Music Recommendation - Multi Genre music
Music Generation - Classical music

ME228 Project

Project by :-

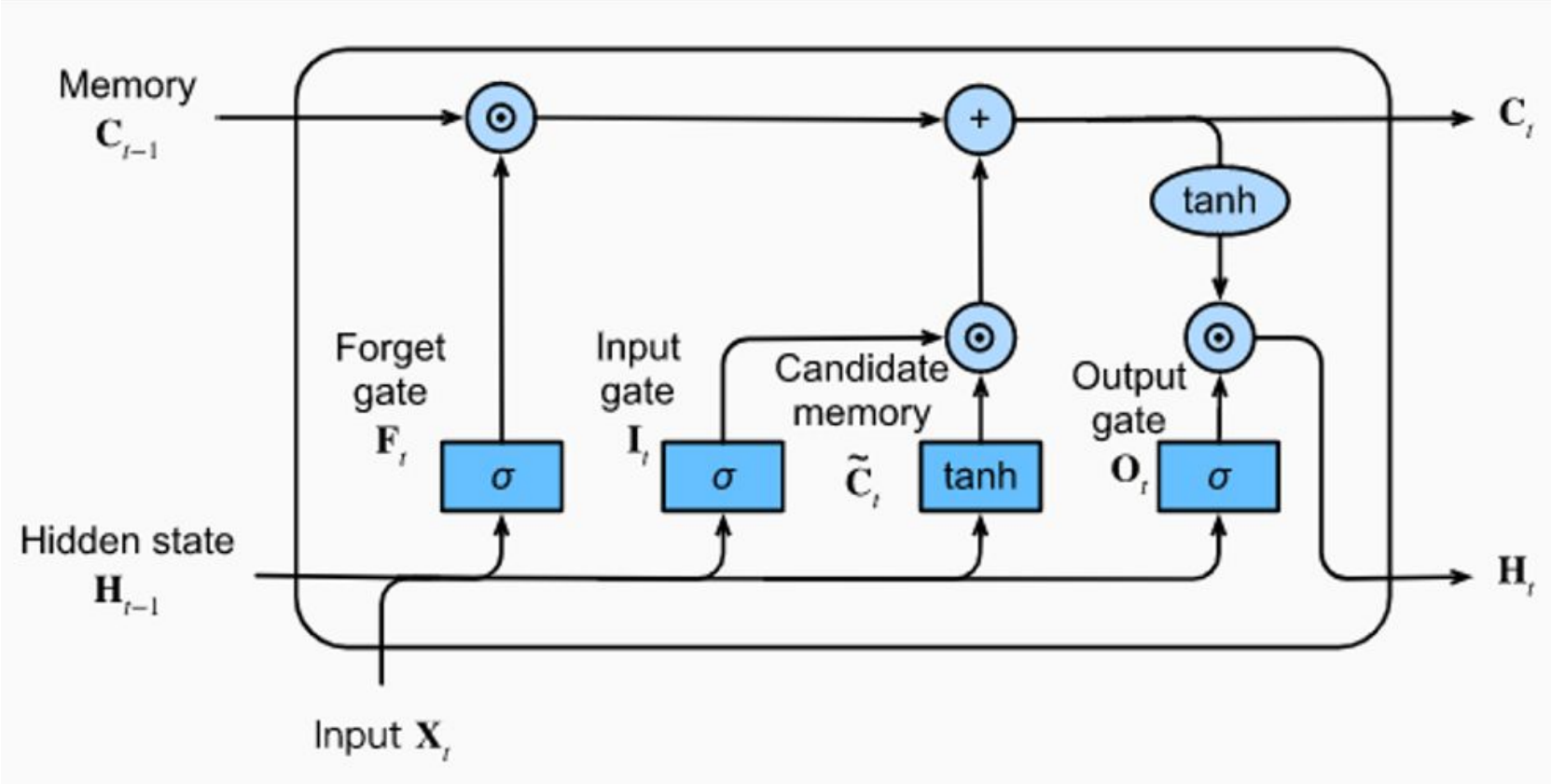
Adarsh Prajapati - 22B0050
Vaishnavi Kukde - 22B2197
Dhruvi Modi - 22B2194

Music Generation

Dataset Used : Classical Music MiDi (Beethoven)

Ref : http://www.piano-midi.de/beeth.htm

Model to be used : **LSTM - Long Short Term Memory**

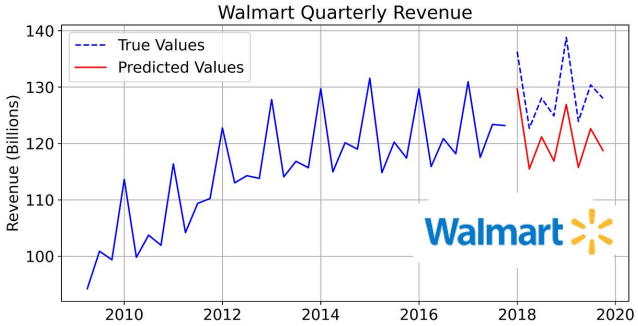


LSTM (Long Short-Term Memory) is a type of recurrent neural network (RNN) architecture designed to address the vanishing gradient problem, which occurs when training traditional RNNs over long sequences.

- **Memory Cells**: LSTM has memory cells that can store information for long periods. Think of these cells as containers that can keep important information without letting it fade away quickly.
- **Forget, Input, Output Gates**: LSTM has mechanisms called gates that control the flow of information. The forget gate decides what information to discard from the cell, the input gate decides what new information to store, and the output gate decides what information to output.
- **Long-Term Dependencies**: Unlike traditional recurrent neural networks, LSTM is designed to handle long-term dependencies in data. This means it's better at remembering context or patterns that occurred far back in the sequence, making it suitable for tasks like language translation or speech recognition where understanding context is crucial.

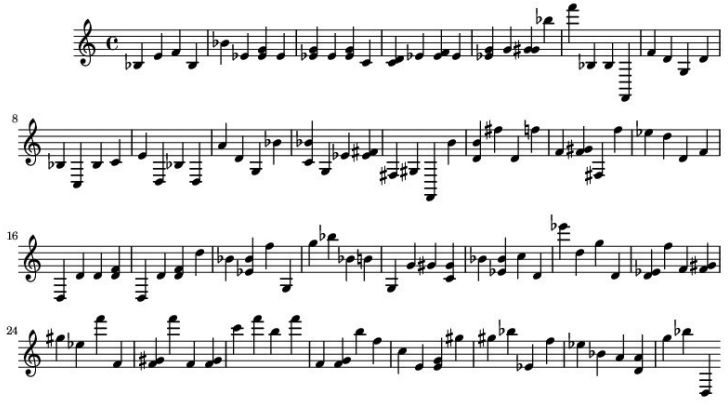
LSTM Usage -

- **Sequential Data Processing**: Its ability to capture long-range dependencies makes it well-suited for analyzing sequences of variable lengths.
- **Language Modeling and Translation**: LSTM-based models are widely employed in language modeling tasks, such as predicting the next word in a sentence or translating text from one language to another.
- **Time Series Forecasting**: LSTM networks are effective for forecasting future values in time series data, such as stock prices, weather patterns, or energy Consumption.
- **Anomaly Detection**: LSTM can be used for anomaly detection in various domains, including cybersecurity, manufacturing, and healthcare.



Input Data :

- The data is composed of MiDi files of classical piano performances by Beethoven.
- There are a total of 29 MiDi files to be used and the duration is around 5-7 minutes of each song.
- Total we have around 170 minutes of music to learn from.
- The files are uploaded and used to get the chords and notes by library music21.



Plan of Action :

1. Get the midi files input using the parse() function of music21
2. Extracting the notes and chords of the midi files.
3. Get the total number of notes in the dataset and the unique notes.
4. Find the rare notes(<100 occurrences) in the corpus, and remove them for better training.
5. Make feature list for sets of 40 notes, and target list for the next note in the respective list of 40 notes.
6. Encode the feature notes into integers and one-hot-encode the targets.
7. Normalize the input data by dividing it number of unique characters.
8. Employ an LSTM with the required parameters, for training the model.
9. The function *Melody_Generator(Note_Count)* is used for creating the new melody midi file with the number of notes passed as input.

Sample Output Audio :

Ref : The sample audio generated

Music Recommendation

Dataset Used : Gtzan Dataset Music Genre Classification

Ref : The GTZAN dataset

Input Data :

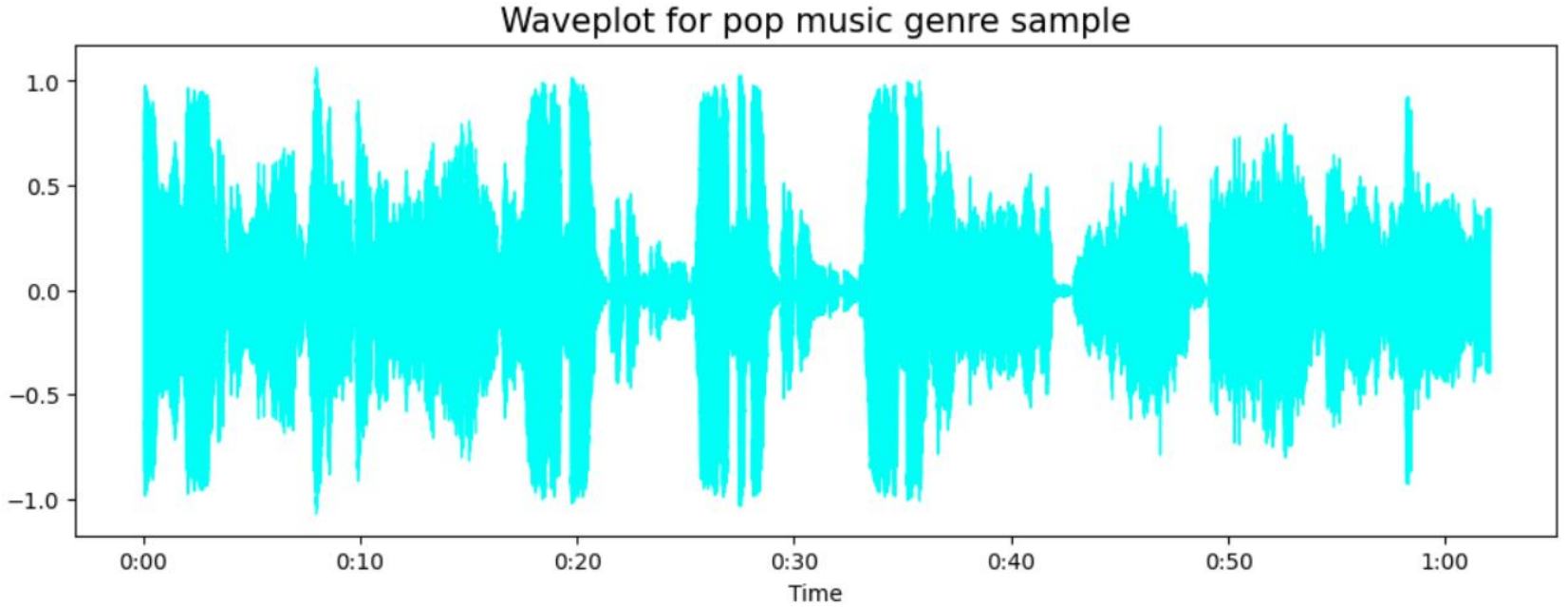
- The data is composed of wav files of different genres of music listed in the last coum, ‘label’ of the dataset.
- There are a total of 9990 wav files to be used and the duration is 30 seconds of each song.
- Total we have around 83 hours of music to analyse and learn from.
- We also have csv files containing various features of the wav audio files.
- We have fed csv file as the data set to our ML model.

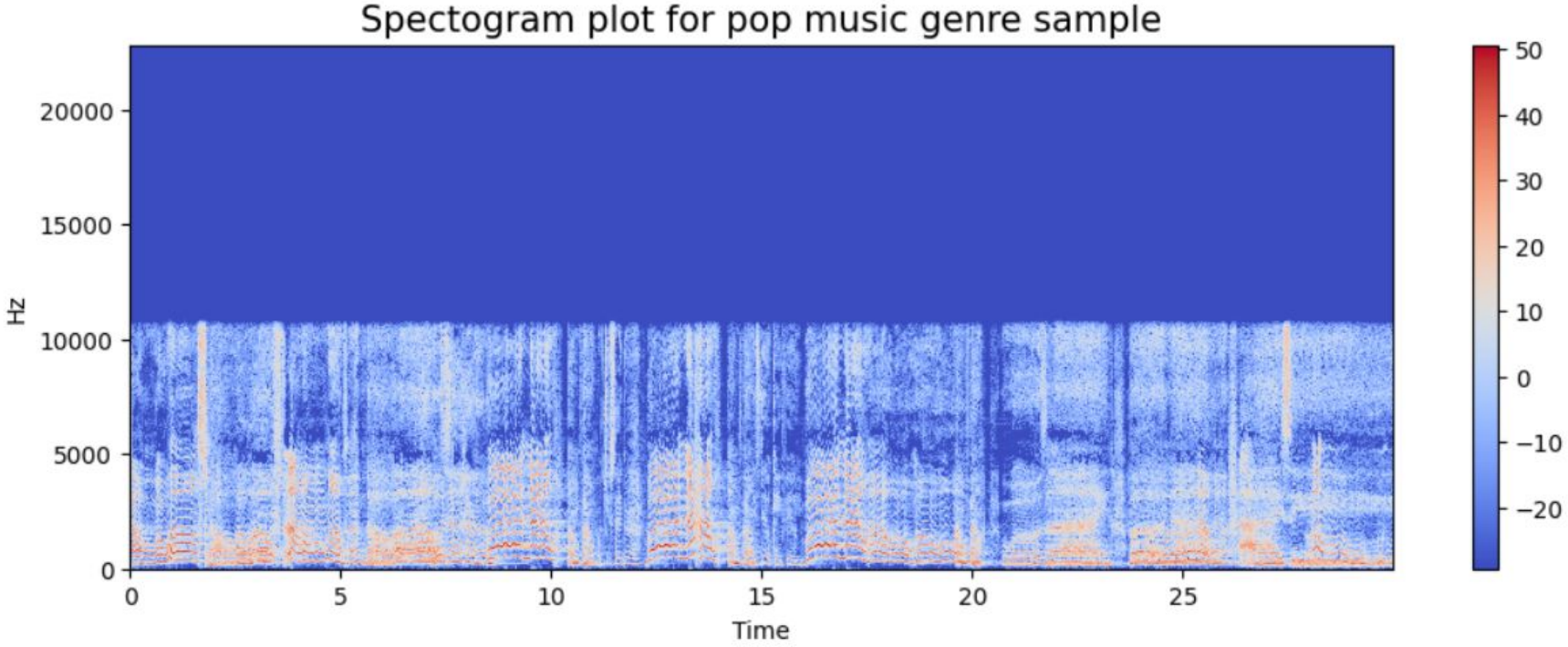
Plan of Action :

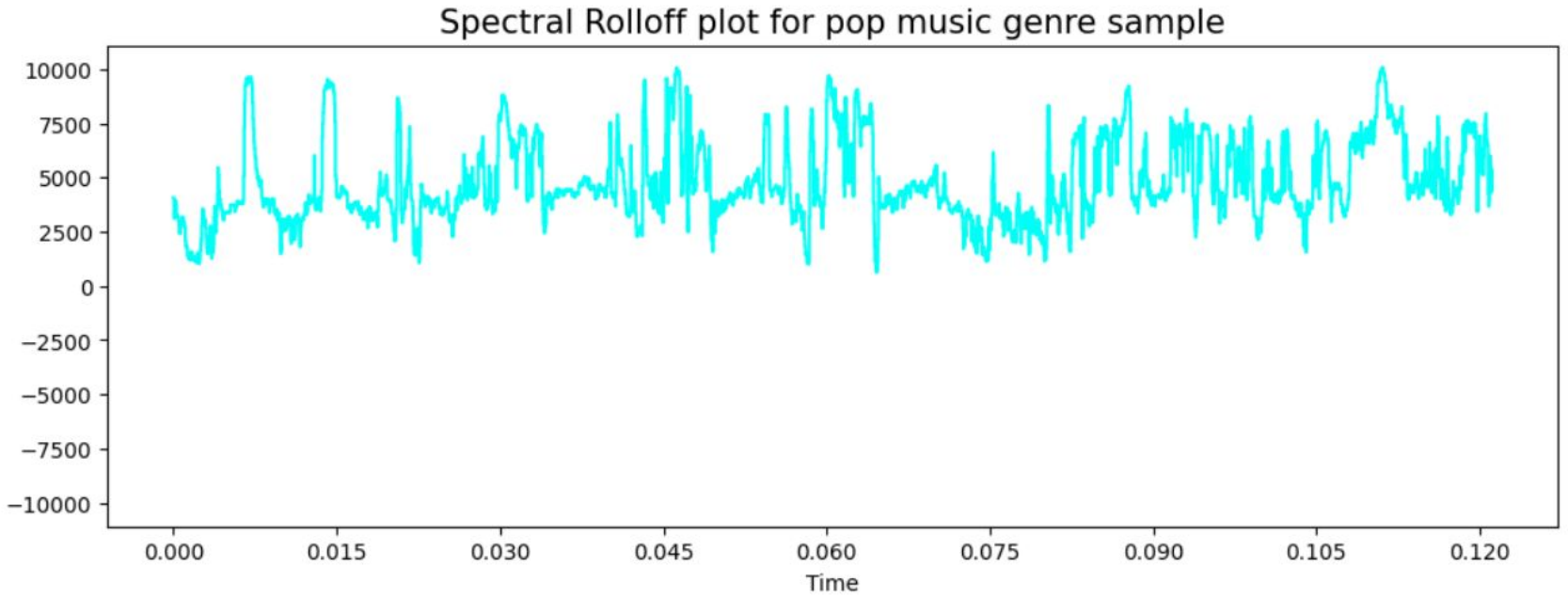
- Get the **wav files** input.
- Visualizing the given data into plots confirming the **different genres** of music present in the audio files of the dataset.
- Plotting **wave plots, spectrograms, spectral rolloff plots, chroma plots and zero crossing plots**. These plots are commonly used for analysing audio data.
- Applying **PCA** to reduce the dimensionality of the large data set.
- Creating a **correlation heatmap before and after VIF** to gauge strength of relationship between various features.

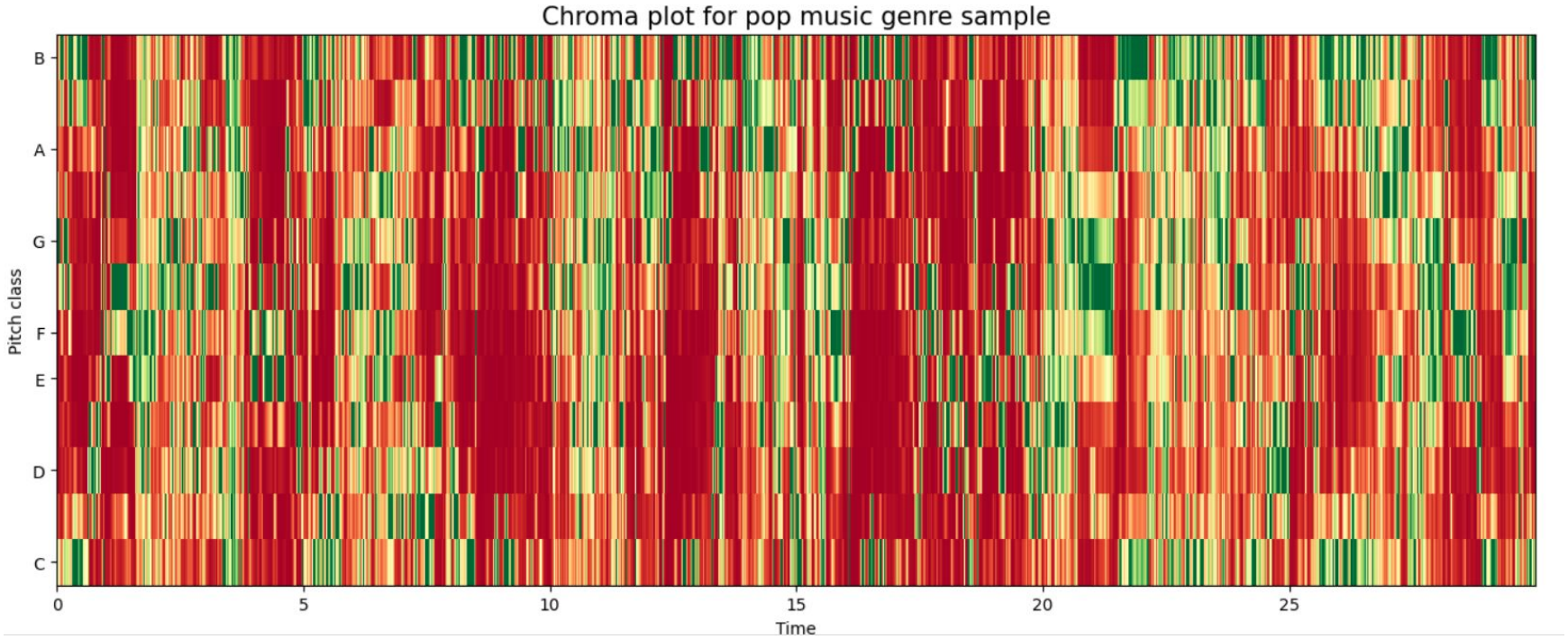
Plan of Action :

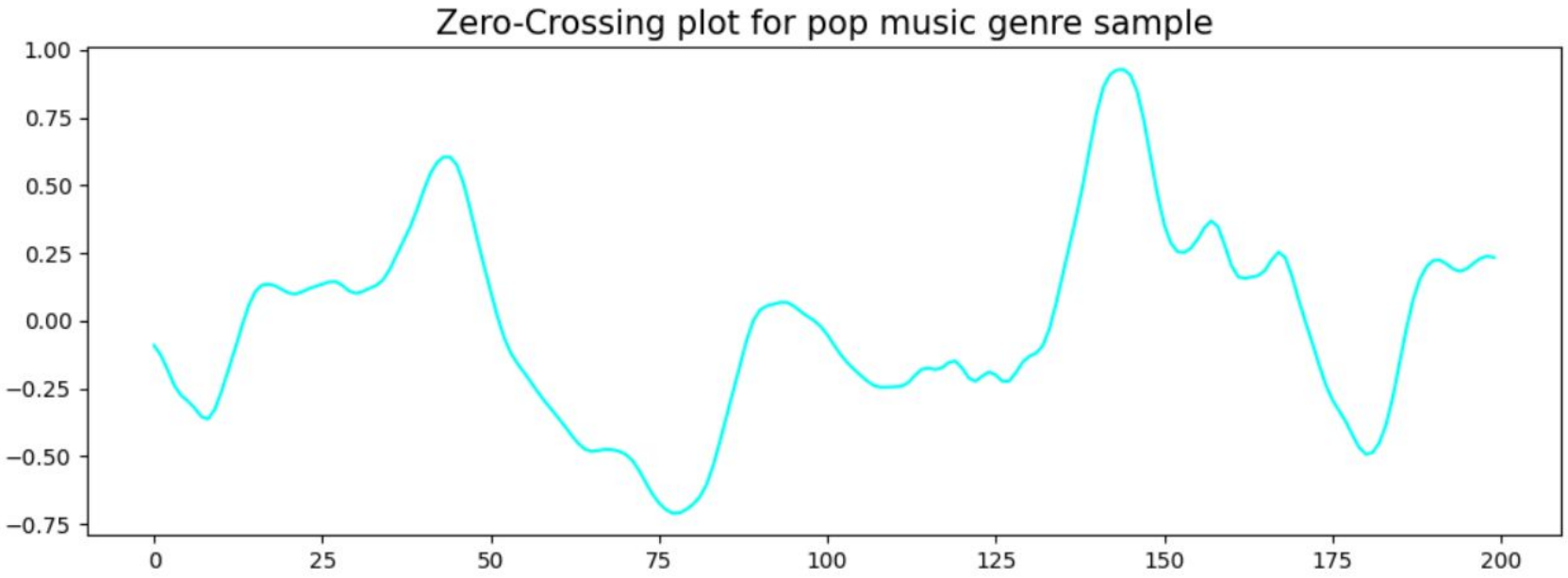
- Preparing the data and model by splitting into train and test sets.
- Training the model on **KNN classifier**.
- Plotting the **Confusion matrix** and calculating **precision, recall and f1-scores.**
- Testing on the test dataset.
- Preparing dataset for recommendation.
- Finding similarity using **cosine similarity** feature.
- **Recommending music** for each genre and checking if it is similar to the input music files of given genres using **euclidean distances**.

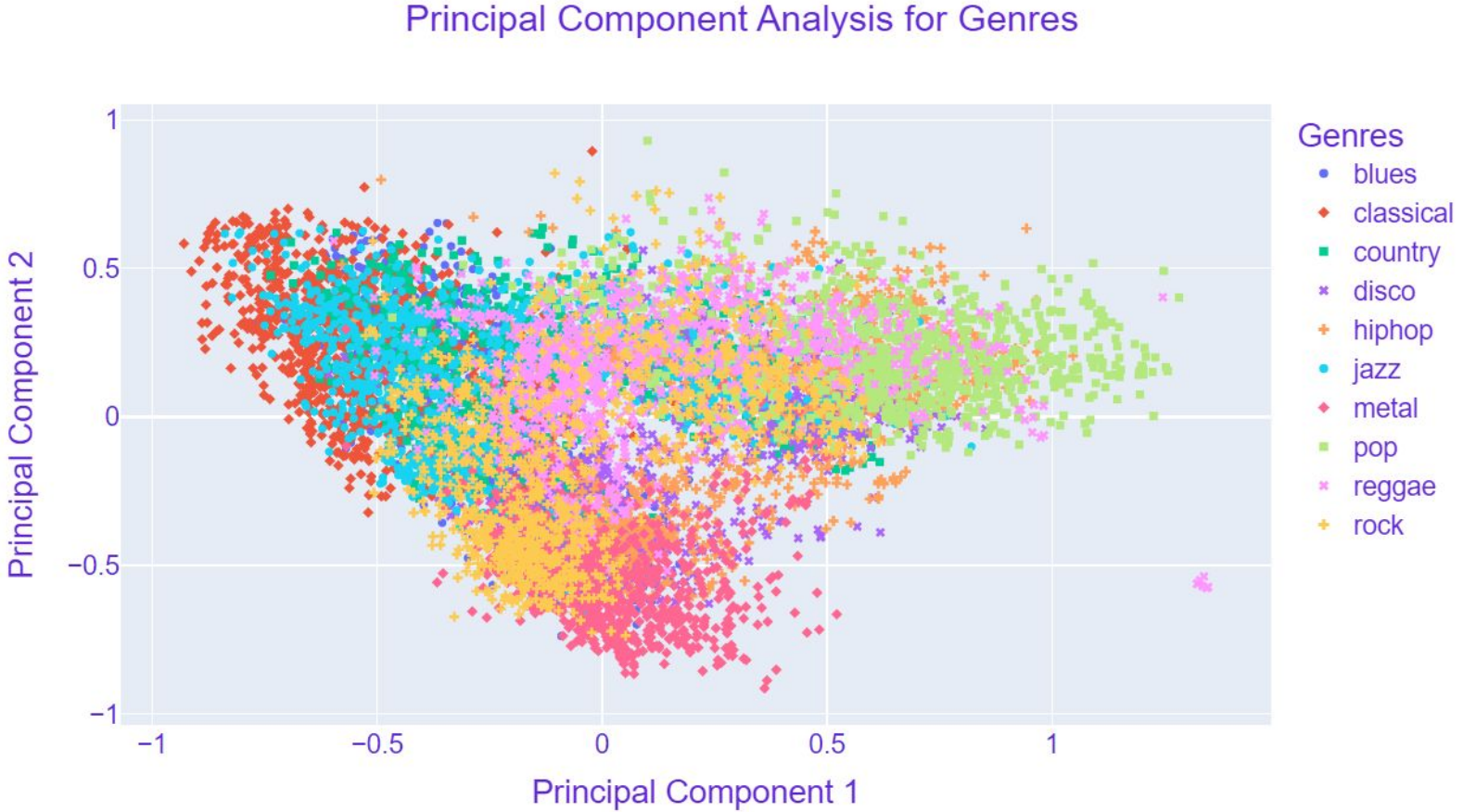

Waveplot for pop music genre sample
1.0
0.5
0.0
−0.5
−1.0
0:00
0:10
0:20
0:30
0:40
0:50
1:00
Time


Spectogram plot for pop music genre sample
Hz
20000
15000
10000
5000
0
0
5
10
15
20
25
Time
50
40
30
20
10
0
−10
−20

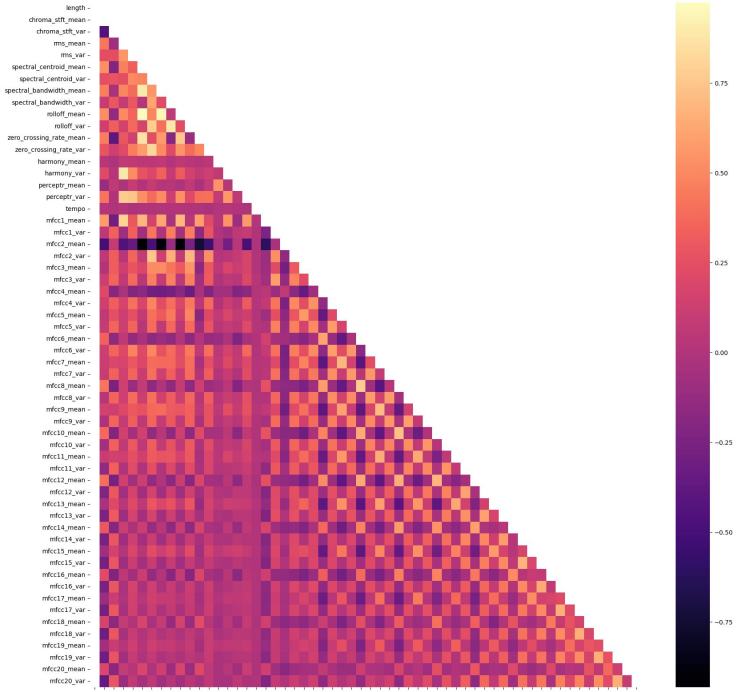

Spectral Rolloff plot for pop music genre sample
10000
7500
5000
2500
0
−2500
−5000
−7500
−10000
0.000
0.015
0.030
0.045
0.060
0.075
0.090
0.105
0.120
Time


Chroma plot for pop music genre sample
B
A
G
F
E
D
C
Pitch class
0
5
10
15
20
25
Time

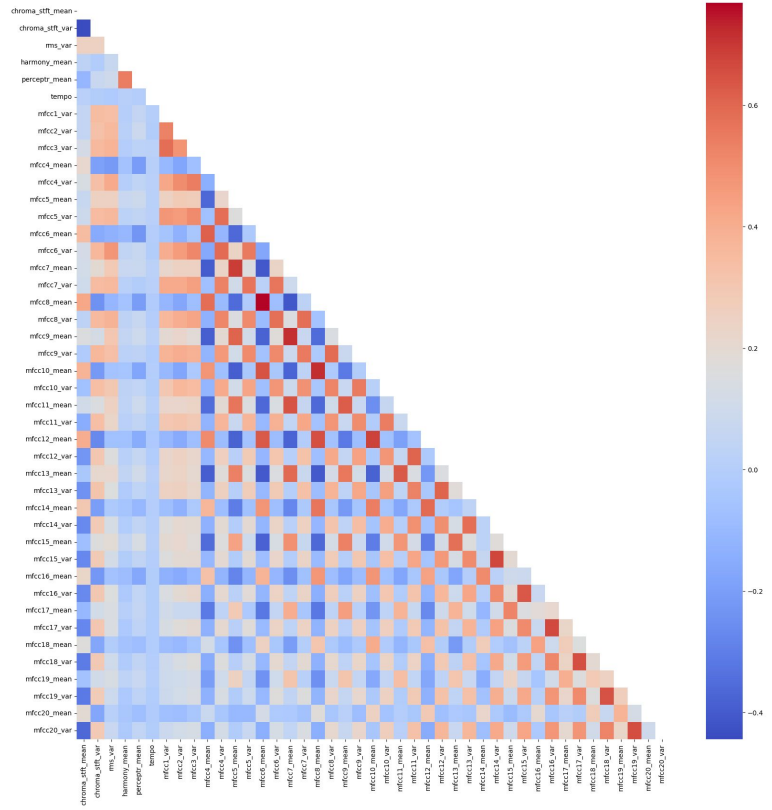

Zero-Crossing plot for pop music genre sample
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
0
25
50
75
100
125
150
175
200


Principal Component Analysis for Genres
Principal Component 2
Principal Component 1
1
0.5
0
−0.5
−1
−1
−0.5
0
0.5
1
Genres
blues
classical
country
disco
hiphop
jazz
metal
pop
reggae
rock

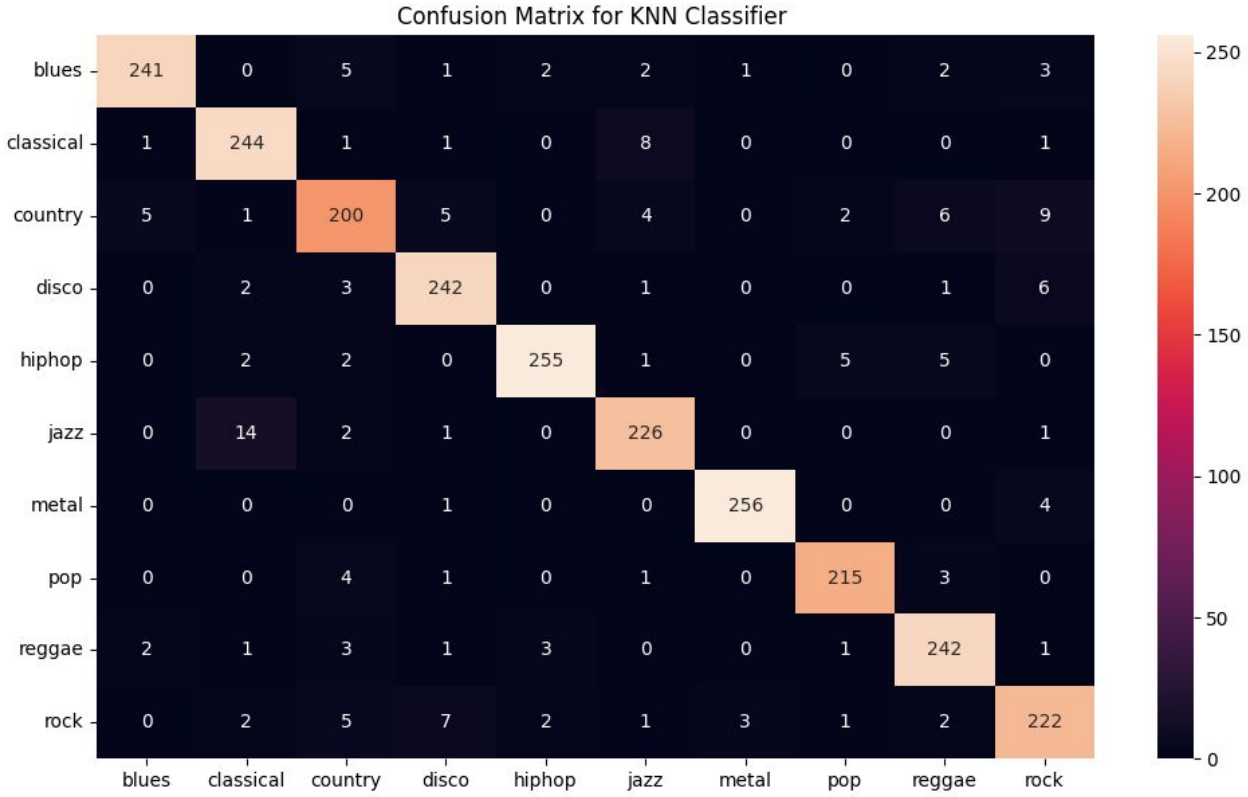
Correlation Heatmap (before VIF)



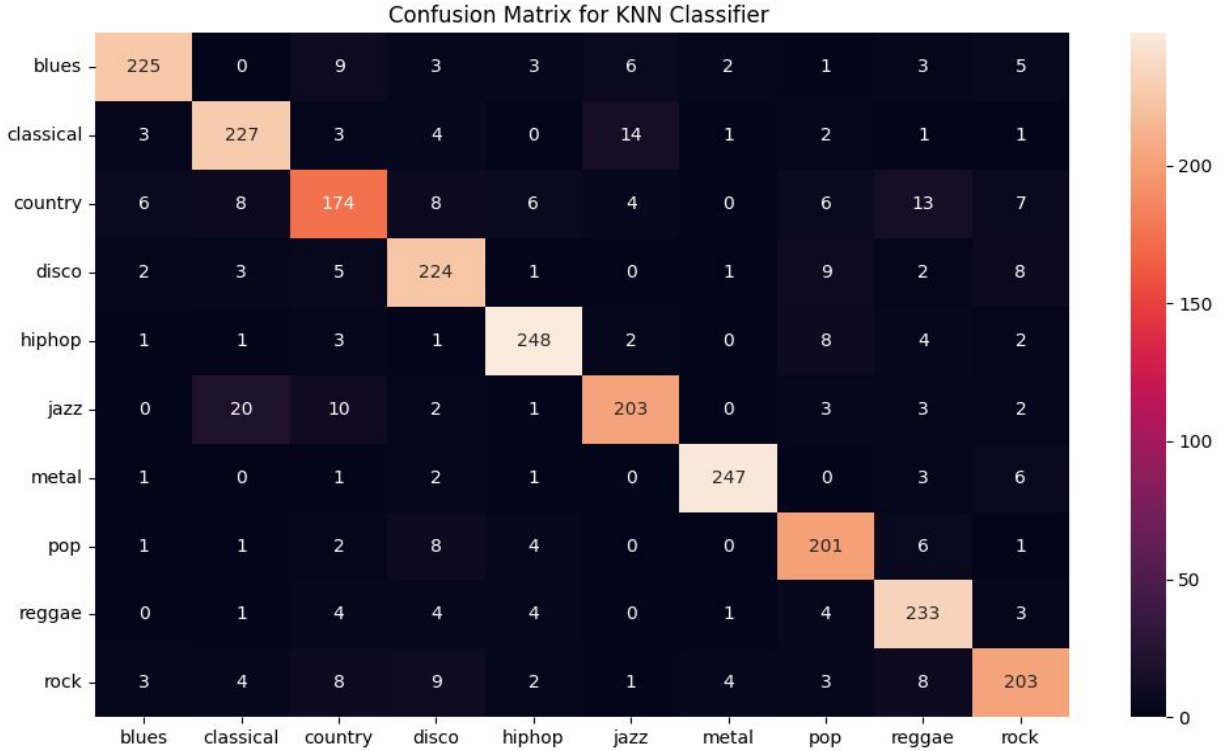
Correlation Heatmap (After VIF)



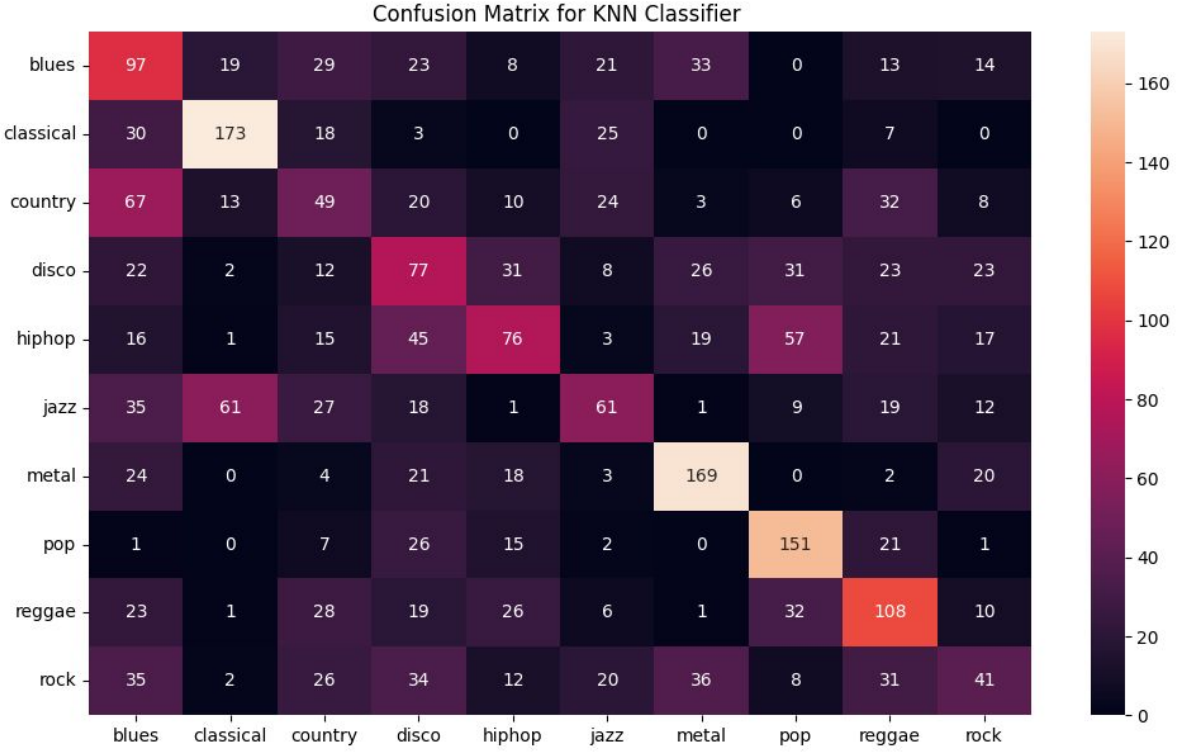
KNN on scaled data (scaled using MinMaxScaler)



KNN Classifier accuracy: 0.927654650645305



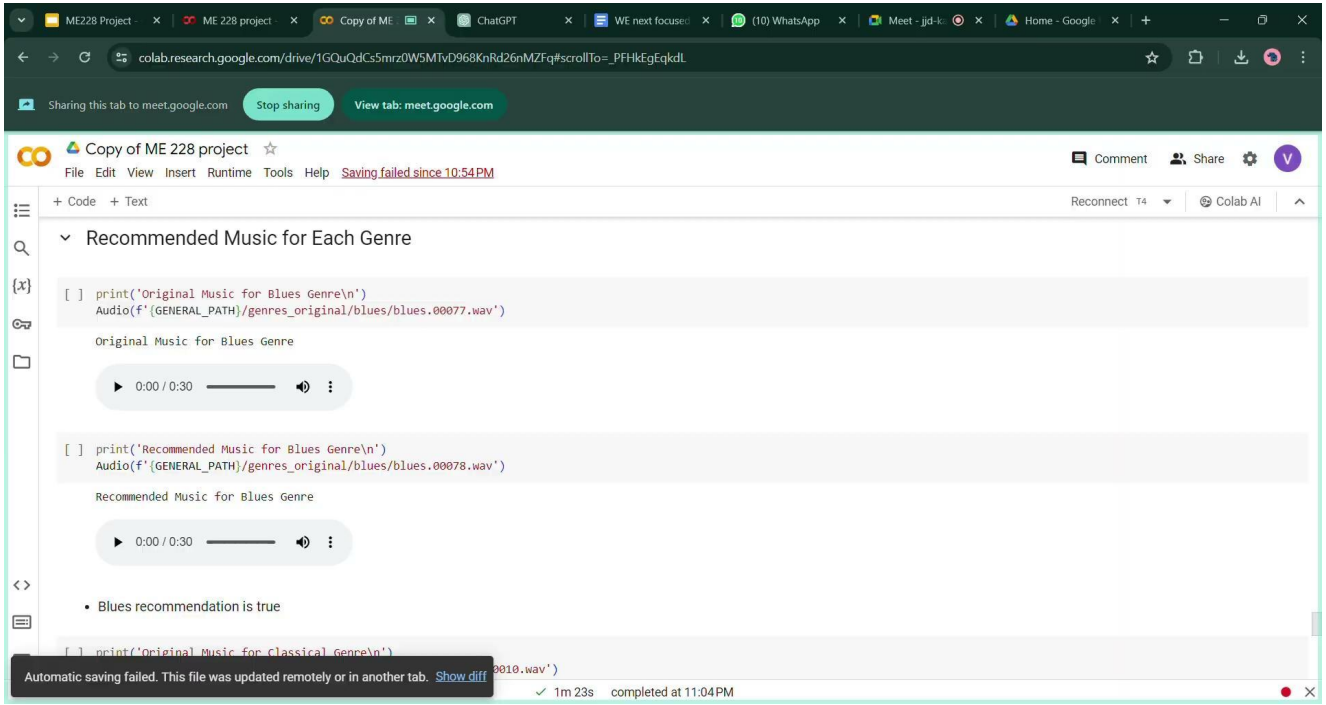
Data after VIF
KNN Classifier accuracy:
0.8703948375611926



PCA

KNN Classifier accuracy: 0.8703948375611926

Recommendation algorithm



https://drive.google.com/file/d/1XenX1bENdCfd4ro00O1Ub7AnE9l5g7m0/view?usp=sharing

Thank You!