Machine Learning

Course-End Project: Creating Cohorts of Songs

Col Rakesh Pedram: Nov 2023 cohort

Approach

This project builds on the skill sets honed in the previous modules and includes machine learning. The data set was examined and data wrangling tasks carried to check if data was clean and was ready to process in the algorithms. Fields like dancebility, livemess ets are in range (0,1) with the popularity scaled from 0 to 100.

Skills of linear regression, correlation and clustering were used in iterative manner to find a model that accurately predicts popularity and makes a prediction.

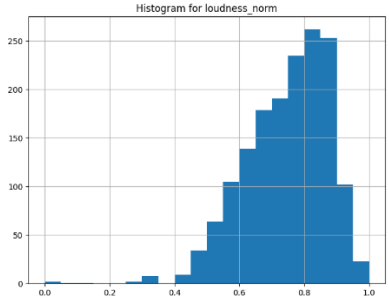
It was extremely difficult to make correlations and lots of domain expertise is required for the same. The model made is not perfect and can be improved but I went through the process of generating different models and checking their efficacy in prediction using train and test methodology.

Steps to execute

I have done my proj in google colab importing my data set from a csv file on google drive. Inspection and cleaning was carried out using descriptive analysis and means outliers for every field were identified.

Exploratory data Analysis

The Histograms and box plot was generated to examine data



The unique values in categorical data were identified.

```
album      90
name      954
dtype: int64
```

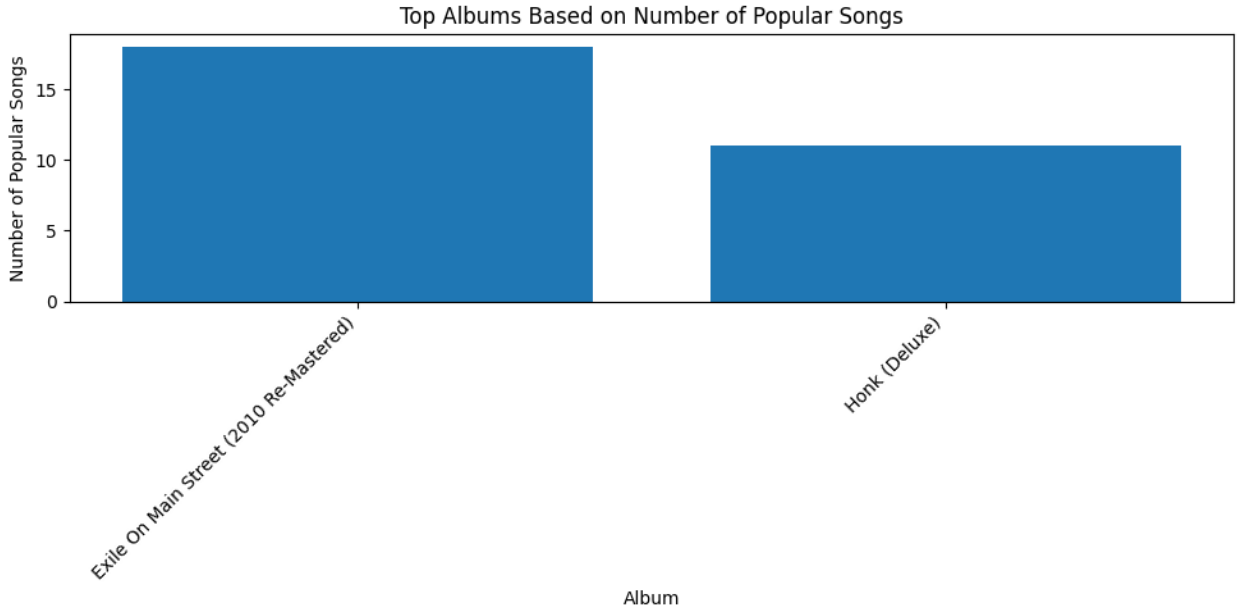
The values were very high to consider one hot encoding of the data.

Normalisation

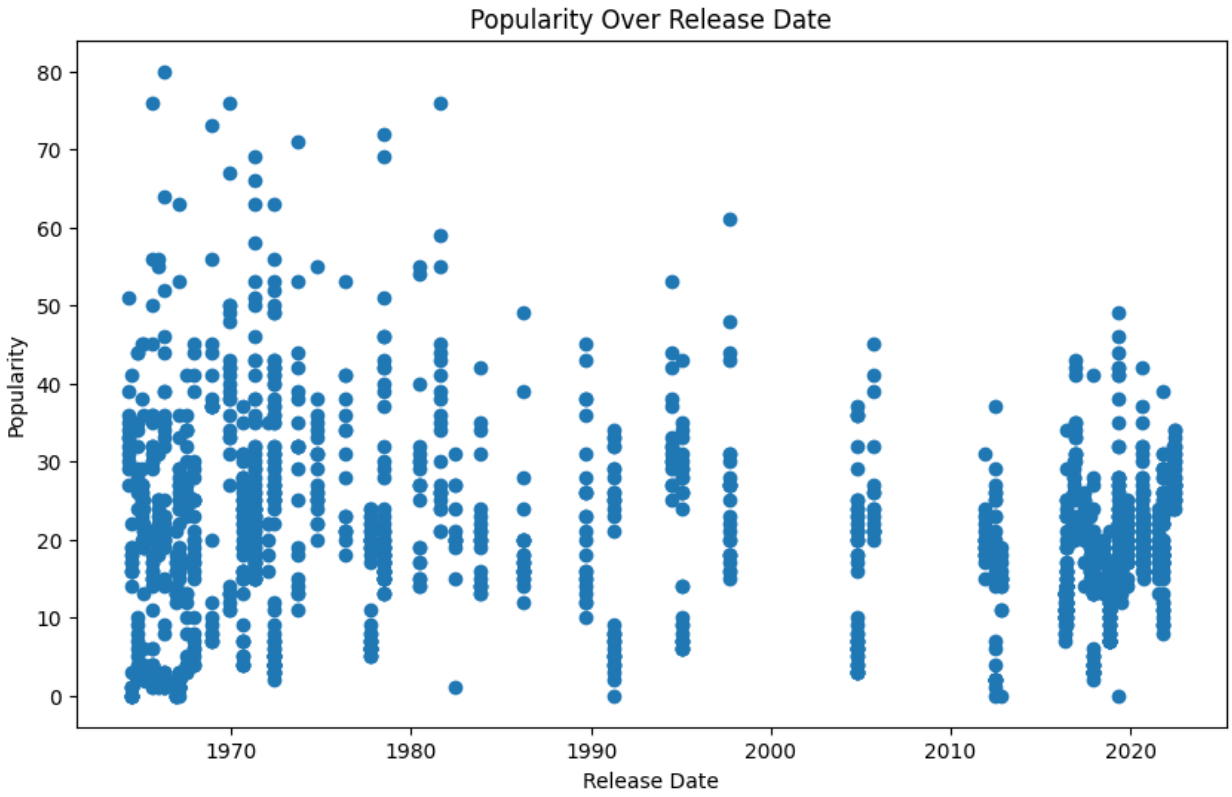
- The fields loudness, popularity, tempo and duration_ms were all normalized to prevent bias of magnitude on the fields selected.
- The cleaned df was then merged and saved in a separate csv to permit loading without going through the whole drata wrangling process.
- The data field was also converted into int value and normalised

Finding the two albums to recom based on popular songs.

The popularity field has a 75 percentile value of 27. Hence I fixed my popularity threshold as 30. Based on this I den two albums to recommend.



Made a scatter plot of rel date vs popularity



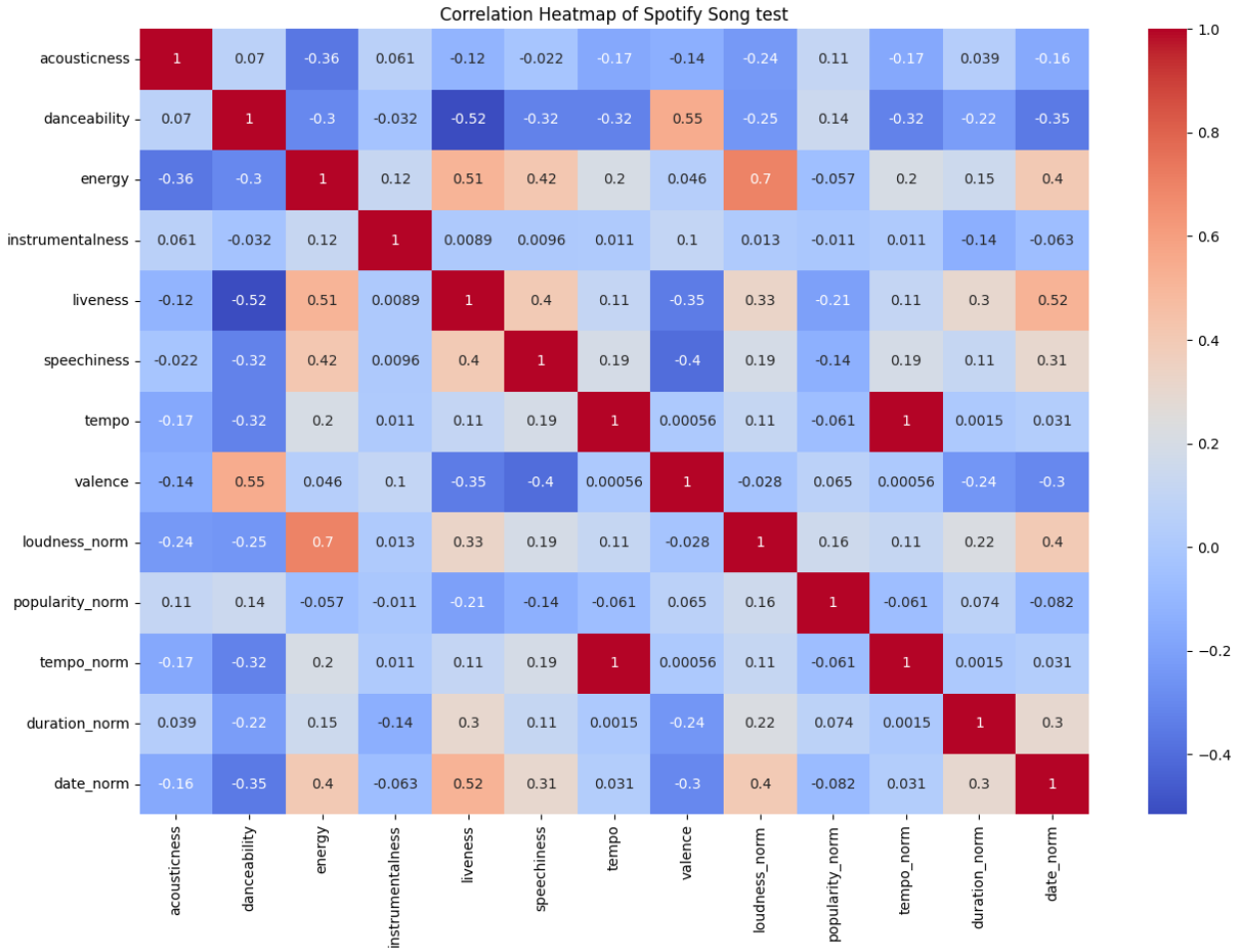
This plot shows reducing popularity score with passage of time.

Deep Dive into Data

There after the characteristics of the songs in these recommendation were eyeballed to iden pattern and data analysis done to generate correlation

Correlation

Correlation matrix was generated with seaborn

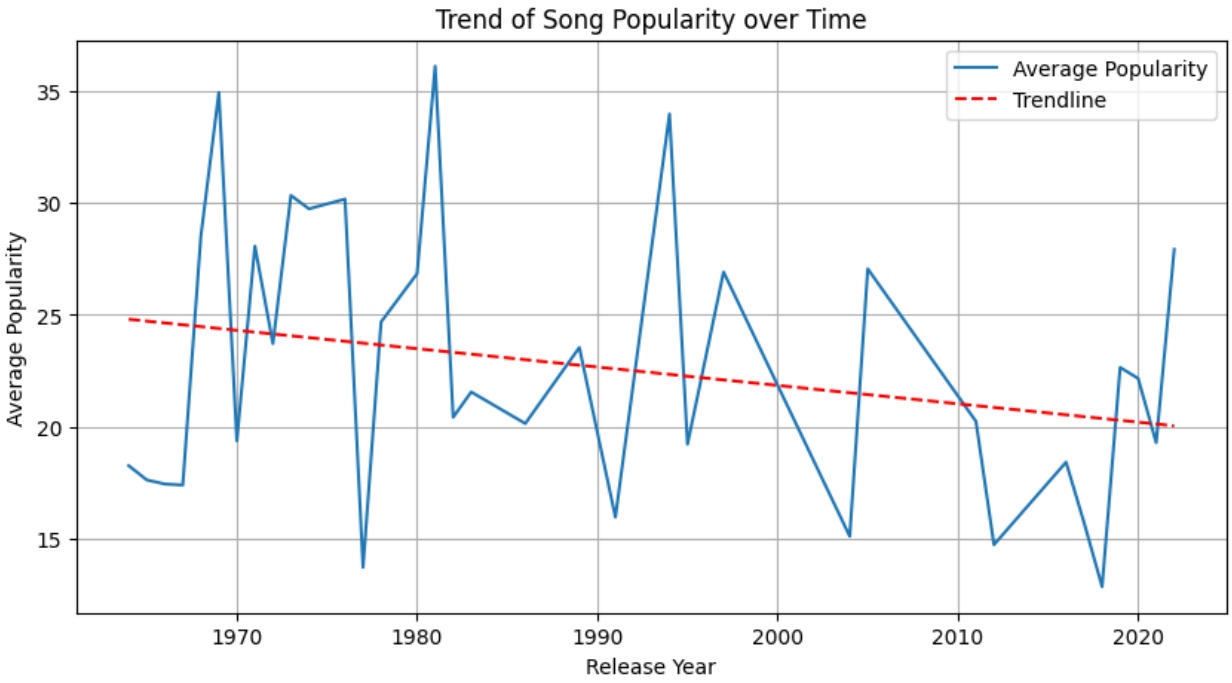


This gave a a broad view in identifying variables that have a correlation with normalized popularity.

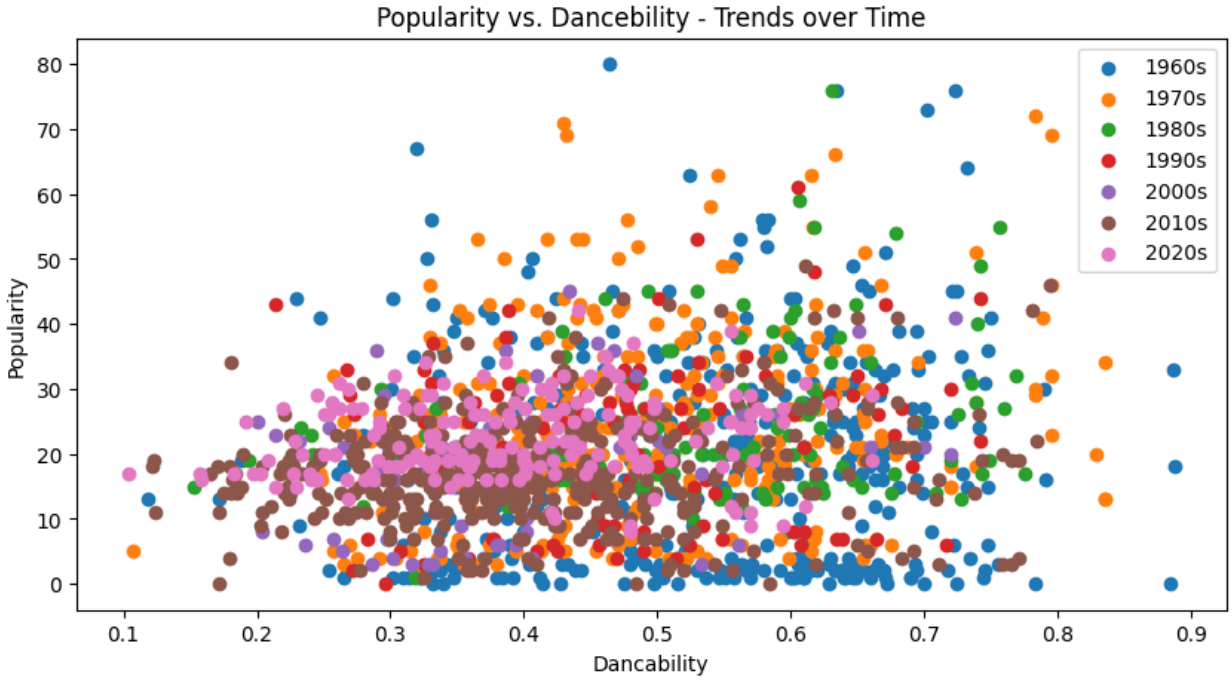
- I have identified dancebility, liveness,, loudness_norm and speechinees as likely candidates for further study on the model to predict popularity.

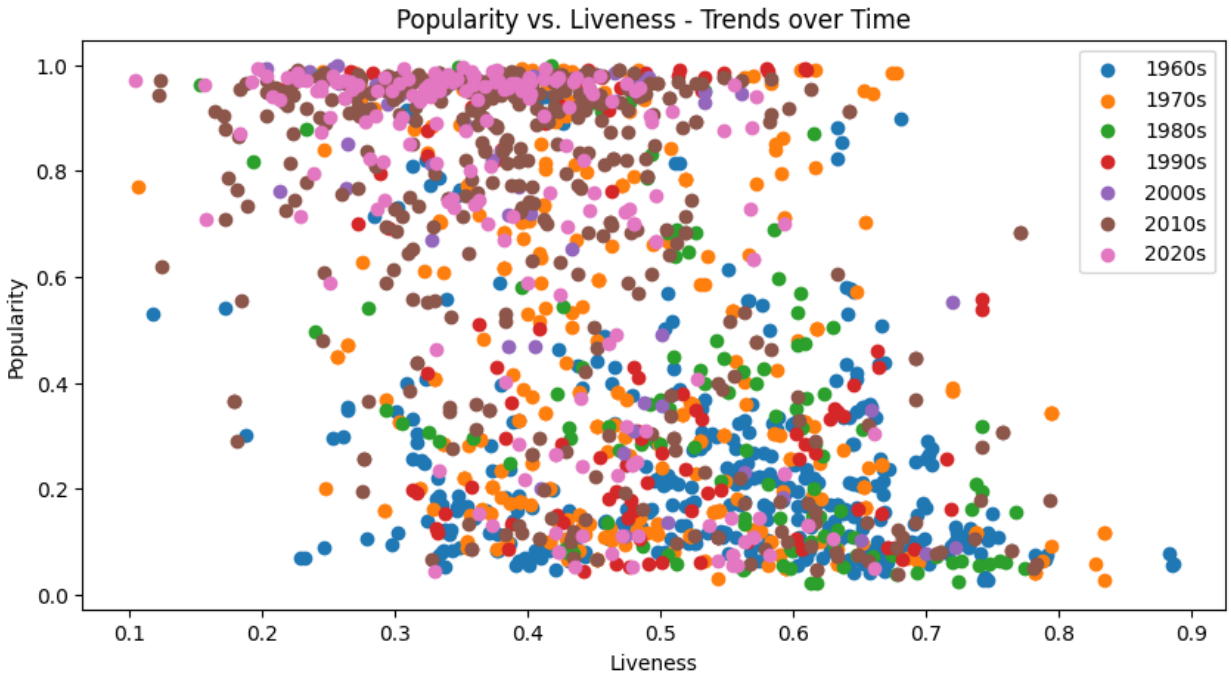
Popularity dependence over time

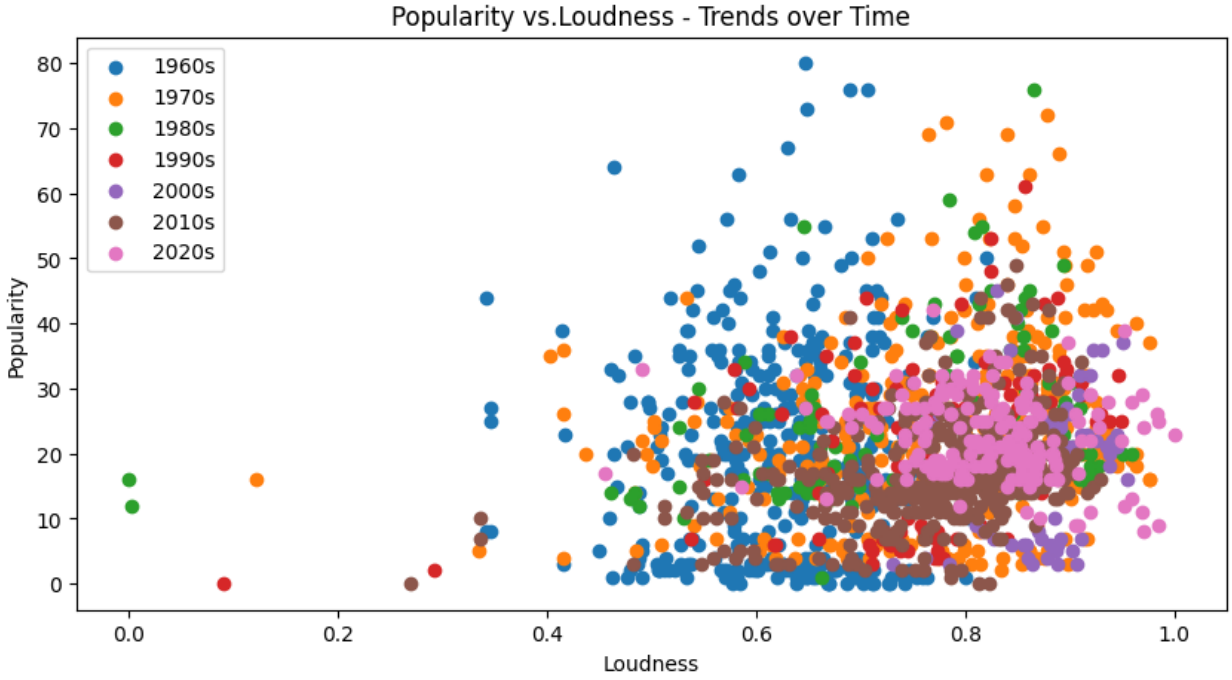
I plotted a chart how popularity means over the year changed over the time. This was very logical and showed a negative trend. Trendline was plotted over the grapg using numpy.polifit library.



Different features ere plotted over time with a scatter plot of featyre Vs popularity made over various decade blocks







Linear regression

A linear regression algo was run on the following parameters 'liveness','danceability','loudness_norm', 'speechiness','acousticness' with dependent variable poularity_norm.

The model was run by dividing the dataset in a 50:20 split for training and testing.
The results were

```
Intercept: -0.010078236326016843
Coefficients: [-0.09693499  0.07493575  0.36851932 -0.24004502
0.10126966]
```

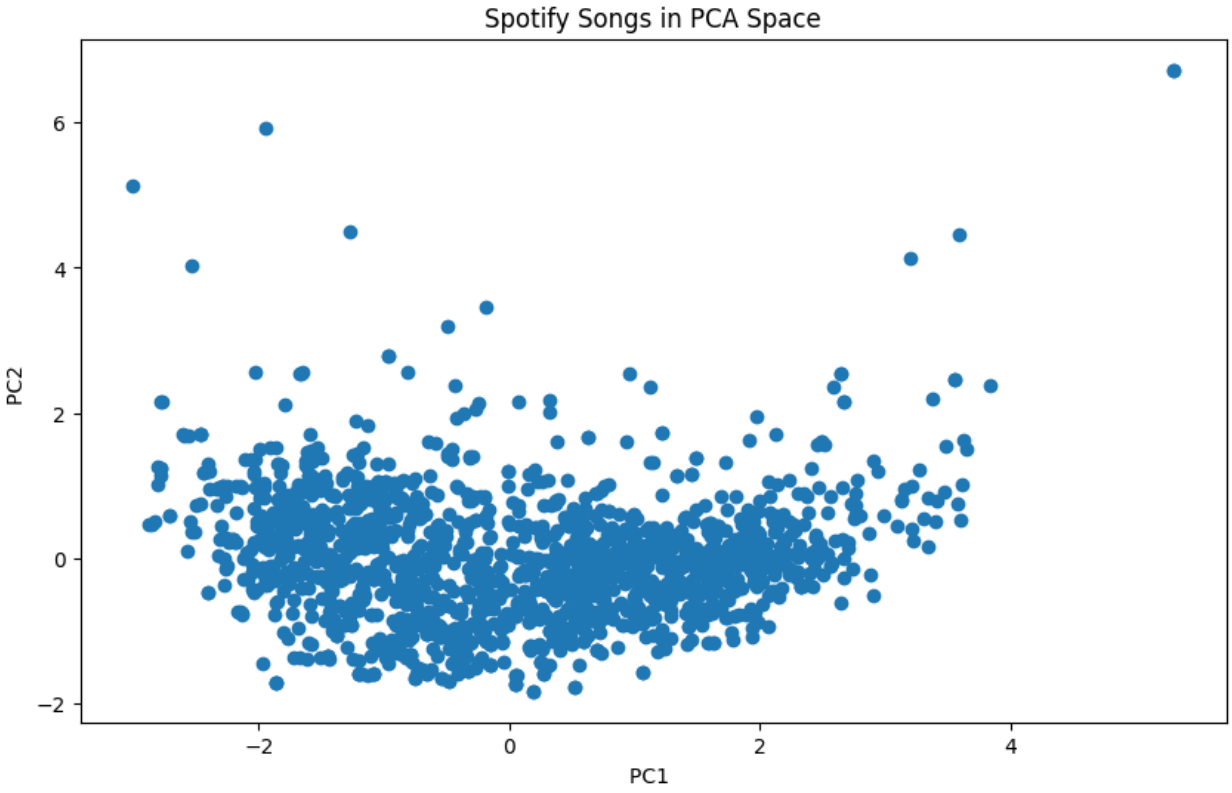
The evaluation of the prediction on the test data set is as listed:
R-squared: 0.13463616860962335
Mean Squared Error: 0.01979423053111896
The R square value were not impressive and I tried many other features but this is the best fit I could get.

Principle component analysis

For reducing the dimensionality of the data, PCA analysis was carried out. The more the dimensionality increases the model gets complex and overfits. Hence a PCA was carried out on the same factors as iden in linear regression.

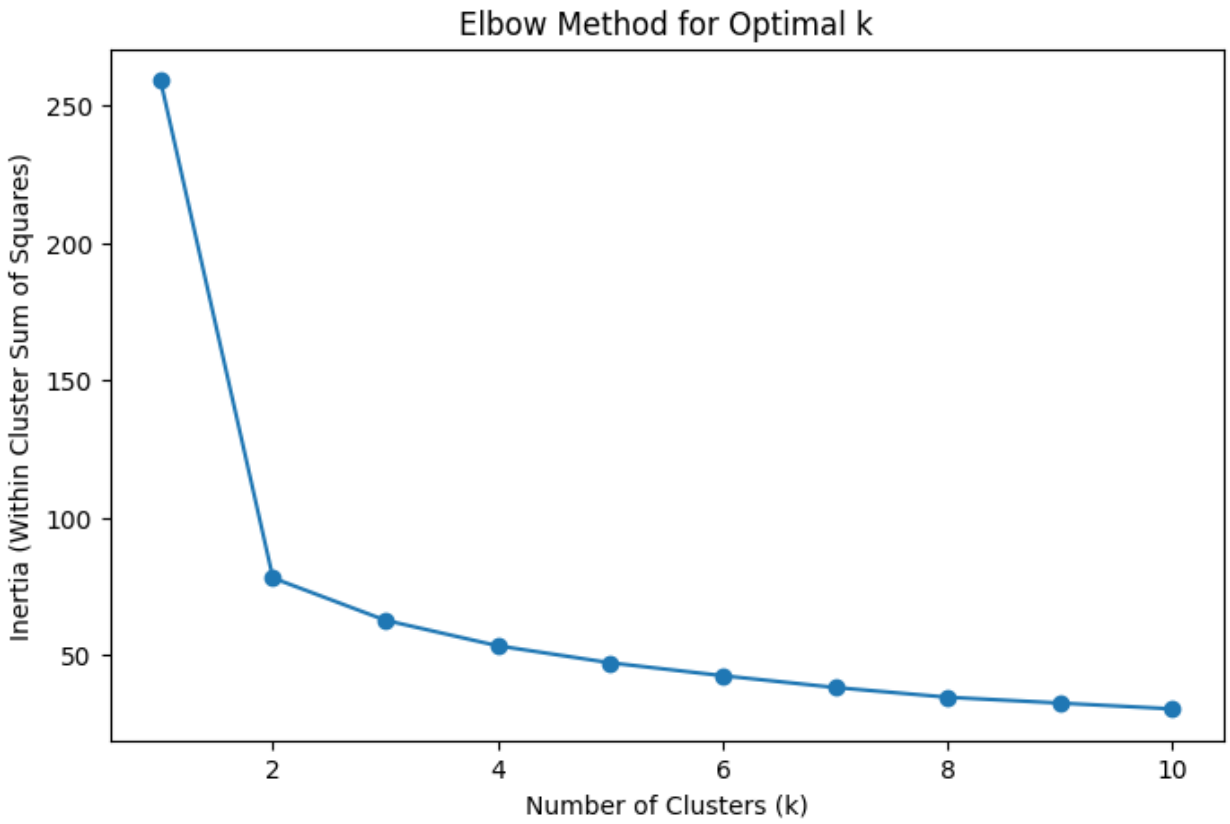
The loading of the model is as listed:-
0.50677948 0.20583035]
[[0.46747945 0.46188434]
 [-0.53501817 -0.16676239]
 [0.40056891 -0.86801114]
 [0.57858712 0.07355147]]

Each feature had a strong contribution to both the PC1 and PC2 component of the analysis.
The scatter plot is below



Clustering

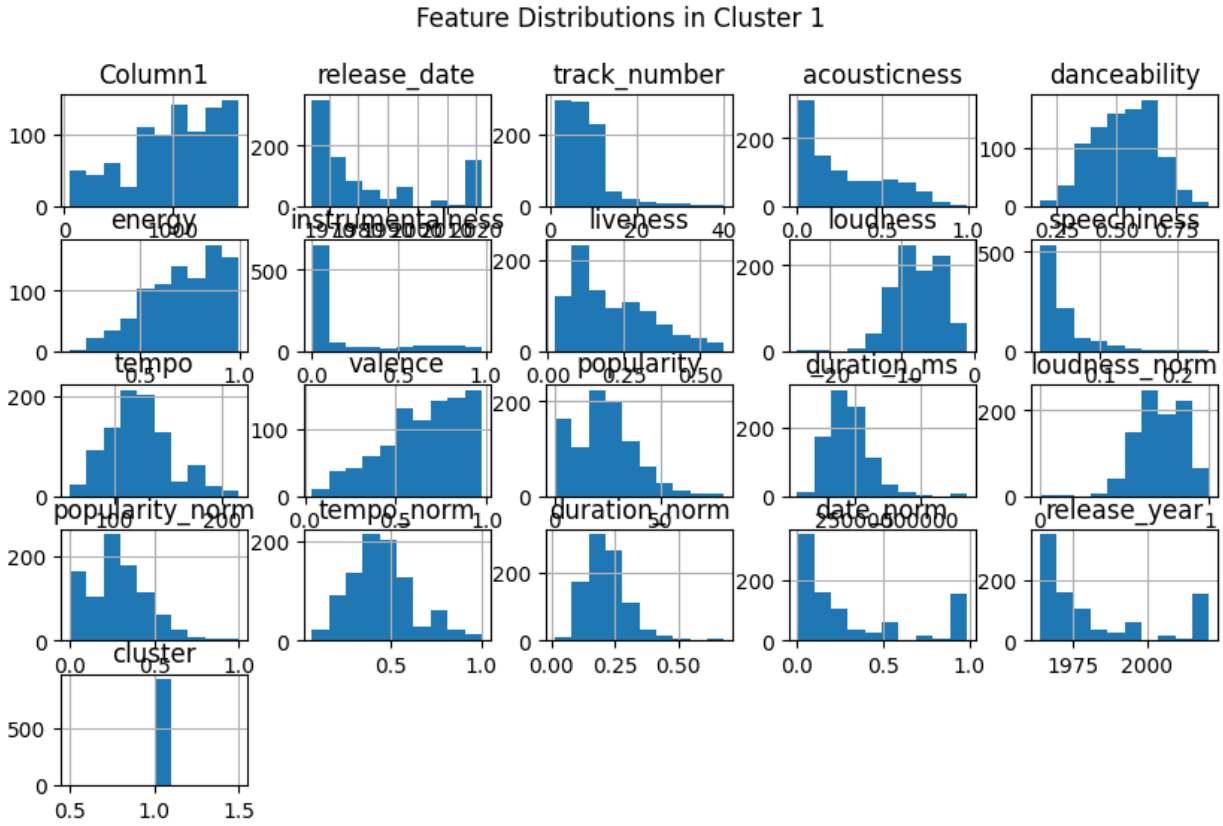
Kmeans algorithm was then used for creating the clusters. The number of clusters were first identified by using the elbow method.



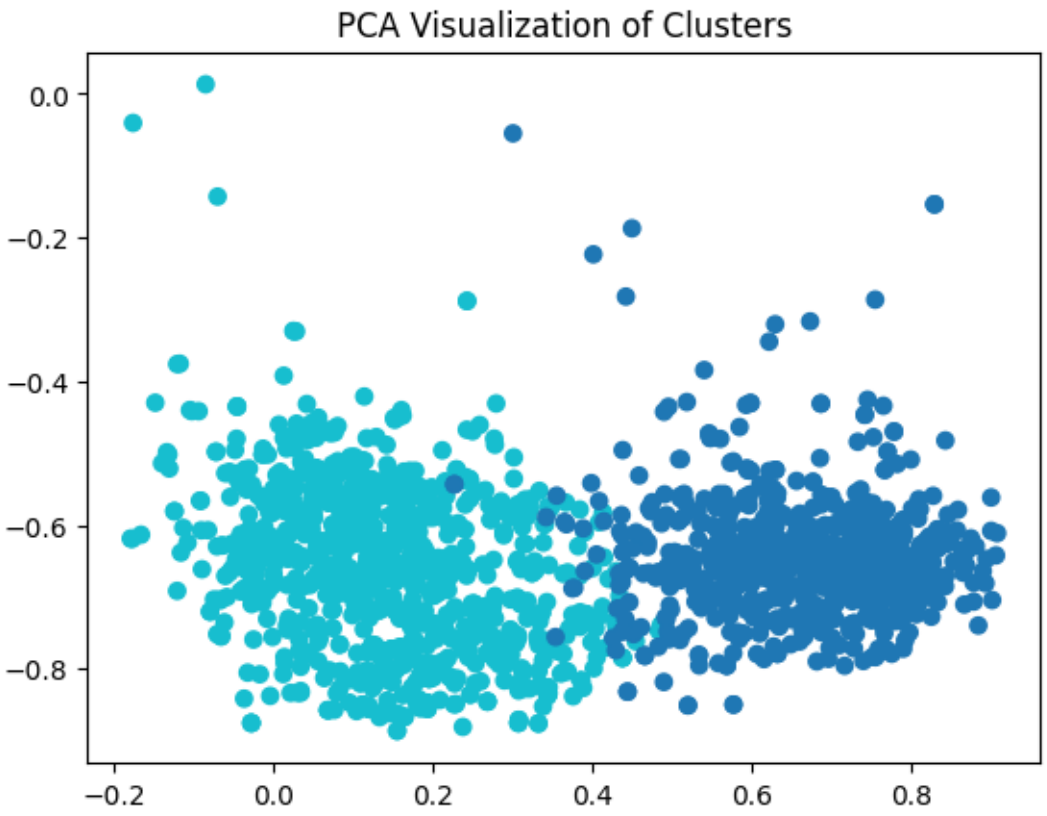
N_cluster was set to 2.
And Kmean function applied in the matrix.

Feature statistics was plotted for each cluster.

Finally the clusters were plotted for visualization



The Cluster visualization in mathplot is below



Conclusion

This was a tough exercise in data analysis. The challenges of modelling from alive data set were understood. Domain knowledge of the field is very important in identifying features and the analysis can be subjective.

Nice challenging project learnt a lot.