クラスタリングによる同一ラベルデータ要素のグループ化のためのデータ属性重要度算出手法

中村　朋健† 　上土井陽子†† 　若林　真一†† 　吉田　典可†††

† 広島市立大学大学院 情報科学研究科　〒 731–3194 広島市安佐南区大塚東三丁目 4–1
†† 広島市立大学 情報科学部　〒 731–3194 広島市安佐南区大塚東三丁目 4–1
††† 前所属 広島市立大学 情報科学部　〒 731–3194 広島市安佐南区大塚東三丁目 4–1
E-mail: †tomotake@lcl.ce.hiroshima-cu.ac.jp, ††{yoko,wakaba}@ce.hiroshima-cu.ac.jp

あらまし　我々の目標は高次元データを処理可能なクラスタリング手法を用いて，教師ありで同一クラスラベルを持つデータ要素をグループ化することである．クラスタリング手法を用い同一クラスラベルを持つデータ要素をグループ化するために，我々は各属性がラベルに与える影響の大きさ（属性の重要度）を算出する．属性の重要度算出には我々の開発している特徴抽出手法 [6] を使用する．クラスラベルに関してセンシティブな属性を見つけることで，高精度に同一クラスラベルを持つデータ要素の集合を形成できる．クラスラベルに基づいた各属性の重要度を算出し，各属性の値の変換やダミー要素を追加することで，クラスタリング手法による同一クラスタラベルを持つデータ要素をグループ化することを試みる．ベンチマークデータを用いたシミュレーション実験では算出した重要度を基に値を変換したデータセットをクラスタリングし，重要度算出の有効性を示す．

キーワード　データマイニング，クラスタリング

Detection of Important Attributes for Supervised Clustering

Tomotake NAKAMURA†, Yoko KAMIDOI††, Shin'ichi WAKABAYASHI††, and Noriyoshi YOSHIDA†††

† Graduate School of Information Sciences, Hiroshima City University
3–4–1, Ozuka-higashi, Asaminami-ku, Hiroshima, 731–3194 Japan
†† Faculty of Information Sciences, Hiroshima City University
3–4–1, Ozuka-higashi, Asaminami-ku, Hiroshima, 731–3194 Japan
††† Formerly, Faculty of Information Sciences, Hiroshima City University
3–4–1, Ozuka-higashi, Asaminami-ku, Hiroshima, 731–3194 Japan
E-mail: †tomotake@lcl.ce.hiroshima-cu.ac.jp, ††{yoko,wakaba}@ce.hiroshima-cu.ac.jp

Abstract　Our aim is to group data objects which have a unique class label with clustering in which high dimensional datasets can be processed. In order to do with clustering, we compute the degree of the influence of each attribute for class labels. To find important attributes which have the large degree of the influence for class labels, we use the feature extraction method which we have developed. We can construct a set of data objects which have a single class label with high probability by finding sensitive attributes for the class label. Next, we group data objects which have a unique class label by clustering methods. From experimental simulation, we show an effectiveness of important attribute detection, by performing clustering of transformed benchmark data sets as two class classification problems.

Key words　Data Mining, Clustering

1. はじめに

情報社会の進展と記憶装置の低価格化により，個人情報や個体情報などを蓄積した高次元，かつ，大規模なデータセットが急速に増加している．個人情報などはプライバシー保護の課題を考慮した上で，企業や医療分野での意思決定において効果的に利用できる．個人情報などを蓄積した高次元，かつ，大規模なデータセットを利用するための手法として外的基準に基づくクラシフィケーション（教師あり分類）や外的基準を使用しないクラスタリング（教師なし分類）などがある．クラシフィケーション問題とは，入力として，データセット DS とその部分集合 S が与えられたときに，DS のすべての他のデータ要素から S のデータ要素を区別するための条件もしくはルールの集合を出力する問題である．

クラシフィケーションモデルは顧客がローンを支払う能力があるかどうかを判断する場合や [3]，潜伏したウイルスにより発病する可能性がある患者かどうかを判断する場合などに役立つ．クラシフィケーションにより急速に広まる伝染病のウイルスに侵されている人々や家畜やペットなどの個体を識別することで迅速に医療処置でき，被害を少なく抑えられるだろう．しかし，データ要素を区別するための条件やルールはユーザによって異なることが多いが，様々なユーザに対応する条件やルールの決定は難しい．

クラシフィケーション手法では，高次元なデータセットに対して概要となる属性の部分空間を作成する手法 [1] がある．サポートベクターマシン（SVM）や BRSVM などの従来のクラシフィケーションでは大規模なデータセットからリアルタイムにクラシフィケーションルールを作成することは難しい [5]．CVM [10] は大規模なデータセットからでも高速にクラシフィケーションルールを作成することが可能な場合もあるが，CVM はデータの分布に大きく依存するため実行してみなければ処理時間が分からない問題点を持つ．

一般のクラスタリング手法ではデータ空間の自然なクラスタを見つけることが目標であり，特定のデータ要素集合を他と区別することを目標としていないため，一般のクラスタリング手法をクラスラベルの分類に使用することは難しい．しかし，クラスタリング手法は入力パラメータの設定によりクラスタのサイズや形状を変化させることが可能であり，様々なユーザの要求に対話的に応えることが容易である．クラシフィケーション問題のようにデータ要素を区別する条件やルールを明確に定めるのではなく，データ要素を区別するための重要な属性を検出し様々なユーザの要求に対話的に応えられるクラシフィケーションシステムの開発を目標としている．本稿ではその目標の準備として大規模データを高速に分類することに優れ，かつ，パラメータの設定により様々な結果を出力可能なクラスタリング手法，例えば FlexDice [7] や OptiGird [4] によって，テストデータセットから効率よく同一ラベルを持つデータ要素を集める例を示す．以下で同一ラベルを持つデータ要素を集めることを分類クラスタリングと呼ぶこととする．クラスタリング手法を用いて分類クラスタリングするために，トレーニングデータセットから高速にクラスラベルでデータセットを分類するための情報の作成方法を提案する．

クラスタリング手法を教師ありで同一ラベルを持つデータ要素をグループ化することにおいて使用する試みははじまったばかりであり，高次元データへの適用には至っていない [2]．一方で，高次元データクラスタリング手法 O-Cluster をクラシフィケーション問題のデータセットに適用し，同一ラベルを持つデータ要素のグループ化の精度を評価 [8] することが試みられている．しかし，文献 [9] では O-Cluster は教師なし分類手法のままでデータに適用され，同一ラベルを持つデータ要素のグループ化に関して評価されている．O-Cluster と同等の結果を出力可能な OptiGrid [4] でも教師なし分類手法として O-Cluster と同等の結果を出力可能であると考えられる．

分類クラスタリングはなるべく少ないクラスタ数で同一ラベルのデータ要素を同じクラスタに集めることを目標としている．一般に教師なし分類法であるクラスタリングを教師ありクラシフィケーション問題に適用するために，我々はデータセットの重要な属性の検出手法とその手法に基づく属性の重要度算出手法を提案する．重要な属性の検出手法と重要度算出により，クラスタリング手法を教師あり分類手法として使用し，クラスタリング手法による分類クラスタリングの効果を高めた．

2. 重要度算出とその応用

我々は属性ごとに重みを変えてクラスタリングすることで分類クラスタリングの精度が向上するのではないかと考えた．形成分類クラスタリングの精度を向上させるために，クラスラベルに影響力の高い属性（重要な属性）を見つけ出し，属性の重要度を算出する．

本章では第 2.1 節において，クラスタリング手法を用いて分類クラスタリングするために使用する重要な属性の検出手法と重要な属性に基づいた属性の重要度算出手法を提案する．重要な属性を検出するために，我々が開発を進めている特徴抽出手法 [6] を使用する．属性の重要度は検出した重要な属性に基づいて算出される．第 2.2 節において重要度算出手法の応用例を示す．

本稿の 4. 章ではクラスタリング手法を用いて重要な属

性検出手法と重要度算出手法の有効性を示すが，このとき高次元データセットを高速，かつ，高精度にクラスタリング可能な FlexDice [7] を使用する．FlexDice は連続する密な領域のデータ要素群をクラスタとして集め，疎な領域のデータ要素群を 1 つのノイズクラスタとして集めるクラスタリング手法である．FlexDice は，まず，階層的にトップダウン方式でセルを構築・削除しながらセル間の隣接関係を作成する．ここで，セルとは入力データの全属性を含む入力データ空間の部分空間である．次に，作成した隣接関係に基づいて密なセルを結合させセルクラスタを形成する．

2.1 重要度算出手法

2 クラス分類問題であるトレーニングデータセットを利用し，テストデータを分類クラスタリングするために必要となる重要な属性を検出する．トレーニングデータセットはクラスラベルが記された属性を含むデータセットであり，テストデータはクラスラベルが記された属性を含まないデータセットのことである．重要な属性の検出は，まず，重要な属性を見つける種となる属性（FIA: First Important Attributes）の集合を検出し，次に FIA に含まれる属性を使用して属性の重要度を算出する．属性の重要度は重要な属性として数えられた回数で算出される．

トレーニングデータセットを利用した属性の重要度算出手法を図 1 に示し，以下に図 1 を補足説明する．ただし，本稿で対象とするデータセットはラベル “0” とラベル “1” の 2 つのクラスに分類されるデータセットである．図 1 で用いた用語のうち我々が定義した用語の説明は紙面の都合上，補足説明内のみに記す．

[***Step 1***]

2 値のクラスラベルがついたトレーニングデータセットから同一クラスラベルのデータ要素を集める．クラスラベルごとに集めたデータ要素の集合，つまりラベル “0” を持ったデータ要素の集合であるデータセット $DS0$，ラベル “1” を持ったデータ要素の集合である $DS1$ を作成する．以下では $DS0$ をラベル “0” データセット，$DS1$ をラベル “1” データセットと呼ぶ．

[***Step 2***]

各属性に関してラベル “0” データセット $DS0$ とラベル “1” データセット $DS1$ の値ベクトルを作成する．ここで，属性 a に関するデータセット DS の値ベクトル $\boldsymbol{V}(DS, a)$ は以下の式で表される．

$$\boldsymbol{V}(DS, a) = (Va_1, \ldots, Va_j, \ldots, Va_{VID_a})$$

VID_a は属性 a の値識別子数である．値識別子とは，ある属性の存在し得る値一つ一つに付けた名前である．Va_j とは，属性 a に関して，データセット DS に含まれる全データ要素数に対する値識別子 j に対応する要素数の割合である．ただし，属性 a の存在し得る値が多すぎるときは，属性 a の値を大まかに離散化し値識別子を付ける．例えば，属性 a に関して作成したラベル “0” データセット $DS0$ とラベル “1” データセット $DS1$ の値ベクトルは図 2 の $\boldsymbol{V}(DS0, a)$ と $\boldsymbol{V}(DS1, a)$ ように描画できる．図 2 は，属性 a においてラベル “0” データセット $DS0$ に含まれる全データ要素数に対する値識別子 0，1，2，3，4，5 の割合がそれぞれ 10%，30%，20%，10%，0%，40%であった例と，ラベル “1” データセット $DS1$ に含まれる全データ要素数に対する値識別子 0，1，2，3，4，5 の割合がそれぞれ 20%，20%，5%，0%，30%，25%であった例である．

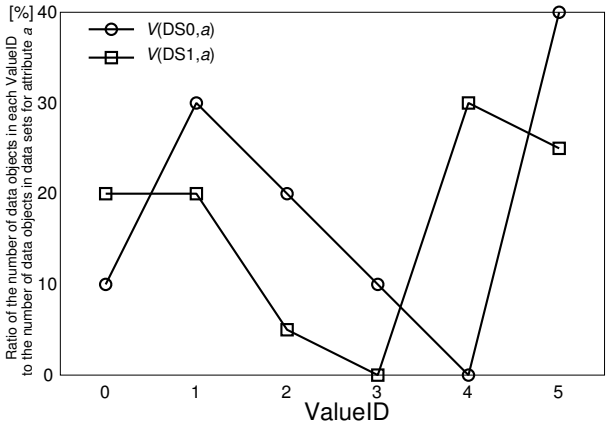

図 2　属性 a に関するラベル “0” データセット $DS0$ の値ベクトル $\boldsymbol{V}(DS0, a)$ とラベル “1” データセット $DS1$ の値ベクトル $\boldsymbol{V}(DS1, a)$ の例

図 2 の値識別子におけるラベル “0” データセット $DS0$ とラベル “1” データセット $DS1$ の割合の差で，閾値より大きいものがあるかどうかすべての属性で調べる．ここで閾値が 25 であるとすると，値識別子 4 におけるラベル “0” データセット $DS0$ とラベル “1” データセット $DS1$ の差が 30 であるため，属性 a は最初に検出された重要な属性の集合（First Important Attribute, FIA）の要素となる．

[***Step 3***]

最初に検出された重要な属性の集合 FIA に属する属性だけを使用してクラスタリングする．最初に検出された重要な属性の集合 FIA に属する属性が 3 つのみであれば 3 属性だけからなる部分データ空間上のデータセットとしてラベル “0” データセット $DS0$ とラベル “1” データセット $DS1$ をそれぞれクラスタリングする．ラベル “0” データセット $DS0$ から形成されたクラスタを $C1_{DS0}$，$C2_{DS0}$，... とし，ラベル “1” データセット $DS1$ から形成されたクラスタを $C1_{DS1}$，$C2_{DS1}$，... とする．

Step 1: トレーニングデータセットをクラスラベルによって 2 つのデータセット $DS0$ と $DS1$ に分ける．

Step 2: 各属性に関して $DS0$ と $DS1$ の値ベクトルを作成する．*Step 2* で求めた $DS0$ と $DS1$ の各値識別子に対する割合の差において，与えた閾値以上の差がある値が 1 つ以上ある属性を抽出し，その属性の集合を FIA とする．

Step 3: FIA に属する属性の値のみからなるデータセットとして，$DS0$ と $DS1$ をそれぞれクラスタリングする．

Step 4: 各属性に関して作成されたクラスタの値ベクトルを作成する．$DS1$ から形成された 1 つのクラスタの値ベクトルを $DS0$ から形成された全クラスタの値ベクトルに加え，値ベクトルデータセットを作成する．各属性に関して，特徴抽出手法 [6] を用いて，特徴となる値ベクトルを抽出する．

Step 5: *Step 4* において抽出した値ベクトルが $DS0$ に加えた値ベクトルである属性ごとの回数を求め，カウンタに格納する．$DS0$ に加えていない $DS1$ から作成される値ベクトルがあれば加えて同様の操作を繰り返す．なければ FIA を重要な属性とし，求めたカウンタの値の大きい属性順に重要な属性とする．

図 1　分類クラスタリングのための 2 つに分類されているトレーニングデータセットを用いた属性の重要度検出手法

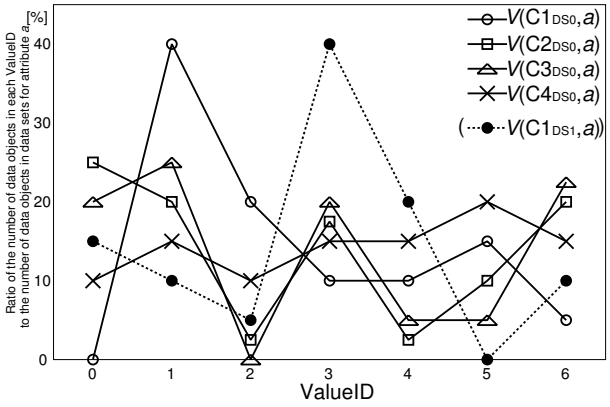

図 3　属性 a に関するラベル “0” データセット $DS0$ から形成した各クラスタの値ベクトル $\boldsymbol{V}(C1_{DS0}, a)$，$\boldsymbol{V}(C2_{DS0}, a)$，$\boldsymbol{V}(C3_{DS0}, a)$，そして $\boldsymbol{V}(C4_{DS0}, a)$ の例

[***Step 4***]

各属性に関して作成されたクラスタの値ベクトルを作成する．ラベル “0” データセット $DS0$ を *Step 3* でクラスタリングしたとき 4 つのクラスタ $C1_{DS0}$，$C2_{DS0}$，$C3_{DS0}$，そして $C4_{DS0}$ が作成されたと仮定すると，属性 D_i に関して値ベクトル $\boldsymbol{V}(C1_{DS0}, a)$，$\boldsymbol{V}(C2_{DS0}, a)$，$\boldsymbol{V}(C3_{DS0}, a)$ そして $\boldsymbol{V}(C4_{DS0}, a)$ を作成する．図 3 はラベル “0” データセット $DS0$ を入力としたときの属性 a に関する値ベクトルである（ただし，値ベクトル $\boldsymbol{V}(C1_{DS1}, a)$ を除く）．*Step 2* と同様に，存在し得る値が多すぎる場合は値域の値を大まかに離散化して値に対応する識別子を付ける．

各属性に関して，作成した 1 つの値ベクトルを 1 つのデータ要素とするデータセットを作成する．ここで作成されたデータセットを値ベクトルデータセットと呼ぶ．属性 a に関して作成した値ベクトルデータセットの属性数は属性 a の値識別子数 VID_a である．1 つの値ベクトルデータセットに含まれるデータ要素は，ラベル “0” データセット $DS0$ から形成された全クラスタに対応する属性 a に関する値ベクトルとラベル “1” データセット $DS1$ から形成された 1 つのクラスタに対応する属性 a に関する値ベクトルである．図 3 は，属性 a に関して，ラベル “0” データセット $DS0$ から作成されたすべての値ベクトル $\boldsymbol{V}(C1_{DS0}, a)$，$\boldsymbol{V}(C2_{DS0}, a)$，$\boldsymbol{V}(C3_{DS0}, a)$，そして $\boldsymbol{V}(C4_{DS0}, a)$ にラベル “1” データセット $DS1$ から形成された 1 つの値ベクトル $\boldsymbol{V}(C1_{DS1}, a)$ を加えた例である．本稿ではラベル “0” データセット $DS0$ から作成された値ベクトルの集合にラベル “1” データセット $DS1$ から作成された値ベクトルを加える例を示しているが，場合によっては逆のラベル “1” データセット $DS1$ から作成された値ベクトルの集合にラベル “0” データセット $DS0$ から作成された値ベクトルを加えることも考えられる．次に特徴抽出手法で特異な分布を持つ値ベクトルを抽出する．

[***Step 5***]

Step 4 で加えた値ベクトルと *Step 4* で抽出された特異な分布を持つ値ベクトルが一致するかどうかを調べ，一致した属性を重要な属性として選ぶ．各属性に関して加えた値ベクトルが特徴抽出手法で抽出した値ベクトルである回数を調べ，回数を記憶するためのカウンタに格納する．カウンタは入力データの属性数だけ存在する．図 3 の例では，値ベクトル $\boldsymbol{V}(C1_{DS1}, a)$ と類似した分布の値ベクトルが存在しないため，値ベクトル $\boldsymbol{V}(C1_{DS1}, a)$ が特異な分布をもつ値ベクトルとして抽出され，着目している属性 a のカウンタをインクリメントする．

カウンタの値に基づいて各属性の重要度を算出する．カウンタの値が大きい属性順に属性の重要度が高いものとする．また，最初に検出された重要な属性の集合 FIA に属する属性を最も重要な属性とするかどうかはユーザ

表 1　変換前のデータセットと重要度（$IDeg$）の例

	A_1	A_2	A_3	A_4
v_1	80	40	2	0
v_2	10	40	0	1
v_3	50	60	7	1
v_4	90	10	5	0
v_5	70	90	4	0
$IDeg$	4	1	3	2

表 2　変換後のデータセットの例

	A_1	A_2	A_3	A_4
v_1	18	30	11	0
v_2	0	30	0	60
v_3	10	50	40	60
v_4	20	0	29	0
v_5	15	80	6	0

が定めても良いものとする．

2.2 応　　用

分類クラスタリングするために必要となる属性の重要度を算出し，テストデータセットに含まれるデータ要素の値を算出した属性の重要度に基づいてデータセットを変換しクラスタリングする．使用するクラスタリング手法によって様々な値の変換が考えられる．

データ要素間の距離を用いてクラスタリングする手法ならば，重要度の高い属性は値域を広げて各データ要素のその属性の値を正規化する．

ここでの正規化とは間隔の等しい離散値を，各値の間隔を等しくするという制約の下で縮小，または，拡大させることである．重要度の低い属性は値域を狭めて各データ要素のその属性の値を正規化する．例として表 1 に示すデータセット $\{v_1, v_2, v_3, v_4, v_5\}$ を重要度 $IDeg$ に基づいて変換する．表 1 の A_1，A_2，A_3，A_4 は属性を表し，$IDeg$ は値が小さいほど重要度が高いことを示す．各属性の値を重要度の高い順に値域が 80，60，40，20 となるように正規化すると，表 2 となる．変換後は各データ要素間の距離をユークリッド距離などの距離関数で求めてクラスタリングする．このように値を変換してクラスタリングすることで，クラスラベルに基づいたデータ要素集合の形成が可能となる．

FlexDice のような格子構造を用いるクラスタリング手法では，属性ごとに値域幅を変えただけでは効果がなく，変換後のデータセットに各属性の値域幅を等しくするダミーデータ要素を加えることで分類クラスタリングが可能になる．ここで，各属性の値域幅とは各属性における最大値と最小値の差である．

表 3　収入データセットの属性名と値域

No.	属性名	最小値	最大値	値域幅
1	age	17	90	74
2	workclass	0	8	9
3	fnlwgt	12,285	1,484,705	1,472,421
4	education	0	15	16
5	education-num	1	16	16
6	marital-status	0	6	7
7	occupation	0	14	15
8	relationship	0	5	6
9	race	0	4	5
10	sex	0	1	2
11	capital-gain	0	99,999	100,000
12	capital-loss	0	4,356	4,357
13	hours-per-week	1	99	99
14	native-country	0	41	42
15	class label	0	1	2

3. 実験の準備

本章では 4. 章のシミュレーション実験の準備として，第 3.1 節において入力データセットについて説明し，第 3.2 において分類クラスタリングしたときの結果を評価する評価関数を定義する．

3.1 入力データセット

4. 章のシミュレーション実験において使用するベンチマークデータは KDD アーカイブ [11] の “Census Income Database”（収入データ）と “The Insurance Company Benchmark (COIL 2000)”（保険データ）である．収入データと保険データのデータ要素は 2 つに分類されている．収入データのデータ要素は収入が$50,000 以下であるか$50,000 より多いかに分類され，保険データのデータ要素は保険に未加入の人か保険に加入している人かに分類される．

収入データは表 3 に示す 15 属性を含む 32,560 個のデータ要素から構成される．保険データは 86 属性を含む 5,821 個のデータ要素から構成される．収入データの属性 2，4，6，7，8，9，10，14，15 はテキストデータである．数値のみで処理可能なクラスタリング手法を使用するため，テキストデータは任意に 0 からはじまる整数値に変換した．保険データはすべて数値データであった．

3.2 評 価 関 数

データセットに対して分類クラスタリングしたときの評価尺度としてクラシフィケーションエラー E_c がよく使用される．クラシフィケーションエラー E_c は入力データ要素全体に対して誤って分類された要素数の割合である．クラスタリングアルゴリズムが要素数 $s_1, \ldots, s_i, \ldots, s_k$ の k 個のクラスタを出力し，クラスタ Ci の最多共通ラ

ベルを主クラスラベルとする．また，クラスタ Ci の主クラスラベルを持つデータ要素数が m_i であるとき，クラシフィケーションエラー E_C は以下の式によって定義される．ここで N は入力データ要素数である．

$$E_C = \frac{\sum_{i=1}^{k}(s_i - m_i)}{\sum_{i=1}^{k} s_i} = \frac{\sum_{i=1}^{k}(s_i - m_i)}{N}$$

しかし，クラシフィケーションエラー E_C はクラスラベル間のデータ要素数の差が大きすぎるデータセットであると妥当な評価ができない．例えば，データセットに対するクラスラベルが “0” のデータ要素数の割合が 80%であり，クラスラベルが “1” のデータ要素数の割合が 20%である場合を考える．上記の評価方法では形成されたクラスタの 1 つに含まれるクラスラベルが “0” のデータ要素数が 1,000 個，クラスラベルが “1” のデータ要素数が 900 個のクラスタがあれば，“0” をクラスタの主クラスラベルとする．このようにデータ要素数の大きさだけで主クラスラベルを定めることで，データセットに含まれるデータ要素が元々少ないクラスラベルのデータ要素群を集めたときに，評価関数で評価されない問題がある．また，クラシフィケーションエラー E_C は最良値は 0 であるが，最悪値がデータセットのバランスによって異なる．上記の例であると最悪値は 0.2 となり，データセットのバランスによって最悪値が異なることになり，評価尺度としては望ましくない．

我々はバランスのとれていないデータセットであってもバランスのとれたデータセットとして評価可能なクラシフィケーションエラー E'_C を定義する．クラシフィケーションエラー E'_C は最良値が 0 であり，最悪値が 0.5 である．主クラスラベルは最多共通ラベルではなく，入力データのクラスラベルの各値を持つデータ要素が同数であったと仮定したときの比率が高いクラスラベルとする．上記の例のクラスタではクラスラベルが “0” のデータ要素と “1” のデータ要素を同数であると仮定するためにそれぞれのデータ要素数に $\frac{1}{0.8}$ 倍と $\frac{1}{0.2}$ 倍する．このときクラスラベルが “0” のデータ要素数に相当する値は $1000 \times \frac{1}{0.8}$，クラスラベルが “1” のデータ要素数に相当する値は $900 \times \frac{1}{0.2}$ となり，値の大きいクラスラベル “1” を主クラスラベルとする．

本稿ではクラスラベルが “0”，または，“1” の 2 値であるデータセットのみを扱うため，クラシフィケーションエラー E'_C をクラスラベルが “0” と “1” のときについて定義する．入力データセットにおいて，クラスラベルが “0” の全データ要素数を主クラスラベル全要素数 $MCL0$，クラスラベルが “1” の全データ要素数を主クラスラベル全要素数 $MCL1$ とする．クラスタリングアルゴリズムが要素数 $s_1, \ldots, s_i, \ldots, s_k$ の k 個のクラスタ $C1, \ldots, Ci, \ldots, Ck$ を出力したとする．クラスタ Ci のクラスラベルが “0” のデータ要素数を $N0_{Ci}$，クラスラベルが “1” のデータ要素数を $N1_{Ci}$ とする．クラスタ Ci のデータ要素数に対する主クラスラベルを持つデータ要素数が MCL_{Ci} であるとき，クラシフィケーションエラー E'_C は以下の式によって定義される．

$$E'_C = \sum_{i=0}^{k} \frac{MCL_{Ci}}{N0_{Ci} \times \frac{N}{MCL0} + N1_{Ci} \times \frac{N}{MCL1}}$$

本稿では分類クラスタリングしたときの精度をクラシフィケーションエラー E'_C とクラスタ数で評価する．クラスタはなるべく大きく形成したいので，クラスタ数が少ないほど良い結果とする．

4. シミュレーション実験

本章では第 3.1 節で説明した 2 つのデータセットと 2 つのクラスタリングアルゴリズムを使用し，重要な属性の検出手法と属性の重要度算出手法を評価する．シミュレーション実験の説明には 2. 章の図 1 を使用する．

入力データセットを教師なしのクラスタリングのみで分類クラスタリングしたときの結果と第 2.1 節で説明した属性の重要度決定手法を用いてクラスタリング手法で分類クラスタリングしたときの結果を比較する．結果の比較には，ほぼ等しい数に分類されたときのクラシフィケーションエラー E'_C を用いて評価する．本章の属性の重要度算出手法と特徴抽出手法において使用したクラスタリング手法はいずれも FlexDice である．

4.1 収入データに対する実験

収入データではクラスタ数が約 20 個作成されたときのクラシフィケーションエラー E'_C を比較する．収入データを単純に FlexDice と OptiGrid でクラスタリングするとそれぞれ 22 個と 20 個のクラスタが形成された．このときのクラシフィケーションエラー E'_C はそれぞれ 0.466 と 0.334 であった．

属性の重要度算出手法では $Step$ 3 の閾値を 0.20 とすると，属性 6，8，10 が FIA の要素となった．収入データを収入が\$50,000 以下のラベル “0” データセット $DS0$ と\$50,000 より多いラベル “1” データセット $DS1$ をデータセットとして FIA に含まれる属性だけに基づいてクラスタリングすると，ラベル “0” データセット $DS0$ とデータセット $DS1$ はそれぞれ 42 個と 7 個のクラスタが形成された．$Step$ 6 ではラベル “1” データセット $DS1$ を入力として形成されたクラスタの値ベクトルをラベル “0” データセット $DS0$ を入力として形成された全クラスタの値ベクトルに 1 つずつ加えた値ベクトルデータセットにおいて特徴となる値ベクトルを抽出した．属性ごとのカウンタの値を重要度とした．ここで FIA から抽出さ

れた属性群はカウンタの値にかかわらず最も重要な属性とした．重要度ごとに属性の値を変換し OptiGrid でクラスタリングするとクラシフィケーションエラー E'_C は 0.307 であった．重要度ごとに各データ要素の属性の値を変換し，さらにダミーデータ要素を加えて FlexDice でクラスタリングするとクラシフィケーションエラー E'_C は 0.305 であった．このときの OptiGrid と FlexDice で形成されたクラスタ数は共に 20 個であった．

4.2 保険データに対する実験

保険データではクラスタが約 5 個作成されたときのクラシフィケーションエラー E'_C を比較する．保険データを単純に FlexDice でクラスタリングすると 6 個のクラスタが形成された．このときのクラシフィケーションエラー E'_C は 0.468 であった．

属性の重要度算出手法では *Step* 3 の閾値を 0.15 とすると，属性 44，47，59，65，68 が FIA の要素となった．保険データを未加入者のラベル “0” データセット $DS0$ と加入者のラベル “1” データセット $DS1$ を FIA から構成されるデータセットとしてクラスタリングするとラベル “0” データセット $DS0$ とラベル “1” データセット $DS1$ はそれぞれ 37 個と 6 個のクラスタが形成された．*Step* 6 ではラベル “1” データセット $DS1$ から形成されたクラスタの値ベクトルをラベル “0” データセット $DS0$ から形成された全クラスタの値ベクトルに 1 つずつ加え特徴となる値ベクトルを抽出した．属性ごとのカウンタの値を重要度とした．ここでも FIA から抽出された属性群はカウンタの値にかかわらず最も重要な属性とした．重要度ごとに属性の値を変換し，ダミーデータ要素を加えて FlexDice でクラスタリングするとクラシフィケーションエラー E'_C は 0.354 であった．このときの FlexDice で形成されたクラスタ数は 5 個であった．

4.3 実験の考察

第 4.1 節と第 4.2 節の結果から FlexDice と OptiGrid は我々の提案した重要な属性の検出手法に基づいて検出した属性の重要度を使用し，教師ありで分類クラスタリングすることで教師なし分類クラスタリングするよりも良い結果を出力できた．第 4.1 節の収入データに対する結果で FlexDice のみでデータ要素を集めた結果は OptiGrid のみでデータ要素を集めた結果よりも悪かったのは，クラスタ数をあらかじめ約 20 個に定めていたためと考えられる．

第 4.2 節の高次元な保険データへ適用した実験結果より，我々の提案した重要な属性の検出手法に基づいて検出した属性と，その重要度を用いたクラスタリング手法は高次元データセットに対して分類クラスタリングを容易にし，教師ありでデータ要素を集めることで教師なしのクラスタリングでデータ要素を集めるよりも良い結果を出力できた．保険データは高次元であるがデータ要素数が少なく，大規模，かつ，高次元なデータセットに適用できた例ではない．今後は大規模，かつ，高次元なデータセットを使用し，有効性を示したい．

今回の結果における属性の重要度を定めた後の各属性の値変換は，重要度に応じて変換した 1 例のみを使用した．良質な変換により，重要な属性の検出手法と重要度算出手法を使用したクラスタリング手法による分類クラスタリングのクラシフィケーションエラー E'_C は大きく改善すると考えられる．今後は様々な応用分野に適した変換方法を提案し，有効性を示したい．

5. おわりに

我々は，FlexDice や OptiGrid のように高次元なデータセットをクラスタリング可能な手法を用いることで，高次元なデータセットに対しても効率よく同一ラベルを持つデータ要素を集めること（分類クラスタリング）ができるのではないかと考えた．クラスタリング手法は一般に教師なし分類法であり，クラスタリング手法のみで分類クラスタリングすることは難しいが，入力パラメータの設定により様々な結果を出力できるため多くのユーザの要求に応えられる結果を出力可能である．クラスタリング手法を分類クラスタリングの教師あり手法として使用するために，第 2.1 節で重要な属性の検出手法を提案し，重要な属性検出手法に基づいて属性の重要度を算出した．

第 3.2 節において，バランスのとれていないベンチマークデータでも分類クラスタリングした結果の妥当な評価が可能な評価関数を定義した．この評価関数のことをクラシフィケーションエラー E'_C と呼ぶ．属性の重要度算出の有効性を調べるために，重要な属性検出手法から導かれた属性の重要度を入力データセットの値の変換に使用し，変換したデータセットにクラスタリング手法を適用して結果をクラシフィケーションエラー E'_C で評価した．

4. 章のシミュレーション実験では，教師なしクラスタリング手法の結果を分類クラスタリングの結果とするより，重要な属性検出手法に基づいた属性の重要度算出を用いてクラスタリング手法で教師ありで分類クラスタリングした結果の方が精度の高い分類ができていることを示した．また，クラスタリング手法を使用することで高次元なベンチマークデータに対して高精度に分類クラスタリングできたことを示した．

今後は属性の重要度決定後に使用する値変換手法を開発し，クラスタリング手法による教師ありクラシフィケーションの有効性を高めたいと考えている．

文　献

[1] C. C. Aggarwal, “Towards exploratory test instance specific algorithms for high dimensional classification,” Proc. of the 11st ACM-SIGKDD Int. Conf. on Knowledge discovery in data mining (KDD ’05), pp. 526–531, 2005.

[2] C. F. Eick, N. Zeidat and Z. Zhao, “Supervised clustering – algorithms and benefits,” Proc. of the 16th IEEE Int. Conf. on Tools with Artificial Intelligence (ICTAI ’04), pp. 774–776, 2004.

[3] J. Han and M. Kamber, “Data Mining: Concepts and Techniques,” Morgan Kaufmann, San Francisco, 2001.

[4] A. Hinneburg and D. A. Keim, “Optimal Grid-Clustering: Towards breaking the curse of dimensionality in high-dimensional clustering,” Proc. of the 25th Int. Conf. on Very Large Data Bases (VLDB ’99), pp. 506–517, 1999.

[5] T. Luo, L. O. Hall, D. B. Goldgof, “Bit reduction support vector machine,” Proc. of the 5th IEEE Int. Conf. on Data Mining (ICDM ’05), pp. 733–736, 2005.

[6] 中村 朋健, 上土井 陽子, 若林 真一, 吉田 典可, “FlexDice を用いたクラスタリング結果の特徴抽出”, 第 16 回データ工学ワークショップ (DEWS ’05) 論文集, 3C-o1, 2005.

[7] 中村 朋健, 上土井 陽子, 若林 真一, 吉田 典可, “FlexDice：高次元な大規模データセットに対する高速クラスタリング手法”, 情報処理学会論文誌: データベース (電子情報通信学会 データ工学研究専門委員会共同編集), Vol. 46, No. SIG 18 (TOD 28), pp. 40–49, 2005.

[8] B. L. Milenova and M. M. Campos, “O-Cluter: Scalable clutering of large high dimensional data sets,” Proc. of the 2nd IEEE Int. Conf. on Data Mining (ICDM ’02), pp. 290–297, 2002.

[9] B. L. Milenova and M. M. Campos, “Clustering large databases with numeric and nominal values using orthogonal projections,” Proc. of the 29th Int. Conf. on Very Large Data Bases (VLDB ’03), 2003.

[10] I. W. Tsang, J. T. Kwok and P.-M. Cheung, “Core Vector Machines: Fast SVM training on very large data sets,” Journal of machine learning research 6, pp. 363–392, 2005.

[11] The University of California, Irvine Knowledge Discovery in Databases Archive, “The insurance company benchmark (COIL 2000),” http://kdd.ics.uci.edu/.