COMP90049 Report

Anonymous

1 Introduction

In this project, we predict music genres from their text, metadata and audio features by using supervised machine learning algorithms. Our dataset is derived from public datasets [1, 2], which include 8556 songs and their features & genres. Sklearn [3] is implemented to apply different learning models in order to be compared and get highest accuracy music genres prediction result. NLTK [4] TF-IDF [5] are implemented to handle the importance of words in titles and tags. Adaptive Boost and Bagging are implemented to improve classification accuracy.

From past researches, many models and features are experimented to have great effect on music genre classification. C. McKay and I. Fujinaga showed hierarchical classification of features selection can get improvement on performance [6]; A. Schindler and A. Rauber discovered the combination of loudness information and the distribution of segment length has highest 89% prediction rate[2]; J. Bergstra, N. Casagrande, D. Erhan et al. demonstrated AdaBoost is effective in music genre predictions [7]; T. George, C. Prry found timbrel texture, rhythmic content and pitch content of music signals have most accurate prediction in audio features [8].

2 Research Questions

1. Does Naive Bayes get more accurate prediction than K-NN models?

2. Which combination of features produces best prediction accuracy?

3 Preprocess & Explore

Before getting start, different features are handled to make them suitable for applying different kinds of models. 8556 songs are split into training set (7678 songs), validation set (450 songs) and test sets (428 songs). Training set and validation set are labeled with genres but test set are not.

3.1 Normalization & One hot

The scale of features influences the importance in models, the smallest value in audio features are less than 0.001 and the largest value is about 500, data needs to be normalized to ensure all features are equally important in prediction. The Key feature are track integers. One-hot is implemented to make it still meaningful after normalization process.

3.2 Drop high correlated features

From correlation table and heatmap, despite metadata features have low correlations between each other, many audio features are highly correlated. Only 18 out of 148 features have no correlation larger than 0.3. Other features are at least having high correlations to several features. Even 7 features in Figure 1 have larger than 0.5 correlations to all other features. High

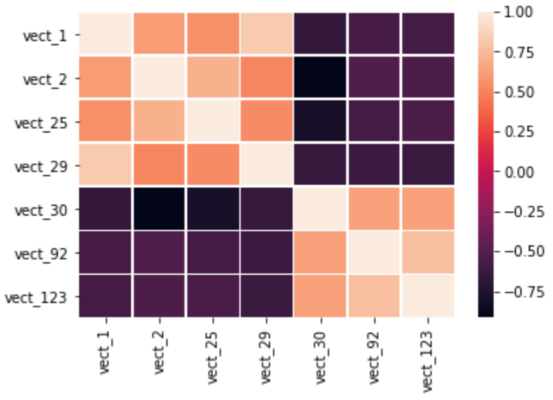


Figure 1: vect features which have > |0.5| correlations to all other features

correlation features influence the prediction of naive bayes model and the selection of random forest. In order to get accurate models, when implementing random forest features selection and naive bayes model, we need to drop high correlated audio features.

3.3 Random forest select feature

As L. Breiman proposed, random forest feature selection is highly accurate, generlize better and interpretable [9]. We use random forest to select 10 most important features when training models. By comparing the prediction accuracy of using important features with all features, we can distinguish whether top 10 features are more influential to the training model than other features.

3.4 Handle text data

Title and tags might be important features in genres classification. They contain direct descriptions of music contents, but they are stores as strings in the dataset. In order to process data in supervised learning models, we need to transform string features to numerical data, which can be easily analyzed. Therefore, after removing punctuation, NLTK [4] and TF-IDF [10] are implemented to get the importance of each words in all words for musics. During the Term Frequency (TF) process, words frequency are counted to measure their weights in importance. On the other side, Inverse Document Frequency (IDF) process decreases the weights of stop words and increases the weights of rarely occurring words in calculation. All words in titles and tags become features in the dataset separately and values are given to measure the importance of each word to each song.

3.5 Labels distribution deviation

By observing the distributions of all features in the dataset, I found training set and validation set have significant difference in label distribution. Figure 2 shows the label distributions of two datasets. Compared to training set, validation set labels are more balanced. Training set has higher proportions of *folk* and *classic pop & rock* songs, whereas validation set has higher proportion of *pop* songs. Moreover, the proportion of *jazz* is extremely low in training set. These differences will influence prediction accuracy. Training models are more likely to predict songs to higher proportional genres. Therefore, F1 is a more appropriate candidate metric to evaluate the performance of models.

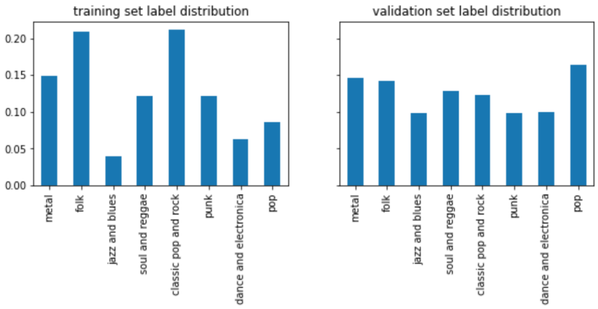


Figure 2: training set and validation set label distribution

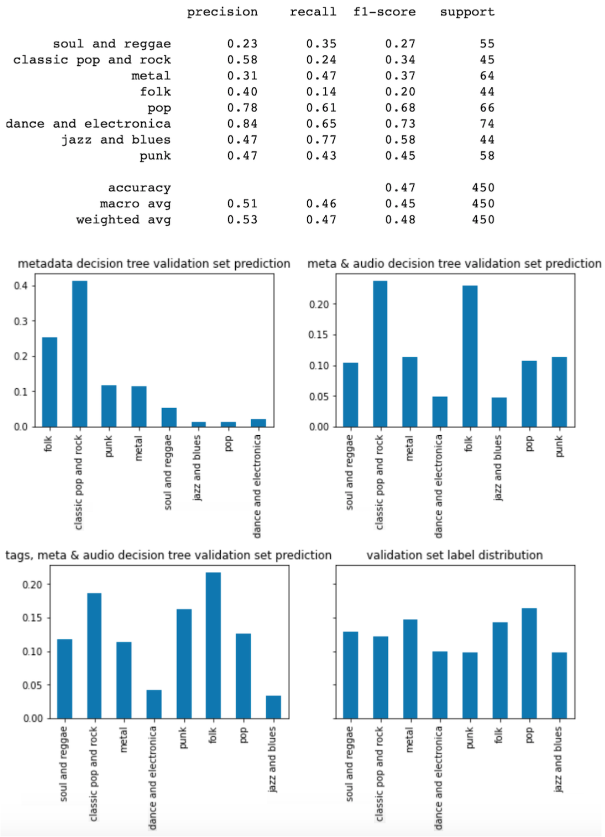
4 Models

4.1 Decision Tree

Decision Tree Classification makes decision on given features and deduces genre labels. Figure 3 shows the comparison of predicted and real label distributions. Obviously, the performance improves when more features implemented in the model, since more complicated decisions can be made based on multiple features in the dataset. The prediction f1 score is 0.23 when only using metadata features and it rises to 0.47 after all three kinds of features are given. The prediction accuracy also increased to 41.406%. Despite the accuracy and f1-score improve significantly, the values are still lower than 0.50. From the figure we can find the model predicts a lot of songs to *folk* and *classic pop & rock* whereas most of them are wrong: the recall is only 0.14 and 0.24. As section 2.6 mentioned, the training set has a large proportion of songs belongs to these two genres and the imbalance of datasets might causes decision tree model to predict most songs to them. Therefore this model is unsuitable for our datasets.

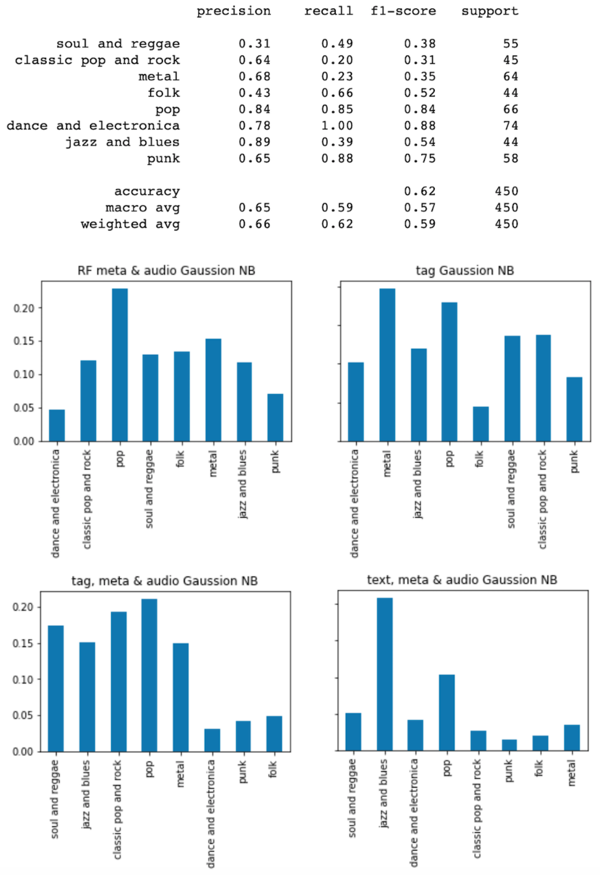
4.2 Gaussian Naive Bayes

Naive Bayes Classifier assumes all features are independent, so as section 2 mentioned, we use Random Forest [9] to select most important features in the dataset and make sure they have low correlations between each other. After transforming all titles and tags to numerical values, all features are continuous data and Gaussian Naive Bayes model is suitable for our



```
                      precision    recall  f1-score   support

       soul and reggae       0.23      0.35      0.27        55
  classic pop and rock       0.58      0.24      0.34        45
                 metal       0.31      0.47      0.37        64
                  folk       0.40      0.14      0.20        44
                   pop       0.78      0.61      0.68        66
dance and electronica       0.84      0.65      0.73        74
        jazz and blues       0.47      0.77      0.58        44
                  punk       0.47      0.43      0.45        58

              accuracy                           0.47       450
             macro avg       0.51      0.46      0.45       450
          weighted avg       0.53      0.47      0.48       450
```

Figure 3: Decision Tree best prediction details and all inout datasets label distributions



```
                      precision    recall  f1-score   support

       soul and reggae       0.31      0.49      0.38        55
  classic pop and rock       0.64      0.20      0.31        45
                 metal       0.68      0.23      0.35        64
                  folk       0.43      0.66      0.52        44
                   pop       0.84      0.85      0.84        66
dance and electronica       0.78      1.00      0.88        74
        jazz and blues       0.89      0.39      0.54        44
                  punk       0.65      0.88      0.75        58

              accuracy                           0.62       450
             macro avg       0.65      0.59      0.57       450
          weighted avg       0.66      0.62      0.59       450
```

Figure 4: Gaussian NB best prediction details and all input datasets label distributions

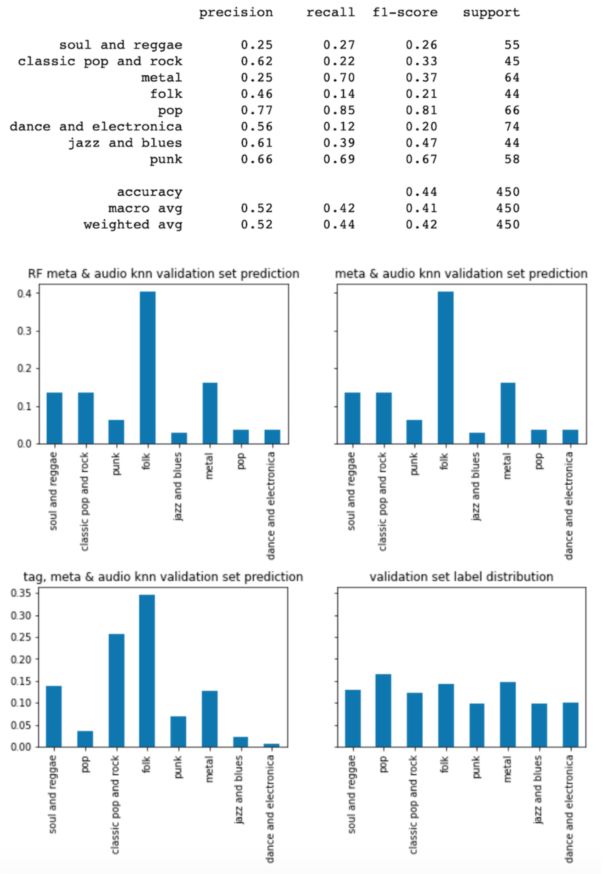
dataset. All different combinations of datasets using Gaussian Naive Bayes are shown in Figure 4. The label distribution is relatively balanced compared to Decision Tree. By using random forest selected metadata and audio features, the prediction f1-score of validation set is 0.50 and its test set prediction accuracy is 38.281%; using only tags feature the f1-score is 0.41. However, combining all tags feature with metadata and audio features gives prediction model a dramatic improvement. The f1-score reaches 0.62 and prediction accuracy reaches 55.468%. Observing the predictions of each genre in the labels, I find tags perform well in *pop* and *dance & electronica* songs. Their f1-scores are 0.84 and 0.88. The model finds characteristics of these two genres and predicts large proportions of them correctly. Nevertheless, if I concatenate titles feature to the training set, the f1-score drops to 0.40, which means titles feature cannot show the characteristics of *pop* music and it is not helpful in text features.

4.3 K-Nearest Neighbors

As I applied different combinations of features into K-NN model, the prediction results are underachieved. The highest validation set f1-score I can get is 0.44 and the best prediction accuracy is 35.156%. Only metadata features are effective in K-NN model and these features predict *pop* songs well (0.81) but *folk* songs (0.21) and *dance & electronica* songs (0.20) bad. The reason might be other features are not representitve for music genres in the model and K-NN cannot find neighbors for other features.

4.4 Random Forest

Random forest can also perform an ensemble learning method for classification [9]. Random decision forests correct for decision trees' habit of over-fitting to their training set [11]. As previous studies proved, my random forest model also outperforms decision tree model as Figure 6. By implementing all tags, metadata and audio features, the model also reaches 0.62 in f1-score and 55.468% in prediction accuracy, the same as Gaussian Naive Bayes Model. Tags

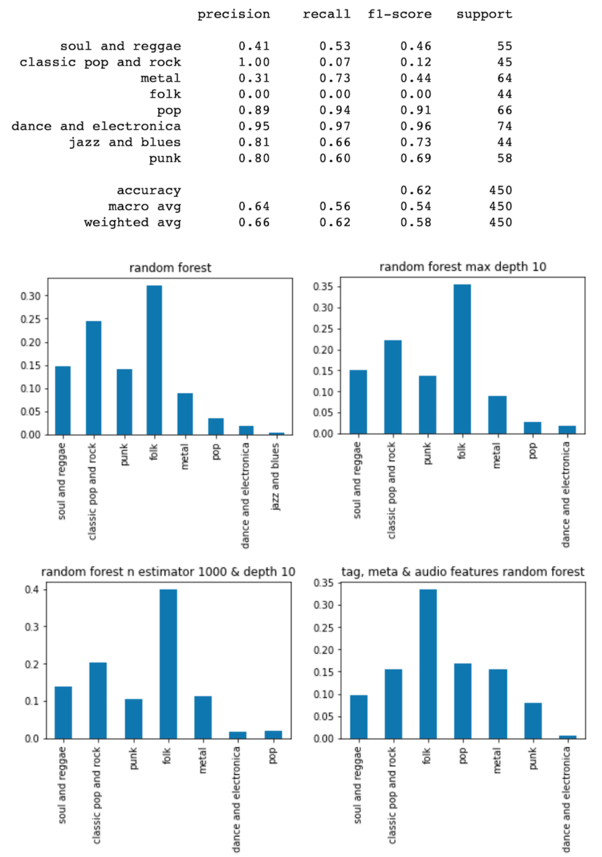


```
                      precision    recall  f1-score   support

      soul and reggae       0.25      0.27      0.26        55
  classic pop and rock       0.62      0.22      0.33        45
                 metal       0.25      0.70      0.37        64
                  folk       0.46      0.14      0.21        44
                   pop       0.77      0.85      0.81        66
dance and electronica       0.56      0.12      0.20        74
        jazz and blues       0.61      0.39      0.47        44
                  punk       0.66      0.69      0.67        58

              accuracy                           0.44       450
             macro avg       0.52      0.42      0.41       450
          weighted avg       0.52      0.44      0.42       450
```

Figure 5: K-NN best prediction details and all input datasets label distributions

feature also improves its performance in *pop* and *dance & electronica* musics. Moreover, the *punk* music also has higher f1-score (0.79). Similar to Decision Tree Model, the performances of *folk* and *classic pop & rock* are still bad (0.04 and 0.11) because high proportions of these genres in training set.

4.5 Support Vector Classifier



```
                      precision    recall  f1-score   support

      soul and reggae       0.41      0.53      0.46        55
  classic pop and rock       1.00      0.07      0.12        45
                 metal       0.31      0.73      0.44        64
                  folk       0.00      0.00      0.00        44
                   pop       0.89      0.94      0.91        66
dance and electronica       0.95      0.97      0.96        74
        jazz and blues       0.81      0.66      0.73        44
                  punk       0.80      0.60      0.69        58

              accuracy                           0.62       450
             macro avg       0.64      0.56      0.54       450
          weighted avg       0.66      0.62      0.58       450
```

Figure 6: Random Forest best prediction details and all input datasets label distributions

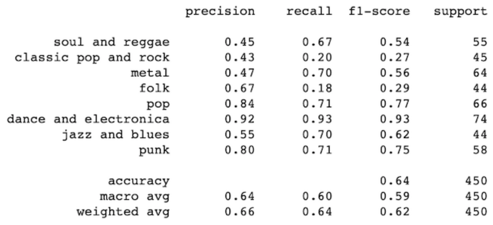
Support Vector Classifier maps their inputs into high-dimensional feature spaces so that examples of the separate categories are divided by clear gaps. The mapping can be divided to linear and non-linear according to whether categories can be separated by a linear line. Figure 7 shows the prediction results of both two types of mappings for training set. The f1-score for non-linear SVC implementing random forest selected tags, metadata and audio features reaches 0.62 and its accuracy is 48.437%. On the other side, linear SVC has higher f1-score 0.64 and accuracy 49.218%. Although SVC models have higher f1-scores than Gaussian Naive Bayes, their accuracy is much lower. By observing detailed prediction distributions, I find they cannot solve the problem of imbalance label distribution in training set for *folk* and *classic pop & rocks* musics. Their f1-score are still much lower than other genres (0.28 and 0.27).

4.6 Overview

The table below compares the highest validation set f1-score and test set prediction accuracy of each supervised learning model. Zero-R is considered as baseline model. here.

Models	f1-score	accuracy
Zero-R	0.02	12.22%
DT	0.47	41.416%
GNB	0.62	55.468%
KNN	0.44	35.156%
RF	0.62	55.468%
SVC	0.64	49.218%

As the table shows, Gaussian Naive Bayes, Random Forest and SVC models get high



```
                      precision    recall  f1-score   support

      soul and reggae       0.45      0.67      0.54        55
   classic pop and rock       0.43      0.20      0.27        45
                  metal       0.47      0.70      0.56        64
                   folk       0.67      0.18      0.29        44
                    pop       0.84      0.71      0.77        66
 dance and electronica       0.92      0.93      0.93        74
         jazz and blues       0.55      0.70      0.62        44
                   punk       0.80      0.71      0.75        58

               accuracy                           0.64       450
              macro avg       0.64      0.60      0.59       450
           weighted avg       0.66      0.64      0.62       450
```

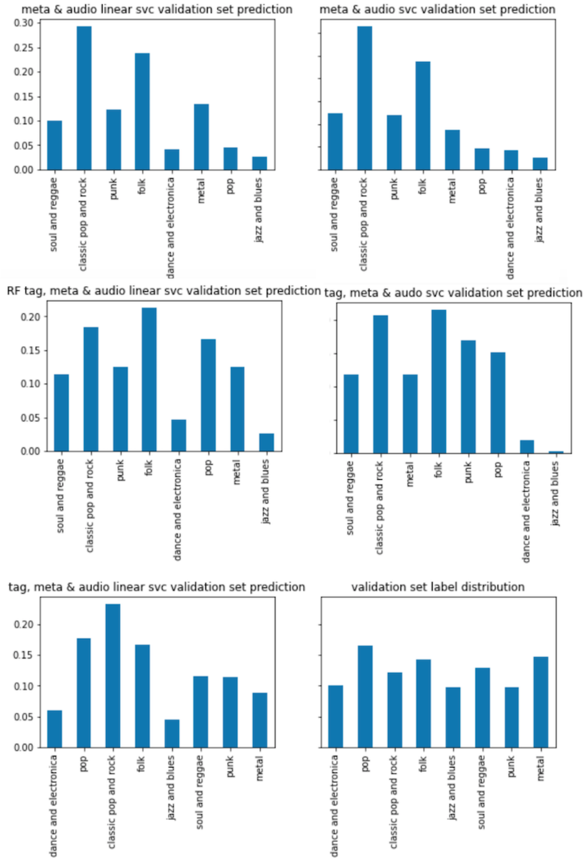


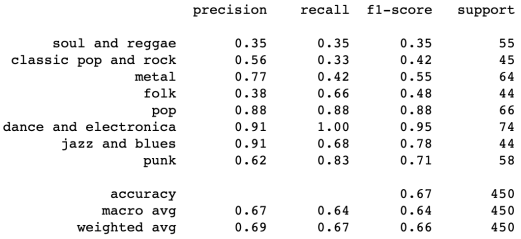
Figure 7: SVC best prediction details and all input datasets label distributions

f1-scores since tags feature improves their performance on *pop* and *dance & electronica* musics. Gaussian Naive Bayes and Random Forest also have the highest prediction accuracy since its prediction is more balanced and has less influence caused by the imbalance distribution of training set.

5 AdaBoost & Bagging

After attempting supervised learning models to train our dataset, boost and bagging can be used to improve performance of models. Boost and Bagging are ensemble meta-algorithms: Boost can be used to reduce the bias and variance of learning; Bagging can be used to improve stability and accuracy of learning. J. Bergstra, N. Casagrande, D. Erhan et al. also shows Adaptive Boost (AdaBoost) is an effective algorithm to improve music genre classification [7]. Gaussian Naive Bayes is the best model which has best f1-score and accuracy in section 3 and it is effective in Boost and Bagging. Therefore, I choose to boost and bagging this model in order to reach a higher performance.

By implementing adaptive boost and bagging to the model, the prediction results are shown in Figure 8. The f1-score increases to 0.67 and accuracy increase to 60.156%. Compared scores of each genre labels, previous low *metal* and *jazz & blue* musics are improved significantly about 20%. Many other music types including previously high genre - *dance & electronica* also have improvements. Decreased scores in *soul & reggae*, *folk* and *punk* are small compared to the improvement performance in total.



```
                      precision    recall  f1-score   support

      soul and reggae       0.35      0.35      0.35        55
   classic pop and rock       0.56      0.33      0.42        45
                  metal       0.77      0.42      0.55        64
                   folk       0.38      0.66      0.48        44
                    pop       0.88      0.88      0.88        66
 dance and electronica       0.91      1.00      0.95        74
         jazz and blues       0.91      0.68      0.78        44
                   punk       0.62      0.83      0.71        58

               accuracy                           0.67       450
              macro avg       0.67      0.64      0.64       450
           weighted avg       0.69      0.67      0.66       450
```

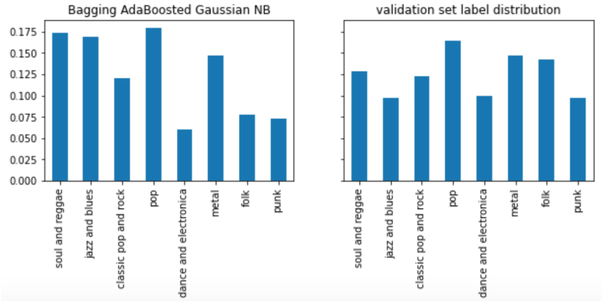


Figure 8: prediction details of Bagging AdaBoosted Gaussian NB and comparison to validation set distribution

6 Error Analysis

Overfitting has been problem during the K-NN and MLP part. The large difference in training scores and validation scores shows the variance exists in models. Dataset can be trained before implementing different models in order to decrease the variance. Moreover, as I mentioned above, f1-scores are much higher than accuracy in models because there exists evaluation variance. In order to reduce its influence, training and validation set can be crossed to get more balanced distribution.

7 Conclusions

Unfortunately, regarding to the performance and time limitations, I have to give up many features and algorithms I would like to attempt. Individual feature can be inspected intensively to find detailed correlations and new features can also be created by combining several related features. Besides, Linear SVC Model also predicts high f1-scores and accuracy result, it could be improved through adaptive boost and bagging but the time complexity is much higher than I expected.

Through all the models and algorithms I attempted, the result shows Bagging Adaptive Boosted Gaussian Naive Bayes Model predicts most accurate music genre classification among all the models I discussed. The features it used are the combination of TD-IDF handled tags feature, all metadata features and low correlated audio features. Moreover, I also find tags feature is good at handling pop and dance & electronica musics; metadata is good at handling pop songs. The theories proposed by J. Bergstra et al. [7], which shows adaptive boost improve music genre classification, is also proved by the result. Random forest also shows loudness and duration are two most important features in metadata as A. Schindler and A. Rauber [2] wrote in their paper.

References

[1] T. Bertin-Mahieux, D. P. Ellis, B. Whitman, and P. Lamere, “The million song dataset,” *Proceedings of the 12th International Society for Music Information Retrieval Conference, ISMIR 2011*, pp. 591–596, 2011.

[2] A. Schindler and A. Rauber, “Capturing the temporal domain in echonest features for improved classification effectiveness,” *Proceedings of the 10th InternationalWorkshop on Adaptive Multimedia Retrieval (AMR)*, vol. 8382, pp. 214–227, 2014, ISSN: 16113349. DOI: 10 . 1007 / 978 - 3 - 319 - 12093-5{_}13.

[3] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[4] E. Loper and S. Bird, “NLTK: the Natural Language Toolkit,” *CoRR*, vol. cs.CL/0205, 2002. DOI: 10.3115/1118108.1118117.

[5] C. Sammut, G. Webb, and E. I, *Bias Variance Decomposition.* Springer, 2011, pp. 100–101.

[6] C. Mckay and I. Fujinaga, “Automatic Genre Classification Using High-Level Musical Feature Sets,” *Proceedings of the 5th International Conference on Music Information Retrieval*, pp. 525–530, 2004.

[7] J. Bergstra, N. Casagrande, D. Erhan, D. Eck, and B. Kégl, “Aggregate features and ADABOOST for music classification,” *Machine Learning*, vol. 65, no. 2-3, pp. 473–484, 2006, ISSN: 08856125. DOI: 10.1007/s10994-006-9019-7.

[8] T. George and C. Perry, “Musical Genre Classification of Audio Signals,” *IEEE Transactions on Speech and Audio Processing*, vol. 10e, no. 5, pp. 293–302, 2002, ISSN: 22195491.

[9] L. Breiman, “Random Forests,” *Machine Learning*, vol. 45, pp. 5–32, 2001. DOI: https : / / doi . org / 10 . 1023 / A : 1010933404324.

[10] "TF–IDF," in *Encyclopedia of Machine Learning*, C. Sammut and G. I. Webb, Eds., Boston, MA: Springer US, 2010, pp. 986–987, ISBN: 978-0-387-30164-8. DOI: 10.1007/978-0-387-30164-8{_}832. [Online]. Available: https://doi.org/10.1007/978-0-387-30164-8_832.

[11] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, 2nd ed. Springer, 2008, pp. 587–588, ISBN: 0-387-95284-5.