

圧縮類似度による楽譜からの作曲者の判定

足達 花絵[†] 岡部 正幸[‡] 梅村 恭司^{††}

[†] 豊橋技術科学大学 情報工学系 〒441-8580 愛知県豊橋市天白町雲雀ヶ丘 1-1

[‡] 豊橋技術科学大学 情報メディア基盤センター 〒441-8580 愛知県豊橋市天白町雲雀ヶ丘 1-1

^{††} 豊橋技術科学大学 情報・知能工学系 〒441-8580 愛知県豊橋市天白町雲雀ヶ丘 1-1

E-mail: [†] adachi@ss.cs.tut.ac.jp, [‡] okabe@imc.tut.ac.jp, ^{††} umemura @tut.jp

あらまし 楽曲からの作曲者などの分類, 判定をコンピュータが自動的に行うことは, 単なる楽曲の類似判定にくらべ, はるかに難しい問題である. 本研究では, Keogh らが提案する, Compression-based Dissimilarity Measure(CDM) [1]を用いて, 楽譜からの作曲者の判定を行う. CDM とは, 文字列化した時系列データの圧縮を用いて, 様々な時系列データを類似判定するための尺度である. この尺度を使うとき, 楽曲をどのように文字列化するか重要な問題であり, また, CDM は圧縮プログラムの選択によって, その性能が変わる. ここでは, 作曲者 5 人, 各 10 曲のピアノ曲の楽譜データで, 判定が可能な文字列化の方法. 及び, 既存の圧縮方式である bzip2, gzip, ZIP を用いて, 類似判定を行った結果を報告する.

キーワード 圧縮類似度, 音楽情報

1. はじめに

人間は, 楽曲を聴いたとき, その曲の音使いや曲調などから, その楽曲の部分部分の特徴を総合して, 作曲者の作風を認識することが可能である. 一方, このような認識をコンピュータが自動的に行うことは, 楽曲そのものの類似判定を総合しなければならず, 楽曲の判定に比べ難しい問題である. ジャンルの判定は楽曲の和声進行やリズムを総合するアプローチが可能であるが, 作曲者の作風はジャンルに比べより詳細な音の進行を総合的にとらえる必要がある. 本研究では, 文字列の変遷を情報量の見地から判定する類似度として Keogh らが提案する, Compression-based Dissimilarity Measure(CDM)[1]を利用して楽譜からの作曲者の判定を行う. CDM とは, 文字列化された時系列データの圧縮を用いて, 文字列化されたデータ間の時系列情報を類似判定するための尺度であり, 多くの時系列データにおいて, 生成の由来を判定するのに性能が高いということが知られている類似度である. この尺度を使用するとき, 楽曲をどのように文字列化するか重要な問題である. ここでは, 作曲者 5 人, 各 10 曲のピアノ曲のデータに対し, 1 対 1 で作曲者の判定が可能な文字列化の手法. その手法を用いて判定した結果, 時代の大きく異なる作曲家同士においては, 判定が可能であった. また, CDM は圧縮プログラムの選択によって, その性能が変わる. 既存の圧縮方式である bzip2, gzip, ZIP を用いて類似判定を行った結果を報告する.

2. Compression-based Dissimilarity Measure(CDM)

本研究では, 圧縮による類似度として, Keogh らが提案する Compression-based Dissimilarity Measure(CDM) [1]を用いる. CDM は式(1)により求められる.

$$CDM(x,y) = C(x,y) / (C(x) + C(y)) \quad (1)$$

- x, y : 文字列
- $C(x), C(y)$: 文字列 x, y をそれぞれデータ圧縮したファイルサイズ
- $C(x, y)$: 文字列 x, y を連結し, データ圧縮したファイルサイズ

時系列データ同士が類似しているほど, 連結し圧縮したデータサイズ $C(x, y)$ は, 各々に圧縮したデータサイズを足した $C(x) + C(y)$ より, 圧縮に必要な辞書などの情報を共有するため CDM は小さい値となる. このとき, 文字列化されたデータを圧縮するときに検出される文字列のパターンが共通であるかどうかを判定することになる. 楽曲に利用するときには, 音の遷移のパターンが共有されるかを検出する類似度として使える可能性がある. そして, 我々は音の遷移のパターンの分布に作風があるという仮説をもっている. なお, CDM は圧縮プログラムの方式によって, 認識する共通パターンが変化し, 結果として, 性能が変化するという特徴がある.

本実験では作曲者の判定のために K-NN(k 近傍法)を

使用した．K-NN とは，特徴空間における最も近い訓練例に基づいた統計分類の手法である．以下に，本研究で用いた K-NN を用いたクラスが未知のデータのラベルの決定方法の手順を説明する．決定した未知データのラベルをここでは，解答とする．

- ① クラスが未知であるデータと全ての既知のデータとの CDM を算出し距離とする．
- ② クラスが未知でありデータと最も距離が近いデータを順に k 個取り出し多数決を取る．

図 3 の例では，k=3 の場合，Mozart が 2 つ，Satie が 1 つとなり未知データは Mozart とラベル付けされる．本研究では以上の方法によって決定されたラベルを解答とする．

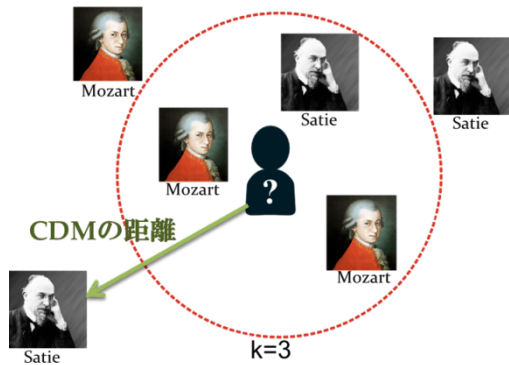


図 3：K-NN の例(Mozart-Satie)

3.3. 実験

作曲家 5 人，各 10 曲ずつ楽譜データを用意し，ある曲が 2 人の作曲家のうち，どちらの作曲した曲であるかを判定する実験を行った．圧縮方法は，先行研究で良い結果を得ることができた bzip2 を使用した．

まず，作曲家を 2 名選ぶ．次に，ある作曲家のある曲の一つ選ぶ．その曲の作曲者が不明であると仮定して，その曲と作曲者の分かっている残りの 19 曲との CDM の距離を計算し，距離の近い 3 曲を求める．その 3 曲の作曲者のうち多い作曲者を，選んだ曲の作曲者と推定する．こうすると推定結果の判定が 20 個できるので，これを正解が偶然以上に多いかを判定する．この手続きを図 4 に示す．実験に使った曲のリストについては，付録として示す．

この例においては，Mozart と Satie を作曲家に選び，それぞれの楽曲で CDM 距離の小さかった 3 曲で作曲家の多数決をした場合に，真の作曲家と同じになるかを判定した結果，正解が 17 で不正解が 3 となる．判定が偶然レベルであるときに，これ以上に差が存在する確率は，0.0012 以下であることが符号検定で示される．これを標準的な有意水準の値と比較した．



図 4：2 組の作曲家についての有意水準の計算

3.4. 考察

表 3 は，この実験を 5 人の作曲家の間で総当たりを行い，その符号検定の有意水準を求めた結果である．Mozart-Satie，Bach-Debussy，Bach-Satie では有意水準 1% で CDM を用いて作曲者の判定能力があると言える．

また，作曲家の判定が出来なかった作曲家の組合せは，Bach-Mozart，Mozart-Chopin，Mozart-Debussy，Debussy-Satie などである．作曲家の活躍した時代を比べると，Bach(1685-1750)，Mozart(1756-1791)，Chopin(1810-1849)，Debussy(1862-1918)，Satie(1866-1925) となっており，判定可能であった作曲家同士は，時代が大きく異なることがわかる．時代が似ていれば，異なる作曲家であっても作風が似ることは想定され，作曲家の判定が難しくなることは合理的な結果と考えられる．一方で，わずかに，20 曲のデータのなかでありながら，Mozart と Satie のように十分に有意な結果が得られることは，作曲家の判定という難しい問題への端緒としては興味深い結果が得られたといえる．

表 3：符号検定の有意水準

					
Bach		0.7483	0.2517	0.005909	0.005909
Mozart			0.4119	0.2517	0.001288
Chopin				0.13159	0.02069
Debussy					0.2517
Satie					

本研究で用いた圧縮類似度，Compression-based Dissimilarity Measure (CDM) [1]は圧縮プログラムの選択によって，その判定精度が異なる．菊池ら[9]の先行研究では，最も良い結果を得られた圧縮方法は bzip2 であった．しかし，3.1 の文字列化手法での楽譜からの作曲者の判定に適した圧縮方法について，まだ明らかでない．よって，既存の圧縮方式である bzip2, gzip, ZIP の 3 つの圧縮方法で比較を行い，3.1 の文字列化の手法での良い圧縮プログラムの検討を行う．

ここで問題となるのは，圧縮プログラムが統計的な解析をする文字列の長さである，本論文での実験では，ある瞬間の状態が 88 文字で表現され，音符が 5 つで構成されるような単純な 1 フレーズに対応する文字列の長さは 400 文字以上の長さになる．このような長い文字列についても，それを分析の対象として圧縮することができないと，今回のタスクのための圧縮プログラムとして適さない．

4.1. 圧縮方法

本実験使用した圧縮方法について以下に示す．

(1) bzip2

bzip2[3]は Julian Seward により開発されたオープンソースのデータ圧縮アルゴリズム．より効率的な圧縮のためにブロックソート法，MTF(Move-To-Front)法，ハフマン符号化を用いた圧縮方法．gzip や ZIP と比べ圧縮率が高い．

ブロックソート法は，1994 年に M.Burrows と D.J.Weeler によって提案された変換を使う．この変換結果が繰り返しのある文字列を認識する機能を提供する．変換のときの並びかえが辞書と等価であり，並び替えに使う文字列の長さに制限がないので，結果として，圧縮の際の辞書の長さに制限がない．

(2) gzip

gzip[4]は，Jean-Loup Gailly と Mark Adler により開発されたデータ圧縮アルゴリズムである．LZ77 法[12][13](正確にはその変形版である LZSS 法[12][13])とハフマン符号化を組み合わせた可逆データ圧縮アルゴリズムである Deflate[6]アルゴリズムが用いられている．Deflate アルゴリズムは，可逆圧縮，圧縮速度が比較的高速，展開が非常に高速，パテントフリーであるという特徴がある．LZ77 法[12]は，データを以前出現したデータの中から最も一致する部分を指すことによって圧縮する方法である．

現在，Deflate で使用されているのは，LZ77 法を基にした LZSS 法である．LZSS 法では，今見ている位置を先頭として，固定長のスライド窓と呼ばれる過去バ

ッファの中で一致している部分を探し，その場所の位置と一致長を出力する．もし一致していないならば，その代わりに文字を出力する．現在の Deflate のスライド窓は 32[Kbyte]とされている．

(3) ZIP

ZIP[2]は，Phil Katz が考案，開発したデータ圧縮形式，及びファイルフォーマットである．複数の圧縮，あるいは非圧縮のファイルデータを集め 1 つのアーカイブ・ファイルに格納するためのファイルフォーマットで，「圧縮または非圧縮のファイルデータ 1 つ分」をヘッダとフッタで挟んだものを一単位とし，それを複数つなげ，その末尾にアーカイブ全体のメタ情報を含んだフッタ情報を置く．

また，ZIP は，gzip 同様，LZ77 法とハフマン符号化を組み合わせた可逆データ圧縮アルゴリズムである Deflate[6]アルゴリズムが採用されている．

4.2. 実験結果

表 4，表 5，表 6 は，bzip2, gzip, ZIP の 3 つの圧縮方法で作曲者を判定したときの有意水準である．

圧縮方法に bzip2 を用いたときは，Mozart-Satie, Bach-Debussy, Bach-Satie の 3 つの場合において有意水準 1%で判定が可能である．しかし，gzip, ZIP では，判定可能な作曲者がいなかった．

4.3. 考察

CDM の文献[1]においては，使用する圧縮方法についての言及はなかったが，それは使用する文字列は，2 種類の文字ではなく，多数種類の文字種からなる文字列となり，長さについては，10 文字程度の状態であったからであろう．

本研究で用いた楽譜の文字列化の手法では，圧縮の際，必要な辞書の長さが長くなる．そのため，gzip や ZIP と違い辞書の長さに制限のない bzip2 で最も良い正解率になったと考えられる．先行研究[12]においても，場の状態を表現する文字は 2 種類であり，今回の実験と同様な性質があったためと推測できる．

CDM は有用な距離尺度であるが，それを利用する場合に利用する圧縮プログラムの性質が大きく関わること、および、圧縮するときに分析するための文字列の長さの上限が、想定するパターンの長さに関係する事を示したのは，関連研究[14]と同様に，興味深い実験結果である．

表 4 有意水準(bzip2)

bzip2	Bach	Mozart	Chopin	Debussy	Satie
Bach		0.74828	0.25172	0.00591	0.00591
Mozart			0.4119	0.25172	0.00129
Chopin				0.13159	0.02069
Debussy					0.25172
Satie					

表 5 有意水準(gzip)

gzip	Bach	Mozart	Chopin	Debussy	Satie
Bach		0.74828	0.13159	0.13159	0.05766
Mozart			0.5881	0.5881	0.74828
Chopin				0.25172	0.5881
Debussy					0.5881
Satie					

表 6 有意水準(ZIP)

zip	Bach	Mozart	Chopin	Debussy	Satie
Bach		0.5881	0.5881	0.5881	0.86841
Mozart			0.5881	0.5881	0.86841
Chopin				0.5881	0.5881
Debussy					0.5881
Satie					

5. 変化点を取ることの妥当性の検証

文字列化を行うときに、変化点を取るのがよいのかどうかは明らかでないという議論があった。変化点以外のものをそのまま扱うことで、分析する文字列の長さはさらに長くなるが、bzip2 では長さの問題は生じない。そこで、bzip2 を用いて、変化点以外の情報をふくめて実験を行った結果を表 7 に示す。

一見して、5%有意の判定の場所が消滅し、システムの性能が低下しているのが分かる。従って、変化点だけをとりることが妥当であることが分かる。

表 7 有意水準(bzip2 で全情報利用)

	Bach	Mozart	Chopin	Debussy	Satie
Bach		0.2517	0.0577	0.1316	0.2517
Mozart			0.1316	0.4119	0.2517
Chopin				0.2517	0.0577
Debussy					0.0577
Satie					

6. おわりに

本稿では、Keogh らが提案する類似度、Compression-based Dissimilarity Measure(CDM)[1]を用いた、作曲者の判定のための楽譜の文字列化の手法の提案。及び、5 人の作曲者を対象に楽譜からの作曲者の判定を行った。提案する文字列化の方法は、時代が

大きく異なる作曲者同士では判定が可能であることを確認した。また、CDM は、使用する圧縮方法によって、その判定精度が異なる。そのため本研究の手法において比較的良い結果を得ることができる圧縮方法の検討を行った。その結果、良い結果を得られる圧縮方式が先行研究と同様に bzip2 であることが確かめられた。

しかし、作曲者の数が 5 人と少ない。また、時代の近い作曲者同士の判定では良い結果は得られなかった。よって、今後の課題としては、判定精度の向上のために、楽譜の文字列化の手法の検討をすること。また、本研究では楽譜から作曲者の判定を行ったが、実際の演奏による MIDI データなどノイズの多いデータでの判定を可能にすることなどが挙げられる。

参 考 文 献

- [1] Eamonn Keogh, Stefano Lonardi, Chotirat Ann Ratanamahatana, Towards Parameter-Free Data Mining: Proceedings of the tenth ACM SIGKDD, pp. 206-215, 2004
- [2] Info-ZIP. <http://www.info-zip.org/>
- [3] The bzip2 and libbzip2 official home page. <http://www.bzip.org/>
- [4] The gzip home page. <http://www.gzip.org/>
- [5] RFC 1950(zlib). <http://www.ietf.org/rfc/rfc1950.txt>
- [6] RFC 1951(Deflate). <http://www.ietf.org/rfc/rfc1951.txt>
- [7] RFC 1952(gzip). <http://www.ietf.org/rfc/rfc1952.txt>
- [8] 岡之原大輔, 藤本健, 小川秀明: よくわかる最新データ圧縮技術の基本と仕組み.
- [9] 菊地誠, 阿部洋丈, 岡部正幸, 梅村恭司: Compression-based Dissimilarity Measure (CDM) を用いた人感センサ情報の類似判定, 情報学ワークショップ 2009(WINF2009) 論文集, (2009), 185-188.
- [10] 菊池誠, 梅村恭司: 多チャンネル焦電型センサ出力における圧縮プログラムを用いた類似判定に関する研究 修士学位論文.
- [11] 後藤真孝, 緒方淳: 音楽・音声の音響信号の認識・理解研究の動向 コンピュータソフトウェア(日本ソフトウェア科学会論文誌), Vol.26, No.1, pp.4-24, February 2009. http://staff.aist.go.jp/m.goto/PAPER/JSSSTC_S200902goto.pdf (2012/12/8)
- [12] 広井誠. レンジ・コードを用いて圧縮率を改善する LZ77 符号によるファイルの圧縮とその改良(前編). Interface. Vol. 32, No. 6, pp. 182-191, 2006
- [13] 広井誠. レンジ・コードを用いて圧縮率を改善する LZ77 符号によるファイルの圧縮とその改良(後編). Interface. Vol. 32, No. 7, pp. 142-154, 2006
- [14] 村上雅紀, 岡部正幸, 梅村恭司: 赤外線センサによる行動予測の実現性のためのデータ照合、DEIM2011 B9-3.

付録

下記に実験に利用した曲名のリストを記述する。

データ名	タイトル
Bach1	シャコンヌ
Bach2	フランス組曲 第5番 ガボット
Bach3	G線上のアリア
Bach4	2声のインベンション第1番
Bach5	2声のインベンション第8番
Bach6	カンタータ第140「目覚めよと呼ぶ声あり」
Bach7	平均律クラヴィア集第2番
Bach8	平均律クラヴィア集第3番
Bach9	平均律クラヴィア集第6番
Bach10	平均律クラヴィア集2 第1番
Chopin1	エチュード Op.10-3「別れの曲」
Chopin2	エチュード Op.10-4
Chopin3	エチュード Op.10-5「黒鍵」
Chopin4	エチュード Op.10-12「革命」
Chopin5	エチュード Op.25-9「エオリアン・ハーブ」
Chopin6	エチュード Op.25-12「大洋」
Chopin7	ワルツ第1番 Op.18「華麗なる大円舞曲」
Chopin8	ポロネーズ第6番 Op.53「英雄ポロネーズ」
Chopin9	マズルカ第5番 Op7-1
Chopin10	ワルツ6番 Op.64-1「子犬のワルツ」
Debussy1	前奏曲集第1巻より「亜麻色の髪の乙女」
Debussy2	アラベスク1番
Debussy3	アラベスク2番
Debussy4	美しき夕べ
Debussy5	前奏曲集第1巻より「沈める寺」
Debussy6	子供の領分より「ゴリウオーグのケーキウォーク」
Debussy7	小さな黒人
Debussy8	ベルガマスク組曲「パスピエ」
Debussy9	ベルガマスク組曲「プレリュード」
Debussy10	夢
Satie1	快い絶望(1906~13年時期の6つの作品)
Satie2	幻想ワルツ
Satie3	グノシエンヌ第1番
Satie4	グノシエンヌ第2番
Satie5	グノシエンヌ第3番
Satie6	ジュ・トゥ・ヴ(おまえが欲しい)
Satie7	金の粉
Satie8	貧しき者の夢
Satie9	空想歌(くぼんだ夢)
Satie10	優しく
Mozart1	アヴェ・ヴェルム・コルプス
Mozart2	戴冠式ミサより「アニヌス・デイ」
Mozart3	春へのあこがれ
Mozart4	トルコマーチ
Mozart5	ウィーンのソナチネ第1番 第4楽章 K.439b
Mozart6	ピアノソナタ イ短調 1楽章 K.310
Mozart7	歌劇「ドン・ジョヴァンニ」よりツェルリーナのアリア「ぶってよ、マゼット」
Mozart8	ピアノのためのフーガ ト長調 kv375g
Mozart9	ピアノソナタ第15番 ハ長調 第1楽章 K.545
Mozart10	メヌエット kv2