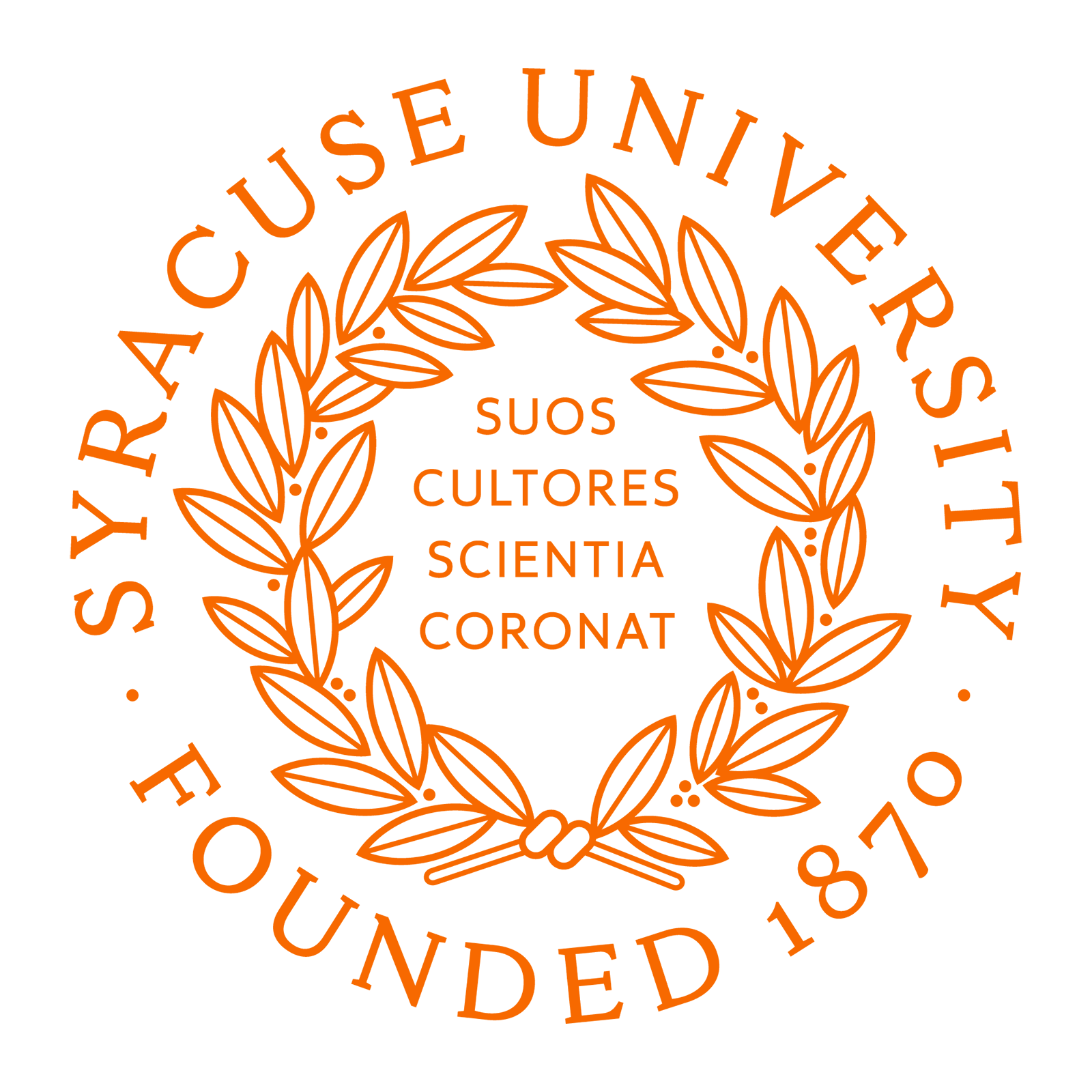
**IST 718: Big Data Analytics**

****

**Project Report: Spotify Data Analysis**

**By: Group 16**

**Aditya Vinod Kini**

**Jaishree Palaniswamy**

**Rahul Dilip Wable**

**Sanjana Sanjay Bhot**

**Table of Contents**

[**Abstract**](#_rwzkan8dwzcf) **3**

[1.1. Project overview](#_t3qcz8pnwi9g) 3

[1.2. List of predictions](#_5yzr05zfe7dg) 3

[1.3. List of inferences](#_3j3oh0phdrc2) 3

[1.4. Conclusion summary](#_eu6ugclk59jh) 4

[1.5. List of other goals](#_wdh8g4mwtgro) 4

[**Data Collection/ Cleaning / Exploration**](#_vwzt3m96c2v) **4**

[2.1 Data Description](#_5sx4j0yo59et) 5

[2.2 Data Cleaning](#_pmtoeewx5khi) 5

[2.3 Data Exploration](#_zctl8h8du2ly) 5

[**Methodology**](#_kx59hik75na0) **7**

[**Models**](#_opjwyywt4eg3) **8**

[4.1 Linear Regression](#_6shljz567oid) 8

[4.2 Random Forest Regression](#_wvpm53z28q20) 9

[4.3 Principal Component Analysis (PCA)](#_7nw7i9sxd7rv) 11

[4.4 K-means Clustering](#_r3cpe8ux8sby) 12

[**Conclusion**](#_vwzt3m96c2v) **14**

# **Abstract**

## **1.1. Project overview**

On many music streaming sites, music recommendations are not representative of the kind of music that their users actually enjoy. Rather many music streaming services mainly use relational algorithms, which essentially recommend songs from similar user playlists. This method does not personalize the user's feedback, so we chose to try an alternate approach that analyses the features of the song. Our goal is to design a song recommendation system that would suggest songs based on the characteristics of the music that users enjoy.

## **1.2. List of predictions**

* The main focus of our project is to provide recommendations of songs to a user. Our system will help the users to improve their listening experience which is unique and more personalized to them. It will also help them explore similar artists and genres from a wide range of Spotify playlists of songs.
* Understand the evolution of music over time and the characteristics of various genres of music popular among the users.
* To create an artist-based recommendation system that recommends users with similar artists.

## **1.3. List of inferences**

* In our project, we used the linear regression and random forest regression model to identify the relationship between various attributes of music for determining the popularity of songs.
* We used the Principal component analysis on the user-generated data for the purpose of songs and artist recommendations tailored to the user’s music taste.
* K means clustering algorithm was used for finding the correlated songs and artists.

## **1.4. Conclusion summary**

* After analysing the characteristics of the songs, we concluded that some attributes positively correlate to the popularity while some negatively affects the popularity. Factors like danceability, liveliness are some key performing attributes that contribute to a song’s success.
* Trend in attributes of songs changes with change in the year. We used time series analysis to see how each attribute evolved over time.
* Recommending songs based on similar artists is not an effective way of recommending songs to the user, each user has a unique taste in music and recommending songs based on the attributes that the user enjoys gives a more personalized user experience.

# **Data Collection/ Cleaning / Exploration**

## **2.1 Data Description**

In our project, we have explored and analyzed the Spotify dataset available on Kaggle. The dataset consists of around 160,000 songs released between 1921-2020, including the details about the artist, the year it was published, duration, and many such features. The main dataset (data.csv) consists of 19 columns including the target variable and 169910 rows. Few available data are grouped by artist, year, or genre as separate CSV files.

* Data by the artist – 15 columns and 27622 rows
* Data by genres – 14 columns and 2665 rows
* Data by year – 14 columns and 101 rows

Dataset Link: https://www.kaggle.com/yamaerenay/spotify-dataset-19212020-160k-tracks

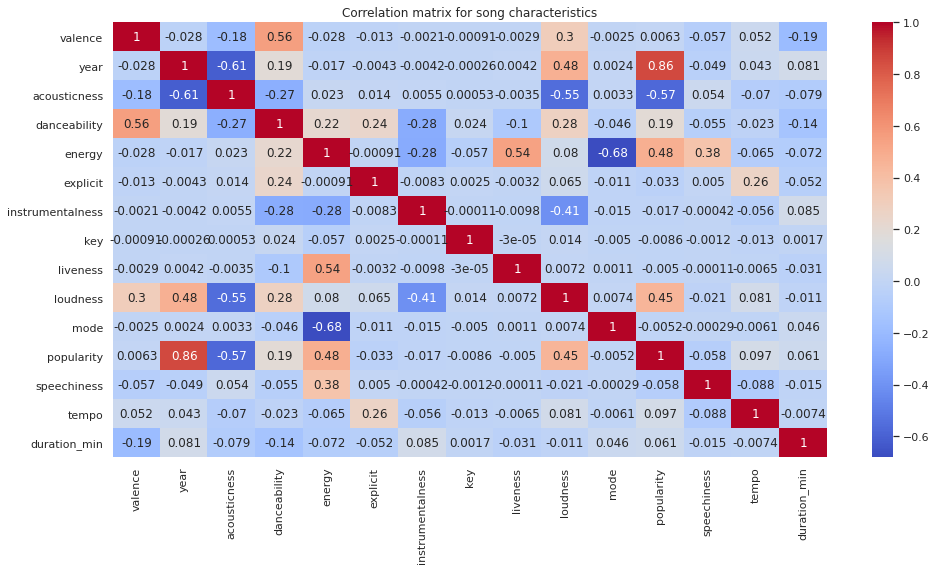
## **2.2 Data Cleaning**

Our Spotify dataset consisted of 169910 rows and 19 columns. There are various attributes in the dataset which would help us classify the songs based on customer preferences. Some major attributes contributing to the classification are author of the song, year when the song was published, danceability, valence, energy, tempo, loudness, instrumentals, and liveness. As part of data pre-processing, null values from the dataset are identified and removed. Few unimportant columns like ID, key, release date which does not contribute much to our analysis were dropped. We also checked for duplicates in the dataset and removed them. The duration of songs was measured in milliseconds, but in order to have better understanding the duration of songs have been converted to minutes. On analysing the datatypes of the attributes, all the attributes are found to be in string, so it’s been converted to float which could be better for passing on to the model.

As part of feature engineering, we have used Vector Assembler in order to combine all the features into a single vector for training models. Also, in order to transform all the variables on the same scale we have used Standard Scaler. It transforms the data and makes it appropriate to apply on a model.

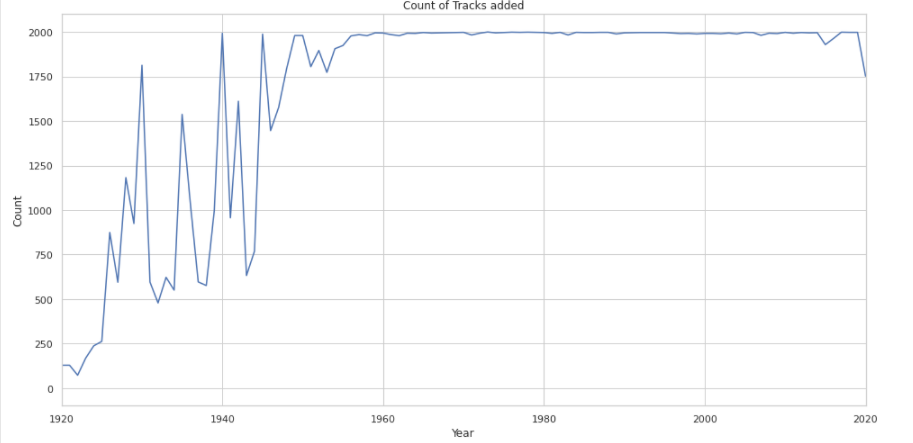
## **2.3 Data Exploration**

The dataset contains a variety of different metrics for songs. Some names offer an indication of what they mean, such as speed, loudness, energy. There are some unique characteristics that are difficult to grasp if you are not a musician. For example, acoustics, liveliness, and speech are technical words that we do not always hear. Some of these characteristics can be associated. The following correlation matrix shows how each feature of a song, negatively or positively affects the other feature. We can see that popularity of the song and the year in which the song was released are positively correlated as new songs tend to be more popular among the users.

****

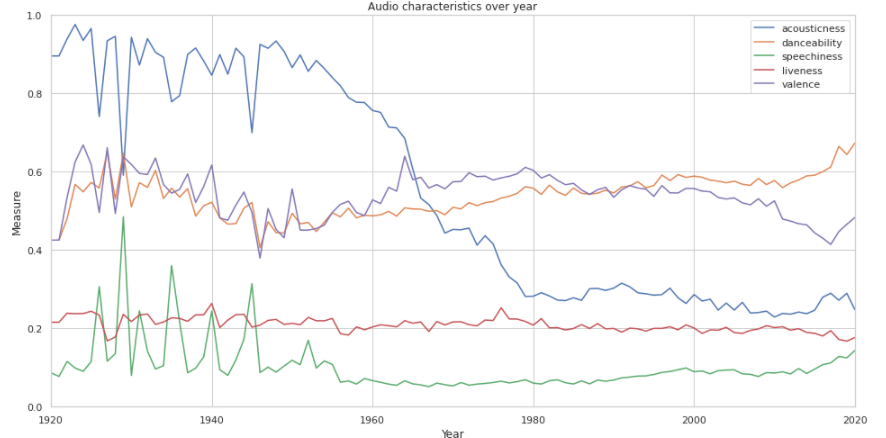
**(Fig 2.1 Correlation Matrix for song characteristics)**

The Time Series graph between the average number of songs released each year, helped us understand the trend in the music industry over several decades. From the year 1920 to 1950 there was a huge uncertainty on the number of songs released every year. This time frame is when the music industry emerged, when records replaced sheet music as the most important product in the music business. After 1950 we can see a consistency in the number of songs released every year.

****

**(Fig 2.2 Time Series graph for Average number of songs released each year)**

A song has various characteristics that may be responsible for its popularity. It is difficult to make a conclusion if the song will be popular or not just based on these attributes. After plotting the line graph for some of the characteristics, we can see that these characteristics follow a trend that is highly influenced by when the song was released (Year). People tend to like certain types of music with similar characteristics for certain periods of time. These characteristics may vary with changes in people’s preferences over the years.

****

**(Fig 2.3 Time Series graph for Audio characteristics over year)**

# **Methodology**

For implementing our project, we followed the CRISP-DM methodology along with the Data Science life cycle. We first pre-processes, cleaned and transformed the data to prepare it for our data analysis models and solve any data quality issues. We implemented Exploratory Data Analysis to visualize the data and gain some high level insights about each feature in the dataset and how they are associated with one another. This helped us get an overview of the data and to understand its characteristics. Next, we created a Linear regression model and a Random forest regression model to identify the relationship between various characteristics of the song affecting popularity. We then implemented Principal Component Analysis for generating song and artist recommendations based on the similar characteristics of the song. We finally created a K-means clustering to group the similar songs into clusters.

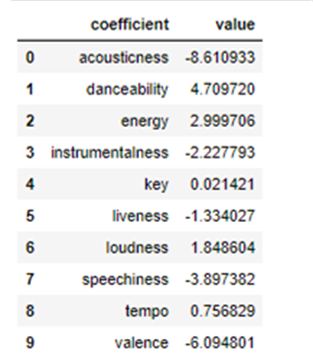
# **Models**

## **4.1 Linear Regression**

Linear regression is a simple Supervised Learning algorithm used to predict the value of a dependent variable(y) for an independent variable(x) value by modeling a linear relationship(y = mx + c) between the variables input(x) and output(y). In this project we used Linear Regression method to determine which predictor features are significant among a list of features available to us in the dataset to predict the popularity of the songs. The reason we first used this method is that in contrast to some of the other machine learning algorithms, linear regression has a substantially lower time complexity and is a very simple algorithm that can be applied very easily to produce adequate results.

From Fig 4.1.1 we can observe that we have obtained values for various features present in our dataset after applying Linear Regression. One of the interesting things we can notice here is that some of the values are negative (acousticness) and some of them are positive(key). Hence, we can say that the most important three features are: acousticness, danceability and valence whereas key is the least important feature.

Here we received a Root Mean Square Error value as 15.87 (approx.) a standard way to measure the error of a model in predicting quantitative data.



**(Fig 4.1.1 Relationship between features from Linear Regression)**

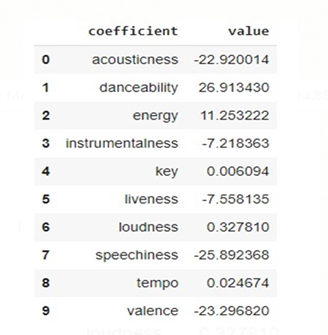
While utilizing linear regression in the project we found that it is highly vulnerable to outliers. So before applying Linear Regression to the dataset, outliers should be evaluated and eliminated. In summary, we found out that Linear Regression is a great method to evaluate the relationships between variables.

## **4.2 Random Forest Regression**

One of the benefits of random forests is that the majority of existing machine learning systems can be used for both classification and regression problems. In our project we have utilised the Random Forest Regression method for developing insights for features of songs affecting popularity. This algorithm is also very simple to calculate the relative value of each feature on the prediction. By looking at the value of the feature importance, it helped us to determine which features we can drop because they do not contribute enough (or sometimes nothing at all) to the prediction process.

From Fig 4.2.1 we can observe that we have obtained values for each of the features present in our dataset after applying Random Forest Regression. In the similar way when compared to the linear regression we have obtained values as negative (acousticness) and positive(key). Hence from also we can interpret that the important features are danceability, speechiness and valence. On the other hand, the least important features are key and tempo.

Here we received a Root Mean Square Error value as 14.89 (approx.).



**(Fig 4.1.1 Feature Importance from Random Forest Regression)**

*Comparison between the RMSE values*:

Based on the results when we can make a comparison of the RMSE values we can conclude that the RMSE value for Random Forest Regression is low as compared to Linear Regression obtained above. Also, we found out that Random Forest has an efficient way of calculating missing data and preserves precision when there is a significant proportion of missing data.

**Cross Validation and Train-Test results:**

Cross validation is a method of model validation which splits the data in creative ways in order to obtain the better estimates of “real world” model performance and minimize validation error. In this project we have performed Cross Validation by making a split in the ratio of 0.75 and 0.25 for train and test data respectively. We obtained a total of 126434 rows for train data whereas 42028 rows for test data with 15 columns each. We obtained the following MSE values for train and test data:

Mean Squared Error for Train data= 251.66739235223832

Mean Squared Error for Test data= 251.5384799678215

K-fold validation is a popular method of cross validation which shuffles the data and splits it into *k* number of folds (groups). In our project we have taken the value of K= 3 that means the process of shuffling or iteration will be performed for 3 times. Here, the data set is split into 3 folds. In the first iteration, the first fold is used to test the model and the rest are used to train the model. In the second iteration, 2nd fold is used as the testing set while the rest serve as the training set. This process is repeated until each fold of the 3 folds have been used as the testing set.

Based on the performance of the K fold validation we have obtained a Cross Validation Average metrics value as 251.73 (approx.)

## **4.3 Principal Component Analysis (PCA)**

Now we moved further to implement the PCA method for generating new variables, which are linear composites of the original variables known as principal components. First of all, we need to clarify the scree plot to view the PCA result. A scree graph helped us to understand how much variance is captured from the data by each main variable. From Fig 4.3.1 illustrates the scree plot we obtained which shows that at K=2 the plot tends to flatten which means there is minimum variation. Hence, we can say that the value of K=2 is optimal in our project.

## 

**(Fig 4.3.1 Scree Plot)**

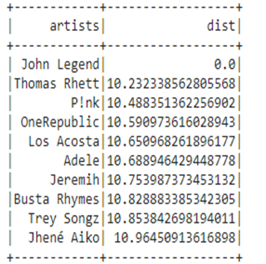
This plot helped us to analyse the trend of decreasing variability due to each successive component used in a principal component analysis or a factor analysis to select the number of relevant components or factors to be considered.

In this project we have used this method for the purpose of reducing dimensionality and for developing recommendations for artists and songs. From Fig 4.3.2 we can observe that when we provided an input as “Well Done” we obtained a list of different songs which are similar to the input and close based on the Euclidean distance value. The song followed by “Well Done” in the list is most similar and the song listed last in the results is least similar with “Well Done”.



**(Fig 4.3.2 Output for song recommendation)**

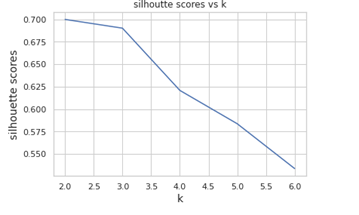
Similarly, for recommendation of artist Fig 4.3.3 shows that for the input artist as “John Legend” we have obtained a list of similar artists based on the Euclidean distance. For instance, “Thomas Rhett” is more similar whereas “Jhene Aiko” is least similar among the results we obtained with respect to the input artist name.



**(Fig 4.3.2 Output for artist recommendation)**

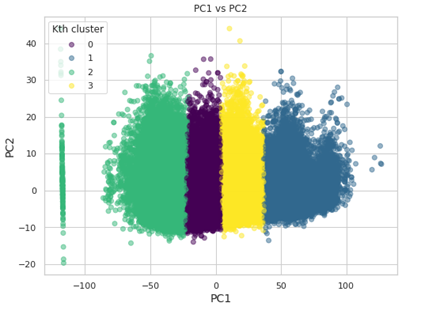
## **4.4 K-means Clustering**

In the fig 4.4.1 shows the silhouette scores vs K plot we created to measure how similar a data observation is to those on its own clusters compared with the observations in the other clusters. Using this method, what we did was to cluster our dataset several times using different k’s, followed by calculating and averaging the silhouette of all the observations, and selected the k we found the most appropriate. Based on our silhouette scores we selected K=3 with a value of 0.6901.



**(Fig 4.4.1 Average silhouette coefficient for different k’s)**

After PCA was applied in the project, we clustered the data again using K-means. We wanted to understand if there is a relation between songs and analyze them. Therefore, to do this through clustering analysis we used K-means clustering method to provide song recommendation based on recent user listening on Spotify The K-means cluster representations show clear divisions and separations in the clusters. For clustering, we want the points to be as similar as possible in the same cluster. Fig 4.4.1 shows what clustering looks like in our project for four clusters. From the plot generated we would conclude that the points with respect to K=0 is where we can observe that the cluster is more compact which means it has less variance which means that the value of K=0 is an optimal value in our project.



**(Fig 4.4.2 K means Cluster)**

# **Conclusion**

Many people complain that they don’t enjoy songs that music streaming services recommend. We decided to explore this problem by designing and implementing a system to recommend music to users based on the features of songs.

* We used the Spotify dataset, removed features that were poorly collected, cleaned the data, and performed feature engineering.
* By examining the various characteristics over time, we were able to find out whether there was a change in musical taste.
* For Linear Regression we obtained the Root Mean Square Error value as 15.87 and for Random Forest Regression the RMSE value as 14.89. We concluded that Random Forest Regression does a better job than Linear Regression model at predicting the significance of each feature affecting the popularity. Based on the results from Linear Regression and Random Forest Regression, acousticness and danceability tend to be the most important features. Thus, we can infer that as acousticness increases, the popularity of a song decreases indicating that people prefer less acoustic songs. Moreover, as danceability increases the popularity of a song also increases indicating that people prefer songs that they could dance to.
* We applied Cross validation technique to effectively split the data into Training set and Testing set, as per the performance of the K fold validation for K=3 we have obtained a Cross Validation Average metrics value as 251.73.
* We performed the Principal Component analysis for developing appropriate songs and artist recommendations for the input provided.