MUSIC GENRE CLASSIFICATION

ARCHIT AGARWAL | DEEPAK REDDY T | PARAMESWARANATH VM

SARATHY VENKATAKRISHNAN | SHIVAM AGARWAL | VINAY P V

AGENDA

PROBLEM STATEMENT

DATA PROCESSING

DATA CLEAN UP

DATA ANALYTICS

DIMENSIONALITY REDUCTION

MODELING

RESULTS

CONCLUSION

PROBLEM STATEMENT

- Build a machine learning model capable of analysing a given audio sample and/or its metadata and classifying it into a given set of music genres (multi-label classification)

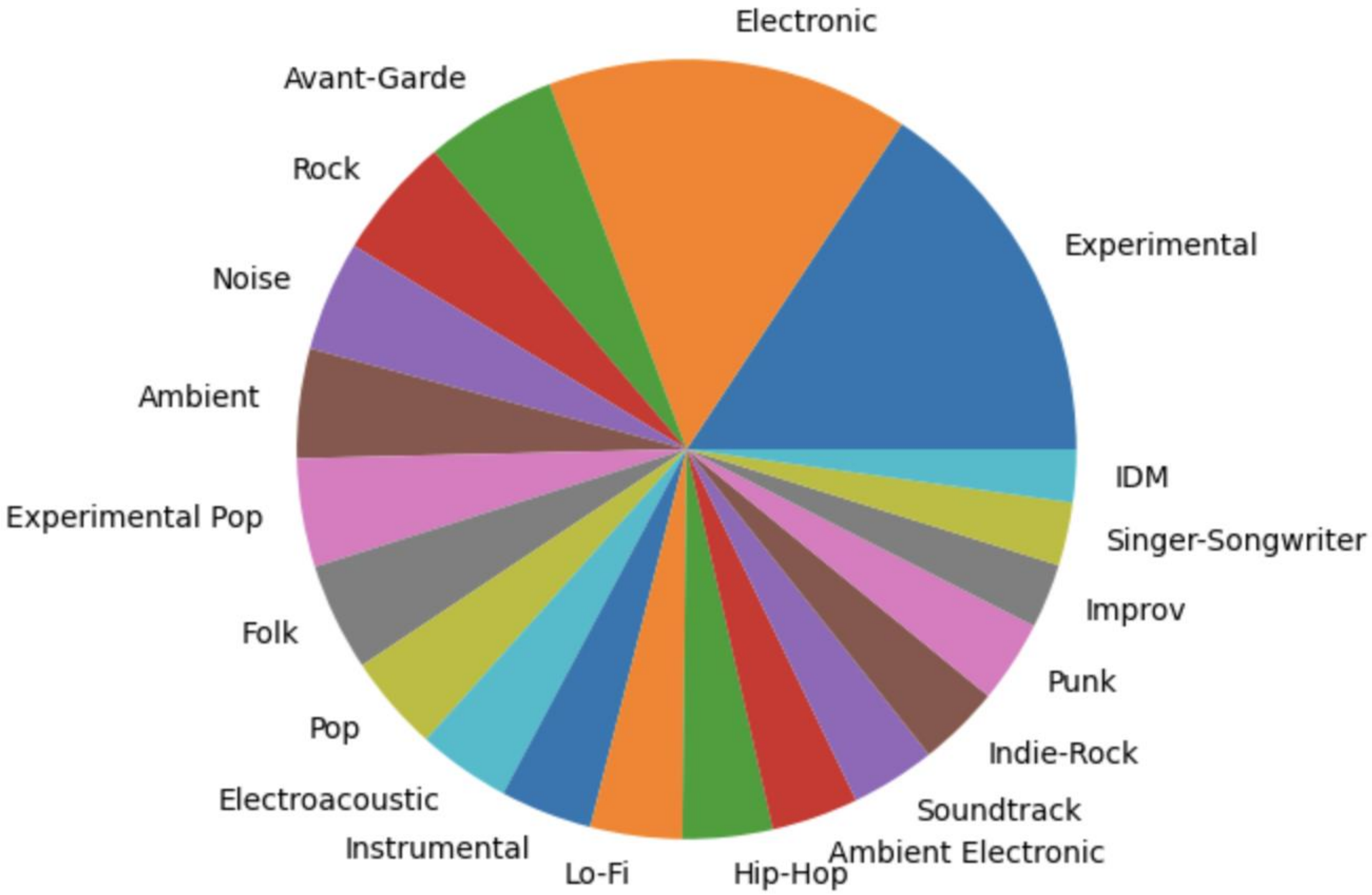
DATASET

- FMA (Free Music Archive, Small dataset): 8,000 tracks of 30 sec each, 7.2 GB

- FMA metadata contains ~520 spectral data parameters generated from music samples (via Python's **librosa** module)
- Also contains more human-understandable attributes (energy, tempo, danceability, etc.) generated from EchoNest's proprietary tools.

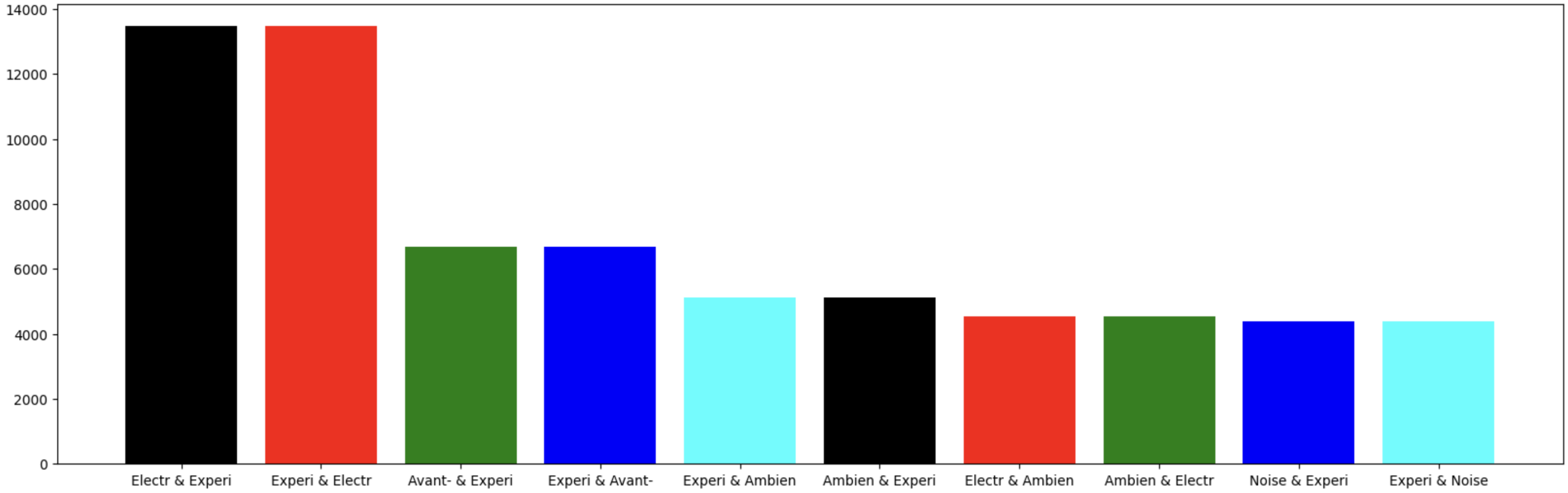
- Data cleanup and augmentation was required to map the numerical genre IDs (ground truth) listed for each track to the corresponding feature vector and filter out the tracks with invalid/non-existent genres.
- Resultant file contains 520 audio extracted features for 104343 datapoints

DATASET ANALYTICS



Track distribution according to genre

DATA ANALYTICS



Inter-genre correlation (pairs of genres most often linked with each other in terms of frequency of occurrence)

FEATURE EXTRACTION

Methods :
2 popular libraries to extract features from Audio are Echonest and Librosa. Librosa has more features compared from Echonest.

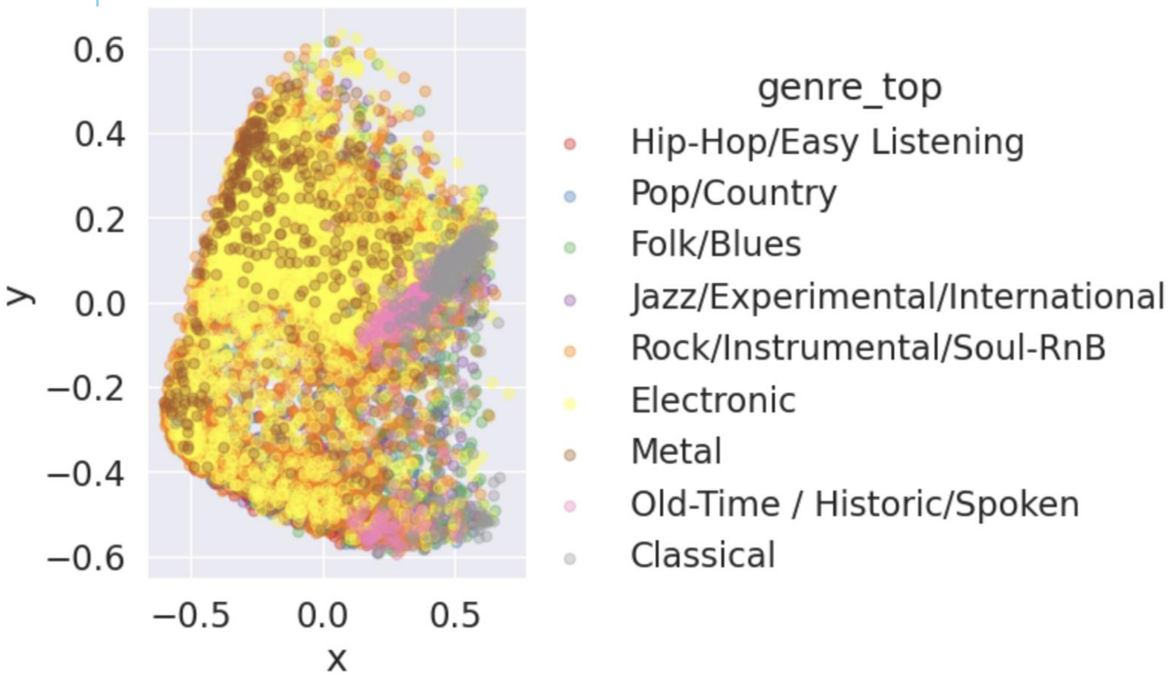
Echonest Features :
❑ acousticness
❑ danceability
❑ energy
❑ instrumentalness
❑ liveness
❑ speechiness
❑ tempo
❑ valence

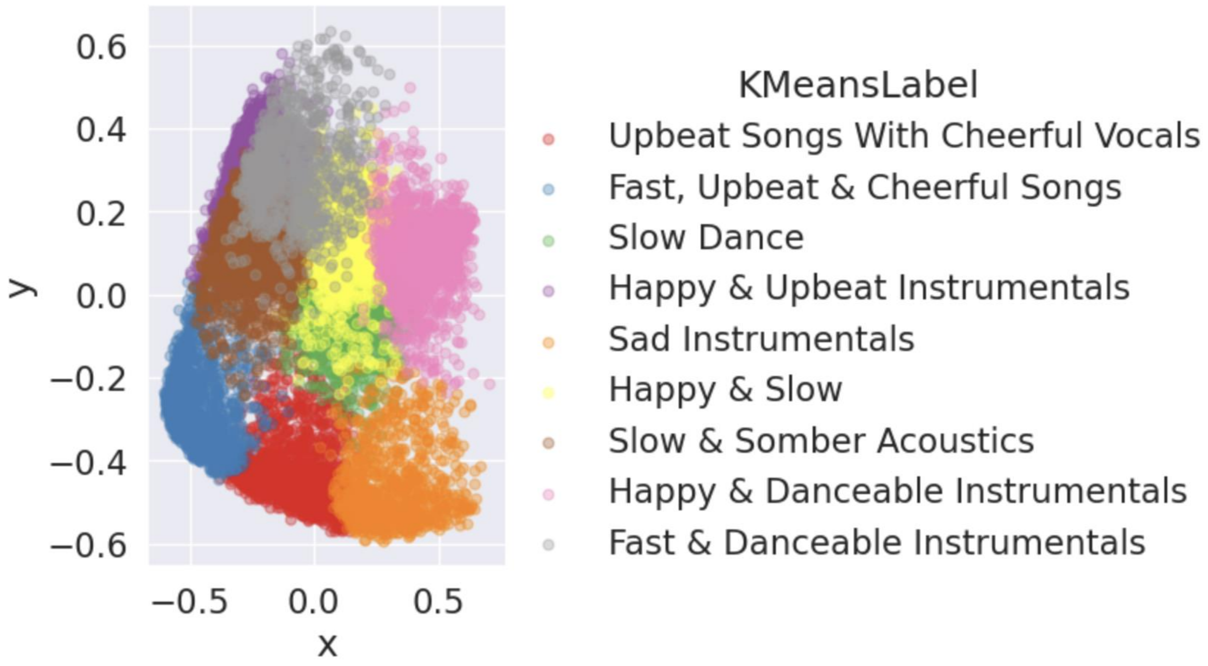
Librosa Features :
Spectral Features:
❑ Spectral Centroid:
❑ Spectral Contrast:
❑ Spectral Bandwidth:
Tonal Features:
❑ Chroma:
❑ CENS:
❑ Tonnetz:
Rhythmic Features:
❑ Tempo
❑ Beat Synchronous Chroma:
Root Mean Square (RMS) Energy
Zero Crossing Rate:

CLUSTERING ECHONEST

MUSIC GENRE

- Comparison of clustering efficiency using standard genres vs. those from K-Means clustering (pseudo-genres created by clubbing human-understandable EchoNest parameters)
- This indicates that traditional human genres are not very clear-cut and are more arbitrary when compared to genres divided by the actual audio characteristics.





MODELLING : FMA BASELINE CLASSIFIERS

Multi-class classification Baseline:

- Baseline results for classification models for FMA Dataset with features librosa and genre_top label.
- Provides a rough baseline of expected accuracy.

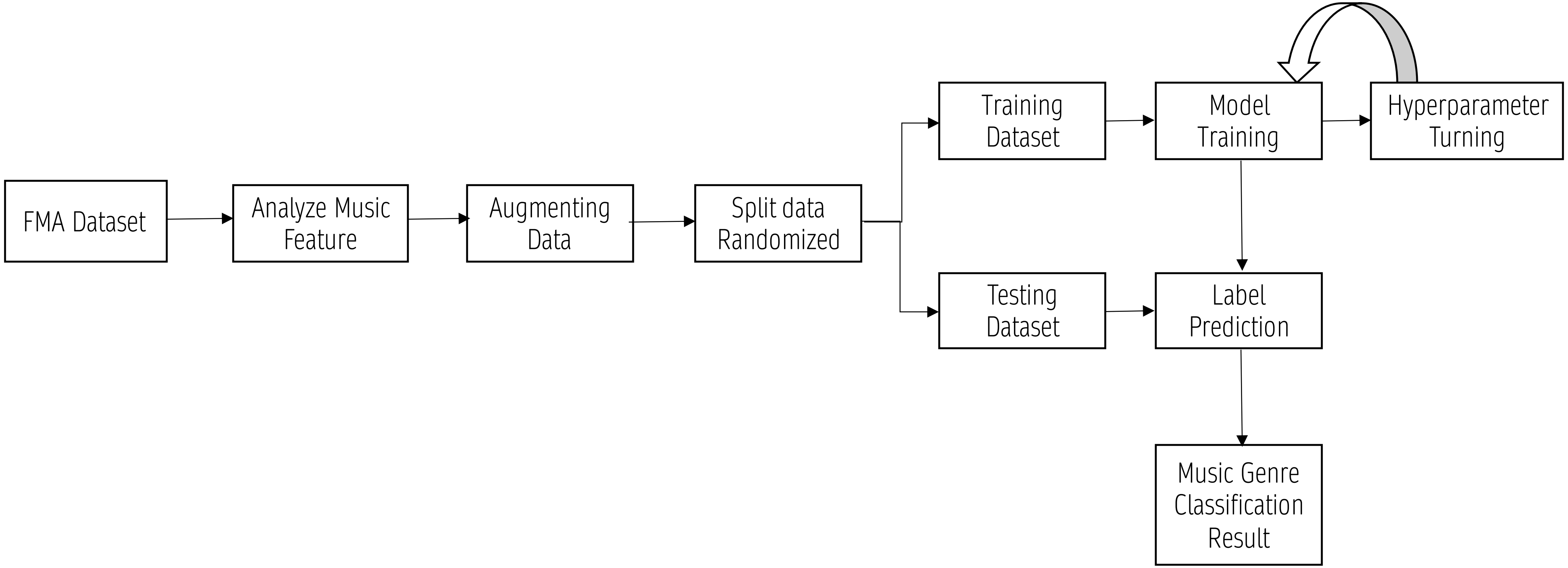
Features from *librosa* :

```
feature_sizes = dict(chroma_stft=12, chroma_cqt=12, chroma_cens=12,
                     tonnetz=6, mfcc=20, rmse=1, zcr=1,
                     spectral_centroid=1, spectral_bandwidth=1,
                     spectral_contrast=7, spectral_rolloff=1)
moments = ('mean', 'std', 'skew', 'kurtosis', 'median', 'min', 'max')
```

Classifier	Accuracy
LogisticRegression	60.71%
KNeighborsClassifier	51.77%
LinearSVC	58.73%
SVC (kernel= "rdf)	62.88%
SVC(kernel="poly")	61.95%
AdaBoost	43.06%
DecisionTreeClassifier	58.73%
RandomForestClassifier	47.30%
MLPClassifier ((100,) Hidden Layers)	57.02%
MLPClassifier ((200, 50) Hidden Layers)	57.33%
GaussianNB	9.91%

MODELLING : MULTI-LABEL CLASSIFICATION

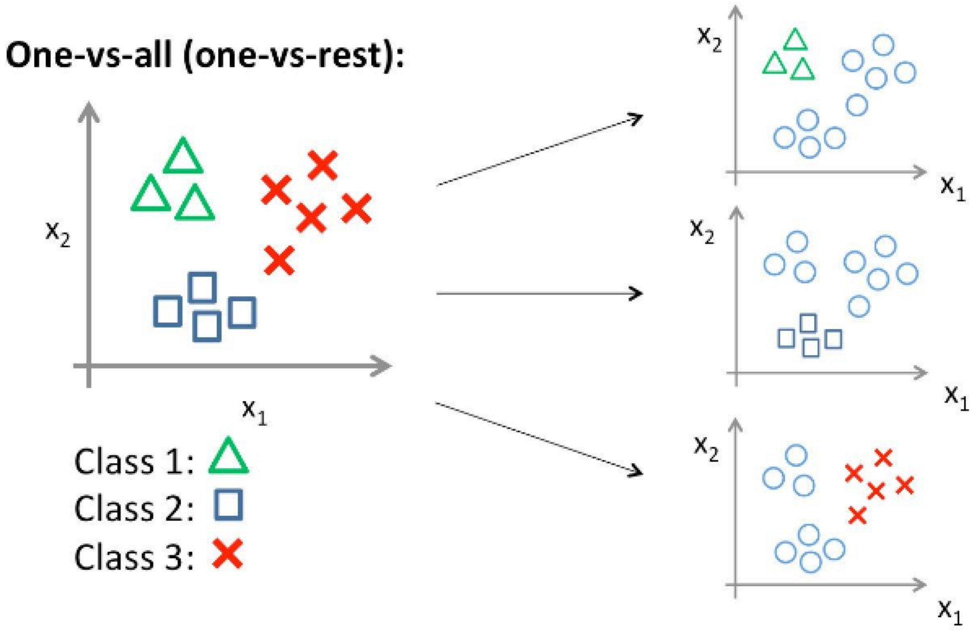
- Approach



MODELLING : MULTI-LABEL CLASSIFICATION

USING RANDOM FOREST – ONE VS REST CLASSIFIER

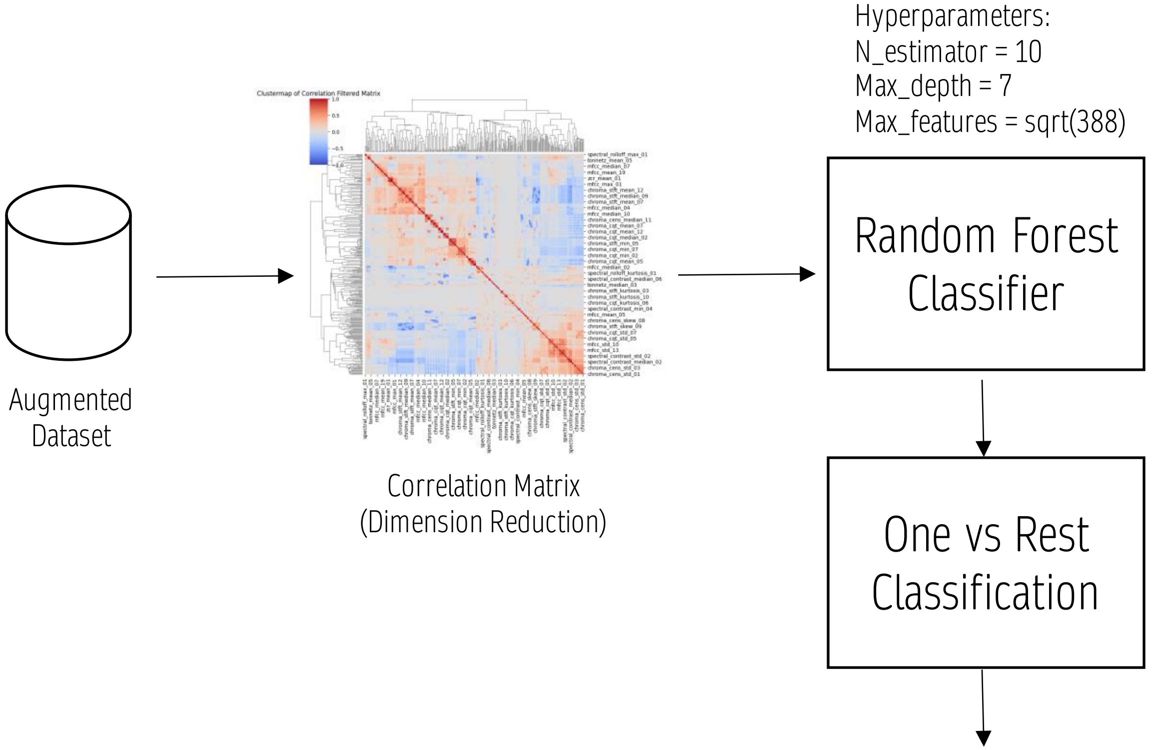
- Dataset contains a list of genre for each track id
- We are inclined to build a multi-label classifier to predict music genre
- Using one-vs-rest meta-estimator for multi-class classification, we can train one binary classifier per class and apply bagging to obtain required result
- Augmented dataset may still contain correlated features. Applying dimension reduction will further improve the result



MODELLING : MULTI-LABEL CLASSIFICATION

USING FEATURE CORRELATION + RANDOM FOREST CLASSIER

- Augmented data feature correlation is obtained using correlation matrix
- All the features having strong correlation (>=0.7) are dropped from dataset
- Dataset features
  - Before : (104343, 520)
  - After : (104343, 388)
- Resultant data is trained using a Random Forest Classifier with One vs Rest meta-estimator.
- Model hyper-parameters are turned to obtain best possible result

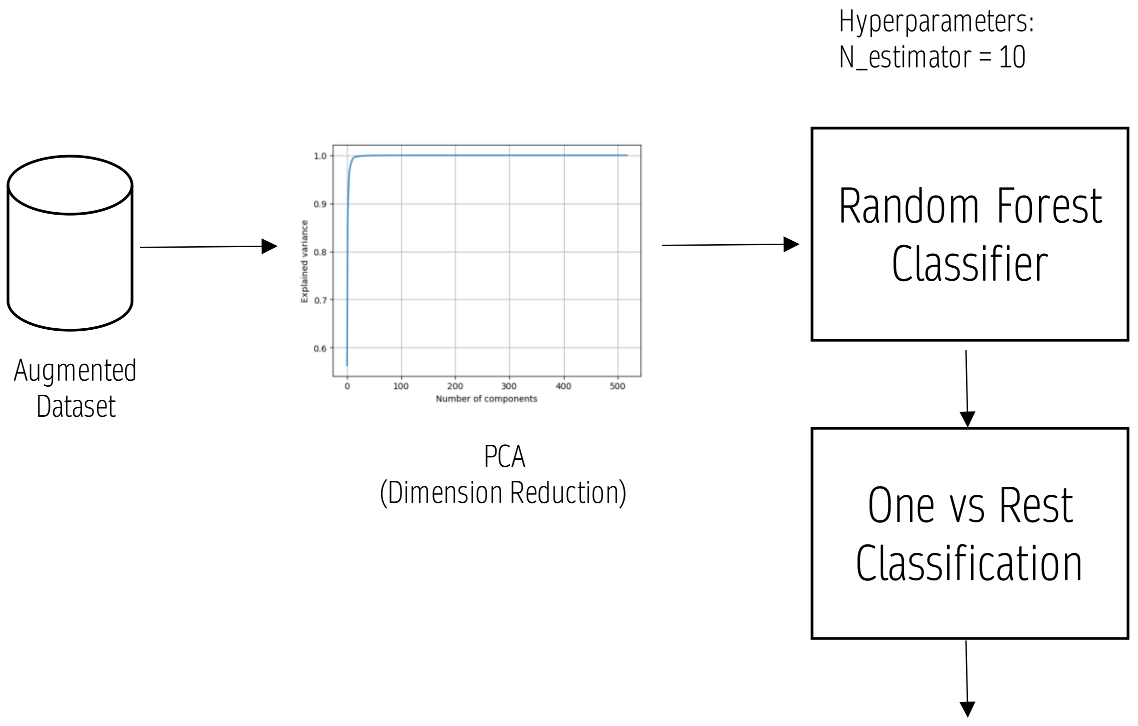



```
Precision: 0.7888674057426961
Recall: 0.6945196697211199
F1 Score: 0.7386931424070351
Hamming Loss: 0.14574754098360657
```

MODELLING : MULTI-LABEL CLASSIFICATION

USING PCA + RANDOM FOREST CLASSIER

- Principal Component Analysis (PCA) is a known technique used for dimensionality reduction.
- By applying PCA for all the features and drawing a graph, it is observed that the resultant graph is very steep, and features are highly correlated
- PCA with 99.9% variance resulted in reduction to just 33 features.
- Resultant data is trained using a Random Forest Classifier with One vs Rest meta-estimator.
- Model hyper-parameters are turned to obtain best possible result



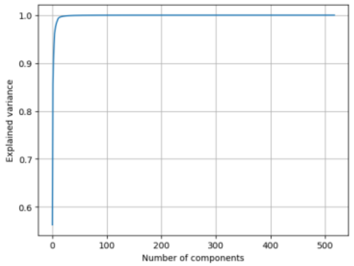

```
Precision: 0.7868247015374045
Recall: 0.6581978357227337
F1 Score: 0.716786489194907
Hamming Loss: 0.15428032786885246
```

MODELLING : MULTI-LABEL CLASSIFICATION

USING PCA + DEEP NEURAL NETWORK

- A Deep Neural Network (DNN) is trained with the same features as obtained earlier from PCA.
- This is to experiment the outcome of multi-label genre classification using neural network.
- Result obtained are recorded and observed that there is not much significant difference w.r.t F1 score.

Augmented Dataset



PCA
(Dimension Reduction)

```
model = keras.Sequential([
    Dense(256, activation='relu', input_dim=input_dim),
    Dropout(0.5),
    Dense(128, activation='relu'),
    Dropout(0.3),
    Dense(128, activation='relu'),
    Dropout(0.3),
    Dense(output_dim, activation='sigmoid')
])
```

Hyperparameters:
Batch_size = 32
Epochs = 20
Validation_split = 0.2

Deep Neural Network

```
Precision: 0.7868247015374045
Recall: 0.6581978357227337
F1 Score: 0.716786489194907
Hamming Loss: 0.15428032786885246
```

RESULTS & FUTURE WORK

- Observed results
  - Code is updated to github : https://github.com/vinay-pv/DA203o_Genre_Classification
  - Features_augmented is shared on google-drive : https://drive.google.com/file/d/1tANyScpS1TKt3gyqt4e2Q4LhkwMsDmka/view

	Precision	Recall	F1 Score	Hamming Loss	Coverage
Feature Corr + Random F.	0.7888	0.6945	0.7386	0.1457	59.824
PCA + Random F.	0.7868	0.6582	0.7167	0.1542	59.805
PCA + DNN	0.8034	0.6758	0.7173	0.1579	36.753

- Future Work
  - Use the audio files to extract features and test our trained model
  - Follow SOTA solutions available for genre classification and try to improve the same.

Q&A

THANK YOU

OPTIMAL GENRE SELECTION

- Using a K-Means clustering model, the number of optimal genres (which best separates this particular dataset) was obtained.
(elbow method (left) and silhouette score (right))