SPOTIFY SONGS POPULARITY VISUALIZATION AND ANALYSIS

Simrik Rijal (300340875)

Sisir Ghimire Chettri (300340871)

OVERVIEW

Spotify tracks dataset from Kaggle which was collected from Spotify's Web API (2022)

25 MB file size (CSV)

114,000 rows of data

125 genres

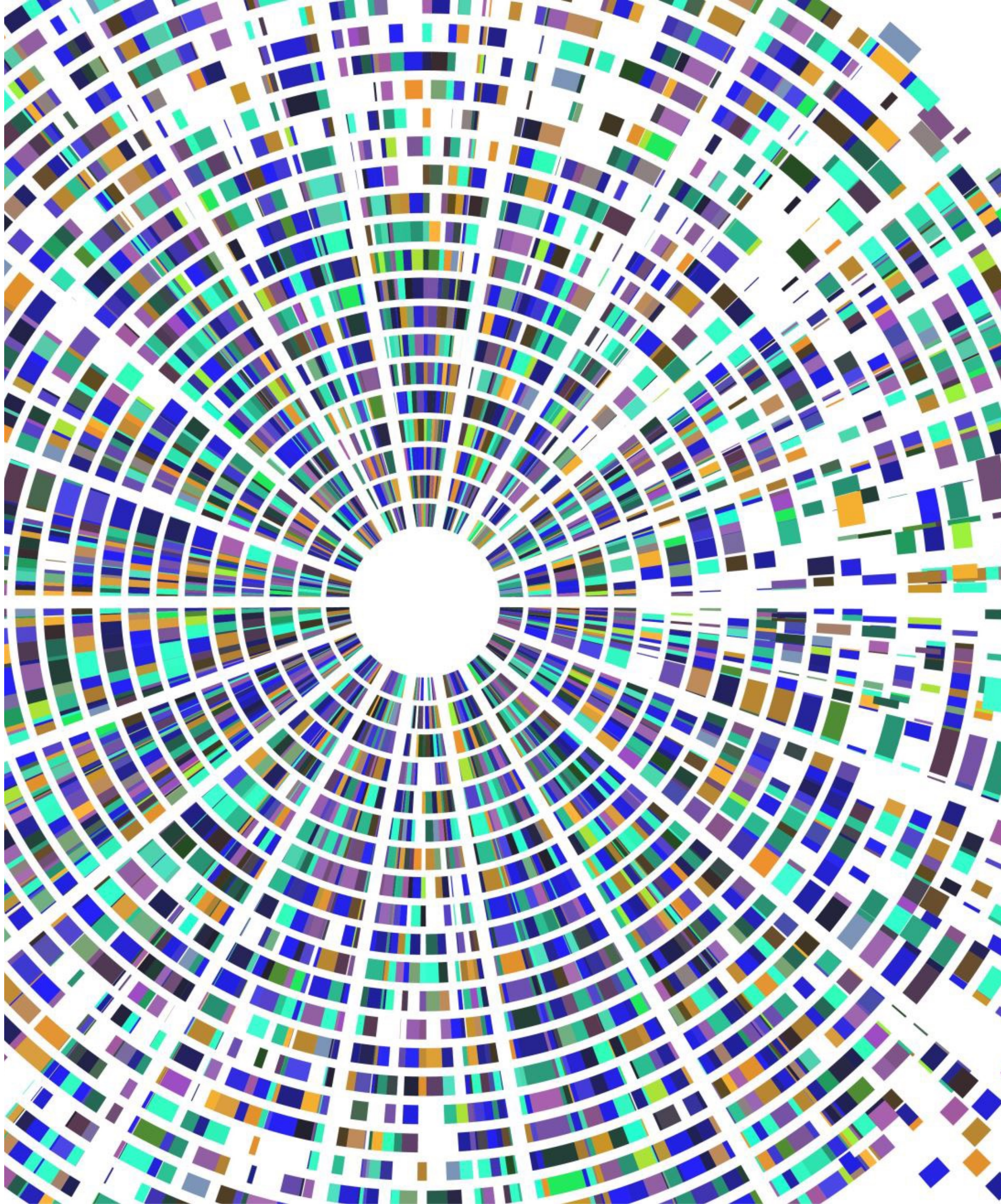
Audio features like danceability, loudness and valence of each track

Dependent variable is Popularity (0-100), with 100 being most popular

Popularity based on artists, genres, tracks and music features

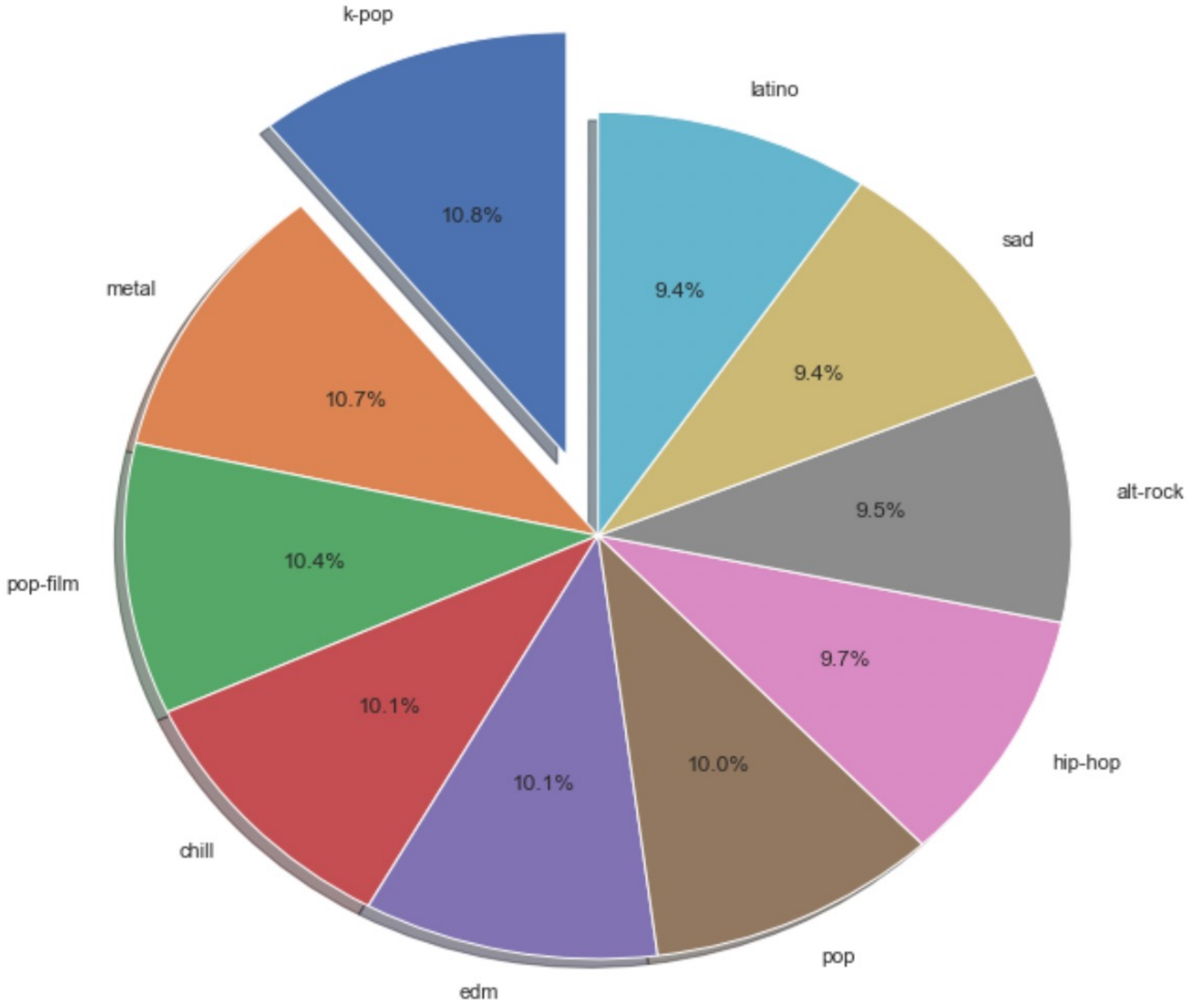
EXPLORATORY DATA VISUALIZATION (EDA)

- Matplotlib, Seaborn and Word cloud libraries



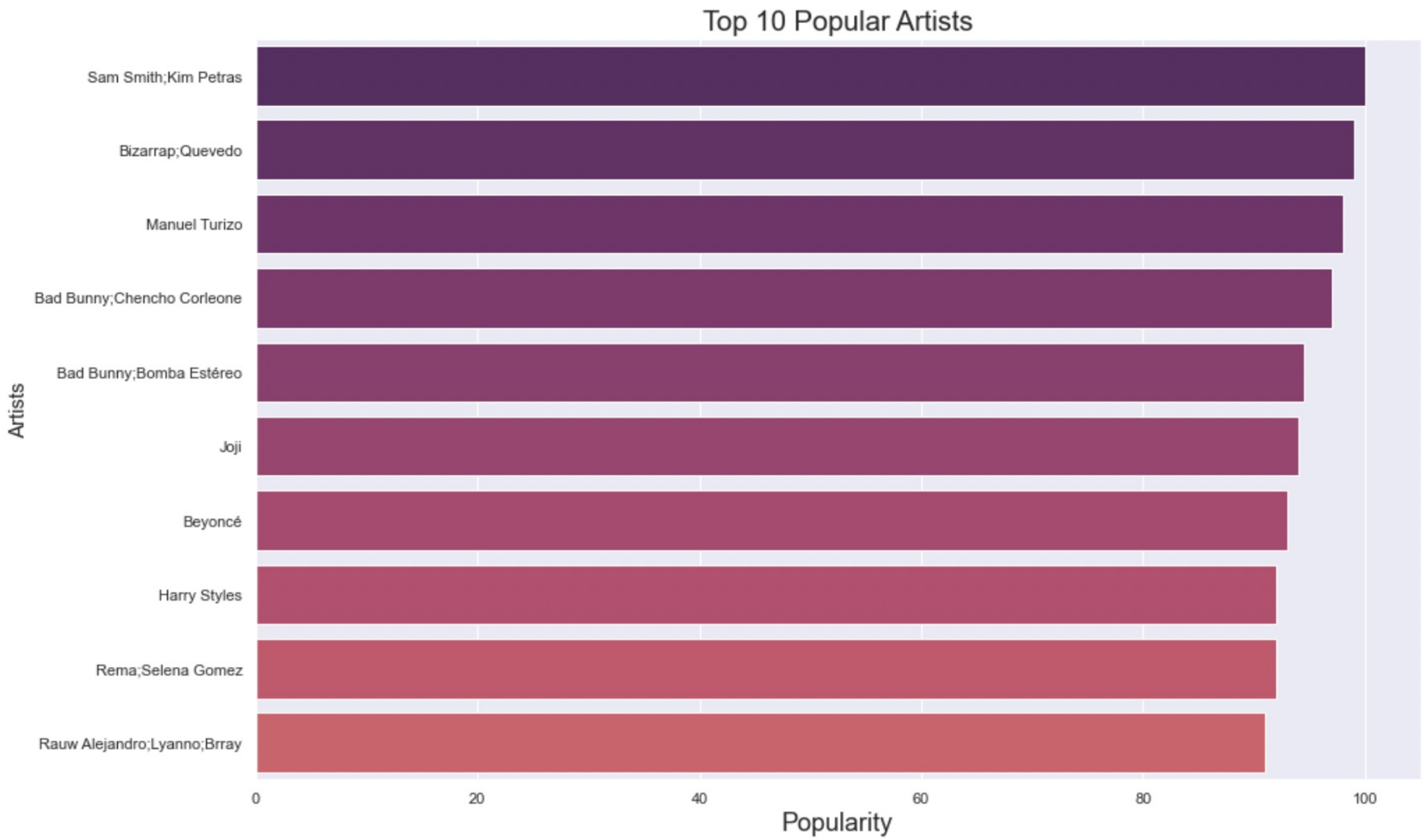
TOP 10 POPULAR GENRES

Genres	Popularity (0-100)
K-pop	59.093750
Metal	58.653595
Pop-film	56.725552
Chill	55.332790
Edm	55.148760
Pop	54.736508
Hip-hop	53.142549
Alt-rock	52.083333
Sad	51.618333
Latino	51.360248



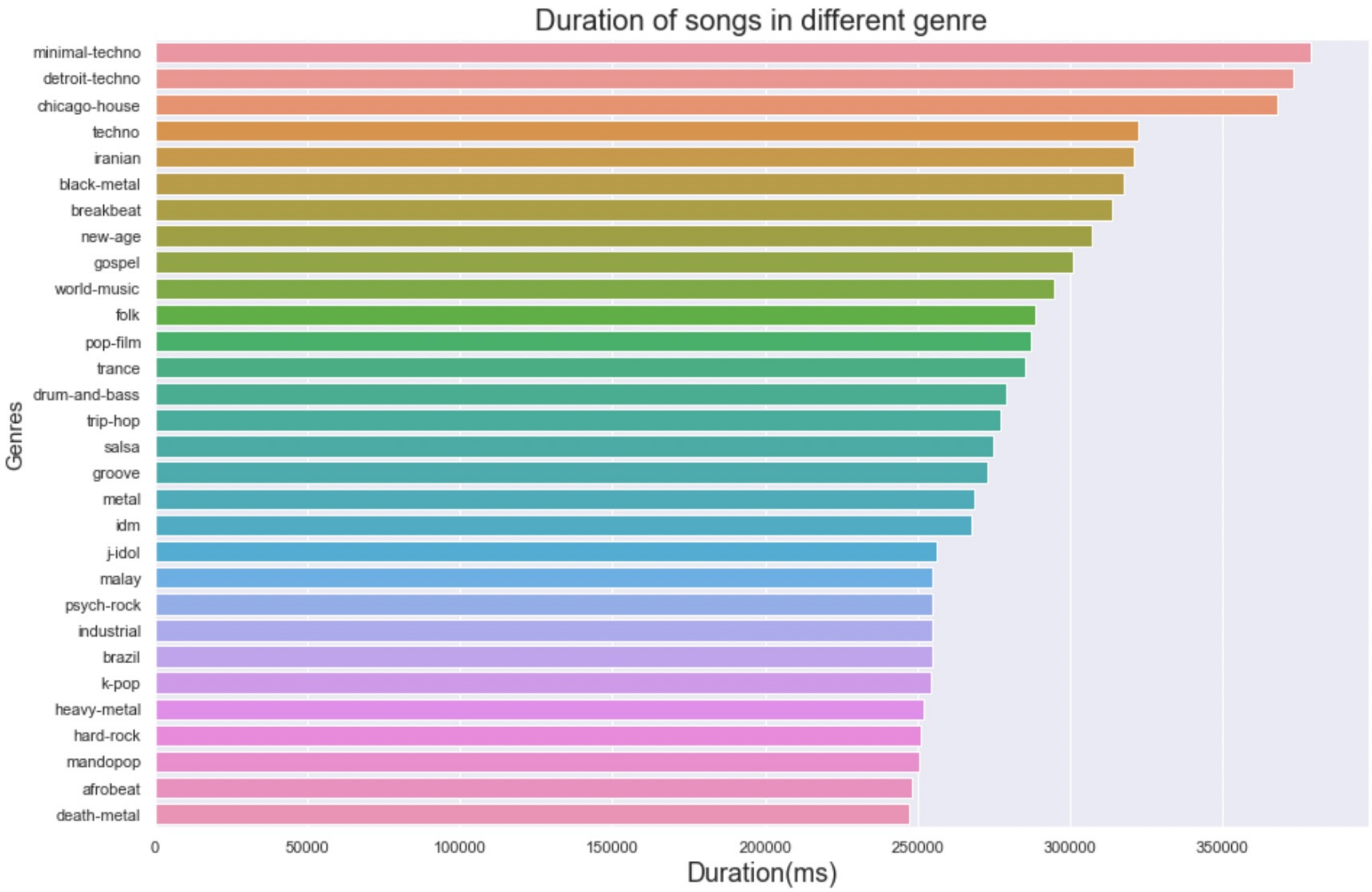
TOP 10 POPULAR ARTISTS

	artists	popularity
0	Sam Smith;Kim Petras	100.0
1	Bizarrap;Quevedo	99.0
2	Manuel Turizo	98.0
3	Bad Bunny;Chencho Corleone	97.0
4	Bad Bunny;Bomba Estéreo	94.5
5	Joji	94.0
6	Beyoncé	93.0
7	Harry Styles	92.0
8	Rema;Selena Gomez	92.0
9	Rauw Alejandro;Lyanno;Brray	91.0



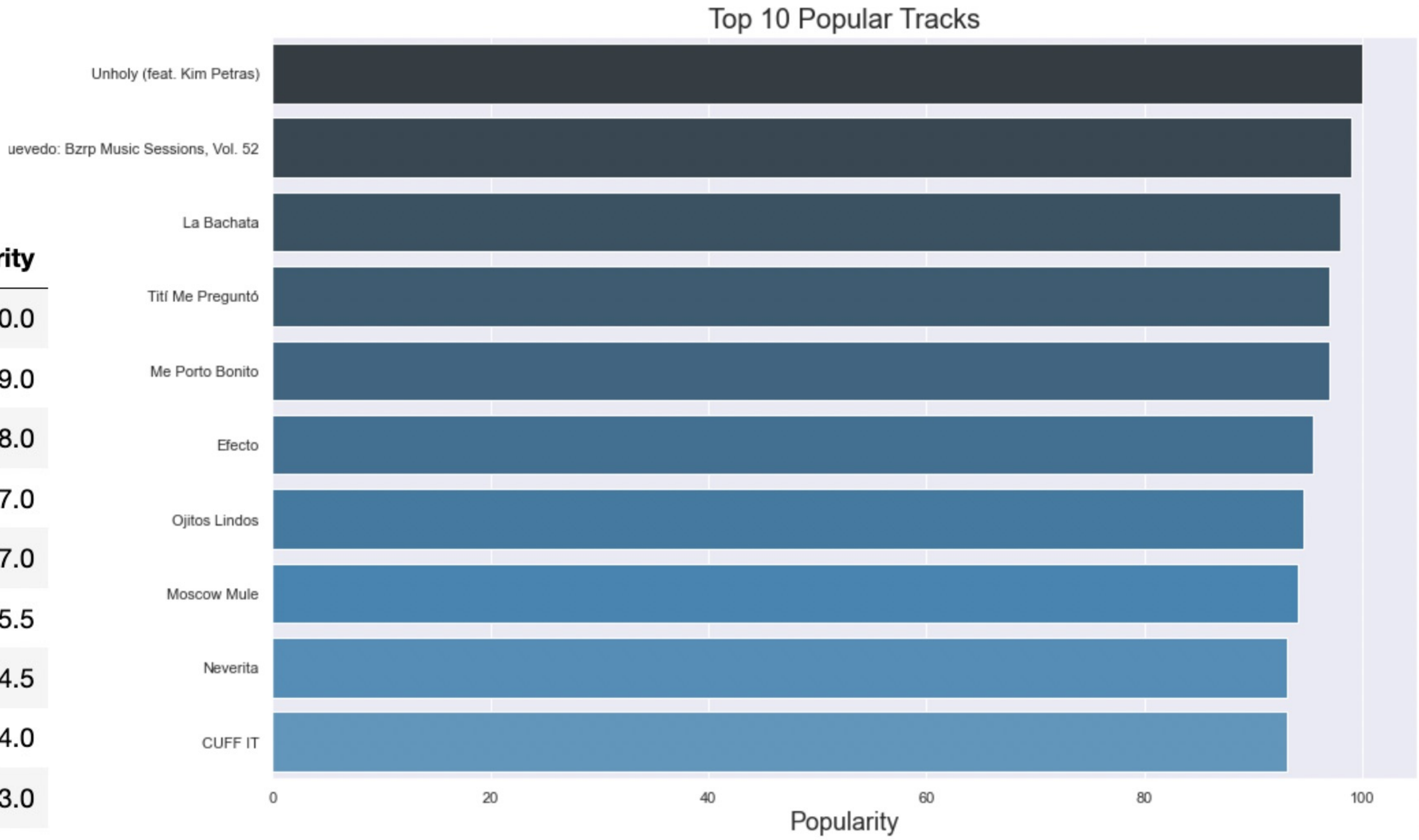
DURATION OF SONG IN DIFFERENT GENRES

	track_genre	duration_ms
0	minimal-techno	378792.150972
1	detroit-techno	373197.913043
2	chicago-house	367712.153602
3	techno	322377.637363
4	iranian	321035.072479
5	black-metal	317465.534937
6	breakbeat	313582.207120
7	new-age	306971.930876
8	gospel	301096.074250
9	world-music	294549.369048



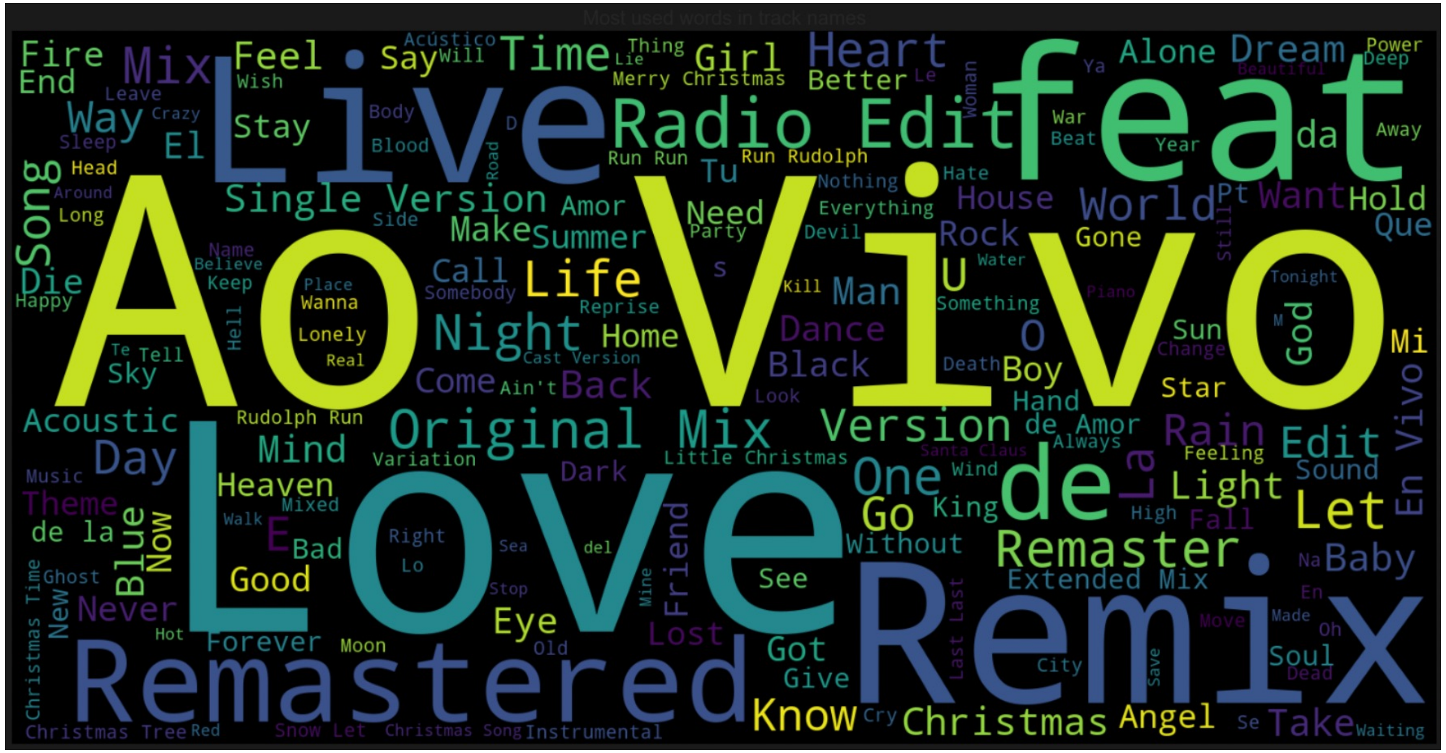
TOP 10 POPULAR TRACKS

	track_name	popularity
0	Unholy (feat. Kim Petras)	100.0
1	Quevedo: Bzrp Music Sessions, Vol. 52	99.0
2	La Bachata	98.0
3	Tití Me Preguntó	97.0
4	Me Porto Bonito	97.0
5	Efecto	95.5
6	Ojitos Lindos	94.5
7	Moscow Mule	94.0
8	Neverita	93.0
9	CUFF IT	93.0

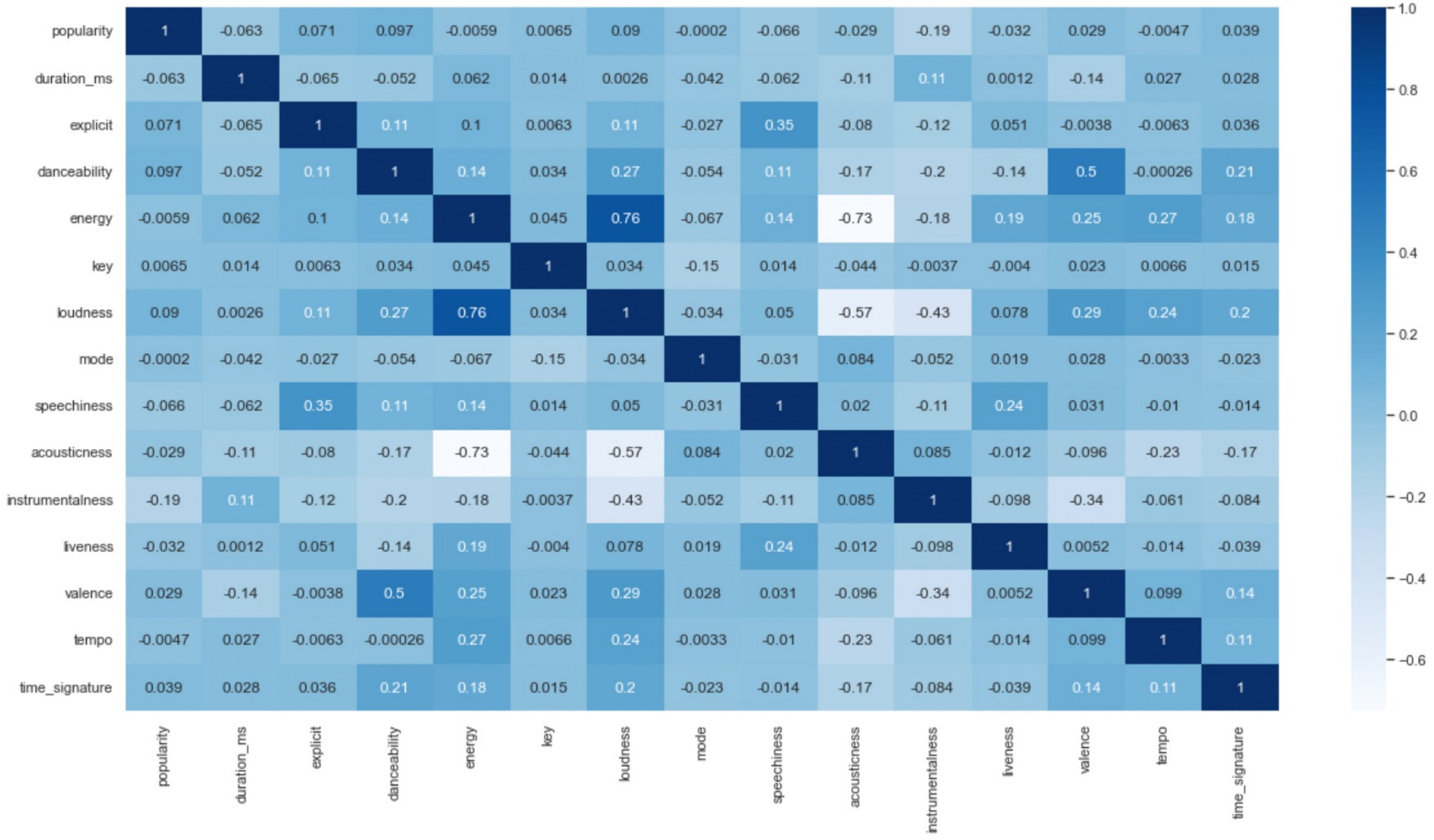


WORD CLOUD OF FREQUENTLY USED WORDS IN TRACK NAMES

1. Love
2. Live
3. Vivo
4. Ao
5. Remix
6. Remastered
7. Feat
8. Radio Edit



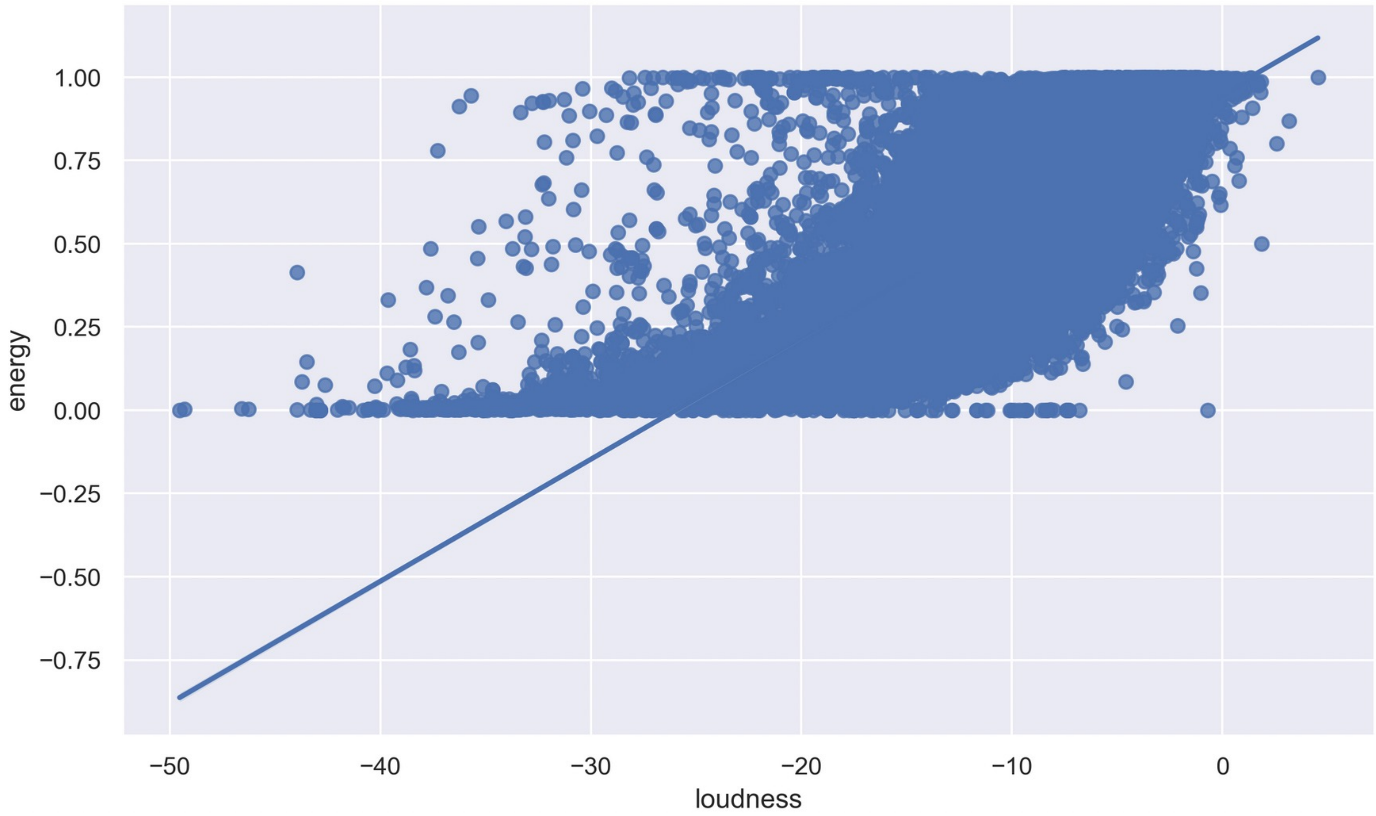
CORRELATION (HEATMAP)



- Energy and loudness are positively correlated with each other because it makes sense for loud music to be energetic. Similarly, valence is also positively correlated with danceability because valence represents songs which are happy and cheerful.
- Whereas, acousticness is negatively correlated with energy and loudness because acoustic songs are quiet and calm.

CORRELATION: LOUDNESS VS ENERGY

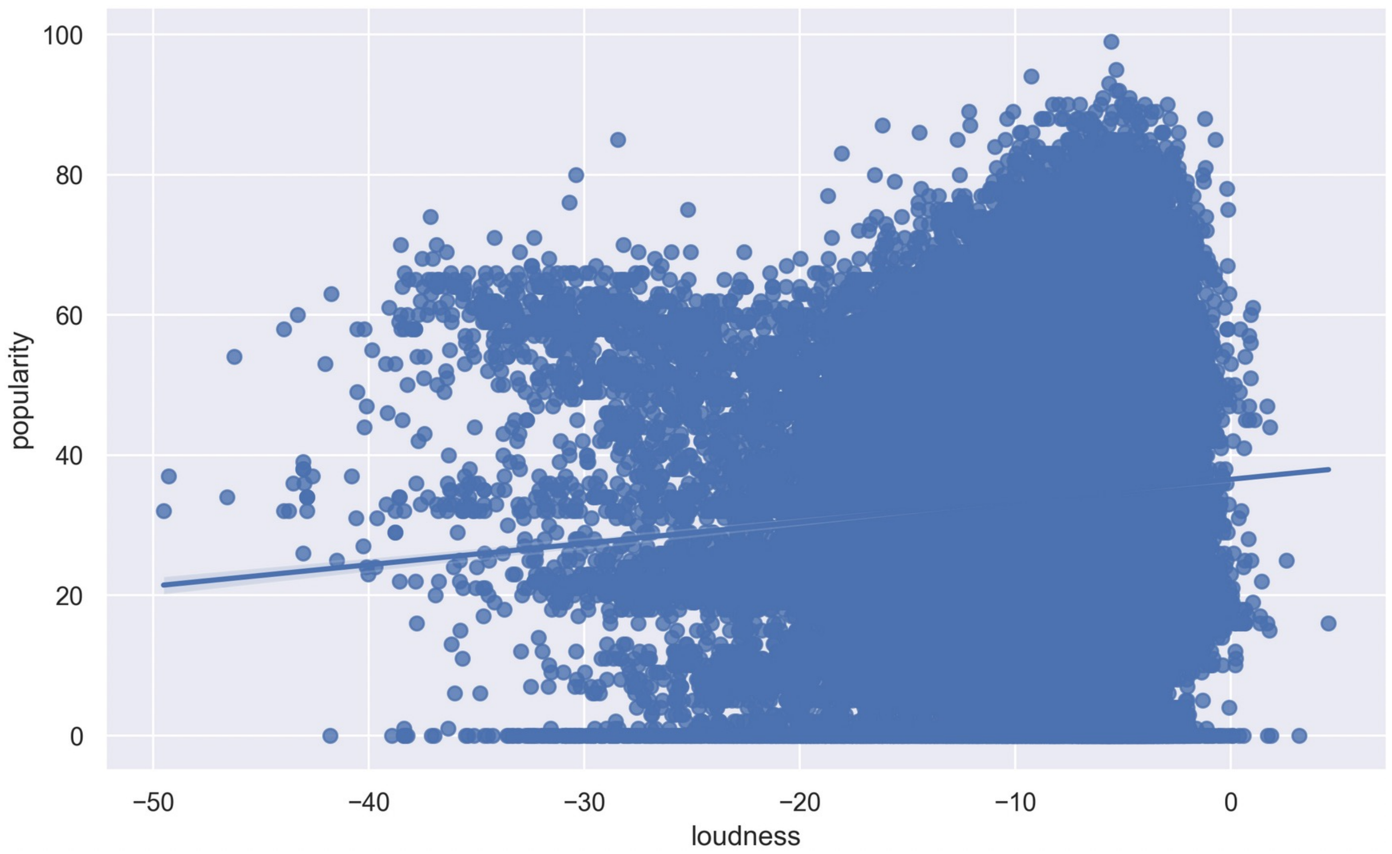
Loudness and energy correlated with each other at almost 45 degree which suggests that there is strong relation between a song being loud and energetic.



CORRELATION: LOUDNESS VS POPULARITY

With over 100,000 songs we can infer that there's slight correlation between the music being and loud and popular.

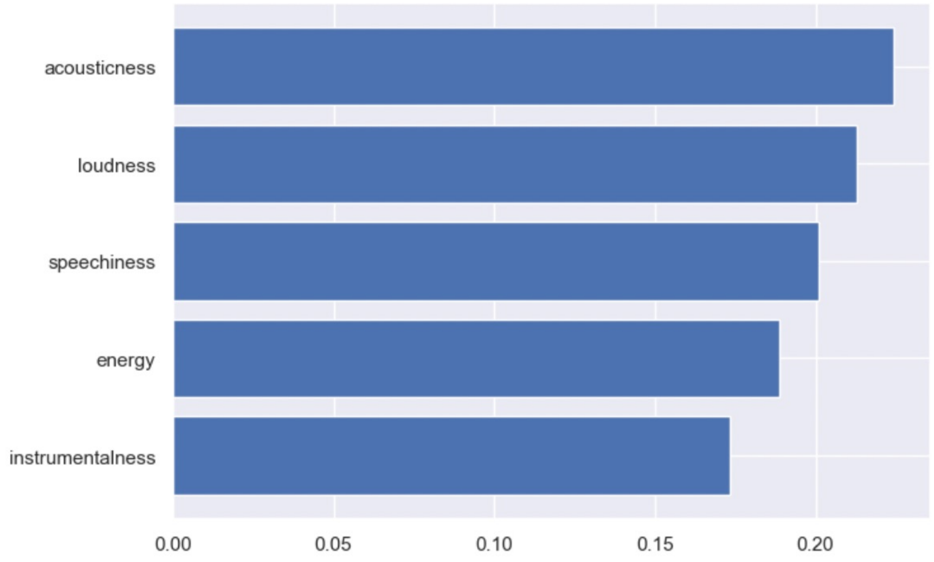
Most of the loud music are popular compared to the ones that are less loud.



FEATURE SELECTION

We tried different models and came up with the best score of important features from RandomForestRegressor.

```
In [638]: random_forest = RandomForestRegressor()

          random_forest.fit(x_train, y_train)
          Y_pred_rf = random_forest.predict(x_test)
          random_forest.score(x_train,y_train)
          acc_random_forest = round(random_forest.score(x_train,y_train) * 100, 2)

          print("Important features")
          pd.Series(random_forest.feature_importances_,x_train.columns).sort_values(ascending=True).plot.barh(width=0.8)
          print('__'*30)
          print(acc_random_forest)

Important features
____________________________________________________________
86.75
```



TRAIN AND TEST SETS

Modelling

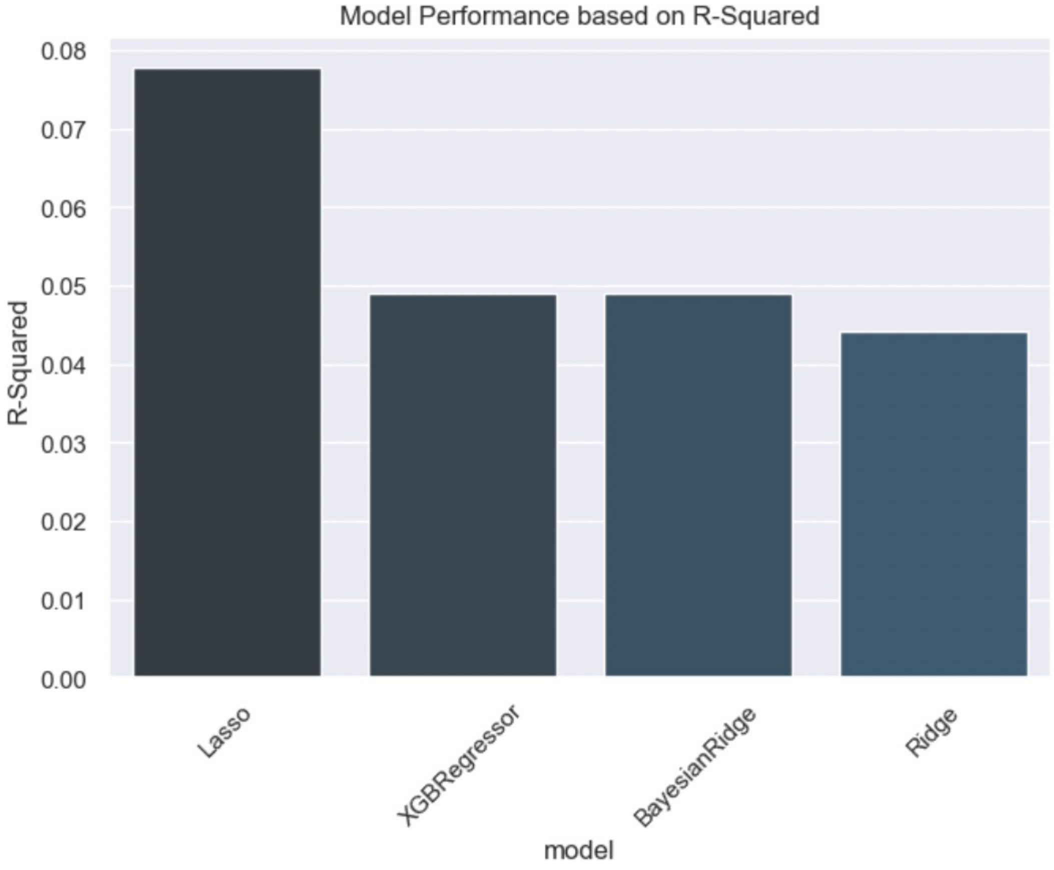
Split the Dataset into Training and Test Sets

```
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import train_test_split

X = df_final.drop(columns=['popularity'])
X = df_final[['loudness', 'acousticness', 'instrumentalness', 'energy', 'speechiness']]
y = df_final['popularity']

x_train,x_test,y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

print("Number of  train sample in train set:",x_train.shape)
print("Number of samples in validation set:",y_test.shape)
```

```
Number of  train sample in train set: (57060, 5)
Number of samples in validation set: (14265,)
```

MODEL OUTCOMES

We can see that even the well performing model, Lasso has relatively very low R-squared which suggests that even though some of the data points are correlated with popularity but can be problematic to precisely predict the value of popularity based on these correlated data points and from above R2 squared.



R SQUARE FROM DIFFERENT MODELS

	model	mean_squared_error	R-Squared	time
2	Lasso	310.32760	0.07780	18
0	XGBRegressor	320.01504	0.04901	0
4	BayesianRidge	320.01735	0.04900	0
1	Ridge	321.66368	0.04411	0

CLASSIFIER MODELING

Classifier Models

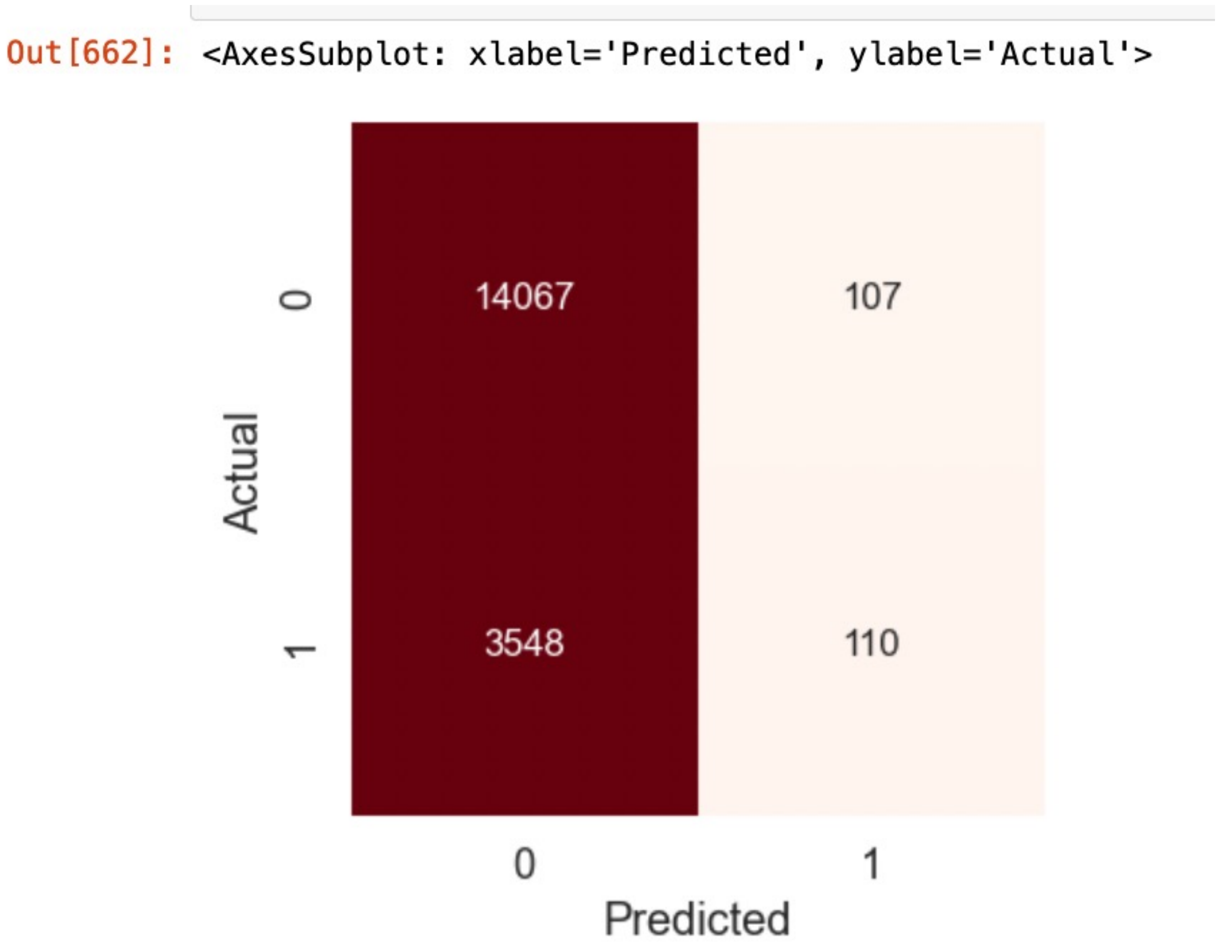
```
from sklearn.linear_model import LogisticRegression #Logistic Regression
from sklearn.naive_bayes import GaussianNB #Naive Bayes
from sklearn.tree import DecisionTreeClassifier #Decision Tree
from sklearn.neighbors import KNeighborsClassifier #KNN
from xgboost import XGBClassifier #XGB
from sklearn.preprocessing import LabelEncoder
from sklearn.ensemble import AdaBoostClassifier

from sklearn.model_selection import train_test_split

from statistics import mean
from sklearn.metrics import accuracy_score, log_loss
from sklearn.model_selection import KFold, cross_val_score

from sklearn.pipeline import Pipeline

df_final['is_popular'] = df['popularity'].apply(lambda x: 1 if x > 50 else 0)

y = df_final['is_popular']
X = df_final.drop(columns=['popularity', 'explicit', 'key', 'mode', 'time_signature', 'is_popular', 'duration_ms'])

x_train,x_test,y_train, y_test = train_test_split(X, y, test_size=0.25, random_state=42)
X
```

MODEL SCORE FROM DIFFERENT CLASSIFIERS

```
[0.79250397 0.79184971 0.79044771 0.7931389  0.7969714 ]
GaussianNB()
Model Score: 79.298

------------------------------------------------------------
[0.75502383 0.7514721  0.7553977  0.75902038 0.75948775]
KNeighborsClassifier()
Model Score: 75.608

------------------------------------------------------------
[0.79577531 0.79549491 0.79306477 0.79781268 0.79958871]
DecisionTreeClassifier(max_depth=5)
Model Score: 79.637

------------------------------------------------------------
[0.79586877 0.79624264 0.79156931 0.79575622 0.80052346]
RandomForestClassifier(max_depth=5, max_leaf_nodes=8, n_estimators=500)
Model Score: 79.599

------------------------------------------------------------
[0.79493411 0.79437331 0.79222357 0.79547579 0.79865395]
AdaBoostClassifier()
Model Score: 79.513

------------------------------------------------------------
```

CONFUSION MATRIX

- True negative and false negative is quite high but we got true negative on the higher side.
- True positive and false positive although there are less occurrences, confusion matrix is showing the false negative on the higher side.



OUT OF SAMPLE PREDICTION

From the analysis, it seems that for a song to be popular it is quite hard and not every feature of song can be easily integrated to make a song popular.

	danceability	energy	loudness	speechiness	acousticness	instrumentalness	liveness	valence	tempo
0	0.838	0.8590	-4.734	3939.4002	0.510	0.900	0.117	0.120	189.12
1	0.733	0.8575	-8.318	0.4010	0.930	0.001	0.328	0.093	119.94
2	0.876	0.6544	-1.888	0.2970	0.740	0.828	0.383	0.334	79.19
3	0.123	0.7484	-6.444	0.0720	0.445	0.974	0.873	0.394	135.96

```
Predicted value for popularity :  0 , which means  "no"
Predicted probability is 0.904
Predicted value for popularity :  0 , which means  "no"
Predicted probability is 0.907
Predicted value for popularity :  0 , which means  "no"
Predicted probability is 0.904
Predicted value for popularity :  0 , which means  "no"
Predicted probability is 0.904
```

LIMITATIONS

This dataset did not have date variable which restricted us from making analysis based on time-series, about how music is evolving with time.

Availability of many features made it quite difficult to choose right set of features.

Since this dataset has a lot of rows it was at times very time consuming to run different models.

LEARNING, SUMMARY AND PREDICTION

- The popularity of a song is influenced by the **danceability**, **loudness** and **valence**.
- The factors that determine the song's genre are **danceability**, **energy** and **valence**.
- **K-pop** music is most popular nowadays.
- **Minimal-techno** genre has the longest track duration.
- Sam Smith's (ft. Kim Petras) **Unholy** is the most popular track.
- High energy dance songs and songs with duration of approximately 3 minutes are more likely to become popular.
- Although our models were not with good scores and highly significant in terms of different metrics, but we learned that with every dataset comes a challenge to build better models and improve scores by applying different tuning to models.

Thank you