7, where the strong negative coefficient with borderline significance suggests that mood preferences can matter more for specific listener groups or playlist contexts. This insight reinforces the need to view valence in combination with other features, such as energy and genre, rather than in isolation.

****

Fig 8:

### **Actionable Recommendations for Valence Modeling**

1. **Avoid blanket mood-based recommendations**Valence does not impact popularity uniformly, some clusters prefer happier tracks, while others favor moodier ones. Spotify should avoid applying the same valence logic across all playlists.
2. **Adapt playlist strategies to cluster-specific mood preferences**Use valence insights only where they significantly influence popularity, focusing mood-based curation in clusters where valence shows a meaningful trend.
3. **Experiment with mood-driven playlists**Design playlists around emotional tone (e.g., happy, mellow, intense) in clusters where valence coefficients were strong, allowing Spotify to cater to diverse emotional listening needs.
4. **Use valence in combination with other audio features**Valence alone is weak as a predictor but becomes more useful when paired with energy, genre, and danceability, enhancing personalization through multi-feature modeling.
5. **Update mood modeling regularly**Listener sentiment and cultural context shift over time, so Spotify should re-evaluate valence effects periodically to ensure emotional tones remain aligned with user engagement trends.

## **(3) Association Rule Mining & Hit-Predictive Feature Patterns**

To identify the combinations of audio characteristics most closely associated with hit songs, we conducted association rule mining on the transformed dataset. We began by selecting core continuous features, such as energy, loudness, valence, danceability, tempo, speechiness, and acousticness, alongside binary genre indicators (e.g., hip-hop, dance, rock, R&B). Continuous features were binarized using their median values to distinguish “high” and “low” traits, enabling discrete pattern extraction. Our target was a binary outcome variable (popular\_bin) indicating whether a song’s popularity exceeded 64.

The analysis surfaced frequent itemsets that reliably co-occur with high popularity. Rules with lift scores greater than 2.0 were considered highly predictive, implying that songs with these feature combinations are at least twice as likely to be hits compared to baseline probability. Notably, the most predictive combinations consistently included high energy, high loudness, high speechiness, and hip-hop genre tags. These features emerged in multiple top-ranked rules, confirming their centrality to hit-making success.

For instance, one of the strongest rules indicated that songs with both loudness\_high and speechiness\_high had a strong probability of also being popular and energetic hip-hop tracks. Another high-confidence rule linked energy\_high with hip-hop, reinforcing the synergy between sonic intensity and rhythmic vocal content.

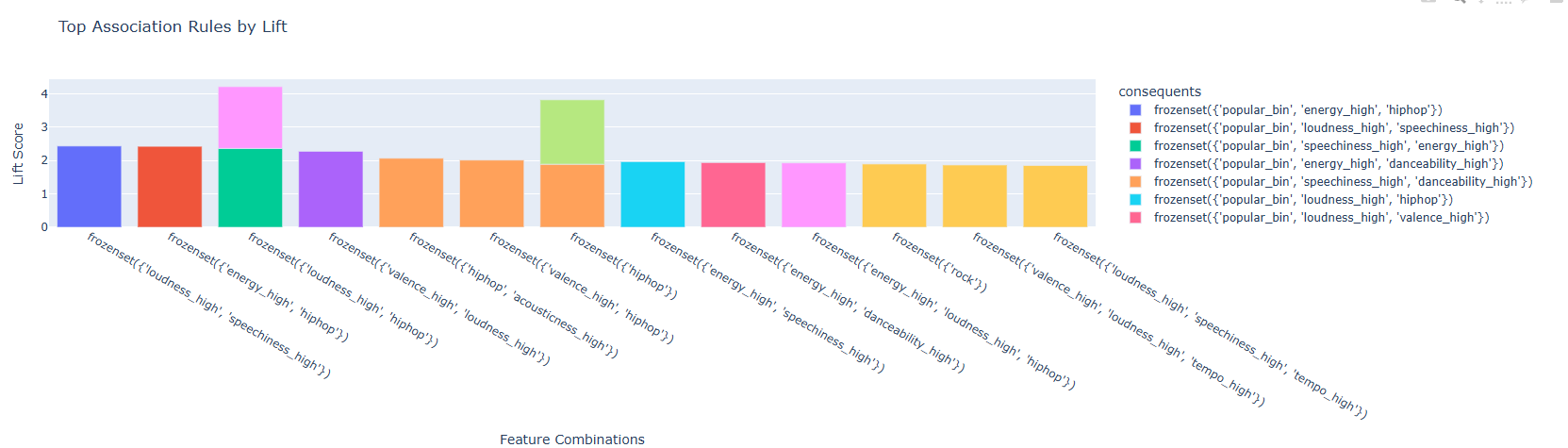


Fig 9:

A bar chart summarizing the top 15 rules by lift provides visual evidence of these dominant combinations, highlighting that songs with intense, expressive, and rhythm-driven features are statistically the most likely to succeed.

## **Final Recommendations**

Based on these findings, we offer the following targeted recommendations for enhancing Spotify’s recommendation and playlist strategy:

1. **Prioritize Songs with Proven Hit Features**

Songs that exhibit high levels of energy, loudness, and speechiness, especially those in the hip-hop and dance genres, should be elevated within flagship playlists. These features form the core structure of many top-performing tracks and align with user engagement patterns.

1. **Develop “Hit Predictor” Playlists Based on Feature Rules**

Spotify can automatically generate playlists using songs that match high-lift feature patterns (e.g., energy\_high + hiphop) to surface new releases with strong hit potential. This can increase listener stickiness while boosting exposure for emerging tracks.

1. **Promote Genre & Mood Fusion**

Tracks that span genres and combine emotionally expressive features (like valence or acousticness) with high energy traits should be emphasized in blended playlists. These combinations not only appeal to diverse user segments but also encourage music discovery and cross-genre exploration.

1. **Leverage Feature-Based Personalization**

User recommendations can be enriched by matching listeners with playlists that reflect their unique preference profiles, such as those who consistently engage with upbeat, high-energy hip-hop or mellow, acoustic-rich tracks. This shift from genre-only curation to feature-level personalization will increase satisfaction and retention.

1. **Continuously Monitor Emerging Hit Patterns**

As listener behavior and production styles evolve, Spotify should regularly re-run association rule mining to identify new combinations that signal rising popularity. This ensures the recommender system remains dynamic and ahead of music trends.

These insights, when combined with our earlier modeling and clustering findings, offer a powerful roadmap for Spotify to strengthen its early-stage hit identification, improve playlist targeting, and personalize listening experiences at scale.

# 

# **APPENDIX**

## **Data Preprocessing & Exploratory Analysis**

Figure A1: Top Songs on Spotify

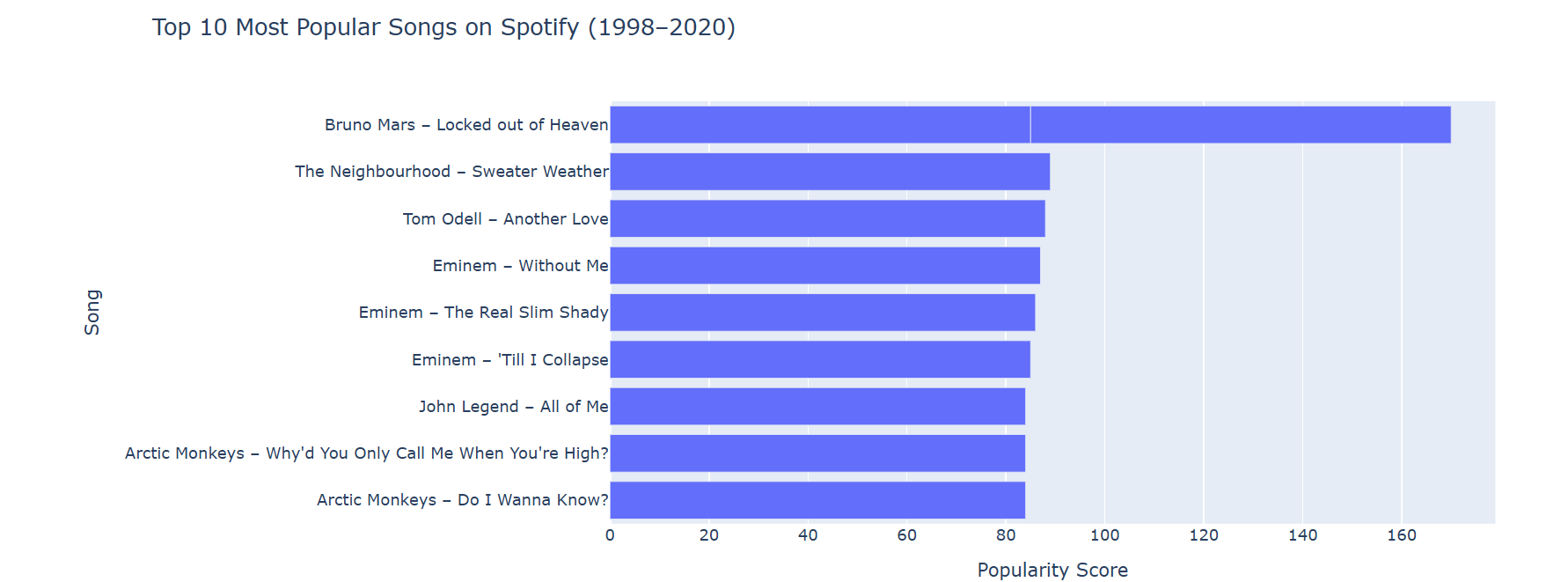


Figure A2: Correlation heatmap between numeric features

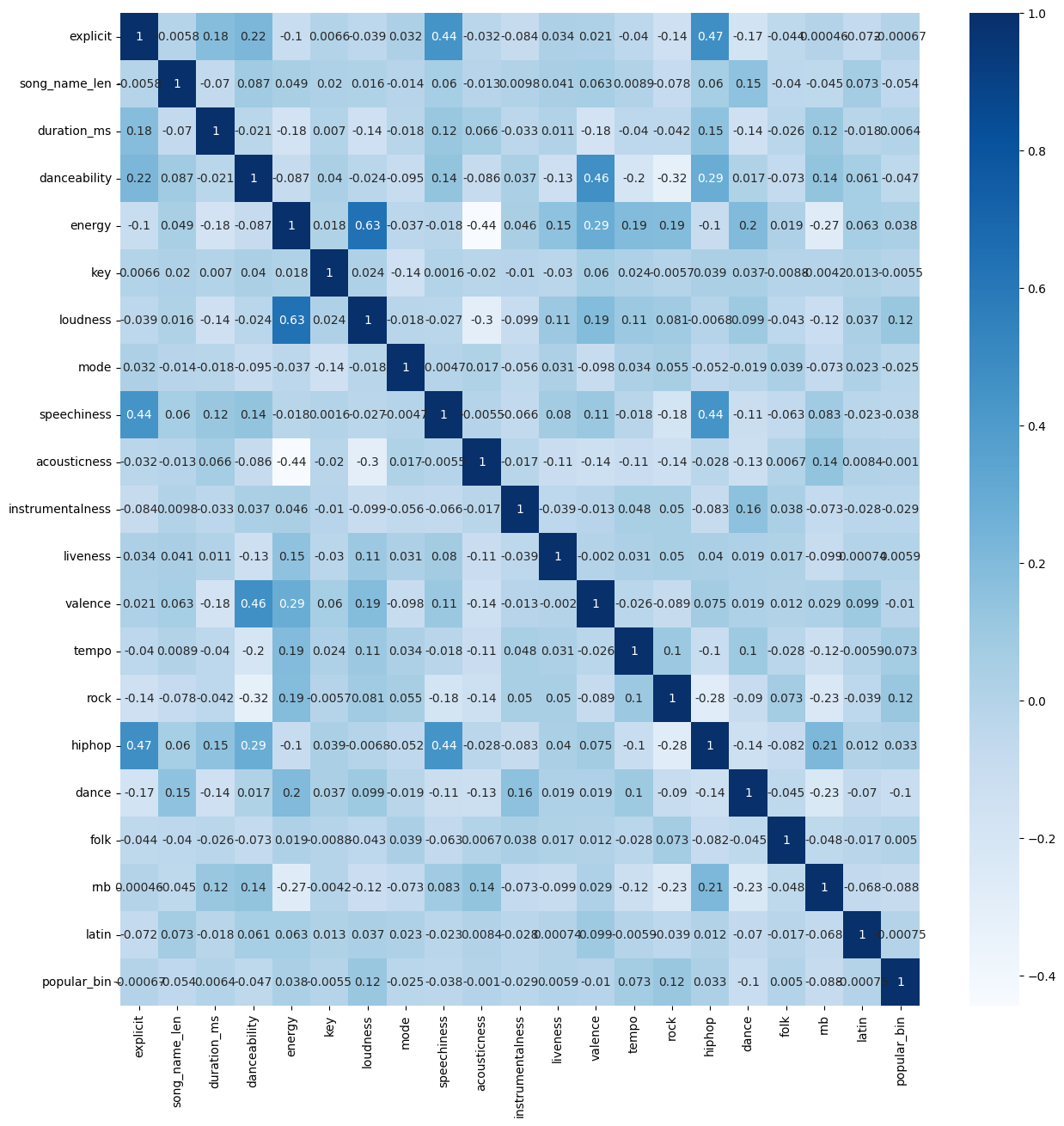


Figure A3: Confusion Matrix and ROC Curve of the Model using Best Combination Features

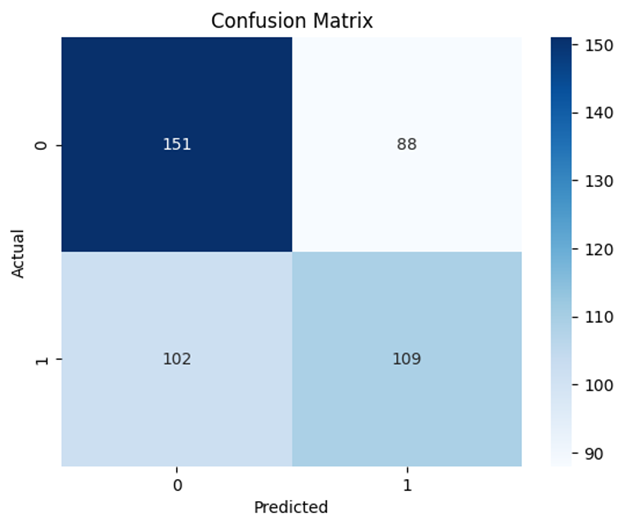
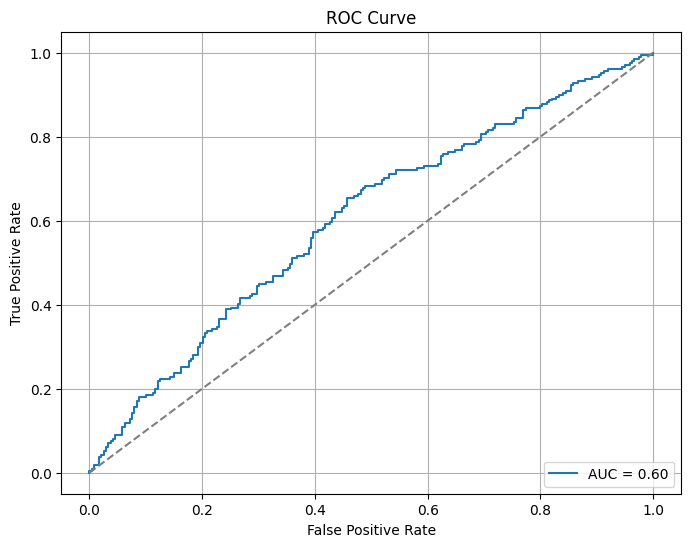
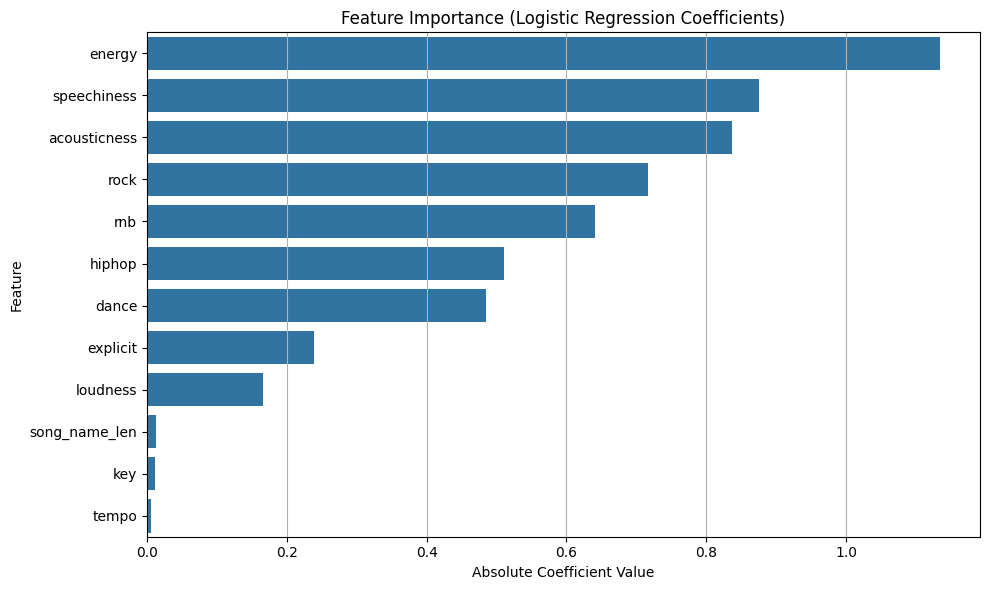
 

Figure A4: Feature Importance of the Best Combination of Features



## **Q1 – Predicting Popularity (Model Comparisons & Tuning)**

Figure B1: Classifier Evaluation of non-best Models (Logistic Regression and Decision Tree)

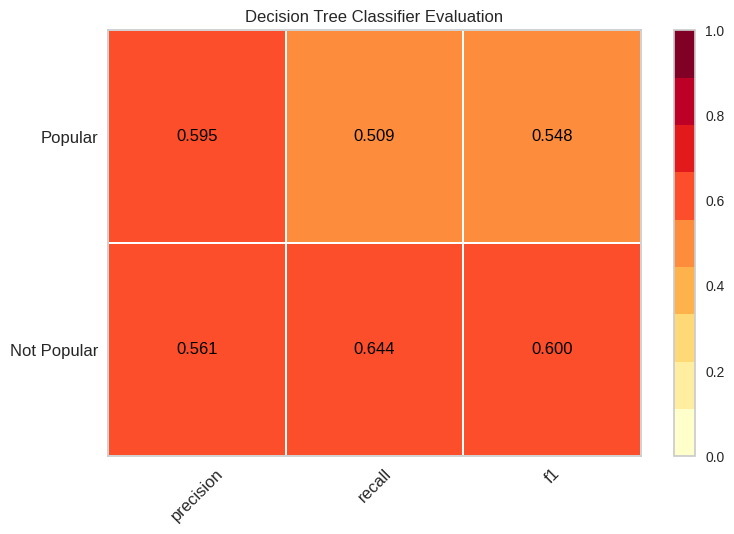
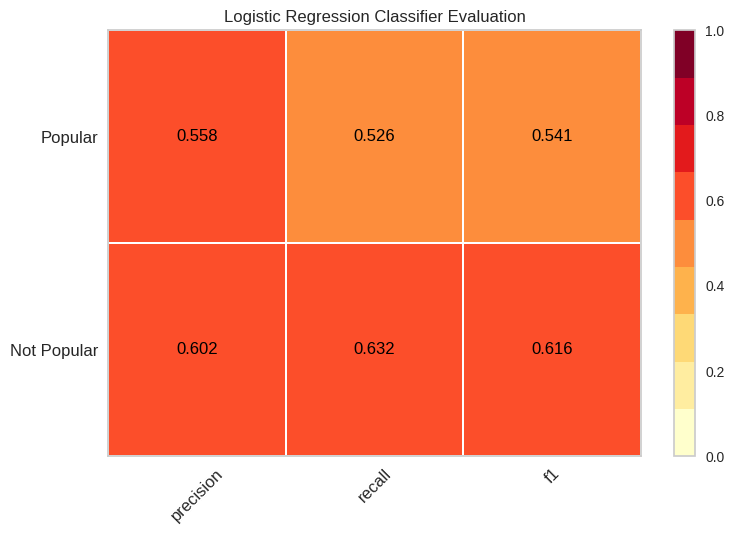


Figure B2:Decision Tree (Original)

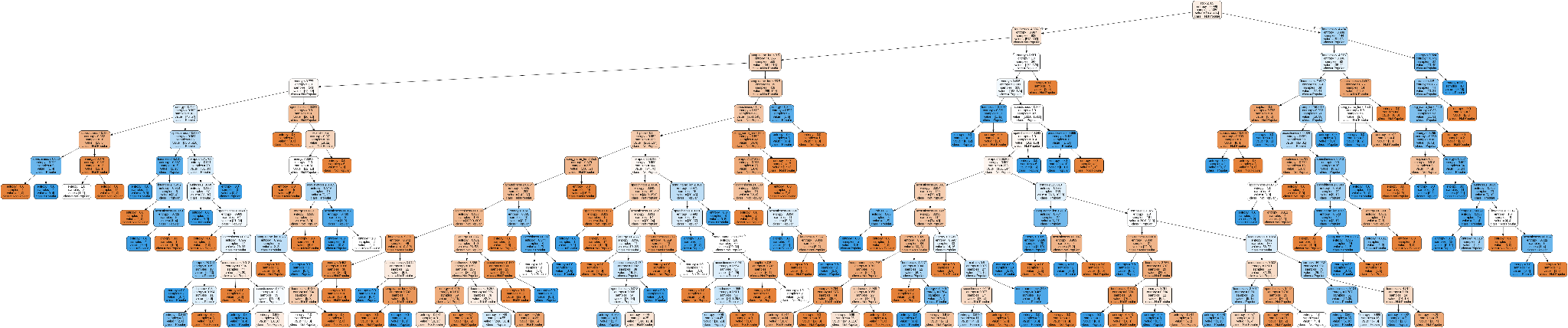


Figure B2.1: Decision Tree Tuning Process

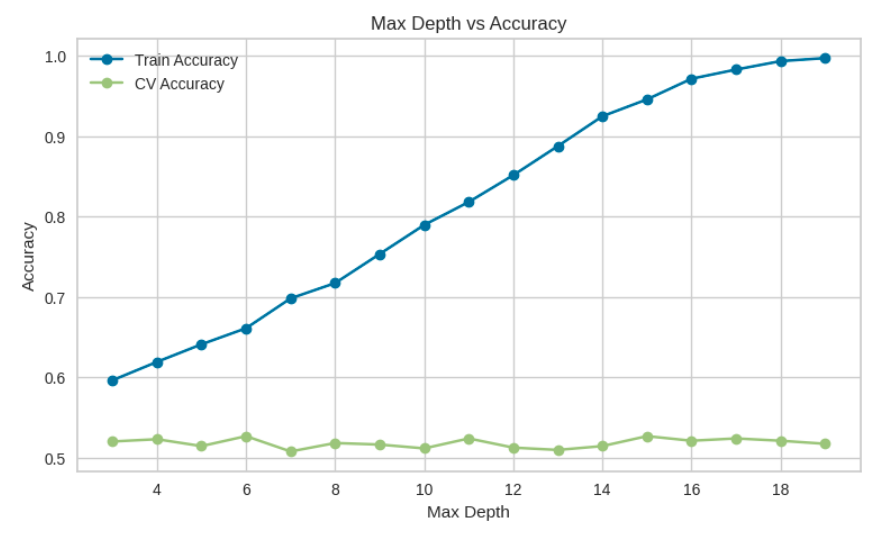


Figure B2.2: Decision Tree Original and Tuned Tree Comparison Metrics

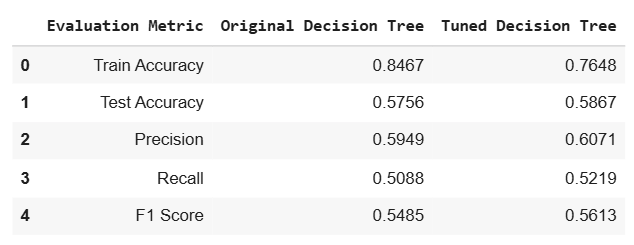


Figure B3: KNN Classifier Evaluation on Training Data

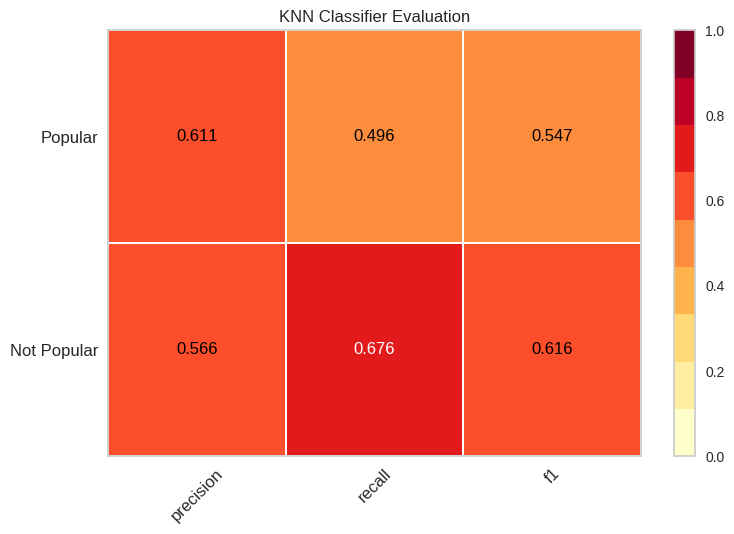
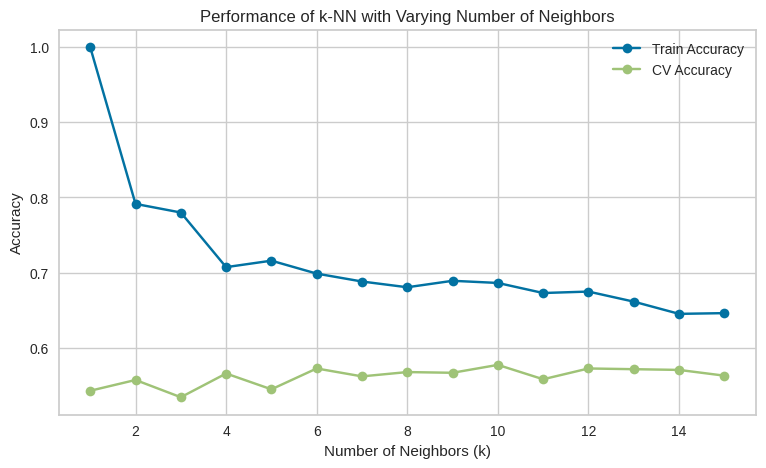
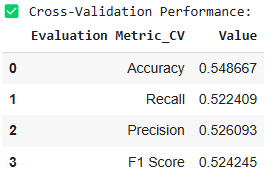


Figure B4: Learning/ Tuning Curves - KNN with Evaluation Metrics



## **Q3 – Clustering, Valence, and Association Mining**

Figure C1: Clustering evaluation plots - K Means

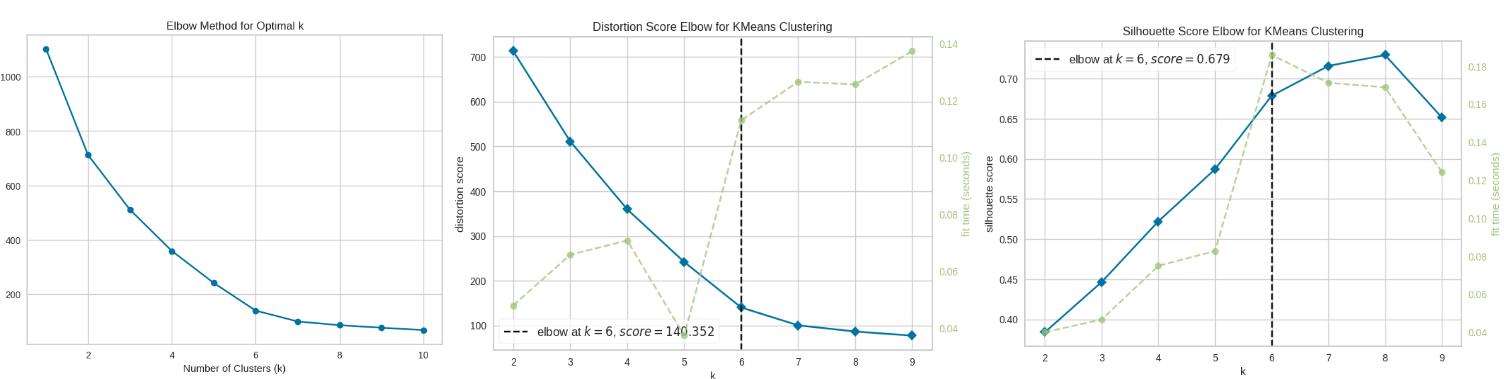


Figure C2: Cluster Evaluation plots - Agglomerative

Figure C2.1: Silhouette Scores - Agglomerative

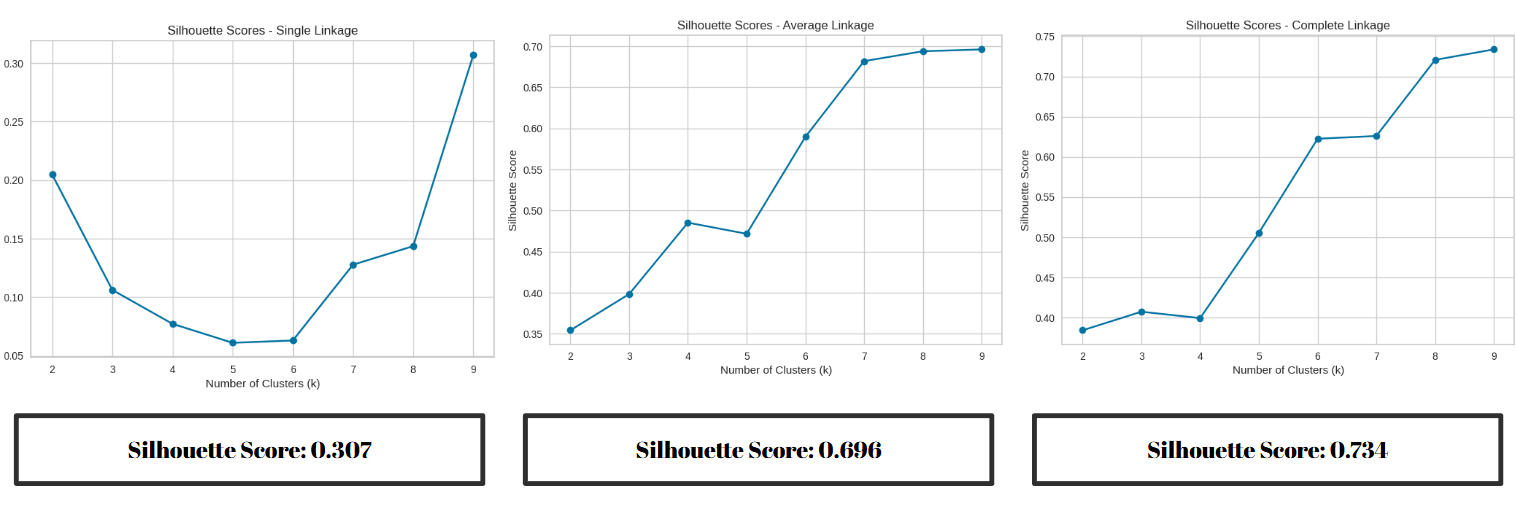


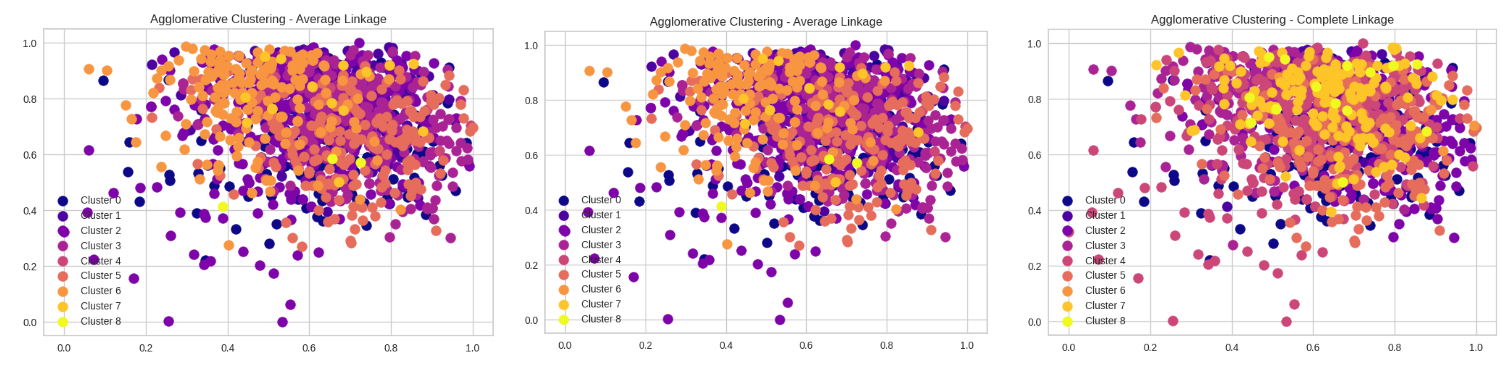
Figure C2.2: Agglomerative Clustering plots

Figure C3: Association rules table (full top 15–20 sorted by lift)

