相関の違いに基づく含意的なアイテム集合の組を発見するためのアルゴリズム

An Algorithm for Mining Implicit Itemset Pairs based on Differences of Correlations

谷口剛 [1*]　原口誠 [1]
Tsuyoshi TANIGUCHI[1] and Makoto HARAGUCHI[1]

[1] 北海道大学大学院情報科学研究科コンピュータサイエンス専攻
[1] Division of Computer Science, Hokkaido University

Abstract: Given a transaction database as a global set of transactions and its local database obtained by some conditioning of the global database, we consider pairs of itemsets whose degrees of correlation are higher in the local database than in the global one. A problem of finding paired itemsets with high correlation in one database is already known as Discovery of Correlation, and has been studied as the highly correlated itemsets are characteristic in the database. However, even noncharacteristic paired itemsets are also meaningful provided the degree of correlation increases significantly in the local database compared with the global one. They can be implicit and hidden evidences showing that something particular to the local database occurs, even though they were not previously realized to be characteristic. From this viewpoint, we have proposed measurement of the significance of paired itemsets by the difference of two correlations before and after the conditioning of the global database, and have defined a notion of DC pairs, whose degrees of difference of correlation are high. Since the measurement of DC pairs is nonmonotonic, DC pair mining problem is difficult. For our difficult problem, we have presented some algorithm for mining DC pairs. The algorithm can efficiently find DC pair to some degree, however we have to improve the algorithm in order to tackle more complicated problem. We discuss some method for an improvement of our system.

1　はじめに

大規模なトランザクションデータベースを対象としたデータマイニングの研究においては，相関ルール [1] や相関しているアイテム集合 (の組) [4, 7]，あるいは Emerging Patterns [5] のように，与えられたデータベースあるいはいくつかのデータベースのうちのいずれかのデータベースにおいて，特徴的であるアイテム集合あるいはアイテム集合の組に注目することが多かった．

上記の意味で特徴的なアイテム集合の組は，例えば一般的な関係をとらえる際には有用である．しかし一方で，ユーザはそのような関係を当たり前であると考えるかもしれない．ここで，チャンス発見の研究 [8] のように，上記の意味で特徴的でないアイテム集合もある条件の下では潜在的に重要である．

例えば，ある都市に出店を考えているスーパーマーケットの店長がいるとする．その店長は，まずは出店しようとしている都市の規模，地域などから定番商品のような一般的な情報を知りたいだろう．このような場合，特徴的なアイテム集合は役に立つ．一方，無事に出店を完了し店が軌道に乗りはじめたとき，その店長は一般的な情報だけでユーザのニーズに応えることはできない．つまり，その店長はユーザのニーズの変化に敏感に反応しなければならない．しかし，ニーズはその変わりはじめに特徴的であるとは限らない．むしろ多くの場合，特徴的なアイテム集合としては識別されないだろう．

そのような特徴的でないアイテム集合を潜在的な特徴としてとらえるために，与えられたデータベースに対して，ある条件付けによるローカルデータベースにおける相関の違いに注目する．もしローカルデータベースに条件付けた結果としてあるアイテム集合の組が非

*連絡先：北海道大学大学院情報科学研究科
〒 060-0814 北海道札幌市北区北 14 条西 9 丁目
tsuyoshi@kb.ist.hokudai.ac.jp

常に高い相関を示すなら，従来の手法でそのアイテム集合の組を発見することができる．しかし，例え非常に大きな相関変化が観察されたとしても，その相関が高い値を示すとは限らない．我々はそのようなアイテム集合の組を隠れた相関として注目する．そのようなアイテム集合の組は潜在的に重要な情報として注目する価値がある．

したがって本研究では，与えられたデータベースに対してそのローカルデータベースにおいて相関が非常に高くなるアイテム集合の組を発見したい．そのようなアイテム集合のことを DC pair と呼ぶ．我々は既に DC pair の考え方を提案し [10]，そのアルゴリズムを改良し [11]，時系列データに対する可能性を示している [12]．ここで，DC pair を発見する問題は，非常に難しい問題である．なぜならば，相関の違いの評価関数は非単調に変化するため，単調性に基づく枝刈りの実現が難しいからである．出現確率が非常に低いアイテム集合の組でさえも顕著な相関変化を示しうると言い換えてもよい．この問題に対し，我々は既にある程度の成果を示したが，より大規模で，より複雑な問題，例えば顕著な相関変化を引き起こす条件さえも見つけるような問題に対処するためにはアルゴリズムの更なる改良が必要である．

そこで本稿では，本研究のシステムがさらに複雑な問題に対処できる可能性を示すために，いくつかの計算効率化の手法を導入する．実験の章において，それらの手法が探索において有効に働いていることを示す．

以下において，本稿の構成を述べる．2 章において，関連研究との比較を行う．3 章において本稿における議論のための準備を行う．4 章において DC pair の考え方を導入し，DC pair 探索問題の定義を行う．5 章において DC pair を探すためのアルゴリズムについて述べる．6 章において実験結果を示す．最終章において，本稿についてまとめ，今後の課題について議論する．

2 関連研究との比較

データマイニングの研究領域には大規模なデータベースから重要な性質やパターンを抽出するために，2 つ以上のデータベースを比較するいくつかの研究がある．*contrast-set mining* [2, 5, 14] として知られるこのデータマイニング技術は，比較しているデータベース間の違いを識別するために用いられる．

例えば，Emerging Patterns [5] は，2 つのデータベースが与えられたときに，一方のデータベースにおいて，もう一方のデータベースよりも，支持度が非常に高いアイテム集合である．そのようなアイテム集合は 2 つのデータベースのうち片方のデータベースをを識別するようなアイテム集合になりえる．また同様の考え方が，χ^2 検定に基づき，あるデータベースにおける特徴的なアイテム集合を得るためのシステムである STUCCO [2] においても用いられている．さらに，Magnum Opus [14] では，与えられたいくつかのデータベースのうちのあるデータベースと相関しているアイテム集合を見つけることにより，上記と同様のことが実現できることを示している．一方，本研究で見つけたいものはあるデータベースにおいて相関が劇的に高くなるアイテム集合の組である．このように，本研究で行っていることは，”*contrast-set mining of correlations between itemsets*” ということができる．

次に，あるデータベースにおいて特徴的な相関を見つけるために多くの方法が提案されている [4, 7]．これらの研究では，アイテム集合間の相関を評価するためにそれぞれの尺度を用いて，与えられたデータベース，あるいは与えられたいくつかのデータベースのうちの 1 つのデータベースにおいて高く相関しているアイテム集合を求めている．このように，これらの研究は，あるデータベースにおいて特徴的であるアイテム集合あるいはアイテム集合の組を見つけるために用いられる．一方，本研究ではあるデータベースにおける相関は高くないが，それを含む全体のデータベースよりは相関が非常に高いアイテム集合の組に特に注目している．本研究で見つけるアイテム集合の組は，特徴的なアイテム集合の組であるかもしれないが，そのようなアイテム集合の組だけであるというわけではない．

3 準備

$I = \{i_1, i_2, \ldots, i_m\}$ を**アイテム**の集合とする．I の部分集合 $X \subseteq I$ を**アイテム集合**という．**トランザクションデータベース**$\mathcal{D}$ はトランザクションの集合とする．ここで，トランザクションはそれぞれがユニークなアイテム集合である．もし $X \subseteq t$ ならば，トランザクション t はアイテム集合 X を**含む**という．トランザクションデータベース $\mathcal{D}$ とアイテム集合 X に対して，$\mathcal{D}$ における X を含むトランザクションの集合を，$O(X, \mathcal{D})$ と記述し，$O(X, \mathcal{D}) = \{t | t \in \mathcal{D} \wedge X \subseteq t\}$ と定義する．そして，$\mathcal{D}$ **における** X **の確率**を $P(X)$ と記述し，$P(X) = |O(X, \mathcal{D})|/|\mathcal{D}|$ と定義する．

アイテム集合 C に対し，C **に関する** $\mathcal{D}$ **のサブデータベース**は $\mathcal{D}_C$ と記述し，$\mathcal{D}$ において C を含むトランザクションの集合，つまり $\mathcal{D}_C = O(C, \mathcal{D})$ と定義する．

アイテム集合 X と Y に対し，トランザクションデータベース $\mathcal{D}$ における X **と** Y **の相関**$correl(X, Y)$ を，$correl(X, Y) = P(X \cup Y)/P(X)P(Y)$ と定義する．サブデータベース $\mathcal{D}_C$ に対して，$\mathcal{D}_C$ **における** X **と** Y **の相関**を $correl_C(X, Y)$ と記述し，$correl_C(X, Y) = P(X \cup Y|C)/P(X|C)P(Y|C)$ と定義する．ここで相関は $\mathcal{D}$ と $\mathcal{D}_C$ において確率が 0 でないアイテム集合に対してのみ定義されることに注意する．本研究において，$correl(X, Y) > 1$ を満たす X と Y の組を特徴的であると考える．なぜならば $P(X|Y) > P(X)$ であるから

である．ここで，$P(Y|X) > P(Y)$ も同様に成り立つことに注意する．同様の理由で，$correl(X,Y) \leq 1$ が成立するような X と Y の組を特徴的ではないと考える．

4 DC pair 探索問題

この節では，DC pair と DC pair を探索する問題について定義する．

アイテム集合 X と Y の組に対して，本研究では，”サブデータベースに条件付けることによって観測される相関の違い”に注目する．相関の違いを以下の比率によって評価する．ここで，C はユーザによって与えられる条件を表すアイテム集合とする．

$$change(X,Y;C) = \frac{correl_C(X,Y)}{correl(X,Y)} = \frac{P(C)P(C|X \cup Y)}{P(C|X)P(C|Y)}.$$

$\rho(>1)$ を相関の違いを表すパラメータとし，アイテム集合 X と Y の組に対して，$change(X,Y;C) \geq \rho$ を満たすような関係を重要であると考える．C はユーザによって与えられると仮定していたので，$P(C)$ を定数と考えることができる．したがって，実際には相関の違いを以下の関数 g によって評価する．

$$g(X,Y;C) = \frac{P(C|X \cup Y)}{P(C|X)P(C|Y)}.$$

$g(X,Y;C) \geq \rho/P(C)$ を満たすようなアイテム集合 X と Y の組を DC pair と呼ぶ．DC pair の探索において大きな制約がないならば，全ての DC pair を見つけることが理想的である．しかし g がアイテム集合 X と Y のいずれかのアイテム数の増加に関して非単調に変化するために，apriori [1] のような単調性に基づく枝刈りを行うことができない．それゆえ，以下のような素朴な考え方によって，求める DC pair を限定する．

> $P(C|X), P(C|Y)$ を低く保ちながら，$P(C|X \cup Y)$ が高くなるような X, Y の組を見つける．

新たなパラメータ $\zeta(0 \leq \zeta \leq 1)$ を用いて，本研究で扱う問題を以下のように定義する．

定義 1. DC pair 探索問題
C を条件付けのためのアイテム集合とする．ρ と ζ が与えられたとき, **DC pair 探索問題**は $P(C|X \cup Y) > \zeta$, $P(C|X) < \epsilon$，$P(C|Y) < \epsilon$ を満たすような全ての X と Y の組を見つけることである．ここで $\epsilon = \sqrt{\zeta \cdot P(C)/\rho}$ とする．
以下の議論において，$X \cup Y$ を DC pair の**組み合わせアイテム集合**，X，Y を DC pair の**要素アイテム集合**と呼ぶ．

5 アルゴリズム

この節では，DC pair を探索するためのアルゴリズムについて議論する．まずはじめに，DC pair 探索するための基本的な方針について説明する．次に，要素アイテム集合識別フェーズにおける枝刈り規則を紹介し，枝刈り効率の改善について検討する．最後に，DC pair 導出フェーズにおける性質を示す．

5.1 DC pair 探索の基本的な方法

まずはじめに，DC pair を探索するための基本的な方法について説明する．DC pair を探索する方法としては主に 2 つの方法が考えられる．1 つは組み合わせアイテム集合から探索する方法，もう 1 つは要素アイテム集合から探索する方法である．本研究では，前者の方法を試みた [10] 上で，探索上の様々な困難を明らかにし，現在は後者の方法に基づき探索を行っている [11, 12]．つまり，最初に要素アイテム集合の候補を見つけ，それらを利用して DC pair を導出する．したがって，DC pair 探索問題は以下の 2 つの段階に分けられる．

要素アイテム集合識別フェーズ
$P(C|X) < \epsilon$ を満たすアイテム集合 X は要素アイテム集合の候補として識別される．

DC pair 導出フェーズ
要素アイテム集合識別フェーズで得られた候補は $P(C|X \cup Y) > \zeta$ を満たすかどうか調べられる．

5.2 要素アイテム集合識別フェーズにおける枝刈り規則と終了条件

要素アイテム集合識別フェーズにおいて，$P(C|X)$ も g と同様，X のアイテム数の増加に関して非単調に変化する．しかし，我々は要素アイテム集合の探索において，ϵ に依存した単調性が存在することを示した．ここで要素アイテム集合識別フェーズにおいては，ボトムアップに要素アイテム集合を探索する問題を考える．以下の枝刈り規則の証明については，[11] を参照して欲しい．

枝刈り規則
探索ノード (アイテム集合)X とその上位集合 $Z(X \subseteq Z' \subseteq Z)$ に対して，もし $P(C \cup Z) \geq \epsilon \cdot P(X)$ が成り立つならば，探索において Z' は X の候補ノードにはならない．

要素アイテム集合の候補のボトムアップ探索において探索している X が上記の枝刈り規則に適用されるとき，Z' は調べる必要がない．したがって，この枝刈り規則を適用するためにサブデータベース $\mathcal{D}_C$ においてボトムアップにアイテム集合 X を調べていく一方，同時に X の上位集合 Z も調べる必要がある．

そのような上位集合 Z を識別するために，本研究では，LCM[13] 等で用いられているバックトラックアルゴリズムと max-miner[3] において極大頻出アイテム集合を識別するために用いられている先読みの考え方を利用している．この考え方を用いるために，アイテム

に対しある辞書順番を仮定し，X を現在調べられているアイテム集合とする．このアイテムの順序付けに従い，$tail(X)$ を X の最大アイテムとし，$T(tail(X))$ を $tail(X)$ よりも大きいアイテムの集合とする．そして，X は $T(tail(X))$ に含まれるアイテム i を加えることにより展開する．以下の条件を満たす時，それ以降の探索を打ち切ってバックトラックする．アルゴリズムの詳細についても [11] を参照されたい．

Phase1 における探索の終了条件
アイテム集合 X と $Z = X \cup T(tail(X))$ に対し，$P(C \cup Z) \geq \epsilon \cdot P(X)$ ならば，X はそれ以上調べなくてよい．

5.3 要素アイテム集合識別フェーズにおける枝刈り効率の改善

前節で説明した枝刈り規則を用いて，我々は既に要素アイテム集合識別フェーズにおける探索効率の向上を実現している．その一方で，枝刈りの効率にはアイテムの辞書順番が大きな影響を与えるにもかかわらず，そのことに関してきちんとした考察，実験を行っていなかった．つまり，要素アイテム集合の探索において探索しているアイテム集合のサイズが小さい段階で枝刈り規則が適用できる場合に，探索効率は高くなる．そして，アイテムをどのようなオーダ，例えば頻度の降順，昇順などにしたがって探索するかによって探索するアイテム集合の順番も大きく異なる．本稿では，どのオーダにしたがって探索するのが本研究の枝刈りにあっているのか確かめるために実験を行った．その実験については，実験の章において議論する．

5.4 DC pair 導出フェーズにおける性質

この節では，DC pair 導出フェーズにおける性質について議論する．DC pair 導出フェーズにおいて，もし要素アイテム集合の数が多ければ，その組み合わせの数は非常に膨大である．しかし，調べるべき組み合わせを識別するために，我々は既に以下の 2 つの性質を利用している．

$O(X, \mathcal{D}_\mathcal{C}) = \{t | t \in \mathcal{D}_\mathcal{C} \wedge X \subseteq t\}$，要素アイテム集合の候補 X, Y に対して

1. X に対し，$X \cap Y \neq \emptyset$ なら，Y は調べる必要はない．
2. $t \in O(X, \mathcal{D}_\mathcal{C})$ に対し，$Y \subseteq t$ なる t が存在しないならば，Y はそれ以上調べる必要はない．

ここでは，2 番目の性質についてのみ簡単に説明する．もし要素アイテム集合の候補 X, Y が DC pair であるならば，必ず部分データベースにおけるいずれかのトランザクションに両方のアイテム集合が含まれる．したがって，もし部分データベースを調べた時点でそのようなトランザクションが存在しないならば，もうそれ以上調べる必要はない．部分データベースにおいて要素アイテム集合の候補を含むトランザクションはそれほど多くないことが期待でき，そのようなトランザクションを調べるだけで探索を終了できる場合がある．

ここで，本稿において新たに追加する計算の工夫と性質を以下に示す．

1. 要素アイテム集合識別フェーズで得られる情報の利用．
2. J を $O(X, \mathcal{D}_\mathcal{C})$ に出現するアイテムの集合とする．もし Y が J の部分集合でないならば，Y は調べる必要はない．

まずは，1 番目の工夫について説明する．要素アイテム集合識別フェーズにおいて，要素アイテム集合の候補を識別する際に，当然それがどのトランザクションに含まれているかの情報を得ることができる．その情報を保存しておくことにより，DC pair 導出フェーズにおける無駄なデータベーススキャンを避けることができる．

次に，2 番目の性質について説明する．もし Y が部分データベースにおいて X と同じトランザクションに含まれるならば，少なくとも X が含まれるトランザクションにアイテムとして現れているはずである．したがって，部分データベースにおいて X が含まれているトランザクションに現れるアイテムを調べておき，もし Y がそのアイテムの集合に含まれないならば，X と同じトランザクションには含まれない．つまり，データベーススキャンすらせずに，Y を調べなくてもよいことがわかる．ある要素の候補に対して，それと関連のあるアイテムがそれほど多くない時に，この性質は有効に働きうると考えられる．

6 実験

この節では，本研究で行った実験における結果を示す．実験の目的は，本稿で議論した改良が DC pair の効率的な探索に寄与するか確かめることと，潜在的に重要な DC pair がデータベースに実際に存在するかを確かめることである．

6.1 データセットと実装

本研究では，*Entree Chicago Recommendation Data* とアメリカの国勢調査のデータ (IPUMS)[9] を用いて実験を行った．

Entree Chicago Recommendation Data は，[11] において我々が行った実験結果との比較を行うために用いた．このデータセットは UCI KDD Archive [6] におけるデータセットの 1 つであり，それぞれが Atlanta や Boston などにおけるレストランの特徴を含む 8 つのデータセットから成る．本研究では，8 つのデータセットをグローバルデータベース $\mathcal{D}$ として 1 つのデータベースに合成した．そして，それぞれの地域 C によ

$\rho = 3.0, \zeta = 0.4$								
region	ϵ	N_{ind}	N_{des}	N_{asc}	N_{cand}	$t_{N_{ind}}(sec)$	$t_{N_{des}}(sec)$	$t_{N_{asc}}(sec)$
Atlanta	0.0922	1.8×10^6	6.0×10^6	6.1×10^5	34949	17.906	50.812	5.547
Boston	0.118	4.5×10^6	1.8×10^7	2.1×10^6	45081	43.985	151.484	18.562
Chicago	0.147	3.1×10^6	5.9×10^6	2.5×10^6	47068	28.735	49.672	20.234
Los Angeles	0.118	1.8×10^6	6.3×10^6	9.6×10^5	11611	17.656	50.625	7.953
New Orleans	0.102	1.4×10^6	2.6×10^6	5.1×10^5	12207	12.735	21.407	4.656
New York	0.196	1.5×10^6	3.8×10^6	9.9×10^5	72490	14.578	32.156	8.797
San Francisco	0.114	2.3×10^6	7.8×10^6	9.3×10^5	53175	21.750	64.484	8.391
Washington DC	0.112	2.9×10^7	1.7×10^8	9.1×10^6	22291	279.500	1369.047	77.828

図 1: インデックス順，頻度の昇順，頻度の降順をオーダとした枝刈りの効果

る条件付けによって，$\mathcal{D}$ におけるローカルデータベース $\mathcal{D}_\mathcal{C}$ を定義する．グローバルデータベースはそれぞれのアイテムがレストランの特徴を表す 265 アイテムの部分集合である 4160 トランザクションから成る．

IPUMS データはアメリカの国勢調査と 2000 年から 2003 年における”American Community Surveys”という調査からの 37 の信頼性の高いサンプルから成る．IPUMS data extraction system を使用することにより，それぞれの研究者は必要なサンプルと変数を選択し，データを抜き出すことができる．我々は時系列の情報を条件とした DC pair を見つけるために，Washington における 1980 年，1990 年，2000 年のサンプルを抽出した．変数は，年齢，性別，人種等様々である．このデータは実際に見つかる DC pair を確認するために用いた．この実験において，グローバルデータベースは 1980 年と 1990 年，1990 年と 2000 年のサンプルを合成したもの，条件はそれぞれ 1990 年と 2000 年として DC pair を導出した．このデータを用いた実験の詳細については，[12] を参照されたい．

本研究のシステムは C 言語で実装され，全ての実験は 1.00 GB RAM，Xeon 3.60 GHz プロセッサのスペックを持つ PC 上で行った．

6.2 要素アイテム集合識別フェーズにおけるアイテムのオーダの枝刈り規則への影響

この節では，要素アイテム集合識別フェーズにおけるアイテムのオーダの枝刈り規則への影響を示す．本実験においては，アイテムのオーダを 1．インデックス順，2．頻度の降順，3．頻度の昇順に設定して探索を試みた．実験結果を図 1 に示す．図 1 において，N_{ind} はアイテムのオーダをインデックス順に基づいて探索した時の探索アイテム集合数である．この結果は，我々が [11] において示した結果でもある．$t_{N_{ind}}$ はその探索における計算時間である．N_{des}，N_{asc} はそれぞれアイテムのオーダを頻度の降順，昇順に設定した時の探索数，$t_{N_{des}}$，$t_{N_{asc}}$ はその探索における計算時間である．N_{cand} は抽出された要素アイテム集合の候補の数を表す．

実験結果より頻度の昇順に基づいた場合，探索数，計算時間ともに我々が既に示した結果に対し，最小で 4 分の 1 程度で探索できることがわかった．この結果に関しては，まだ多くの分析が必要だが，以下のようなことが予想される．高頻度アイテムは様々なアイテム集合に現れるため，結果として要素の候補ではないアイテム集合に含まれることも多い．したがって，深さ優先的に高頻度アイテムが関係するアイテム集合を先に調べてしまうよりは，探索全体を通して枝刈りの対象がある状態で探索を進める方が本研究の探索にはあっている．頻度の降順で探索を行うと著しく探索効率が落ちることからも上記のことが原因であると思われる．

6.3 DC ペアの性質の効果

$Comp_{cand}$ を要素アイテム集合識別フェーズにおいて得られた候補の集合とする．DC pair 導出フェーズにおいて，$Comp_{cand}$ におけるアイテム集合の組が DC pair かどうかを調べる．DC pair 導出フェーズにおける新たな性質の効果を図 2 に示す．$|C|$ は今までの性質を用いた時の $Comp_{cand}$ における探索数である．C から DC pair を探すための計算時間は t_C と表す．$|C_{pro2}|$ は本稿において議論した性質を用いた時の $Comp_{cand}$ における探索数である．C_{pro2} から DC pair を探すための計算時間は $t_{C_{pro2}}$ と表す．また，要素アイテム集合識別フェーズの情報の利用の効果を調べるために，C に対してこの性質のみを利用した計算時間を $t_{C_{pro1}}$ と表す．最後に $|DC|$ は抽出された DC pair の数であり，$|DC_{imp}|$ は DC において相関の度合いが 1 以下である DC pair の数である．

実験結果より，今までの性質のみで探索した結果に対して全ての条件において実行時間が 2 分の 1 程度で導出を終えることができているので，本稿で議論した性質はいずれも DC pair 導出の計算効率に対して寄与していることがわかる．したがって，DC pair 探索フェーズにおける効率的な探索に対する可能性が示された．

6.4 DC pair の例

本稿では，IPUM データを用いて行った実験における結果 [12] の一部を示す．

まず DC pair を導出する前に，1980 年から 2000 年

region	$\|C\|$	$\|C_{pro2}\|$	$\|DC\|$	$\|DC_{imp}\|$	$t_{\|C\|}(sec)$	$t_{\|C_{pro1}\|}(sec)$	$t_{\|C_{pro2}\|}(sec)$
Atlanta	3.0×10^6	2.8×10^6	1.4×10^6	353	132.422	91.031	64.921
Boston	4.5×10^6	3.8×10^6	2.9×10^6	240	223.016	141.422	104.781
Chicago	4.5×10^6	3.7×10^6	3.0×10^6	7	236.282	166.141	116.344
Los Angeles	5.4×10^5	5.0×10^5	2.5×10^5	101	16.375	13.266	8.500
New Orleans	5.3×10^5	4.8×10^5	2.2×10^5	57	20.062	14.703	9.657
New York	9.7×10^6	7.6×10^6	6.8×10^6	44	551.547	313.750	248.985
San Francisco	4.2×10^6	3.6×10^6	2.5×10^6	393	285.953	180.813	134.328
Washington DC	9.1×10^5	7.1×10^5	6.0×10^5	86	59.079	44.484	29.125

図 2: DC pair の性質の効果

まで高い相関を示すものを導出したところ，60 歳以上と未亡人，自営業と家で働いているという関係などある意味当たり前であると考えられるような関係が多数抽出された．

一方，発見した DC pair としては 20 代の農夫と住宅ローンの購買契約を結び家を所有しているという関係の相関が 1980 年と 1990 年を合成したデータベースにおいて 1990 年を条件にしたことにより，相関の値が 1.05 から 3.96 へと大きく変化した．しかし，この 3.96 という値はそれほど高い値とは言えず，結果として先に示した高い相関を示す関係に隠れてしまう．ここで，相関が変化しているので 1990 年においては 20 代農夫のマイホームの購買状況が変わり始めているかもしれない可能性に注目するのは価値がある．われわれはこのような DC pair を見つけたいと考えている．

7 まとめと今後の課題

本稿では，潜在的に重要なアイテム集合の組を発見するための 1 つの考え方として，我々が既に提案している DC pair を発見するためのアルゴリズムの改良について議論した．我々は現在，さらに複雑な問題を扱うための準備を進めており，本稿における実験の結果は，その有効なエビデンスとなりえる．今後は，本稿における実験によって明らかになってきた DC pair の性質を利用し，さらに効率的に DC pair を導出するシステムを目指すことが課題となる．

参考文献

[1] R. Agrawal and R. Srikant, Fast Algorithms for Mining Association Rules in Large Databases, In: the 20th Int'l Conf. on Very Large Data Bases, Morgan Kaufmann, VLDB'94, pp. 487–499, 1994.

[2] S. D. Bay and M. J. Pazzani, Detecting Group Differences: Mining Contrast Sets, Data Mining and Knowledge Discovery, Springer Verlag, vol. 5, no. 3, pp. 213–246, 2001.

[3] R. J. Bayardo Jr., Efficiently Mining Long Patterns from Databases. In: the ACM-SIGMOD Int'l Conf. on Management of Data, ACM Press, pp. 85–93, 1998.

[4] S. Brin, R. Motwani and C. Silverstein, Beyond Market Baskets: Generalizing Association Rules to Correlations. In: the ACM SIGMOD Int'l Conf. on Management of Data, ACM Press, vol. 26, no. 2, pp. 265–276, 1997.

[5] G. Dong and J. Li, Efficient Mining of Emerging Patterns: Discovering Trends and Differences. In: the 5th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining, ACM, pp. 43–52, 1999.

[6] S. Hettich, and S. D. Bay, The UCI KDD Archive, Department of Information and Computer Science, University of California, Irvine, CA, http://kdd.ics.uci.edu, 1999.

[7] S. Morishita and J. Sese, Traversing Itemset Lattices with Statistical Metric Pruning. In: the ACM SIGACT-SIGMOD-SIGART Symposium on Database Systems, ACM, PODS 2000, pp. 226–236, 2000.

[8] Y. Ohsawa and Y. Nara, Understanding Internet Users on Double Helical Model of Chance-Discovery Process. In: the IEEE Int'l Symposium on Intelligent Control, IEEE, pp. 844–849, 2002.

[9] S. Ruggles, M. Sobek, T. Alexander, C. A. Fitch, R. Goeken, P. K. Hall, M. King and C. Ronnander, Integrated Public Use Microdata Series: Version 3.0 [Machine-readable database]. Minneapolis, MN: Minnesota Population Center [producer and distributor], 2004.

[10] T. Taniguchi, M. Haraguchi and Y. Okubo, Discovery of Hidden Correlations in a Local Transaction Database based on Differences of Correlations, In: Machine Learning and Data Mining in Pattern Recognition, Springer Verlag, lnai 3587, MLDM 2005, pp. 537–548, 2005.

[11] T. Taniguchi and M. Haraguchi, An Algorithm for Mining Implicit Itemset Pairs based on Differences of Correlations. In: the 8th Int'l Conf. on Discovery Science, Springer Verlag, lnai 3735, DS 2005, pp. 227–240, 2005.

[12] T. Taniguchi and M. Haraguchi, Discovery of Hidden Correlations in a Local Transaction Database based on Differences of Correlations, In: Engineering Applications of Artificial Intelligence (to appear).

[13] T. Uno, M. Kiyomi and H. Arimura, LCM ver.2: Efficient Mining Algorithms for Frequent/Closed/Maximal Itemsets. In: the IEEE Int'l Conf. on data mining, 2nd Workshop on Frequent Itemset Mining Implementations (FIMI'04), CEUR-WS.org, CEUR Workshop Proceedings, vol. 126, 2004.

[14] G. I. Webb, S. M. Butler and D. A. Newlands, On detecting differences between groups. In: the 9th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining, ACM, pp. 256–65, 2003.