社団法人 情報処理学会 研究報告
IPSJ SIG Technical Reports

2007－DBS－143（67）
2007／7／3

トランザクションデータベースに対する高確信度の相関ルールを用いた外れ値検出手法

成田 和世† 北川 博之†,††

† 筑波大学大学院システム情報工学研究科 〒305-8573 茨城県つくば市天王台1-1-1
†† 筑波大学大学院計算科学研究センター 〒305-8573 茨城県つくば市天王台1-1-1
E-mail: †narita@kde.cs.tsukuba.ac.jp, ††kitagawa@cs.tsukuba.ac.jp

あらまし データから珍しいイベントの発生や，他とは逸脱したオブジェクト，例外などを検出する外れ値検出は重要なデータマイニング技術の一つであり，近年注目を集めている．しかし，既存の多くの研究は，数値を持つ点の集合や数値型の属性を持つレコードデータを対象としており，カテゴリ型の属性やアイテムを持つデータに対する外れ値検出手法はほとんど研究されていない．我々は，特に実世界に数多く存在する，POSデータに代表されるようなトランザクションデータに着目し，高い確信度を持つ相関ルールの情報に基づいて，他のトランザクションと比べて例外的なふるまいをするトランザクション(**外れ値トランザクション**)を検出するためのフレームワークを提案する．相関ルールを利用したトランザクションデータからの外れ値検出の研究は，我々の知る限り他に存在しない．

キーワード 外れ値検出，相関ルール，頻出アイテム集合

Outlier Detection for Transaction Databases using Association Rules with High Confidences

Kazuyo NARITA† and Hiroyuki KITAGAWA†,††

† Graduate School of Systems and Information Engineering, University of Tsukuba,
1-1-1 Tennodai, Tsukuba, Ibaraki, 305-8573 Japan
†† Center for Computational Science, University of Tsukuba,
1-1-1 Tennodai, Tsukuba, Ibaraki, 305-8573 Japan
E-mail: †narita@kde.cs.tsukuba.ac.jp, ††kitagawa@cs.tsukuba.ac.jp

Abstract Outlier Detection, a data mining technique to detect rare events, deviant objects, and exceptions from data, has been drawing increasing attention in recent years. However, much existing research targets record data constructed with numerical attributes or a set of points having numeric values. Very few studies have attempted to detect outliers from data having categorical attributes or items. We focus on transaction databases typified by POS data, and propose a framework for detecting *outlier transactions* behaving abnormally compared to others based on information of association rules with high confidence. To the best of our knowledge, there are no studies detecting outliers from transaction data using association rules.

Key words outlier detection, association rules, frequent itemsets

1. はじめに

情報技術の発展に伴うデータの多様化，巨大化のため，データから有益な情報を抽出する技術であるデータマイニングはますます必要となっている[1]～[5].

近年注目されているデータマイニング技術の一つに，外れ値検出がある[5]～[8]．これは，珍しいイベントの発生や，他とは逸脱したオブジェクト，例外などを検出する技術であり，株価やセンサデータの異常数値発生の監視や，銀行やクレジットカード会社での極端な金額の引き落とし等の検知に用いられるなど，様々な分野で応用されている．

POSデータに代表されるようなトランザクションデータは，実世界に数多く存在しており，他のトランザクションと比べて例外的なふるまいをするトランザクション(**外れ値トランザクション**)の検出は，商品の買占め等の逸脱した行動を取る消費者の発見やノイズの発見，受注システムやロジスティックにお

ける例外的な注文や納品の検知などに役立つと期待できる．しかし，外れ値検出に関する既存の多くの研究は，数値を持つ点の集合や数値型の属性を持つレコードデータを対象としており，アイテムを持つトランザクションデータに対する外れ値検出手法はほとんど研究されていない．

そこで我々は，このようなトランザクションデータに着目し，高い確信度を持つ相関ルールの情報に基づいて，他と比べて例外的な動きをする外れ値トランザクションを検出するフレームワークを提案する．相関ルールを利用してトランザクションデータから外れ値検出を行う研究は，我々の知る限り他に存在しない．

本稿の寄与するところは以下の三点である．

トランザクションに対する外れ値度の定義

本稿は，アイテムのみで構成されたトランザクションデータベースから，他のトランザクションと比べて，あまり起こりえないふるまいするトランザクションを外れ値として検出することを目的とする．トランザクションが“あまり起こりえないふるまい”をしているかどうかを判断するのに，我々は高い確信度を持つ相関ルールの情報を利用する．例を出して説明する．表1は，ある店の購入履歴データの例である．各行が購入者が一度の買い物で購入した商品の内容を表している．一列目はトランザクション ID である．二行目がトランザクションであり，商品（アイテム）の集合で表される．表中，相関ルール $\{Milk\} \rightarrow \{Bread\}$ の確信度は 66.7%である．ここで，トランザクション 002，003 は，$\{Milk\}$ が発生しているにも関わらず，$\{Bread\}$ は発生していない．66.7%という高い確率で成り立つルール $\{Milk\} \rightarrow \{Bread\}$ に反しているこのトランザクションは，起こりにくいふるまいをしていると言える．また，相関ルール $\{Bacon\} \rightarrow \{Egg\}$ の確信度は 80%である．このルールにも反しているトランザクション 002 は，トランザクション 003 よりもさらに起こりにくいふるまいをしていると考えられる．

以上の観察のように，我々が外れ値として検出したいトランザクションは，高い確信度を持つ，言わば常識的な相関ルールにより反するふるまいをするトランザクションである．我々は，与えられたトランザクションデータベース上で，高い確信度で起こる相関ルールの情報を用いてトランザクションの外れ値度の式を導出する．我々が定義する外れ値度は，直観的考察と数学的考察の両方に基づいており，そこから得られる外れ値トランザクションは，外れ値として一定の説得力を持つと期待できる．

アルゴリズムの効率化のための工夫

一般的に，トランザクションデータには大量のトランザクションが存在する．我々が外れ値度の計算に利用する相関ルールは高い確信度を持つものに限定されるが，ルールの数はそれでも十分大きく，計算には時間がかかると考えられる．そこで，トランザクションデータから効率的に外れ値を検出するアルゴリズムを提案する．効率化の工夫点として，(i) 冗長な相関ルールの削除手法と，(ii) 極大頻出アイテム集合を利用した外れ値トランザクションの候補の絞込み手法を示す．

実験による提案手法の有効性の検証

実世界データを用いた実験では，本手法が有用な外れ値トランザクションを検出可能なことを示し，提案アルゴリズムがブルートフォースアルゴリズムに比べて高速に外れ値を検出できることを示す．

以下，本稿の構成は次のようになっている．2. 章で，本稿で用いる記述を定義する．3 章でトランザクションに対する相関ルールの情報を用いた外れ値度の式を導出する．4. 章では高速化の工夫を取り入れた外れ値検出アルゴリズムを提案する．5. 章で実世界データを用いた実験とその結果を示す．6. 章で，本研究の新規性，独自性を説明するため，関連研究に言及する．7. 章でまとめと今後の課題を述べる．

表1　商店の買い物データの例

TID	アイテムの集合
001	Bread, Jam, Milk
002	Bacon, Corn, Jam, Milk
003	Bacon, Egg, Jam, Milk
004	Bacon, Bread, Egg, Milk
005	Bacon, Bread, Egg, Jam, Milk
006	Bacon, Bread, Egg, Milk

2. 準　　備

ここでは，本稿で用いる用語や記述について定義する．トランザクションデータベース T はトランザクションの集合である．T の集合濃度 $|T|$ はトランザクションの個数である．トランザクション $t \in T$ はアイテムの集合であり，集合濃度 $|t|$ は t に含まれるアイテムの個数である．T に存在する全てのアイテムの集合を I，$X \subseteqq I$ をアイテム集合とすると，X の T における**サポート** $sup(X)$ は以下のように定義される．

$$sup(X) = \frac{|\{t | t \in T \wedge X \subseteqq t\}|}{|T|}$$

ユーザによって与えられるしきい値 min_sup に対して，$sup(X) \geq min\_sup$ となるアイテム集合 X を**頻出アイテム集合**と呼ぶ．また，min_sup を**最小サポート**と呼ぶ．

最小サポート min_sup によって得られる全ての頻出アイテム集合の中で，超集合となるアイテム集合が存在しない頻出アイテム集合を，**極大頻出アイテム集合**と呼ぶ．

$X \cap Y = \emptyset$ である二つのアイテム集合 $X, Y \in I$ に対して，X が発生したときに Y が発生する関係性を表す記述 $X \rightarrow Y$ を**相関ルール**と呼ぶ．相関ルール $X \rightarrow Y$ の**確信度** $conf(X \rightarrow Y)$ は次のように定義される．

$$conf(X \rightarrow Y) = \frac{sup(X \cup Y)}{sup(X)}$$

一般に，頻出アイテム集合の集合から生成される相関ルールの数は膨大であるので，ユーザによって与えられるしきい値 min_conf に対して，$conf(X \rightarrow Y) \geq min\_conf$ となる相関ルール $X \rightarrow Y$ のみを有用な情報として生成する．min_conf を**最小確信度**と呼び，ある相関ルール $X \rightarrow Y$ が最小確信度以上の確信度を持つとき，$X \rightarrow Y$ は最小確信度条件を満たすと言う．最小確信度条件を満たす相関ルールを，**高確信度ルー**

ルと呼ぶ．我々が焦点を当てているのは特に高い最小確信度によって得られた高確信度ルールである．特別に言及しない限り，高確信度ルールの確信度は非常に大きいことを想定している．

[定義 1] 違反ルール
アイテム集合$t \subseteq I$が，相関ルール$X \rightarrow Y$に対して$X \subseteq t \wedge Y \not\subseteq t$であるとき，$t$は$X \rightarrow Y$に反すると言い，このような相関ルール$X \rightarrow Y$を，$t$の**違反ルール**と呼ぶ．

3. 外れ値度

本章では，トランザクションデータベースから外れ値となるトランザクションを発見するため，相関ルールを用いた外れ値度を導出する．

高い確信度を持つ相関ルール$X \rightarrow Y$は，Xが発生したとき，高い確率でYが発生することを意味している．逆に，Xが起こったときにYが起こらない確率が低い．つまり，1. 章で観察したように，高確信度ルールに反するトランザクションは，起こりにくいふるまいをしていると考えられる．本稿では，高い確信度を持つ相関ルールが強い従属性を持っていることに着目し，トランザクション$t \in T$が，アイテム集合$X \subseteq t$に対して強い従属性を持っているアイテムをどれだけ含まないかを表す尺度を，外れ値度とする．

ここで，新しい概念を定義する．

[定義 2] 相関性閉包
アイテム集合$t \subseteq I$と，高確信度ルールの集合Rに対して，次のようにして求められるアイテム集合t^+を，tの**相関性閉包**と呼ぶ．

$$t^0 = t$$
$$t^{i+1} = t^i \cup \{e | e \in Y \wedge X \subseteq t^i \wedge X \rightarrow Y \in R\}$$
$$t^+ = t^\infty$$

アイテム集合t^{i+1}は，アイテム集合t^iに違反ルールが存在するとき成長し，存在しないとき収束して相関性閉包t^+となる．相関性閉包t^+は，tがアイテム集合$X \subseteq t$に従属するアイテムを多く含まないほど大きくなる．ここで，トランザクションの外れ値度を，この相関性閉包を用いて次のように導出する．

[定義 3] 外れ値度
トランザクション$t \in T$の外れ値度$od(t)$は，高確信度ルール集合Rに対するtの相関性閉包をt^+としたとき，次式で求められる．

$$od(t) = \frac{|t^+ - t|}{|t^+|}$$

以下に，トランザクションの外れ値を正式に定義する．

[定義 4] 外れ値トランザクション
トランザクション$t \in T$がユーザの与えるしきい値min_odに対して次の条件を満たすとき，tを外れ値と見なし，外れ値トランザクションと呼ぶ．

$$od(t) \geq min\_od$$

min_odを最小外れ値度と呼び，tがmin_od以上の外れ値度を持つとき，tは最小外れ値度条件を満たすと言う．

表 2 基本アルゴリズム（ブルートフォース）

入力: T, min_sup, min_conf, min_od
出力: 外れ値トランザクションの集合 OT

1. T, min_sup から頻出アイテム集合の集合 F を得る
2. F, min_conf から高確信度ルールの集合 R を得る
3. OT=getOutliers(T, R, min_od);

表 3 関数 getOutliers

入力: T, R, min_od
出力: 外れ値トランザクションの集合 OT

1. foreach $t \in T$ {
2. $t^+ = t$;
3. while t^+ が収束しない {
4. foreach $X \rightarrow Y \in R$ do $t^+ = t^+ \cup \{e | e \in Y \wedge X \subseteq t^+\}$;
5. }
6. $od(t)$ を計算
7. if $od(t) \geq min\_od$ then $OT = OT \cup \{t\}$;
8. }

4. 外れ値トランザクション検出アルゴリズム

本章では，3 章で導出した外れ値トランザクションを，トランザクションデータから効率的に検出するアルゴリズムを提案する．

表 2 は外れ値トランザクションを得る手法として，もっとも単純だと考えられるアルゴリズムを示している．このアルゴリズムは三つの処理によって構成されている．1 行目では，トランザクションデータ T と最小サポート min_sup を受け取って，頻出アイテム集合の集合 F を得る．実際の実装には FpGrowth [3] アルゴリズムを用いる．2 行目では，F と最小確信度 min_conf を受け取って，可能な全ての高確信度ルールの集合 R を得る．実際の実装には [2] の相関ルール生成アルゴリズムを用いる．最後に，3 行目の関数 getOutliers で，T と R，最小外れ値度 min_od を受け取って，各トランザクションの相関性閉包を得，外れ値度を計算し，全ての外れ値トランザクションを出力する．関数 getOutliers の具体的なアルゴリズムを表 3 に示す．各トランザクション $t \in T$ の外れ値度を計算するために，各高確信度ルールと t を比較する繰り返し処理を行っている．高確信度ルールの数 $|R|$ とトランザクションの総数 $|T|$ とが一般に巨大であることを考えると，処理時間は非常に大きくなることが予想できる．

提案アルゴリズムの特徴は，より高速な外れ値トランザクションの検出を実現するため，表 2 のブルートフォースアルゴリズムに改良を加え，それぞれ $|R|$，$|T|$ を小さくする二つの手法を用意した点にある．以下では最初にそれら二つの手法を説明し，次に具体的なアルゴリズムを述べる．

4.1 冗長なルールの除去

本小節では，外れ値度の計算に用いる高確信度ルールの数 $|R|$ を削減する手法について述べる．

外れ値度の計算を行うため，全てのトランザクション $t \in T$ について，R を用いて相関性閉包 t^+ を計算する．しかし，実際には，相関性閉包の計算に必要のない，冗長な高確信度ルールが，R の中には多く存在している．

[定義 5] 非冗長ルール

Rを最小確信度条件を満たす全ての相関ルールの集合であるとする．相関ルール$X \to Y \in R$が，$X \cup Y = Z \cup W \wedge X \supset Z$である相関ルール$Z \to W \in R$を持たず，かつ，$X = Z \wedge Y \subset W$である相関ルール$Z \to W \in R$を持たないとき，$X \to Y$を**非冗長ルール**と呼ぶ．また，$R$内の全ての非冗長ルールの集合を，**極小ルール集合**と呼ぶ．

極小ルール集合は，全ての相関ルールの集合から相関性閉包を作る上で冗長な相関ルールのみを削除した集合であるので，当然のことながら，以下の性質を持つ．

[性質 1] 頻出アイテム集合の集合Fから生成される全ての非冗長ルール集合R_{min}を用いて得られるトランザクションtの相関性閉包t^+_{min}と，Fから生成される全ての高確信度ルールの集合Rを用いて得られるtの相関性閉包t^+は，常に$t^+ = t^+_{min}$である．

上記の性質により，提案アルゴリズムでは，頻出アイテム集合の集合Fから高確信度ルールを生成する際，全ての高確信度ルールの集合Rを得ずに，非冗長ルールの集合R_{min}を得る．Fから非冗長ルールだけを生成する具体的なアルゴリズムの説明は後述する．

4.2 外れ値候補検出

効率化のための二つ目の手法は，外れ値度の計算をするトランザクションの数$|T|$を削減するのを目的とする．

今，定義 3 で導出した外れ値度が持つ次の性質に着目する．

[性質 2] $min_conf_1 < min_conf_2$である二つの最小確信度min_conf_1，min_conf_2に対するトランザクション$t \in T$の外れ値度を，それぞれ$od_1(t)$，$od_2(t)$とすると，必ず次のことが成り立つ．

$$od_1(t) \geqq od_2(t)$$

(証明) $min_conf_1 < min_conf_2$のとき，min_conf_1で得られる高確信度ルールの集合R_1と，min_conf_2で得られる高確信度ルールの集合R_2の包含関係は，$R_2 \subseteqq R_1$である．このことから，あるトランザクションtに対して，R_1から得られる相関性閉包を${t_1}^+$，R_2から得られる相関性閉包を${t_2}^+$とすると，${t_2}^+ \subseteqq {t_1}^+$である．すなわち，$min_conf_1 < min_conf_2$のとき，$|{t_2}^+| \leq |{t_1}^+|$である．定数$c$，変数$x$に対して，$\frac{x-c}{x}$は単調増加関数であるので，ある$t$に対して，外れ値度$od_1(t)$は，必ず外れ値度$od_2(t)$以上である．よって，$min_conf_1 < min_conf_2$のとき，$od_1(t) \geqq od_2(t)$が成り立つ．

ここで，新しい定義を導入する．

[定義 6] 極大相関性閉包

Tから得られる極大頻出アイテム集合の集合をMとする．このとき，アイテム集合$t \in I$に対して，次のようにして得られるt^+_{max}を，tの**極大相関性閉包**と呼ぶ．

$$t^0_{max} = t$$
$$t^{i+1}_{max} = t^i_{max} \cup \{e \in mi | mi \in M \wedge mi \cap t^i_{max} \neq \emptyset\}$$
$$t^+_{max} = t^\infty_{max}$$

[性質 3] ある最小サポートmin_supに対して，トランザクション$t \in T$の極大相関性閉包t^+_{max}は，min_supに対して$min_conf = 0\%$のときに作られるtの相関性閉包t^+と等しい．

(証明) 頻出アイテム集合の集合Fが与えられたとき，極大頻出アイテム集合$mi \in M \subseteqq F$に対して，X, Yを，それぞれ$X = mi \cap t^i_{max}$，$Y = mi - X$となるアイテム集合であるとする．$X \neq \emptyset$かつ$Y \neq \emptyset$のとき，XとYはどちらも頻出アイテム集合であり，相関ルール$X \to Y$は，$min_conf = 0\%$のときに作られる全ての高確信度ルールの集合内に必ず存在する．このとき，$t^i_{max} \cup Y = t^i_{max} \cup mi$は自明であるので，トランザクション$t$の極大相関性閉包$t^+_{max}$は，$min_conf = 0\%$のときに作られる$t$の相関性閉包$t^+$と等しい．

これより，外れ値度は与えられる最小サポートに対して上限値を得る．

[定義 7] 外れ値度の上限

t^+_{max}を，ある最小サポートmin_supに対して計算されるトランザクション$t \in T$の極大相関性閉包とする．このとき次のように計算される$od_{max}(t)$を，min_supに対するtの外れ値度$od(t)$の上限と呼ぶ．

$$od_{max}(t) = \frac{|t^+_{max} - t|}{|t^+_{max}|}$$

トランザクションtは，tの外れ値度の上限が最小外れ値度以上のとき，必ず最小外れ値度条件を満たす．

一般に，高確信度ルールの数$|R|$より，極大頻出アイテム集合の数$|M|$のほうが小さい．提案アルゴリズムでは，事前に極大頻出アイテム集合を得て，外れ値を計算する前に，全てのトランザクションの極大相関性閉包，および外れ値度の上限を順次計算し，上限が最小外れ値度以上のトランザクションに対してのみ，初めて外れ値度を順次計算する．次節でアルゴリズムについて説明する．

表 4 提案アルゴリズム

入力: T, min_sup, min_conf, min_od
出力: 外れ値トランザクションの集合 OT
1. T，min_supを用いて頻出アイテム集合の集合Fと極大頻出アイテム集合の集合Mを得る
2. F，min_confから極小ルール集合R_{min}を得る
3. T，M，min_odから外れ値トランザクションの候補の集合Cを得る
4. OT = getOutliers(C, R_{min}, min_od);

4.3 アルゴリズムの流れ

これまでの議論を踏まえて，提案アルゴリズムの全体の流れは表 4 のようになる．

ブルートフォースアルゴリズム (表 2) では頻出アイテム集合の集合Fのみを求めていたのに対し，提案アルゴリズムは同時に極大頻出アイテム集合の集合Mを求める．実装では，FpGrowth [3] アルゴリズムをベースとして頻出アイテム集合を求めながら [4] の MFI-tree と subsetChecking メソッドを利用して極大頻出アイテム集合を判定する．さらに表 4 の 2 行目で，非冗長ルールのみで構成された極大ルール集合R_{min}を得る．我々の実装したアルゴリズムは，[2] のアルゴリズムを，極大ルール集合を求めるために改良したものである．高確信度ルー

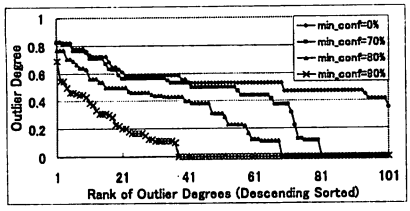

図 1　外れ値度の分布

ルを再帰的に生成しながら，$X \cup Y = Z \cup W \wedge X \supset Z$ である相関ルール $Z \rightarrow W \in R$ を持たない相関ルール $X \rightarrow Y \in R$ をマークする．再起的な処理を抜けた後，得られた全ての高確信度ルールのうち，$X = Z \wedge Y \subset W$ である相関ルール $Z \rightarrow W \in R$ を持たない相関ルール $X \rightarrow Y \in R$ のみをさらにマークする．最終的に，2 回マークされたルールを非冗長ルールとして出力する．3 行目で，M を用いて外れ値トランザクションの候補の集合 C を求める．このアルゴリズムは表 3 と同様の動作で定義 6 に従い極大相関性閉包を求め，外れ値度の上限が最小外れ値度 min_od 以上のトランザクションのみ出力する．4 行目で関数 getOutliers を呼び出し，候補トランザクションに対して正式に外れ値度の計算を行っている．

5. 実　　験

実験に使用したのは，Intel(R) Xeon(TM) 3GHz の CPU と 6GB のメインメモリを持つ RedHat Linux マシンで，Java 1.6.0 で実装した．まず，実験に使用したデータの説明をした後，データから得られる外れ値度の分布を観察し，外れ値トランザクションの検証結果，提案アルゴリズムの処理速度実験の結果を順次示す．

5.1 データセット

実験に使用したデータは実世界データ Zoo である．

Zoo データは，UCI Machine Learning Repository [9] にある動物の生態に関するレコード型のデータをトランザクションデータに変換したものである．元レコードデータは，各レコードに一種類の動物に関する情報が格納されており，動物の名前や足の数，体毛の有無，水生か否か，クラスなど，18 個の属性で構成されている．各動物は，哺乳類，鳥類，昆虫等の 7 クラスのうちのいずれか一つに分類される．動物の名前属性とクラス属性以外の，15 個のブール属性と 1 個の多値属性に対してトランザクションデータへの変換を行い，Zoo データを作成した．ブール属性に対しては，真であるとき，その属性名をアイテムとしてトランザクションに加えた．多値属性に対しては，記述「属性名：属性値」を一個のアイテムと見なした．Zoo データのトランザクション数は 101 個，アイテムの総数は 21 個である．さらに，速度比較実験で，処理時間の比較を行いやすくする目的で，Zoo データの各トランザクションを 200 個ずつ持つトランザクションデータ Zoo200 を用意した．Zoo200 データの全てのアイテム集合のサポート値は，Zoo データにおけるサポート値と等しい．

5.2 外れ値度の分布

まず，外れ値度がどのように分布するかを観察するため，全てのトランザクションに対して外れ値度の計算を行った．図 1 は，Zoo データの外れ値度の分布である．最小サポートを 10%とし，最小確信度を 0，70，80，90%としたときの各外れ値度を計算した．横軸は外れ値度の降順に並べたトランザクションに対応している．縦軸は外れ値度である．最小確信度が大きくなるほど，分散が小さくなり，より少数のトランザクションが比較的大きな外れ値度を持つようになっている．

5.3 外れ値トランザクションの検証

表 5 は $(min\_sup, min\_conf) = (10\%, 90\%)$ のときの，外れ値度の大きいもの上位三つのトランザクションと，対応する動物の名前 (Name) を表している．表 6 に，各相関性閉包 t^+ の成長に貢献した代表的な違反ルールのリストを示す．Crab の外れ値度が大きくなったのは，違反ルール {legs:4}→{toothed backbone breathes} の存在のおかげである．このルールは「4 本足の動物ならば歯と背骨があり肺呼吸をする」ことを表しており，典型的な哺乳類に当てはまるルールである．Crab は本来 8 本足であるが，ここでは「4 対の足」という意味でアイテム legs:4 が使われていると考えられる．しかし，アイテム legs:4 は本来「4 本足」を意味するものであり，「4 対の足」という意味で使われるのは明らかに例外的である．次に外れ値度が大きいのが Housefly と Moth である．これらのトランザクションは全く同一のアイテム集合を持っている．違反ルール {hair}→{milk, backbone, breathes} が哺乳類を表す典型的なルールであったため，その他の哺乳類，鳥類を表すルールまでが違反ルールとなった．元々，Housefly，Moth が属する昆虫クラスは，データ全体で 8 種類しか該当する動物がいない少数派クラスである．そのうち，hair をもつ昆虫は Housefly，Moth の他には 1 種類で，4 番目に外れ値度が大きな Wasp であった．

以上の検証から，導入した外れ値度が，高い確率で成り立つルールに反するほど外れ値であるという我々の想定に基き，外れ値として検出されるべきトランザクションほど高い値を示し，かつそれらのトランザクションが例外的なふるまいをする外れ値として説得力があるものであることが分かった．

表 5　Zoo データの外れ値トランザクション (Top-3)

名前	元トランザクション t
Crab	eggs, aquatic, predator, legs:4
Housefly	hair, eggs, airborne, breathes, legs:6
Moth	hair, eggs, airborne, breathes, legs:6

表 6　外れ値度に貢献した違反ルールのリスト

名前	違反ルールのリスト
Crab	{legs:4}→{toothed, backbone, breathes}
	{toothed, legs:4, tail}→{hair, milk, backbone, breathes}
	{eggs, aquatic, toothed, tail}→{backbone, fins, legs:0}
	{eggs, predator, backbone}→{tail}
	{hair, predator}→{milk, backbone, breathes, catsize}
Housefly	{hair}→{milk, backbone, breathes}
	{eggs, airborne, backbone}→{feathers, breathes, legs:2, tail}
	{hair, backbone}→{milk, toothed, breathes}
Moth	{eggs, airborne, backbone}→{feathers, breathes, legs:2, tail}
	{hair, backbone}→{milk, toothed, breathes}
	{hair}→{milk, backbone, breathes}

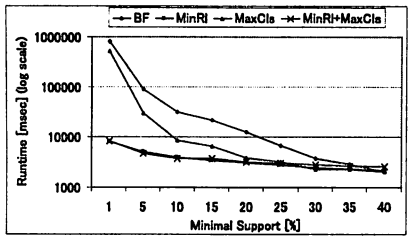

図 2 *min_sup* に対する処理時間

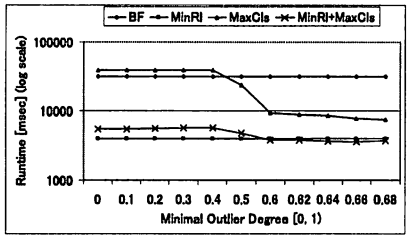

図 3 *min_od* に対する処理時間

5.4 処理時間の比較

ここでは，提案アルゴリズムが効率的に外れ値トランザクションが検出可能であることを確認する実験を行う．極小ルール集合を利用した効率化と，極大相関性閉包を利用した効率化がそれぞれどの程度有効であるかを調べるため，ブルートフォースアルゴリズム (BF)，極小ルール集合を利用した効率化のみを行った外れ値検出 (MinRl)，極大相関性閉包を利用した効率化のみを行った外れ値検出 (MaxCls)，二つの効率化を行う提案手法 (MinRl+MaxCls) の四通りの処理速度を比較する．

図 2 は Zoo200 における最小サポートの変化に対する処理時間の推移（ログスケール）を表している．$(min\_conf, min\_od) = (90\%, 0.6)$ である．最小サポートが小さいときほど BF は時間がかかり，効率化手法の有効性が増しているのが分かる．MinRl+MaxCls は $min\_sup \leqq 10\%$ に対しては MinRl よりわずかに処理時間が小さくなり，もっとも速いが，$min\_sup > 10\%$ では MinRl の処理時間がもっとも速くなった．また，$min\_sup \geqq 30\%$ では MaxCls が二番目に速くなった．最小サポートが大きい場合は，極小ルール集合のみを利用する手法で十分処理の高速化が可能であると言える．概して，MaxCls による効率化より，MinRl による効率化のほうがより処理速度を速くすることが可能であることが分かる．

さらに図 3 は最小外れ値度の変化に対する処理時間の推移（ログスケール）である．最小外れ値度が大きくなるに伴い，MaxCls の効率化手法が有効になっている様子が分かる．

6. 関連研究

この章では関連研究について，本研究との相違点，共通点を明らかにしながら言及する．

外れ値検出に関する研究はこれまでにも様々発表されてきた [5], [6]．しかし，これらの研究はどれも数値型のデータを対象としており，カテゴリ型のアイテムのみで構成されたトランザクションデータを外れ値検出の対象としている本研究とは異なる．

Z. He らは，[7], [8] で，本研究と同様，トランザクションデータに対する外れ値検出に関する研究を行っている．彼らは，データ中に頻繁に出現するはずの頻出アイテム集合を，より持たないトランザクションほど外れ値であるという考えの下で，外れ値度の式を導入している．頻出アイテム集合を利用した外れ値度を計算する彼らの手法で発見される外れ値トランザクションと，相関ルールを利用する本手法で検知される外れ値トランザクションは，当然ながら異なってくる．彼らの研究では，例えば集合濃度が小さいトランザクションならば，多くの頻出アイテム集合を包含していない可能性が大きく，容易に外れ値と見なされてしまう．一方，本研究では，トランザクションとその相関性閉包の差が重要であり，集合濃度の大きい小さいは本質的に外れ値度の大きさに影響しない．

7. おわりに

本稿ではトランザクションデータベースを対象とし，外れ値となるトランザクションを発見する手法を提案した．高い確信度を持つ相関ルールを用いることで相関性閉包の概念を定義し，外れ値度の式を設計した．また，外れ値トランザクションを効率よく検出するため，二つの工夫を取り入れたより高速なアルゴリズムを提案した．実世界データを用いた実験で，提案手法が情報として有効な外れ値トランザクションを検出できることを示した．また，ブルートフォースアルゴリズムとの処理速度比較実験で，提案したアルゴリズムがより高速に外れ値トランザクションを検出可能であることも見せた．相関ルールを利用したトランザクションデータベースからの外れ値検出の研究は，我々が知る限り他に行われていない．

今後の展望として，ストリームトランザクションの外れ値度を計算し，リアルタイムに外れ値トランザクションを検出，ユーザに通知するシステムの構築を考えている．

謝辞 本研究の一部は科学研究費補助金特定領域研究 (#19024006) の助成による．

文　献

[1] J. Han and M. Kamber, "Data Mining: Concepts and Techniques, 2nd ed.," The Morgan Kaufmann Series in Data Management Systems, 2006.

[2] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules in Large Databases," VLDB, 1994, pp. 487-499.

[3] J. Han, J. Pei and Yiwen Yin, "Mining Frequent Patterns without Candidate Generation," ACM SIGMOD International Conference on Management of Data, 2000, pp. 1-12.

[4] G. Grahne and J. Zhu, "Efficiently Using Prefix-trees in Mining Frequent Itemsets," FIMI, 2003.

[5] P. J. Rousseeuw and A. M. Leroy, "Robust Regression and Outlier Detection," John Wiley and Sons, 1987.

[6] S. Papadimitriou, H. Kitagawa, P. B. Gibbons and C. Faloutsos, "LOCI: Fast Outlier Detection Using the Local Correlation Integral," ICDE, 2003, pp. 315-.

[7] Z. He, J. Z. Huang, X. Xu and S. Deng, "Mining Class Outliers: Concepts, Algorithms and Applications," WAIM, 2004, pp. 589-599.

[8] Z. He, X. Xu and S. Deng, "FP-outlier: Frequent pattern based outlier detection," Technology Report, Harbin Institute of Technology, 2002.

[9] http://www.ics.uci.edu/ mlearn/MLRepository.html