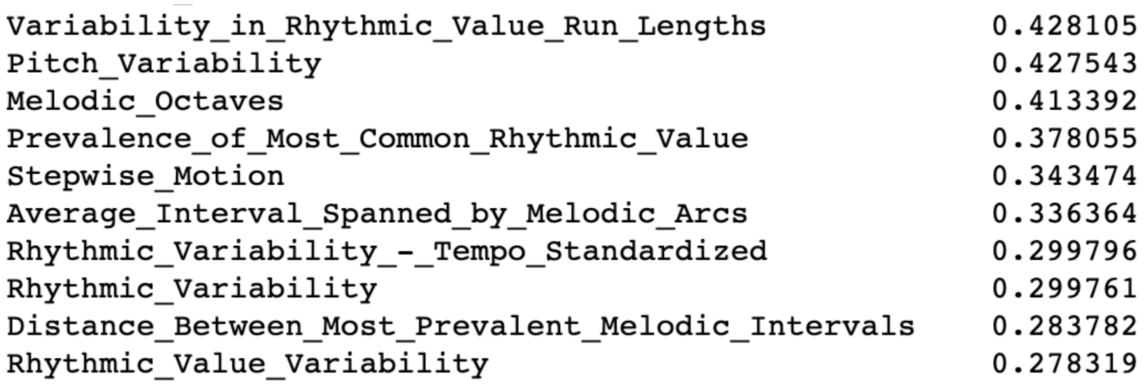
Aybüke Ekiz

# MAIS 202 - PROJECT DELIVERABLE 3

## Art Classifier: Piece-Composer Identifier

### Final Training Results

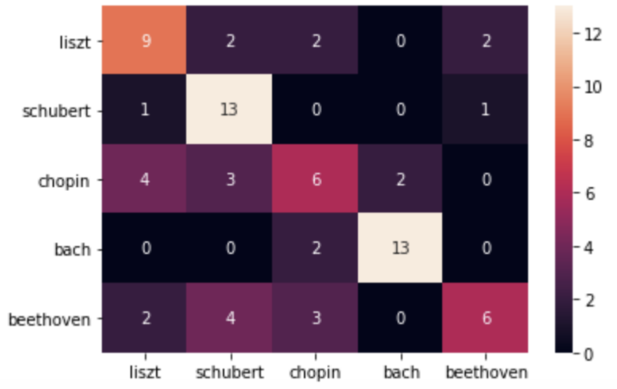
Since my last deliverable, I have done my feature extraction and data pre-processing again, because I wanted to extract more features and experiment with how different features correlate with the composers. I extracted more than 20 features from the midi files using JSymbolic Software. Then, I looked for the correlation between them and the composer, selecting the 10 most correlated features.



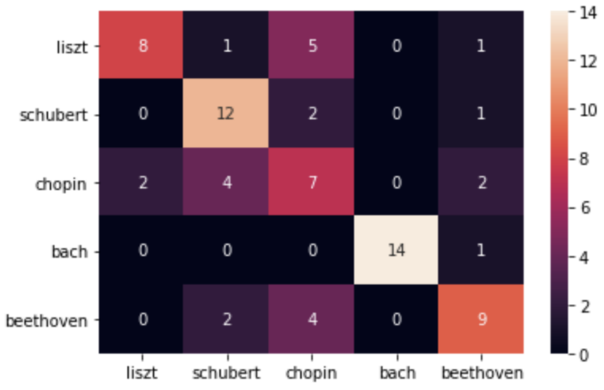
```
Variability_in_Rhythmic_Value_Run_Lengths          0.428105
Pitch_Variability                                  0.427543
Melodic_Octaves                                    0.413392
Prevalence_of_Most_Common_Rhythmic_Value           0.378055
Stepwise_Motion                                    0.343474
Average_Interval_Spanned_by_Melodic_Arcs           0.336364
Rhythmic_Variability_-_Tempo_Standardized          0.299796
Rhythmic_Variability                               0.299761
Distance_Between_Most_Prevalent_Melodic_Intervals  0.283782
Rhythmic_Value_Variability                         0.278319
```

Then, I experimented with various models from scikit-learn. K-neighbours and random forest were the most successful ones. Then, I did hyperparameter tuning on them using my validation data, both manually and by Grid Search. The resulting confusion matrices were:

K-neighbours validation set



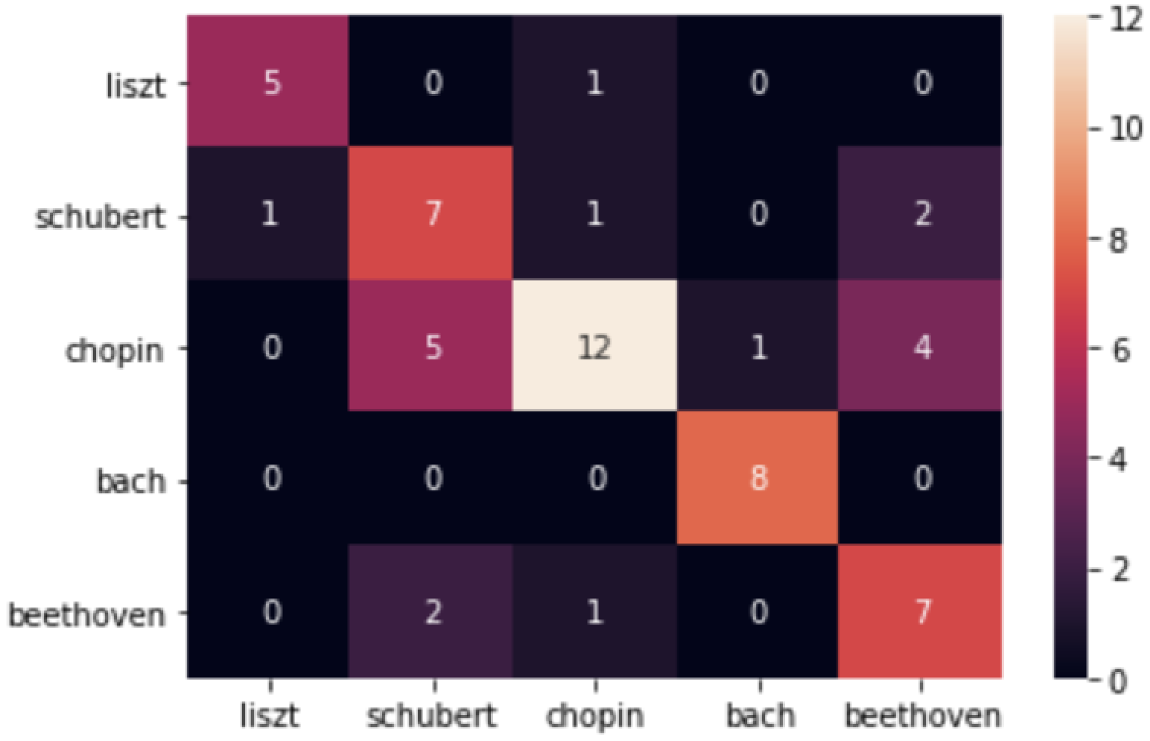
Random Forest validation set



K-neighbours classifier has an accuracy score of 0.63 and the random forest has an accuracy score of 0.67. Even though the accuracy scores are relatively close to each other, I think the confusion matrices look relatively different. Random forest is more confident in its predictions, very accurate for Bach and Beethoven; mixing up between Schubert, Chopin and Liszt. To give some context, Bach is a Baroque composer; Schubert and Beethoven are said to be in transition between Classical and Romantic eras; Liszt and Chopin are Romantic composers.

Aybüke Ekiz

Random Forest Confusion Matrix for the Test Set



It has an accuracy score of 0.68. I found it interesting that in the validation set, the model mixed up Liszt and Chopin a lot, but in the test set not so much. But that is probably because my dataset was small (80-110 pieces for each composer, around 500 pieces in total). If it were a larger dataset, I think the confusions for validation and test sets would be more uniform.

**Final demonstration proposal**

I plan to make a simple game, where the user and computer will both guess the composer of the given piece (The user will hear the piece, the computer will get the extracted features from the midi file) They will get points for correct guesses, and the one with more points after a round of 5 pieces will win. I plan to use Flask, but my experience with Flask is limited with following a tutorial, so I will have to learn it a bit more before building it. (I plan to follow some more tutorials, and the MAIS Flask workshop) I have experience with HTML and CSS though, so I hope it will help.

My plan of the page

After landing on the page, the user will see a textbox explaining the game. Then, after listening to the piece, will be prompted to make a choice between the five composers. Once the user makes his guess, the model's prediction will also be displayed. Then, the title of the piece and its composer will be displayed, and the points of the user and the computer will be updated. After 5 pieces, the winner will be announced. In case of a tie, another piece will be played as a tie-breaker.

The pieces will come from the test set, and they will be randomized, so that every time landing on the page, the user will listen to different pieces. (Hopefully, if I can implement that)