Georgia State University

CSC4850 & DSCI4850 – Machine Learning

[Spring Semester 2024]

Final Project Report

[Predicting Song Popularity Trends Using Advanced Machine Learning Models]

[Group 8]
[Brina Chahal]
[Naveen Thomas]
[Ify Ojiaku]

Table of Contents

1. Introduction

1.1 Abstract

This project aims to predict song popularity trends by analyzing diverse song features over the past two decades. Leveraging multiple datasets encompassing information on music genres, tempo, featured artists, volume, release dates, energy levels, and danceability, advanced machine learning models will uncover changing trends in pop culture. The objective is to identify key factors driving song popularity and create a predictive model capable of forecasting chart-toppers. The project addresses the challenge of understanding what resonates with audiences in the dynamic music landscape. Three primary machine learning models—Support Vector Machines (SVMs), Multi-Layer Neural Networks, and Random Forest—will be implemented and optimized to predict song popularity. Evaluation metrics such as accuracy, precision, recall, and ROC-AUC will assess model performance. Ultimately, this research seeks to empower stakeholders, including music producers, artists, marketers, and streaming platforms, with insights into creating music that captivates contemporary audiences.

1.2 Problem Statement

Creating music that resonates with audiences is challenging, influenced by factors like beats per minute, genre, and energy levels. Understanding the correlation between these traits and song popularity is essential. This project aims to develop a machine learning application analyzing diverse song features to discern their appeal over two decades. By leveraging datasets on music genres, tempo, artists, and more, the goal is to identify trends and key factors driving song popularity, enabling stakeholders to predict future chart-toppers and optimize content creation strategies for the dynamic music industry landscape.

1.3 Contributions

Insights into Music Trends: This project offers insights into evolving audience preferences and trends in music popularity over the past two decades, aiding stakeholders in understanding the factors driving song appeal.

Predictive Modeling: By developing predictive models, the project enables stakeholders to forecast future chart-toppers, empowering them to make proactive decisions in content creation and promotion.

Feature Importance Analysis: Through analyzing the importance of different song features, the project helps prioritize creative efforts, optimizing content production strategies for maximum impact.

Informed Decision-Making: The project's insights support informed decision-making

processes for producers, artists, marketers, and streaming platforms, helping them adapt to changing audience preferences and trends effectively.
Methodological Advancements: The project contributes to methodological advancements in machine learning model development and evaluation, offering practical guidance on applying advanced techniques to music popularity prediction tasks.

2. Technical Description

2.1 Dataset Description

The models incorporate 7 different features to help predict the label of each song piece. Even though there are more features on some of the datasets we use for the experiment, some datasets don't have the records of features like others do. These seven features were common all across each dataset. A brief description of each feature is shown below.

BPM: Measure of tempo in music where higher b.p.m means a faster beat. (also known as beats per minute)
Number of Artists: The number of vocal contributors to the piece
Decibels: the measure of volume that the piece may be assumed to be played at
Energy: Measure of the intensity which can be affected by various factors like volume, speed, attitude, etc.
Valence: Measure of mood in a piece, where a higher metric means a more positive mood
Duration: Total time of how long the piece lasts from beginning to end
Danceability: Measures how suitable the piece may be for dancing or other physical-related activities

The target label is the popularity of the piece, which is scaled from 0-100.

2.2 Decision Tree Description

A decision tree is a classification model that can be used to make predictions using a dataset. The parameters that can be changed in the decision tree are max depth, sample split, and sample leaf. The dataset we originally had was not made for a classification model so we used a threshold range to create classifications for the model. We set the threshold ranges as 85 or higher as high(1), between 85 and 65 as medium (2), and below 65 as low(3). We used the Gini index as the base to split the tree. We then implemented a grid search model to hyperparameter optimize the model so the best model could be used to make predictions. Grid Search uses a list of parameters to substitute and test to find the ideal model. It then runs the model and calculates MSE, R2, and accuracy as the metrics.

2.3 Neural Network Description

The neural network is a prediction model that can be used for regression or classification. The parameters that can be changed in the neural network are the activation function, epoch, and hidden layers. We created an ordinal classification neural network, preprocessed the data, and created classifications for the target value.

We set the threshold ranges as 85 or higher as high(1), between 85 and 65 as medium (2), and below 65 as low(3). We used an ordinal cross-entropy activation function that works well with ordinal classification. Ordinal classification is when the target variable is a multiclass classification where there is an inherent ordering of the target value such as high, medium, and low. We then used loss and accuracy as the metrics for the model

2.4 SVM Description

Another model we used to predict song popularity is the support vector machine model. Even though its main use is recognized mostly in binary classification, it can also handle a regression-type problem similar to the prompt. Three different kernels, (radial basis function, polynomial, and sigmoid), were used to derive which type of SVM was the best while also adjusting the parameters ɣ (gamma) and C.

The SVM model consists of the margin line of decision, as well as its support vectors. These can be altered by the parameters ɣ and C to obtain different results. As part of the experiment, the values were manipulated mainly to adjust how well the model would fit the training data, and how good it was at generalization when it came to test data. Increasing ɣ would make the support vector space increase. Increasing C would make the decision boundary have more definition to its shape and fit the training data more accurately. Each Kernel is tested with different combinations of these parameters as each has a blend that is best suited for it.

To capture the effectiveness of each model, 3 regressive metrics are used for each of the kernels. Mean Absolute Error (MAE) is used to capture the average number that the predictive model may be off by. This serves to capture the direct error margin. The next metric used is Mean Squared Log Error (MSE), which is a value within the interval of [0, 1]. Not only does MSE differ from MAE regarding value, but also helps with identifying how significant outliers may affect the test data, which can’t be captured well with MAE. The last metric, the R2 score, is used to explain how well the model fits the data, which also takes the parameters into account. Its range is also [0, 1], where a higher R2 means that there is higher relevance.

2.5 Linear Regression Description

The linear regression model is a prediction model specifically for regression datasets. We will use the ridge model. The parameters that can be changed for the model are the regularization strength and the solver. We will use a grid search to find the ideal parameters for the model and use R2 and MSE as a metric to evaluate the model.

3. Evaluation

3.1 Decision Tree Results

Decision Tree results:

Max_depth, samples Split, Samples leaf	Accuracy	MSE	R2

1,2,1	.662	.338	.169
2,2,1	.666	.334	.157
3,2,1	.666	.349	.209
1,2,2	.662	.338	.169
1,3,1	.662	.338	.169
1,3,2	.662	.338	.169
1,2,3	.662	.338	.169
2,3,1	.666	.334	.157
3,3,1	.666	.349	.209
3,3,3	.666	.349	.209

The decision tree results show that the ideal depth is 1 and that the ideal depth is 1, samples split is 2, and sample leaf is 2. We see that a higher max depth can decrease the accuracy and increase the error so 1 was ideal. It was the same for the sample split and sample leaf. The overall low accuracy of the model can be seen with the fact that more datasets might be necessary to find a better result.

3.2 Neural Network Results

NN results:

Hidden layers	accuracy	Validation accuracy	loss
1	.042	.035	13.02
2	.157	.189	3.04
3	.075	.0786	9.98
Epoch			
10	.157	.189	3.037
15	.042	.0468	11.9
20	.0055	.0033	19.752 3

The neural network results show that the ideal hidden layers are 2 and the ideal epoch is 10. Changing the epoch can cause the loss to increase and the accuracy to decrease. Changing the hidden layers also causes the accuracy to decrease and loss to increase. This is mainly because of the low number data available in the data set and if more data is added then the accuracy should increase.

3.3 SVM Results

Because many combinations can be tested with the parameters, the default values of $\gamma = 0.1$ and C = 1 are first recorded for reference, which can be shown below.

Test Data Results	RBF	Polynomial	Sigmoid
Mean Absolute Error	7.67768	9.50413	6.76859
Mean Squared Log Error	0.08469	0.16562	0.25545
R2 Score	0.47286	0.37895	0.61717

Figure: Each column in the table is a kernel with each of the recorded values of the metrics

For the default parameters, each kernel has its strengths and weaknesses when looking at each of the metrics. RBF yields the lowest MSE, however, it may not generalize well to new data, which can be reflected by its R2 score. Polynomial yields the lowed R2 score showing a lack of correlation between the model and data, however, note that the polynomial will become the kernel that's most affected by the change in parameters. Sigmoid yields the highest MSE, which may be due to outlier data points affecting the error margin, however, this may not entirely be a drawback as we may not want the data to overfit. This is further proven by its R2 score, which is higher than the other two kernels. After recording the results, the values of gamma and C are adjusted for each kernel to see which combination works best for which.

Test Data Results	RBF $\gamma=0.1$ C = 2	Polynomial $\gamma=0.4$ C = 4	Sigmoid $\gamma=0.2$ C = 3
Mean Absolute Error	5.51239	5.03305	6.24793
Mean Squared Log Error	0.06191	0.13366	0.18816
R2 Score	0.68174	0.79498	0.63949

Figure: Final results showing the proposed combination of gamma and C that works best for each kernel

Starting with the RBF kernel, C was increased to various values to observe its effects. When increasing it past the default value, RBF only worsened during the test results, so it was interpreted that 0.1 would be the best value for γ. When changing C, the metrics were observed at their best when set to 2. When manipulating the polynomial kernel, the metrics were at their best when γ and C were set to 0.4 and 4 respectively. Finally, for the sigmoid kernel, the best parameters found were γ = 0.2 and C = 3 after experimenting with various values.

Compared to iteration with default parameters, the final results yield a much different value for each metric as we're able to improve the model's predictability overall. Even though RBF only had a very slight reduction in overall error, the R2 witnessed a good increase, showing that the model has a decent relevance to the data. Sigmoid overall only receives a slight increase in its prediction ability as MSE is decreased by a fine amount, allowing it to predict outliers more easily. Out of the three, the polynomial kernel went through the most significant change, as its R2 score is very sizable, showing the best correlation with the data. Even though it yields a higher MSE than the RBF kernel, it may not matter as much since it only reflects the ability to capture outliers.

3.4 Linear Regression Results

Regularization Strength	R2	MSE
.1	.0523	121.94
1.0	.0523	.121.94
10.0	.0523	121.94
Solver		
‘auto’	.0523	121.94
‘svd’	.0523	121.94
‘cholesky;	.0523	121.94
‘lsqr’	.0523	121.94
sparse_cg	.0523	121.94
‘sag’	.0523	121.94
‘saga	.0523	121.94

The parameters entered did not change the metrics of the model most likely because there was not enough data for the parameters to make a large enough difference with the metrics.

4. Conclusions

Our project aims to analyze patterns in historical music data, uncovering the key factors that drive a song's popularity.

Our decision tree model, with its accuracy rate of 66%, proves to be an okay tool for predicting performance during testing, showing reliability to a certain extent in capturing underlying patterns in the data. Additional evaluation metrics and domain knowledge should be considered. While the decision tree may not be considered as a model in this report, it can handle various song features, both numerical and categorical. They can also capture non-linear relationships crucial in music analysis.

However, our neural network model currently struggles with a lower accuracy rate, hinting at potential areas for improvement such as fine-tuning model parameters or augmenting the dataset with additional relevant features. Despite these challenges, the neural network model remains a valuable asset in our toolkit, offering the potential for enhanced predictive capabilities with further refinement.

On the other hand, our SVM model, particularly when using the polynomial kernel, showcases promising results with a respectable R2 score showing that the model has strong relevance to the data, while also yielding a respectable mean squared log error. The utilization of the polynomial kernel highlights the importance of non-linear relationships between song features and popularity, suggesting the presence of intricate patterns that traditional linear models may overlook. Additionally, addressing multidimensionality and the nuanced interpretation of feelings between machines and humans could further enhance the effectiveness of our models in capturing the intricacies of music popularity.

By delving deeper into the analysis of model performance and exploring avenues for refinement, we aim to develop robust machine-learning solutions that provide valuable insights for content creators in the music industry.

5. Teamwork

Brina created the slides and abstract for CS project demo day and contributed to sections 1, 4, and 5 of the final report. Naveen developed the neural network and decision tree classifier, and final report conclusion, while Ify implemented the support vector machine and worked on the final report revisions. All three members actively contributed to the project proposal, status updates, and final report, reflecting a collaborative team effort.

6. Code

https://drive.google.com/drive/folders/1w4W9qljFHVI4RBawzDceAqRFHmt2gd48?usp=drive_link

ML Code The code is included in the link

https://github.com/ifojiaku/SVM-Project-Code