Similarity, Data Compression and a dead composer

Koopmans, J.M.A.; van den Berg, D.; Zaytsev, V.

Publication date
2015
Document Version
Final published version
Published in
Student Undergraduate Research e-Journal
License
CC BY-SA

Link to publication

Citation for published version (APA):
Koopmans, J. M. A., van den Berg, D., & Zaytsev, V. (2015). Similarity, Data Compression and a dead composer. *Student Undergraduate Research e-Journal*, *1*(1). http://journals.library.tudelft.nl/index.php/sure/article/view/1068

Similarity, Data Compression and a Dead Composer

J.M.A. Koopmans, D. van den Berg and V. Zaytsev
FNWI, UvA, Amsterdam
jetsekoopmans@hotmail.com, d.vandenberg@uva.nl, vadim@grammarware.net

ABSTRACT

Domenico Scarlatti (1685-1759) is well-known for his 555 keyboard sonatas. Although his work is greatly revered by many professional musicians, some claim that it does not show any compository development. In this paper, his sonatas are clustered by normalized compression distance (NCD), an algorithmical similarity metric with no musical background knowledge. NCD is rooted in Kolmogorov Complexity (KC), a measure that captures the similarity between any two sonatas in a single number. The results show clusters of similar sonatas and suggest Scarlatti's work *does* show compository development, even 'milestone sonatas' marking changes in artistic style during his lifetime.

Keywords
Scarlatti, data compression, similarity, normalized compression distance, Kolmogorov complexity.

Introduction

Domenico Scarlatti (1685-1759) was an Italian composer who spent a significant part of his life in the service of the Portuguese and Spanish courts, both as a composer and music teacher for members of the royal family. Although he is often classified primarily as a baroque composer, his music is considered well influential to the development of the classical style. Here, in relative isolation from the rest of the world, he wrote his 555 piano sonatas, a tremendous number arranged and indexed by several scholars and musicologists including Ralph Kirkpatrick, who published a multi-volume edition of the sonatas in 1953 [1]. His chronological indexation, the Kirkpatrick-index is now widely used. The quality of Scarlatti's sonatas is greatly revered by many professional musicians, although some claim that through his work, one can hardly discover any progress – all of them are fundamentally the same.

In Sutcliffe's book [2], both positive and negative claims are made about Scarlatti. "Scarlatti's style is less varied and less flexible", "discussions of Scarlatti's 'seriousness'", "the consequent claims for an absolute originality", "'whirlwind lifestyle' would describe a lot of the sonatas perfectly", "a nice reminder of Scarlatti's art at a level almost unknown in the general literature", are some interesting claims found in this book. Probably the most interesting claim in this book is: "they continually threaten to float clear of him in an autistic self-sufficiency, a repetition without rationality or purpose". In addition, Sheveloff [3] claims that "Scarlatti's style is composed of 'an abundance of tiny, special details'", Owen [4] noted that "Scarlatti's sonatas are considered creative and innovative, yet somewhat formulaic" and van Schie [5] even claims "no progressive development in style". Are these claims justified? Are they based on experiences with Scarlatti's music? Or are they just arbitrary claims? This leads to the research question: To what extent will there be compository development through the work of Scarlatti that can be proven in a mathematical way?

Services like Spotify, Youtube and Last.fm are able to create playlists with music they claim to be similar. These "similarities" can be set by a human expert, comparing different tracks and subsequently labelling or tagging these. Such method requires specific knowledge of the relevant problem area and is highly subjective by any standards. For this investigation, we require the more rigorous method of Normalized Compression Distance (NCD), a practical similarity metric rooted in Kolmogorov Complexity. In this paper, we use the NCD to detect similarity between every two Scarlatti sonatas.

Kolmogorov complexity

Each sonata of Scarlatti is represented as a normalized string x over a finite alphabet by stripping the MIDI files. The minimum description length of a (bit)string x is known as its Kolmogorov Complexity [6].

Definition 1 *The Kolmogorov complexity of a string x, denoted K(x), is the length of the shortest program that describes x.*

To compare sonatas with each other, the notion of conditional Kolmogorov complexity is used.

Definition 2 *The conditional Kolmogorov complexity of string x given string y, denoted K(x|y), is the length of the shortest program which outputs x, when given y as input.*

Theoretically, when x is exactly the same as y, $K(x|y)$ and $K(y|x)$ are both very low. Because $K(x)$ has no input, it can be rewritten as $K(x|\varnothing)$, where $\varnothing$ denotes an empty input. Sadly, the (conditional) Kolmogorov complexity is not computable simply because one can never guarantee to have found the shortest program, with the exception for a very small number of trivial instances [7]. However, an approximation of the Kolmogorov complexity can be obtained by good data compressors like *bzip2*, *gzip* and *LZMA* [7, 8].

Normalized Compression Distance

To effectively detect similarities between sonatas that other effective distances, like Hamming distance, Euclidean distance, Levenshtein distance [9] and Lempel-Ziv distance [10] can detect separately, the Normalized Compression Distance is introduced [11, 12, 13]. If one object can be significantly compressed given the information contained in the other, two objects are considered to be very similar.

Definition 3 *The Normalized Compression Distance is defined as*

$$NCD(x,y) = \frac{Z(xy) - \min\{Z(x), Z(y)\}}{\max\{Z(x), Z(y)\}}. \qquad (1)$$

Here, *Z(x)* denotes the length of object *x* compressed with compressor *Z* (in this case *LZMA*) approximating the Kolmogorv Complexity. Essentially, if one sonata can be significantly compressed given the information in the other, two sonatas are highly similar. The NCD returns values between *0* and *1 + ε*, where a small constant *ε* is due to imperfections in the compression technique. The NCD is computed for each sonata with every other sonata and is represented in a similarity matrix.

To improve on the results, the MIDI files representing the sonatas are preprocessed. Besides relevant note information, the MIDI file format facilitates plenty of other data, like copyright messages or personal comments left by the editor. For means of comparability and to increase overall performance of the resemblance, this data is stripped off during preprocessing. The relevant information in a MIDI file consists of Note-On events (when a note is pressed) and Note-Off events (when a note is released). In MIDI files, time is represented in ticks. All downloaded MIDI files of Scarlatti have the standard tempo of 120 beats per minute and a resolution of 384 ticks per quarter note. To convert the ticks to music notes 1/32 is chosen as a minimal note and is used as a grid. To align each note to the closest grid line (i.e. to the closest integer multiple of the minimal note), the time (in ticks) between the Note-On and Note-Off events must be divided by 48 (because 384 ticks represent a quarter note, the minimal note 1/32 consists of 48 ticks). For example if a note of 376 ticks is divided by 48, it results in 7,83. This value is rounded to nearest integer 8, so it represents a 8/32 note (i.e. a quarter note).

The NCD-approach to find the amount of similarity between pairs of objects (in our case sonatas) has previously been applied successfully to a broad range of domains. It works well on various concrete examples like detecting plagiarism in student programming assignments [14], OCR [8], a completely automatic construction of a language tree for over 50 Euro- Asian languages [13] and clustering of music [15]. The NCD is used to cluster Scarlatti's sonatas as MIDI files, so that any compository development can be proven. For this purpose, all 555 sonatas are downloaded as MIDI files from *Kunstderfuge.com*, "the largest classical midi resource on the web". To visualize the clusters of Scarlatti's sonatas, a dendrogram is constructed.

Compression techniques compared

The effectiveness of a compression technique critically depends on its intended application. For this reason, several compression techniques are compared by compressing each of Scarlatti's 555 sonatas individually. The graph in figure 1 shows the *bzip2*, *gzip* and *LZMA* techniques as three *Python* modules: *bz2*, *zlib*, and *PyLZMA*. The higher the space savings, the more effective the compression for our purpose, and the better the Kolmogorov complexity is approximated. In our case, the *PyLZMA* module clearly compresses best.

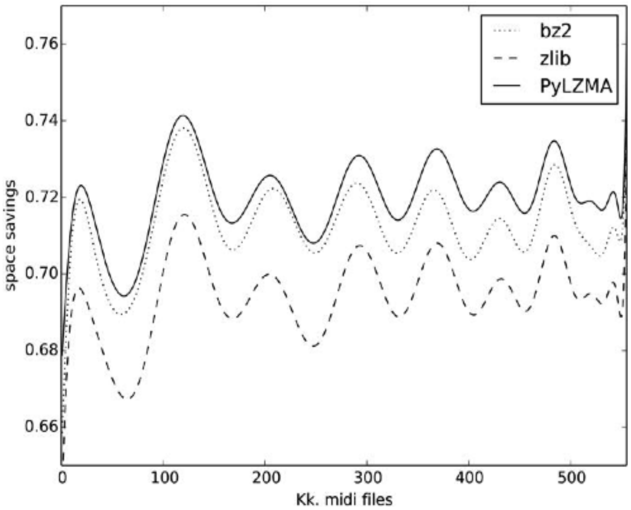

Figure 1: effectiveness of *bz2*, *zlib* and *PyLZMA* compression modules compared on scrubbed midi files of Scarlatti's 555 keyboard sonatas, as indexed by Kirkpatrick

Clustering

Clustering of the NCD values to construct a dendrogram can be done in many different ways. Hierarchical clustering using an agglomerative strategy [16] is often used and there is *Python* library with its implementation. Each sonata starts in its own cluster and two clusters are merged as one moves up the hierarchy. The three most popular methods to compute the distance between two clusters and eventually merge them are single, complete and average linkage [16]. Single linkage clustering sets the distance between two clusters C_1 and C_2 as the shortest findable NCD between two sonatas, one from C_1 and another from C_2. Conversely, complete linkage clustering sets the difference by finding the two sonatas (one from C_1, the other from C_2) that are *furthest* apart in terms of NCD. Average linkage clustering is in some sense a compromise between the two; for all pairs of sonatas spanning C_1 and C_2, the NCD is calculated and then averaged over all pairs, setting the distance between the two clusters.

Two controlled experiments

Before experimenting with Scarlatti's sonatas, the NCD-based clustering method was tested. To do this, two controlled experiments were conducted in situations in which the correct outcome was known in advance.

In the first experiment, twelve jazz pieces, twelve rock pieces and twelve classical pieces were used. Clustering done with the complete linkage method gave the best results. In figure 2, the jazz pieces, rock pieces and finally classical pieces are well clustered but a few classical pieces were erroneously clustered with rock songs. The exact reason is unknown, though our suspicion is that this might be due to relatively rich harmonic diversity. Colloquially speaking, Schumann's *Kinderszenen* might be more akin to Nirvana than to Bach's *Wohltemperirte Clavier*, because both consist of a few closely related chords only.

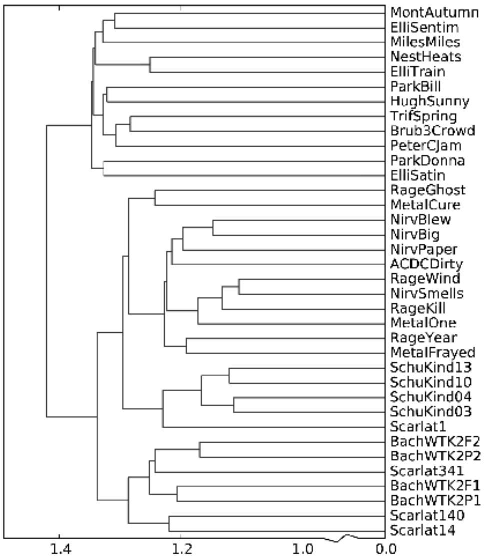

Figure 2: dendrogram of music genre classification

The second experiment is somewhat more specific as it tries to distinguish between three classical music composers: J.S. Bach, D. Scarlatti and R. Schumann. The complete linkage method yields a dendrogram (figure 3) in which each composer clearly has his own cluster, suggesting this method based on NCD works quite well.

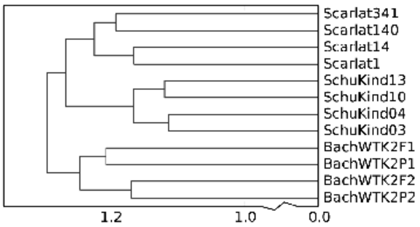

Figure 3: dendrogram of classical music classification

Clustering Scarlatti's 555

Each of the 555 sonatas is compared with every other sonata using the NCD. Because the *LZMA* compression technique is slightly non-symmetrical, meaning compression efficacy may depend on which sonata in the pair comes first, every pair is compressed in reverse order as well. Both the complete similarity matrix and the clustered dendrogram of the 555 sonatas look somewhat intimidating, but a close inspection reveals the method recognizes musical similarities quite well (figure 4).

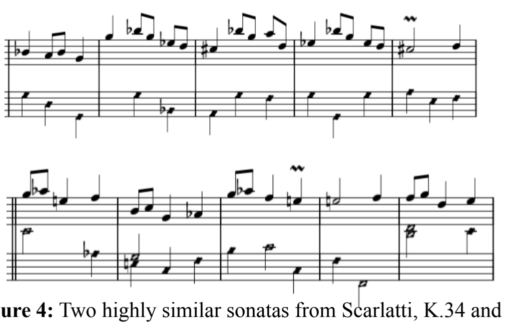

Figure 4: Two highly similar sonatas from Scarlatti, K.34 and K.40, as detected by our NCD-method.

Compository development

The notion "compository development" can be defined based on changes in style through the lifetime of the composer. If a composer (or any artist for that matter) develops his or her style, their later works are different from earlier works, regardless of whether one considers it an improvement or not. An explicit definition of the notion of compository development is given with regard to Scarlatti's sonatas.

Definition 2.6 *Compository development is encountered at sonata k if the average NCD of sonata k and each of its previously composed n sonatas is greater than a constant value c. Mathematically speaking, if the following expression evaluates to true for k − n > 0,*

$$\frac{1}{n}\sum_{i=k-n}^{k-1}(NCD(X_k, X_i)) > c \tag{2}$$

Here, X is the collection of Scarlatti's 555 sonatas in preprocessed MIDI format. Essentially, if there is little to no difference between the kth sonata and each of its *n* previously composed sonatas according to Ralph Kirkpatrick's chronological indexation, there will be no compository development across these n sonatas. To estimate whether compository development occurs throughout Scarlatti's oeuvre, equation 2 will be used for each sonata relative to a number of previously composed sonatas.

The amount of compository development can be measured by means of equation 2. The higher the value of *c,* the less similar a certain sonata is compared to its predecessors. Being less similar signifies change, or compository development, though we do not venture to verdict on the quality of such developments. In figure 5 a least squares polynomial fit is generated of the marks, each representing compository development at that sonata. The higher the mark, the stronger the compository development is. Interesting are the peaks in the line around sonatas K.40, K.100 and K.500. This might indicate an increasing amount of compository development. The 'milestone sonatas', i.e. the sonatas which are compository the most different from earlier work, are sonatas K.40, K.100, K.200 and K.410.

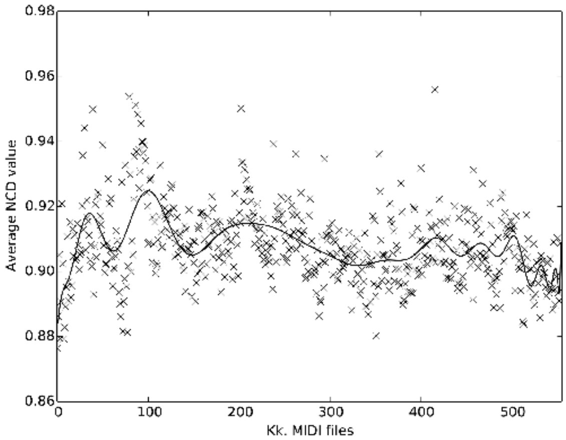

Figure 5: Every mark represents an amount of compository development at the sonata on the x-axis.

Conclusion

This paper gives some computational comment on the question "To what extent is compository development found in Scarlatti's oeuvre?". His sonatas were clustered by means of NCD, a method proven to successfully classify music files in genres and even individual composers. The sonatas in the acquired dendrogram of the 555 sonatas appear clustered, but in a practically random chronological sequence; this way it is hard to prove compository development. Compository development could however be detected by the inverse: first ordering the sonata's chronologically according to Kirkpatrick and subsequently comparing each sonata with previous compositions. Several of Scarlatti's sonatas signify increased changes in style, most notably sonatas K 10, K.120 and K.410, which can therefore be considered as 'milestone sonatas'.

Although artistic debates on composition, style and execution of Scarlatti's sonatas are likely to continue, it is now scientifically speaking quite hard to maintain that Scarlatti's work shows "no progressive development in style" [5] though it must be said that in a legacy as massive as Scarlatti's, there is bound to be some repetition and similarity too.

Related and future work

In music similarity, three computational factors play a central role: (1) the music content, i.e. the audio signal itself, (2) the music context, i.e. metadata in the widest sense and (3) the listeners and their contexts, manifested in user-music interaction traces [17]. In this paper, the music content factor is derived directly from the preprocessed MIDI files by using the LZMA compressor and the complete linkage method. To substantiate or refute the obtained results, it might be interesting to look at other ways of classification (and thus recognize other patterns), for example n-gram distribution of notes [18], Daubechies Wavelet Coefficients histogram [19] and perceptually weighted Euclidean distance [20]. Nebel, Hammer and Villmann [22] have used the NCD to measure the mutual dissimilarities of five composers, including Scarlatti. The second factor, the music context refers to information that is not encoded in the audio file, for example the meaning of song lyrics, background of an artist and the cover of an album. This paper does the opposite: the MIDI files are preprocessed to remove the music context from the sonatas for means of comparability. Third, three computational features can be defined that describe a listener's music taste: diversity, mainstreaming and novelty of the listener's music taste [21].

Role of the student

Jetse Koopmans was working under the supervision of Daan van den Berg (UvA) and Vadim Zaytsev (UvA) during this research. The topic was proposed by the supervisors. The paper was written by the first author.

REFERENCES

[1] R. Kirkpatrick. Domenico Scarlatti. Princeton University Press, 1953.

[2] W.D. Sutcliffe. The Keyboard Sonatas of Domenico Scarlatti and Eighteenth-Century Musical Style. Cambridge University Press.

[3] Sheveloff. Keyboard. 1970, p. 258.

[4] S.R. Owen. On the Similarity of MIDI Documents. Harvard College, 2000, pp. 40–41.

[5] T. van Schie. Enige gedachten bij de Sonates van Scarlatti. 1988. (http://www.tjako.nl/essay_scarlatti.html), consulted Oct 16th , 2015

[6] M. Li and P.M.B. Vitány. An Introduction to Kolmogorov Complexity and its Applications. Springer Verlag, New York, 2nd Edition, 1997.

[7] M. Koucký. A Brief Introduction to Kolmogorov Complexity. MÚ AV ČR, Praha, 2006, p. 4.

[8] R. Cilibrasi and P.M.B. Vitányi. "Clustering by compression". In: Information Theory, IEEE Transactions on 51.4 (Apr. 2005), pp. 1523–1545.

[9] K. Orpen and D. Huron. Measurement of similarity in music: A quantitative approach for non-parametric representations. Computers in Music Research 4, 1992.

[10] G. Cormode, M. Paterson, S. Sahinalp and U. Vishkin. "Communication complexity of document exchange". In: Proc. 11th ACM-SIAM Symposium on Discrete Algorithms (2000), pp. 197–206.

[11] M. Li, J.H. Badger, X. Chen, S. Kwong, P. Kearney and H. Zhang. An information-based sequence distance and its application to whole mitochondrial genome phylogeny. Bioinformatics, 17(2).

[12] M. Li and P.M.B. Vitány. "Algorithmic complexity". In: International Encyclopedia of the Social & Behavioral Sciences (2001), pp. 376–382.

[13] M. Li, X. Chen, X. Li, B. Ma and P.M.B. Vitány. The similarity metric. Proc. 14th ACM-SIAM Symposium on Discrete Algorithms, 2003, pp. 863–872.

[14] X. Chen, B. Francia, M. Li, B. McKinnon and A. Seker. "Shared information and program plagiarism detection". In: Information Theory, IEEE Transactions on 50.7 (July 2004), pp. 1545–1551.

[15] R. Cilibrasi, P.M.B. Vitányi and R. de Wolf. "Algorithmic Clustering of Music Based on String Compression". In: Computer Music Journal 28.4 (Dec. 2004), pp. 49–67.

[16] A. El-Hamdouchi and P. Willett. "Comparison of hierarchic agglomerative clustering methods for document retrieval". In: The Computer Journal 32.3 (June 1989), pp. 220–227.

[17] P. Knees and M. Schedl. Music Retrieval and Recommendation: A Tutorial Overview. In Development in Information Retrieval (2015), pp. 1133-1136.

[18] V. Kumar, H. Pandya and C.V. Jawahar. Identifying Ragas in Indian Music. In Pattern Recognition (ICPR), 2014 22nd International Conference on (2014), pp. 767-772.

[19] T. Li, M. Ogihara and Q. Li. A Comparative Study on Content-Based Music Genre Classification. In Development in Information Retrieval (2003), pp. 282-289.

[20] U. Simsekli. Automatic Music Genre Classification Using Bass Lines. In Pattern Recognition (ICPR), 2010, pp. 4137-4140.

[21] M. Schedl and D. Hauger. Tailoring Music Recommendations to Users by Considering Diversity, Mainstreaminess, and Novelty. In Development in Information Retrieval (2015), pp. 947-950.

[22] D. Nebel, B. Hammer and T. Villmann. About Learning of Supervised Generative Models for Dissimilarity Data. Machine Learning Reports (2013), pp, 1–19.