Music Recommendation System

Interim Project Presentation

jasdeep19047@iiitd.ac.in
siddharth19277@iiitd.ac.in
shanu19104@iiitd.ac.in

INDRAPRASTHA INSTITUTE *of*
INFORMATION TECHNOLOGY **DELHI**

Problem Statement -

In the present day scenario, with around 79 million songs present on the internet, people may find it difficult to listen to the songs of their taste. With algorithms working on metadata of songs and with no audio signals analysed, the user may not experience the best experience.

Dataset -

Obtained GTZAN dataset from Kaggle (click on this to get link)

Further we listened to several songs and effectively added 2 more genres consisting of 100 wav files each. Then we shortened the wav files and extracted length of 30 sec starting from 30 sec for each audio file.

Command ➜ **for %i in (*.mp3) do ffmpeg -ss 30 -t 30 -i "%i" "%~ni.wav"**

Contribution of each Team Member

Jasdeep Singh

- Data Exploration and Collection (100 files for Electric genre)
- Feature extraction, Feature manipulation
- Pre-processing and Data Visualization
- Logistic Regression
- KNN, ANN

Shanu

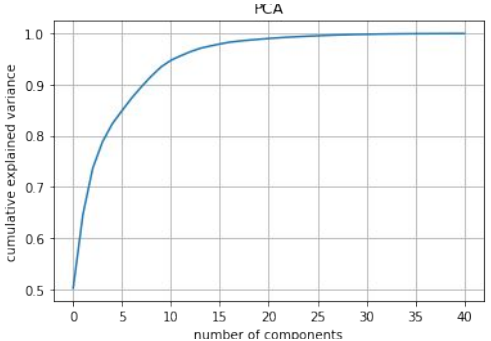
- Data Exploration and Collection (50 files for K-POP genre)
- Feature extraction, Feature manipulation
- Pre-processing and Data Visualization
- Clustering - KMean, Birch, DBSCAN, Gaussian Mixture Model

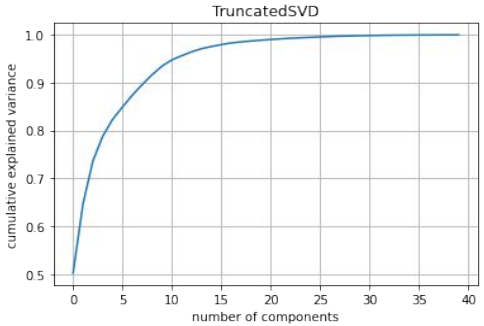
Siddharth Singh Kiryal

- Data Exploration and Collection (50 files for K-POP genre)
- Wavelet and spectrogram image dataset generation
- Data Visualization
- CNN

EDA, Preprocessing and Feature Extraction

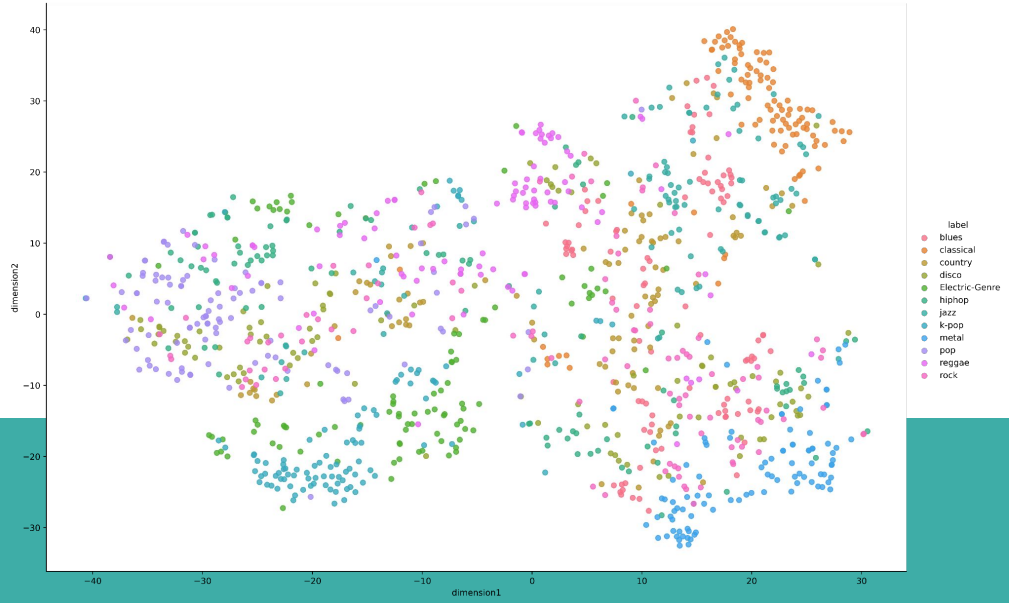
- Used **librosa** library to analyse audio signals and extract features.
- Extracted Features – rms, chroma_stft, spectral_centroid, spectral_bandwidth, spectral_rolloff, zero_crossing_rate, tempo, spectral_contrast, spectral_flatness, flux, harmony, mel_spec.
- As data extracted from each audio feature was very large to process on. So we applied pre-processing in which we stored mean and variance values for feature data.
- Since our output file have 43 features so we applied different feature reduction techniques **PCA** (plotted), **SelectKBest** and **Truncated SVD** to reduce the number of redundant features.



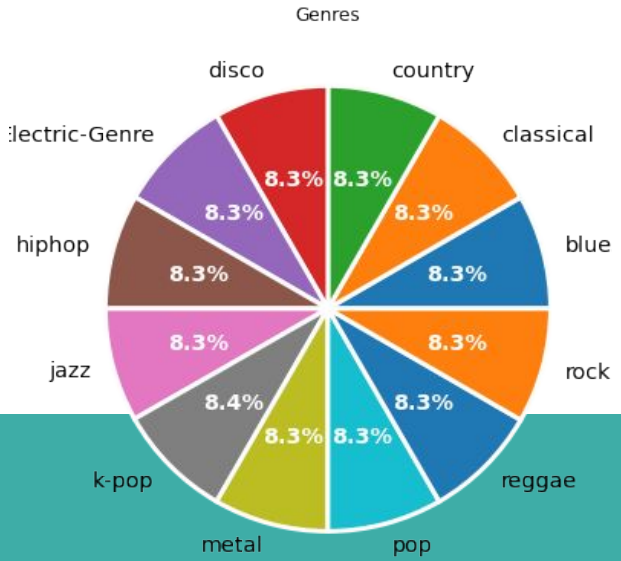


- **Standardization** scales each input variable separately by subtracting the mean (called centering) and dividing by the standard deviation to shift the distribution to have a mean of zero and a standard deviation of one.
- Dataset does not have any missing, NaN, noisy or inconsistent value so there was no need for any kind of Data-cleaning.
- In CNN, the image data is resized to 256 x 256 x 3 (colored), with each genre provided a class number, which is followed by normalization of training/testing sets and several image variables are set before fitting the model.

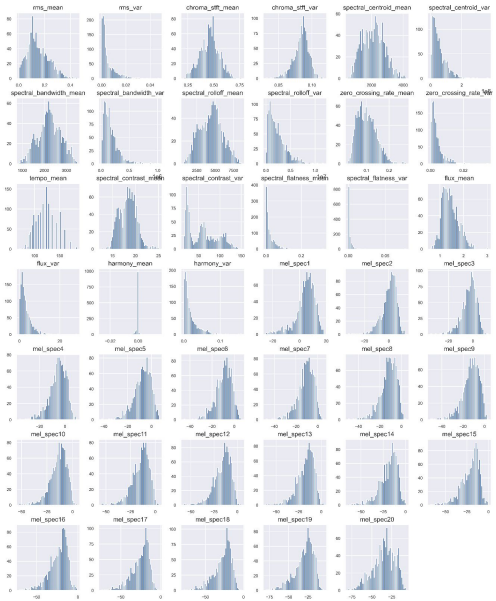
Data Plotting with `TSNE(n_components=2)`



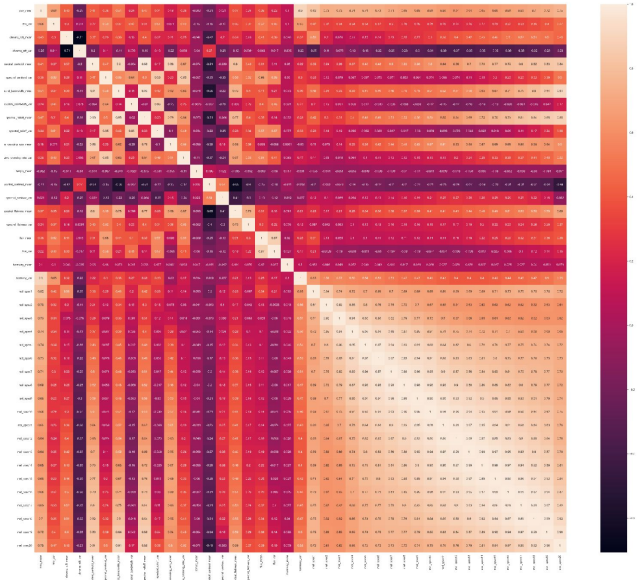
Data Distribution



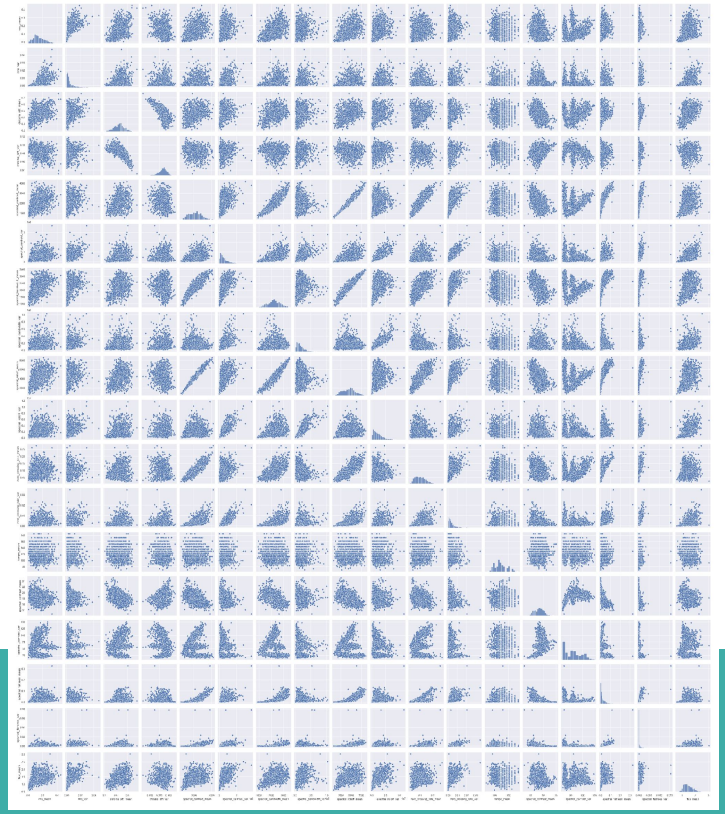
Histogram for all features



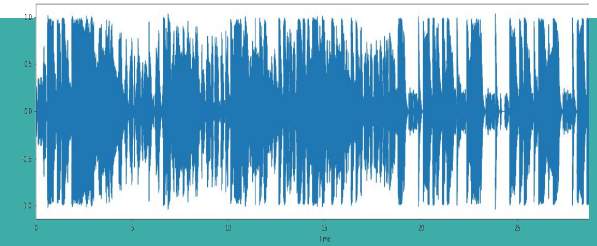
Heatmap



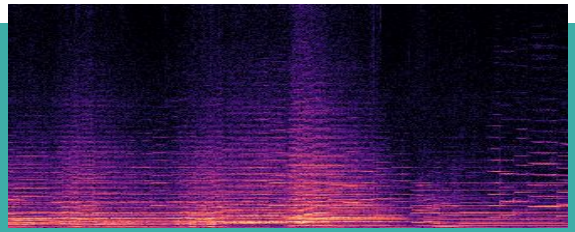
Pair Plot



Waveplot



Spectrogram



Baseline approaches - basic and advanced

CLustering -

Silhouette Score = (b - a) / max(a, b) where b = mean nearest-cluster and a = mean intra-cluster distance
Davies Bouldin Score = within-cluster distances / between-cluster distances, clusters which are farther apart and less dispersed will result in a better score

Model	Hyperparameters	Silhouette Score	Davies Bouldin score
K Means	n_clusters=9 n_init=9 algorithm='auto'	35%	83%
Birch	n_clusters =9 threshold=1.9 branching_factor=20	32%	68%
Gaussian Mixture	n_components=9 covariance_type='spherical'	52%	66%

Clustering is not a classification task hence analyzing accuracy is not a very good idea instead we will check the purity of clusters and hence we have used **Silhouette Score** and **Davies Bouldin Score**(Lower the value better the clustering).

KNN :

- In preprocessing, I used **TruncatedSVD**, **StandardScaler** and **SelectKBest**. The features were wisely chosen with lots of trial in order to achieve best accuracy
- **Grid Search** is a technique which is used to tune the hyperparameters of the mode.
  Hyperparameters : metric='manhattan', n_neighbors=8, weights='distance'

ANN :

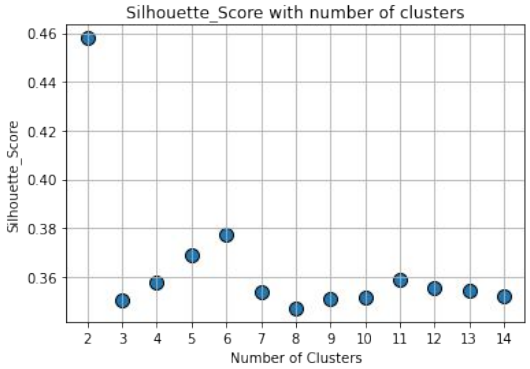
- Applied 4 hidden layers with number of neurons in each layers as - [512, 256, 64, 32]
- **Regularization** and **Activation** layers are used in first 3 layers. The final layer is **Softmax layer**.

CNN :

- The spectrogram and wavelet image data is treated as input, which are of size (256, 256, 3) after preprocessing.
- In contrast to traditional Neural Networks, the neurons in CNN layers are organised in three dimensions: width, height, and depth, and are connected to a smaller part of the layer before them.
- To minimise the spatial dimensions in our model, we employed three 2D convolutional layers, each followed by a **maxpool**.
- Then we built the model with modified parameters and trained it with a 6:4 validation set (currently).
- We use **Adam optimizer** to train our CNN model for 500 epochs with a learning rate of 0.0001, with categorical cross-entropy used as the loss function.

Results & Analysis

For clustering in order find optimal value of clusters, we used **Elbow Method.**



CNN Summary

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 256, 256, 32)	896
max_pooling2d (MaxPooling2D)	(None, 128, 128, 32)	0
conv2d_1 (Conv2D)	(None, 128, 128, 32)	9248
max_pooling2d_1 (MaxPooling2D)	(None, 64, 64, 32)	0
conv2d_2 (Conv2D)	(None, 64, 64, 64)	18496
max_pooling2d_2 (MaxPooling2D)	(None, 32, 32, 64)	0
dropout (Dropout)	(None, 32, 32, 64)	0
flatten (Flatten)	(None, 65536)	0
dense (Dense)	(None, 128)	8388736
dense_1 (Dense)	(None, 10)	1290

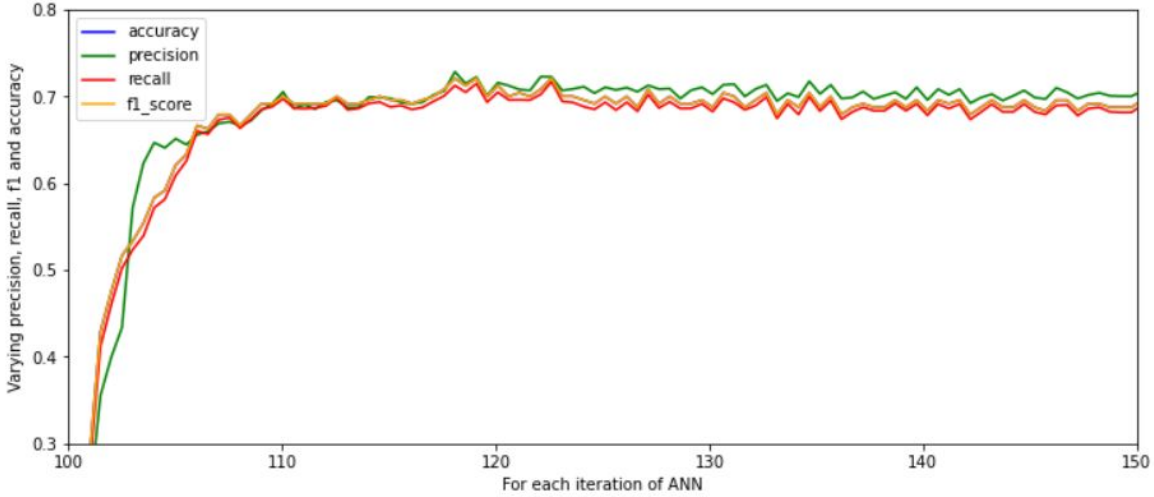
Model Performance

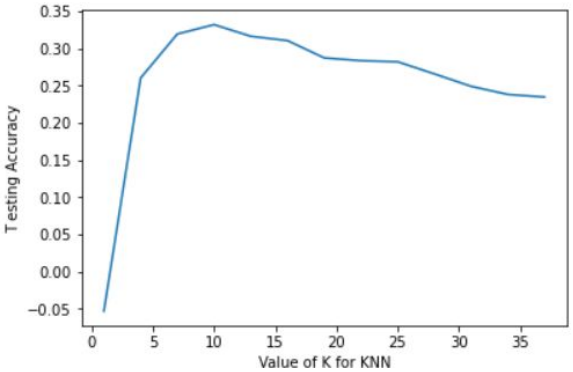
	Accuracy	Precision	F1 score	Recall score
KNN	0.64583333	0.6668102	0.6371789	0.6401271
Logistic Regression	0.73333333	0.7503128	0.7375711	0.7334111
ANN	0.70833331	0.72916666	0.72916666	0.72916666

K Neighbours Regressor

	K	Test Score	Train Score
0	1	-0.052785	1.000000
1	4	0.260487	0.619935
2	7	0.319598	0.528678
3	10	0.332161	0.472743
4	13	0.316511	0.435334
5	16	0.310699	0.394873
6	19	0.287449	0.369059
7	22	0.283707	0.343023
8	25	0.282222	0.321068
9	28	0.265951	0.304268
10	31	0.249222	0.291825
11	34	0.238357	0.278346
12	37	0.234834	0.265232

For 200 iterations in ANN, how accuracy, precision, recall, f1_score varies -





After applying KNN, this was the result for each of the genre obtained -

```
Default :
Accuracy score: 0.6375
Overall Precision: 0.672314073808639
Overall Recall:0.6311173029374784
                precision    recall  f1-score   support

Electric-Genre       0.57      0.62      0.59        21
         blues       1.00      0.47      0.64        19
     classical       0.74      0.85      0.79        20
       country       0.56      0.71      0.63        21
         disco       0.32      0.64      0.43        14
        hiphop       0.69      0.41      0.51        27
          jazz       0.83      0.33      0.48        15
         k-pop       0.73      0.95      0.83        20
         metal       0.88      0.88      0.88        24
           pop       0.67      0.64      0.65        22
        reggae       0.57      0.50      0.53        16
          rock       0.52      0.57      0.55        21

      accuracy                           0.64       240
     macro avg       0.67      0.63      0.62       240
  weighted avg       0.68      0.64      0.63       240
```

Timeline

- We were able to follow our timeline mentioned in the proposal.

What we have done?	What we are planning to do?
Feature Extraction from sound files, EDA and Data Preprocessing	To Apply SVD to on the data set and record valuable output to compare and contrast with all other modes. Apply some of the tuning techniques on the ANN model.
Baseline Models - KNN, K Means, Birch, Gaussian Mixture, Logistic Regression	SVM, increase accuracy of CNN model, applying neural network model on the updated dataset.
Audio file collection, generating spectrograms and wavelets from the updated dataset, data preprocessing, CNN model	DBSCAN, Agglomerative Clustering, OPTICS and search different technique by which we can further improve our model performance
Applied ANN model (Artificial Neural Network)	Build and testing of recommendation system using models