Automatic Figured Bass Annotation Using the New Bach Chorales Figured Bass Dataset

Yaolong Ju[1], Sylvain Margot[1], Cory McKay[2], Luke Dahn[3], Ichiro Fujinaga[1]

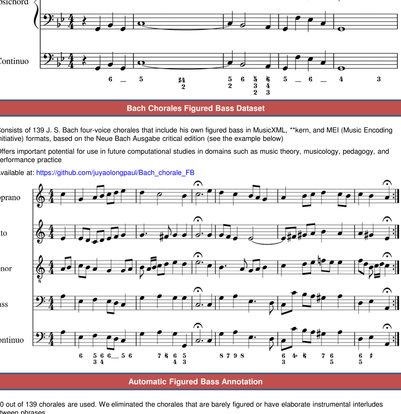
[1] McGill University, Canada [2] Marianopolis College, Canada [3] The University of Utah, USA

Paper in a nutshell:

- **Created a dataset of Bach chorales with his own figured bass**
- **Figured bass was automatically generated for the unfigured Bach chorales**

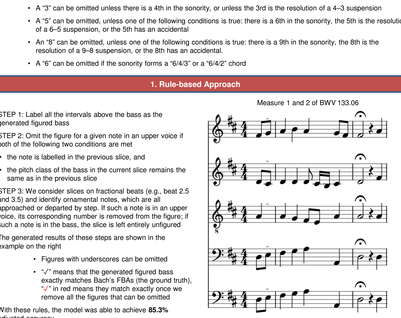
Introduction of Figured Bass

- Indicating intervals to be played above/below a bass note (see below)
- Serving as a guide for improvising the bass continuo accompaniment (see the harpsichord part)
- Figured bass do not always specify all the notes played in the improvisation, and usually omit some figures



Bach Chorales Figured Bass Dataset

- Consists of 139 J. S. Bach four-voice chorales that include his own figured bass in MusicXML, **kern, and MEI (Music Encoding Initiative) formats, based on the Neue Bach Ausgabe critical edition (see the example below)
- Offers important potential for use in future computational studies in domains such as music theory, musicology, pedagogy, and performance practice
- Available at: https://github.com/juyaolongpaul/Bach_chorale_FB



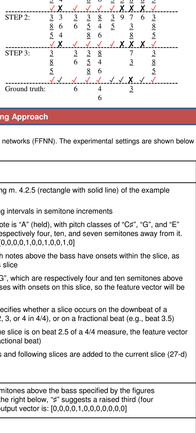
Automatic Figured Bass Annotation

- 120 out of 139 chorales are used. We eliminated the chorales that are barely figured or have elaborate instrumental interludes between phrases
- To learn about Bach's figured bass habits (which is of musicological interest)
- To provide figured bass for those Bach chorales for which no figured bass annotation exist
- The music is segmented into a series of *note onset slices*. A new slice is formed whenever a new note onset occurs in any musical voice, and each slice consists of the vertical set of notes sounding at that moment
- For evaluation, we omit figures that can be implied for both generated figured bass and ground truth and compare the results, which we refers as **adjusted accuracy**. The omission rules are:
  - A "3" can be omitted unless there is a 4th in the sonority, or unless the 3rd is the resolution of a 4–3 suspension
  - A "5" can be omitted, unless one of the following conditions is true: there is a 6th in the sonority, the 5th is the resolution of a 6–5 suspension, or the 5th has an accidental
  - An "8" can be omitted, unless one of the following conditions is true: there is a 9th in the sonority, the 8th is the resolution of a 9–8 suspension, or the 8th has an accidental.
  - A "6" can be omitted if the sonority forms a "6/4/3" or a "6/4/2" chord

1. Rule-based Approach

- STEP 1: Label all the intervals above the bass as the generated figured bass
- STEP 2: Omit the figure for a given note in an upper voice if both of the following two conditions are met
  - the note is labelled in the previous slice, and
  - the pitch class of the bass in the current slice remains the same as in the previous slice
- STEP 3: We consider slices on fractional beats (e.g., beat 2.5 and 3.5) and identify ornamental notes, which are all approached or departed by step. If such a note is in an upper voice, its corresponding number is removed from the figure; if such a note is in the bass, the slice is left entirely unfigured
- The generated results of these steps are shown in the example on the right
  - Figures with underscores can be omitted
  - "✓" means that the generated figured bass exactly matches Bach's FBAs (the ground truth), "✗" in red means they match exactly once we remove all the figures that can be omitted
- With these rules, the model was able to achieve **85.3%** adjusted accuracy

Measure 1 and 2 of BWV 133.06



2. Machine Learning Approach

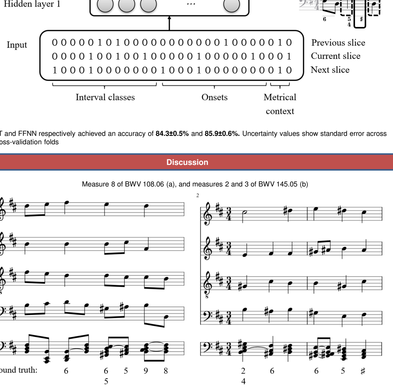
- We experimented with decision trees (DT) and feedforward neural networks (FFNN). The experimental settings are shown below

Evaluation metric	Adjusted accuracy

Input	Three feature vectors, demonstrated using m. 4.2.5 (rectangle with solid line) of the example shown on the right below:
	• 12-d **interval-class** vector representing intervals in semitone increments
	  • In this example, the bass note is "A" (held), with pitch classes of "C#", "G", and "E" (held) above it, which are respectively four, ten, and seven semitones away from it. The feature vector is thus: [0,0,0,0,1,0,0,1,0,0,1,0]
	• 12-d **onset** vector, which specify which notes above the bass have onsets within the slice, as opposed to being held from a previous slice
	  • In this example, "C#" and "G", which are respectively four and ten semitones above the bass, are the pitch classes with onsets on this slice, so the feature vector will be [0,0,0,0,1,0,0,0,0,0,1,0]
	• 3-d **metrical context** vector, which specifies whether a slice occurs on the downbeat of a measure, on another beat (e.g., beat 2, 3, or 4 in 4/4), or on a fractional beat (e.g., beat 3.5)
	  • In this example, because the slice is on beat 2.5 of a 4/4 measure, the feature vector will be [0,0,1] (i.e., it is a fractional beat)
	The features (27-d each) for the previous and following slices are added to the current slice (27-d) as context, making an 81-d input vector

Output	12-d vector: specifying the number of semitones above the bass specified by the figures • In m. 4.2.5 of the example shown on the right below, "#" suggests a raised third (four semitones) above the bass "A". The output vector is: [0,0,0,0,1,0,0,0,0,0,0,0]
Loss-function	Binary cross-entropy
Learning algorithms	DT: CART (Classification And Regression Tree) FFNN: ADAM (Adaptive Moment Estimation) with 3 hidden layers and 300 nodes each layer
Data split	90% (108 chorales) for training, and 10% (12 chorales) for testing 80% (108 chorales) for training, 10% (12 chorales) for validating, and 10% (12 chorales) for testing
Evaluation method	10-fold cross validation

- The machine learning architecture (using FFNN as an example) is shown below:

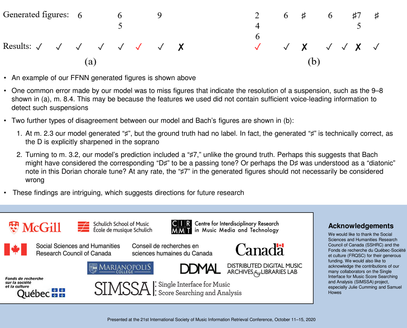


Measure 4 of BWV 117.04

- DT and FFNN respectively achieved an accuracy of **84.3±0.5%** and **85.9±0.6%.** Uncertainty values show standard error across cross-validation folds

Discussion

Measure 8 of BWV 108.06 (a), and measures 2 and 3 of BWV 145.05 (b)



- An example of our FFNN generated figures is shown above
- One common error made by our model was to miss figures that indicate the resolution of a suspension, such as the 9–8 shown in (a), m. 8.4. This may be because the features we used did not contain sufficient voice-leading information to detect such suspensions
- Two further types of disagreement between our model and Bach's figures are shown in (b):
  1. At m. 2.3 our model generated "♯", but the ground truth had no label. In fact, the generated "♯" is technically correct, as the D is explicitly sharpened in the soprano
  2. Turning to m. 3.2, our model's prediction included a "♯7," unlike the ground truth. Perhaps this suggests that Bach might have considered the corresponding "D♯" to be a passing tone? Or perhaps the D♯ was understood as a "diatonic" note in this Dorian chorale tune? At any rate, the "♯7" in the generated figures should not necessarily be considered wrong
- These findings are intriguing, which suggests directions for future research

Acknowledgements

We would like to thank the Social Sciences and Humanities Research Council of Canada (SSHRC), and Fonds de recherche du Québec–Société et culture (FRQSC) for their generous funding. We would also like to acknowledge the contributions of our colleagues in the Digital Music Interface for Music Score Searching and Analysis (SIMSSA) project, especially Julie Cumming and Samuel Howes

Presented at the 21st International Society of Music Information Retrieval Conference, October 11–15, 2020