Big Data, Big Questions: A Closer Look at the Yale–Classical Archives Corpus (c. 2015)

TREVOR de CLERCQ[1]
Middle Tennessee State University

ABSTRACT: This paper responds to the article by Christopher White and Ian Quinn, in which these authors introduce the Yale-Classical Archives Corpus (YCAC). I begin by making some general observations about the corpus, especially with regard to ramifications of the keyboard-performance origins of many pieces in the original MIDI collection. I then assess the accuracy of the scale-degree and local-key fields in the database, which were generated by the Bellman-Budge key-finding algorithm. I point out that some of the inaccuracies from the key-finding algorithm's output may influence the results we obtain from statistical studies of this corpus. I also offer an alternative analysis to the authors' finding that the ratio of V^7 to V chords increases over time in common-practice music. Specifically, I conjecture that this finding may be the result of (or related to) increasing instrumental resources over time. I close with some recommendations for future versions of the corpus, such as enabling end users to help repair transcription errors as well as offer ground truths for harmonic analyses and key area information.

Submitted 2015 December 13; accepted 2016 February 8.

KEYWORDS: *corpus analysis, key-finding, modulation, tonality, machine learning*

IN the current issue of this journal, Christopher White and Ian Quinn provide an overview of the Yale-Classical Archives Corpus (YCAC)—a massive collection of CSV data files derived from user-generated MIDI encodings of more than 14,000 works, spanning from the 1500s to the late 20th century and representing a broad range of Western classical music styles. A corpus of classical music this large promises to be a valuable resource for empirical researchers hoping to answer big questions in this age of big data. After describing some of the technical aspects of the corpus (e.g., data fields, metadata), the authors give an overview of some of its general properties. Perhaps not surprisingly, for example, most works are of German-Austrian origin (e.g., Mozart, Bach, and Beethoven). The authors then describe some limitations and idiosyncrasies of the corpus, and they finish with a sample study. In their sample study, the authors show that the ratio of V^7 to V chords increases over time, generally speaking, implying that composers in later centuries seem to have preferred V^7 chords over regular V chords more than did composers from earlier centuries.

After reading the article and looking closely at the corpus, I believe some additional points are worth noting for any researcher interested in using this corpus as it currently stands. In what follows, I provide some general observations of the corpus that expand upon those discussed by White and Quinn. I also consider the value of four related fields in the corpus (those dealing with local key and local scale degree), which are automatically generated by a key-finding algorithm. I go on to offer alternative interpretations of some of the authors' findings, including the distribution of scale-degree sets and the ratio of triads to seventh chords. Finally, I outline some recommendations that, while requiring time and labor to implement, would improve the corpus and increase its value to the empirical musicology community.

SOME GENERAL OBSERVATIONS

To understand this corpus and its possibilities, it seems prudent to inspect the raw data. One hardly knows where to begin with a corpus of almost 14,000 pieces, so I decided to simply look at the first piece in the first file in the first folder: the Courante from J. S. Bach's English Suite no. 2 in A minor, BWV 807. I have reproduced the first bar of the original score (including the anacrusis) in Figure 1 below. The corresponding

data from the first two fields in the corpus are shown in Table 1[2]. In Table 1, each row represents a new "salami slice" (i.e., the addition or subtraction of a pitch from the texture); the left-hand column represents the offset in quarter notes from the beginning of piece; and the right-hand column represents the pitch content itself. (The "Chord" field, which is the label found in the corpus, is what I assume the authors refer to as the "RawPitchTuple" field in their article.)

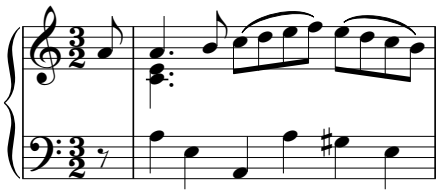

Figure 1. J. S. Bach, English Suite no. 2 in A minor, BWV 807, Courante, m. 1.

Table 1. The first two fields ("offset" and "Chord") from the YCAC, which correspond to the first bar (including the anacrusis) of the Courante from J. S. Bach's English Suite no. 2 in A minor, BWV 807.

offset	Chord
0	<music21.chord.Chord A4>
0.5	<music21.chord.Chord C4 E4 A4 A3>
1.25	<music21.chord.Chord C4 E4 A4>
1.5	<music21.chord.Chord C4 E4 A4 E3>
1.75	<music21.chord.Chord E3>
2	<music21.chord.Chord B4 E3>
2.25	<music21.chord.Chord B4>
2.5	<music21.chord.Chord C5 A2>
2.875	<music21.chord.Chord A2>
3	<music21.chord.Chord D5 A2>
3.25	<music21.chord.Chord D5>
3.5	<music21.chord.Chord E5 A3>
3.875	<music21.chord.Chord A3>
4	<music21.chord.Chord F5 A3>
4.25	<music21.chord.Chord F5>
4.5	<music21.chord.Chord E5 G#3>
4.875	<music21.chord.Chord G#3>
5	<music21.chord.Chord D5 G#3>
5.25	<music21.chord.Chord D5>
5.5	<music21.chord.Chord C5 E3>
5.875	<music21.chord.Chord E3>
6	<music21.chord.Chord B4 E3>
6.25	<music21.chord.Chord B4>

When we compare the corpus encoding in Table 1 to the music notation in Figure 1, a few issues are quickly apparent. To begin with, we can see here, as White and Quinn openly admit in their article, that the corpus does not contain much useful metric information. For example, the downbeat of bar 1 occurs at offset 0.5 due to the eighth-note pickup. Integer values in the corpus, therefore, may or may not represent strong beats. Additionally, the use of a quarter note as the standard offset, which in this case is only half a beat given the 3/2 meter, makes the raw data in the corpus somewhat difficult (although not impossible) to compare manually with the score[3]. Measure numbers have to be calculated knowing the meter and pickup length for each individual work, and unfortunately, no meter information is included in the metadata file or the pitch data files. The lack of usable metrical information is a serious lacuna, in my opinion, and severely limits what can be done with this corpus as it currently stands. That said, the authors are upfront about this issue, and there are ways to remediate it, as I discuss below.

A second issue can be found with the event at offset 1.25. At this moment in the piece, the corpus encodes a new event occurring on the sixteenth note before the second quarter note of bar 1. At first glance, an event at this location seems rather odd, since the notes on the downbeat of bar 1 all last a quarter note or longer. This event (C4, E4, A4) is, I would guess, the result of a human piano player lifting his or her left hand from the note A3 in the bass to the note E3 in the bass, which results in a recorded duration for the first bass note of only a dotted-eighth note instead of a full quarter note. The event at offset 1.25, therefore, represents a spurious simultaneity, generated by the mechanics of piano performance. These types of spurious simultaneities continue throughout this one-bar excerpt (and the piece as a whole), creating upwards of twice as many salami slices as necessary. (The excerpt in Figure 1 should only have 12 salami slices, by my calculation, instead of the 23 listed in Table 1.) These are not pitch errors, per se, but they are misrepresentations of the notated score, if one assumes the salami slices are meant to represent the original score faithfully. It is somewhat more difficult, for example, to track the bass line correctly or judge chord inversions, since a low bass part is not consistently reflected in the salami slices despite its presence in the score.

To repair these spurious simultaneities would not be impossible. For example, a researcher could import the original MIDI file into a digital audio workstation (DAW), such as Avid Pro Tools, and then quantize the "note off" events to an eighth-note grid. This process would have to be carefully done, though, since this particular composition includes a few sixteenth notes and mordents (performed as 32nd notes), which would require a different quantization level. Importing the MIDI file into a DAW would also allow the researcher to correct any meter problems; but this repair process would have to be done on a file-by-file basis, which would require a significant amount of time for even a portion of a corpus this size. Files that were originally encoded via MIDI-keyboard performances could be excluded from the corpus, as the authors mention, but as of yet there is no clear or easy way to do so.

The fact that many pieces were encoded via MIDI-keyboard performances impacts the corpus in at least one other important way. Specifically, I would estimate that the corpus as a whole has a strong sampling bias towards keyboard and piano works. This aspect can be seen in Figure 4 of the authors' article, which shows that the two most common genres in the corpus are "SoloSonata" and "OtherSolo." In the "SoloSonata" genre, for instance—which includes 1,208 unique entries in the metadata file—only 47 pieces by my count are for an instrument other than piano, organ, or harpsichord. A similarly lopsided distribution (although not as dramatic) can be found in the "OtherSolo" genre, of which about 80% of the 1,162 unique entries in the metadata file are for a keyboard instrument, and about 80% of the remaining 20% turn out to be for guitar. To be clear, the corpus does include many symphonic and orchestral works. All nine of Beethoven's symphonies have been encoded, for example. But for J. S. Bach, who is the second-most well-represented composer after Mozart in terms of number of pieces in the corpus, neither the Orchestral Suites (BWV 1066–1069) nor the Brandenburg Concertos (BWV 1046–1051) are included, notable omissions to be sure. In fairness, White and Quinn state clearly that their corpus represents only the priorities of a particular group of music aficionados—individuals who were dedicated to converting notated scores into MIDI information and then posting it on the web. That said, we should keep these sampling biases in mind, as they potentially impact the inferences we make from this data.

ON THE VALUE OF THE LOCAL KEY AND SCALE DEGREE FIELDS

Because the corpus lacks any information about meter, and since meter provides the necessary context for rhythm, it would be difficult to say much of value about rhythmic aspects of the corpus, at least as the corpus currently stands. Our analytical attention, therefore, would seem to have to be directed primarily to the domain of pitch. Like meter for rhythm, tonality provides the context for pitch (at least for the vast majority of the musical works included in this corpus). In other words, it would be difficult to say much of value about the pitch content of this corpus without reference to a key. Accordingly, the authors and their assistants have encoded the global key of each work in the metadata file. But the global key does not provide much meaningful context either, since modulation is a common feature of common-practice-era music. To give the raw pitch data some meaningful context, therefore, the authors encode at each salami slice the local tonic and mode (major or minor). From this local key information, the authors can generate a representation of the local scale-degree content of the salami slice (reported as "LocalSDNormalOrder" in their paper, though labeled as "LocalSDForm_BassSD" in the corpus itself, since it also indicates which scale degree is in the bass). The local key information is generated by the Bellman-Budge key-finding

algorithm (Bellman, 2006; Budge, 1943), and another field also reports the confidence level (specifically, a correlation coefficient varying from 0 to 1, with 1 being the highest "confidence").

It is worth examining the output from this key-finding algorithm in more detail, since it forms the basis of the pitch context in the corpus. Again, I chose to examine the first piece in the first file in the first folder of the corpus (the Courante from J. S. Bach's English Suite no. 2 in A minor, BWV 807). The opening 12 bars of score, along with the output from the key-finding algorithm, are shown below in Figure 2. (The reader is encouraged at this point to play through or listen to the piece, noting how well the output from the key-finding algorithm reflects his or her hearing.)

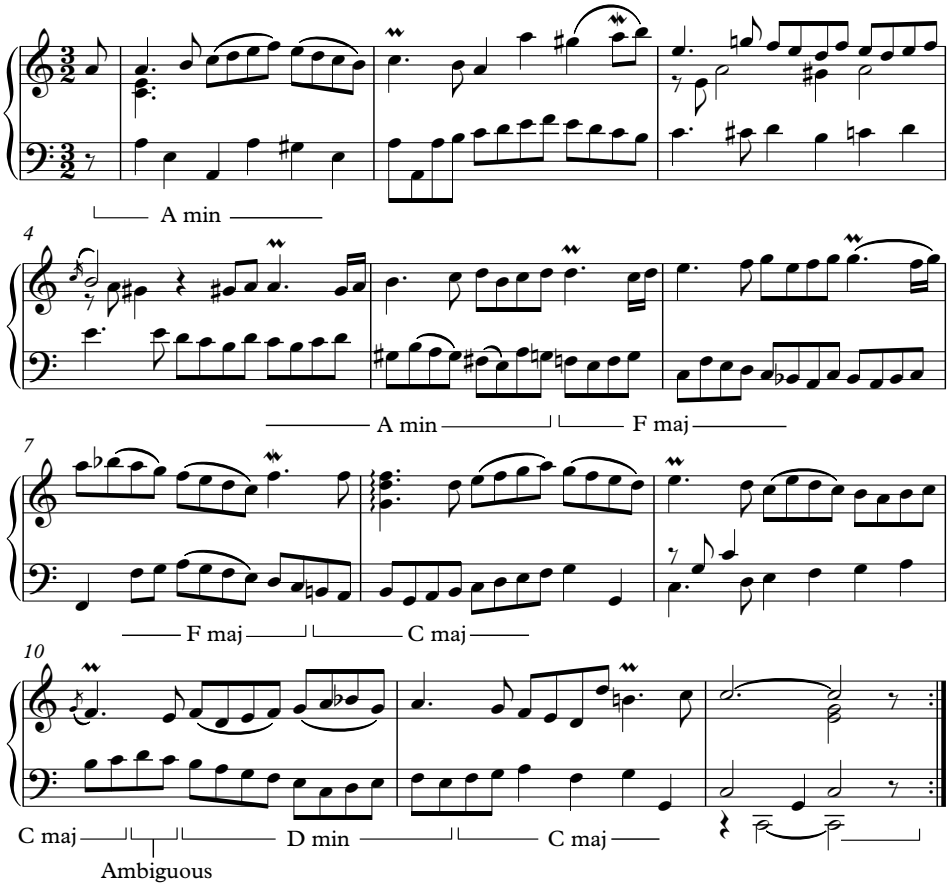

Figure 2. J. S. Bach, English Suite no. 2 in A minor, BWV 807, Courante, mm. 1–12, showing the key areas as encoded in the YCAC by the Bellman-Budge key-finding algorithm.

To my ear, this piece begins in A minor, modulates to C major somewhere around the downbeat of bar 6, and stays in C major until the double bar at the end of m. 12. The key-finding algorithm correctly identifies the opening and ending keys, but in contrast to my hearing, it posits a move to F major around the end of bar 5 until the the B-natural in the bass near the end of bar 7, as well as a move to D minor in bar 10 (which is preceded by an "ambiguous" span of two eighth notes). The modulation to D minor seems like a clear error on the part of the key-finding algorithm, influenced presumably by the B-flat near the end of bar 10. The move to F major, though, is somewhat understandable, since the IV chord is heavily tonicized during this passage. Nonetheless, the key-finding algorithm does not reflect my personal understanding of the scale-degree content of these 12 bars very well. Overall, I would say it only labels about 75% of the local keys correctly (9 out of 12 bars), assuming one takes my hearing as the "correct" hearing.

In all fairness, had I been presented with the material that the algorithm labels as F major (especially the latter half of bar 6 and the first half of bar 7) in isolation, I would have undoubtedly labeled it as F major myself. The problem is that the key-finding algorithm has no knowledge of key hierarchies or tonicization, or expectations about key relationships in works from this era. (This is not the authors' fault,

of course, but just the nature of current key-finding algorithms.) Instead, the algorithm is simply trying to find the best fit for a set of idealized scale-degree distributions across a sliding set of eight windows. (See the authors' original article for more information on the exact process.) The specific scale-degree distributions for the algorithm are given in Bellman (2006, p. 82), where one can see that the scale degrees of the tonic chord (in either a major or minor key) receive the highest three weightings. As a result, we can predict that the key-finding algorithm will have the tendency to interpret any given measure as containing tonic. It is no surprise, therefore, that the tonicization of F major is interpreted as a key change.

To better assess the operation of the key-finding algorithm, I thought it would be worthwhile to examine another composition in the corpus. I chose next to look at the first piece in the second file in the first folder: movement 2 from Beethoven's Piano Sonata no. 26, op. 81a. As it turns out, this particular piece is quite thorny with regard to key areas. For example, while the global tonic of the piece is ostensibly C minor, there are only four bars at the beginning (mm. 5-8), a brief moment in the middle (the last half of m. 20), and four bars at the end (mm. 37-40) that are clearly in C minor, at least to my ears. Assessing the accuracy of the key-finding algorithm in this highly chromatic and modulatory setting seemed somewhat unfair. Accordingly, I moved on to the second piece in the second file in the first folder: the third movement from the same work (Beethoven Piano Sonata no. 26, op. 81a). The first few bars of this movement are reproduced below in Figure 3.

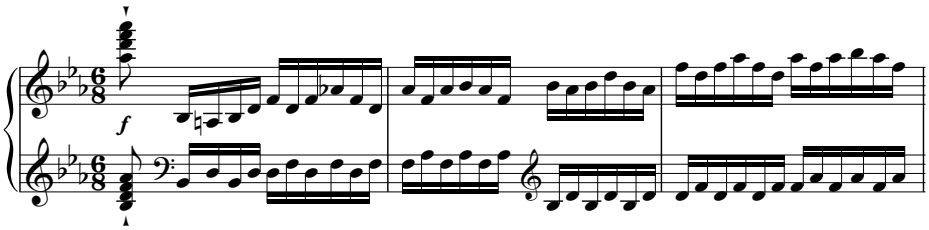

Figure 3. Beethoven Piano Sonata, no. 26, op. 81a, III, mm. 1–3.

For those readers unfamiliar with this piece, it is worth noting that the arpeggiation of what is an obvious dominant-seventh chord continues for the first 10 bars of the movement, followed by the first instance of the E-flat major tonic on the downbeat of bar 11. Unfortunately, because the key-finding algorithm prefers a scale-degree distribution that maps the most frequently-occurring scale-degrees to a tonic chord, the first seven and a half measures of the piece are analyzed as in B-flat major. The key-finding algorithm does get on track, however, by the middle of bar 8. The music stays in E-flat major for about the next 26 bars (up until bar 36 or so), and the key-finding algorithm does label these bars correctly as in E-flat major since there is only a bit of brief tonicization via a few V/ii chords, none of which last longer than a single beat. The situation gets somewhat more complicated around bar 36, however, as the music modulates to the dominant key area of B-flat major. I have reproduced the transition passage below in Figure 4, which includes the key areas encoded in the YCAC by the Bellman-Budge algorithm.

As I hear this passage, the music fairly abruptly switches to B-flat major on the downbeat of bar 37; bars 38–40 outline what is essentially an augmented sixth chord in B-flat, which resolves to V in bars 41–44; the augmented sixth chord then returns for bars 45–48, followed again by V in bars 49–52, after which we arrive at our new theme in bar 53, which is clearly in B-flat major. As Figure 4 shows, the key-finding algorithm tends to identify changing keys instead of changing chords. The prolonged augmented sixth chord (admittedly, missing its characteristic augmented sixth until the last eighth-note of bar 48) proves to be a serious challenge for the key-finding algorithm. Although the algorithm does analyze some of bars 37–52 as having a B-flat tonic (albeit, in a minor mode), it often defaults to labeling each prolonged chord as tonic itself.

Overall, given the first 55 measures of this work, I assess the key-finding algorithm to assign key areas correctly for only about 65% of the passage. I have some questions and concerns, therefore, regarding the value and accuracy of the local key and local scale-degree fields in the corpus. This apparent shortcoming is not the authors' fault, of course. We are simply encountering the inherent limitations and problems with a correlation-based distributional key-finding algorithm. Improved key-finding performance might be attained with a probabilistic distributional key-finding algorithm, as described in Temperley (2007), which reportedly achieves a success rate of around 85% at the local level. But as of yet, no known key-finding algorithm exists that can consistently model how a trained human listener would hear key

structures in a classical work. (An effective key-finding algorithm would presumably require more knowledge than solely scale-degree distributions, such as knowledge about typical harmonic progressions, meter, melodic phrasing, etc. This also assumes, rather naively, that two trained human listeners will analyze the local keys of a piece of music in the same way, which is doubtful given a large set of compositions [see de Clercq & Temperley, 2011 and Temperley & de Clercq, 2013].)

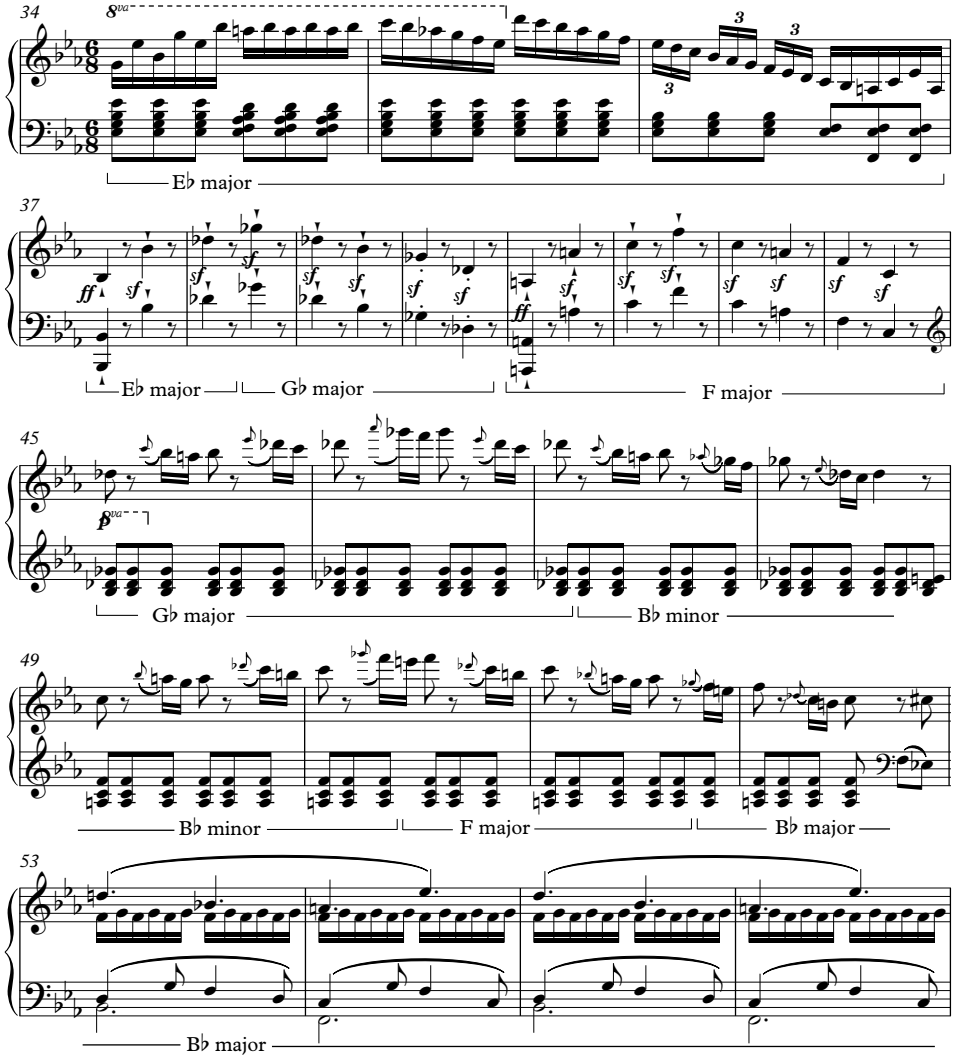

Figure 4. Beethoven Piano Sonata, no. 26, op. 81a, III, mm. 34–56, showing the key areas as encoded in the YCAC by the Bellman-Budge key-finding algorithm.

Until a more accurate key-finding algorithm is developed (see Temperley [2012] for a recent overview of the field), we should be very careful about drawing inferences from any results derived from a computer-generated scale-degree analysis of a large body of musical works. In Figure 5 of their article, for example, White and Quinn report the most frequent scale-degree sets found in the YCAC, transposed to C major. As this figure shows, the tonic chord is the most frequent. In fact, seven of the top-ten scale-degree sets can be viewed as subsets of the tonic chord. (The other three scale-degree sets in the top ten, perhaps not surprisingly, are subsets of the dominant chord.) But if the key-finding algorithm has the tendency to read any prolonged passage as a tonic harmony instead of a chromatic harmony, then we should not be surprised if the corpus appears to be mostly tonic subsets[4]. The results shown in the authors' Figure 5, therefore, may simply be reflecting (or at least, strongly shaped by) the original distribution of the key-finding algorithm itself.

AN ALTERNATIVE ANALYSIS OF THE SAMPLE CASE

If the rhythmic information in the corpus cannot be analyzed because there is no metric context, and the pitch information is problematic because the encoded tonal context is often suspect, what sorts of tests can we as researchers conduct on this corpus? I cannot say that I am entirely sure, although the sample case that the authors offer at the end of their article may be one viable example. In this case, the authors test the hypothetical hypothesis that later tonal composers preferred V^7 chords over V chords, while earlier tonal composers preferred the opposite. Ignoring the mysterious origins of this hypothesis[5], the corpus might be able to provide an answer. Admittedly, the identification of V and V^7 chords does require information about the local tonic, so we cannot expect our results to be ideal. But since the main problem with the key-finding algorithm seems to be that it views tonicizations and prolonged chromatic chords as tonic chords in a spurious key area, we might presume that V or V^7 chords are only underrepresented in the final statistics.

For the sake of argument, at least, let us simply presume that the key-finding algorithm models our hearing very well. The next step would be to investigate how the proportion of V and V^7 chords might be tallied in a work from the corpus. Consider, for example, the consequent phrase from the parallel period that opens the second movement of Mozart's Symphony no. 7, as shown in Figure 5. I will assume here that the consequent phrase modulates from G major to the key of D major. Given this modulation, I hear two V^7 chords in this passage: the first during beat 4 of bar 7, the second during beat 2 of bar 8. Assuming the modulation occurs prior to the downbeat of bar 7 (which is how the Bellman-Budge algorithm analyzes this passage), I would not label any instances of a simple V triad here.

In contrast, the salami slice analysis of this passage would not find any V^7 chords, because the violin arpeggiations break up both of the V^7 chords over time. The salami slice method would instead identify only a single V triad occurring exactly on beat 4 of bar 7. In my analysis of the passage, therefore, the ratio of V^7 to V (2:0) is higher than and directly opposite the ratio found via the salami slice method (0:1). The reason for this discrepancy, of course, is that the salami slice approach does not merge notes displaced over time into a single harmonic entity. Especially in thinner textures, a listener often has to infer harmonies by combining adjacent pitches into a single sonority. The Mozart excerpt below, for example—despite the four staves—is essentially a three-voice texture. Simply put, it is impossible for a three-voice texture to play a fully-fledged dominant-seventh chord; these sonorities can only be implied.

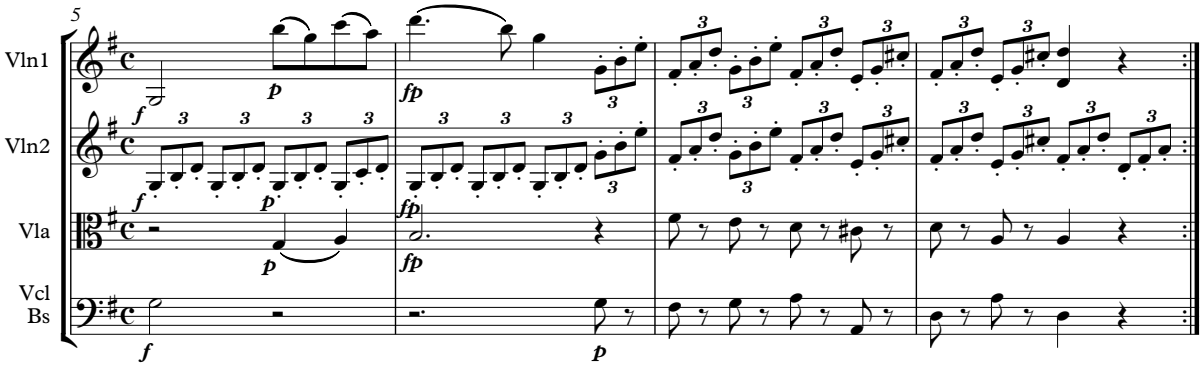

Figure 5. W. A. Mozart, Symphony no. 7, K. 45, II, mm. 5–8.

I do not know exactly how White and Quinn inventoried the number of V and V^7 chords in the YCAC, since they do not offer many details of this sample case in their article. I can only assume they simply added up each salami-slice instance of those scale-degrees sets and then organized the results by date. With this approach, their results may not be showing an increase in the ratio of V^7 to V chords that a listener would perceive; rather, their results may be showing only an increase in the ratio of V^7 to V chords that we would find at any given instant on the highest level of the musical surface. Since thicker textures inherently have a higher probability of generating chords with higher cardinalities (i.e., the number of notes played at the same time), their results may simply be reflecting a general thickening of texture and instrumental resources over time. Indeed, it is well-known that typical ensemble sizes grew dramatically from the late 1600s to the late 1800s. It should not be surprising, therefore, that we find—on the most

immediate and superficial level of the music captured by a salami slice—more four-note dominant-functioning chords (i.e., V^7) than three-note dominant-functioning chords (i.e., V). In fact, the authors report that the increase in dominant-seventh chords mirrors a broader historical trend toward a greater proportion of seventh chords overall. This increase in surface-level seventh chords would not be surprising, given a corresponding increase in the number of instruments per work, since the odds of a composer writing a four-note chord at any given moment would presumably increase as the number of available instruments and voices increased.

FUTURE WORK

In the preceding sections, I have detailed a number of my reservations about the YCAC and its role in corpus research. I do not mean to imply, however, that I think the YCAC will not be a useful resource for empirical musicology. To the contrary, I believe the YCAC holds great potential for statistical studies of music. In its current version—the YCAC circa 2015—the corpus has many limitations, as the authors themselves state clearly. I hope that the corpus continues to be improved, with new versions released on a regular basis, each of which will presumably build upon the extensive work that has already been done. What might these improvements be? I have touched on a few suggestions already, but it seems worth discussing possible directions for future work, which I will do here.

For one, the accuracy of the encodings needs to more closely reflect the original scores. The authors mention that it has been estimated that 8% of the encoded pitches from classicalarchives.com do not match the published score (2016, p. 6). This error rate seems dangerously high, in my view. As Huron warns, a 1% pitch error rate potentially translates to a 2% error rate for intervals, a 4% error rate for chord identification, and an 8% error rate for two-chord harmonic progressions (2013). Along similar lines, the encodings should ideally, one day, include accurate meter and rhythmic information.

How can the pitch errors be corrected and meter information be added in such a large database? I cannot say exactly. It would be a lot of work for many people to go through and make these changes. Perhaps, though, the authors could post the raw MIDI data online, so that users could update the raw data before the next batch parsing. The original MIDI data was crowd-sourced, after all, so maybe a crowd-sourced approach would work here, too. Similarly, it might help if the authors released any custom computer code that parses the MIDI files into the YCAC data format. Future researchers may only want to investigate a select portion of the corpus. The YCAC would be a great starting point, upon which another user could make improvements and additions on an as-needed basis.

Along these same lines, it seems as if there should be some way for musical experts to easily repair or offer alternatives for the local key areas of the pieces. I am doubtful that a highly-accurate key-finding algorithm will be developed in the next decade (although I am hopeful), given the complexities involved with human perception. Having a "ground truth" for at least a portion of this corpus would be a valuable resource, since very few encoded harmonic analyses of entire classical pieces are available, to my knowledge. (Temperley [2009] is one possible example, except the analyses are for only excerpts, not entire pieces.) Of course, multiple analyses by different expert listeners of the same set of works would be ideal, if only to gauge the extent of ambiguity in key perception and harmonic analysis. A ground truth would also be useful to train or test key-finding algorithms.

To conclude, I do believe that big data has the potential to answer some big questions in the field of musicology. That said, I think users of the YCAC should have some big questions about any inferences drawn from its big data. The YCAC definitely has value—one that I hope increases as further improvements and refinements are made. For now, though—to put it in terms of internet slang—with regard to the YCAC, YMMV[6].

NOTES

[1] Correspondence concerning this article should be addressed to: Trevor de Clercq, Department of Recording Industry, 1301 East Main Street, Box 21, Middle Tennessee State University, Murfreesboro, TN 37132, USA, trevor.declercq@mtsu.edu.

[2] I am using the version of the YCAC that I downloaded from the YCAC website (http://ycac.yale.edu) on November 16, 2015.

[3] That being said, sometimes the offset is measured in eighth notes, presumably due to a user entering the meter of the composition incorrectly. For example, the second movement of Beethoven's piano sonata no. 26, op. 81a, is in a 2/4 meter, but the corpus encodes the piece as having four quarter notes per bar. (Each eighth note in the score is represented as an offset integer in the corpus.)

[4] The implementation of the Bellman-Budge key-finding algorithm in this corpus seems to have the tendency to take a prolonged chord as the local tonic for durations of anything more than about 8 quarter notes. This duration is the length of the authors' sliding window, so there seems to be a relationship between the window size and the accuracy of the key-finding algorithm. Indeed, Bellman (2006, p. 88) warns us that "the optimum [window] width will vary depending on the nature of the music, particularly in relation to the amount of different pitch classes present in the texture."

[5] One might wonder how and where the authors came up with this hypothetical hypothesis, as it does not remind me of any typical intuitions found in the extant music theory scholarship. My best guess is that it derives from White's findings (2013) that an *n*-gram algorithm run on the YCAC posits V as the highest ranked chord for the period 1650-1750 and V^7 as the highest ranked chord for the period 1801-1900.

[6] YMMV is the common abbreviation for "Your Mileage May Vary," typically understood to mean that the results from or utility of the thing in question will vary from user to user.

REFERENCES

Bellman, H. (2006). About the determination of key of a musical excerpt. In K. Kronland-Martinet, T. Voinier, & S. Ystad (Eds.), *Proceedings of Computer Music Modeling and Retrieval* (pp. 76-91). Heidelberg: Springer. http://dx.doi.org/10.1007/11751069_7

Budge, H. (1943). *A Study of Chord Frequencies Based on the Music of Representative Composers of the Eighteenth and Nineteenth Centuries*. Unpublished doctoral dissertation, Columbia University, New York.

de Clercq, T., & Temperley, D. (2011). A corpus analysis of rock harmony. *Popular Music*, *30*(1), 47–70. http://dx.doi.org/10.1017/S026114301000067X

Huron, D. (2013). On the virtuous and the vexatious in an age of big data. *Music Perception*, *31*(1), 4-9. http://dx.doi.org/10.1525/mp.2013.31.1.4

Temperley. D. (2007). *Music and probability*. Cambridge, MA: MIT Press.

Temperley, D. (2009). A statistical analysis of tonal harmony. Retrieved from http://theory.esm.rochester.edu/temperley/kp-stats/index.html

Temperley, D. (2012). Computational models of music cognition. In D. Deutsch (Ed.), *The Psychology of Music* (pp. 327-368). Amsterdam: Elsevier.

Temperley, D., & de Clercq, T. (2013). Statistical analysis of harmony and melody in rock music. *Journal of New Music Research*, *42*(3), 187-204. http://dx.doi.org/10.1080/09298215.2013.788039

White, C. W. (2013). An alphabet reduction for chordal *n*-grams. In J. Yust, J. Wild, & J. A. Burgoyne (Eds.), *Proceedings of the 4th International Conference on Mathematics and Computation in Music* (pp. 201-212). Heidelberg: Springer. http://dx.doi.org/10.1007/978-3-642-39357-0_16

White, C. W., & Quinn, I. (2016). The Yale-Classical Archives Corpus. *Empirical Musicology Review*, *11*(1), 50–58. http://dx.doi.org/10.18061/emr.v11i1.4958