トランザクションの異常行動検知システムの開発に関する研究

著者	成田 和世
雑誌名	2007年度CSテクニカルレポート・システム開発型研究プロジェクト特集号
発行年	2007
その他のタイトル	System Development for Detecting Outlier Transactions
URL	http://hdl.handle.net/2241/104475

トランザクションの異常行動検知システムの開発に関する研究

成　田　和　世[†1]

巨大なデータから珍しいイベントや逸脱したオブジェクト，例外等を発見する技術である外れ値検出は，幅広い応用範囲を持つことから，近年ますます注目されている．しかし，既存の外れ値検出手法は連続値を属性値として持つ数値型データを対象とするものが多い．我々はこれまで，トランザクションデータを対象とし，トランザクションが持つアイテム集合と本来ならば共起するはずのアイテムを，そのトランザクションが数多く持たないトランザクションを，外れ値と見なして検出する手法を提案してきた．本稿では，ユーザが本手法をよりインタラクティブに行い，かつ検出結果の解析を容易にするためのシステムの開発について述べた．

System Development for Detecting Outlier Transactions

Kazuyo Narita[†1]

Outlier detection, a data mining technique to detect rare events, deviant objects, and exceptions from data, has been drawing increasing attention in recent years. Most existing outlier detection algorithms focus on numerical data sets. Targeting transaction databases, we detect transactions in which many items are not observed even though they should occur in association with other itemsets in the transactions. In this paper we describe the development of a system for detectiong the outlier transactions.

1. はじめに

情報技術の発達に伴い，ディジタルデータは多様化，巨大化の一途を辿っている．膨大な量のデータを人手で検証し，重要な情報を特定することは不可能であることから，データから有益な情報を体系的に抽出するデータマイニング技術は，必要不可欠である．巨大なデータから珍しいイベントや逸脱したオブジェクト，例外等を発見する技術である外れ値検出は，幅広い応用範囲を持つことから，近年ますます注目されている．しかし，既存の外れ値検出手法は連続値を属性値として持つ数値型データを対象としたものが多い[1)–10)]．我々はこれまでの研究で，トランザクションデータを対象とし，データ中に存在する規則性や特徴から著しく逸脱しているトランザクションを外れ値トランザクションと見なして検出する手法を提案した[19),20)]．我々が想定する外れ値トランザクションは以下に述べる考えに基づくものである．

表 1 はある商店の購入履歴データの例である．各行が買い物客の一回分の買い物に対応している．一列目はトランザクション ID を表している．二列目はトランザクションであり，買い物客が購入した商品 (アイテム) の集合が示されている．また，サポートのしきい値 50%，確信度のしきい値 80%を与えたときに表 1 から生成される相関ルールを表 2 に示す．ただし，ここでは 100%の確信度を持つルールは無視する．

表 1　購入履歴データの例
Table 1　Purchase Data of a Store

TID	アイテム集合
001	Bread, Jam, Milk
002	Bacon, Corn, Jam, Milk
003	Bread, Jam, Milk
004	Bacon, Bread, Corn, Egg, Milk
005	Bacon, Bread, Corn, Egg, Jam, Milk
006	Bread, Corn, Jam, Milk
007	Bacon, Bread, Egg, Milk
008	Bacon, Bread, Egg, Jam, Milk
009	Bread, Jam, Milk
010	Bacon, Egg, Milk

ルール $\{Milk\} \rightarrow \{Bread\}$ が 80%以上の確信度を持つことから，$\{Bread\}$ は $\{Milk\}$ に対して強い相関性を持っていることが分かる．しかし表 1 では，トランザクション 002 と 010 が $\{Milk\}$ を含んでいるにも関わらず $\{Bread\}$ が存在していない．高い相関

†1 筑波大学大学院システム情報工学研究科
Graduate School of Systems and Information Engineering, University of Tsukuba

表 2 80%以上の確信度を持つ相関ルール
Table 2 Association Rules with a Minimal Confidence 80%

ID	相関ルール
$rule_1$	$\{Jam\} \rightarrow \{Bread\}$
$rule_2$	$\{Jam, Milk\} \rightarrow \{Bread\}$
$rule_3$	$\{Jam\} \rightarrow \{Bread, Milk\}$
$rule_4$	$\{Bacon\} \rightarrow \{Egg\}$
$rule_5$	$\{Bacon, Milk\} \rightarrow \{Egg\}$
$rule_6$	$\{Bacon\} \rightarrow \{Egg, Milk\}$
$rule_7$	$\{Milk\} \rightarrow \{Bread\}$

性を持つはずのルールに違反しているこれらのトランザクションは，起こりにくいトランザクションであると言える．さらにトランザクション 002 に至っては，このルールだけでなく，表 2 にある全てのルールに対して，左辺のアイテム集合を含んでいるが右辺のアイテム集合は持っておらず，違反していることが分かる．トランザクション 002 はトランザクション 010 よりも更に起こりにくいトランザクションであると言える．

以上の観察から，我々は，高い確信度を持つ相関ルールに数多く違反しているトランザクション，より正確に述べるならば，トランザクションが持つアイテム集合と本来ならば共起するはずのアイテムを数多く持たないトランザクションを，入力データ中の規則性から大きく逸脱した外れ値であると考える．このような外れ値トランザクションを発見するため，我々はこれまでの研究で，直観的で数学的見地からも説得力のある外れ値度の式を導出し，外れ値トランザクションを効率よく検出するためのアルゴリズムを提案した．実データと人工データを用いた多くの実験から，提案手法が (i) 外れ値として価値のあるトランザクションを検出可能であること，(ii) 十分な検出精度を有していること，(iii) ブルートフォースアルゴリズムに比べ，より高速な検出処理が実現できていることが分かっている．

しかし同時に，いくつかの実験結果から，提案手法にはパラメタセンシティビティがあり，ユーザが利用するには扱いにくい側面が存在するため，本稿では外れ値トランザクションの検出をユーザが簡単に行えるようにすることを目的とし，システムの開発を行う．以下，本稿の構成は次のようである．2 で本研究に関連する研究について言及する．3 で本稿で用いる記述や定義について述べる．4 では相関ルールに基づく外れ値度の式を導出する．5 では二つの処理高速化の工夫を取り入れた提案アルゴリズムを提示する．6 で，実データと人工データを用いた実験結果から提案手法の性能及び問題点を明らかにした後，7 でそれらの問題点を踏まえてユーザが本手法をより快適に利用できるようにするべく開発したシステムについて説明する．8 章でまとめる．

2. 関 連 研 究

これまで，数値を属性値として持つ数値型データを対象とした外れ値検出手法が数多く提案されてきた[1)–10)]．これらは連続値を扱う手法であり，カテゴリカルなアイテムで構成されるトランザクションデータを対象としている本研究とは，着目しているデータやアプローチがかなり異なる．

カテゴリカルな値を属性値に持つカテゴリ型レコードデータに対して，外れ値や例外となるレコードを検出する手法もいくつか研究されている[11)–14)]．ベイジアンネットワークに基づく手法では，確率モデルを立ててレコードの尤度を推定し，尤度値が非常に小さいレコードを例外として検出する[11),12)]．Chan らは本研究と同様，相関ルールを利用した検出手法を提案している[13)]．しかし，我々が提案する外れ値度が確率的な計算を必要としない一方で，彼らは確率論的な考えに基づき外れ値度を導出している[13)]．Das らもまた，確率的な計算に基づき例外となるレコードを検出する手法を提案している[14)]．彼らは，統計的にありえない属性値集合の組み合わせを持つレコードを異常値と見なし，レコード中に存在する全ての属性値集合の組み合わせに対して周辺確率を推定し，最小値を示す確率をそのレコードの正常度としている．これらの研究は，トランザクションデータを直接的には想定していない．また，本手法では彼らの研究と異なり，外れ値度が確率的な観点に基づいていない．

He らはトランザクションデータに対して，外れ値となるトランザクションを検出するための手法を提案している[?),?)]．彼らはより多くの頻出アイテム集合を含んでいないトランザクションほど珍しいトランザクションであると考え，頻出アイテム集合を利用した正常度の式を導出している．He らの手法は本手法とよく似ていることから，本稿では提案手法の検出精度評価実験において，彼らの手法を比較対象としている．そのため，彼らの手法を以下で概説する．F を頻出アイテム集合の完全集合，$sup(X)$ を頻出アイテム集合 $X \in F$ のサポート値とする．このとき，He らの定義するトランザクション t の正常度 $nd(t)$ は次の式で得られる．

$$nd(t) = \frac{\sum_{X \subseteq t \wedge X \in F} sup(X)}{|F|}.$$

$nd(t)$ が低いほど，対応するトランザクション t は異常であると見なされる．彼らは正常度の低い順にトップ k 個のトランザクションを外れ値として出力している．

3. 準　　備

ここでは，本稿で用いる記述や定義について説明する．トランザクションデータ DB はトランザクションの集合である．$|DB|$ は DB の集合濃度を表し，全てのトランザクションの数である．I は DB におけるアイテムのドメイン，すなわち，DB に存在する全てのアイテムの集合である．トランザクション $t \in DB$ はアイテムの集合で，$t \subseteq I$ である．

アイテム集合 $X \subseteq I$ の**サポート** $sup(X)$ は，以下の式で求められる．

$$sup(X) = \frac{\{t|t \in DB \wedge X \subseteq t\}}{|DB|}$$

ユーザが与えるしきい値 $msup$ に対して，$sup(X) \geq msup$ であるならば，X を**頻出アイテム集合**であるとする．このようなしきい値 $msup$ を**最小サポート**と呼ぶ．**極大頻出アイテム集合**とは，全ての頻出アイテム集合の集合の中で，超集合を持たない頻出アイテム集合のことである．アイテム集合 $X, Y \subseteq I$ $(X \cap Y = \emptyset)$ に対して，記述 $X \rightarrow Y$ を**相関ルール**と呼ぶ．このとき左辺のアイテム集合 X を**前提部**，右辺のアイテム集合 Y を**結論部**という．相関ルールの**確信度** $conf(X \rightarrow Y)$ は次式で計算される．

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

ユーザが与えるしきい値 $mconf$ に対して，条件 $conf(X \rightarrow Y) \geq mconf$ を満足する相関ルール $X \rightarrow Y$ を，本稿では**高確信度ルール**と呼ぶ．また，このようなしきい値 $mconf$ を**最小確信度**と呼ぶ．

4. 外 れ 値 度

我々が想定する外れ値トランザクションは，トランザクションが持つアイテム集合と本来ならば共起するはずのアイテムを数多く持たないトランザクションである．このようなトランザクションを発見するために，本稿では高い確信度を持つ相関ルールを利用する．

確信度 $conf(X \rightarrow Y)$ は，X が起こったときに同時に Y が起こる確率に等しい．よって明らかに，確信度値が大きいとき，アイテム集合 X に対して，アイテム集合 Y は共起する傾向にある．このとき，あるトランザクション t に X が包含されているならば，同時に Y 内の全てのアイテム $e \in Y$ も t に含まれているはずであり，t で e が発生しないことは稀であると考えられる．アイテム間の相関性に反するこのようなアイテム e が数多く存在するトランザクションが，外れ値であると考える．

定義 1　非充足相関ルール

R を相関ルールの集合，Z をアイテム集合とする．ある相関ルール $X \rightarrow Y \in R$ が，$X \subseteq Z$ であり，$Y \nsubseteq Z$ であるとき，$X \rightarrow Y \in R$ を Z の**非充足相関ルール**と呼ぶ．

今，トランザクション t の非充足相関ルールを完全に被覆する次のようなアイテム集合を考える．

定義 2　相関性閉包

t をトランザクション，R を相関ルールの集合とする．t に対して次のような処理を行い得られるアイテム集合 t^+ を，**相関性閉包**と呼ぶ．

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

定義 2 より，トランザクション t の相関性閉包は，非充足相関ルールが数多く存在するほど大きくなり，t とかけ離れていく．本稿では高い確信度値を持つ相関ルールを想定しているので，相関性閉包が元と比べて大きくなるような t ほど，入力データ中の規則性から大きく逸脱していると考える．

定義 3　外れ値度

トランザクション t の外れ値度は $od(t) = \frac{|t^+ - t|}{|t^+|}$ で定義される．

ここで，再び表 1 と表 2 を例に，トランザクションの相関性閉包と外れ値度を算出する様子を見せる．今，トランザクション 002 を t とし相関性閉包の計算を行うと，まず定義**??**に従い，t^0 が得られる．トランザクション 002 は表 2 の全ての相関ルールに違反しているので，各ルールの結論部の和集合 $\{Bread, Egg, Milk\}$ を用いて，t^1 は以下のようになる．

$$t^0 = \{Bacon, Corn, Jam, Milk\}$$
$$t^1 = t^0 \cup \{Bread, Egg, Milk\}$$
$$= \{Bacon, Corn, Bread, Egg, Jam, Milk\}$$

再び表 2 の各ルールと t^1 を見比べ，非充足ルールがないかどうか調べる．ここでは明らかに $t^1 = t^2$ である

表 3 ブルートフォースアルゴリズム
Table 3 Brute Force Algorithm

入力: DB, $msup$, $mconf$, mod	
出力: a complete set of outliers O	
1.	頻出アイテム集合の集合 F をマイニング;
2.	F から高確信度ルールの集合 R を生成;
3.	O = getOutliers(DB, R);

表 4 関数 getOutliers
Table 4 Function getOutliers

Function getOutliers(トランザクション集合 DB, 相関ルール集合 R)	
1.	$O = \emptyset$;
2.	foreach $t \in DB$
3.	$t^0 = t$;
4.	$i = 0$, $R' = R$;
5.	until t^i が成長をやめる
6.	$TMP = \emptyset$;
7.	foreach $r = X \rightarrow Y \in R'$
8.	if $X \subseteq t^i$ then
9.	Y を TMP に加え, r を R' から削除;
10.	$t^{i+1} = t^i \cup TMP$, $i++$;
11.	$od(t) = \frac{\|t^i - t\|}{\|t^i\|}$
12.	if $od(t) \geq mod$, then t を O に加える;
13.	return O

ので，繰り返しは終了し，t の相関性閉包 $t^+ = t^1$ が得られる．よって，トランザクション 002 の外れ値度は $|t^+ - t| = 2$ と $|t^+| = 6$ より，$2/6 = 0.33$ である．

定義 4 外れ値トランザクション
しきい値 mod に対して，$od(t) \geq mod$ であるとき，トランザクション t を外れ値と見なす．また，このような mod を**最小外れ値度**と呼ぶ．

5. 外れ値検出アルゴリズム

本節では提案した外れ値度に基づき外れ値トランザクションを検出するためのアルゴリズムについて述べる．最初に，最もナイーブなアルゴリズムを表 3 に示す．表 3 ではまず，頻出アイテム集合をマイニングし (1 行目)，そこから高確信度ルールを生成する (2 行目)．得られたルールを引数として，関数 getOutliers を呼び出し入力データ中の全てのトランザクションの外れ値度を計算し，外れ値トランザクションを出力する．関数 getOutliers のアルゴリズムは表 4 に別途リストする．getOutliers は各トランザクション $t \in DB$ に対して，定義 2 で定めた処理に基づき相関ルール集合 R 内の相関ルールを一つ一つチェックしながら，相関性閉包を作成する．

このようなナイーブアルゴリズムの計算オーダーはトランザクションの集合 DB と相関ルールの集合 R に対して $O(|DB| \times |R|)$ である．一般に $|DB|$ と $|R|$ が巨大であることから，処理には時間がかかると考えられる．そこで処理の高速化を目的として，外れ値度の計算が真に必要とされるトランザクションの絞り込みと，計算に必要でない冗長な相関ルールを削除する工夫を取り入れる．

以下では処理を高速化するための二つの工夫について順番に言及する．

5.1 候補外れ値トランザクションの絞り込み

外れ値トランザクションになる見込みのないトランザクションを，外れ値度を計算する前に刈り取ることで，処理速度の向上が期待できる．このような候補外れ値トランザクションの絞り込みは，外れ値度の上限を求めることで可能となる．今，以下に示す外れ値度の性質に着目する．

性質 1 ある最小サポートが与えられたとき，二つの異なる最小確信度 min_conf_1，min_conf_2 ($min\_conf_1 < min\_conf_2$) に対して，各々から計算でき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，両者の間には次の関係が成り立つ．

$$od_1(t) \geq od_2(t)$$

証明 1 ある最小サポートから得られた頻出アイテム集合の集合に対して，R_1 をしきい値 min_conf_1 から生成される高確信度ルールの集合，R_2 をしきい値 min_conf_2 から生成される高確信度ルールの集合とする．$min\_conf_1 < min\_conf_2$ であるとき，$R_2 \subseteq R_1$ であることは自明である．このとき，あるトランザクション t に対して，${t_1}^+$ を R_1 から作られる t の相関性閉包，${t_2}^+$ を R_2 から作られる t の相関性閉包とすると，当然 ${t_2}^+ \subseteq {t_1}^+$ である．すなわち，$min\_conf_1 < min\_conf_2$ であるとき，$|{t_2}^+| \leq |{t_1}^+|$ となる．ある定数 c と変数 x に対し，$\frac{x-c}{x}$ は単調増加関数であることから，ある t に対して外れ値度は単調増加するため，$od_1(t)$ は $od_2(t)$ より小さくなる．したがって，$min\_conf_1 < min\_conf_2$ であるとき，$od_1(t) \geq od_2(t)$ が成り立つことは正しい．

定義 5 極大相関性閉包
M を最小サポート値 $msup$ に対する全ての極大頻出アイテム集合の集合，t をトランザクションとする．t に対して次の処理により得られるアイテム集合 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 2　M を最小サポート値 $msup$ に対する全ての極大頻出アイテム集合の集合，t^+_{max} を M におけるトランザクション t の極大相関性閉包とする．このとき，t^+_{max} は最小サポートが $msup$ で，$mconf = 0$ のときの t の相関性閉包に等しい．

証明 2　$mi \in M$ に対して，あるトランザクション t が $t \cap mi \neq \emptyset$ であるとき，X を $t \cap mi$，Y を $mi - X$ とすると，相関ルール $X \rightarrow Y$ は $mconf = 0\%$のときの高確信度ルールの集合 R_0 の中に，必ず含まれている．よって，上記性質は正しく成り立つ．

性質 1 と性質 2 から，外れ値度の上限が次のように得られる．

定義 6　外れ値度の上限
R を相関ルールの集合，M を極大頻出アイテム集合の完全集合であるとする．トランザクション t の外れ値度の**上限** $od_{max}(t)$ は，$od_{max}(t) = \frac{|t^+_{max} - t|}{|t^+_{max}|}$ で得られる．

極大頻出アイテム集合の集合 M は，頻出アイテム集合をマイニングする際に得ることができる．これを用いて，外れ値度の計算を行う前に全てのトランザクションを一度チェックし，外れ値度の上限 $od_{max}(t)$ を計算する．$od_{max}(t) < mod$ であるトランザクション t は，外れ値トランザクションに成り得ないために，事前に刈り取る．一般に，$|R| >> |M|$ であるため，R を用いて外れ値度をきちんと計算しなければならないトランザクションの数が減れば，処理速度の向上が期待できる．

5.2 冗長ルールの除去

高確信度ルールの集合 R の中には，相関性閉包の作成に冗長なルールが存在している．このようなルールは，外れ値度の計算には必要ないので，あらかじめ除去することとする．

再び表 2 を例に取り説明する．定義**??**より，相関性閉包 t^+ は，t^i に非充足相関ルールが存在するとき成長する．ここで，$\{Bacon\} \in t$ であるようなトランザクション t を考える．このとき，表 2 のルール $rule_6$ の結論部 {Egg, Milk} は，t の相関性閉包を作るのに用いられる．一方で，ルール $rule_4$ の結論部 {Egg} も，同様に用いられることが分かる．しかし，$rule_4$ の利用は明らかに冗長である．何故なら，アイテム Egg が $rule_4$ によって t の相関性閉包に追加されるとき，Egg は必ず $rule_6$ によっても t の相関性閉包に追加されるからである．また，$rule_5$ も冗長ルールである．何故ならば，$rule_5$ が相関性閉包の成長に用いられるトランザクションは必ず $\{Bacon, Milk\}$ を持っているが，このとき，必ず $rule_6$ も，相関性閉包の成長に用いられるからである．

表 5　提案アルゴリズム
Table 5　Proposed Algorithm

入力: DB, $msup$, $mconf$, mod
出力: 外れ値トランザクションの完全集合 O

```
1.  頻出アイテム集合の集合 F, 極大頻出アイテム集合の集合 M を得る;
2.  /* 外れ値の候補トランザクションの集合 C を得る */
3.  foreach t ∈ DB
4.    t の極大相関性閉包 t^+_max を作成;
5.    if od_max(t) ≥ mod, t を C に加える;
6.  if C = ∅ then exit;
7.  /* 最小ルール集合 R_min を得る */
8.  foreach l_k ∈ F (k ≥ 2)
9.    H_1 = {{h ∈ l_k | conf((l_k − {h}) → {h}) ≥ mconf};
10.   R_min = genMinRule(l_k, H_1);
11. foreach マークのない r_1 ∈ R_min と，異なる r_2 ∈ R_min
12.   if r_2 が定義 7 の条件 (ii) を満たす
13.     r_1 にマークをつける;
14. マークのついているルールを R_min から除去;
15. O = getOutliers(C, R_min);
```

上記のような観察から，ある最小確信度によって生成される全ての高確信度の集合の中には，明らかに相関性閉包を作るのに冗長であるルールが存在している．冗長なルールの定義を次にまとめる．

定義 7　冗長ルール
R を相関ルールの集合とする．相関ルール $X \rightarrow Y \in R$ は，次の二つの条件のうち，少なくともいずれか一つを満たすような他の異なる相関ルール $V \rightarrow W \in R$ が存在するとき，R において冗長ルールであると言える．
(i) $V \subset X \wedge V \cup W = X \cup Y$.
(ii) $V = X \wedge W \supset Y$.

高確信度ルールを生成する処理の中で，我々は定義 7 で定義される冗長ルールを全て除去し，残りの高確信度ルールだけを相関性閉包の作成に利用する．このような残りの高確信度ルールの集合を，我々は**最小ルール集合**と呼ぶ．表 2 の場合，最小ルール集合は $\{rule_3, rule_6, rule_7\}$ である．

トランザクション t に対し，t^+ を相関ルールの集合 R によって作成された相関性閉包，t^+_{min} を R に対する最小ルール集合 R_{min} から作成された相関性閉包とする．このとき必ず $t^+_{min} = t^+$ である．

表 6 関数 genMinRule
Table 6 Function genMinRule

```
Function genMinrule(頻出アイテム集合 l_k,
                    集合濃度 m のアイテム集合の集合 H_m)
 1.  if k > m + 1
 2.    TMP_1 = ∅;
 3.    H_{m+1} = apriori-gen(H_m);
 4.    foreach h_{m+1} ∈ H_{m+1}
 5.      r = h_{m+1} → (l_k − h_{m+1});
 6.      if conf(r) ≥ mconf then r を TMP_1 に加える;
 7.      else H_{m+1} = H_m − h_{m+1};
 8.    TMP_2 = genMinRule(l_k, H_{m+1});
 9.    foreach r_1 = X → Y ∈ TMP_1
10.      foreach r_2 = V → W ∈ TMP_2
 6.      if conf(r) ≥ mconf then appendr to TMP_1;
11.        if r_2 が定義 7 の条件 (i) を満たす
12.          r_1 にマークをつける;
13.  return TMP_1 ∪ TMP_2;
```

5.3 外れ値トランザクション検出アルゴリズム

これまでで説明した二つの処理速度向上のための工夫を取り入れた提案アルゴリズムを，表 5 に示す．

1 行目で，提案アルゴリズムは頻出アイテム集合をマイニングすると共に，全ての極大頻出アイテム集合を得る．頻出アイテム集合をマイニングしながら極大頻出アイテム集合を得るために，我々は FP-growth アルゴリズム[15] を改良したものを実装した．これは，[17] で提案されている MFI-tree と subset-checking メソッドを利用したものである．3-6 行目で，外れ値の候補トランザクションの絞り込みを行う．外れ値度の上限を計算する手順は，外れ値度を計算する手順と非常に良く似ているため，ここではスペースの関係上，説明を省く．

8-14 行目で，高確信度ルールを生成しながら最小ルール集合を得るための処理が行われている．高確信度ルール生成のアルゴリズムは基本的に[16] の手法に準じている．関数 genMinRule(表 6) を再帰的に呼び出しながら，まず定義 7 の条件 (i) に当てはまる高確信度ルールを冗長ルールであるとしてマークする．表 6 の 3 行目で呼び出される apriori-gen は，H_m から集合濃度が $m+1$ の可能な全てのアイテム集合を生成する関数であり，[16] で提案されているものである．表 5 の 9 行目以降，再帰的処理を抜けると，今度は定義 7 の条件 (ii) に当てはまる高確信度ルールがないか調べ，もし存在していたら，それは冗長ルールであるとしてマークする．最後に，これまでの処理でマークされたルールを，冗長ルールであるとして全て削除し，残りの高確信度ルールの集合を最小ルール集合として得る．最後の行で，提案アルゴリズムは関数 getOutliers を呼び，候補として残ったトランザクションに対してのみ外れ値度をきちんと計算し，外れ値トランザクションの集合を得る．

6. 実　　験

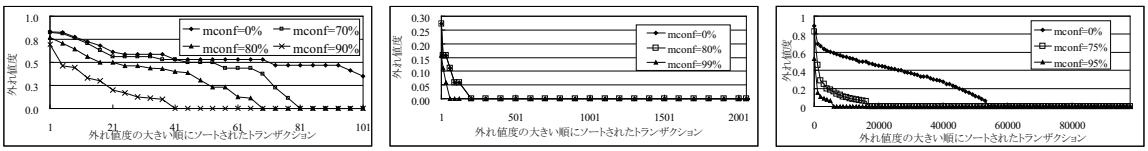
始めに実験に使用した実データについて説明し，それから本システムで検出される外れ値トランザクションの検証，He らの関連研究[?] を比較対象とした検出精度評価，ブルートフォースアルゴリズムを比較対象とした処理の高速性の評価の順で，実験結果を示す．

6.1 データセット

実験で使用したのは UCI Machine Learning Repository[18] で配布されている Zoo データ，KDD Cup 99 データ，および IBM のデータ生成器で生成した人工データ Synthetic である．

Zoo データは，元々は動物の生態を表したデータベースであり，15 個のブール値属性と 2 個の非バイナリ属性を持つレコードデータである．各レコードが一匹の動物に対応しており，哺乳類，鳥類，爬虫類，両生類，魚類，昆虫，昆虫以外の節足動物の 7 クラスのうちの 1 つに分類される．レコードの総数は全部で 101 個である．ここから動物名属性とクラス属性を削除した 16 個の属性からなるデータを，以下の手順でトランザクションデータに変換した．ブール値属性に対しては，属性値が真の場合のみ，属性名をアイテムと見なし，対応するトランザクションに追加した．非バイナリ属性に対しては，属性と属性値のペアを一個のアイテムと見なして対応するトランザクションに追加した．こうして変換したトランザクションデータを Zoo と名づけ，利用する．さらに，処理速度の評価実験で処理時間を比較しやすくするため，Zoo のトランザクションを各 10,000 個ずつ持つデータセット Zoo10000 を用意した．ここで，Zoo10000 における各アイテム集合のサポート値は，Zoo におけるそれと等値である．

KDD Cup 99 は，元々はネットワークへのアクセスログ情報を持つレコードデータである．各レコードは正常クラスと，guess password クラスや warezmaster クラスなどの不正侵入を表すクラスのいずれか一つに属している．不正侵入を表すクラスの一つである guess password に属するレコードを正解外れ値とみなし，97%の正常クラスに属するレコードと 3%の正解外れ値から成るデータセットを作成した．ほとんどの属性が連続値であるため，それらの属性値を 5 つのレベルに区分しカテゴリ化し，属性と属性値のペアを一個のアイテムと見なして対応するトランザクションに追加した．こうして変換したトランザクションデー



(a) Zoo (b) KDD Cup 99 (c) Synthetic

図 1 外れ値度の分布
Fig. 1 Distribution of Outlier Degrees

タを KDD Cup 99 とする．

Synthetic データは，異なるパラメタセットを与えて生成した二種類のデータから成り立っている．一つはデフォルトのパラメタセットを与えて作ったデータで，10,000 個のアイテムをドメインとして持ち，約 9,7000 個のトランザクションから構成されている．もう一方は同じアイテムのドメインを持つが，アイテム集合の分布の仕方は異なるようにパラメタセットを与えて生成したデータで，約 1,100 個のトランザクションで構成されている．これを正解外れ値トランザクションの正解と見なし，二つのデータを統合し一つにまとめたデータが，実験で使用する Synthetic データである．

Zoo データを提案手法で検出される外れ値トランザクションの検証と処理速度の評価実験に用い，KDD Cup 99 と Synthetic データを精度評価実験と処理速度の評価実験に用いる．

6.2 外れ値度の分布

まず，提案外れ値度がどのように分布するかを観察するため，各データセットの全てのトランザクションの外れ値度を計算した．最小サポート $msup$ を，Zoo データに対しては 10%に，KDD Cup 99 データに対しては 95%に，Synthetic データに対しては 0.05%に固定した．そして，いくつかの $mconf$ 値に対してトランザクションの外れ値度を計算した．図 1a-c の横軸は，外れ値度の大きい順に左からソートされたトランザクションのランクに対応している．縦軸が外れ値度を表している．これらの図から，$mconf$ 値が大きくなるほど，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様子が分かる．図 1c では，$mconf = 95\%$に対する折れ線が，完全に $mconf = 0\%$に対する折れ線と一致している．これは，KDD Cup 99 データにおいて，生成される相関ルールの確信度がどれも非常に大きいことを表している．

6.3 外れ値トランザクションの検証

ここでは提案手法を Zoo に適用し，外れ値として得られたトランザクションが真に外れ値として説得力のあるものであるかどうかを検証する．

表 7 は Zoo から $(msup, mconf) = (20\%, 90\%)$ で外れ値検出を行ったときの，外れ値度の大きい上位 3 件のトランザクションを示している．一列目にはトランザクションが表している動物の名前が，二列目にはトランザクションが，三列目にはそのトランザクションの典型的な非充足相関ルールがリストされている．カニ (Clab) は本来 8 本足のはずだが,「4 対の足」という意味で，ここではアイテム (# of Legs, 4) が用いられていると考えられる．しかしながら，(# of Legs, 4) は通常「4 本足」を意味するアイテムであり,「4 対の足」を意味する使い方は明らかに例外であると言える．イエバエ (Housefly) とガ (Moth) は，それぞれの対応するアイテム集合が同一であるため，外れ値度が等しくなった．元々，イエバエやガのような昆虫クラスに属するトランザクションは，全体でも 8 種類しか存在しない．その中でも，アイテム Hair を持つ昆虫はイエバエとガ，そして 4 番目に外れ値度の大きかったカリバチ (Wasp) のみである．Hair は，比較的大きなクラスである哺乳類や鳥類に属するトランザクションで頻出するアイテムである．哺乳類や鳥類の持つ「毛」と昆虫の持つ「毛」は本来異なるものであるにも関わらず，Zoo データでは同一のアイテム Hair で表されているため，少数派のクラスで Hair を持つトランザクションの逸脱性が大きくなった．

提案手法で検出されたこれらの外れ値トランザクションは，例外的な属性値の組み合わせを持つ珍しいトランザクションであり，外れ値として説得力のあるものであると言える．

6.4 精 度 比 較

ここでは提案手法と関連手法[7] の検出精度を調べ，比較する．精度を測る尺度として，Detection Rate (d_rate) と Detection Precision (d_prec) を利用している．各尺度は以下の式で求められる．

$$d\_rate = \frac{\text{検出された正解外れ値の数}}{\text{全正解外れ値の数}}$$

$$d\_prec = \frac{\text{検出された正解外れ値の数}}{\text{実際に検出された外れ値の数}}$$

表 7 Zoo における上位 3 件の外れ値トランザクション
Table 7 Top-k Outlier Transactions on Zoo

動物名	トランザクション	典型的な非充足相関ルール
Crab	{Eggs, Aquatic, Predator, (# of Legs, 4)}	{(# of Legs, 4)}→{Toothed, Backbone, Breathes}
Housefly/Moth	{Hair, Eggs, Airborne, Breathes, (# of Legs, 6)}	{Hair}→{Milk, Backbone, Breathes}

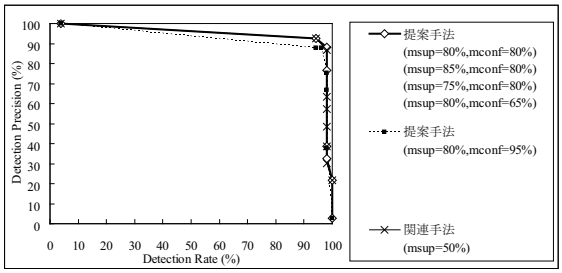


図 2 検出精度 (KDD Cup 99)
Fig. 2 Accuracy Comparison (KDD Cup 99)

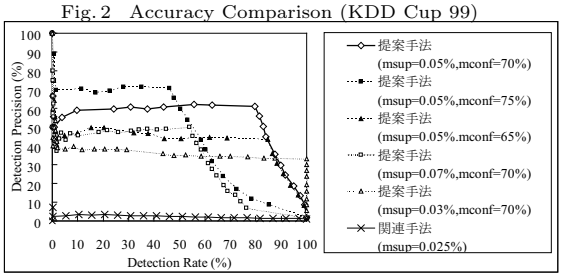


図 3 検出精度 (Synthetic)
Fig. 3 Accuracy Comparison (Synthetic)

異なる入力パラメタを持つ提案手法と関連手法を公平に評価するため，各手法のパラメタを固定し，外れ値度の大きさの順にソートされたトランザクション上位 k 件を外れ値と見なしたときの精度を測った．両手法共に，最大 F-measure ($\frac{2\times d\_rate\times d\_prec}{d\_rate+d\_prec}$) が最も良くなる最適なパラメタセットを与えて精度グラフを作成した．また，提案手法に対しては，パラメタのセンシティビティを観察するために，最適パラメタを前後に動かしたときに得られる精度の動きについてもグラフに載せている．

図 2 は KDD Cup 99 データに対する精度グラフである．提案手法にはパラメタセット (min_sup, min_conf) = (80%, 80%) を与え，関連手法には min_sup = 50%を与えた．両者の精度の動きはほぼ一致し，最大 F-measure は共に 94.4% (d_rate = 96.2%, d_prec = 92.7%) を記録した．パラメタのセンシティビティを観察するため，提案手法で min_sup と min_conf をそれぞれ上下に動かしたときの精度の動きも同図に載せている．(min_sup, min_conf) = (80%, 95%) を与えたときに得られる精度は，(min_sup, min_conf) = (80%, 80%) のときと比べて若干下がったが，それでも十分な精度が得られている様子が分かる．その他のパラメタセットを与えたときは，(min_sup, min_conf) = (80%, 80%) と結果が変わらなかった．

図 3 は Synthetic データに対する精度グラフである．提案手法に (min_sup, min_conf) = (0.05%, 70%) を与えたとき，最も良い最大 F-measure 値 69.3%が得られた (d_rate = 79.8%, d_prec = 61.2%)．関連手法は，min_sup = 0.025%のとき最も良い最大 F-measure 値 5.7%が得られている (d_rate = 25.7%, d_prec = 3.2%)．明らかに関連手法は，正解外れ値トランザクションをほとんど検出できていない．これは，Synthetic データが KDD Cup 99 と比べて，アイテム間の相関性があまり強くない性質があるためと考えられる．最小サポート値が高いと，Synthetic データからマイニングされる頻出アイテム集合の数は非常に少なくなるため，関連手法では多くのトランザクションで正常度が 0 となり，正常度が著しく低いとして外れ値トランザクションと見なされた．同様に，提案手法でも最小サポート値が高い場合は，高確信度ルールがほとんど生成されないために多くのトランザクションが外れ値と見なされた．

しかし，逆に最小サポート値が低い場合は，提案手法では十分な高確信度ルールが得られて，高い検出精度を示すことが出来た一方で，関連手法ではあまりに多くの頻出アイテム集合がマイニングされすぎてしまい，どのトランザクションの正常度も大きくなり，ほとんど差がつかなくなった．結果として，単純に頻出アイテム集合をあまり持ってない，集合濃度の小さいトランザクションが外れ値として検出されることとなった．

提案手法のパラメタセンシティビティを観察するため，min_sup = 0.05%に対して最小確信度値 65%と 75%を，min_conf = 70% に対して最小サポート値 0.03%と 0.07%を与えて精度を測った．与えるパラメタセットが異なると，得られる外れ値トランザクションが異なってくる様子が，図 3 から分かる．パラメタセンシティビティに関するより詳しい分析は後で述べる．

結論として，関連手法と異なり，提案手法ではアイテム間の相関性があまり強くない，より一般的なトランザクションデータに対しても，十分な検出精度を得

ることが出来ることが分かった．

6.5 処理速度比較

ここでは，提案アルゴリズムにより処理速度が向上している様子を確認する．以下の 4 つの手法で処理速度を比較する．一つは，表 3 に示されているブルートフォースアルゴリズム (BF) である．二つ目は，最小ルール集合を利用した冗長ルールの除去工夫のみを取り入れたアルゴリズム (MinR)，三つ目は，極大相関性閉包を利用した候補外れ値トランザクションの絞込み工夫を取り入れた手法 (MaxC)，そして最後に，冗長ルールを除去し，候補トランザクションを絞り込む二つの工夫を取り入れた手法 (MinR+MaxC) である．

図 4–6 は，パラメタ $msup$ を変化させたときの各データに対する処理時間の変動の様子を，ログスケールで表したものである．Zoo10000 データに対しては，パラメタ $(mconf, mod) = (90\%, 0.6)$ を与えた (図 4)．KDD Cup 99 データに対しては，パラメタ $(mconf, mod) = (95\%, 0.15)$ を与えた (図 5)．Synthetic データに対しては，パラメタ $(mconf, mod) = (75\%, 0.4)$ を与えた (図 6)．どのグラフでも，$msup$ 値が小さくなるにつれ，BF は非常に処理に時間がかかるようになっていき，提案手法の有効性が増していっている様子が分かる．Zoo10000 データでは，ある程度小さな $msup$ 値に対して MinR+MaxC を用いることで最大の高速性を得られた．全体的に，MinR は MaxC よりもずっと処理時間を削減できていることが分かった．

6.6 パラメタセンシティビティ

これまでに挙げられた実験結果から，提案手法の外れ値トランザクションを検出する能力は十分であると言える．しかし，与えるパラメタセットによって検出されるトランザクションが変化することから，パラメタセンシティビティが存在することも分かっている．本手法の使い方として，ユーザが入力パラメタセットを変えて何度も本手法を実行し，結果を検証していくことは，容易に想像できる．また，与えるパラメタによって最速の処理アルゴリズムが変わってくることも，ユーザにとっては使いにくい点である．以上のことから，本稿ではより快適に本手法を利用できるようにする目的で，ユーザがよりインタラクティブにパラメタを与えて本手法を実行し，快適に結果の検証を行えるようにするためのシステムの開発を行った．

適切なパラメタを発見する手法の提案が本研究では重要と考えるが，本稿ではこの問題を今後の課題と位置づけている．

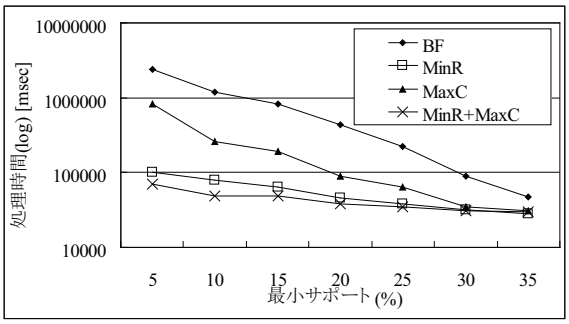


図 4 処理時間の比較 (Zoo10000)
Fig. 4 Runtime Comparison (Zoo10000)

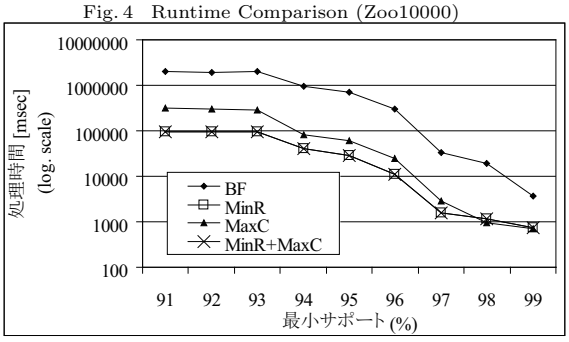


図 5 処理時間の比較 (KDD Cup 99)
Fig. 5 Runtime Comparison (KDD Cup 99)

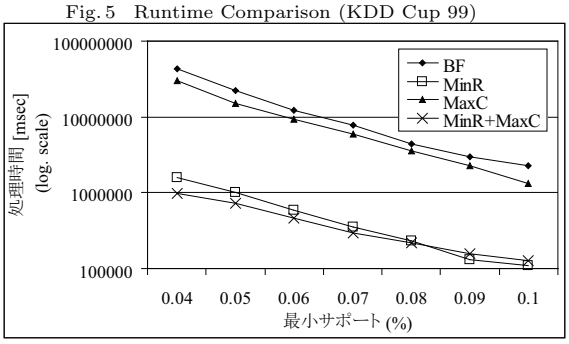


図 6 処理時間の比較 (Synthetic)
Fig. 6 Runtime Comparison (Synthetic)

7. 外れ値トランザクション検出システム

本手法による外れ値トランザクション検出を，ユーザがよりインタラクティブに行い，かつ検出結果の解析を容易にするためのシステムの開発について述べる．図 7 は，我々が開発した外れ値検出システムの構成図である．本システムは大きく二つに分かれている．これまでに説明した検出アルゴリズムを実装した外れ値トランザクション検出器と，本手法をユーザがより使いやすくするための検出結果解析器である．

ユーザが実際に用いるシステムの実行画面が図 8 に示されている．図 8 のデータ入力部に外れ値を検出したいデータを入力し，パラメタ入力部にパラメタセッ

トを与え，実行 (Run) ボタンを押すことで内部の外れ値トランザクション検出器が外れ値の検出を行う．検出器が検出結果を出力すると，それを用いて検出結果解析器の機能の一つである検出結果出力器が，図 8 のデータ表示部へ入力データを表示し，外れ値と見なされたトランザクションに対応する行をハイライトする．

また，入力データは一般に巨大であり，ただトランザクションをハイライトしただけではユーザは検出された外れ値がデータ全体にどのように分布しているのか把握できない．そこで，本システムでは外れ値分布図生成機能を実装した．図 8 の左下に二つ並んだ矩形の右側部分は，入力データの全体図である．検出が行われると，データ表示部で外れ値としてハイライトされたトランザクションに対応する位置に，同色の線が描かれる．これによって，ユーザは外れ値がデータ中のどの位置にどのように分布しているか，全体的に把握できるようになっている．

精度比較機能は，入力データに対する正解外れ値をユーザがあらかじめ知っているときに，検出精度を測るための機能である．この機能を用いる際，ユーザは入力データのどのトランザクションが正解外れ値であるかを示す正解外れ値情報を追加的に与える．精度比較機能は与えられた正解外れ値情報と検出結果を比べ，図 8 の精度表示部に Detection Rete と Detection Precision を出力する．同時に，正解トランザクションは，データ表示部の最左列をハイライトする．また，図 8 左下に二つ並んだ矩形の左側部分に，その分布の様子を描画する．

非充足ルール表示機能は，検出結果から各トランザクションの非充足相関ルールの情報を受け取り，データ表示部にルールを出力する仕様となっている．

この他の機能として，処理高速化のための二つの工夫のうち，どの組み合わせを用いたアルゴリズムを選択するのが最も処理が高速であるかを予測する手助けをするため，データから得られる頻出アイテム集合の数や，最小ルール集合の集合濃度，図**??**のような外れ値度の上限の分布の様子を出力する機能の追加も考えているが，時間の都合上，本稿では未実装である．

8. おわりに

本稿では，提案手法による外れ値トランザクション検出を，ユーザがよりインタラクティブに行い，かつ検出結果の解析を容易にするためのシステムの開発について述べた．まず提案手法について説明するため，外れ値度の式の定義を述べ，二つの検出処理高速化のための工夫を取り入れたアルゴリズムについて言及した．さらに，実データと人工データを用いた実験結果を示し，提案手法の有効性および問題点を明らかにした上で，ユーザが本手法をより快適に使用するためのシステムの開発について述べた．

今後の課題として，最速な検出手法をユーザが予測する手助けをするための機能の実装や，最適なパラメタを発見する方法の検討が挙げられる．

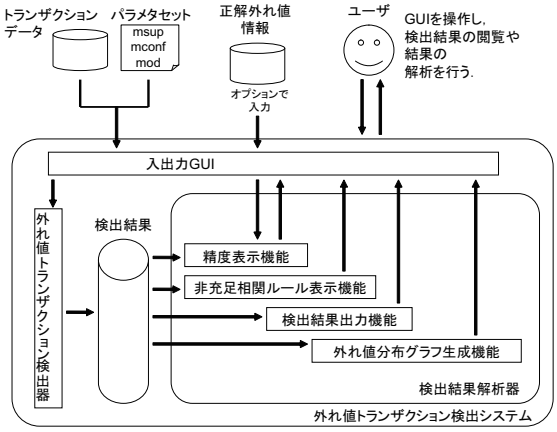


図 7 外れ値トランザクション検出システム
Fig. 7 System for Detecting Outlier Transactions

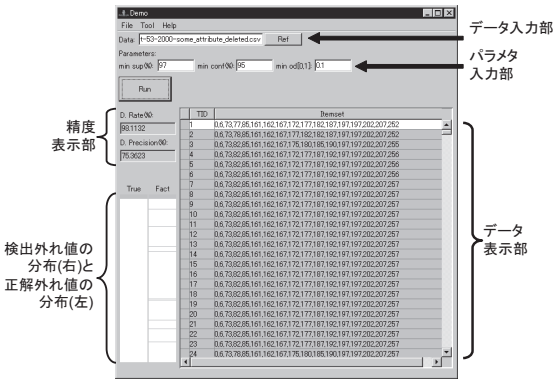


図 8 入出力 GUI
Fig. 8 GUI for Input/Output

謝辞 本システムを開発するにあたりご指導頂いた，筑波大学大学院システム情報工学研究科の北川博之教授に感謝いたします．本研究の一部は魅力ある大学院教育イニシアティブ「実践 IT 力を備えた高度情報学人材育成プログラム」による．

参 考 文 献

1) E. M. Knorr and R. T. Ng, “Algorithms for Mining Distance-Based Outliers in Large Datasets,” VLDB, 1998, pp.392-403.
2) S. Ramaswamy, R. Rastogi and K. Shim, “Ef-

ficient Algorithms for Mining Outliers from Large Data Sets," SIGMOD Conference, 2000, pp.427-438.
3) C. C. Aggarwal and P. S. Yu, "Outlier Detection for High Dimensional Data," SIGMOD Conference, 2001, pp.37-46.
4) A. Arning, R. Agrawal and P. Raghavan, "A Linear Method for Deviation Detection in Large Databases," KDD, 1996, pp.164-169.
5) M. M. Breunig, H. P. Kriegel, R. T. Ng and J. Sander, "LOF: Identifying Density-Based Local Outliers," SIGMOD Conference, 2000, pp.93-104.
6) H. V. Jagadish, N. Koudas and S. Muthukrishnan, "Mining Deviants in a Time Series Database," VLDB, 1999, pp.102-113.
7) E. M. Knorr and R. T. Ng, "Finding Intentional Knowledge of Distance-Based Outliers," VLDB, 1999, pp.211-222.
8) S. Papadimitriou, H. Kitagawa, P. B. Gibbons and C. Faloutsos, "LOCI: Fast Outlier Detection Using the Local Correlation Integral," ICDE, 2003, pp.315-.
9) P. J. Rousseeuw and A. M. Leroy, "Robust Regression and Outlier Detection," John Wiley and Sons, 1987.
10) C. Zhu, H. Kitagawa and C. Faloutsos, "Example-Based Robust Outlier Detection in High Dimensional Datasets," ICDM, 2005, pp.829-832.
11) A. Bronstein, J. Das, M. Duro, R. Friedrich, G. Kleyner, M. Mueller, S. Singhal and I. Cohen, "Self-aware services: using Bayesian networks for detectinganomalies in Internet-based services," International Symposium on Integrated Network Management, 2001, pp. 623-638.
12) D. Pelleg, "Scalable and Practical Probability Density Estimators for Scientific Anomaly Detection, " Doctoral Thesis of Carnegie Mellon University, 2004.
13) P. K. Chan, M. V. Mahoney and M. H. Arshad, "A Machine Learning Approach to Anomaly Detection," Technical Report of Florida Institute of Technology, 2003.
14) K. Das and J. G. Schneider, "Detecting anomalous records in categorical datasets," KDD, 2007, pp. 220-229.
15) J. Han, J. Pei and Y. Yin, "Mining Frequent Patterns without Candidate Generation," Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, 2000, pp.1-12.
16) R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules in Large Databases," VLDB, 1994, pp.487-499.
17) G. Grahne and J. Zhu, "Efficiently Using Prefix-trees in Mining Frequent Itemsets," FIMI, 2003.
18) UCI Machine Learning Repository, http://www.ics.uci.edu/∼mlearn/MLRepository.html
19) K. Narita and H. Kitagawa, "Detecting Outliers in Categorical Record Databases Based on Attribute Associations," APWeb, 2008. (to appear)
20) 成田和世，北川博之,「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」，電子情報通信学会技術研究報告 Vol. 107, No. 131, pp. 399-404.