VIT
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

Spotify Recommendation System

Dolly Agarwala[20BCE1863]
UG Scholar School of Computer Science and Engineering, Vellore Institute of Technology University,
Chennai, India
dolly.agarwala2020@vitstudent.ac.in

Dhruvi Ochani[20BCE1882]
UG Scholar School of Computer Science and Engineering, Vellore Institute of Technology University,
Chennai, India
dhruvi.ochani2020@vitstudent.ac.in

Nayan Khemka[20BCE1884]
UG Scholar School of Computer Science and Engineering, Vellore Institute of Technology University,
Chennai, India
nayan.khemka2020@vitstudent.ac.in

Aryan Bhardwaj[20BCE1908]
UG Scholar School of Computer Science and Engineering, Vellore Institute of Technology University,
Chennai, India
aryan.bhardwaj2020@vitstudent.ac.in

N.Aishwarya[20BRS1143]
UG Scholar School of Computer Science and Engineering, Vellore Institute of Technology University,
Chennai, India
aishwarya.n2020@vitstudent.ac.in

***Abstract*— This project focuses on the provision of a music recommendation system, in which music websites would offer recommendations on songs based on the user's interests, and can help users who find it difficult to decide which songs to listen to in online music.**

Keywords — recommendation system, user-based collaborative filtering, KNN,KMEDOID,KMEANS,Naïve Bayes

1. INTRODUCTION

By offering music recommendations, this method helps the user find the tracks they would like to listen to without having to search the entire collection. Users of music recommenders can filter and find tracks based on their preferences. A decent music recommendation engine should always be able to identify interests dynamically and create playlists based upon these tastes. However, the industry has a fantastic potential to gather users who are interested in music because of the creation of recommender systems. More importantly, it poses challenges for us in better modeling and understanding customer musical preferences

2. LITREATURE SURVEY

1] Current challenges and visions in music recommender systems research[1]

The paper first tries to identify and shed light on what are believed to be the most pressing challenges Music Recommendation Systems (MRS) research is facing, from both academic and industry perspectives. The paper identified several grand challenges the research field of music recommender systems (MRS) is facing. It also presented a visionary outlook of (1) psychologically inspired MRS, which consider in the recommendation process factors such as listeners' emotion and personality, (2) situation-aware MRS, which holistically model contextual and environmental aspects of the music consumption process, infer listener needs and intents, and eventually integrate these models at large scale in the recommendation process, and (3) culture-aware MRS, which exploits the fact that music taste highly depends on the cultural background of the listener, where culture can be defined in manifold ways, including historical, political, linguistic, or religious similarities.

2]Combining Spotify and Twitter data for generating a recent and public dataset for music recommendation[2]

This paper attempts to present a dataset based on listening and streaming histories of Spotify users who have posted what they're listening to on Twitter.The authors have implemented and collaborated on a filtering based collaborative system based to show an application of this dataset. The authors used a User-based CF technique to demonstrate the value of our dataset. User-based CF only makes product recommendations based on previous user-item interactions. A user-item interaction for the music recommendation system indicates that a user listened to a certain track by a particular artist. Therefore, previous user-item interactions serve as a representation of a user's listening history.The authors have demonstrated the usefulness of the supplied dataset for comparing and contrasting various methods for music recommendation. The study therefore identifies two areas where action is required: The first step is to add more context-based information to the dataset, such as the time stamp or the geolocation. Second, hybrid recommender systems that make use of this extra context-based data are of interest.

3]Design and test music recommendation system for online music websites using collaborative filtering approach[3]

This study aims to create a music recommendation system on online music sites to provide convenience for consumers in choosing music products according to their preferences. This study explains the problem of users of online music sites who do not rank a song that has been heard, and overcome it with a recommendation system using user-based collaborative filtering methods. The method used in this study is the RAD(Rapid Application Development) method with four existing phases, and at the assessment stage involves the analyst and user. RAD is appropriate for producing a software system with urgent needs and a short time in its completion. When the system presents so much data from many items and the user is faced with a variety of choices, the existence of a rating can be used as help to help a user in making choices and narrowing down searches. The user-based collaborative filtering recommendation system has two-stage namely the distance measurement step (metric calculation) and the step of providing recommendation through rating prediction. At the distance measurement stage, rating data is to measure the distance from an active user to another user. After getting a rating prediction value, the system then provides music recommendation based on the rating prediction with the highest value, and displays according to the value of n.

4]A Survey of Music Recommendation Systems and Future Perspectives[4]

This paper proposes a motivation-based model using the empirical studies of human behavior, sports education, music psychology. This paper explains basic metadata-based model and two popular music recommender approaches: collaborative filtering and content-based model. Though they have achieved great success, their drawbacks such as popularity bias and human efforts are obvious. Moreover, the use of hybrid models would outperform a single model since it incorporates the advantages of both methods.Its complexity is not fully studied yet. Due to the subjective nature in music and the issues existing in the previous methods, two human-centered approaches are proposed. By considering affective and social information, emotion-based model and context-based model largely improved the quality of recommendation. However, this research is still at an early stage.

5] Music Recommender System Based on Genre using Convolutional Recurrent Neural Networks[5]

The research develops a music recommender system that can give recommendations based on similarity of features on audio signals. It uses convolutional recurrent neural networks (CRNN) for feature extraction and similarity distance to look for similarity between features. The results of this study indicate that users prefer recommendations that consider music genres compared to recommendations based solely on similarity.The experiments are based on user responses to given music recommendations. The following are the conclusions based on experiment results. First, the music recommender system should consider the music genre information to increase the quality of music recommendations. Second, CRNNs that consider both the frequency features and time sequence patterns have overall better performance. It indicates the effectiveness of its hybrid structure to extract the music features. Based on our analyses, we can suggest for future research to add other music features in order to improve the accuracy of the recommender system, such as using tempo gram for capturing local tempo at a certain time.

6] Music Recommendation System using Machine Learning[6]

In this project, a sample data set of songs has been used to find correlations between users and songs so that a new song will be recommended to them based on their previous history. This has been implemented in this project using libraries like NumPy, Pandas. Cosine similarity will also be used along with CountVectorizer. Along with this,a front end with flask that will show us the recommended songs when a specific song is processed. Feature selection is one of the main concepts of DM and Machine Learning. Where it is a process of selecting necessary useful variables in a dataset to improve the results of machine learning and make it more accurate, there are a lot of columns in the predictor variable. So, the correlation coefficient is calculated to see which of them are important and these are then used for training methods..

7]Combining usage and content in an online recommendation system for music in the Long Tail[7]

The study presents a hybrid music recommender system that integrates use and content data in this research. The system examines the collection of things assessed by users and computes the similarity between pairs of items, providing a matrix that represents the similarities between all pairs of items based on a similarity measure. The similarity is calculated by the Euclidean distance through the 16 audio features.
Usage-based recommendation is made on the basis of the similarity matrix between tracks. For each recommendable music track m its k closest neighbors N(m) is fetched . These are the k tracks with maximum similarity to m. This research proposes and assesses a music recommender system that integrates usage content data.Few metrics have been suggested for contrasting the effectiveness of the options. Arriving at the conclusion that Mix is a superior alternative than the other two systems overall for providing music recommendations.

3. PROPOSED WORK

A. Data Source

The data source used for our project falls under primary data which is raw data collected directly from sources as the dataset consists of soundtracks and information regarding their features like release date, tempo, acousticness, etc. [8]

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	valence	year	acousticne	artists	danceabili	duration_r	energy	explicit	id	instrument	key	liveness	loudness	mode	name	popularity	release_da	speechine	tempo
2	0.0594	1921	0.982	['Sergei Ra	0.279	831667	0.211	0	4BJqT0PrA	0.878	10	0.665	-20.096	1	Piano Con	4	1921	0.0366	80.954
3	0.963	1921	0.732	['Dennis Da	0.819	180533	0.341	0	7xPhfUan2	0	7	0.16	-12.441	1	Clancy Lov	5	1921	0.415	60.936
4	0.0394	1921	0.961	['KHP Kridh	0.328	500062	0.166	0	1o6I8BglA	0.913	3	0.101	-14.85	1	Gati Bali	5	1921	0.0339	110.339
5	0.165	1921	0.967	['Frank Par	0.275	210000	0.309	0	3ftBPsC5v	2.77E-05	5	0.381	-9.316	1	Danny Boy	3	1921	0.0354	100.109
6	0.253	1921	0.957	['Phil Rega	0.418	166693	0.193	0	4d6HGyGT	1.68E-06	3	0.229	-10.096	1	When Irish	2	1921	0.038	101.665
7	0.196	1921	0.579	['KHP Kridh	0.697	395076	0.346	0	4pyw9DVH	0.168	2	0.13	-12.506	1	Gati Mardi	6	1921	0.07	119.824
8	0.406	1921	0.996	['John McC	0.518	159507	0.203	0	5uNZnElqC	0	0	0.115	-10.589	1	The Weari	4	1921	0.0615	66.221
9	0.0731	1921	0.993	['Sergei Ra	0.389	218773	0.088	0	02GDntOX	0.527	1	0.363	-21.091	0	Morceaux	2	1921	0.0456	92.867
10	0.721	1921	0.996	['Ignacio C	0.485	161520	0.13	0	05xDjWH9	0.151	5	0.104	-21.508	0	La MaÃ±ar	0	########	0.0483	64.678
11	0.771	1921	0.982	['FortugÃ©	0.684	196560	0.257	0	08zfJvRLp	0	8	0.504	-16.415	1	Il Etait Syn	0	1921	0.399	109.378
12	0.826	1921	0.995	['Maurice	0.463	147133	0.26	0	0BMkRpQt	0	9	0.258	-16.894	1	Dans La Vi	0	1921	0.0557	85.146
13	0.578	1921	0.994	['Ignacio C	0.378	155413	0.115	0	0F30WM8	0.906	10	0.11	-27.039	0	Por Que M	0	########	0.0414	70.37
14	0.493	1921	0.99	['Georgel']	0.315	190800	0.363	0	0H3k2CvJv	0	5	0.292	-12.562	0	La VipÃ¨re	0	1921	0.0546	174.532
15	0.212	1921	0.912	['Mehmet	0.415	184973	0.42	0	0LcXzABeA	0.89	8	0.108	-10.766	0	Ud Taksim	0	1921	0.114	70.758
16	0.493	1921	0.0175	['Zay Gatsb	0.527	205072	0.691	1	0MJZ4hh6	0.384	7	0.358	-7.298	1	Power Is P	0	########	0.0326	159.935
17	0.282	1921	0.989	['Sergei Ra	0.384	221013	0.171	0	0NFeJgmT	0.82	7	0.116	-20.476	0	10 PrÃ©lu	4	1921	0.0319	107.698
18	0.218	1921	0.957	['Phil Rega	0.259	186467	0.212	0	0Nk5f07H	0.000222	2	0.236	-13.3	1	Come Bacl	1	1921	0.0358	85.726
19	0.664	1921	0.996	['Hector B	0.541	250747	0.283	0	0POO8XaU	0.898	9	0.393	-14.808	1	RÃ¡kÃ³czy	0	1921	0.0477	108.986
20	0.0778	1921	0.148	['THE GUY	0.604	204957	0.418	1	0QQmUf4	0.0382	4	0.102	-11.566	0	When We	0	########	0.0417	80.073
21	0.527	1921	0.971	['Christoph	0.54	122000	0.0848	0	0QU5xT6N	0.00196	5	0.0887	-16.055	0	A Ballynure	0	1921	0.075	100.296
22	0.672	1921	0.994	['FortugÃ©	0.67	191333	0.113	0	0QvUUTHY	0	10	0.213	-16.57	0	Je Suis Tou	0	1921	0.196	87.162
23	0.24	1921	0.994	['John McC	0.4	187333	0.155	0	0RPKAq5yl	4.33E-05	4	0.103	-13.976	1	Mother Ma	0	1921	0.0873	170.251
24	0.422	1921	0.995	['Ignacio C	0.648	154240	0.0995	0	0SK1upzAF	0.846	11	0.112	-22.429	1	Flor March	0	########	0.105	71.978
25	0.381	1921	0.995	['Hanende	0.223	158908	0.245	0	0UqiUmGD	0.876	7	0.354	-14.387	1	Ä°mtidadÄ	0	########	0.0387	142.136
26	0.41	1921	0.97	['Mehmet	0.269	86988	0.143	0	0VhIFYGSp	0.469	4	0.166	-9.003	1	Ney Taksir	0	1921	0.0413	141.386
27	0.0731	1921	0.993	['Sergei Ra	0.389	218773	0.088	0	0eQsdik7G	0.527	1	0.363	-21.091	0	Morceaux	0	1921	0.0456	92.867
28	0.678	1921	0.996	['Maurice	0.5	181733	0.274	0	0i7MdVu0	0	2	0.302	-14.001	1	Je M'donn	0	1921	0.041	79.218
29	0.723	1921	0.388	['Mehmet	0.685	155063	0.698	0	0osXBirvQ	1.97E-06	4	0.0768	-8.184	0	Korkma SÃ	0	1921	0.0421	133.951

B. *Exploratory Data Analytics*

```
summary(spotify_df)
```

```
##      track              artist
uri              danceability

##   Length:41099       Length:41099
Length:41099        Min.   :0.0000

##   Class :character   Class :character
Class :character    1st Qu.:0.4200

##   Mode  :character   Mode  :character
Mode  :character    Median :0.5520

##
Mean   :0.5397


##
3rd Qu.:0.6690

##
Max.   :0.9880

##       energy                key
loudness              mode

##   Min.   :0.000251   Min.   : 0.000
Min.   :-49.253   Min.   :0.0000

##   1st Qu.:0.396000   1st Qu.: 2.000
1st Qu.:-12.816   1st Qu.:0.0000

##   Median :0.601000   Median : 5.000
Median : -9.257   Median :1.0000

##   Mean   :0.579544   Mean   : 5.214
Mean   :-10.221   Mean   :0.6934

##   3rd Qu.:0.787000   3rd Qu.: 8.000
3rd Qu.: -6.375   3rd Qu.:1.0000
```

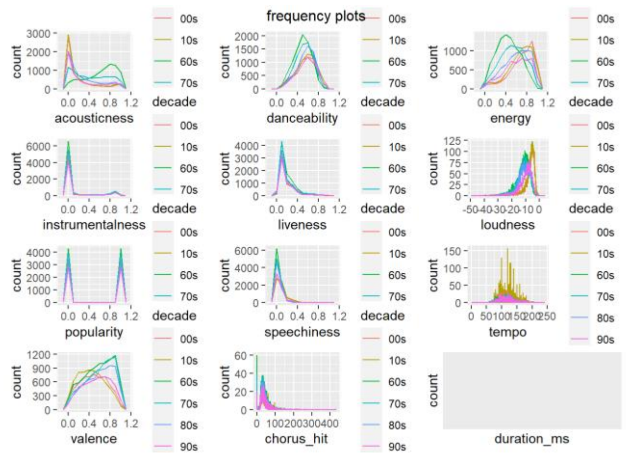
```
##   Max.   :1.000000   Max.   :11.000
Max.   :  3.744   Max.   :1.0000

##    speechiness         acousticness      i
nstrumentalness      liveness

##   Min.   :0.00000   Min.   :0.0000   M
in.   :0.00000   Min.   :0.0130

##   1st Qu.:0.03370   1st Qu.:0.0394   1
st Qu.:0.00000   1st Qu.:0.0940

##   Median :0.04340   Median :0.2580   M
edian :0.00012   Median :0.1320

##   Mean   :0.07295   Mean   :0.3642   M
ean   :0.15440   Mean   :0.2015

##   3rd Qu.:0.06980   3rd Qu.:0.6760   3
rd Qu.:0.06120   3rd Qu.:0.2610

##   Max.   :0.96000   Max.   :0.9960   M
ax.   :1.00000   Max.   :0.9990

##      valence            tempo          du
ration_ms        time_signature

##   Min.   :0.0000   Min.   :  0.0   Min
.   :  15168   Min.   :0.000

##   1st Qu.:0.3300   1st Qu.: 97.4   1st
Qu.: 172916   1st Qu.:4.000

##   Median :0.5590   Median :117.6   Med
ian : 217907   Median :4.000

##   Mean   :0.5425   Mean   :119.3   Mea
n   : 234876   Mean   :3.894

##   3rd Qu.:0.7680   3rd Qu.:136.5   3rd
Qu.: 266773   3rd Qu.:4.000

##   Max.   :0.9960   Max.   :241.4   Max
.   :4170227   Max.   :5.000
```

```
##     chorus_hit        sections
popularity      decade

##  Min.    :  0.00   Min.    :  0.00    Mi
n.    :0.0    Length:41099

##  1st Qu.: 27.60   1st Qu.:   8.00    1s
t Qu.:0.0    Class :character

##  Median : 35.85   Median : 10.00    Me
dian :0.0    Mode  :character

##  Mean   : 40.11   Mean   : 10.48    Me
an    :0.5

##  3rd Qu.: 47.63   3rd Qu.: 12.00    3r
d Qu.:1.0

##  Max.   :433.18   Max.   :169.00    Ma
x.    :1.0
```

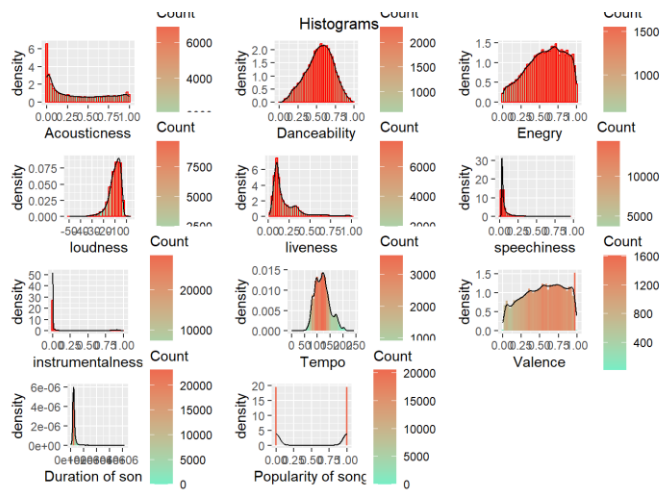
```
dim(spotify_df)
```

```
## [1] 41099    20
```

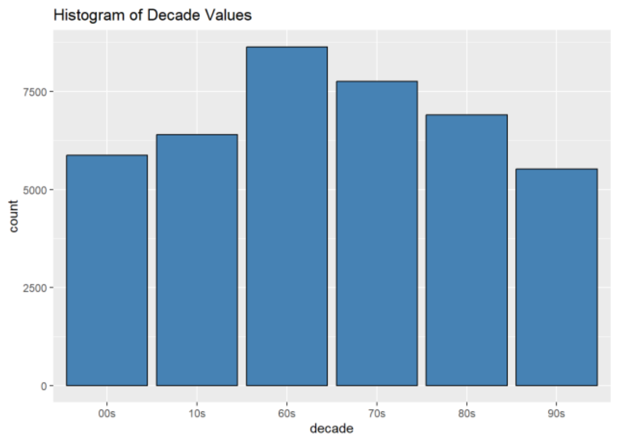
```
#Frequency plots
```



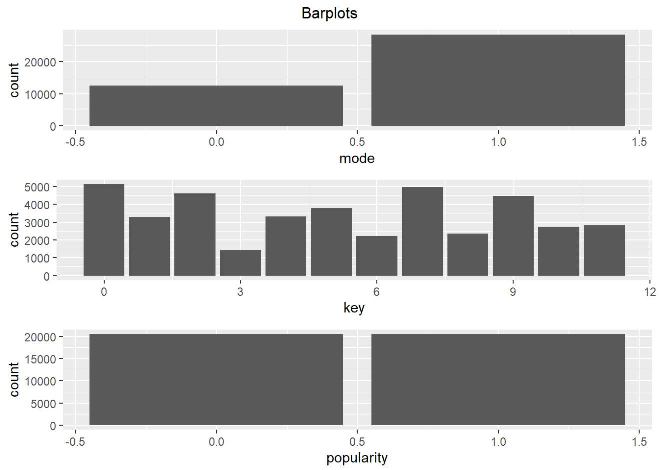

```
## Histograms
```



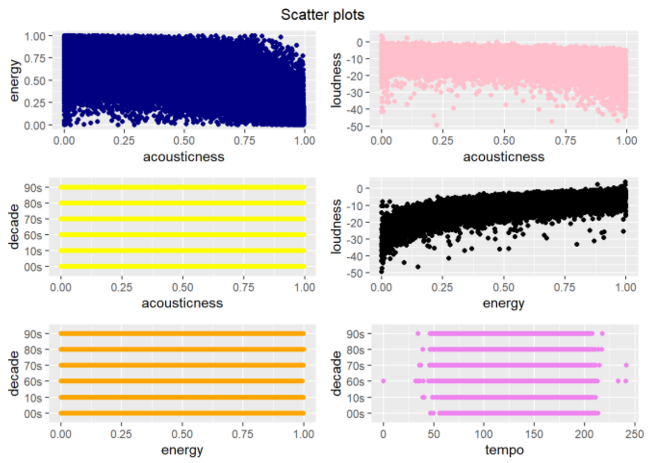

```
#create histogram of values for decade
```

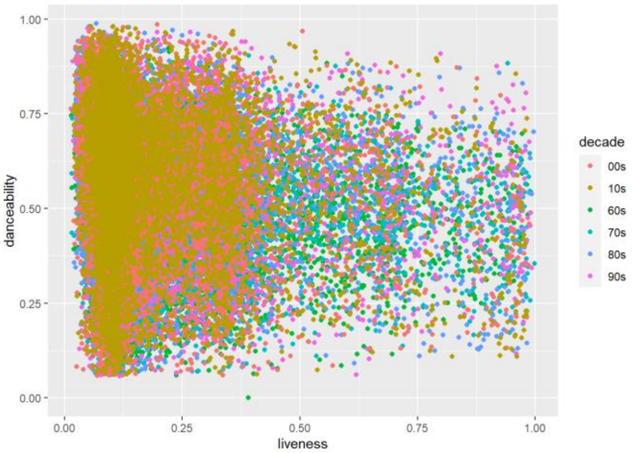


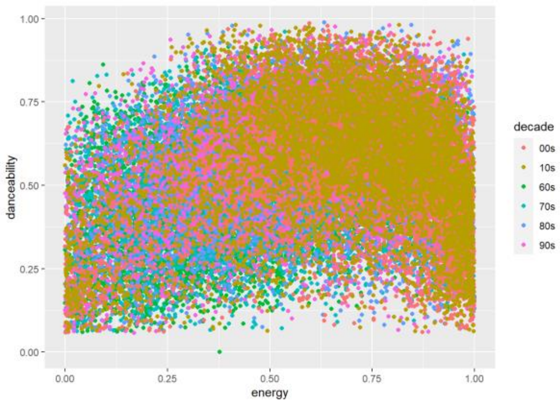

```
#bar plots
```



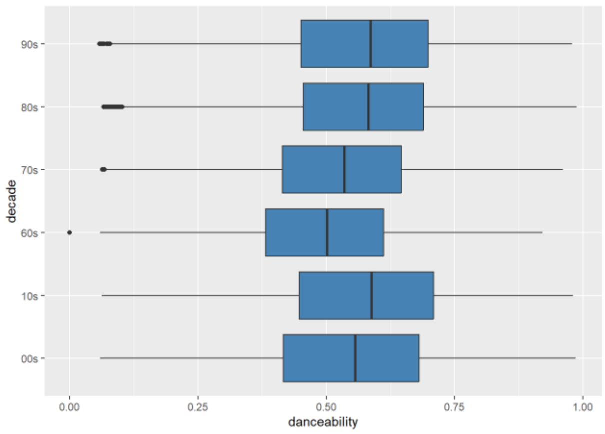

```
#Scatter plots
```



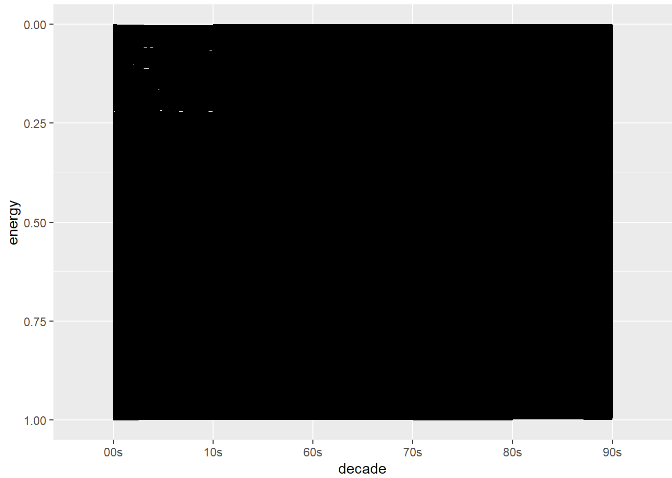


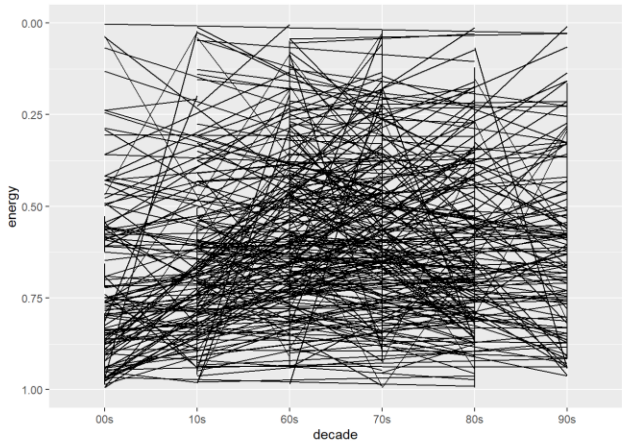



```
#Box plot
```

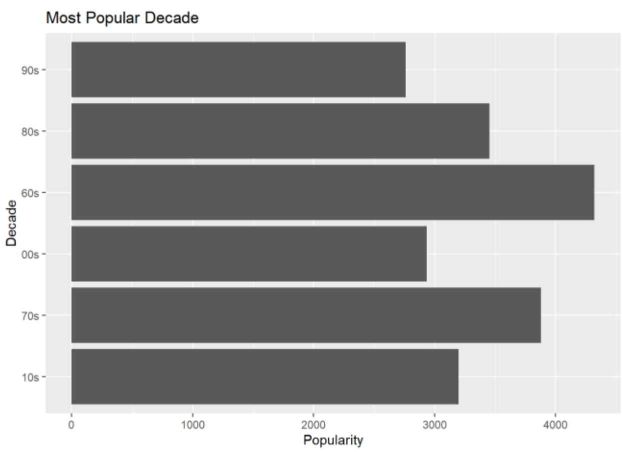



```
#line plot for all records vs sample 1000 record
```






```
#most popular decade
```



C. ANALYTICAL MODELS

Descriptive: Segregating the tracks which are popular and which aren't and looking into their characteristics.

Diagnostic: Identifying the reason why there are certain tracks which are more popular than other tracks and what characteristics sets them apart.

Predictive:

Predicting which songs would be popular on the basis of the analysis done previously with respect to the features of the songs.

Prescriptive:

After analysis we figure out what features need to be prioritized while working/selecting a song so as to make it a popular/favorable hit.

D. DATA ANALYTICS MODELS

D.1 THEORY

a) KNN - One of the simplest machine learning algorithms, based on the supervised learning method, is K-Nearest Neighbour.
The K-NN algorithm makes the assumption that the new case and the existing cases are comparable, and it places the new instance in the category that is most like the existing categories. A new data point is classified using the K-NN algorithm based on similarity after all the existing data has been stored. This means that utilising the K-NN method, fresh data can be quickly and accurately sorted into a suitable category. Although the K-NN approach is most frequently employed for classification problems, it can also be utilised for regression.

b) KMEDOID - The data set of n items is divided into k clusters using the traditional clustering partitioning technique known as k-medoids, where the number of clusters is supposed to be known in advance (which implies that the programmer must specify k before the execution of a k-medoids algorithm). Methods like the silhouette approach can be used to judge the "goodness" of the supplied value of k.

c) NAIVE BAYES - The Naive Bayes algorithm, which is based on the Bayes theorem, is used to resolve classification issues. The major application for it is text classification with a large training set. One of the most straightforward and efficient classification methods is the naive bayes algorithm, which aids in the development of quick machine learning models capable of making accurate predictions. It makes predictions based on object probabilities because it is a probabilistic classifier.

d) KMEANS - K-Means Clustering divides the unlabeled dataset into various clusters. Here, K specifies how many pre-defined clusters must be produced as part of the process; for example, if K=2, there will be two clusters, if K=3, there will be three clusters, and so on. It gives us the ability to divide the data into various groups and provides a practical method for automatically identifying the groups in the unlabeled dataset without the need for any training. Each cluster has a centroid assigned to it because the algorithm is centroid-based. This algorithm's primary goal is to reduce the total distances between each data point and its corresponding clusters.

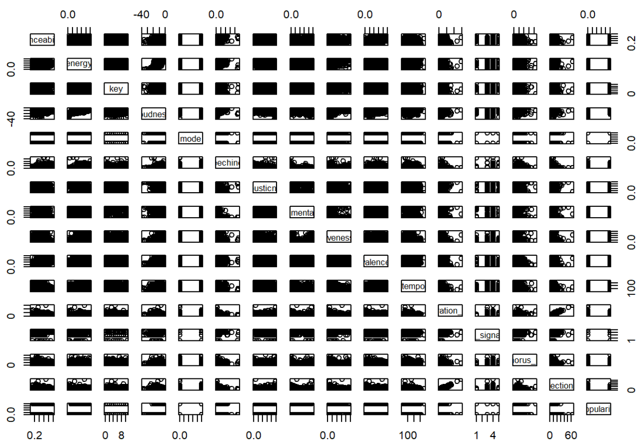
D.2 MODELS

a) KMEANS

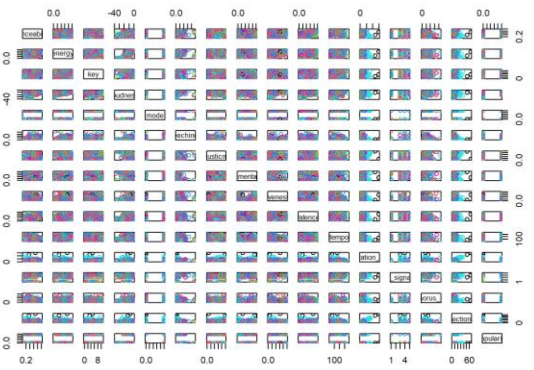
```
# Fitting K-Means clustering Model to tr
aining dataset

kmeans.re$cluster

# Confusion Matrix

cm <- table(data_spotify_df$decade, kmea
ns.re$cluster)

cm
```

```
##        1  2  3  4  5  6

##   00s  0 37 22 47  6 18

##   10s  0 46 25 59 10 24

##   60s  0 51 33 64 16 35

##   70s  0 66 37 45  6 36

##   80s  0 67 20 64  8 29

##   90s  2 46 18 33  5 25
```

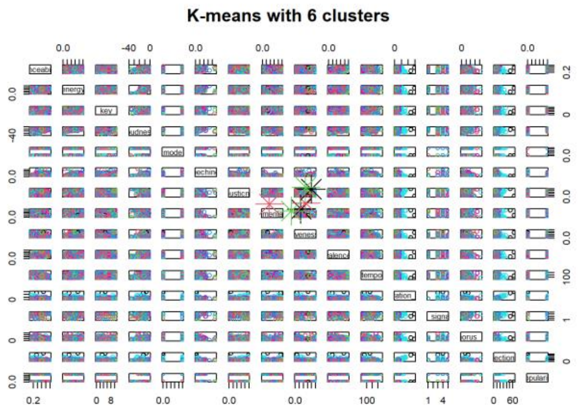
```
# Model Evaluation and visualization

plot(data_new)
```



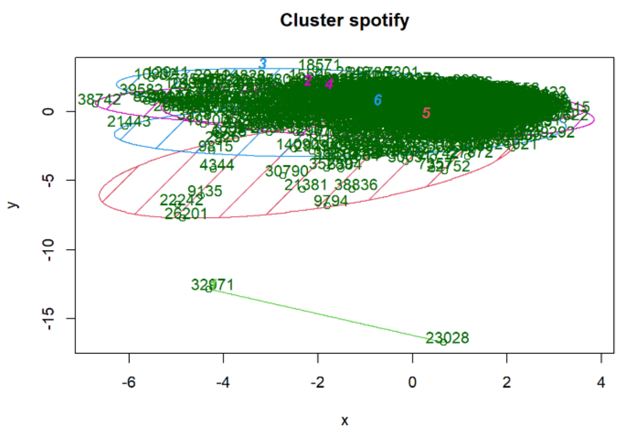
```
plot(data_new, col = kmeans.re$cluster)
```



```
plot(data_new, col = kmeans.re$cluster,
main = "K-means with 6 clusters")

## Plotting cluster centers
```




```
# Visualizing clusters
```



These two components explain 32.86 % of the point variability.

b) *KNN*

```
# Fitting KNN Model to training dataset

classifier_knn <- knn(train = train_scale, te
st = test_scale, cl = train_cl$decade, k = 1)

classifier_knn
```

```
##        classifier_knn
##          00s  10s  60s  70s  80s  90s
##    00s  624  566  116  192  244  331
##    10s  568  841  154  168  260  265
##    60s  146  157 1584  695  302  167
##    70s  182  215  668  838  509  327
##    80s  252  248  285  497  738  419
##    90s  329  253  211  317  417  420
```

```
# Model Evaluation - Choosing K
```

K	*ACCURACY*
2	0.34781109962082
3	0.359944846604619
5	0.379317476732161
7	0.395105136159945
15	0.411375387797311
19	**0.41709755256808**

c) *NAÏVE BAYE'S ALGORITHM*

```
# Fitting Naive Bayes Model
# to training dataset

classifier_cl <- naiveBayes(decade ~ .,
data = train_cl)

classifier_cl
## Naive Bayes Classifier for Discrete P
redictors

##

## Call:

## naiveBayes.default(x = X, y = Y, lapl
ace = laplace)

##
```

```
## A-priori probabilities:
## Y
##       00s       10s       60s       7
0s       80s       90s
## 0.1429162 0.1556249 0.2103324 0.18882
54 0.1681080 0.1341931
```

```r
# Predicting on test data'
y_pred <- predict(classifier_cl, newdata
= test_cl)
# Confusion Matrix
cm <- table(test_cl$decade, y_pred)
cm
```

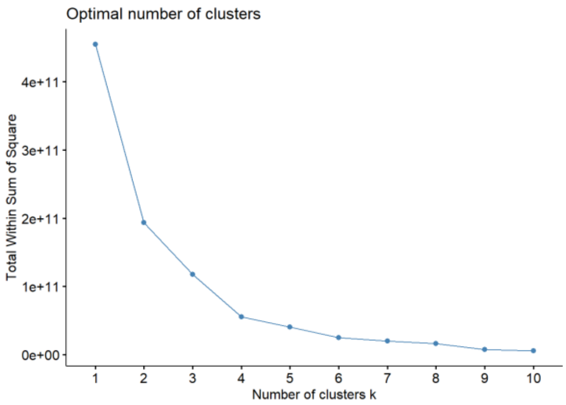
```
##      y_pred
##         00s  10s  60s  70s  80s  90s
##   00s  144 1062  261   81  514    8
##   10s  102 1320  332   71  425    7
##   60s   22  121 1924  177  797    7
##   70s   68  216  951  228 1271    8
##   80s   65  319  502  142 1386   22
##   90s   67  483  394  122  869   15
```

```r
# Model Evaluation
confusionMatrix(cm)
```

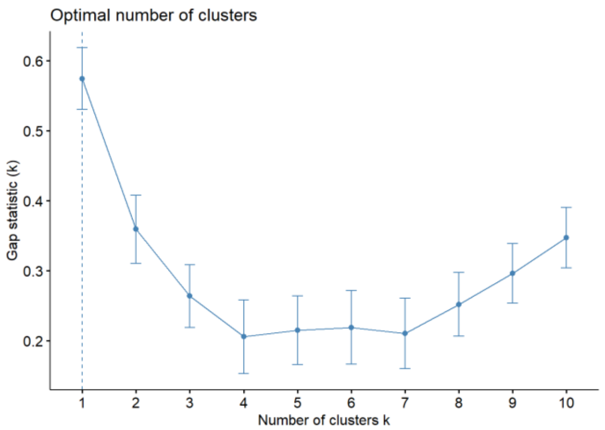
```
## Confusion Matrix and Statistics
##
##      y_pred
##         00s  10s  60s  70s  80s  90s
##   00s  144 1062  261   81  514    8
##   10s  102 1320  332   71  425    7
##   60s   22  121 1924  177  797    7
##   70s   68  216  951  228 1271    8
##   80s   65  319  502  142 1386   22
##   90s   67  483  394  122  869   15
##
## Overall Statistics
##
##  Accuracy : 0.3459
##  95% CI : (0.3382, 0.3537)
##     No Information Rate : 0.3628
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.2044
##
##  Mcnemar's Test P-Value : <2e-16
##
## Statistics by Class:
##
##                      Class: 00s Class
: 10s Class: 60s Class: 70s Class: 80s
## Sensitivity            0.307692    0.
37489     0.4409    0.27771    0.26340
## Specificity            0.862772    0.
91468     0.8891    0.81625    0.88638
## Pos Pred Value         0.069565    0.
58485     0.6312    0.08315    0.56897
## Neg Pred Value         0.973940    0.
82027     0.7870    0.94958    0.67879
## Prevalence             0.032269    0.
24278     0.3009    0.05661    0.36282
## Detection Rate         0.009929    0.
09102     0.1327    0.01572    0.09557
## Detection Prevalence   0.142729    0.
15562     0.2102    0.18906    0.16797
## Balanced Accuracy      0.585232    0.
64479     0.6650    0.54698    0.57489
##                      Class: 90s
## Sensitivity            0.223881
## Specificity            0.865960
## Pos Pred Value         0.007692
## Neg Pred Value         0.995858
## Prevalence             0.004620
## Detection Rate         0.001034
## Detection Prevalence   0.134455
## Balanced Accuracy      0.544920
```

d) K- MEDOID

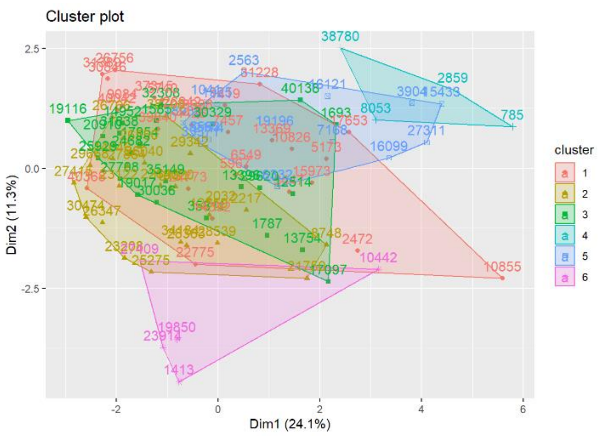
```
fviz_nbclust(data_new, pam, method = "ws
s")
```




```
#plot number of clusters vs. gap statist
ic

fviz_gap_stat(gap_stat)
```




```
#perform k-medoids clustering with k = 6
clusters

#plot results of final k-medoids model

fviz_cluster(kmed, data = data_new)
```



4. RESULTS AND ANALYSIS

In this project we have used machine learning algorithms such as K-Means and K-Mediod which clusters songs so that it can recommend users based on the cluster their recent listening history falls in. The most efficient way for our data, according to our analysis is using K-Mediod. It can effectively find the cluster your song history lies in. It finds songs with similarities to user's past listening history.

We have also implemented 2 classification algorithms KNN and Naïve Baye's which classifies songs based on the various features into the decade the song belongs in. KNN with k value of 14 seems to have the maximum accuracy. This will also let us know what decade the user mostly listens to and hence will help in recommending songs as well.

Our findings allow us to identify additional music aspects for future research in order to increase the recommender system's accuracy, such as employing tempo metre to record the local tempo at a certain time.

In future we can combine both the clustering and classification algorithms to make a better predicting model, which could offer an even better solution to our problem statement.

5. REFRENCES

[1] Schedl, M., Zamani, H., Chen, CW. et al. Current challenges and visions in music recommender systems research. Int J Multimed Info Retr 7, 95–116 (2018). https://doi.org/10.1007/s13735-018-0154-2

[2] Pichl, Martin & Zangerle, Eva & Specht, Guenther. (2015). Combining Spotify and Twitter Data for Generating a Recent and Public Dataset for Music Recommendation. 1313.

[3] Oswari, Teddy & Yusnitasari, Tristyanti & Kusumawati, Reni & Mittal, Saurabh. (2020). Design and test music recommendation system for online music websites using collaborative filtering approach. International Journal of Digital Signals and Smart Systems. 4. 64. 10.1504/IJDSSS.2020.106073.

[4] Song, Yading & Dixon, Simon & Pearce, Marcus. (2012). A Survey of Music Recommendation Systems and Future Perspectives.

[5] 4th International Conference on Computer Science and Computational Intelligence 2019 (ICCSCI), 12–13 September 2019, Music Recommender System Based on Genre using Convolutional Recurrent Neural Networks

[6] International Journal of Scientific Research in Computer Science, Engineering and Information Technology (Nov-Dec 2021) Music Recommendation System using Machine Learning

[7] Domingues, M.A., Gouyon, F., Jorge, A.M. et al. Combining usage and content in an online recommendation system for music in the Long Tail. Int J Multimed Info Retr 2, 3–13 (2013). https://doi.org/10.1007/s13735-012-0025-1

[8] Source dataset https://www.kaggle.com/code/akiboy96/spotify-song-popularity-genre-exploration/data