

CSE343 : Machine Learning Project Proposal

OK Computer : Employing Machine Learning to analyse the impact of different parameters in foretelling the next earworm

Ansh Arora
2019022

Jishnu Raj Parashar
2019048

Nandika Jain
2019064

Tushar Mohan
2019393

Abstract

Music, like other forms of art, is something that has historically been subjective to the listener's ears. However, in the age of TikTok and Reels, it is claimed that the songs regularly reaching the top of the charts are getting repetitive, almost following a pattern that guarantees its success. Some artists have even gone on to make rather braggadocious claims that they know well before the release of their song whether it's going to be a hit or not. Are we, as consumers of these tracks, so naive? Is there in fact a way to find out whether a song would be hit or not purely based on its characteristics?

In this project, we wish to answer these burning questions - What is it, after all, that seems to make a song more popular than others, and can we accurately predict the popularity of a track. Modern machine learning techniques might help us identify the features in a song that make it a hit, as well as accurately tell if a song, based solely on the selected characteristics, would be popular or not.

1 Introduction

Music being one of the most subjective forms of art is theoretically not supposed to be something that should be predictable to hit levels of popularity. However, models given to us in the Machine Learning paradigm enable us to in fact actually find if this is true or not. While artists claim that they can tell if a song is going to be popular or not, we aim to find out if this is true and if yes, what exactly are the properties of a song that makes it more popular as compared to others. We aim to use techniques such as data visualization, feature extraction and selection, regression, classification and clustering to aid our study.

Our main goal is to find which features affect the

popularity of the song the most and create a model that is able to as accurately as possible predict how popular a song can get based on these characteristics. Our study can eventually find its uses in helping budding artists to produce music that is more likely to get popular, as well as help people shrug off their mainstream bug.

2 Literature Survey

1. Araujo et al in *Predicting Music Popularity on Streaming Platforms* make use of SVM, Random Forest and Gaussian Naïve-Bayes models to classify whether a song is successful or not based on its appearance on the Spotify Top 50 Global charts ^[1]. They frame the problem as a classification one, and seem to get mixed results in their study.
2. Ni et al in *Hit Song Science Once Again a Science* make use of features such as duration, loudness and tempo to binarily classify a song as hit or not hit, employing a shifting perceptron ^[2]. They make use of data from the EchoNest project and present a simplistic view on the topic, also giving information on how the patterns have varied over the different decades.
3. Raza et al in *Predicting a hit song with ML* make use of Logistic Regression, Decision Trees, Naïve Bayes and Random Forest methods to find out if there is a secret formula to predict which song will be on the top of charts on release ^[3]. This study also made use of lyrics of the songs in its dataset, performing sentiment analysis over them and distinguishing between hit and not-hit based on the data achieved.

3 Dataset

We were after the latest possible data available for our tests. Thus we set out to create our own dataset and made use of Spotify Charts data from 2017-21, which

{ID, Rank, Track, Artist, Streams, Week, Album_Name, Explicit, Track_Number_on_Album, Artist_Followers, Artist_Genres, Song_Length, Acousticness, Danceability, Energy, Instrumentalness, Liveness, Loudness, Speechiness, Tempo, Mode, Key, Valence.}

4 Data Visualization, Preprocessing

A correlation matrix is a table showing correlation coefficients between variables. Each cell in the table shows the correlation between two variables. Typically the same variables shown in the rows and columns. We use this tool to observe if there exist any obvious patterns in our data that we can observe and utilize. We used Pandas' correlation function and then mapped it to figure using the Seaborn library.



A bar chart titled "Count of Song Keys" showing the frequency of different musical keys. The x-axis is labeled "Key" and the y-axis is labeled "count". The bars are ordered by count in descending order.

Key	Count
1	640
0	460
7	380
11	370
8	360
6	350
5	340
9	330
2	320
10	320
4	270
3	100

A scatter plot showing the relationship between the rank of an artist (Y-axis, 0 to 200) and the number of followers (X-axis, 0 to 8e7). The plot shows a dense cluster of points at low follower counts and low ranks, with a few outliers at high follower counts and high ranks.

[illegible]

2

4.4 Principal Component Analysis (PCA)

PCA is a technique for reducing the number of dimensions in a dataset whilst retaining most information. PCA involves standardization, computation of covariance matrix and eigenvectors to identify principal components. Given the fact that we have a comparatively high-dimensional dataset, it helps us in visualizing the data in a meaningful manner. We used Scikit-Learn's implementation to reproduce the same.

4.5 Data Standardization

It is a scaling technique used to standardize the data and make it more workable with. The values are centred around the mean over a unit standard deviation. Formula used -

$$z_i = \frac{x_i - x_{mean}}{\sigma}$$

5 Methodologies

To achieve our aim of finding the features that most affect the popularity, we are trying multiple varieties of models. As of now, we have used Classification Algorithms - Logistic Regression and Support Vector Machines and have made use of Regression Algorithms - Logistic Regression, Regularized Regression and Support Vector Regression. We have made use of Holdout as well as K-Fold Cross Validation to get the most accurate scores to the data possible.

5.1 Classification

We created thresholds on what can be identified as a hit song vs a non-hit. We tried multiple threshold values ranging from 10 to 100, and got a true representation around a value of 50 for the models we tried out.

5.1.1 Logistic Regression

Logistic Regression model makes use of sigmoid function to squeeze the calculated values from Linear Regression into a range of 0 to 1. Since we are doing a binary classification hence, we set a threshold of 0.5, which implies that any value above 0.5 is classified as 1 and the rest as 0.

5.1.2 SVC

Support Vector Machine is a statistical algorithm which clusters the data points in order to learn a hyperplane

which is efficiently able to divide the data points into 2 halves for binary classification.

5.2 Regression

To process the Rank labels, we created a Hit Score value measure. This assigns a value of 1 to songs that have hit the peak of the chart and a minimum value of 0.3 to any song that has entered the chart. Songs that have not charted are assigned a value of 0, certifying a non-hit.

5.2.1 Linear Regression

We applied Baseline Linear Regression on 5-fold and 10-fold division of the training set and calculated the average RMSE observed on the predicted Hit Score and calculated Hit Score.

5.2.2 Regularized Regression

We used GridSearchCV to find the best possible lambdas on each of the 10 k-folds in both L1 (Lasso) and L2 (Ridge) regressions, and then calculated the RMSE values of the predicted Hit Score with our test Hit Scores.

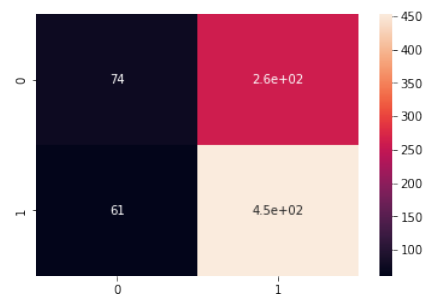
5.2.3 SVR

We made use of Support Vector Regression on a 5-fold division, fitting hyperplane decision boundaries through the training data using SKLearn's implementation to get Hit Score predictions, and then calculated the RMSEs observed.

6 Results and Analysis

From the various models that we applied on our curated dataset, we can say that the features in our dataset have negligible correlation with each other hence our assumption of treating them as independent and identically distributed stands strong.

6.1 Classification



For binary classification, we grouped all the songs into either a hit or not hit by setting a threshold value on the chart rank. SVC performed better in binary classification with an accuracy of 71.85% with a polynomial kernel and 69.29% with a linear kernel.

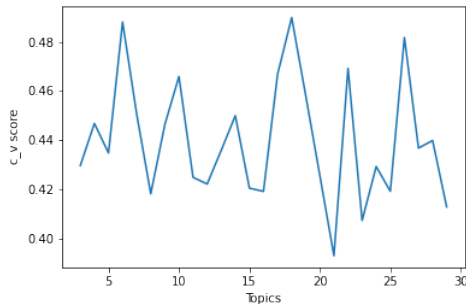
Logistic regression performed fairly with an average accuracy of 67.644% with the SGD classifier peaking at 71% with precision of 84%.

6.2 Regression

Linear Regression performed surprisingly well with an average RMSE of 0.039 in K-Fold validation. Support Vector Regression didn't perform as well as we hoped it would with an average RMSE of 0.044 on a 5 fold K-Fold Cross Validation. However, SVR showed a lower error on the training set, which signifies a High Variance model.

On performing the t-test and f-test on Linear Regression model with 20% test set, we get a p-value of 0.66 which is greater than 0.05 hence, we can say that the mean of residuals of the holdout set is statistically similar to 0.

6.3 LDA



We tested LDA for topics 3-30 and for each run we calculated the coherence scores, i.e. c_v scores, which gives us a sense of how well do the topics fit the given set of documents. The highest score subsequently obtained was 0.48, corresponding to 18 topics.

7 Conclusion

From the results and analysis, we did find a correlation between the features that we extracted and the songs' corresponding popularity, though with a slightly less accuracy from what we expected. With different models and sets of features, we might be able to come up with the desired results.

7.1 Learnings

Having experimented with multiple forms of data and features, at this early stage we can definitely say that there does seem to exist some correlation between certain features and the song's likeliness to be a Hit or have a higher Hit Score. The metric scores obtained on certain tests, especially the ones on the gradient paradigm seem to show that this in fact might actually be a better measure for popularity as opposed to the classification models created in the works done before.

7.2 Work Left

The project has been well on schedule so far. Going ahead, we aim to try more models as per the schedule and experiment with more selection of features, incorporating the ones extracted from lyrics as well since early tests on limited lyrics Datasets have returned promising results. We also aim to expand our Dataset a bit to further strengthen our claims by using the Million Song Dataset (MSD), which presents the same acoustic features as the ones we have been using so far. We will then finally account for issues of Bias and Variance and try to combat them as well using Regularization and Hyperparameter tuning.

7.3 Contributions

We have all aided each other in different parts of the project, and this has been mainly a collective effort.

1. **Ansh** - Lyrics Curation, Exploratory Data Analysis, Support Vector Regression, Regularized Regression, Report + Powerpoint
2. **Jishnu** - Data Curation, Linear/L1/L2 Regression, Logistic Regression, Report + Powerpoint
3. **Nandika** - Data Curation, Linear/L1/L2 Regression, Logistic Regression, Report + Powerpoint
4. **Tushar** - Lyrics Curation, Exploratory Data Analysis, Latent Dirichlet Allocation (LDA), Report + Powerpoint

References

- [1] Araujo et al *Predicting Music Popularity on Streaming Platforms*. [2] Ni et al *Hit Song Science Once Again a Science*. [3] Raza et al *Predicting a hit song with ML*. [4] *Unlocking the True Power of Support Vector Regression*. [5] *Canonical correlation analysis*.