Music Recommendation System

Final Project Presentation

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

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

Problem Statement & Dataset

PROBLEM STATEMENT

In the present day scenario, due to many songs, recommendation systems built on music metadata (like song title or artist name, the track length, the BPM, or genre) do not give the best user experience. We are trying to solve this problem and provide them with a better music experience by audio sampling the wav files and extracting the features. The goal is to improve efficiency and introduce automation for the said task.

DATASET

Obtained 1000 files from the GTZAN dataset (link). Added two genres with 100 wav files each (Electric and k-pop). Extracted 30 sec from each wav file (from 30s offset) for further audio signal processing. Extracted features include RMS, chroma_stft, spectral_centroid, and many others using Librosa to create a 1200x43 dimensional dataset. The spectrogram and wavelet plots of all the 30s audio files were generated, shuffled, and divided into train and test datasets for image-based classification purposes. The images were rescaled and converted into 256 x 256 x 3 (RGB) format. Fitting several set image control parameters to the training dataset.

Progress summary until intermediate submission

- Data collection, Data extraction, Data organization, Remove Nan values, Remove highly skewed features, Detected and Remove outliers, Remove features based on confusion matrix, Normalise dataset, Spectrogram and waveform generation, Images normalization and standardization, Applying image transforms (using tensorflow).

- Applied preprocessing techniques to bring dataset into most accurate form. Techniques applied - PCA, TSNE, Normalization, Standardization, Truncated SVD, SelectKBest.

- Methodology - In recommendation system, we have 3 types of system - Content-Based Filtering, Demographic Filtering, Collaborative Filtering.

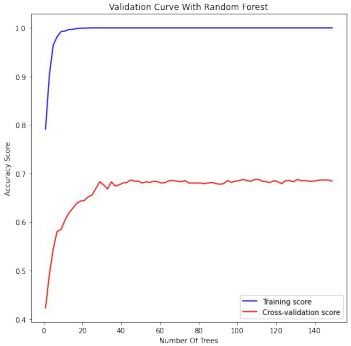
- We aim to make content - Based Filtering. For that we first analysed the genre classification performance of model on different model -
  1. KNN
  2. ANN
  3. CNN
  4. Clustering (KMean, Gaussian, Birch, and DBSCAN)
  5. Logistic Regression

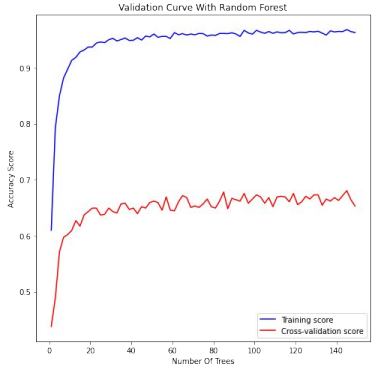
- Loss minimization were studied, applied to reduce the rmse value (loss) and increase accuracy.

Progress after intermediate submission - Classification

Random Forest

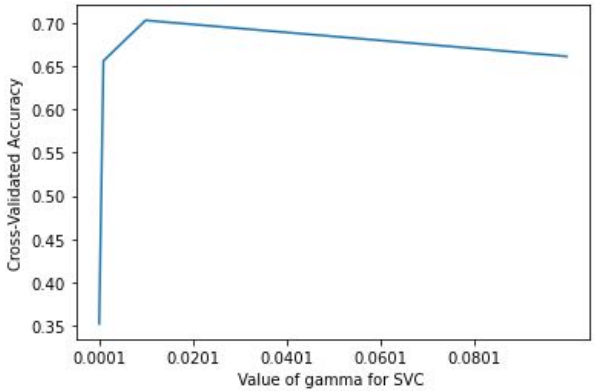
- Applied random forest on training data using k-Fold and analyse model using ‘validation_curve’.
- GridSearchCV to tune the parameters of random forest -
  - Hyperparameters (n_estimators=275,min_samples_split=9, min_samples_leaf=3,max_depth=145,bootstrap=True)
- After tuning the parameters, we didn’t notice that much improvement in model accuracy on testing data but while training our validation accuracy improves.





Svm

- Initially data was preprocessed - dimensionality reduction using TruncatedSVD, Standard scaler and SelectKBest. (43 ➔ 25)
- Applied SVM on the training data using Strategic Kfold. The base model was giving an accuracy of 65%. After Kfold, it increased to 69%.
- Applied grid search on linear, rbf and poly with variable parameter for gamma and C. Best hyper parameters came out to be {'C': 10, 'gamma': 0.01, 'kernel': 'rbf'}.
- On further applying K Fold on top of that, accuracy for best split for train was observed to be 76.6% and for test 71.54%



Progress after intermediate submission - Recommendation

Clustering

Recommendation system using Clustering. Then, in PCA projected space, we calculated the euclidean distance between the picked song and the recommended song then choose the best 10 shortest distance songs from a list of songs for recommendation.

```
Select song id(between 0 to 1199) you would like to listen:
134 ---> classical.00034.wav
801 ---> metal.00001.wav
1111 ---> rock.00011.wav
979 ---> pop.00079.wav
912 ---> pop.00012.wav
467 ---> press-pause-walz-main-version-01-38-17207.wav
586 ---> hiphop.00086.wav
684 ---> jazz.00085.wav
587 ---> hiphop.00087.wav
963 ---> pop.00063.wav
277 ---> country.00077.wav
1166 ---> rock.00066.wav
1057 ---> reggae.00057.wav
827 ---> metal.00027.wav
983 ---> pop.00083.wav
----------------------------------------------------------------
277
0    classical.00053.wav
1         rock.00093.wav
2      country.00098.wav
3    classical.00054.wav
4      country.00066.wav
5    classical.00042.wav
6         rock.00048.wav
7         jazz.00066.wav
8      country.00072.wav
9         rock.00006.wav
Name: fileName, dtype: object
----------------------------------------------------------------
Want more!!!!(y/n) : n
```

KNN

On the best model found during classification, we further built our recommendation system. When user inputs a file name, we find the featuremap for that file. Then we plot that file on out N dimensional space. Then we calculate N nearest neighbors on the basis of euclidean distance. Then for these mapped data points, we got there music files and recommended to the user.

```
If you like
 blues.00002.wav

 You might enjoy the following songs:

1 blues  -->     blues.00002.wav         with a distance of      0.0
2 rock   -->     rock.00011.wav          with a distance of      2.7803497824049
3 rock   -->     rock.00019.wav          with a distance of      3.001315490674597
4 hiphop  -->    hiphop.00070.wav        with a distance of      3.146432577232529
5 country  -->   country.00053.wav       with a distance of      3.180620099214508
6 k-pop  -->     02. Tamed-Dashed.wav    with a distance of      3.227529953712183
```

Progress after intermediate submission

ANN

The Recommendation System was built on the best fitted model. We picked the in-between layer, in our case we picked the layer('dense1') with the number of neurons - 128. The model was used to predict the output only till the 'dense1' layer. (Figure 20). The output is called the embedding. Cosine similarity of sklearn.metrics.pairwise library is used.

When the user inputs the file name present
in the database, we find the feature vector corresponding to the file. Then we calculate the similarity score for all the files and sort this in decreasing order of similarity score. The top 6 indexes are used to find the file name and recommend it to the user. The recommendation for the blues genre was found to be the most accurate.

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

Similar songs to blues.00002.wav

1 blues          -->  blues.00002.wav
                  Recommendation score = 0.9999998

2 rock           -->  rock.00009.wav
                  Recommendation score = 0.88283974

3 disco          -->  disco.00086.wav
                  Recommendation score = 0.87353134

4 country        -->  country.00072.wav
                  Recommendation score = 0.8699937

5 country        -->  country.00066.wav
                  Recommendation score = 0.8678256

6 blues          -->  blues.00080.wav
                  Recommendation score = 0.8677113

7 reggae         -->  reggae.00066.wav
                  Recommendation score = 0.86568165

8 blues          -->  blues.00010.wav
                  Recommendation score = 0.8656725

9 blues          -->  blues.00000.wav
                  Recommendation score = 0.86476517

10 rock          -->  rock.00064.wav
                  Recommendation score = 0.8637908


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

Analysis and Ablation

In ANN, CNN, and KNN, the evaluation metrics use accuracy, precision, F1-score, Recall score. Further insights into the model have been obtained by analyzing the confusion matrix, heat maps, learning curves (for train-test accuracy and loss), Silhouette and Davies Bouldin Scores evaluate clusters' separation (purity) for clustering models. All 12 genres were encoded (0-11) for coding and NN classification purposes.

CNN

Several measures were taken to increase the accuracy of the model, or reducing the sparsity of the dataset. Several pretrained models on the same dataset, techniques like Cut Mix and Mixup augmentation, increasing image sizes (for better feature visibility) were tried. Dropout reduction, batch and layer size optimization were also tried. Random shuffling and other techniques were also used to smartly select models to feed the model. But still, the accuracy only improved to atmost 55% for the given dataset. But due to lack of differences in the image spectrogram samples, it was difficult to find a good model on the image dataset.
In CNN the best accuracy is obtained for classical genre while the worst is for reggae genre. There are several true label genres mistaken for some other genre because of similarity in spectrograms but heat map is not symmetrical because of true label spectrograms being subsets of predicted labels spectrograms in similarity.

Figure 11: Confusion matrix of CNN for predicted and true labels

Analysis and Ablation

- Considering the analysis of state of art KNN [12], the model was made, which gave good performance results (69 %). In this model jazz and rock were the genres which were giving lower accuracy 57 and 62.

Model	Best Fitted Genre	Worst Fitted Genre
CNN	Classical	Reggae
SVM	K-pop	Disco, Country
Logistic	Electric Genre	Rock, Disco
KNN	Metal	Disco, Rock
ANN	Classical	Disco, Country

After building recommendation system from different models, we analyzed the recommended song from different models and we found that recommendation results showed most suitable songs in case of ANN across all the genres.
In ANN, we are able to find the music files which have features that are 88% similar to the recommended songs because the applied model of ANN had a better fit. ANN was giving us best accuracy on testing data and smallest loss on training data as compared to other models.

Contribution of each Team Member

Jasdeep Singh	Data Scraping / Collection, Feature Extraction using librosa, Data analysis, Data Visualization, PCA, Normalization, Standardization, Truncated SVD, SelectKBest, Logistic and Analysis, SVM and Analysis, ANN and Analysis, GridSearchCV and Analysis. Recommendation system - ANN, KNN
Shanu	Data Collection and preprocessing,Feature Extraction using Librosa, Data Dimension Reduction - PCA, TSNE, Normalization, K Means, DBSCAN, Birch and Gaussian Clustering, Random Forest. Recommendation using Clustering algorithms
Siddharth Singh Kiryal	Data Scraping/Collection and preprocessing: song length reduction (to 3s), spectrogram and waveform generation, feature extraction, spectrogram image train-test generation, Dimensionality Reduction- Normalization, Standardization, Image transformations, genre prediction, analysis and evaluation - heatmaps plots, test, validation loss accuracies plots etc.

BACKUP SLIDES

References

[1] https://www.kaggle.com/andradaolteanu/gtzan-dataset-music-genre-classification
[2] https://towardsdatascience.com/silhouette-coefficient-validating-clustering-techniques-e976bb81d10c
[3] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.davies_bouldin_score.html
[4] https://towardsdatascience.com/silhouette-coefficient-validating-clustering-techniques-e976bb81d10c
[5] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.davies_bouldin_score.html
[6] https://machinelearningmastery.com/learning-curves-for-diagnosing-machine-learning-model-performance/
[7] https://machinelearningmastery.com/overfitting-machine-learning-models/
[8] https://github.com/christianversloot/machine-learning-articles/blob/main/how-to-check-if-your-deep-learning-model-is-underfitting-or-overfitting.md
[9] https://machinelearningmastery.com/overfitting-and-underfitting-with-machine-learning-algorithms/#:~:text=Overfitting%3A%20Good%20performance%20on%20the,poor%20generalization%20to%20other%20data
[10] https://www.researchgate.net/publication/237054335_The_GTZAN_dataset_Its_contents_its_faults_their_effects_on_evaluation_and_its_future_use
[11] https://cs229.stanford.edu/proj2011/HaggbladeHongKao-MusicGenreClassification.pdf
[12] https://www.researchgate.net/publication/334780384_Automatic_music_genre_recognition_for_in-car_infotainment#pf7

Model	Accuracy	Precision	F1 score	Recall score
KNN	0.6667	0.6892	0.6693	0.6667
Logistic Regression	0.7333	0.7503	0.7376	0.7334
ANN	0.8083	0.7927	0.8042	0.8023
CNN	0.52	0.5175	0.5083	0.5166
Random Forest	0.580	0.67	0.665	0.685
SVM	0.73	0.74	0.73	0.72

Table 1: Accuracy, Precision, F1 and Precision scores for different models observed

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%

Table2: Hyperparameters, Silhouette Scores, and Davies Bouldin Scores for different clustering models

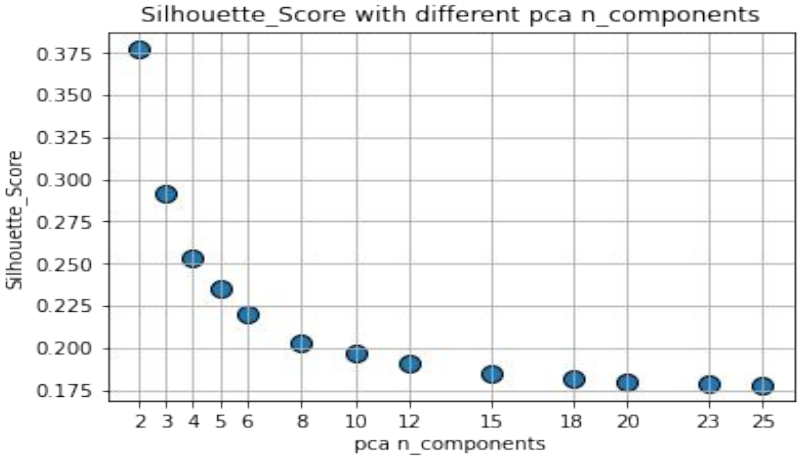


Figure 4: Silhouette_Score with different PCA components in K_Mean(n_clusters=9 n_init=9)

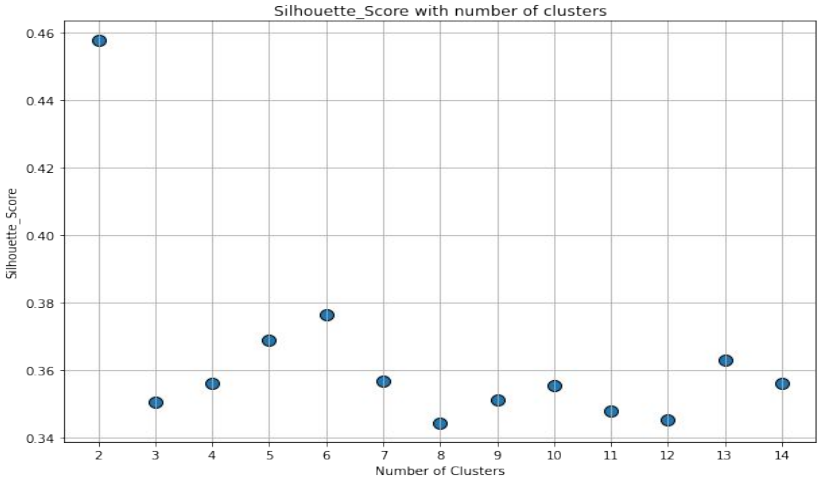


Figure 3: Silhouette_Score with number of clusters (K_Mean)

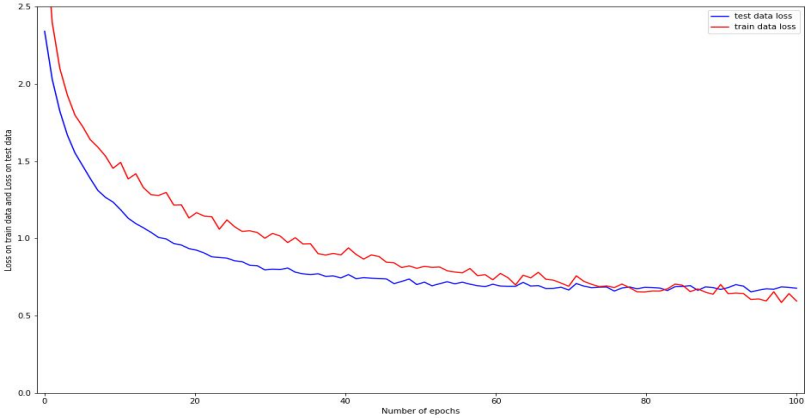


Figure 2: ANN - Varying test loss and train loss with epochs

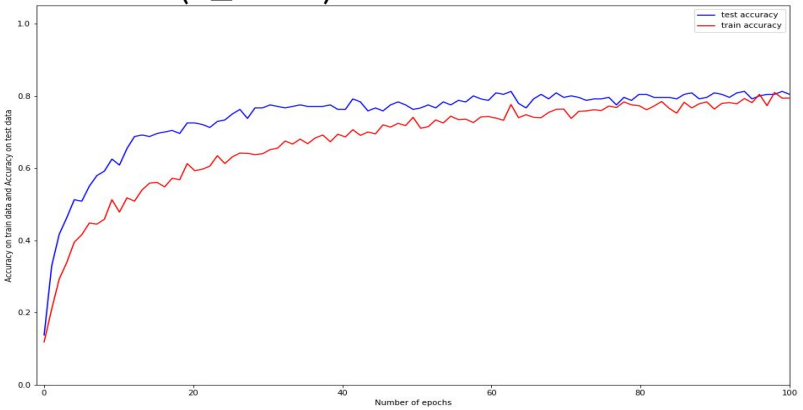


Figure 1: ANN - Varying test and train accuracy with epochs

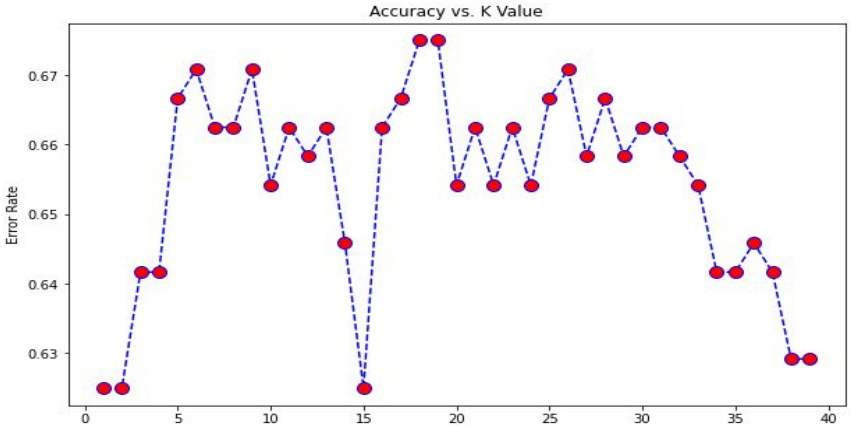


Figure 5: Accuracy varying with k number of neighbors

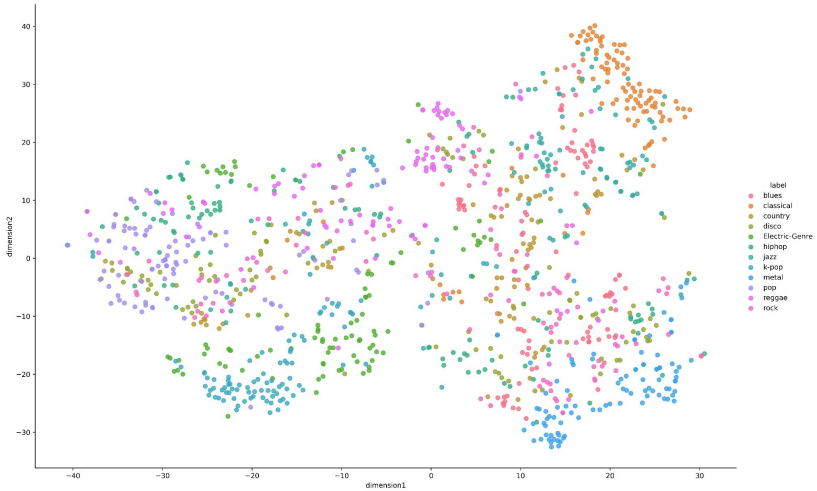


Figure 6: TSNE(n_components=2,n_iter=500) on original data

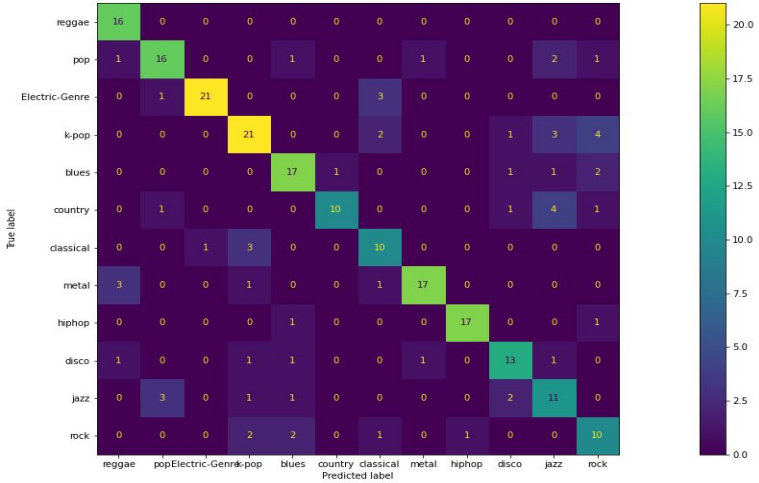


Figure 7: Confusion matrix of ANN for predicted and true labels

	precision	recall	f1-score	support
Electric-Genre	0.89	0.89	0.89	18
blues	0.81	0.85	0.83	20
classical	1.00	0.91	0.95	23
country	0.65	0.92	0.76	12
disco	0.58	0.64	0.61	22
hiphop	0.70	0.89	0.78	18
jazz	0.82	0.88	0.85	16
k-pop	0.94	0.83	0.88	18
metal	0.93	0.86	0.89	29
pop	0.83	0.77	0.80	26
reggae	0.68	0.68	0.68	19
rock	0.69	0.47	0.56	19
accuracy			0.80	240
macro avg	0.79	0.80	0.79	240
weighted avg	0.80	0.80	0.80	240

Figure 7: Class Wise accuracy of ANN

```
Model: "sequential_6"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_14 (Conv2D)           (None, 254, 254, 32)      896
_________________________________________________________________
max_pooling2d_14 (MaxPooling (None, 127, 127, 32)      0
_________________________________________________________________
conv2d_15 (Conv2D)           (None, 125, 125, 32)      9248
_________________________________________________________________
max_pooling2d_15 (MaxPooling (None, 63, 63, 32)        0
_________________________________________________________________
conv2d_16 (Conv2D)           (None, 62, 62, 32)        4128
_________________________________________________________________
max_pooling2d_16 (MaxPooling (None, 31, 31, 32)        0
_________________________________________________________________
flatten_6 (Flatten)          (None, 30752)             0
_________________________________________________________________
dense_12 (Dense)             (None, 64)                1968192
_________________________________________________________________
dropout_6 (Dropout)          (None, 64)                0
_________________________________________________________________
dense_13 (Dense)             (None, 12)                780
=================================================================
Total params: 1,983,244
Trainable params: 1,983,244
Non-trainable params: 0
_________________________________________________________________
```

Figure 8: CNN model architecture used

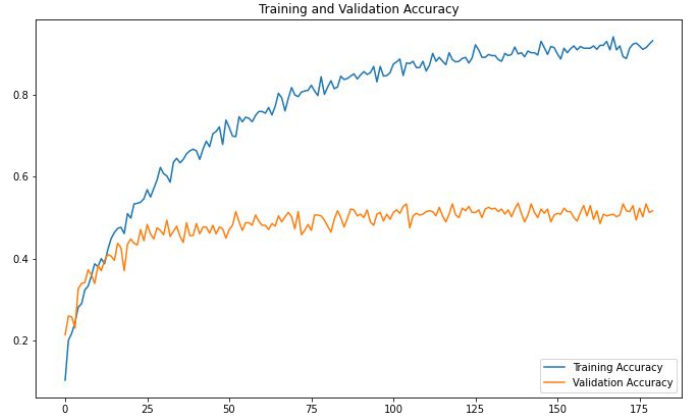


Figure 9: CNN - Varying validation and train accuracy with epochs (for analysis purpose) (is overfitted)

Figure 10: CNN - Varying validation and train loss with epochs (for analysis purpose) (is overfitted)

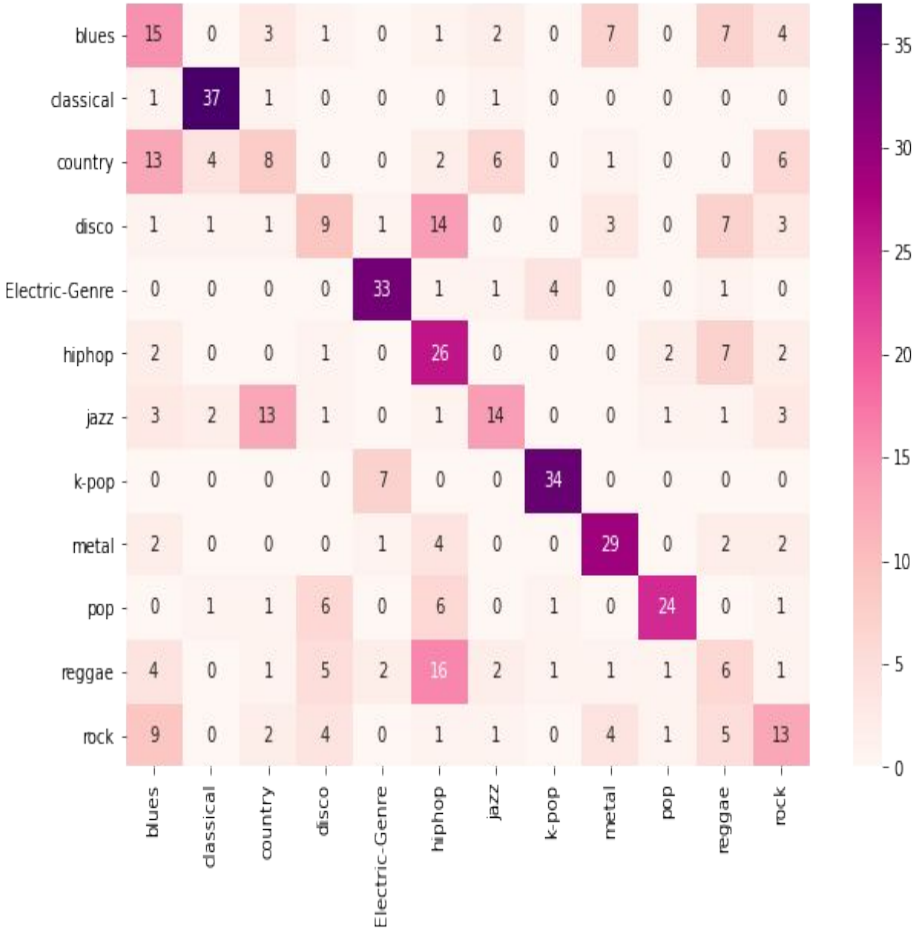


Figure 11: Confusion matrix of CNN for predicted and true labels

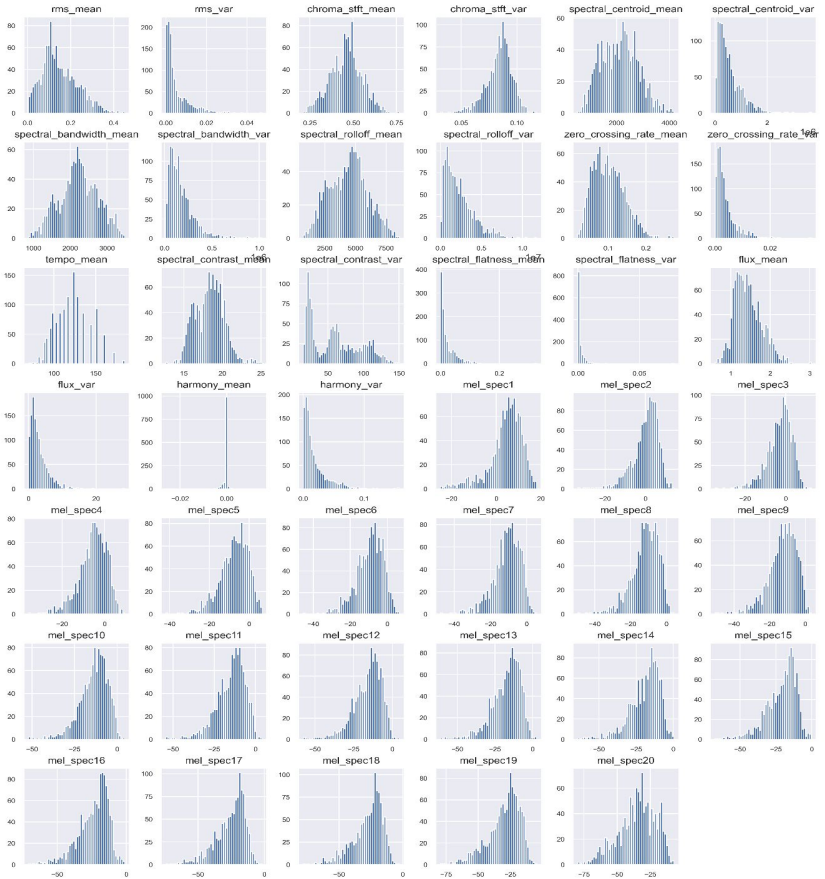


Figure 12: Histogram for all the features extracted using librosa

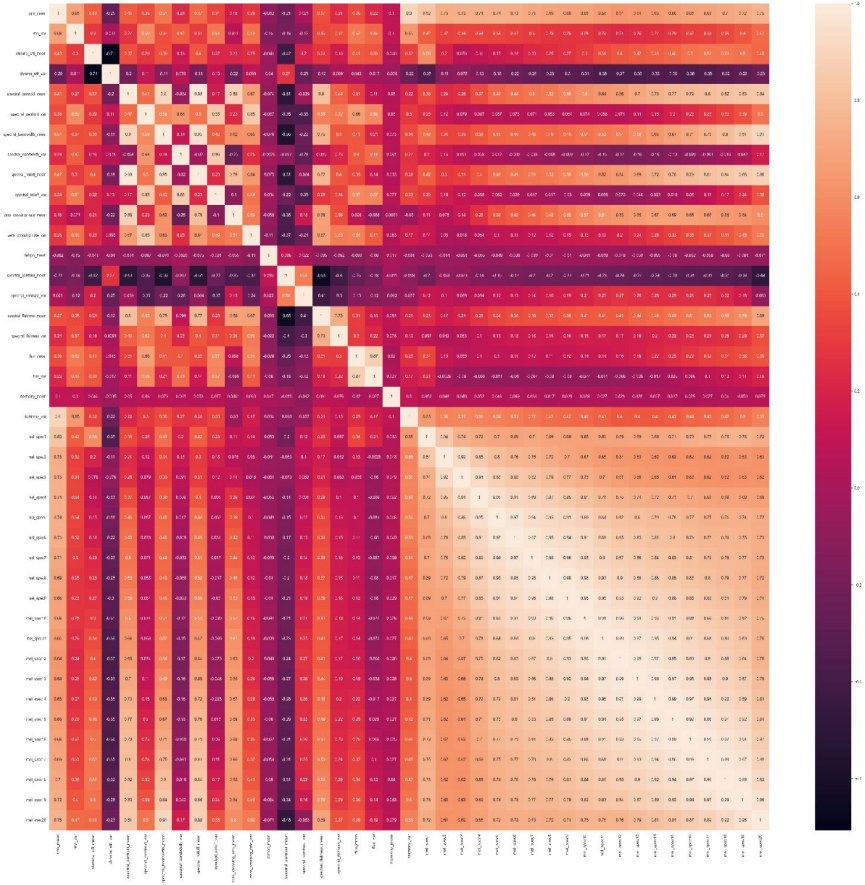


Figure 13: Heat map of initial data set 1200 rows, 43 column

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

Similar songs to blues.00002.wav

1 blues          -->  blues.00002.wav
                      Recommendation score = 0.9999998

2 rock           -->  rock.00009.wav
                      Recommendation score = 0.88283974

3 disco          -->  disco.00086.wav
                      Recommendation score = 0.87353134

4 country        -->  country.00072.wav
                      Recommendation score = 0.8699937

5 country        -->  country.00066.wav
                      Recommendation score = 0.8678256

6 blues          -->  blues.00080.wav
                      Recommendation score = 0.8677113

7 reggae         -->  reggae.00066.wav
                      Recommendation score = 0.86568165

8 blues          -->  blues.00010.wav
                      Recommendation score = 0.8656725

9 blues          -->  blues.00000.wav
                      Recommendation score = 0.86476517

10 rock          -->  rock.00064.wav
                      Recommendation score = 0.8637908

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

Figure 14: Recommendation System for Ann using the classification model

```
If you like
 blues.00002.wav

 You might enjoy the following songs:

1 blues  -->     blues.00002.wav        with a distance of      0.0
2 rock   -->     rock.00011.wav         with a distance of      2.7803497824049
3 rock   -->     rock.00019.wav         with a distance of      3.001315490674597
4 hiphop  -->    hiphop.00070.wav       with a distance of      3.146432577232529
5 country  -->   country.00053.wav      with a distance of      3.180620099214508
6 k-pop  -->     02. Tamed-Dashed.wav   with a distance of      3.227529953712183
```

Figure 15: Recommendation System for KNN using K nearest neighbor on the best hyperparameter found on the classification model.

```
Select song id(between 0 to 1199) you would like to listen:
134 ---> classical.00034.wav
801 ---> metal.00001.wav
1111 ---> rock.00011.wav
979 ---> pop.00079.wav
912 ---> pop.00012.wav
467 ---> press-pause-walz-main-version-01-38-17207.wav
586 ---> hiphop.00086.wav
684 ---> jazz.00085.wav
587 ---> hiphop.00087.wav
963 ---> pop.00063.wav
277 ---> country.00077.wav
1166 ---> rock.00066.wav
1057 ---> reggae.00057.wav
827 ---> metal.00027.wav
983 ---> pop.00083.wav
------------------------------------------------------------
277
0    classical.00053.wav
1         rock.00093.wav
2       country.00098.wav
3    classical.00054.wav
4       country.00066.wav
5    classical.00042.wav
6         rock.00048.wav
7         jazz.00066.wav
8       country.00072.wav
9         rock.00006.wav
Name: fileName, dtype: object
------------------------------------------------------------
Want more!!!!(y/n) : n
```

Figure 17 : Recommendation using Clustering Algorithms

	precision	recall	f1-score	support
Electric-Genre	0.89	0.73	0.80	22
blues	0.73	0.89	0.80	18
classical	0.82	0.93	0.87	15
country	0.55	0.65	0.59	17
disco	0.45	0.53	0.49	19
hiphop	0.71	0.77	0.74	26
jazz	0.67	0.71	0.69	14
k-pop	0.86	0.86	0.86	21
metal	0.86	0.95	0.90	20
pop	0.86	0.82	0.84	22
reggae	0.75	0.65	0.70	23
rock	0.57	0.35	0.43	23
accuracy			0.73	240
macro avg	0.73	0.74	0.73	240
weighted avg	0.73	0.73	0.72	240

Figure 16: Best hyperparameters for SVM - KFold accuracy, classification report.

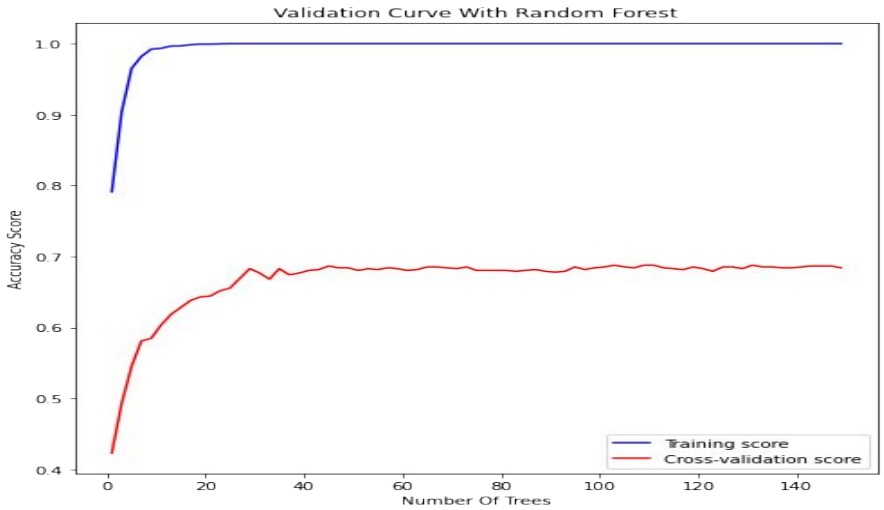


Figure 18: Validation Curve with basic Random Forest

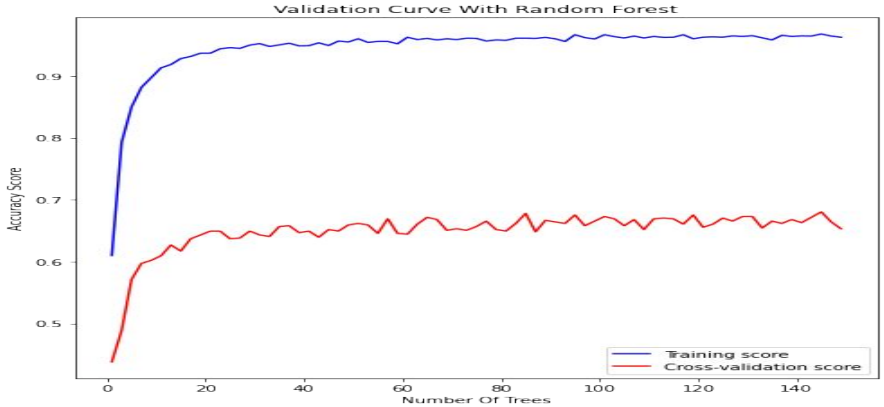


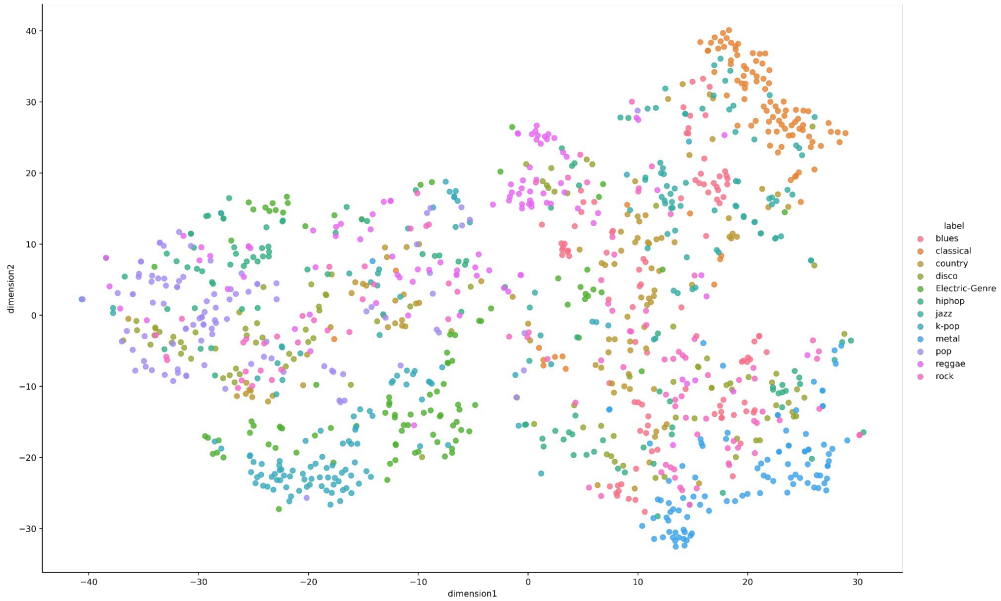
Figure 19: Validation curve with parameter tuned Random Forest

```
Model: "sequential_18"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 flatten_18 (Flatten)        (None, 41)                0

 dense_0 (Dense)             (None, 256)               10752

 normalize_0 (BatchNormaliza  (None, 256)              1024
 tion)

 dropout_0 (Dropout)         (None, 256)               0

 dense_1 (Dense)             (None, 128)               32896

 normalize_1 (BatchNormaliza  (None, 128)              512
 tion)

 dropout_1 (Dropout)         (None, 128)               0

 output (Dense)              (None, 12)                1548

=================================================================
Total params: 46,732
Trainable params: 45,964
Non-trainable params: 768
_________________________________________________________________
(1200, 128)
```

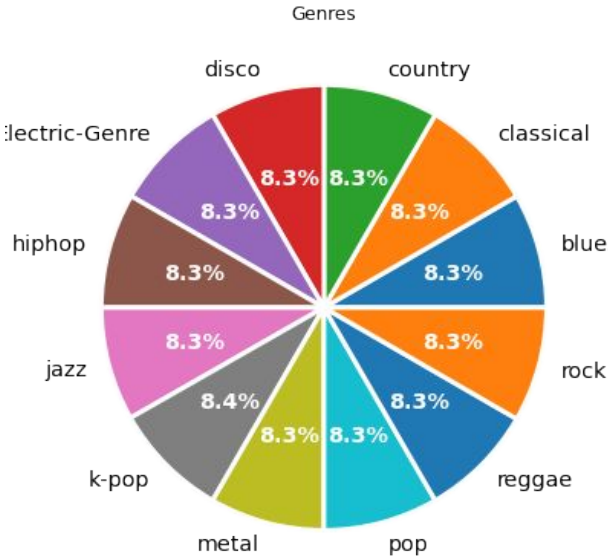
Figure 20: The model summary for ANN model

EDA

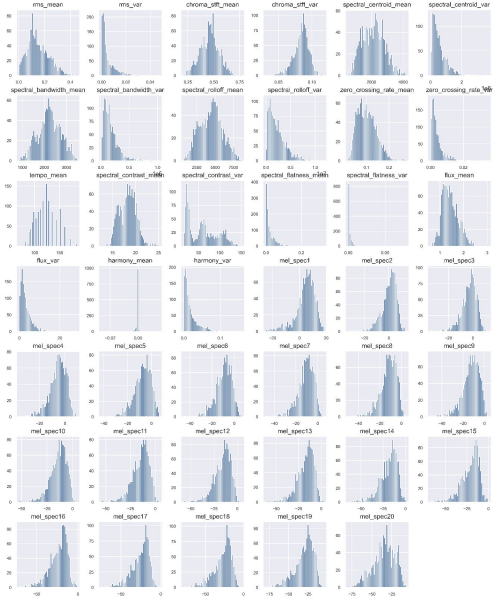
Data Plotting with `TSNE(n_components=2)`



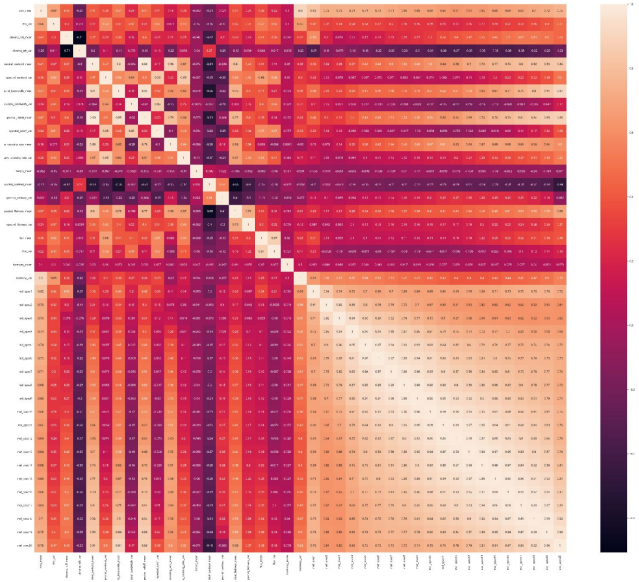
Data Distribution



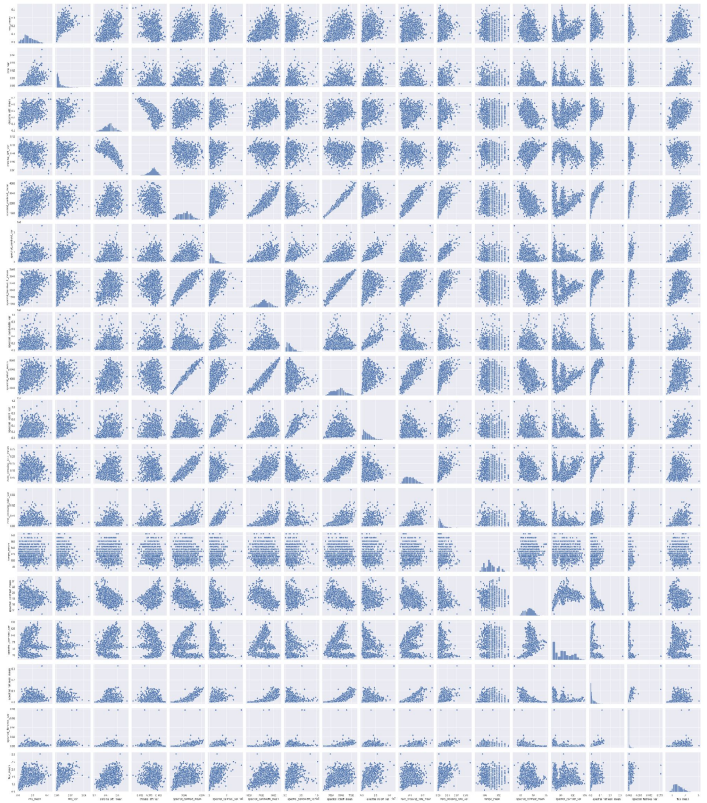
Histogram for all features



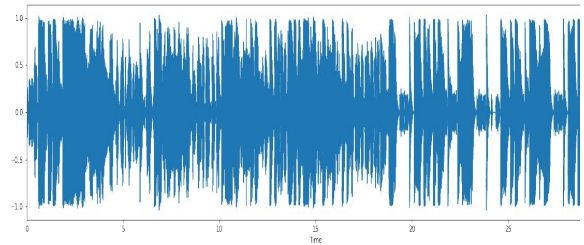
Heatmap



Pair Plot



Waveplot



Spectrogram