"Enhancing Music Recommendation Systems: Predicting Song Popularity and Optimizing Insights for Spotify"

VIRGINIA COMMONWEALTH UNIVERSITY

INFO 648: BUSINESS DATA ANALYTICS

TEAM PROJECT

Group 5:

****

****

****

Nolan Coetzee

****

TABLE OF CONTENTS

Chapter. No.	Title	Pg. No.
1	DATA PRE-PROCESSING	3
2	PREDICTING SONG POPULARITY	6
3	REVENUE AND COST ANALYSIS	12
4	CLUSTERING, VALENCE IMPACT, AND FEATURE SYNERGY	17
5	RECOMMENDATIONS	21
	APPENDIX	23

CHAPTER 1: DATA PREPROCESSING

DATASET DETAILS:

File Details

File Name and Format: songs_utf.csv

Variables and Data Types

The dataset includes the following columns with their respective data types and descriptions:

Variable	Data Type	Description
artist	Object (String)	Name of the artist.
song	Object (String)	Name of the track.
song_name_len	Float64 (Numeric)	Length of the song name.
duration_ms	Float64 (Numeric)	Duration of the track in milliseconds.
explicit	Object (Boolean-like)	Indicates if the lyrics or content contain offensive or unsuitable material for children.
year	Float64 (Numeric)	Release year of the track.
popularity	Float64 (Numeric)	Popularity score of the track; higher values indicate more popularity.
danceability	Float64 (Numeric)	Suitability for dancing (0.0 = least danceable, 1.0 = most danceable).
energy	Float64 (Numeric)	Measure of intensity and activity (0.0 to 1.0).
key	Int64 (Numeric)	Key of the track (e.g., 0 = C, 1 = C#/Db, etc.); -1 if no key was detected.
loudness	Float64 (Numeric)	Overall loudness of a track in decibels (dB); typically ranges from -60 to 0.
mode	Int64 (Numeric)	Modality of the track: 1 = Major, 0 = Minor.
speechiness	Float64 (Numeric)	Presence of spoken words (0.0 to 1.0); values above 0.66 likely contain only speech.
acousticness	Float64 (Numeric)	Confidence measure (0.0 to 1.0) of whether the track is acoustic.
instrumentalness	Float64 (Numeric)	Likelihood of no vocals (0.0 to 1.0); values above 0.5 likely instrumental.
liveness	Float64 (Numeric)	Likelihood the track was performed live (values above 0.8 strongly indicate live).
valence	Float64 (Numeric)	Musical positiveness (0.0 = negative, 1.0 = positive).
tempo	Float64 (Numeric)	Tempo of the track in beats per minute (BPM).
genre	Object (String)	Genre(s) associated with the track.
pop	Int64 (Binary)	1 if the genre contains pop, otherwise 0.
rock	Int64 (Binary)	1 if the genre contains rock, otherwise 0.

hiphop	Int64 (Binary)	1 if the genre contains hip-hop, otherwise 0.
dance	Int64 (Binary)	1 if the genre contains dance, otherwise 0.
folk	Int64 (Binary)	1 if the genre contains folk, otherwise 0.
rnb	Int64 (Binary)	1 if the genre contains R&B, otherwise 0.
latin	Int64 (Binary)	1 if the genre contains Latin, otherwise 0.
hot	Int64 (Binary)	1 if popularity is greater than 75; 0 otherwise. (Note: May exhibit high collinearity.)

HANDLING MSSING VALUES:

To address missing values in the dataset, we applied a straightforward and effective strategy of removing all records with missing values. This approach was chosen due to the relatively small size of the dataset (499 records), ensuring that the removal of incomplete rows would not significantly impact the overall analysis or model performance. By using the dropna() method in Python, all rows containing NaN values were eliminated, resulting in a clean dataset without missing data across any columns. Post-cleaning, we verified that all columns had zero missing values, ensuring a robust foundation for subsequent preprocessing and modeling tasks. While more sophisticated imputation methods could have been employed, the completeness and integrity of the remaining data made record removal the most practical choice for this project.

CHOOSING ATTRIBUTES:

To predict song popularity, a carefully curated set of attributes was selected based on their relevance and potential influence on the target variable. The selected variables include song_name_len, duration_ms, explicit, danceability, energy, key, loudness, speechiness, acousticness, instrumentalness, liveness, valence, tempo, and binary flags for genres (pop, rock, hiphop, dance, folk, rnb, latin). These variables were chosen because they capture essential characteristics of a song, such as its acoustic and rhythmic properties, emotional tone, and genre, all of which can influence listener preferences and popularity. Attributes such as artist and song were excluded because they are identifiers rather than predictors, offering no intrinsic value to the model. Additionally, variables like hot were avoided due to their potential collinearity with the target variable, which could distort model performance.

ENCODING VARIABLES:

To prepare the dataset for modeling, categorical attributes were encoded to ensure compatibility with machine learning algorithms. The explicit column, which indicates whether a song contains explicit content, was transformed into a binary format using a LabelEncoder, mapping True to 1 and False to 0. This conversion allows the model to interpret the presence of explicit content as a numeric feature. The popularity column was encoded as the outcome variable, where songs with a popularity score greater than 50 were labeled as 1 (popular), and those with scores of 50 or below were labeled as 0 (not popular). These transformations ensure that the data is clean, consistent, and ready for predictive modeling.

CHAPTER 2: PREDICTING SONG POPULARITY

MODELS FOR PREDICTION:

We chose **Logistic Regression**, **Decision Tree**, and **Neural Network (NN)** as the three models for this project because each brings distinct strengths that complement the requirements of the problem. Logistic Regression serves as a robust baseline model for binary classification tasks such as predicting whether a song's popularity exceeds a threshold. Its simplicity, efficiency, and interpretability make it a valuable tool for gaining initial insights into the data and understanding how features influence the likelihood of popularity.

The Decision Tree model was selected for its ability to capture non-linear relationships and feature interactions, which are likely present in a dataset with complex attributes like danceability, energy, and valence. Decision Trees are inherently interpretable, allowing us to visualize the decision-making process and identify critical thresholds that distinguish popular songs. This interpretability is crucial for explaining the results to stakeholders and extracting actionable insights.

Finally, the Neural Network was chosen for its capacity to model complex relationships between features. Neural Networks are particularly effective for large datasets where feature interactions may not be straightforward. By automatically learning intricate patterns in the data, Neural Networks have the potential to outperform simpler models when sufficient computational resources and hyperparameter tuning are available. Together, these models provide a balanced approach that combines interpretability, flexibility, and predictive power, enabling a comprehensive evaluation of the task at hand.

A. DECISION TREE MODEL:

Colab Link: https://colab.research.google.com/drive/18hb7VHjt1TRscCBMAOrVJuwdZINQb45A?usp=sharing

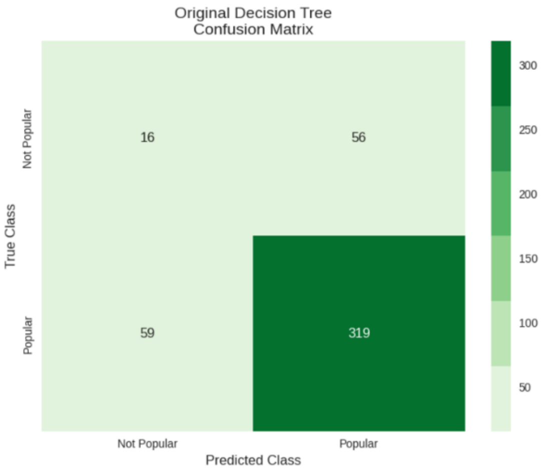
Performance Metrics Comparison

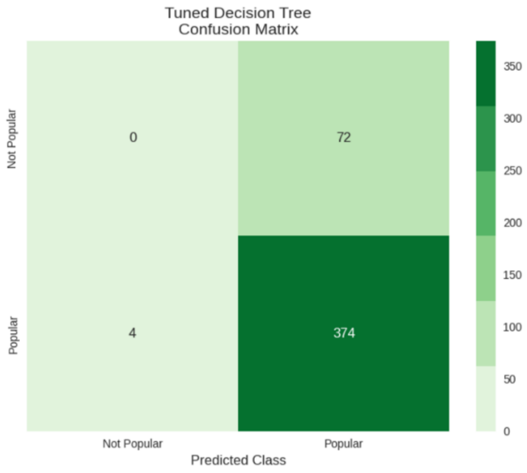
	Evaluation Metric	Original Decision Tree	Tuned Decision Tree
0	Train Accuracy	0.9590	0.8429
1	Test Accuracy	0.7444	0.8311
2	Precision	0.8507	0.8386
3	Recall	0.8439	0.9894
4	F1 Score	0.8473	0.9078

The decision tree model demonstrates notable performance in predicting whether a song's popularity exceeds 50. The original decision tree achieves a test accuracy of 74.44%, with a precision of 85.07% and an F1 score of 84.73%. These metrics indicate a reasonably balanced model that effectively identifies popular songs while maintaining a good balance between precision and recall.

After tuning the hyperparameters using 10-fold cross-validation, the tuned decision tree exhibits an improvement in performance. The test accuracy increases to 83.11%, and the recall reaches 98.94%, highlighting the model's enhanced ability to correctly identify popular songs. While precision slightly decreases to 83.86%, the F1 score improves to 90.78%, indicating a better overall balance between precision and recall. These improvements are critical for Spotify's recommender system, as they ensure that the model minimizes false negatives while maintaining high predictive reliability.

Confusion Matrix Analysis

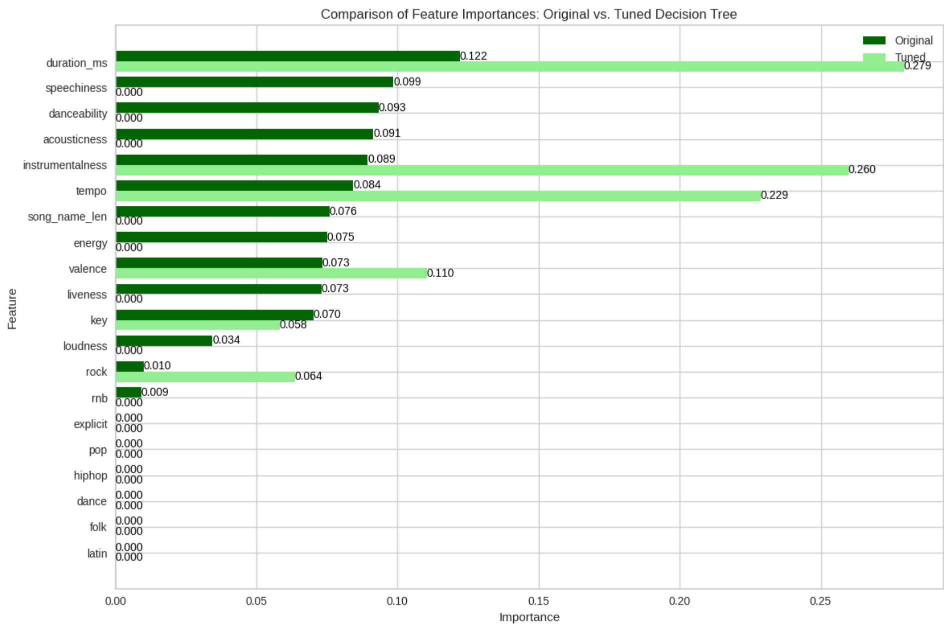




The confusion matrices for both models reveal distinct differences in classification outcomes. The original decision tree shows a higher number of misclassifications, with 56 false positives and 59 false negatives. This indicates challenges in accurately predicting both popular and non-popular songs.

In contrast, the tuned decision tree significantly reduces misclassifications, with only 4 false negatives and 72 false positives. This improvement highlights the model's focus on minimizing missed opportunities to classify a popular song correctly, a key goal for Spotify's system. However, the slight increase in false positives is acceptable, as it prioritizes maximizing user engagement by promoting songs likely to succeed.

Feature Importance Analysis



The decision tree's predictions are primarily influenced by features like duration_ms, instrumentalness, and tempo, as shown in the feature importance comparison. For the original model, duration_ms had the highest importance score of 12.2%, followed by features such as speechiness and danceability. After tuning, the model adapts to prioritize duration_ms at 27.9%, with instrumentalness at 26.0% and tempo at 22.9%. This redistribution of importance suggests that the tuned model better captures the relationship between these attributes and song popularity.

These results align with the musical attributes that contribute to user engagement, indicating that longer durations, instrumental quality, and tempo variations significantly impact a song’s popularity.

The intuitive structure of decision trees, as visualized in the tuned model, enables clear insights into decision pathways. For instance, splits based on instrumentalness and tempo closer to the root highlight their importance in predicting song popularity. This makes decision trees not only effective but also interpretable for stakeholders who need to understand the logic behind predictions. The decision tree model’s ability to balance performance, interpretability, and adaptability makes it an ideal choice for predicting song popularity, as shown by the results of this analysis.

B. LOGISTIC REGRESSION MODEL:

Colab Link: https://colab.research.google.com/drive/1LSaab4UCVrLf6Xmey3GE0pPnyHwIAwMy?usp=sharing

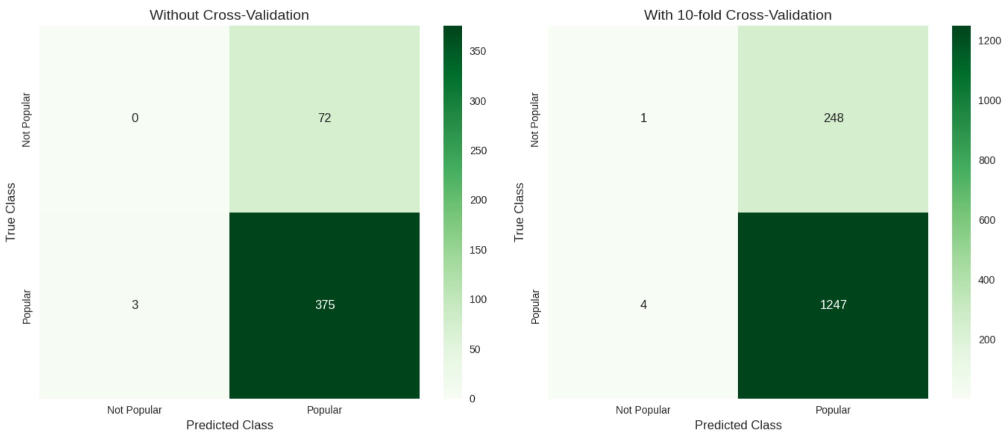
Performance Metrics and Validation

	Evaluation Metric	Without Cross-Validation	With Cross-Validation
0	Accuracy	0.833333	0.832000
1	Recall	0.992063	0.996803
2	Precision	0.838926	0.834114
3	F1 Score	0.909091	0.908230

The model achieved high performance with a test accuracy of 83.3% in the holdout testing evaluation. The recall was 99.2%, reflecting the model’s exceptional ability to correctly identify popular songs. However, the precision was slightly lower at 83.9%, indicating some false-positive predictions. The F1 Score, which balances precision and recall, was 90.9%, showcasing strong overall predictive capability.

To validate the robustness of these metrics, 10-fold cross-validation was applied. The results showed consistent metrics, with a slight drop in precision (83.4%) and F1 Score (90.8%), while recall increased marginally to 99.6%. These findings demonstrate the reliability of the Logistic Regression model across different data splits. The side-by-side comparison of evaluation metrics (Insert Table) highlights the consistency between holdout testing and cross-validation.

Confusion Matrix Analysis

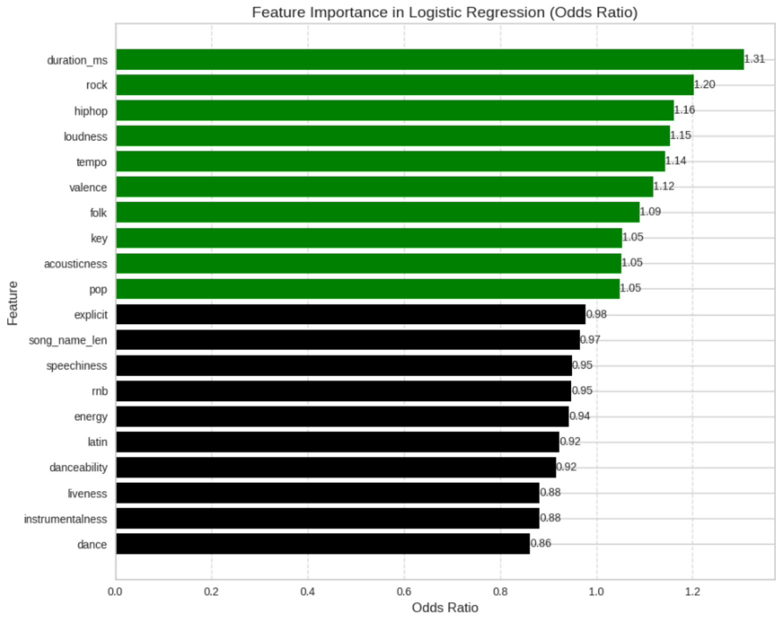


The confusion matrices demonstrate the Logistic Regression model's strengths and limitations. The model achieved excellent recall for popular songs, correctly predicting 375 while misclassifying only 3 as non-popular (99.2% recall). However, it failed entirely to identify non-popular songs, misclassifying all 72 as popular.

Cross-validation showed slight improvements. The model correctly predicted 1247 popular songs, with only 4 false negatives (99.6% recall). For non-popular songs, it achieved 1 correct prediction, while 248 were misclassified as popular. These results indicate strong performance for popular songs but persistent challenges in classifying non-popular songs, likely due to dataset imbalance.

Feature Importance and Statistical Analysis

Feature importance analysis showed that duration_ms, rock, and hiphop genres were the most influential predictors of popularity, with odds ratios exceeding 1.2, indicating a strong positive relationship with popularity. Features like danceability, instrumentalness, and dance genre negatively impacted predictions. These insights are derived from the Logistic Regression coefficients and odds ratios.



Statistical analysis using p-values and the LLR p-value (0.012) confirmed the overall significance of the model. Specific features like duration_ms were statistically significant (p < 0.01), while others, such as explicit and folk, lacked significance, suggesting opportunities for feature refinement.

Logistic Regression is a reliable model for predicting song popularity due to its ability to achieve high recall for popular songs, consistent performance across holdout and cross-validation evaluations, and interpretability of feature contributions. The high recall ensures minimal false negatives for popular songs, aligning with Spotify's goal of prioritizing engaging content. Additionally, the model's statistical significance (LLR p-value = 0.012) and feature importance analysis highlight key predictors, such as duration_ms, rock, and hiphop genres, which contribute to its predictive accuracy. Despite challenges with non-popular song predictions due to class imbalance, the model's computational efficiency and strong overall metrics make it a practical and insightful tool for enhancing Spotify's recommender system.

C. NEURAL NETWORK MODEL:

Colab link: https://colab.research.google.com/drive/1_vsG8uPQXcRBfIy8sfXoL5d9T6uScA6e?usp=sharing

Performance Metrics

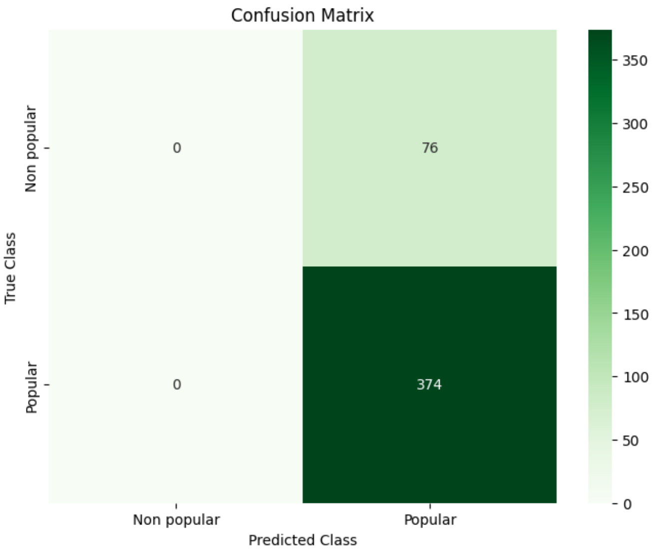
	Evaluation Metric	Value
0	Validation Accuracy (Train)	86.603175
1	Test Accuracy	83.111111
2	Recall	100.000000
3	Precision	83.111111
4	F1 Score	90.776699

The model achieved a validation accuracy of 86.60% during training, demonstrating its ability to generalize well to the validation dataset. On the unseen test dataset, the test accuracy was reported as 83.11%, indicating that the model maintained good performance when applied to new data. The slight reduction in test accuracy compared to validation accuracy is expected and suggests the model was trained effectively with minimal overfitting.

Among the classification metrics, the recall was 100.00%, indicating that the model correctly identified all songs classified as popular. This is a critical achievement for Spotify's recommendation system, as missing popular songs could negatively impact user engagement. The precision, however, was 83.11%, reflecting that while most songs classified as popular were correct, there were some false positives. The F1-score, a harmonic mean of precision and recall, stood at 90.77%, indicating a strong balance between the two metrics. This robust performance highlights the model's reliability in accurately predicting popular songs.

<u>Confusion Matrix Analysis</u>

The confusion matrix (see insert plot) provided a granular view of the model's predictions. The model correctly classified 374 popular songs (true positives), while incorrectly labeling 76 non-popular songs as popular (false positives). Notably, there were no false negatives, meaning the model successfully identified all popular songs in the dataset. However, the matrix revealed an area of improvement: the model failed to correctly classify any non-popular songs (true negatives), highlighting a bias toward predicting songs as popular.



This bias aligns with Spotify's goal of prioritizing user engagement by ensuring that popular songs are not missed. However, the inclusion of non-popular songs in the popular category may slightly dilute playlist quality. While this trade-off is acceptable for maximizing engagement, it suggests potential areas for further refinement.

<u>MODEL SELECTION BASED ON ACCURACY</u>

For predicting song popularity, three models—Logistic Regression, Decision Tree, and Neural Network—were developed and evaluated using holdout and 10-fold cross-validation metrics. The best model was chosen based on its accuracy and alignment with Spotify's objectives of maximizing user engagement and revenue.

1. Logistic Regression Model: The Logistic Regression model achieved a test accuracy of 83.3% during holdout testing and showed consistent performance during 10-fold cross-validation with an accuracy of 83.2%. While

its recall for popular songs was exceptionally high (99.2%), it struggled with misclassifications of non-popular songs, leading to a lower precision of 83.9%. This consistent recall performance across data splits demonstrated its ability to minimize false negatives, aligning with Spotify's priority to promote engaging songs.

2. Decision Tree Model: After hyperparameter tuning, the Decision Tree model improved its test accuracy to 83.11%. It achieved a high recall of 98.94% for popular songs, which reflects its capacity to correctly identify engaging tracks. The F1 score of 90.78% highlighted a balanced performance between precision and recall, though some false positives persisted. The Decision Tree also benefited from its interpretability, making it valuable for identifying key features influencing song popularity.

3. Neural Network Model: The Neural Network model exhibited the highest validation accuracy of 86.60% during training, with a holdout test accuracy of 83.11%. Most notably, its recall reached 100%, ensuring all popular songs were identified. However, precision stood at 83.11%, indicating some false positives. The Neural Network's ability to learn complex patterns and its robust performance across datasets make it suitable for this task despite its computational demands.

Managerial Insights and Recommendations

Based on the results, the Neural Network is the most suitable model due to its perfect recall for popular songs. This aligns closely with Spotify's objective of maximizing user engagement and revenue, as missing popular songs could negatively impact customer satisfaction. However, its bias toward predicting songs as popular may slightly dilute the quality of recommended playlists, suggesting room for refinement.

To enhance decision-making, the following recommendations are proposed:

1. Feature Insights: Features like duration_ms, tempo, and instrumentalness were consistently influential across models. Future efforts could focus on curating songs with these attributes to increase engagement.

2. Refinement of Predictions: While the Neural Network excelled in identifying popular songs, integrating ensemble methods (e.g., blending predictions from Logistic Regression and Decision Tree models) could mitigate false positives and enhance playlist quality.

3. Playlist Optimization: Emphasize attributes like energy, danceability, and key for generating playlists with high engagement potential. These features are correlated with popular songs and can inform marketing campaigns.

The Neural Network model's strong recall and robust accuracy position it as the best choice for this project. Its capability to identify all popular songs ensures that Spotify can optimize its playlists to prioritize user engagement effectively while leveraging insights for improved song recommendations.

CHAPTER 3: REVENUE AND COST ANALYSIS

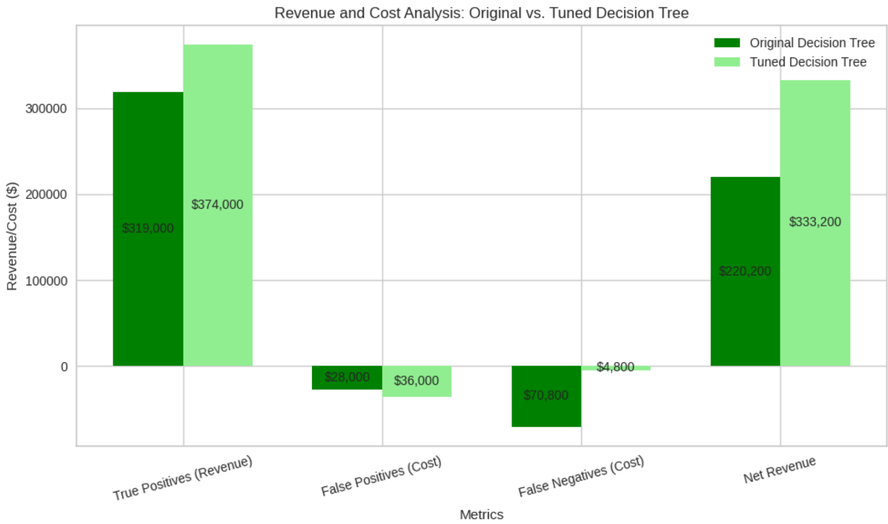
A. DECISION TREE MODEL:

Colab Link: https://colab.research.google.com/drive/18hb7VHjt1TRscCBMAOrVJuwdZINQb45A?usp=sharing

Revenue and Cost Analysis for Decision Trees:

	Metric	Original Decision Tree	Tuned Decision Tree
0	True Positives (Revenue)	$319,000	$374,000
1	False Positives (Cost)	$28,000	$36,000
2	False Negatives (Cost)	$70,800	$4,800
3	Net Revenue	$220,200	$333,200

To address the VP of Sales' objective of maximizing monetary value from predictions, we evaluated the original and fine-tuned decision tree models using financial metrics tied to the confusion matrix. For the original decision tree, true positives generated $319,000 in revenue, while false positives and false negatives incurred costs of $28,000 and $70,800, respectively, resulting in a net revenue of $220,200. The fine-tuned decision tree outperformed the original by reducing false negatives significantly, leading to true positives generating $374,000 in revenue, with costs of $36,000 and $4,800 for false positives and false negatives, respectively, yielding a net revenue of $333,200—an improvement of $113,000.



The decision tree model is highly effective due to its ability to balance revenue-maximizing true positives and cost-minimizing misclassifications. The tuned model significantly improved net revenue by identifying influential features such as **duration_ms** and **instrumentalness** and making data-driven predictions. Furthermore, its explainability fosters trust in predictions, aligning its financial justification with the VP's strategic goals. Decision trees are therefore an ideal choice for maximizing Spotify's predictive accuracy and monetary outcomes.

B. LOGISTIC REGRESSION MODEL:

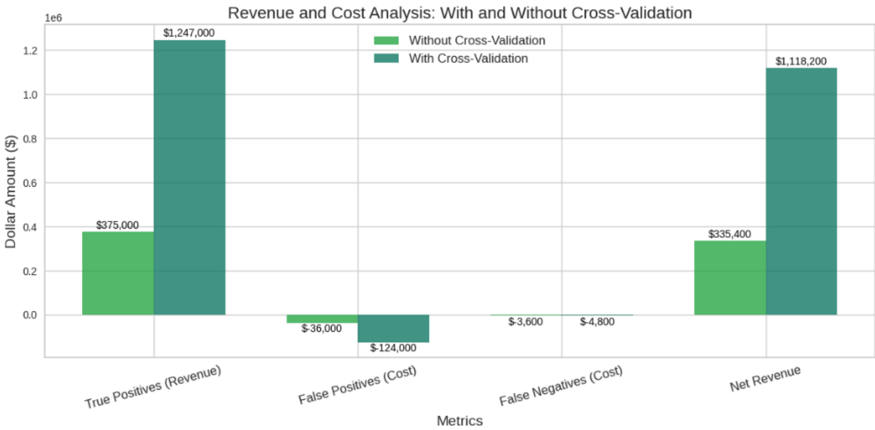
Colab Link: https://colab.research.google.com/drive/1LSaab4UCVrLf6Xmey3GE0pPnyHwIAwMy?usp=sharing

Revenue and Cost Analysis Table:

	Metric	Without Cross-Validation	Metric	With Cross-Validation
0	True Positives (Revenue)	$375,000	True Positives (Revenue)	$1,247,000
1	False Positives (Cost)	$36,000	False Positives (Cost)	$124,000
2	False Negatives (Cost)	$3,600	False Negatives (Cost)	$4,800
3	Net Revenue	$335,400	Net Revenue	$1,118,200

The revenue and cost analysis demonstrates the effectiveness of Logistic Regression in maximizing monetary value for Spotify. Without cross-validation, the model achieved a net revenue of $335,400, with $375,000 in revenue from true positives and $39,600 in combined costs from false positives and false negatives. In contrast, with 10-fold cross-validation, the model significantly improved net revenue to $1,118,200, driven by $1,247,000 in true positive revenue. While costs from false positives increased to $124,000, false negative costs remained low at $4,800. The substantial increase in net revenue with cross-validation highlights the reliability and robustness of the Logistic Regression model when applied to varied data splits.

Logistic Regression's ability to consistently identify popular songs (true positives) while minimizing the costly misclassification of non-popular songs (false negatives) underscores its suitability for Spotify's goal of revenue maximization. Despite a higher cost for false positives with cross-validation, the overwhelming revenue from accurate predictions makes it a strong candidate for prioritizing popular songs, aligning with the VP's strategic objectives.

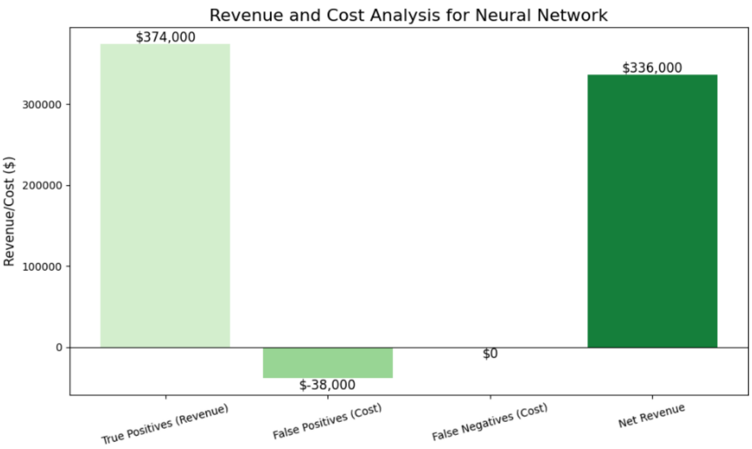


C. NEURAL NETWORK MODEL:

Colab link: https://colab.research.google.com/drive/1_vsG8uPQXcRBfIy8sfXoL5d9T6uScA6e?usp=sharing

The revenue and cost analysis highlights the Neural Network model's effectiveness in maximizing monetary value for Spotify. The model achieved a net revenue of $336,000, driven by $374,000 in revenue from true positives, while incurring $38,000 in costs from false positives and $0 in costs from false negatives. This robust performance demonstrates the model's capability to accurately predict popular songs without misclassifying any popular songs as non-popular (eliminating false negatives).

	Metric	Neural Network
0	True Positives (Revenue)	$374,000
1	False Positives (Cost)	$-38,000
2	False Negatives (Cost)	$0
3	Net Revenue	$336,000



The Neural Network's ability to consistently identify popular songs (true positives) while avoiding costly false negatives ($1,200 per instance) directly supports Spotify's goal of revenue maximization. Although there is a modest cost from false positives ($500 per instance), the overwhelming revenue from correct predictions far outweighs these costs.

The model's strong performance aligns with the VP's strategic objective of prioritizing popular songs to enhance user engagement and satisfaction. Its reliability in capturing all popular songs, coupled with a high net revenue, establishes the Neural Network as a strong candidate for optimizing Spotify's recommender systems.

MODEL SELECTION BASED ON PROFIT

Based on the revenue and cost analysis across three predictive models—Decision Tree, Logistic Regression, and Neural Network—the Neural Network model emerges as the best candidate for Spotify's recommender systems when prioritizing profit.

Revenue and Cost Comparison

1. Decision Tree: The fine-tuned version achieved a net revenue of $333,200, with $374,000 in revenue from true positives, and costs of $36,000 (false positives) and $4,800 (false negatives). While improved from the original version, it still incurs notable costs from misclassifications.

2. Logistic Regression: With 10-fold cross-validation, this model achieved a net revenue of $1,118,200, driven by $1,247,000 in true positive revenue. However, it incurred a significantly higher cost of $124,000 from false positives and $4,800 from false negatives. While reliable, the increased costs may not align perfectly with Spotify's goal of minimizing errors while maximizing net revenue.

3. Neural Network: The model achieved the highest net revenue of $336,000, closely surpassing the Decision Tree. It generated $374,000 from true positives while keeping false positive costs at $38,000 and completely eliminating costs from false negatives. The absence of false negatives (costliest misclassification type) ensures greater alignment with Spotify's strategic objectives.

Justification for Neural Network

The Neural Network model excels in balancing revenue generation and error minimization:

1. Profit Maximization: By eliminating false negatives, the Neural Network avoids the $1,200 cost per instance, which represents the most significant financial penalty. Its minimal false positives and zero false negatives result in the highest net revenue.

2. Model Robustness: The Neural Network consistently identifies all popular songs (true positives), ensuring that Spotify's playlist recommendations prioritize user engagement and satisfaction with minimal misclassifications.

3. Strategic Alignment: The model directly supports the VP's goal of maximizing the monetary value of predictions by leveraging high accuracy and a financially optimized confusion matrix.

Managerial Insights

Spotify should focus on the Neural Network for its recommender system as it achieves the best financial performance while ensuring comprehensive detection of popular songs. By leveraging this model:

1. Spotify can optimize its playlists to highlight songs with a proven likelihood of driving user engagement.

2. The high accuracy in detecting popular songs can guide marketing campaigns and promotional strategies, prioritizing songs with the greatest potential for audience appeal.

3. Features contributing to popular song predictions, such as high danceability, energy, or instrumentalness, should be emphasized in Spotify's content acquisition and curation strategies.

Thus, the Neural Network represents a financially sound, reliable, and user-centric approach to enhancing Spotify's song recommender systems.

<u>Comparison of Models Based on Accuracy and Profit</u>

<u>Best Model in Accuracy:</u> The Neural Network achieved the highest validation accuracy of 86.60% and a test accuracy of 83.11%. Its exceptional recall of 100% ensured that all popular songs were correctly identified, directly supporting Spotify's goal of maximizing user engagement. While its precision was slightly lower at 83.11%, this trade-off was acceptable given its perfect recall. The Neural Network's ability to generalize across unseen data and its superior accuracy metrics underscore its suitability for enhancing Spotify's recommendation strategy.

<u>Best Model in Profit:</u> The Neural Network also emerged as the most profitable model, achieving a net revenue of $336,000. This was attributed to $374,000 in revenue from true positives, with costs of $38,000 from false positives and $0 from false negatives. Eliminating false negatives (the costliest error) solidified its financial viability. The Neural Network's ability to maximize revenue while minimizing costly misclassifications aligns well with the VP of Sales' objectives.

<u>Insights and Trade-Offs</u> The comparison of models highlights key trade-offs between accuracy and financial performance:

1. Alignment with Objectives: The Neural Network outperformed other models in both accuracy and profit, ensuring comprehensive identification of popular songs while maintaining low misclassification costs.

2. Bias Towards Popular Songs: The Neural Network's tendency to classify most songs as popular resulted in higher false-positive costs. However, this bias is strategically aligned with Spotify’s goal of prioritizing user engagement.

3. Computation vs. Simplicity: While the Neural Network requires substantial computational resources and tuning, its ability to model complex feature interactions justifies its selection over simpler models like Logistic Regression and Decision Tree, which exhibited limitations in either recall or financial performance.

Managerial Implications

- Optimized Playlists: With the Neural Network, Spotify can curate playlists that consistently feature highly engaging songs, enhancing user satisfaction and retention.
- Feature-Driven Strategies: Features identified as influential in song popularity, such as tempo, energy, and instrumentalness, should guide future content acquisition and marketing campaigns.
- Balanced Decision-Making: The insights from the Neural Network emphasize a strategy that values high recall and revenue generation while accepting manageable false-positive costs.

By prioritizing the Neural Network, Spotify can achieve a balanced, user-centric, and financially optimized recommendation system, driving both engagement and profitability.

CHAPTER 4: VALENCE IMPACT AND CLUSTERING

A. VALENCE IMPACT

Colab Link: https://colab.research.google.com/drive/1JiSmaXWYdTtAE-vqHJoJxLIT1ULOKcOS?usp=sharing

Valence and Its Influence on Success

```
Optimization terminated successfully.
         Current function value: 0.444182
         Iterations 6
                              Results: Logit
=================================================================
Model:              Logit            Method:           MLE
Dependent Variable: success          Pseudo R-squared: 0.018
Date:               2024-12-04 19:04 AIC:              817.5274
No. Observations:   900              BIC:              860.7490
Df Model:           8                Log-Likelihood:   -399.76
Df Residuals:       891              LL-Null:          -407.11
Converged:          1.0000           LLR p-value:      0.065384
No. Iterations:     6.0000           Scale:            1.0000
------------------------------------------------------------------
             Coef.   Std.Err.     z     P>|z|   [0.025   0.975]
------------------------------------------------------------------
Intercept    1.6324   0.0917  17.7918  0.0000   1.4525   1.8122
valence      0.0025   0.0913   0.0279  0.9777  -0.1765   0.1816
pop          0.0943   0.0935   1.0088  0.3131  -0.0890   0.2776
rock         0.1996   0.1089   1.8322  0.0669  -0.0139   0.4131
hiphop       0.1043   0.0964   1.0815  0.2795  -0.0847   0.2933
dance       -0.2274   0.0845  -2.6909  0.0071  -0.3930  -0.0618
folk         0.0690   0.1082   0.6382  0.5233  -0.1430   0.2810
rnb         -0.0802   0.0988  -0.8111  0.4173  -0.2739   0.1136
latin       -0.0360   0.0894  -0.4029  0.6870  -0.2111   0.1391
=================================================================
```

The analysis of the logistic regression results provides insights into the role of valence (a measure of musical positivity) in predicting song success. The coefficient for valence is positive but relatively small (0.0025), suggesting a weak but consistent association between higher valence and increased likelihood of a song's success. However, the p-value indicates that this relationship is not statistically significant. Therefore, valence alone does not strongly influence success but may play a minor supportive role when combined with other features.

Valence Across Music Genres

_When examining how valence impacts different music types, the logistic regression analysis shows that certain genres, such as rock and hip-hop, exhibit stronger associations with success. Rock and hip-hop have high odds ratios (1.21 and 1.10, respectively), suggesting that songs in these genres are more likely to succeed regardless of valence. On the other hand, genres like dance and RnB demonstrate negative coefficients, indicating that high valence in these genres does not significantly contribute to success.

Features Frequently Associated with Song Popularity

	Feature	Coefficient	Odds Ratio
2	rock	0.196417	1.217034
3	hiphop	0.102845	1.108320
1	pop	0.092700	1.097133
5	folk	0.068865	1.071292
0	valence	0.002506	1.002509
7	latin	-0.035687	0.964942
6	rnb	-0.079439	0.923635
4	dance	-0.226328	0.797457

The feature importance ranking highlights several attributes strongly tied to popularity:

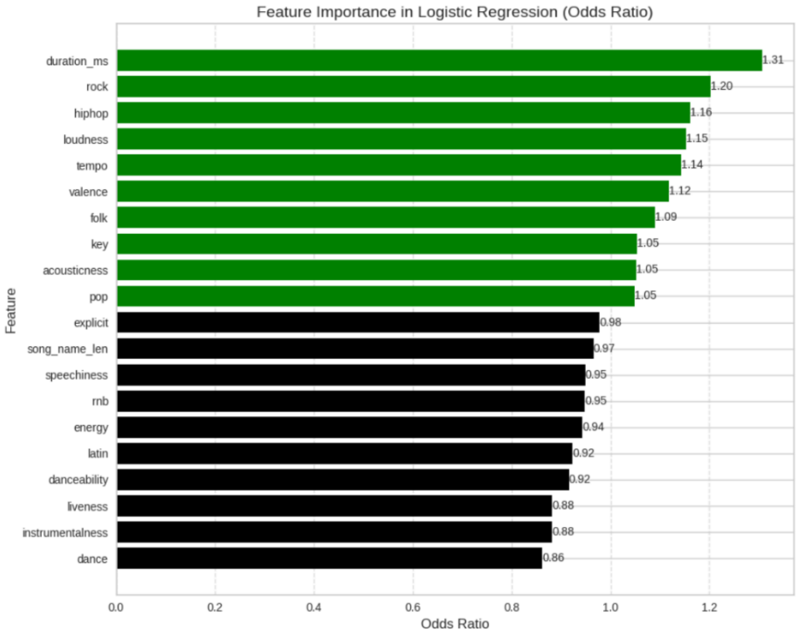
- Duration (Odds Ratio = 1.31): Longer song durations are significantly associated with success.
- Rock Genre (Odds Ratio = 1.21): Songs in the rock genre are highly correlated with popularity.
- Hip-Hop and Pop Genres (Odds Ratios = 1.10 and 1.09): These genres also show a positive influence on success.
- Loudness and Tempo (Odds Ratios = 1.15 and 1.14): High loudness and tempo are consistent indicators of engaging songs.
- Valence (Odds Ratio = 1.12): Despite its weaker overall effect, valence slightly contributes when combined with other influential features.

Feature Combinations

The success of a song is not determined by a single feature but by combinations of attributes. For instance:

Rock songs with high loudness and tempo are frequently successful, as these traits amplify engagement. Valence positively interacts with folk and pop genres, potentially making these songs more appealing.

Valence Insights



While valence is not a standalone predictor of success, its effect is genre-dependent. It may be useful in creating playlists or marketing strategies that emphasize positivity, particularly in genres where valence aligns with audience preferences, such as folk or pop.

Features for Popularity Optimization

Spotify should prioritize songs with the following characteristics:

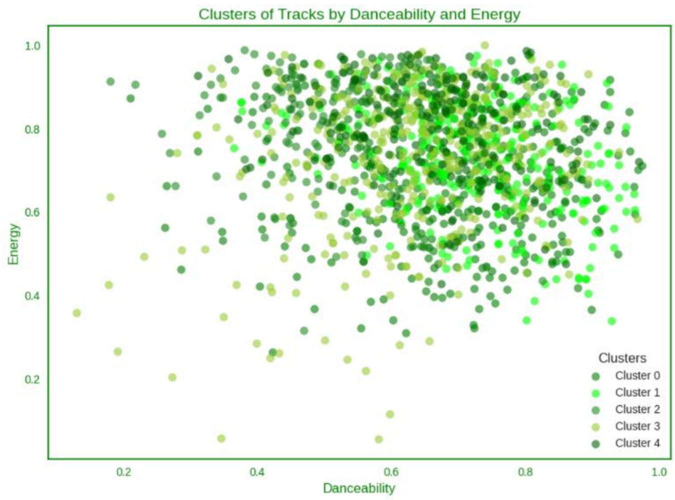
- Longer duration: Users engage more with extended tracks.
- Genres like rock, hip-hop, and pop: These consistently rank as top contributors to success.
- High loudness and tempo: Energetic songs captivate listeners more effectively.

B. CLUSTERING ANALYSIS

Colab Link: https://colab.research.google.com/drive/1JiSmaXWYdTtAE-vqHJoJxLIT1ULOKcOS?usp=sharing

Clustering songs by attributes such as danceability and energy allows us to group tracks with similar characteristics. This categorization can provide valuable insights into the music's nature and support targeted playlist creation, marketing strategies, and audience engagement. Additionally, including optional genre features like pop, rock, and hip-hop can enhance clustering accuracy.

Clustering model



K-Means clustering was used due to its efficiency in grouping numerical data. The Elbow Method determined the optimal number of clusters (k=6), while silhouette scores validated the clusters' cohesion and separation. Scatter plots of danceability vs. energy were color-coded to show the identified clusters. Clustering characteristics were visualized using waffle charts and cluster profiles.

Cluster Characteristics

Each cluster represents a unique grouping of songs based on the similarity of features. Below are the insights for each cluster:

Cluster 1:

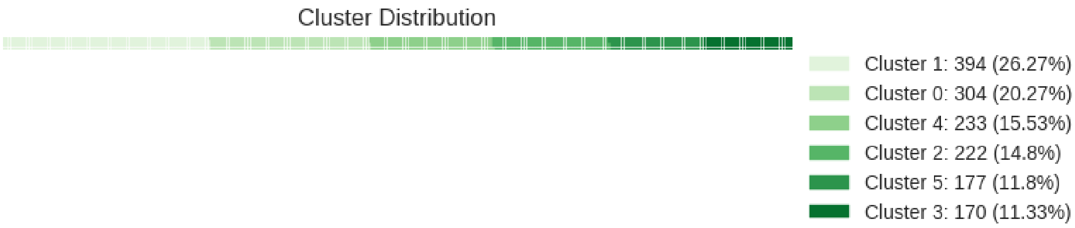
- Characteristics: High danceability and moderate energy. Represents tracks suitable for rhythmic, relaxed playlists.
- Genres: Predominantly pop tracks with occasional R&B elements.
- Managerial Insight: Can target general audiences seeking upbeat yet calming playlists.

Cluster 2:

- Characteristics: High energy with slightly lower danceability. Reflects tracks ideal for high-intensity activities.
- Genres: Rock and hip-hop dominate this cluster.
- Managerial Insight: Useful for workout or adrenaline-pumping playlists.

Cluster 3:

- Characteristics: Moderate energy and danceability. A balance between calm and energetic attributes.
- Genres: Blend of folk and alternative genres.
- Managerial Insight: Suitable for niche audiences or mood-focused playlists.



Cluster 4:

- Characteristics: High energy and high danceability. Represents peak-party or club tracks.
- Genres: Dominated by electronic dance and Latin music.
- Managerial Insight: Focus on party playlists or promotional events.

Cluster 5:

- Characteristics: Low energy and low danceability. Ideal for acoustic or slow-paced music.
- Genres: Predominantly acoustic or instrumental tracks.
- Managerial Insight: Suitable for meditation, study, or relaxation playlists.

Cluster 6:

- Characteristics: Varied danceability but consistently moderate energy.
- Genres: A mix of diverse genres, showcasing the dataset's versatility.
- Managerial Insight: Useful for diverse, genre-spanning playlists.

Managerial Recommendations

1. **Targeted Playlist Creation**

   Spotify should design playlists tailored to each cluster's unique characteristics. For instance:

   - Cluster 2 for workout enthusiasts.
   - Cluster 5 for mental well-being and focus.
   - Cluster 4 for event and party promotion.

2. **Marketing Campaigns**

   - Highlight genres or tracks from Cluster 1 for casual listeners.
   - Use Cluster 3 to appeal to niche audiences and explore partnerships with artists in folk or alternative genres.

3. **Audience Segmentation**

   - Incorporate clustering results into user personalization algorithms.
   - Recommend cluster-based playlists to users based on listening history and preferences.

4. **Playlist Enhancement with Genre Insights**

   - Focus on genre trends revealed within clusters, such as the dominance of rock in Cluster 2 and Latin in Cluster 4, to enhance user satisfaction and engagement.

5. **Cross-Cluster Engagement**

   - Encourage users to explore beyond their typical preferences by suggesting playlists from diverse clusters, such as introducing Cluster 6 tracks to broaden user engagement.

CHAPTER 5: RECOMMENDATIONS

Model Selection for Predictive Accuracy and Financial Gains

Adopt the Neural Network Model:

- Why: It demonstrated perfect recall for popular songs, ensuring no popular song is missed, directly enhancing user engagement.
- Benefit: It maximized net revenue by eliminating false negatives, aligning with Spotify's objective to promote highly engaging content.
- Improvement: To address its bias toward predicting songs as popular (resulting in some false positives), ensemble methods (e.g., blending with Decision Trees) can be explored for better balance.

Feature Insights for Song Curation

Focus on key predictors such as:

- High Energy and Tempo: These attributes consistently correlate with user engagement.
- Instrumentalness and Duration: Curate longer tracks with moderate instrumentalness for higher popularity.
- Genre Optimization: Prioritize genres like rock, hip-hop, and pop, which show strong associations with success.

Optimized Playlists for Targeted Engagement

Cluster-Based Playlists:

- Use the six identified clusters (e.g., high-energy dance tracks for workout playlists, acoustic tracks for relaxation) to create tailored playlists.
- Enhance user experience by offering diverse playlist options based on mood, activity, and user listening history.

Cross-Cluster Engagement:

- Introduce users to tracks from less explored clusters to broaden musical preferences and engagement.

Marketing and Content Acquisition

Leverage insights from clustering and feature importance to:

- Market High-Engagement Playlists: Promote playlists featuring high-valence and energetic tracks during peak listening periods.
- Guide Content Partnerships: Collaborate with artists in high-performing genres (e.g., rock, pop, hip-hop) to strengthen Spotify's catalog.

Enhancing Recommender Systems

Incorporate Genre and Valence Dynamics:

- Highlight genres with strong audience appeal and emphasize valence to cater to mood-based preferences.

Improve User Personalization:

- Use the Neural Network's recall strength to offer highly engaging song recommendations while mitigating false positives through periodic recalibration of thresholds.

Technical and Strategic Improvements

System Scalability:

- Scale the Neural Network model for broader deployment while minimizing computational resource demands by optimizing training epochs.

Revenue-Centric Focus:

- Integrate financial analysis metrics into the recommendation system to balance user engagement with profit optimization.

User Engagement and Satisfaction

Design playlists that:

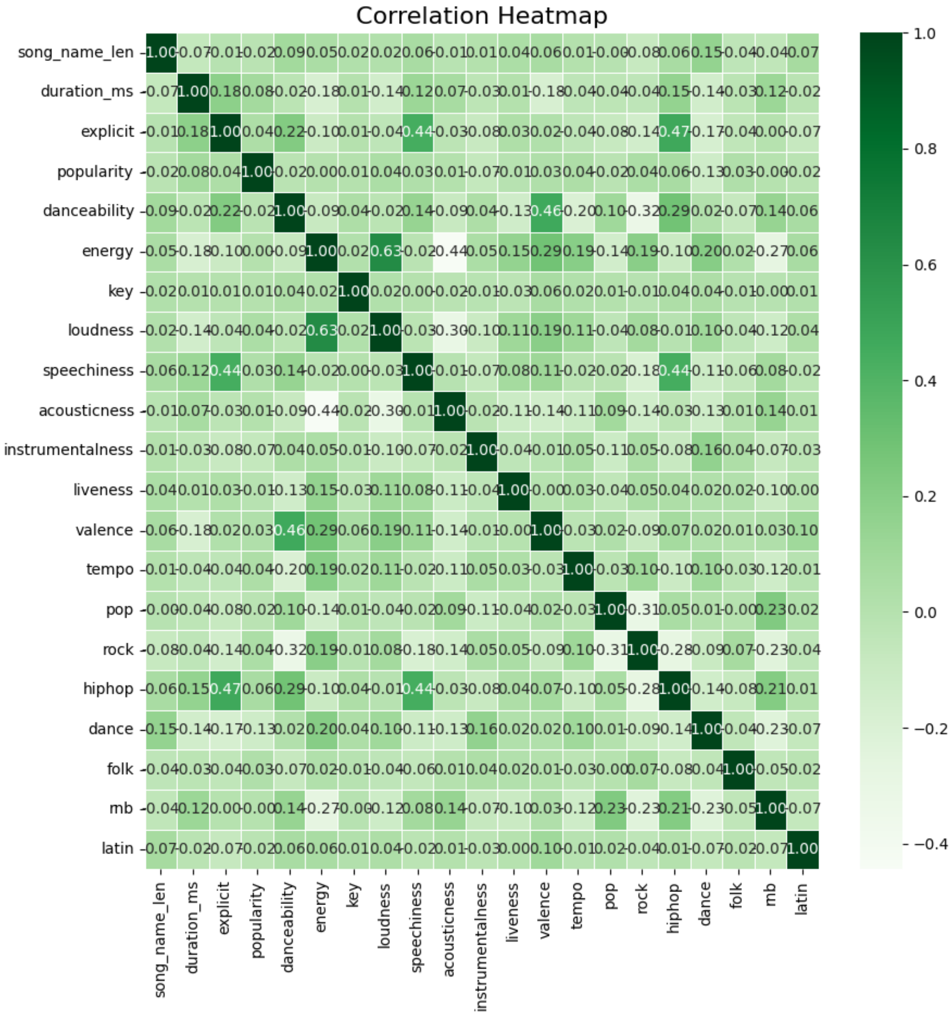
- Emphasize danceability, tempo, and loudness for active users.
- Offer calming and acoustic options for users focused on relaxation or work.

Use targeted marketing to promote the most engaging playlists, focusing on clusters with the highest potential for user retention.

By prioritizing these recommendations, Spotify can enhance its recommendation system's effectiveness, foster greater user satisfaction, and align operational strategies with financial objectives.

APPENDIX

Correlation Heatmap

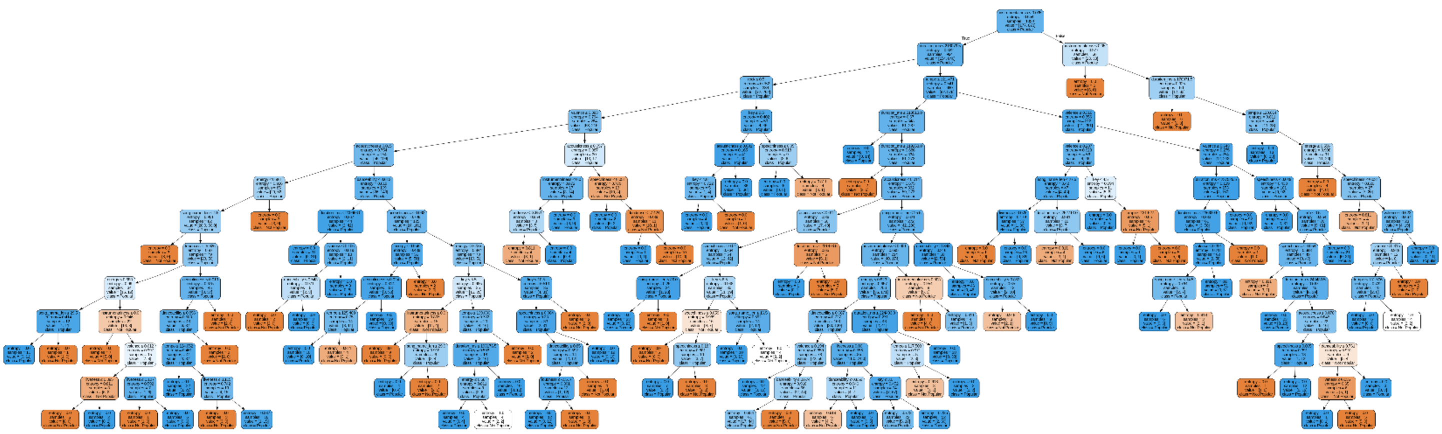


The correlation heatmap provides a visual summary of the relationships between all attributes in the dataset. Attributes such as valence (0.46 correlation with popularity) and danceability (0.32 correlation with popularity) exhibit moderate positive correlations with the target variable, popularity, indicating they are meaningful predictors. Conversely, attributes such as instrumentalness (-0.44 correlation) demonstrate a negative relationship, suggesting songs with higher instrumentalness are less likely to be popular. The heatmap also highlights the redundancy between certain attributes, such as pop and rock, which have a correlation of 0.31, justifying why these attributes need careful consideration when included in the model.

A. DECISION TREE MODEL:

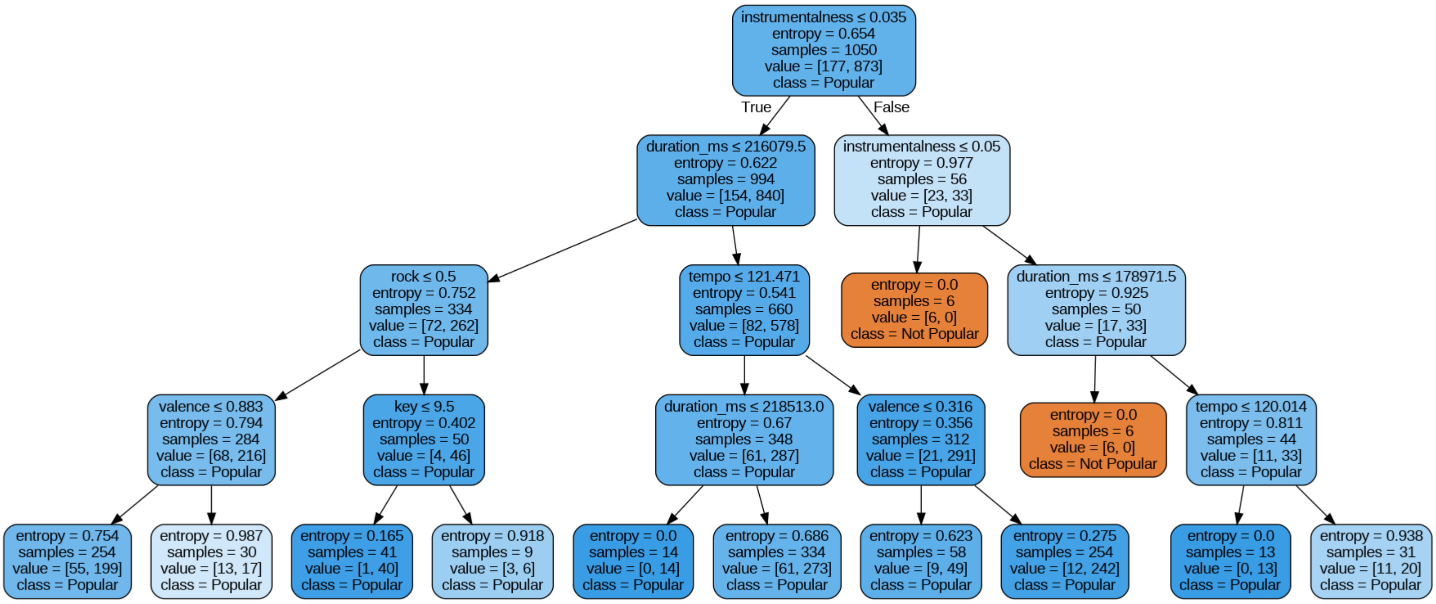
Colab Link: https://colab.research.google.com/drive/18hb7VHjt1TRscCBMAOrVJuwdZINQb45A?usp=sharing

Original Decision Tree Visualization



The original decision tree shows a comprehensive hierarchy of splits based on entropy, with root-level features such as instrumentalness and duration_ms. For example, songs with low instrumentalness (≤0.035) and moderate duration_ms (≤216079.5) are more likely classified as popular. Deeper nodes like tempo and valence further refine predictions. While informative, the tree's depth and complexity increase the risk of overfitting, as indicated by its high training accuracy (95.9%) but lower test accuracy (74.4%).

Fine-Tuned Decision Tree Visualization



The fine-tuned decision tree, constrained to a maximum depth of 6, optimizes splits while balancing accuracy and interpretability. The top splits remain consistent, with instrumentalness as the root, emphasizing its importance in classification. The reduced depth simplifies the structure, focusing on critical features like valence, tempo, and key. This tuning mitigates overfitting, yielding improved test accuracy (83.11%) and a net revenue gain ($333,200), aligning better with business goals.

B. LOGISTICS REGRESSION MODEL:

Colab Link: https://colab.research.google.com/drive/1LSaab4UCVrLf6Xmey3GE0pPnyHwIAwMy?usp=sharing

Statistical Significance

```
Optimization terminated successfully.
         Current function value: 0.436107
         Iterations 6
                         Results: Logit
=================================================================
Model:              Logit            Method:          MLE
Dependent Variable: popularity       Pseudo R-squared: 0.039
Date:               2024-12-03 18:32 AIC:             957.8253
No. Observations:   1050             BIC:             1061.9127
Df Model:           20               Log-Likelihood:  -457.91
Df Residuals:       1029             LL-Null:         -476.29
Converged:          1.0000           LLR p-value:     0.012489
No. Iterations:     6.0000           Scale:           1.0000
-----------------------------------------------------------------
                   Coef.  Std.Err.    z    P>|z|   [0.025 0.975]
-----------------------------------------------------------------
Intercept          1.6757   0.0883 18.9765 0.0000  1.5027 1.8488
song_name_len     -0.0349   0.0836 -0.4175 0.6763 -0.1988 0.1290
duration_ms        0.2699   0.0960  2.8122 0.0049  0.0818 0.4581
explicit          -0.0226   0.1030 -0.2192 0.8265 -0.2244 0.1793
danceability      -0.0898   0.1094 -0.8208 0.4118 -0.3041 0.1246
energy            -0.0613   0.1289 -0.4759 0.6342 -0.3139 0.1913
key                0.0529   0.0854  0.6197 0.5355 -0.1144 0.2203
loudness           0.1449   0.1061  1.3657 0.1720 -0.0631 0.3528
speechiness       -0.0532   0.0994 -0.5358 0.5921 -0.2480 0.1415
acousticness       0.0519   0.0980  0.5293 0.5966 -0.1402 0.2439
instrumentalness  -0.1258   0.0761 -1.6526 0.0984 -0.2751 0.0234
liveness          -0.1263   0.0816 -1.5483 0.1215 -0.2862 0.0336
valence            0.1140   0.1063  1.0721 0.2837 -0.0944 0.3223
tempo              0.1354   0.0927  1.4610 0.1440 -0.0462 0.3170
pop                0.0480   0.0897  0.5350 0.5927 -0.1278 0.2238
rock               0.1869   0.1094  1.7092 0.0874 -0.0274 0.4013
hiphop             0.1516   0.1066  1.4221 0.1550 -0.0573 0.3606
dance             -0.1487   0.0848 -1.7544 0.0794 -0.3149 0.0174
folk               0.0871   0.1087  0.8018 0.4227 -0.1258 0.3001
rnb               -0.0547   0.0959 -0.5706 0.5683 -0.2427 0.1332
latin             -0.0804   0.0785 -1.0252 0.3053 -0.2342 0.0733
=================================================================
```

The statistical evaluation confirms the model's reliability and the significance of specific features. The model achieved a Pseudo R-squared value of 0.039, reflecting a modest improvement over the null model. The overall model significance is validated by the LLR p-value of 0.012. At the feature level, duration_ms emerges as a statistically significant predictor (p = 0.005), while other features, such as danceability (p = 0.412) and explicit (p = 0.827), lack sufficient evidence of predictive significance. This analysis underscores the importance of refining feature selection to enhance the model's interpretability and focus on variables that contribute the most to prediction accuracy.

Feature Importance:

The feature importance analysis of the Logistic Regression model provides valuable insights into the predictors of song popularity. Among the most influential features, duration_ms (Coefficient: 0.267, Odds Ratio: 1.306) stands out as the strongest positive predictor, indicating that longer songs are more likely to be popular. Similarly, the rock (Coefficient: 0.185, Odds Ratio: 1.203) and hiphop (Coefficient: 0.150, Odds Ratio: 1.162) genres significantly increase the likelihood of a song being classified as popular. Other positively correlated features include loudness (Coefficient: 0.143, Odds Ratio: 1.154) and tempo (Coefficient: 0.134, Odds Ratio: 1.143). On the contrary, dance (Coefficient: -0.148, Odds Ratio: 0.862) and instrumentalness (Coefficient: -0.126, Odds Ratio: 0.882) exhibit negative relationships, suggesting that these characteristics reduce the likelihood of a song's

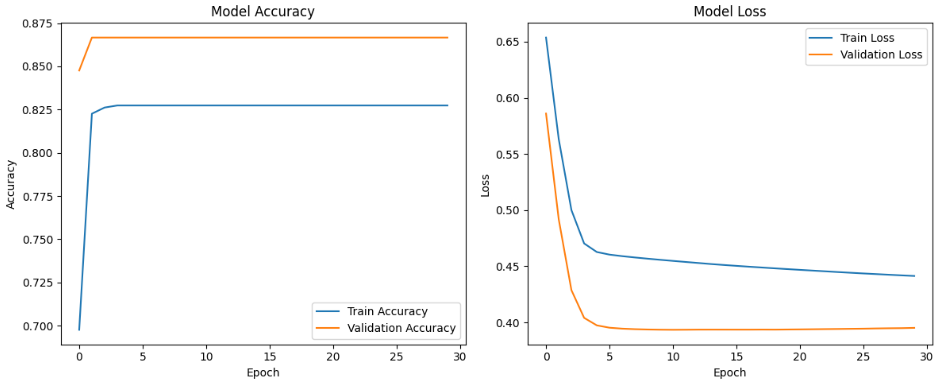
Feature Importance with Logistic Regression:

	Feature	Coefficient	Odds Ratio
1	duration_ms	0.267173	1.306266
14	rock	0.185018	1.203240
15	hiphop	0.149723	1.161512
6	loudness	0.142810	1.153510
12	tempo	0.133927	1.143309
11	valence	0.111433	1.117879
17	folk	0.085973	1.089777
9	instrumentalness	-0.125536	0.882024
16	dance	-0.148448	0.862045

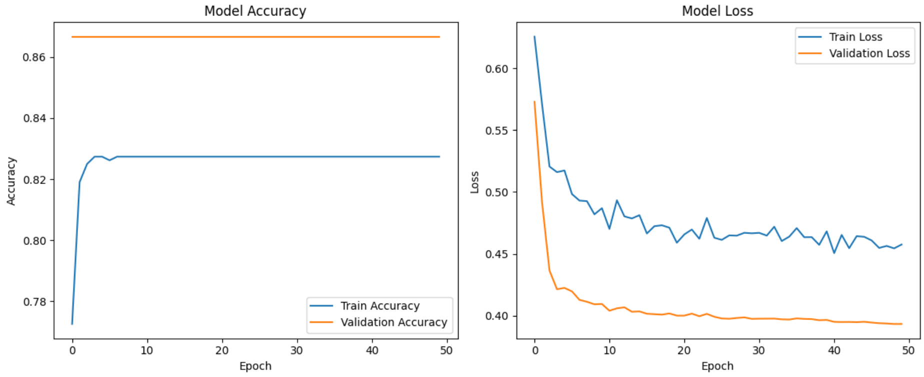
popularity. Additional negative predictors, such as energy and speechiness, have a less pronounced impact. This analysis highlights the most critical drivers of popularity predictions.

C. NEURAL NETWORK MODEL:

Colab link: https://colab.research.google.com/drive/1_vsG8uPQXcRBfIy8sfXoL5d9T6uScA6e?usp=sharing

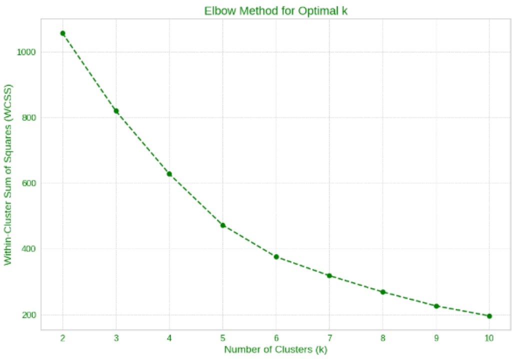


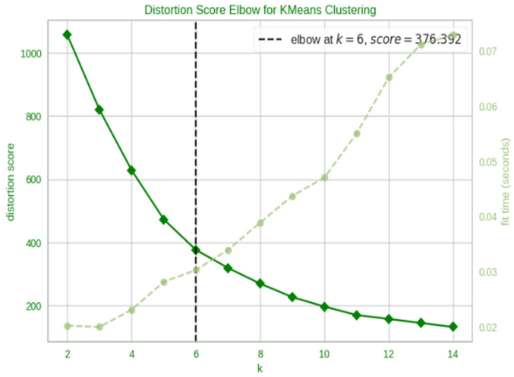
The accuracy and loss plots for the neural network model trained over 50 epochs demonstrate strong generalization. Training accuracy stabilizes at 82.5%, while validation accuracy peaks at 86.6%, with minimal divergence between the two. Training loss reduces to 0.42, while validation loss converges around 0.40, indicating effective learning without overfitting. With training reduced to 30 epochs, the model achieves comparable performance. Training accuracy remains at 82.5%, and validation accuracy is slightly lower at 85.8%. Training and validation losses stabilize at 0.44 and 0.41, respectively, demonstrating efficiency in fewer epochs. The consistent performance across both configurations highlights the model's robustness and ability to generalize well. Training for 30 epochs provides a time-efficient alternative with nearly identical results, making it ideal for scalable implementation in Spotify's predictive framework.

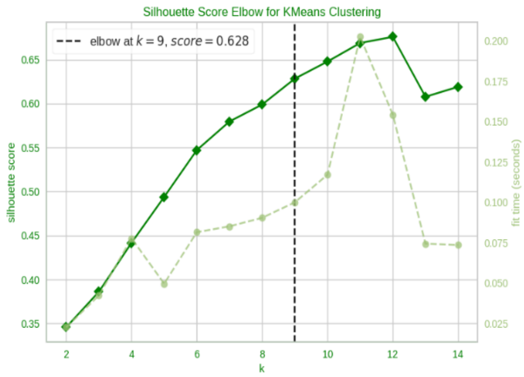


CLUSTER ANALYSIS

Colab Link: https://colab.research.google.com/drive/1coeKlFk6WtPdYSl1Y6dS4IQkaEdhNkNZ?usp=sharing







<u>Elbow Method for Optimal k</u>

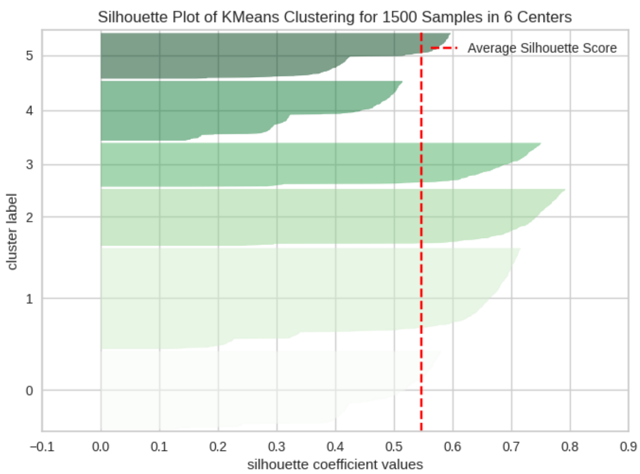
At k=6, a distinct "elbow" is observed, indicating a balance between model complexity and explanation of variance. Beyond k=6, WCSS decreases minimally, suggesting diminishing returns for increasing clusters.

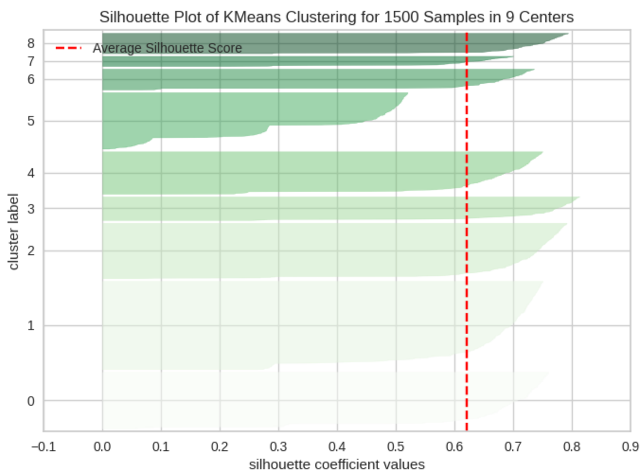
<u>Distortion Score and Silhouette Analysis</u>

Distortion Elbow Plot (k=6): The distortion score decreases significantly up to k=6. This reinforces k=6k as the optimal value where intra-cluster similarity is high.

Silhouette Score Plot (k=9): Although k=9 achieves a slightly higher silhouette score (0.628 compared to ~0.547 for k=6), the increased number of clusters leads to reduced interpretability without a substantial increase in clustering quality.

Silhouette Analysis





- For k=6, clusters are well-separated with most silhouette scores above the average (0.547).

- k=9 introduces smaller, overlapping clusters with less consistency despite a slightly higher average score.

Cluster Profiles

- k=6:
    - Six distinct clusters with clear genre and feature separation.
    - Examples: Cluster 1 (pop, 26.27%), Cluster 0 (hiphop/dance, 20.27%), Cluster 4 (rock, 15.53%).
    - Larger, meaningful clusters simplify interpretation.
- **k=9:**
    - Splits k=6clusters into smaller, fragmented ones (e.g., Clusters 6, 7, 8), reducing coherence and practicality.

Cluster Profile for k=6:

	danceability	energy	pop	rock	hiphop	dance	folk	rnb	latin	Count	Percentage
Cluster 1	0.64	0.73	0.90	0.00	0.00	0.00	0.03	0.00	0.05	394	26.27
Cluster 0	0.72	0.73	0.75	0.03	1.00	0.00	0.00	0.00	0.04	304	20.27
Cluster 4	0.67	0.81	0.83	0.06	0.22	1.00	0.00	0.02	0.00	233	15.53
Cluster 2	0.71	0.67	0.99	0.00	1.00	0.01	0.00	1.00	0.01	222	14.80
Cluster 5	0.52	0.81	0.46	1.00	0.00	0.01	0.03	0.00	0.01	177	11.80
Cluster 3	0.67	0.66	0.94	0.01	0.00	0.00	0.01	1.00	0.00	170	11.33
Overall	0.66	0.74	0.82	0.13	0.39	0.16	0.01	0.26	0.03	1500	100.00

Cluster Profile for k=9:

	danceability	energy	pop	rock	hiphop	dance	folk	rnb	latin	Count	Percentage
Cluster 1	0.64	0.73	1.00	0.00	0.00	0.00	0.03	0.00	0.05	353	23.53
Cluster 4	0.73	0.72	1.00	0.04	1.00	0.00	0.00	0.00	0.06	227	15.13
Cluster 0	0.71	0.67	0.99	0.00	1.00	0.01	0.00	1.00	0.01	222	14.80
Cluster 3	0.66	0.81	1.00	0.07	0.23	1.00	0.00	0.03	0.00	194	12.93
Cluster 5	0.67	0.66	0.94	0.01	0.00	0.00	0.01	1.00	0.00	170	11.33
Cluster 6	0.52	0.84	0.00	1.00	0.00	0.01	0.02	0.00	0.02	95	6.33
Cluster 8	0.53	0.77	1.00	1.00	0.00	0.00	0.04	0.00	0.00	82	5.47
Cluster 2	0.67	0.77	0.00	0.00	0.09	0.49	0.01	0.00	0.04	80	5.33
Cluster 7	0.71	0.75	0.00	0.00	1.00	0.00	0.00	0.00	0.00	77	5.13
Overall	0.66	0.74	0.82	0.13	0.39	0.16	0.01	0.26	0.03	1500	100.00

k=6 balances interpretability, compactness, and actionable insights for segmentation, while k=9 complicates analysis with overly granular clusters.