時系列データにおける情報量計算法を用いた類似度と圧縮類似度との性能比較

高本　綺架†　　吉田　光男††　　梅村　恭司††

† 豊橋技術科学大学 情報・知能工学専攻　〒 441–8580 愛知県豊橋市天伯町雲雀ヶ丘 1-1
†† 豊橋技術科学大学 情報・知能工学系　〒 441–8580 愛知県豊橋市天伯町雲雀ヶ丘 1-1
E-mail: †a153350@edu.tut.ac.jp, ††yoshida@cs.tut.ac.jp, †††umemura@tut.jp

あらまし　時系列データの処理において，処理に使う類似度は処理結果の品質を決定する．我々は，Compression-based DissimilarityMeasrue（CDM）に着目し，それを改良することを行ってきた．CDM は情報量計算に関連するという考えのもと，文字列の情報量を計算する方法を提案し，実験を行ってきた．これまでの改良では，楽曲の作曲者判定という分類問題で性能を測定してきたが，本稿では CDM の提唱者が用意した時系列データで，分類を行う際に提案手法を使用し CDM と提案方法で判定の正解数を比較した．その結果，これらデータにおいても我々の提案する情報量計算による手法のほうが優れているという結果を得た．

キーワード　時系列データ，圧縮プログラム，情報量，Compression-based Dissimilarity Measure

1 はじめに

本論文では，時系列データ処理の分類問題を扱う。本研究では Keogh らの用意したデータ [1] を用いて，時系列データの分類の性能を Keogh らの提案する手法と比較する．Keogh らは比較的短いデータにおいては，Dynamic Time Warp(DTM) [2] を推奨し，長いデータでは，Compression-based Dissimilarity Measure (CDM) [3] を使うことを推奨している．まず，CDM では時系列データを Symbolic aggregate approxomation (SAX) [8] を用いて文字列化する前処理をする．CDM は類似している 2 つの文字列は同時に圧縮するほうが別々に圧縮するよりも圧縮率が高いという原理に基づいている．言い方をかえると，圧縮に使える頻出部分文字列を特定することで，類似度を測る方法である．我々は作曲者ごとに好みのフレーズがあるのではという考えのもと，好みのフレーズが頻出文字列になることを想定し，楽曲の作者の判定というタスクにおいて CDM を使用し，その改良を続けてきた．作曲者判定のタスクについては，CDM と同様に圧縮プログラムを使った別の類似度 NCD [4] を用いた報告があり [5], これも CDM が作曲者判定に有望であると考えた理由となっている．我々は，この作曲者判定というタスクにおいて，CDM で判定を行ったものをベースラインとした分類問題において，CDM よりも良い振る舞いをする手法を報告した [6]．この方法は，情報量の観点から厳密に定式化されているので，CDM よりも再実験をおこなうことが容易であるという特徴もある．さらに，この方法では，CDM ではできなかった特徴選択などの操作ができ，それが有用であるケースがあることも [7]．これは，CDM ではできなかった機能である．

我々の提案する手法は，与えられた文字列についてある文字列分割を考え，その分割した文字列の確率を推定し，その情報量を計算する．その合計を元の文字列における情報量とする方法である．そして，分割については可能な分割のなかで最小の情報量（最大の確率）を与える分割を選ぶ．この方法は音楽に依存しないため，音楽でない時系列データに対して CDM よりも性能が良くなる可能性が高いと考えた．

そこで，我々は提案手法が CDM の提唱者らが提供するの時系列データに対して分類タスクを実行した．実験には UCR Time Series に掲載されているデータの中から 50 種類のデータを選出し，CDM と提案手法を用いてクラス分類を行い，性能を比較した．その結果，CDM に対して提案手法の性能が高いことがわかった．

本稿における貢献は，分類タスクにおいて情報量を求めることによる提案手法が，CDM より多くの時系列データの処理において良い性能を示すことを明らかにしたことである．

2 関連研究

2.1 Compression-based Dissimilarity Measure (CDM)

Compression-based Dissimilarity Measure (CDM) は Keogh らの提案するパラメータを必要としないデータマイニングアルゴリズムの一つである．CDM は以下の式で定義される．

$$\frac{C(xy)}{C(x)+C(y)} \tag{1}$$

式 1 において，x，y は対象となる文字列であり，2 つの文字列を連結したものが xy である．また，$C(x)$，$C(y)$，$C(xy)$ はそれぞれの文字列を圧縮した際のファイルサイズである．文字列 x と文字列 y が類似しているならば，CDM の値は小さくなり，逆に 2 つの文字列に関連がなければ CDM の値は 1 に近付く．

CDM はパラメータを必要としない類似尺度であるが，実際には文字列を圧縮する際に用いる圧縮プログラムによって性能が変化する．本研究では，分析のパターンの長さに上限のない

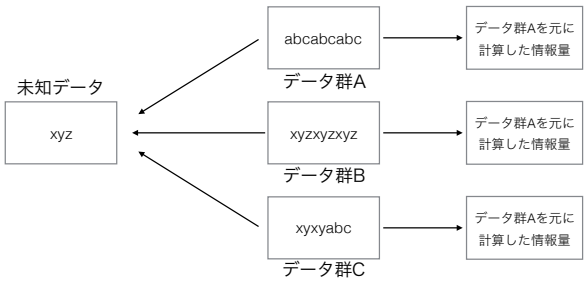

図 1　情報量計算の例

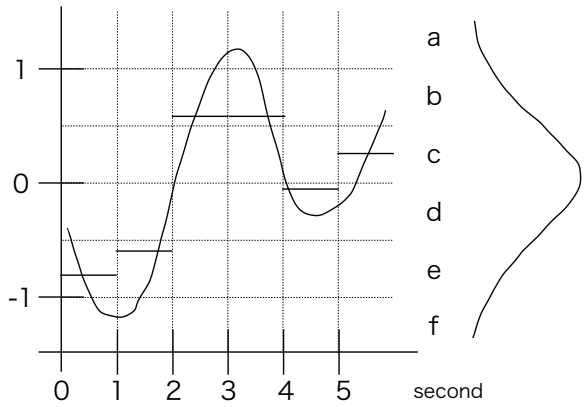

図 2　SAX を用いた時系列データ変換例

方法とするため，文字列の圧縮に bzip2 を用いる．

2.2　情報量計算手法

これまでの研究として，文字列が持つ情報量を情報理論に基づいて計算する手法が提案されている．一般に文字列が持つ情報量は 1 文字あたりの情報量の合計として計算される．長さ N の文字列 S が持つ情報量を，文字をベースとして計算する場合の情報量 I_c は式 2 のように表せる．

$$I_c(S) = -\log_2(\prod_{i=1}^{N} P(c_i)) = -\sum_{i=1}^{N} \log_2 P(c_i) \quad (2)$$

式 2 は文字列 S において i 番目に出現する文字 c_i の情報量を加算して文字列の情報量としている．

しかし，実際の文字列は単語のような特定の部分文字列が繰り返し出現することで構成されていると考えられる．従って，1 文字あたりの情報量を加算した場合，文字列が持つ繰り返しを考慮することが困難である．そこで，文字列を特定のパターンが繰り返し出現する系列であると考え，文字列の全部分文字列を用いて情報量を計算する．長さ N の文字列 S において，文字列をベースとした情報量は式 3 のように表すことができる．

$$I(S; D_c) = \min_{\pi_s \in \pi(S)} \left(-\sum_{t \in \pi_s} \log_2 P(t; D_c) \right) \quad (3)$$

式 3 において，$\pi(S)$ は分割された文字列の集合であり，その分割方法は 2^{N-1} 通り存在する．π_s は $\pi(S)$ の要素を示し，t は各分割方法で出現する部分文字列である．また，分類先のデータ群 C の文字列を連結したものを D_c とする．

情報量を計算する場合，全部分文字列の出現確率を計算する必要がある．本手法では出現確率を計算するために，既知データをクラスごとに分類，結合したデータ群を用いる．判定の様子を図 1 に示す．

図 1 において．未知データ x の文字列 “xyz” の分割方法は 2^{N-1} 通り存在するため，{“xyz”}{“xy”, “z”}{“x”, “yz”}{“x”, “y”, “z”} の 4 通りとなる．これら 4 つの分割方法が式 3 における $\pi(S)$ であり，“abc” のような分割方法が π_s に相当する．未知データに出現する部分文字列は “xyz”，“xy”，“yz”，“x”，“y”，“z” であり，これらが式 3 における t に相当する．これらの部分文字列の出現頻度をクラスごとにまとめたデータ群から計算する．図 1 の例であれば，クラス A のデータ群における部分文字列 “xy” の出現頻度は 0 である．一方クラス B のデータ群における出現頻度は 3 であり，クラス C のデータ群における出現頻度は 2 である．対象データに出現する全部分文字列の頻度をそれぞれ計算し，以下の式を用いて文字列の出現確率を計算する．

$$\hat{P}(t; D_c) = \frac{freq(t; D_c)}{N_{D_c}} \quad (4)$$

式 4 において，t は対象データの部分文字列であり，N_{D_c} は頻度を計算するデータ群の文字列の長さである．各部分文字列の出現頻度を計算し，全ての組み合わせから情報量が最小となるものを選出することで，未知データの最適な分割を導くことができる．情報量が最小となるものは，出現確率が最大となるものである．また情報量はそれぞれの既知データ群を用いて計算されるため，クラス A のデータ群を元に計算された情報量とクラス B のデータ群を用いて計算された情報量，クラス C のデータ群を用いて計算された情報量の 3 種類が算出される．計算された 3 つの情報量のなかで最も小さいデータ群のラベルが未知データのラベルとなる．

2.3　Symbolic aggregate approxomation (SAX)

CDM やそれを元に考案された情報量計算手法を用いて判定を行う場合，時系列データを文字列に変換する必要がある．本研究では時系列データの文字列化に Symbolic aggregate approxomation (SAX) [8] を用いる．図 2 の時系列データを SAX を用いて変換した場合 “eebbdc” となる．

2.4　CDM と情報量計算法の関係

CDM は圧縮ファイルサイズを利用する方法であるが，文字

列の情報量は圧縮サイズの下限と考えることが自然である．そして，CDM を使うときの圧縮プログラムは対象のデータに対して，圧縮の効果があるものでないと利用できないのは当然である．従って，圧縮率が高いものを選ぶのが自然な選択となっている．ここで，CDM は圧縮結果を使うのではなく，ファイルサイズだけを使うことに着目すれば，圧縮サイズをもとめる方法として情報量を考えることは自然である．情報量計算では，すべての可能な分割のなかで最小の情報量を求めている．これは，圧縮プログラムで利用する頻出文字列について，つながっている文字列という制限のなかではあるが，可能な文字列をすべて試みて，一番圧縮されるときのファイルサイズを求めていることに相当する操作となっている．

CDM は，２つのデータの間の類似度である．このため，これをクラス分類問題に使うには，最近傍法などの方法を組み合わせる必要がある．最近傍法では，近くのデータとして選ぶデータ数というパラメータがあるが，これは状況によって変化させる必要があり，その決定は難しい．そこで，提案の情報量計算においては，分類のクラスごとにデータをまとめ，そこでの出現確率を推定することをする．複数のデータにまたがって確率によって情報量を計算することにより，個々のデータで情報量を計算するよりも，より安定な値が得られることが期待できる．また，頻出する文字列は，個別のファイルというよりは分類のクラスの中に頻出する文字列が対象となり，よりクラス分類に適した尺度になると考えられる．

最後に，圧縮プログラムは，復元のために入力とは関係ない情報が含まれる可能性があり，CDM の値について利用する圧縮プログラムに依存するばかりか，その値の解釈が難しい．情報量計算であれば，この方法は，分析対象の出現確率を最大化する分割をもとめ，その分割においての確率の大小の比較と等価であると解釈できる．ここには，復元のための情報は含まれず，かつ，求まった値の理論的な解釈もできる．

情報量計算によるクラス分類は，圧縮プログラムとしてありとあらゆる分割を考えて良い圧縮法を考えるという圧縮プログラムであり，かつその圧縮率については，単一のデータの分析ではなくて，ある分類クラスこごとに分析をするというものを使ったものに相当する．

2.5 提案する情報量計算方法と固定長の分析との関係

情報量計算は，可変長かつ任意長の文字列を対象に情報量をもとめるものであるが，固定長（例えば２のバイグラム）でのモデル化に比べて，複雑で計算コストが高いと推測される．まず，今回の方法は対象となる時系列のモチーフ（頻出パターン）は数十の文字列から構成されることが想定され，２文字では不足することは明らかである．そのうえ，モチーフの長さの上限を予見することが難しい．このようなことから，全ての分割を考えることに意味がある．

CDM においては，圧縮プログラムが分析するパターンの長さの上限が２文字であるということは稀である．例えば，zip の圧縮方法ではパターンから構成されるモチーフを認識することができるのため，結果的に長いモチーフを処理できる．また，ブロックソート法を用いた圧縮の場合には，どのように長い文字列からなるモチーフであっても圧縮で利用できるアルゴリズムとなっている．このような理由からも，文字列のすべての分割に対して処理することが CDM の自然な改良となっていると考えられる．

3 提 案 手 法

CDM を元に考案された文字列の情報量を計算する手法は音楽の分類判定に対して効果的であることが示唆されている．しかし，これまでの研究では情報量計算手法が音楽以外のデータに対して有効であるかは検証されていない．そこで本研究では，情報量計算手法を用いて音楽以外の時系列データの分類判定を行うことを提案する．

4 実験と考察

4.1 実験に使用したデータ

実験には UCR Time Series [1] に掲載されているデータを使用する．これらの時系列データはテストデータとトレーニングデータで構成されている．各データはクラスと数値データで構成されており，本研究ではこれを SAX を用いて文字列に変換し判定を行う．本実験では以下の表 1 に示す，50 種類のデータを用いる．表 1 にはデータの名前，テストデータの数，テストデータのクラス数の 3 つが示されている．

4.2 予 備 調 査

データの作成者は CDM 以外の方法も提案しているため，UCR Time Series に含まれているデータが全て CDM で動作するとは限らない．CDM と情報量による分類を比較するときに，まず CDM で動作するものを選んで比較するほうが，CDM の改良としての評価が明確になると考えた．従って，予備調査で CDM が有効に働くデータを特定する．

4.2.1 予備調査の方法

UCR Time Series に掲載されているデータに対して，全てのデータを SAX を用いることで文字列に変換し CDM を用いて分類を行う．CDM を用いた判定では，1 つのテストデータとトレーニングデータとの距離をそれぞれ計算する．ある 1 つのテストデータと全てのトレーニングデータの CDM を計算し，最も CDM が小さいものを選出する．選ばれたデータのクラスが対象となるテストデータのクラスとなる．判定されたテストデータのクラスがそのデータのクラスと一致していれば正解とし，一致していなければ不正解とする．これらの処理を全てのテストデータに対して行い，その正解数をカウントする．

4.2.2 予備調査の結果

情報量を用いた判定では，学習データをクラスごとにまとめてデータ群を作成する．1 つのテストデータの情報量をクラスごとのデータ群を用いて計算し，各クラスの情報量の中から最も小さいものをテストデータのクラスとする．

表 1　実験に使用したデータ

データ名	データ数	クラス数
50words	455	50
Beef	30	5
BeetleFly	20	2
BirdChicken	20	2
Car	60	4
CBF	900	3
ChlorineConcentration	3840	3
CinC_ECG_torso	1380	4
Computers	250	2
Cricket_X	390	12
Cricket_Y	390	12
Cricket_Z	390	12
DiatomSizeReduction	306	4
DistalPhalanxOutlineAgeGroup	400	3
ECG5000	4500	5
ECGFiveDays	861	2
FaceAll	1690	14
FaceFour	88	4
FacesUCR	2050	14
FISH	175	7
GunPoint	150	2
Haptics	308	5
Herring	64	2
InlineSkate	550	7
InsectWingbeatSound	1980	11
LargeKitchenAppliances	375	3
MALLAT	2345	8
Meat	60	3
MedicalImages	760	10
MiddlePhalanxOutlineAgeGroup	400	3
MoteStrain	1252	2
NonInvasiveFatalECG_Thorax1	1965	42
NonInvasiveFatalECG_Thorax2	1965	42
OliveOil	30	4
OSULeaf	242	6
Phoneme	1896	39
RefrigerationDevices	375	3
ScreenType	375	3
SonyAIBORobotSurface	601	2
Strawberry	613	2
SwedishLeaf	625	15
Symbols	995	6
synthetic_control	300	6
Trace	100	4
Two_Patterns	4000	4
TwoLeadECG	1139	2
WordsSynonyms	638	25
Worms	181	5
WormsTwoClass	181	2
yoga	3000	2

4.3　CDM が動作しているデータに対する実験

4.3.1　実 験 方 法

予備調査で行なった CDM を用いたクラス分類において，正解数がランダムで分類を行なった場合よりも多いデータに対して情報量計算を行い，その正解数を比較する実験を行う．データに対して情報量計算を行うために，トレーニングデータをクラスごとに連結しデータ群を作成する．あるテストデータに対して各データ群を用いて情報量を計算する．各データ群をもとに計算された情報量から最も小さいものを選びその情報量を計算するために用いたデータ群のクラスがテストデータのクラスとなる．判定されたクラスとテストデータのクラスが一致していれば正解とし，一致していなければ不正解とする．

4.3.2　実 験 結 果

CDM が動作した 30 種類のデータに対してクラスの分類を行なった結果を表 2 に示す．表 2 において “bzip2” の列はベースライン手法である CDM を用いて分類を行なった場合の判定結果である．そして “情報量計算” の列は情報量計算手法を用いて分類を行なった場合の判定結果である．それぞれの列に記載されている数値は分類を行なった場合の正解数を表している．また，各データの数は “データ数” の列に記載されている．例えば Beef のデータであれば，データの総数が 30 個であり，bzip2 を用いて判定を行なった場合の正解数が 30 個，情報量計算手法を用いて判定を行なった場合の正解数が 15 個となる．

bzip2 を用いた判定結果と情報量計算を用いた判定結果を比較し，情報量計算手法の有効性を検証する．表 2 のデータにおいて，データ名に下線が引かれているものは情報量計算手法が bzip2 の正解数を上回っているデータを示している．このことから，情報量計算手法が bzip2 の正解数を上回っているデータは全 30 種類中 19 種類であることがわかる．この結果は過半数であるが，符号検定による危険率は 0.10 であり，データの種類ごとに検定すると性能の向上が有意であるとは言えなかった．この検定は，個々のデータの正解・不正解を，正解数にまとめたうえで勝ち負けの 1 ビットに変換しており，情報が失われていると考えられる．

そこで，それぞれのデータについて比較をする．CDM と提案方法の両方で正解であったデータは 6491 個，CDM と提案方法の両方で不正解であったデータは 5840 個，CDM だけ正解であったものが 3193 個, 提案方法だけ正解であったものが，6841 個であった．これをマクネマー検定で性能の差を検定する．性能差がないことを帰無仮説とすると，危険率 3.17×10^{-297}，で帰無仮説は棄却され，性能の向上は統計的に有意であった．

提案方法のほうが正解数が多い 19 種類のものについて分析すると，そこに含まれるデータの数が多いものであった．この説明としては，提案方法における確率の最尤値による推定が正確な場合に，CDM より性能が高くなること考えるのは自然である．実際の応用においては，クラスごとに教師データを必要なだけ集めることはできると考えられる．また，頻度の少ない場合の確率の推定方法を最尤推定から取り替えることで性能改善できる可能性がある．

CDM のほうが正解数が多い 11 種類のものに分析する．これは CDM が全数正解となっているものが多い．提案方法は CDM の動作原理を分析して考案したものである．従って，CDM が全数が正解であるような種類のデータは提案方法でも全数正解

であることを期待していたが，そのようになっていない．この不正解のデータをさらに精査すれば，提案方法に不足しているものが明らかになると考えられる．

4.4 CDM が動作しなかったデータも含める実験

前節の結果より CDM で判定できるデータについて，間違えるケースがあることが明かになったため，CDM が動作しないものについての振る舞いの比較を行うことも参考になると考えた．また，実際の状況では CDM あるいは提案方法がうまく動くかどうか予見することは難しいことを考え，提供される全データを対象に実験を行うこととする．

4.4.1 実験方法

予備調査で行なった CDM を用いたクラス分類において，提供されるすべてのデータについて，その正解数を比較する実験を行う．情報量計算を用いた判定方法は 4.3 節と同様である．データに対して情報量計算を行うために，トレーニングデータをクラスごとに連結しデータ群を作成する．あるテストデータに対して各データ群を用いて情報量を計算する．各データ群をもとに計算された情報量から最も小さいものを選びその情報量を計算するために用いたデータ群のクラスがテストデータのクラスとなる．判定されたクラスとテストデータのクラスが一致していれば正解とし，一致していなければ不正解とする．

4.4.2 実験結果

本実験では CDM が効果的に動作しなかった 20 種類のデータを追加し，全部で 50 種類のデータに対して情報量計算を用いて判定を行なった．注目すべきことに追加した 20 種類のデータの全てのデータで正解数が向上し，合計 39 種類のデータで，正解数が CDM を上回ったことである．符号検定を行うと，性能の向上は危険率 4.51×10^{-5} で統計的有意である．

さらに個別のデータでみると，CDM と提案方法の両方で正解であったデータは 8729 個，CDM と提案方法の両方で不正解であったデータは 16735 個，CDM だけ正解であったものが 4678 個, 提案方法だけ正解であったものが，15518 個であった．これをマクネマー検定を用いて性能の差を検定する．性能差がないことを帰無仮説とすると，危険率 3.85×10^{-1335} で帰無仮説は棄却され，性能の向上は統計的に有意であった．

提案方法が CDM を手本に考案されたことを考えると，CDM で動作しないデータは提案方法でもうまく動作しない傾向があることが予測されたが，実際には提案方法が CDM よりも多くのデータで有効に動作することが期待できる

4.5 考　　察

CDM がうまく動作しないデータについても，我々の方法ではうまく動作したことは注目すべき結果である．一般に CDM が適応できるかどうかわからないという条件での応用を考えなければならないので，CDM が十分に動作しないものも含めて評価をすることも妥当であると考えらえる．

CDM が動作している場合に，提案方法との差が小さいことは，両方とも類似の観点から類似度を計算していることを考えると自然なことである．その場合であっても，データ数が多いときには性能が向上するのであれば，提案方法を使うときに，そこに注意して，データ数を用意するようにすれば，実用上は大きな問題にならないように考える．

5 おわりに

本稿では，CDM による時系列データの分類法と情報量を推定することによる分類報の比較を行った．テストデータは CDM の提唱者が公開している時系列データを使用した．まず，CDM が動作していると考えられる 30 種類のうち，過半数の 19 種類で性能の改善ができた．この結果では統計的な有意差は取れなかったが，データ個別にみると改善したもの 6861 個，改悪したもの 3193 個であり，これは統計的に有意な差である．

次に，提供されている全部のデータ 50 種類について実験すると，改善したものが 39 種類となり，これは有意な結果である．全体のデータ個別にみると，改善したもの 15518 個，改悪したもの 4678 個であり，これも統計的に有意な結果である．

今後の課題としては，CDM よりも提案方法が劣っているケースの解析である．提案方法は，圧縮のサイズのかわりに情報量を使っているため，少なくとも CDM と同じには動作すると考えているが，まだ考慮がたりないところが残されている可能性がある．現在，考えられる原因としては，頻度の少ないところでの情報量（確率）の推定方法に問題がありそうであるが，この分析と改善は今後の課題である．

文　　献

[1] Chen, Yanping and Keogh, Eamonn and Hu, Bing and Begum, Nurjahan and Bagnall, Anthony and Mueen, Abdullah and Batista, Gustavo. "The UCR Time Series Classification Archive."2015.

[2] Jeong, Young-Seon, Myong K. Jeong, and Olufemi A. Omitaomu. "Weighted dynamic time warping for time series classification." Pattern Recognition 44, no. 9 (2011): 2231-2240.

[3] Keogh, Eamonn, Stefano Lonardi, and Chotirat Ann Ratanamahatana. "Towards parameter-free data mining." In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 206-215. ACM, 2004.

[4] Cilibrasi, Rudi, and Paul MB Vitányi. "Clustering by compression." IEEE Transactions on Information theory 51, no. 4 (2005): 1523-1545.

[5] Anan, Yoko, Kohei Hatano, Hideo Bannai, Masayuki Takeda, and Ken Satoh. "Polyphonic Music Classification on Symbolic Data Using Dissimilarity Functions." In ISMIR, pp. 229-234. 2012.

[6] Takamoto, Ayaka, Mitsuo Yoshida, Kyoji Umemura, and Yuko Ichikawa. "Computing information quantity as similarity measure for music classification task." In Advanced Informatics, Concepts, Theory, and Applications (ICAICTA), 2017 International Conference on, pp. 1-6. IEEE, 2017.

[7] Takamoto, Ayaka, Mitsuo Yoshida, Kyoji Umemura, and Yuko Ichikawa. "Feature Selection for Composer Classification Method using Quantity of Information." In 2018 10th International Conference on Knowledge and Smart Technology (KST), pp. 30-33. IEEE, 2018.

[8] Lin, Jessica, Eamonn Keogh, Stefano Lonardi, and Bill Chiu. "A symbolic representation of time series, with im-

表 2　CDM と情報量計算を用いたクラス分類の結果 1

データ名	データ数	チャンスレベル	bzip2	情報量計算	向上	低下
50words	455	9.1	24	110	107	21
Beef	30	6	30	15	0	15
BeetleFly	20	10	20	13	0	7
BirdChicken	20	10	20	13	0	13
CBF	900	300	506	525	237	218
ChlorineConcentration	3840	1280	1490	2182	1360	668
Computers	250	125	250	151	0	99
Cricket_Y	390	32.5	54	94	79	39
DiatomSizeReduction	306	76.5	131	241	143	33
DistalPhalanxOutlineAgeGroup	400	133.3	139	338	221	22
ECG5000	4500	900	2670	4075	1611	206
FaceAll	1690	120.7	264	498	427	193
FaceFour	88	22	22	23	18	17
FISH	175	25	27	110	98	15
GunPoint	150	75	96	130	47	13
Herring	64	32	36	34	16	18
LargeKitchenAppliances	375	125	375	261	0	114
Meat	60	20	60	55	0	5
MedicalImages	760	76	295	369	219	145
NonInvasiveFatalECG_Thorax1	1965	46.8	66	778	759	47
OliveOil	30	7.5	30	24	0	6
OSULeaf	242	40.3	41	138	110	13
RefrigerationDevices	375	125	375	177	0	198
ScreenType	375	125	375	178	0	197
SwedishLeaf	625	41.7	45	303	286	28
synthetic_control	300	50	300	160	0	140
WordsSynonyms	638	25.5	63	165	151	49
Worms	181	36.2	63	91	61	33
WormsTwoClass	181	90.5	82	112	59	29
yoga	3000	1500	1735	1989	852	598
合計	22385	5466.6	9684	13352	6861	3194

表 3　CDM と情報量計算を用いたクラス分類の結果 2

データ名	データ数	チャンスレベル	bzip2	情報量計算	向上	低下
Car	60	15	12	41	31	2
CinC_ECG_torso	1380	345	237	472	380	145
Cricket_X	390	32.5	20	121	114	13
Cricket_Z	390	32.5	30	100	90	20
ECGFiveDays	861	430.5	234	553	409	90
FacesUCR	2050	146.4	129	579	544	94
Haptics	308	61.6	37	102	88	23
InlineSkate	550	78.6	41	120	110	31
InsectWingbeatSound	1980	180	124	351	332	95
MALLAT	2345	293.3	237	1626	1460	71
MiddlePhalanxOutlineAgeGroup	400	133.3	66	305	258	19
MoteStrain	1252	626	480	1005	619	94
NonInvasiveFatalECG_Thorax2	1965	46.8	29	1069	1051	11
Phoneme	1896	48.6	11	217	216	10
SonyAIBORobotSurface	601	300.5	178	302	212	88
Strawberry	613	321.5	288	556	288	20
Symbols	995	165.8	144	392	338	90
Trace	100	25	14	94	81	1
Two_Patterns	4000	1000	910	2075	1585	420
TwoLeadECG	1139	569.5	502	815	461	148
合計	23275	4852.4	3723	10895	8667	1485
全体の合計	45660	10319	13407	24247	15528	4679

plications for streaming algorithms." In Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery, pp. 2-11. ACM, 2003.