Music Information Retrieval from MAESTRO files: Music Era Classification

Brian Nutwell, bnutwell@gmail.com

INTRODUCTION

In this project I will extract information from classical music piano performances, with the goal of demonstrating that simple features based on music theory can be used to accurately identify (classify) the major classical music era in which the piece was composed.

1 PROJECT DESCRIPTION

This project is a machine learning study on compositional elements of classical music pieces, leveraging the MAESTRO 3.0.0 dataset originally published in 2018. Activities are listed below, corresponding to the sections of this report.

- Demonstrate the ability to **access MIDI** (Musical Instrument Digital Interface) files using existing Python libraries *(MIDO Docs: MIDI Files, 2023)*, and extract the contents into data structures. For each performance, I extracted musical note timing, duration, and velocity from MIDI and exported to a CSV file. See **Data Sources** section.
- From Wikipedia, scrape a catalog of musical composers by "era" of classical music, and clean the composer-name data to connect each MAESTRO piece to a single musical era and composer's nationality. See section on **Musical Eras**.
- Analyze the music data for **basic music theory concepts**. In this step, a single set of 12 features is drawn from the harmonic tones present in each piece – see section on **Feature Extraction** : **Scale Tone Prevalence**.
- Perform **classification** of extracted features by training a variety of models with the classical era metadata (using R). Seven models were tuned, trained and compared: Logistic Regression, Linear Discriminant Analysis, Quadratic Discriminant Analysis, KNN, Gradient Boosting, Random Forest, and Neural Network. See the section on **Models and Tuning**.
- Finally, **demonstrate correlation of the models to classical music era**.
- As a stretch goal, demonstrate that the derived features can additionally be used to identify and discriminate individual compositions, even when

performed by different musicians. See initial results at the end of the **Feature Extraction** section.

Figure 1 shows the original project workflow concept. See the **Appendix** for context of this project as a subset of a larger goal – training a collaborative music-performance AI.

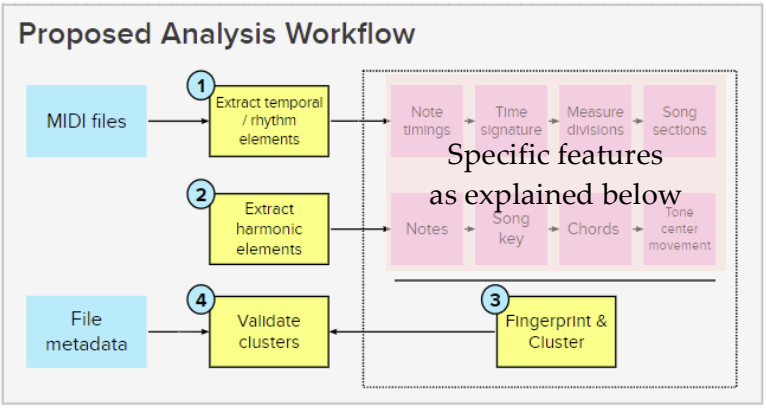

Figure 1 -- Proposed workflow

2 DATA SOURCE

The MAESTRO dataset (*Hawthorne et al., 2018*) consists of both audio and MIDI recordings from ten years (between 2004~2018) of the International Piano-e-Competition, in which solo piano performances were recorded and digitized. The dataset includes 1275 individual performance, each containing on the order of 10^4 individual notes with timing and expression attributes. For this project I will use only the simplified MIDI files. Source data can be found here:

https://magenta.tensorflow.org/datasets/maestro

See *Figure 2* for an example list of pieces and metadata. The dataset authors have provided a suggested Train/Test/Validation split – for the current analysis, I have performed my own data segmentation.

canonical_composer	canonical_title	split	year	duration
Domenico Scarlatti	Sonata, K239	test	2004	196.0154474
Domenico Scarlatti	Two Sonatas	test	2006	373.6107375
Felix Mendelssohn	Etude in A Minor	validation	2008	76.04426156
Felix Mendelssohn	Fantasy in F-sharp Minor, Op. 28 (Complete)	test	2017	661.2050934
Felix Mendelssohn	Rondo Capriccioso in E Minor, Op. 14	validation	2013	362.8361756
Felix Mendelssohn	Rondo Capriccioso in E Minor, Op. 14	validation	2011	359.4574397
Felix Mendelssohn	Variations Serieuses Op. 54	train	2017	724.4814928
Felix Mendelssohn	Variations Serieuses Op. 54	train	2017	665.7766522
Franz Liszt	"La Campanella"	train	2008	269.9716852
Franz Liszt	Annes de Pelerinage III: Le jeux d'eau a la Villa d'Este	train	2011	428.2728089
Franz Liszt	AprÃ¨s une lecture de Dante: Fantasia quasi Sonata, S.162	train	2014	1027.3494
Franz Liszt	AprÃ©s une lecture de Dante: Fantasia quasi Sonata, S.16	train	2014	997.2679664

Figure 2 -- MAESTRO dataset contents also include MIDI and WAV filenames (not shown). Note that some pieces are performed more than once within the dataset.

3 PROJECT HYPOTHESES

- Primary hypothesis: Different eras of classical piano music can distinguished by increasing harmonic complexity, as shown by a small set of extracted features.
- Secondary hypothesis: Different performances of the same classical piano piece can be shown to have nearly identical features, distinct from other pieces.

4 MUSICAL ERAS

For labeling the musical "era" of each piece, I scraped Wikipedia for a full list of composers and their associated eras. Where a composer was listed twice, I categorized them in the earliest listed era. I then linked each MAESTRO piece to a single era based on the composer's name field.

(Note: due to the fluid nature of cultural history, there are not fixed dates for the start or end of an era, and some composers produced work representative of more than one era in their lifetimes. So perfect classification may not be a reasonable goal for this project.)

Wikipedia lists seventeen distinct eras of classical music. After initial classification trials showed significant confusion/overlap between (for example) Early, High, and Late Baroque-era pieces, and with small numbers of pieces to train in some eras, I chose to simplify the list of eras as shown in *Figure 3*.

Index	Era (detailed)	Index2	Era (high level)
1	Middle Ages	**1**	Middle Ages
2	Renaissance		
3	Renaissance-Baroque Transition	**2**	Renaissance
4	Early Baroque		
5	Middle Baroque		
6	Late Baroque		
7	Early Galante	**3**	Baroque
8	Early Classical		
9	Middle Classical		
10	Late Classical		
11	Classical-Romantic Transition	**4**	Classical
12	Early Romantic		
13	Middle Romantic		
14	Late Romantic	**5**	Romantic
15	Post Romantic		
16	Modernist		
17	Postmodernist	**6**	Modern

Figure 3 -- Musical Eras. This project will classify eras to the "Index2" shown above in blue.

The MAESTRO dataset contains pieces from eras 3~6 that are commonly performed in piano competitions. See the table of data era labels below.

Era	**Baroque**	**Classical**	**Romantic**	**Modern**
Index2	3	4	5	6
# of recordings	**197**	**429**	**484**	**153**

5 FEATURE EXTRACTION: SCALE TONE PREVALENCE

5.1 Domain Background and Concept

The basic concept for feature extraction is based on music domain knowledge.

Western music is divided into 12 evenly-spaced musical "notes" in an octave (named A through G-sharp/A-flat), and each octave repeats with higher versions of the "same" note at double the vibration frequency. These octaves are further divided into musical "keys" or "scales": subsets of seven of the possible twelve notes, at prescribed intervals. Notes from the same key are intended to sound more pleasing together. The same scale pattern may be used with different starting notes, so for example a piece in A major and a piece in C major will use different subsets of notes but share the intervallic relationship of "scale tones" between the tonic note (A or C) and the other tones.

Generally speaking, as musical eras have progressed, my hypothesis is that more recent forms of music are more complex: they pass through different keys within the same piece, and/or more frequently include tones not in the primary key.

5.2 Feature Extraction

For each piece in the MAESTRO catalog, I extracted every invididual note that was played. Then I summed all the durations of each note, to make a histogram of note prevalence. From the possible 128 notes in MIDI data, a piano has 88 keys and thus 88 potential notes. Most MAESTRO pieces use ~80 unique notes. See *Figure 4* for an example: Song #128 contains 9828 note events between tones 30~100. ~7 specific notes are more prevalent than the others.

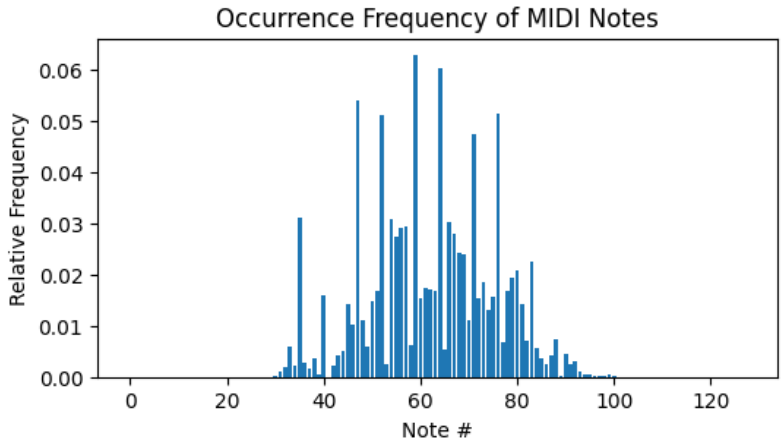

Figure 4 -- Note duration (as a % of total) for one recorded piece #128

Next we collapse or "fold" the octaves by summing the durations of all notes with the tone "A", and do likewise for all 12 tones. This produces *Figure 5 (left)*.

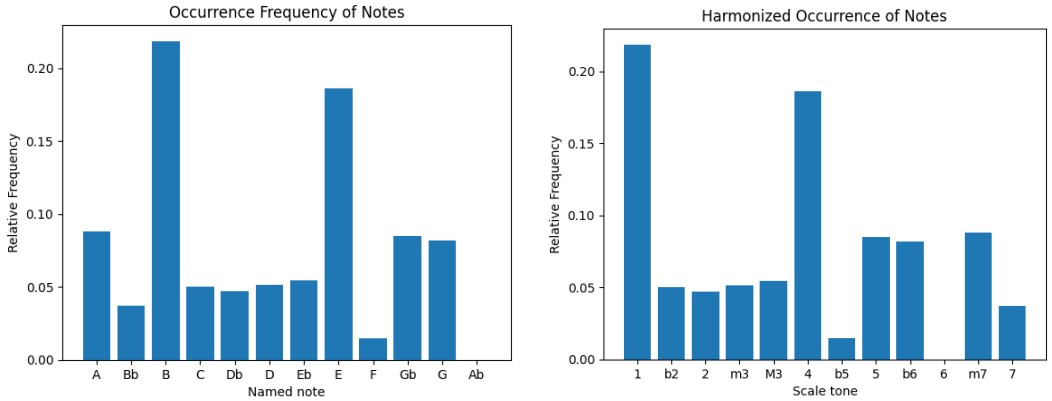

Figure 5 -- Note occurrence (left) and scale tone occurrence (right) for song #128

And finally we re-center the data by selecting the highest-frequency note as the "1" index, so that we can compare songs performed in different keys. This produces *Figure 5 (right)*: **12 features of each performance** that can be compared directly between eras and performances.

Note: I developed this approach independently, but it is quite similar to the "folded octave histogram" approach taken by Tzanetakis & Cook (2002) for harmonic content features, which I discovered after my HW5 update paper. Primary differences in my approach are:

- I use all 12 tones directly as features, rather than extracted values and ratios for the primary notes as in their 2002 paper.
- After "folding" the histogram, I do not take their additional step of ordering the scale tones by the "circle of fifths" transformation. As I am using all 12 features, this reordering would not change the model results. I feel my approach improves readability of the histograms for a non-expert audience.

One additional comment: the above feature extraction was performed in Python to take advantage of the MIDO library, for MIDI file usage. The machine learning portion and the remainder of the project were then done in R.

5.3 Feature Validity Evaluation

We can examine the distribution of scale tones across all pieces in the full dataset, as in *Figure 6*. Far from being randomly distributed, we see that musically pleasing chord tones such as the 4th and perfect-5th (indexes 6 and 8 of 12) are

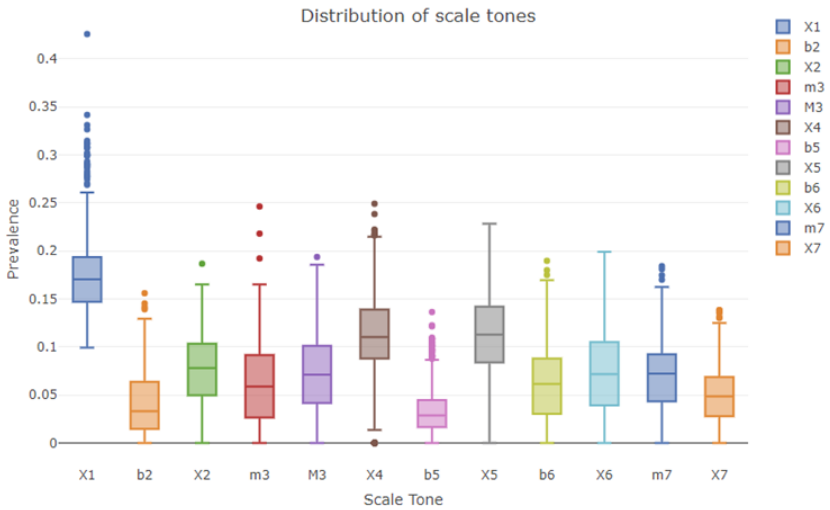

Figure 6 -- Scale tone prevalence distribution shows clear musical trends

represented far more often than "outside the scale" tones such as the flat-2nd and flat-5th (indexes 2 and 7).

We can also see trends in the changing usage of scale tones through eras. *Figure 7* below contrasts the distribution of the perfect-5th (left) with the flat-2nd note (right). There are clear trends showing increasing usage of more challenging or sophisticated harmonic elements in later eras, at the expense of simpler intervals.

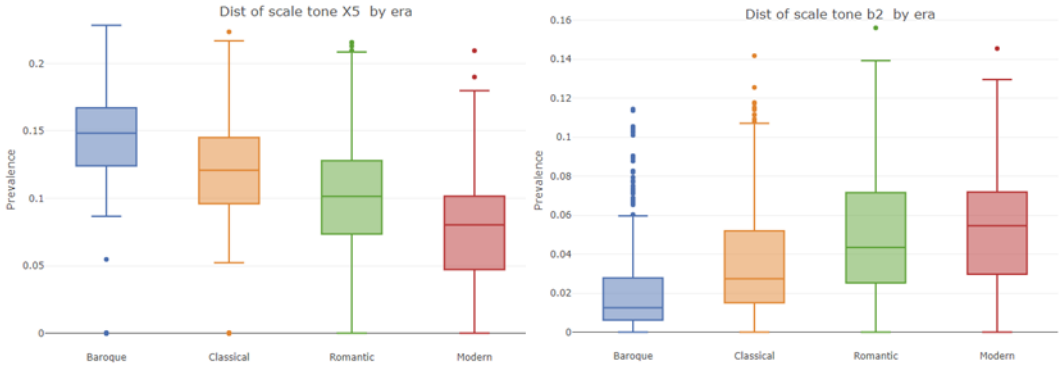

Figure 7 -- Changing prevalence of perfect-5th (left) and flat-2nd (right) by era

Another insight can be gained by comparing two different performances of the same piece (songs 19 and 20, A. Scriabin *Sonata #2*, performed by different competitors in 2017 and 2018) against a different piece by the same composer (song 21, A Scriabin *Sonata #3*, performed in 2018). We can see the different harmonic

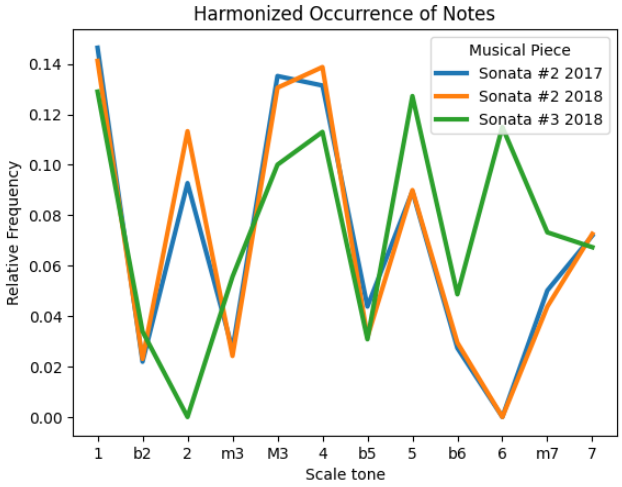

Figure 8 -- Tone prevalence in performances of the same vs. different compositions. Blue and orange lines are very similar, as unique performances of the same piece.

"fingerprint" of each piece in *Figure 8* below. Sonata #3 has much higher use of the natural 5th and 6th tones, and much lower use of the 2nd. Without conducting statistically rigorous study, the analysis suggests we can distinguish unique compositions by these 12 features.

Finally, in *Figure 9* we can see the feature correlation for all 12 scale tones across the entire dataset. The "checkerboard" pattern indicates that tones a half-step apart do not commonly occur at high rates together (lighter cells are negative correlation), which is musically logical as any pair of adjacent tones sounds discordant.

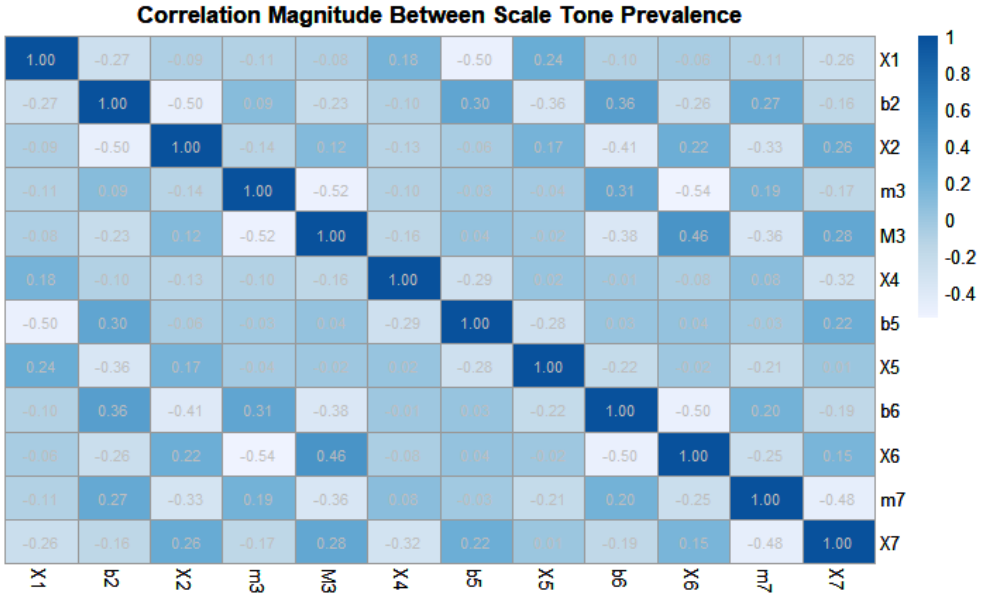

Figure 9 -- Scale Tone Correlation shows low correlation (white cells), especially for adjacent discordant tones one half-step apart.

These analyses indicate this feature-set is reasonable, to proceed to modeling.

6 MODELS & TUNING

After splitting the data 80/20 into Training and Test datasets, 7 different models were trained to classify the musical pieces into eras. Each model was trained with 10-fold cross-validation from the Training set.

- Logistic Regression (family = gaussian)
- Linear Discriminant Analysis
- Quadratic Discriminant Analysis (method = mle)
- KNN (k=3)

- Random Forest (ntrees=500)
- Boosting using the Caret package (as gbm() multiclass boosting is broken)

For the KNN model, I investigated various values of k. Best results were obtained with k=1 and k=3. As some individual data points in the dataset are repeated performances of the same piece, we chose k=3 to avoid any "single matching data point" occurrences between Test and Training causing the model to have artificially high accuracy.

For the Random Forest, I experimented with the number of trees as shown in *Figure 8*. This parameter had marginal impact on the performance; I selected n=500 (the default) as it was slightly superior to other settings.

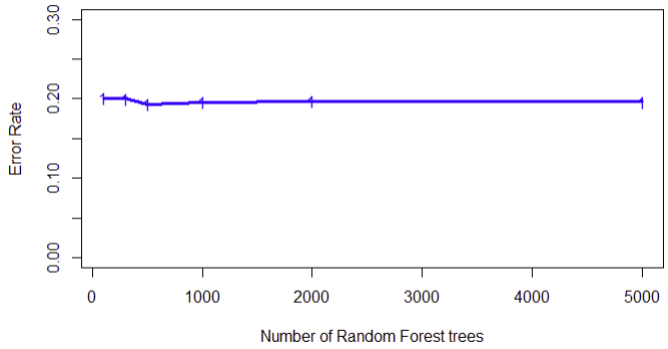

Figure 10 -- Training Error vs. # of Random Forest trees

For the Boosting model, we used the Caret package in R. The Train() function produced a recommended n.trees = 500 and interaction.depth = 3.

7 MODEL ACCURACY

7.1 Training Set Cross Validation

Performance in n Monte Carlo trials on the Training dataset is shown in the table of error rates and *Figure 8* below. Random Forest (n=500 trees) has the best performance with minimal variation between runs, followed by KNN (k=3) and Boosting. Other baseline models had much higher error rates.

	LDA	QDA	Logistic Regression	KNN	Random Forest	Boosting
Error Rate	0.473	0.418	0.522	0.272	**0.230**	0.289

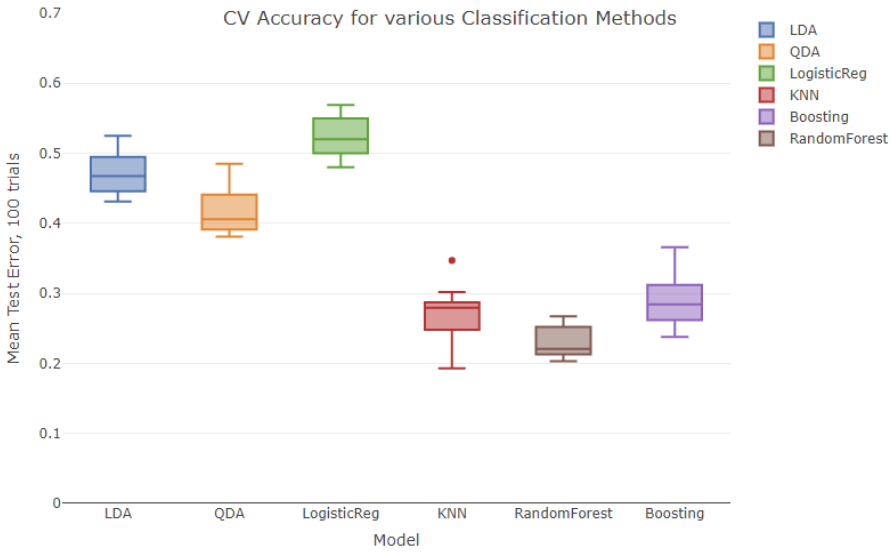

Figure 11 -- CV Accuracy on Training Dataset

7.2 Test Performance

The final performance against the Test dataset is shown in the table below. We see that KNN and Random Forest models achieve roughly 80% accuracy in classifying the correct musical era.

	LDA	QDA	Logistic Regression	KNN	Random Forest	Boosting
Error Rate	0.451	0.423	0.530	**0.198**	**0.202**	0.281

A confusion matrix from the Random Forest model (below) provides more insight into performance on the 258 Test data points. The model has the most trouble with the Modern era pieces (class 6) and also confuses the Romantic era (class 5) with both earlier and later eras.

Table: Musical Era Label

		Actual			
		3	4	5	6
Predicted	3	**43**	5	4	0
	4	5	**70**	7	2
	5	1	12	**78**	14
	6	0	1	0	**11**

The number of pieces mis-classified by two or more eras is only 8/258, or 3%

7.3 Feature Importance

Figure 12 shows the importance of each feature from the Random Forest classifier. We see that scale tones X2 or 2nd interval, X6 or 6th, b2 or flat-2nd, and X5 or perfect-5th, are highly ranked in both measures of importance. Changes in these scale tones are the most impactful in identifying an era, which is consistent with our earlier observations during feature preparation.

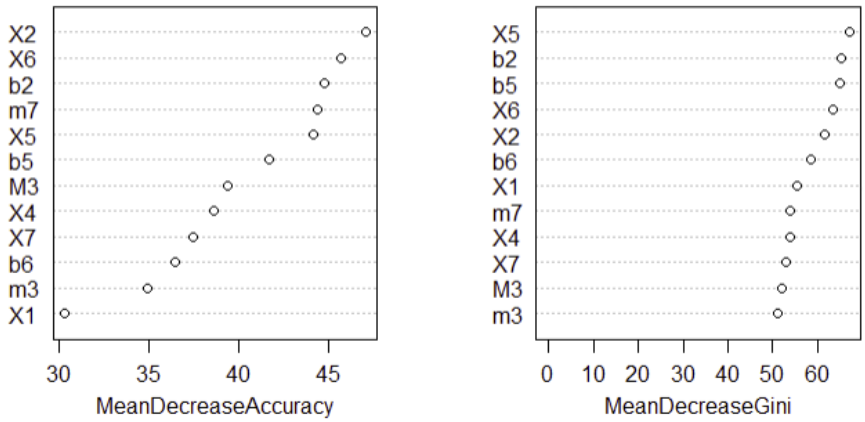

Figure 12 -- Scale Tone feature importance for Random Forest model

8 DISCUSSION AND POTENTIAL FURTHER RESEARCH

For our primary hypothesis, "Different eras of classical piano music can distinguished by increasing harmonic complexity, as shown by a small set of extracted features", we have demonstrated **that these models CAN distinguish eras at an 80% accuracy**.

For a multi-class classification problem such as this one, Random Forest and K-Nearest-Neighbors have proven to be the most powerful algorithms. Although ensemble methods such as Random Forest lack simple explainability, we can use metrics such as feature importance and the median value for each class to provide a measure of insight into the workings of the algorithm.

The present approach of extracting "scale tone prevalence" as a measure of harmonic complexity is demonstrated to explain a significant amount of the variance between musical eras -- despite containing no information about rhythm, expression and volume dynamics, or any other elements of composition and

performance. Rhythmic features would be a suggested priority area of further study to improve the model fidelity.

Another avenue would be improved labeling of individual compositions. For example, the composer Ludwig van Beethoven's early pieces are often regarded as traditionally Classical, while his later works may be considered to mark the start of the Romantic era (*Wikipedia: Ludwig van Beethoven*). But in this present work I have labeled them all as Romantic for convenience. More sophisticated labeling based on deeper domain knowledge could possibly further increase the model fidelity.

9 LESSONS LEARNED

This is my ninth course in the Computational track of the OMSA. From this course itself, the primary lessons learned were a deeper understanding of the algorithms behind a variety of machine learning models, how to approach parameter selection and tuning for each, and how to interpret the results. My personal reference file of R models, cross-validation approaches, and best practices for visualizing output is much more comprehensive now vs. in previous semesters.

From this project, I gained a new appreciation for the power of even basic domain knowledge to guide data cleaning and feature development/selection. I was able to reduce a problem with 1300 observations of ~10^4 data points each (>20 million total data points) to a set of 12 features for each observation, a thousand-fold reduction in problem scope, while achieving surprisingly high accuracy. It was also a good opportunity to explore a several different algorithmic approaches to classification in a reasonably large local dataset.

10 APPENDIX

10.1 Higher-level project concept

This project constitutes a key sub-task of an overall concept for an AI music collaboration tool that could generate complementary music parts for performance in real time (*Figure 13*). For ISYE7406 Spring 2023, I propose to address the elements in the blue spline below: extracting information from MIDI files of classical piano performances, and demonstrating various analyses including classification. Time permitting, I may try to capture some of the analysis attributes into agent rules for the proposed responsive music-generation tool.

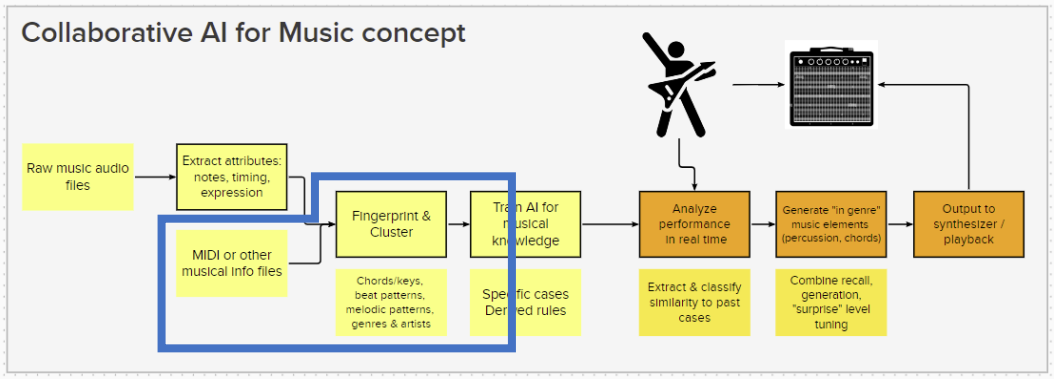

Figure 13 -- Overall concept, with ISYE7406 project in blue spline

10.2 Sample MAESTRO file metadata

The providers have already proposed a segregation into test-train-validation datasets. Note that some pieces are repeated by different performers in

different years – this will be a good test of our ability to extract matching features from similar but not identical pieces.

We can also see that the "canonical composer" field has been pre-cleaned, but the "canonical title" data contains some variations.

canonical_composer	canonical_title	split	year	duration
Domenico Scarlatti	Sonata, K239	test	2004	196.0154474
Domenico Scarlatti	Two Sonatas	test	2006	373.6107375
Felix Mendelssohn	Etude in A Minor	validation	2008	76.04426156
Felix Mendelssohn	Fantasy in F-sharp Minor, Op. 28 (Complete)	test	2017	661.2050934
Felix Mendelssohn	Rondo Capriccioso in E Minor, Op. 14	validation	2013	362.8361756
Felix Mendelssohn	Rondo Capriccioso in E Minor, Op. 14	validation	2011	359.4574397
Felix Mendelssohn	Variations Serieuses Op. 54	train	2017	724.4814928
Felix Mendelssohn	Variations Serieuses Op. 54	train	2017	665.7766522
Franz Liszt	"La Campanella"	train	2008	269.9716852
Franz Liszt	Annes de Pelerinage III: Le jeux d'eau a la Villa d'Este	train	2011	428.2728089
Franz Liszt	AprÃ¨s une lecture de Dante: Fantasia quasi Sonata, S.16	train	2014	1027.3494
Franz Liszt	AprÃ©s une lecture de Dante: Fantasia quasi Sonata, S.1(	train	2014	997.2679664

Figure 14 ***-- Sample metadata***

10.3 References

Hawthorne, C., Stasyuk, A., Roberts, A., Simon, I., Huang, C.-Z. A., Dieleman, S., Elsen, E., Engel, J., & Eck, D. (2018). *Enabling Factorized Piano Music Modeling and Generation with the MAESTRO Dataset*.

MIDO Docs: MIDI Files. (2023). https://mido.readthedocs.io/en/latest/Midi_files.html

List of Classical Music Composers by Era: Wikipedia (2023) https://en.wikipedia.org/wiki/List_of_classical_music_composers_by_era

Ludwig van Beethoven: Wikipedia (2023) https://en.wikipedia.org/wiki/Ludwig_van_Beethoven

Tzanetakis, G., Cook, P. (2002) *Musical Genre Classification of Audio Signals*, IEE Transactions on Speech and Audio Processing, Vol. 10, No. 5.