Music Genre Classification Using Principal Component Analysis Combined with K-Nearest Neighbors and Logistic Regression

MATH UN2015 - Linear Algebra and Probability Fall 2024
Amanda Li, Chris Hyorok Lee, Dominick Vaske

Abstract

Music streaming platforms have long utilized a variety of user-based and content-based recommendation systems in their services. In this paper, we present a novel approach to Music Genre Classification by integrating K-Nearest Neighbors (KNN) and Logistic Regression with Principal Component Analysis (PCA). By reducing the dimensionality of our data from 12 to 6, we were able to capture 91% of the total variance and develop a classification model with accuracies of 89% and 90% using KNN and logistic regression, respectively.

Introduction

In a digital world, the music listening experience is largely influenced by the music recommendation system, which is an intelligent system that employs data mining and machine learning techniques to propose songs or playlists based on a user's listening history preferences, and behavior.[1] As music enthusiasts, we wanted to dive deep into the relationship between content-based approach to music recommendation system and the music that we listened to. By leveraging the public metadata of songs provided by the Spotify Web API, we wish to create a model that accurately predicts whether a certain song fits into one of the 5 distinct genres. A combination of PCA and KNN have been utilized widely for prediction in a variety of applications. Researchers have used this method to predict Blast Furnace Slag Viscosity with an accuracy reaching 99%[2]. Furthermore, one-vs-rest logistic regression have been used for multi-class classification tasks such as music genre classification tasks reaching a promising Area Under Curve (AUC) of 0.894[3]. This project explores the classification of song genres using dimensionality reduction via Principal Component Analysis followed by prediction via Nearest Neighbor and Linear Regression Methods.

We picked 5 distinct musical genres and selected 100 songs for each genre for a total of 500 songs. Then we used Python code to access Spotify API to collect audio features such as Acousticness, Danceability, Energy, Instrumentalness, Key, Liveness, Loudness, Mode, Speechiness, Tempo, Time Signature, and Valence for our 5 genres. We preprocessed data by removing irrelevant features to focus on the most informative aspects of our data. The remaining features were then standardized to ensure a consistent scaling, making them suitable for dimensionality reduction. PCA was applied to reduce the dataset's dimensionality while retaining the majority of the variance. Starting with 10 different parameters (after dropping 3 for irrelevance), we reduced the dimensionality to 6 principal components, which retained 91% of the variance. The reduced dataset served as an input for a KNN classifier. KNN predicts the genre of a test song by identifying its closest k "neighbors" in the principal component space.

Cross-validation was conducted to optimize the value of k to be 7, which yielded the highest classification accuracy, 89%. As an alternative prediction method, Logistic Regression was employed to compare its performance against KNN. Using the logistic regression model with the line of best fit, test songs were classified based on where they fell in relation to this line. Our logistic regression model achieved a prediction accuracy of 90%.

Method

Raw Data Collection

Each member of the group generated Spotify playlists for their given genres, and we collected 100 songs for Heavy Metal, Latino Pop, Neo Soul, Punk Pop, and Study Music. Dataset of 500 songs was generated using Python code by sending GET requests to Spotify Web API to fetch relevant audio features as JSON objects. Afterwards, JSON objects were parsed into a CSV file with Acousticness, Danceability, Energy, Instrumentalness, Key, Liveness, Loudness, Mode, Speechiness, Tempo, Time Signature, Valence, id, Duration, Genre as columns. This CSV file was then loaded into a pandas DataFrame for further data processing.

Principal Component Analysis (PCA)[4]

In our model, we used the first 6 principal components that described 91% of the total variance for k-th nearest neighbor and logistic regression prediction. We first standardize the variables to ensure all variables are on the same scale using equation 1. We then perform singular value decomposition on the standardized data, following equation 2.

$$A_{standardized} = \frac{A - M}{S}$$

Equation 1. Standardization of Raw Data. A is the 500 x 10 raw data matrix, M is a 1 x 10 vector consisting of the mean value of each column in A, S is a 1 X 10 vector consisting of the standard deviation of each column in A

$$a.\ A_{standardized} = U\Sigma V^T$$

$$b.\ u_i = \frac{1}{\sigma_i} A v_i$$

Equation 2. Singular Value Decomposition of Standardized Data. a. U is an orthogonal square matrix constructed using formula in b., where each u_i is a column in U. Σ contains the singular values by finding the eigenvalues A^TA and taking their positive square roots, ordered in descending order along the diagonal entries. V^T transpose of matrix V which is an orthogonal matrix constructed by finding eigenvectors of A^TA and normalizing them to have length 1. b. σ_i is the corresponding singular value in Σ, and v_i is the corresponding eigenvector in V.

Then, we choose the first k principal components that describe > 90% of the total variance in the data by dividing each singular value in Σ by the trace of Σ. Lastly, we truncate V to keep the first

k columns and multiply the standardized data with truncated V to obtain the transformed data using the first k principal components.

K-Nearest Neighbors (KNN)

KNN is a simple, non-parametric classification algorithm. It predicts the class of a data point by considering the K nearest points in a dataset and assigning the corresponding majority class among them.[5]

After applying Principal Component Analysis (PCA), to ensure the data was suitable for subsequent distance calculations, we standardized all features to a scale between 0 and 1. This step is critical, as distance metrics like Euclidean distance are highly sensitive to differences in feature magnitudes, which could otherwise distort proximity measurements.

The choice of KKK, the number of neighbors considered during classification, was another key step. Selecting an appropriate KKK is a balance between avoiding overfitting, which can occur with very small values of KKK (e.g., K<5K < 5K<5), and avoiding underfitting, which happens with excessively large values of KKK. We began our analysis using a moderate K=5K = 5K=5, which is a common default. To ensure robustness, we cross-validated results by analyzing performance across a range of KKK values, ultimately selecting the value that yielded the best classification accuracy.

The KNN algorithm employed the default Euclidean distance to calculate proximity between data points. This metric is defined as:

$$d(p, q) = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}$$

Equation 3. Euclidean Distance Formula.

where ppp and qqq are points in nnn-dimensional space. Euclidean distance was chosen for its simplicity and effectiveness in capturing geometric relationships between points in the PCA-reduced space.

During the classification process, for each test song, KNN computed the distances to all training data points and identified the KKK nearest neighbors. The algorithm then assigned the most common genre among these neighbors as the predicted class for the test sample. To evaluate performance, predictions were compared to true labels using a combination of metrics, including overall accuracy, a confusion matrix, and a detailed classification report (precision, recall, and F1-score). These metrics provided a comprehensive understanding of the classifier's effectiveness and areas for improvement.

Logistic Regression (LR)

Logistic Regression (LR) is a supervised classification system that, given an input, outputs a discrete outcome. This linear classifier is generally used for binary classification tasks, but for this multi-class classification, the LR is implemented using the one-vs-rest method with 5 separate classifiers.[6]

Dataset preparation is identical to KNN, and we divide the dataset into 80% training data and 20% testing data. The model trains on the training data by each of the 5 classifiers predicting whether a sample belongs to a specific class or not, which makes each classification binary. Then among the 5 separate binary classifiers, the class with the highest probability is chosen as the predicted class.

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

Equation 4. Sigmoid Function.

During training, the sigmoid function is applied to the linear combination of the audio features to produce probabilities for each binary classification. Sigmoid function maps these values to the range [0,1], making the values suitable for calculating probabilities. During prediction, the model calculates the probabilities for all 5 classes and assigns the class with the highest probability.

To evaluate the model, the overall accuracy is measured by dividing the number of correct predictions by total predictions and an additional confusion matrix is generated by calculating the true positives, false positives, true negatives, and false negatives of each genre.

Results and Discussion

Data Collection

Radar chart in Figure 1 reveals crucial details about the relationship between audio features of the 5 genres of music. As seen in Figure 1, 5 genres show distinct mean values for each of the 12 audio features. We see that Heavy Metal has the highest Energy while Study Music has the highest Acousticness and Instrumentalness. Latino Pop has the highest valence(measure of positiveness) and Neo Soul indicates the highest danceability. These audio features are effective as training data because of clear differences in mean values. In contrast, the mean values of audio features such as liveness and key are very similar, suggesting that these features are unlikely to effectively distinguish between genres. By analyzing the relationships between different audio features and genres, we selected Acousticness, Danceability, Energy, Instrumentalness, Loudness, Speechiness, Tempo, and Valence for our training and validation.

Figure 1. Mean Value comparison of the audio features in heavy metal, latino pop, neo-soul, punk pop, and study music

PCA

PCA was used to reduce the dimensionality of our data for further analysis. After standardizing our raw data and performing singular value decomposition, we constructed a scree plot in Figure 2 to visualize the amount of variance each principal component captures, with respective percentage recorded in Table 1. A relatively low percent variance in the first principal component signifies that the parameters in our dataset are not closely correlated.

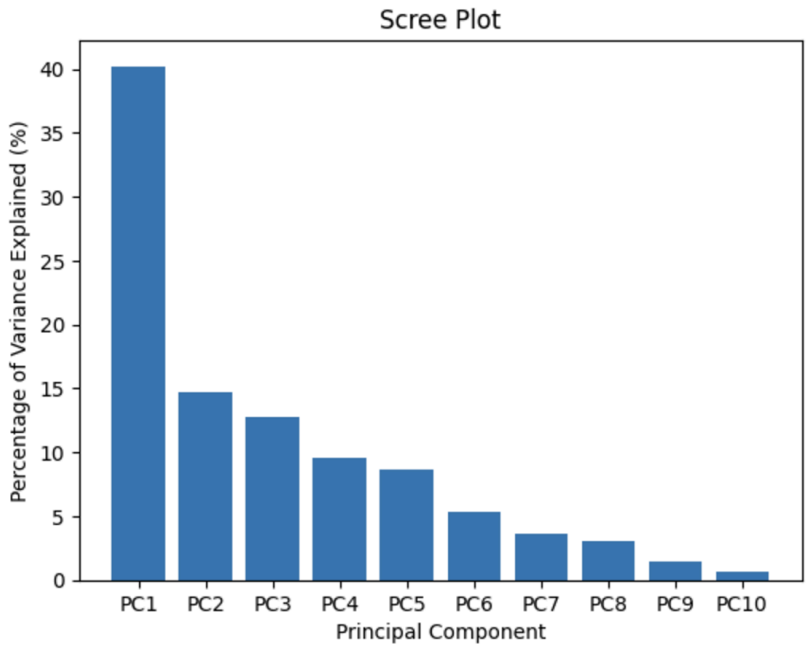

Figure 2. Scree Plot. A visualization of the percent variance each principal component captures.

PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10
40.21	14.71	12.77	9.59	8.63	5.35	3.60	3.00	1.48	0.64

Table 1. Percent of Variance Captured By Each Principal Component (PC).

We chose to use the first 6 principal components for dimensionality reduction, which preserves 91.3% of the variance. We therefore reduced the data's dimensionality from 12 to 6. By reducing the dimensionality of our data, we lower noise in our data and increase computational efficiency.

KNN and Logistic Regression

The results of the accuracy of each model are summarized in Table 3. Both models had a high accuracy while Logistic Regression had a slightly higher overall accuracy of 0.9. From Figure 3 and Figure 4, we can see that Study Music had 24 true positives and 0 false positives for both KNN and Logistic Regression. This is likely due to the fact that Study Music playlist was composed of all instrumental music and was largely a single instrument track compared to other genres which had multiple instruments and voices. Latino Pop also indicated a very strong accuracy in classification for both models, but results for other genres vary from model to model.

	Accuracy
K-Nearest Neighbor	**0.89**
Logistic Regression	**0.9**

Table 3. Overall Accuracy for KNN and Logistic Regression

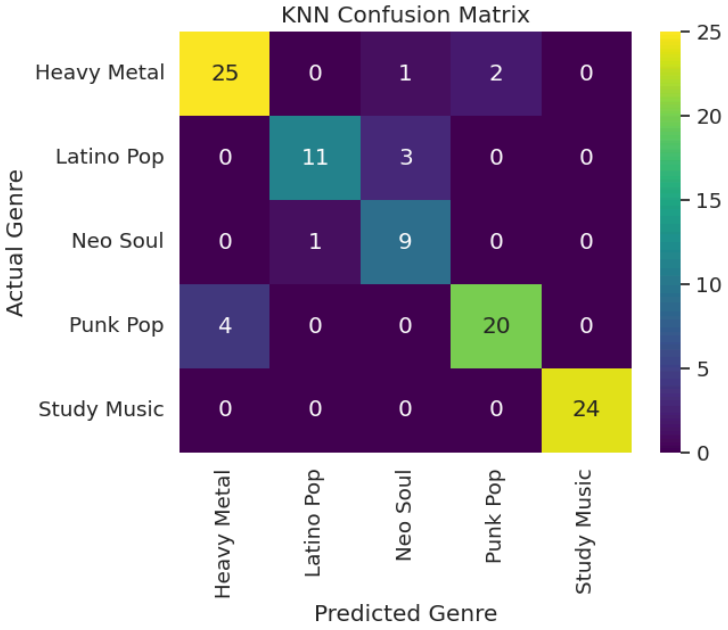

Figure 3. KNN Confusion Matrix

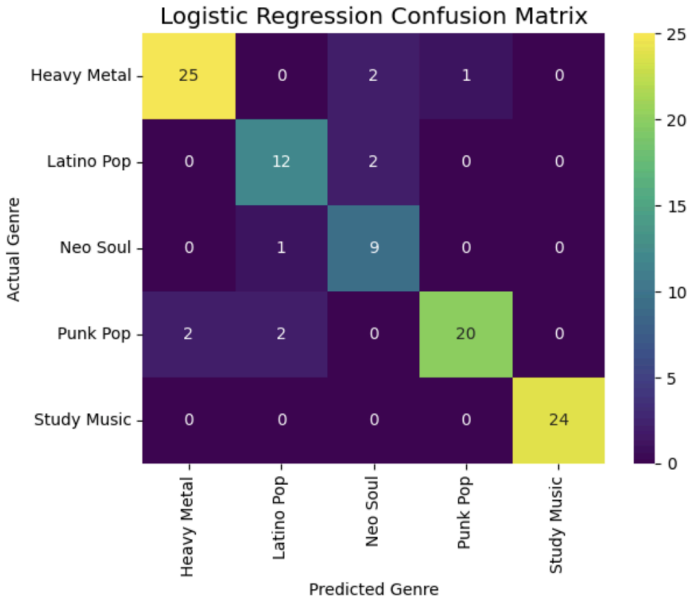

Figure 4. Logistic Regression Confusion Matrix

Heavy Metal and Latino Pop performed moderately well with minor confusion between genres. Both models misclassified a few Heavy Metal songs as Punk Pop, with KNN misclassifying 4 Heavy Metal songs as Punk Pop. This shows that the two genres have similar musical elements and their similar instrumentation, loudness, tempo, and danceability explains this behavior.

Neo Soul was by far the lowest-performing genre, and both models misclassified some Neo Soul songs as Latino Pop or Heavy Metal. This indicates that Heavy Metal and Latino Pop had overlapping characteristics in audio features with Neo Soul. It is also worth noting that genres with smaller training data (Neo Soul and Latino Pop) experienced slightly lower performance, possibly due to reduced representation during training.

As a last experiment, to identify the optimal value of K for the KNN classifier, we trialed multiple values ranging from small to large, analyzing the impact on classification accuracy. Starting with a moderate default value of K=5, we observed how varying K affected the balance between overfitting and underfitting data. Smaller values of K showed higher sensitivity to noise in the dataset, leading to overfitting, while large values tended to oversimplify classifications, resulting in underfitting. Using cross-validation, we tested the classifier across various values of K and determined that K = 7 provided the highest accuracy (92%), striking the best balance between generalization and precision. This process ensured that our final model was both robust and effective for genre classification.

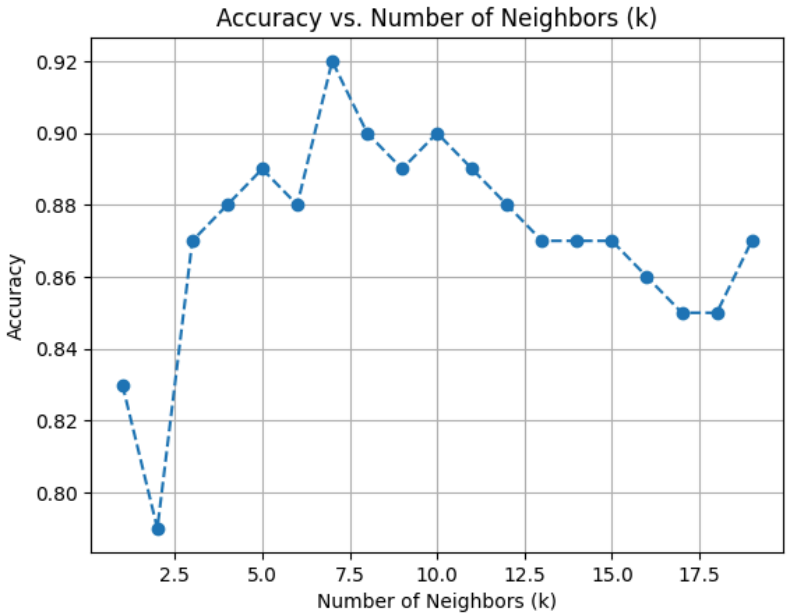

Figure 5. Accuracy of Different K Values in KNN Classifier

Conclusion

In this study, PCA was combined with Kth Nearest Neighbor and Logistic Regression to train and classify 5 genres of music, Heavy Metal, Latino Pop, Neo Soul, Punk Pop, and Study Music. 12 audio parameters were collected using Spotify API, 2 parameters were excluded during data preprocessing, and PCA was performed to reduce the dimensionality of the dataset to 6 dimensional. Using KNN and Logistic Regression, the accuracy of our prediction model was 89% and 90%, respectively. Study music overall yielded the highest prediction accuracy, potentially due to its high distinction in features compared to other genres. In both models, Heavy Metal, Punk Pop, and Neo Soul had some mixed predictions, indicating that there could be potential feature overlaps between these genres. Overall, we have shown that using PCA combined with KNN and logistic regression is an efficient method to construct music classification models.

A potential room for improvement involves using a larger dataset for each genre, which can potentially improve the accuracy scores of the models. Possible future direction could include comparing alternative dimensionality reduction techniques such as Min-Max Scaling with PCA, and using other machine learning algorithms such as Support Vector Machines, Convolution Neural Networks, Recurrent Neural Networks, and Long Short-Term Memory Networks that are commonly used to solve Music Genre Classification problems.[6]

This project has the potential to completely change how we think about music genres and classification. By going beyond basic audio features and adding features like rhythm patterns, how songs change over time, or even information about the artists (e.g. their popularity) and playlists, we could uncover surprising connections between genres. For example, we might find out why Neo Soul and Latino Pop were confused occasionally in our classifier or even discover new genres that no one has officially named yet!

Using the previously aforementioned advanced tools that are already available, we could make predictions more accurately and learn more about how music works. Imagine creating a system that not only classifies songs perfectly, but could also then show us how genres are evolving and influencing each other. We believe most would agree the general style of music present in America's Top 40 today is different from those 10 years ago. A better performing model of ours could analyze these changes and lead to better music recommendations for listeners and new insights into what makes music so universally loved. At the end of the day, this project and all future iterations of, seek to do more than just classify songs–we wish to explore the science behind music in ways that haven't been done before.

References

[1]Mukhopadhyay, S., Kumar, A. et al. (2024). *Enhanced Music Recommendation Systems: A Comparative Study of Content-Based Filtering and K-Means Clustering Approaches.* IIETA RIA, 380138, 365-376. [Link to Article]

[2]Jiang, D., Zhang, J., Wang, Z. et al. A Prediction Model of Blast Furnace Slag Viscosity Based on Principal Component Analysis and K-Nearest Neighbor Regression. JOM 72, 3908–3916 (2020). [Link to Article]

[3]Bahuleyan, H., Music genre classification using machine learning techniques. arXiv preprint arXiv:1804.01149 (2018). [Link to Article]

[4]Greenacre, M., Groenen, P.J.F., Hastie, T. et al. Principal component analysis. Nat Rev Methods Primers 2, 100 (2022). [Link to Article]

[5]Cover, T. M., & Hart, P. E. (1967). Nearest Neighbor Pattern Classification. *IEEE Transactions on Information Theory*, *13*(1), 21-27. [Link to Article]

[6]Quinto, R. J., Atienza, R. O. et al. (2017). *Jazz music sub-genre classification using deep learning.* TENCON 2017 Region 10 Conference, IEEE, 3111-3116. [Link to Article]

Appendix

Google Colab Link:
https://colab.research.google.com/drive/1f9WhFJ5qMf5qKZJdPMck89NJFFQY3nt4?usp=sharing