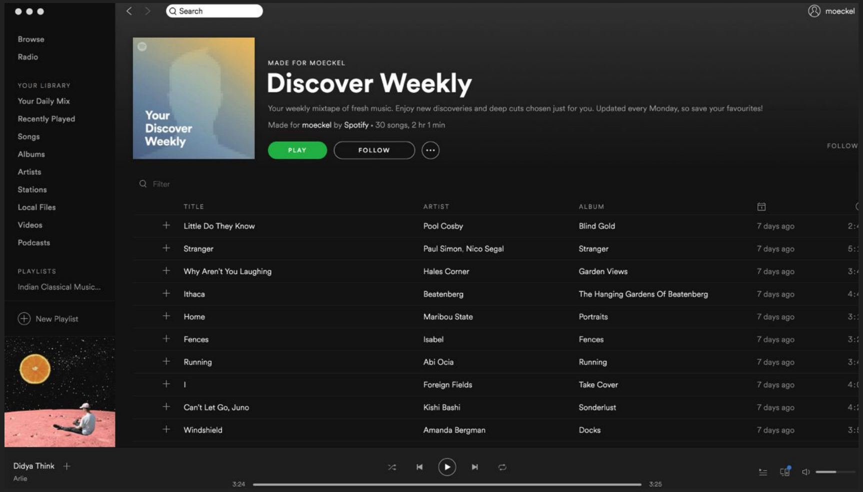
# Stacking Classifier Approach for Music Genre Classification

By Momoe Nomoto & Yuechen Wang

# Problem Description

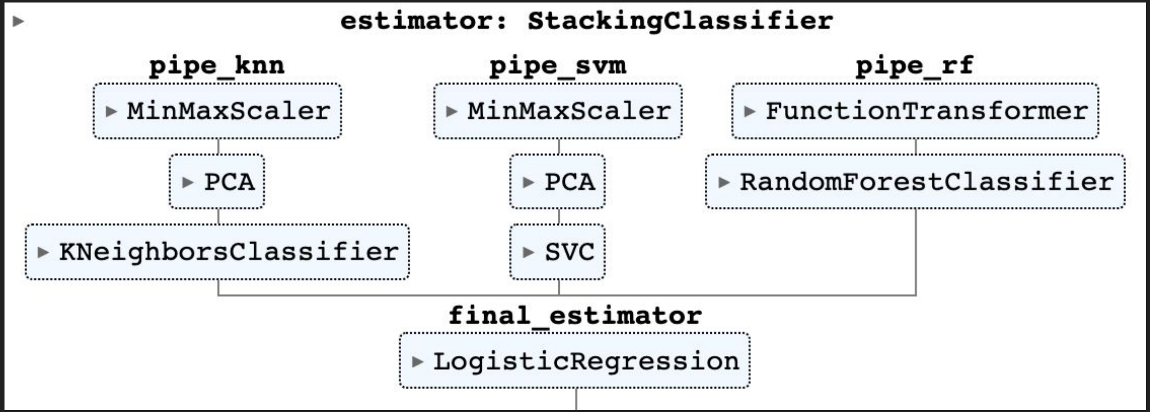
To manage large digital library for music, a genre classification system is needed. So far, a musician still needs to choose label for his/her song when he/she uploads the music to the platform, and many old songs do not have genre labels on them, which poses challenges for music categorization.




The Application of Music genre classification

- Automatic categorization of old music without genre labels
- Music recommendation

# Our Solution




3 Basic Models

KNN

Support Vector Machine

Random Forest

Stacking

Meta Model

Logistic Regression

# Data & Feature Extraction

## Dataset Source

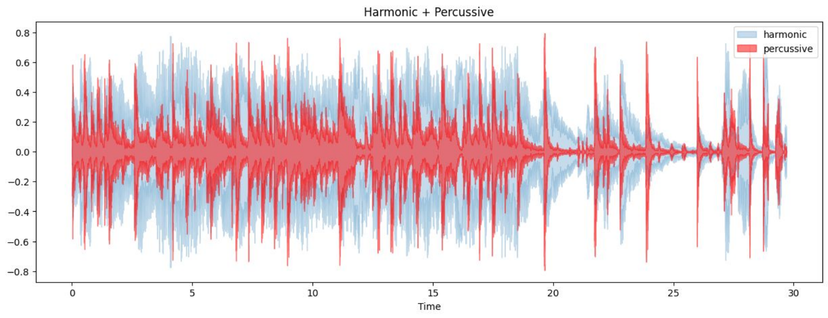
Spotify Playlist

## Extraction Package

Python Librosa

## Dataset Info

1120 songs
140 songs per genre




79 Features
8 Genres

```
Index(['song_id', 'mfcc_a_0', 'mfcc_a_1', 'mfcc_a_2', 'mfcc_a_3', 'mfcc_a_4',
       'mfcc_a_5', 'mfcc_a_6', 'mfcc_a_7', 'mfcc_a_8', 'mfcc_a_9', 'mfcc_a_10',
       'mfcc_a_11', 'mfcc_a_12', 'mfcc_std_0', 'mfcc_std_1', 'mfcc_std_2',
       'mfcc_std_3', 'mfcc_std_4', 'mfcc_std_5', 'mfcc_std_6', 'mfcc_std_7',
       'mfcc_std_8', 'mfcc_std_9', 'mfcc_std_10', 'mfcc_std_11', 'mfcc_std_12',
       'chroma_a_0', 'chroma_a_1', 'chroma_a_2', 'chroma_a_3', 'chroma_a_4',
       'chroma_a_5', 'chroma_a_6', 'chroma_a_7', 'chroma_a_8', 'chroma_a_9',
       'chroma_a_10', 'chroma_a_11', 'chroma_std_0', 'chroma_std_1',
       'chroma_std_2', 'chroma_std_3', 'chroma_std_4', 'chroma_std_5',
       'chroma_std_6', 'chroma_std_7', 'chroma_std_8', 'chroma_std_9',
       'chroma_std_10', 'chroma_std_11', 'rolloff_a', 'rolloff_std',
       'melspect_a', 'melspect_std', 'rmseP_a', 'rmseP_std', 'rmseH_a',
       'rmseH_std', 'centroid_a', 'centroid_std', 'bw_a', 'bw_std',
       'contrast_a', 'contrast_std', 'polyfeat_a', 'polyfeat_std', 'tonnetz_a',
       'tonnetz_std', 'zcr_a', 'zcr_std', 'onset_a', 'onset_std', 'bpm',
       'rmseP_skew', 'rmseP_kurtosis', 'rmseH_skew', 'rmseH_kurtosis',
       'beats_a', 'beats_std', 'genre'],
      dtype='object')
```

{Blues, Classical, Disco, Electronic, Hiphop, Jazz, Pop, and Rock}

# KNN

Step 1: Feature Scaling & PCA

Choose number of component: 25 components account for 88% of the variance

Step 2: Hyperparameter Tuning

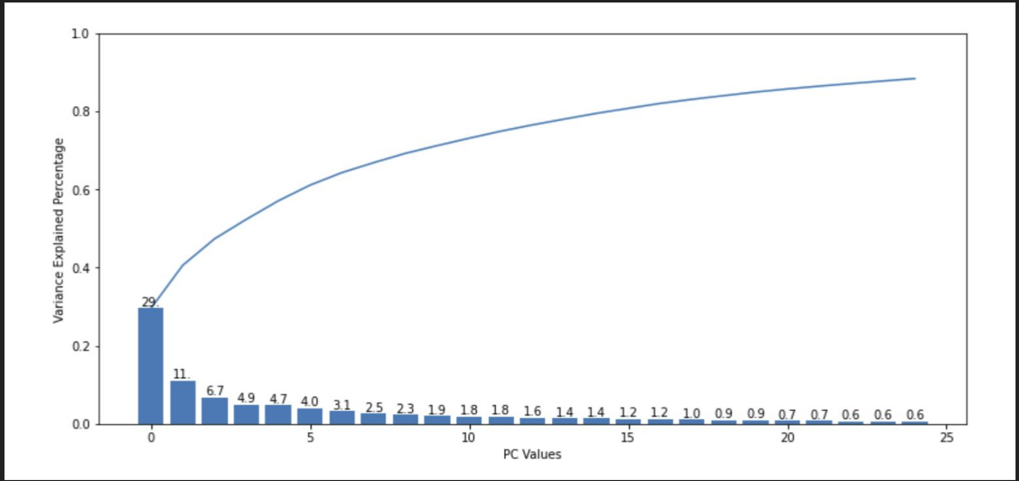
Leaf size: (1,50)

# Neighbors: (1,30)

Distance metrics:

Minkowski, Manhattan, Euclidean

Result: Test Accuracy = 0.632



# Support Vector Machine

Step 1: Feature Scaling & PCA

Choose number of component: 40 components account for 95% of the variance

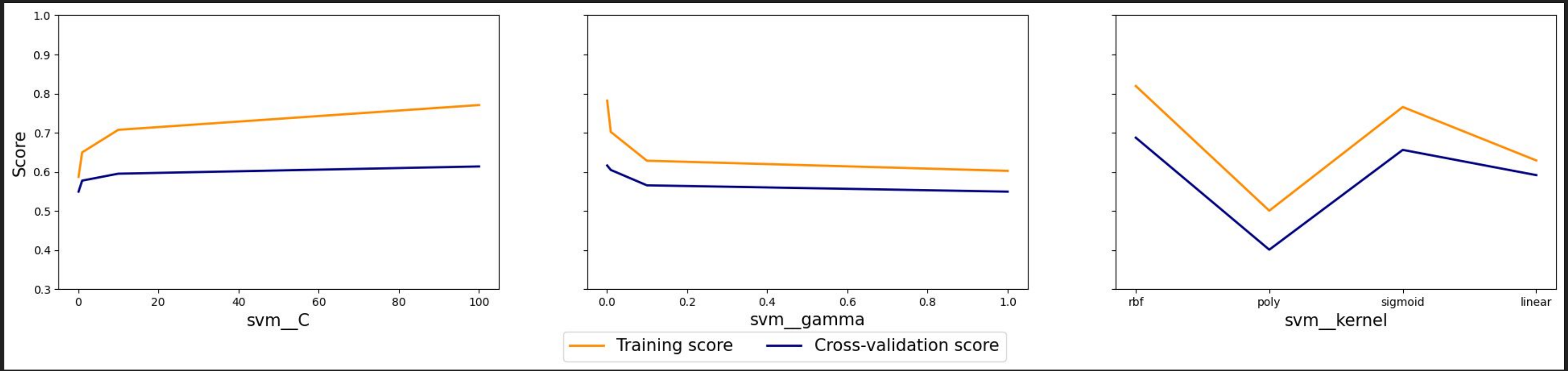
Step 2: Hyperparameter Tuning

C: [0.1, 1, 10, 100]

Type of Kernel: [RBF, Linear, Polynomial, Sigmoid]

Gamma: [1, 0.1, 0.01, 0.001]

Result:
Test Accuracy = 0.743



# Random Forest

Individual Decision Tree Classifier → Ensemble

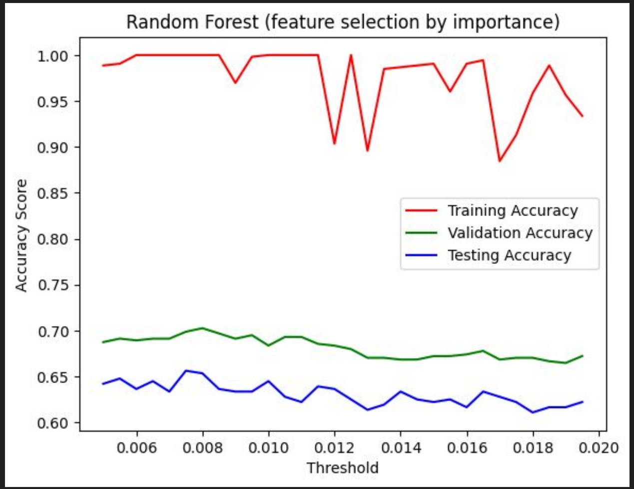
Test Accuracy: 0.477

- Bagging — Test Accuracy: 0.599
- Gradient Boosting — Test Accuracy: 0.639

Random Forest (Best Cross Validation Result: 0.639)

Feature Selection

- Importance Score √
- Correlation




# trees: 200
Minimal samples per leaf: 1
Max depth: 10

Threshold: 0.0075, Test Accuracy: 0.656

# Stacking Classifier

Final estimator: Logistic Regression

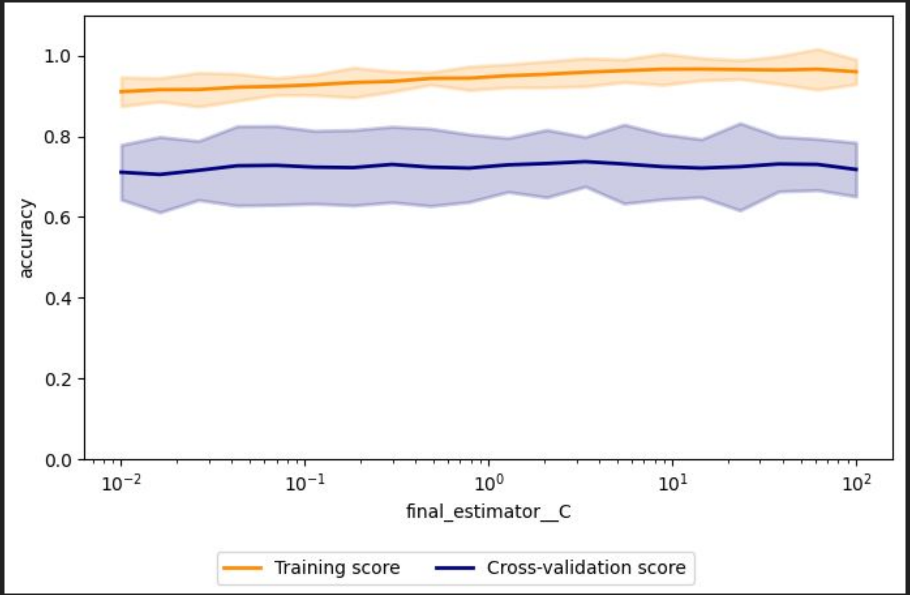
Simple & best for classification

Hyperparameter Tuning

5-fold stratified cross validation

⬇

Regularization C: np.logspace(-2, 2, 20)



# Test Results

Unseen dataset: 30 songs from each genre

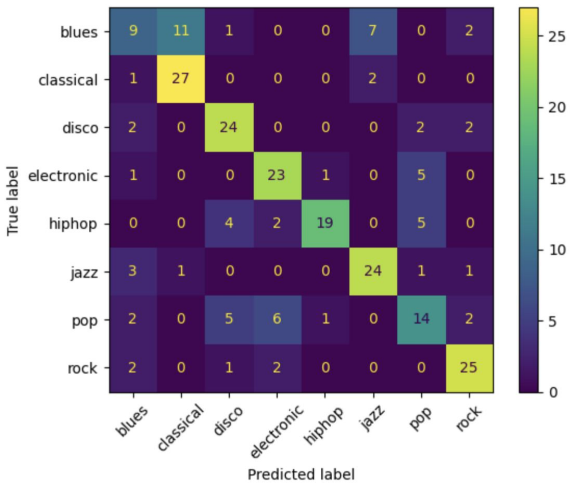
Stacking Classifier Test Accuracy: 0.69

In Comparison:

Base KNN Classifier Test Accuracy: 0.629

Base SVM Classifier Test Accuracy: 0.692

Base RF Classifier Test Accuracy: 0.671




| | Precision | Recall | F1-Score |
|---|---|---|---|
| Blues | 0.41 | 0.30 | 0.35 |
| Classical | 0.68 | 0.95 | 0.79 |
| Disco | 0.53 | 0.70 | 0.60 |
| Electronic | 0.77 | 0.75 | 0.76 |
| Hiphop | 0.69 | 0.63 | 0.66 |
| Jazz | 0.88 | 0.76 | 0.81 |
| Pop | 0.54 | 0.44 | 0.48 |
| Rock | 0.66 | 0.83 | 0.74 |
| Accuracy | | | 0.67 |
| Macro Avg | 0.65 | 0.67 | 0.65 |
| Weighted Avg | 0.67 | 0.67 | 0.66 |

# Observations

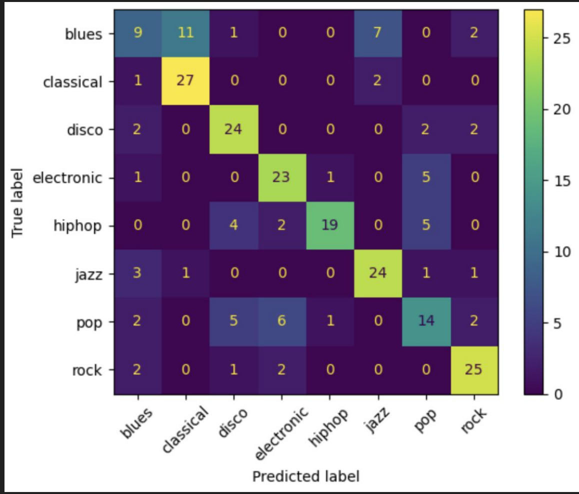
Blues + Pop: Low Recall

Blues often predicted as classical and jazz

Classical + Rock: High Recall

Both have distinctive audio features

Jazz: High Precision




| | Precision | Recall | F1-Score |
|---|---|---|---|
| Blues | 0.41 | 0.30 | 0.35 |
| Classical | 0.68 | 0.95 | 0.79 |
| Disco | 0.53 | 0.70 | 0.60 |
| Electronic | 0.77 | 0.75 | 0.76 |
| Hiphop | 0.69 | 0.63 | 0.66 |
| Jazz | 0.88 | 0.76 | 0.81 |
| Pop | 0.54 | 0.44 | 0.48 |
| Rock | 0.66 | 0.83 | 0.74 |
| Accuracy | | | 0.67 |
| Macro Avg | 0.65 | 0.67 | 0.65 |
| Weighted Avg | 0.67 | 0.67 | 0.66 |

# Model Limitation and Future Improvements

1. Larger Dataset
   a. More data from each genre
2. Train with music that are from one genre
   a. Reduce collinearity
3. More diverse set of genres
   a. Spotify API provides >100 genres
4. Multi-genre classification
   a. Ex. Pop 85%, Hiphop 10%, Rock 5%


```
{
    "genres": ["acoustic", "afrobeat", "alt-rock", "alternative", "ambient", "anime", "black-metal", "bluegrass", "blues", "bossanova", "brazil", "breakbeat", "british", "cantopop", "chicago-house", "children", "chill", "classical", "club", "comedy", "country", "dance", "dancehall", "death-metal", "deep-house", "detroit-techno", "disco", "disney", "drum-and-bass", "dub", "dubstep", "edm", "electro", "electronic", "emo", "folk", "forro", "french", "funk", "garage", "german", "gospel", "goth", "grindcore", "groove", "grunge", "guitar", "happy", "hard-rock", "hardcore", "hardstyle", "heavy-metal", "hip-hop", "holidays", "honky-tonk", "house", "idm", "indian", "indie", "indie-pop", "industrial", "iranian", "j-dance", "j-idol", "j-pop", "j-rock", "jazz", "k-pop", "kids", "latin", "latino", "malay", "mandopop", "metal", "metal-misc", "metalcore", "minimal-techno", "movies", "mpb", "new-age", "new-release", "opera", "pagode", "party", "philippines-opm", "piano", "pop", "pop-film", "post-dubstep", "power-pop", "progressive-house", "psych-rock", "punk", "punk-rock", "r-n-b", "rainy-day", "reggae", "reggaeton", "road-trip", "rock", "rock-n-roll", "rockabilly", "romance", "sad", "salsa", "samba", "sertanejo", "show-tunes", "singer-songwriter", "ska", "sleep", "songwriter", "soul", "soundtracks", "spanish", "study", "summer", "swedish", "synth-pop", "tango", "techno", "trance", "trip-hop", "turkish", "work-out", "world-music"]
}
```

# Thank you!