**MUSIC RECOMMENDATION SYSTEM**

**LET’S GROW MORE DATA SCIENCE INTERNSHIP**

BY: IHECHILURU WINNER

EMAIL: [ihechiluruwinner@gmail.com](mailto:ihechiluruwinner@gmail.com)

Linked-in: <https://www.linkedin.com/in/ihechiluru-winner/>

Problem Statement:

Music recommender systems that suggest songs to users based on their listening patterns.

Dataset Link: <https://www.kaggle.com/c/kkbox-music-recommendation-challenge/data>

## **Brief**

In this task, we were asked to predict the chances of a user listening to a song repetitively after the first observable listening event within a time window was triggered. If there are recurring listening event(s) triggered within a month after the user’s very first observable listening event, its target is marked 1, and 0 otherwise in the training set. The same rule applies to the testing set.

The train and the test data are selected from users listening history in a given time period. Note that this time period is chosen to be before the [WSDM-KKBox Churn Prediction](https://www.kaggle.com/c/kkbox-churn-prediction-challenge) time period. The train and test sets are split based on time, and the split of public/private is based on unique user/song pairs.

# **Tables**

## **train.csv**

* msno: user-id
* song\_id: song id
* source\_system\_tab: the name of the tab where the event was triggered. System tabs are used to categorize KKBOX mobile apps functions. For example, tab my library contains functions to manipulate the local storage, and tab search contains functions relating to the search.
* source\_screen\_name: name of the layout a user sees.
* source\_type: an entry point a user first plays music on mobile apps. An entry point could be an album, online playlist, song .. etc.
* target: this is the target variable. target=1 means there is recurring listening event(s) triggered within a month after the user’s very first observable listening event, target=0 otherwise.

## **test.csv**

* id: row id (will be used for submission)
* msno: user-id
* song\_id: song id
* source\_system\_tab: the name of the tab where the event was triggered. System tabs are used to categorize KKBOX mobile apps functions. For example, tab my library contains functions to manipulate the local storage, and tab search contains functions relating to the search.
* source\_screen\_name: name of the layout a user sees.
* source\_type: an entry point a user first plays music on mobile apps. An entry point could be an album, online playlist, song .. etc.

## **songs.csv**

The songs. Note that data is in Unicode.

* song\_id
* song\_length: in ms
* genre\_ids: genre category. Some songs have multiple genres and they are separated by |
* artist\_name
* composer
* lyricist
* language

## **members.csv**

user information.

* msno
* city
* bd: age. Note: this column has outlier values, please use your judgement.
* gender
* registered\_via: registration method
* registration\_init\_time: format %Y%m%d
* expiration\_date: format %Y%m%d

## **song\_extra\_info.csv**

* song\_id
* song name - the name of the song.
* isrc - [International Standard Recording Code](https://en.wikipedia.org/wiki/International_Standard_Recording_Code), theoretically can be used as an identity of a song. However, what is worth noting is, ISRCs generated from providers have not been officially verified; therefore the information in ISRC, such as country code and reference year, can be misleading/incorrect. Multiple songs could share one ISRC since a single recording could be re-published several times.

**Algorithms Used:**

1. **KNN Algorithm:** K- Nearest Neighbour [Classification] K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data. K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a good suite category by using K- NN algorithm. It comes under the supervised learning technique.
2. **KMeans Algorithm:**

K-means is a centroid-based unsupervised machine learning algorithm, or a distance-based algorithm, where we calculate the distances to assign a point to a cluster. In K-Means, each cluster is associated with a centroid.

***The main objective of the K-Means algorithm is to minimize the sum of distances between the points and their respective cluster centroid.***

**3. Tools Used:**

Tools used in the major project are:

a) Google Colab

b) Pandas Library

c) Seaborn Function to Plot Graph, and

d) Sklearn – Machine Learning website to import all the algorithms,

confusion matrix and accuracy score.

**• KNN Algorithm:**

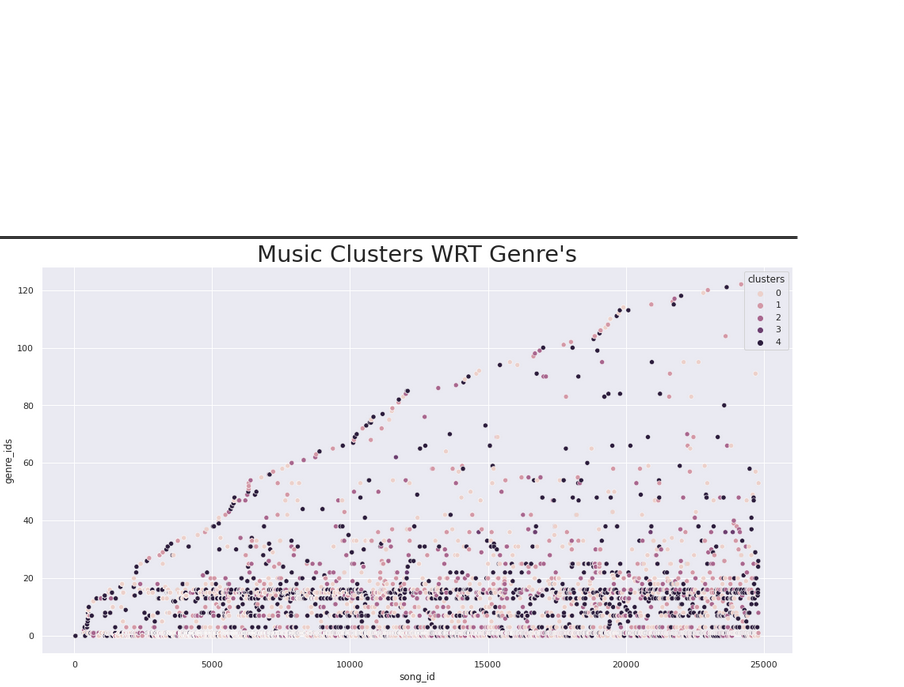
o Training Score= 0.957983

o Testing Score= 0.95

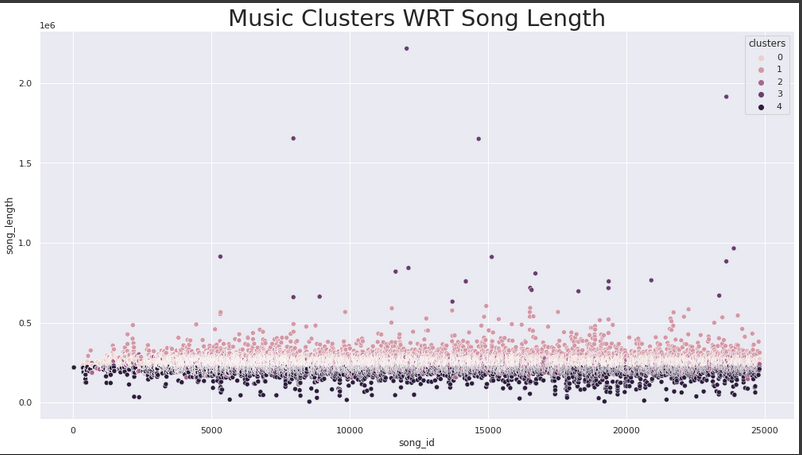
o Accuracy Score= 0.95

**KMeans Algorithm:**

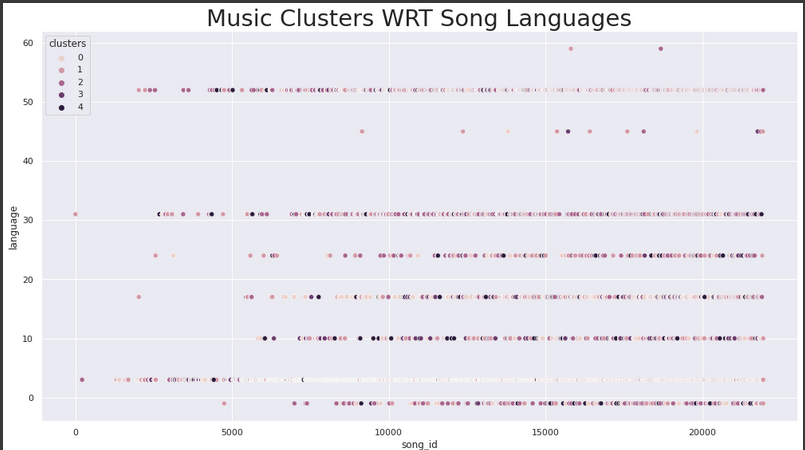
Clusters Representing which genre’s users listen to



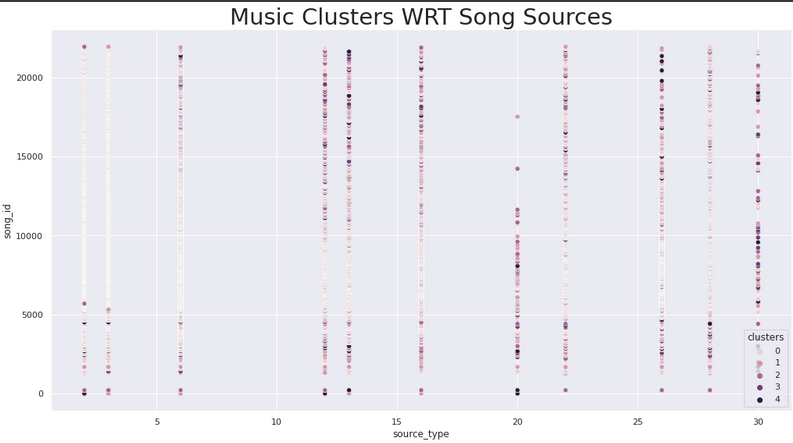
Song Lengths Users Listen to Most



Clusters For The Languages User’s Listen To:



Clusters for The Sources Song Sources



Kindly access the link below for my google colab notebook

<https://colab.research.google.com/drive/1JpJzEnM7J9KMYRVMMRwbnwVCq3lTp2-N?usp=sharing>

Regards,

Ihechiluru Winner.