

# 相関ルールマイニングを用いた混乱プロジェクトの特徴分析 ランダムサンプリングデータへの適用

浜野 康裕<sup>†</sup>, 天寄 聡介<sup>‡</sup>, 水野 修<sup>†</sup>, 菊野 亨<sup>†</sup>

<sup>†</sup>大阪大学 情報科学研究科

<sup>‡</sup>鳥取環境大学 情報システム学科

## 概要

ソフトウェアプロジェクト管理にとって、混乱プロジェクトに強い影響を与えるリスク要因を明らかにすることは重要である。これまでの研究では、40プロジェクトのデータに相関ルールマイニング手法を適用して、そうしたリスク要因と相関ルールを得ている。ところが、利用した40プロジェクトへの過度の依存と類似した相関ルールの存在が検討課題として残されていた。本研究では課題を解決するために二つの試みに挑戦する。まず、40プロジェクトから36プロジェクトをランダムに選び、相関ルールマイニング手法を繰り返し適用する。次に、求まる相関ルールに対して、前提の重なりと信頼度を考慮した絞り込みを適用する。評価実験の結果、検討課題がいずれも肯定的に解決出来ていることが確認できた。

## Characterization of Relationship Between Runaway Projects and Risk Factors Using Association Rules Mining

Yasuhiro Hamano<sup>†</sup>, Sousuke Amasaki<sup>‡</sup>, Osamu Mizuno<sup>†</sup>, Tohru Kikuno<sup>†</sup>

<sup>†</sup>Graduate School of Information Science and Technology, Osaka University.

<sup>‡</sup>Dept. of Information Systems, Tottori University of Environmental Studies.

### Abstract

For software project management, it is very important to identify risk factors which make project into runaway. In the previous study, we got risk factors and association rules by applying association rules mining to 40 project data. But there remain two basic questions: strong dependencies to 40 project data and existence of similar association rules. This paper tries to answer these questions and presents two new ideas. The first is to select randomly 36 project data from 40 project data and apply association rules mining repeatedly. The second is to delete some association rules based on reliability and redundancies in their antecedents. As the result of experimental evaluation, it was shown that strong dependencies are not observed in the resultant rules and such similar association rules are successfully deleted.

## 1 まえがき

ソフトウェア開発サイクルが短くなる一方で、予算や品質に関する要求は厳しさを増しており、開発現場を取り巻く環境も厳しいものとなっている。そのような条件下で、ソフトウェア開発プロジェクトを成功に導くことは非常に重要な課題となっている。

そこで近年、様々なメトリクスが注目され、それ

をプロジェクト管理に利用する取り組みが活発に展開されてきている [1-3]。一方、メトリクスを利用してデータが実際に収集されているが、多様なソフトウェア開発環境が存在する中で、ある環境に適合した重要なメトリクスを発見することが強く求められてきている。

我々は、ある企業のソフトウェア開発プロジェク

トを対象にして、プロジェクトの混乱状況を引き起こすリスク要因を特定する研究に取り組んできている [10, 12]. これまでにアンケートデータに対して相関ルールマイニング手法を適用することで、プロジェクトの混乱とリスク要因との間の関係を提示することに成功した [12]. しかし、得られたリスク要因が利用しているデータに強く依存しすぎているのではないかというデータ依存の問題と、得られた相関ルールに極めて類似したものが含まれているという冗長性の問題があった.

本研究では課題を解決するために二つの試みに挑戦する. まず、40 プロジェクトから 36 プロジェクトをランダムに選び、相関ルールマイニング手法を繰り返し適用する. 次に、求まる相関ルールに対して、前提の重なりと信頼度を考慮した絞り込みを適用する. 評価実験の結果、検討課題がいずれも肯定的に解決出来ていることが確認できた.

## 2 相関ルールマイニングによる特徴分析

従来研究 [12] では、ソフトウェア開発プロジェクト現場からアンケート (図 1) により収集されたアンケートデータ (表 1) に対して、相関ルールマイニング手法 [4] を適用した. ここではその概要について述べる.

### 2.1 アンケート表

アンケート表の設計に当たっては、リスク管理に関する専門書や論文 [1-3, 5-8, 11] と協力企業における内部規約を調査した. その結果、混乱プロジェクトを引き起こすリスク要因を次の 5 つの主要な問題点に整理した [10].

1. 要求仕様の定義と理解に関する問題点
2. 実現すべきプロダクトの規模や機能の見積りに関する問題点

問題分析アンケート	
1.	要求仕様の定義と理解に関する問題点
1.1	ソフトによる実現を要求する側が、何を実現したいかわかっていなかった。
1.2	要求側の説明力不足。
1.3	要求側の理解力不足。
1.4	開発側の理解内容・積極さ、要求側に説明・確認不足。
2.	開発すべきソフトウェアの規模や機能の見積りに関する問題点
2.1	見積りの大抵よとの把握不足。(実際に訂正してしまった)
2.2	過去の成功パターンで安易に安堵した(見積もり後部が不明確であった)
2.3	例外発覚などの見積項目抜け(見えていない範囲しか見積もっていなかった)
2.4	見積り精度を過小評価してしまった
2.5	取組の責任に安堵してしまった。
3.	開発計画の作成方法とその内容に関する問題点
3.1	マネージャによる実進性の把握不足
3.2	作業分解計画 (WBS) およびプロジェクト組織構成の明確化不十分
3.3	各作業分解の工程前後要物の定義が不十分
3.4	マイルストーン、チェックポイント、レビュー時期の設定がなかった
3.5	前デマの引きずり、剥込み等で予定工程分が確保できず
3.6	計画に対する関係者全員のコミットメントがなかった。
4.	開発チームの編成と人材(能力)に関する問題点
4.1	スキル不足
4.2	開発スキルを身振せず、その時に当てられる人員しか確保しなかった。
4.3	人材力不足
5.	技術的な事項や外的事項に対するプロジェクト管理に関する問題点
5.1	進捗状況が把握できていなかった
5.2	進捗管理方法不明瞭
5.3	管理のための最小限のデータ(進捗、工数など)収集を行っていなかった。

各項目に対し、極めて同意する(3)、同意する(2)、同意しない(1)、分からない(-)を記入して下さい。

図 1: 混乱予測アンケート

3. 開発計画の作成方法とその内容に関する問題点
4. 開発チームの編成と人材(能力)に関する問題点
5. 技術的な事項や外的事項に対するプロジェクト管理に関する問題点

これらの 5 つの主要な問題点はそれぞれより詳細なレベルの調査項目 (リスク要因) に展開されている (図 1 参照). なお、これらの調査項目は開発終了時点で全て記入可能であることを目安にして作成されている. 回答者は各調査項目に対して 4 通りの返答が許されている. それらは、「極めて同意する (3)」、「同意する (2)」、「同意しない (1)」、「分からない (-)」である.

### 2.2 対象プロジェクト

あるソフトウェア開発企業の協力を得て、1996 年から 1998 年までに実施された 40 プロジェクトを対象に分析を行った. 表 1 に示すデータは 1999 年にマネージャから得られたアンケートデータである [10].

表 1: 回収されたアンケートデータ (40 プロジェクト)

Projects in '98-'99	1: 要求仕様					2: 見積り					3: 計画					4: 体制					5: 信頼					混乱判定
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	
d1	1	1	1	1	1	2	3	3	2	1	2	1	1	1	1	1	2	-	1	1	1	1	1	1	成功	
d2	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	成功	
d3	1	1	1	1	3	1	1	2	3	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	成功	
d4	3	3	2	2	3	1	1	2	2	1	2	2	1	1	1	1	1	1	1	1	1	1	1	1	成功	
d5	1	1	1	1	2	1	1	1	1	1	1	2	1	2	2	1	1	1	1	1	1	1	1	1	成功	
d6	1	3	2	1	1	2	2	2	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	成功	
d7	1	1	2	3	2	1	1	1	1	1	1	2	1	3	1	1	1	1	1	1	1	1	1	1	成功	
d8	1	2	3	3	1	1	1	2	1	1	1	2	2	1	1	2	2	1	1	1	1	1	1	1	成功	
d9	1	2	1	2	3	1	1	1	1	1	1	2	2	1	2	1	1	1	1	1	1	1	1	1	成功	
d10	1	1	1	1	2	1	2	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	成功	
d11	1	3	3	2	1	1	1	3	3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	成功	
d12	1	2	2	2	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	成功	
d13	1	2	1	2	1	1	1	1	1	1	1	2	3	3	1	2	2	2	2	2	2	2	2	2	成功	
d14	1	1	1	1	3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	成功	
d15	1	2	2	2	2	1	2	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	成功	
d16	1	1	1	1	2	1	2	1	1	1	2	2	3	2	2	1	1	1	1	1	1	1	1	1	成功	
d17	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	成功	
d18	1	1	1	1	3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	成功	
d19	1	2	3	2	3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	成功	
d20	1	2	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	成功	
d21	1	2	2	2	1	1	2	3	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	成功	
d22	1	2	3	3	2	2	3	2	3	2	3	3	2	3	2	3	2	3	2	3	2	3	2	3	混乱	
d23	2	2	1	2	3	1	1	2	3	1	2	1	2	3	1	1	1	1	1	1	1	1	1	1	混乱	
d24	2	2	3	3	3	2	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	混乱	
d25	3	2	1	1	3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	混乱	
d26	1	2	3	2	2	3	1	2	2	2	1	2	2	2	1	2	2	2	1	2	2	2	1	2	混乱	
d27	1	2	2	2	2	1	3	2	3	3	1	2	2	1	1	2	2	2	1	1	1	1	1	1	混乱	
d28	2	3	3	2	2	1	1	3	3	2	1	3	3	1	2	3	2	1	2	2	1	2	2	1	混乱	
d29	3	2	3	2	1	1	2	3	1	1	1	2	2	2	3	1	2	1	2	1	3	3	3	3	混乱	
d30	3	2	3	2	1	1	2	2	2	1	1	2	2	2	2	1	1	1	1	1	1	1	1	1	混乱	
d31	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	混乱	
d32	2	3	3	3	2	2	3	3	3	3	3	3	2	3	3	3	3	3	3	3	3	3	3	3	1	混乱
d33	2	1	1	1	1	1	2	1	2	1	1	1	3	2	1	1	1	1	1	1	1	1	1	1	1	成功
d34	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	成功
d35	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	成功
d36	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	成功
d37	3	1	1	2	2	1	2	2	3	1	2	2	1	1	1	2	3	1	1	3	3	1	3	1	1	混乱
d38	2	3	3	3	2	-	3	-	-	3	3	3	1	2	2	-	3	1	1	1	1	1	1	1	1	混乱
d39	2	2	2	2	3	2	1	3	1	1	1	3	3	3	3	3	1	1	1	1	1	1	1	1	2	混乱

## 2.3 相関ルールマイニング手法

相関ルールマイニングは、相関ルール (以下ルールとする) と呼ばれる事象間の強い関係を知識として発見する分析手法である。

データ集合全体を分析して (つまり、データのマイニングを実施して)、「ある事象 A が発生するならば別の事象 B も発生する」という事実を発見し、それをルールとして抽出する。このとき、抽出されたルールを  $A \implies B$  と表記し、A を前提、B を結論と呼ぶ。

このルールの重要度を評価するパラメータとして、「信頼度 (confidence)」と「支持度 (support)」の 2 つがある。まず、信頼度とは事象 A が発生した場合に事象 B も同時に発生する確率 ( $p(B | A)$ ) を表す。つまり、この値が 1 に近づくほど、ルール的前提と結論の結び付きが強いことを意味する。また、支持度

とはルールの出現頻度を表すもので、データ集合全体の中で A と B が同時に発生する確率 ( $p(A \wedge B)$ ) である。

実際のルール抽出では最低信頼度と最低支持度を設定し、その条件を満たすルールだけを抽出する。

### 2.3.1 アンケートデータの 2 値化

相関ルールマイニングでは順序尺度は扱えず、名義的にしか扱えない。つまりアンケート回答の「極めて同意する (3)」と「同意する (2)」のリスクの大きさの違いを考慮した分析が出来ない。そこでデータの前処理としてリスク要因に対するアンケート表の回答の結果の「極めて同意する (3)」と「同意する (2)」を”リスクが高い”として一つのカテゴリに分類する。

具体的には、回答結果が「極めて同意する (3)」と「同意する (2)」であったものをリスクが高い (記号 H で表す) とする。一方、「同意しない (1)」ものをリスクが低い (記号 L で表す) とした。また「分からない (-)」ものは欠損値として扱う。

混乱判定については、2 値の変数  $R = \{Runaway, Success\}$  (混乱が Runaway, 成功が Success に対応する) として扱う。

## 2.4 相関ルールマイニングの適用

2 値化済みのデータセットに対して相関ルールマイニングを適用することで、ルールを得ることが出来る。引き続き、混乱プロジェクトとリスク要因の関係を表すルールだけに絞り込む操作を行う。

具体的には、前提にあるリスク要因についてリスクが高い (H) という回答が存在し、かつ、結論が  $R = Runaway$  となっているルールだけに絞り込む。以上の操作により得られたルールが表 2 に示す 16 個のルールである。

## 2.5 検討課題

従来研究には次の二つの課題が懸念され、確認する必要があった。

### 1. プロジェクトデータへの過度の依存性

得られているリスク要因とルールが、利用した40プロジェクトのデータセットに過度に依存している恐れ。つまり、40個のデータセットが一部でも欠落すると、得られる結果が大きく変わるのではないか。

### 2. 類似したルールの存在

得られているルールにはその前提が極めて類似したものがいくつか残っているが、それらが必要なものであるかという疑問。つまり、前提に重なりのあるルールは削除しても予測精度に影響がないのではないか。

## 3 提案する分析法

### 3.1 本研究のアプローチ

本研究では検討課題1の解決のため、40個の1組のデータセットに対して複数回のランダムサンプリングを行うことで、複数組の（実際には914組の）データセットを用意する。その結果、複数のデータセットに対して相関ルールマイニングを適用し、頻繁に現れるルールのみを有効であると判断する。それによりデータセットに過度に依存しないリスク要因とルールを取得できる。

また、検討課題2の解決のため、前提に重なりのある類似したルール間で比較を行い、冗長と考えるルールを除去することで、より重要なリスク要因への絞り込みの見通しを明確にする。

こうして求まる結果の評価としては、従来研究で得られているルール、リスク要因そのものの比較と、ルールの自己適用による混乱予測結果の精度比較の2つを行う。

分析法の大まかな流れを図2に示す。次節で詳細なフェーズについて説明する。

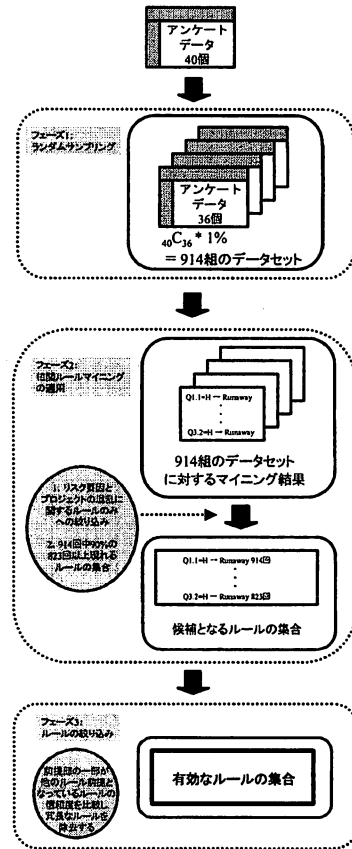


図 2: 分析の流れ

### 3.2 分析法の詳細説明

#### フェーズ1（ランダムサンプリング）

40個のプロジェクトデータから90%のデータをランダムサンプリングする。40個のデータから36個を選ぶ場合の数  ${}_{40}C_{36}=91470$  通りのデータセットを作成し、その1%に相当する914組のデータセットに対してマイニングを行う。どのデータセットを用いるかはランダムに決定する。

#### フェーズ2（相関ルールマイニングの適用）

フェーズ1で得られた914組のデータセットのそれぞれに対して、相関ルールマイニング手法を適用する。今回は、最低信頼度0.50、最低支持度0.20としてルール抽出を行う。

次に、それぞれのルールの集合から、従来手法と同様の方法で、プロジェクトの混乱とリスク要因に関するルールのみを抽出する。

最後に、抽出したルールの中から有効なルールだけを選び出す作業を実施する。各ルールは最低で1回、最大で914回現れる。今回は、914回の90%である823回以上の回数現れたルールのみを有効なルールとする。また、各ルールの信頼度と支持度は平均値を採用する。

### フェーズ3 (ルールの絞り込み)

文献[9]と同じ冗長なルールの削減手法を採用して、さらにルールの絞り込みを行う。あるルール的前提の一部が、別のルール的前提となっている場合に注目する。

例えば、ルール  $R1 : X \wedge Y \implies Z$  (信頼度  $c$ ,  $X \neq \emptyset$ ) と  $R2 : Y \implies Z$  (信頼度  $c'$ ) について考える。絞り込みの判定基準には、 $c/c'$  の値  $\gamma$  を用いる。 $\gamma$  がある閾値以下ならば、 $R1$  は冗長であると判断し削除する。すなわち、 $X$  という新たな要素が  $R2$  の前提に追加されることによる影響が、 $R2$  から  $R1$  への信頼度の増加の形として、明確に表れていなければ、 $R1$  を削除する。絞り込みの順序は次の通りとする。最初にすべてのルール間で前提の比較を行い、上記の関係にあるルールについて  $\gamma$  の値を計算する。そして、 $\gamma$  が閾値以下のすべてのルールを削除する。もしあるルール的前提の一部が他の複数のルールの前提となっている場合は、 $\gamma$  の値が複数得られるが、その中の最小値を用いる。

## 4 提案分析法の適用

### 4.1 適用結果

提案分析法を表1の40個のアンケートデータに適用した。今回の分析では、フェーズ3での閾値は1.2としている。

最終的に得られた相関ルールを表3に示す。従来研究で求まっていた表2と比較すると、ルールのが16から12へと、25%も減っていることが分かる。その有効性の詳細な分析は次節で行う。

### 4.2 比較評価

#### 4.2.1 リスク要因

信頼度を1.0から0.1ずつ区切った場合、各区間のルールに含まれるリスク要因の状況を表4に挙げた。複数のルールに含まれているリスク要因に関しては、最も高い区間を代表として挙げている。この表はリスク要因間の重要度の関係を表すと考えられる。

表4より、今回の分析結果では  $Q1.4$  は信頼度が低いルールで現れたことが分かる。従来研究では信頼度の高いルールに  $Q1.4$  が含まれていたが、今回はそのルールがフェーズ3で除去されている。

今回新たに得られたリスク要因は  $Q3.6$  と  $Q5.2$  である。この二つのリスク要因に関しても信頼度が低いルールにしか現れていないので、従来研究で得られていたリスク要因に準じると考えられる。

#### 4.2.2 ルールの自己適用

次に、今回ルール抽出に用いたアンケートデータに対して、ルールを自己適用して、混乱予測を試みた。具体的には、前提がルールと一致した場合にプロジェクトを混乱であると予測している。

信頼度が一定値以上のルールだけを用いて予測したときの結果を表5に示す。例えば、表5の0.70以上のデータをみると、本研究と従来研究で若干の差が生じていることが観測される。

表 2: 従来研究で得られていたルール

前提	結論	信頼度
$Q1.4 = H \wedge Q2.3 = H \wedge Q3.3 = H$	$R = \text{Runaway}$	0.91
$Q2.3 = H \wedge Q3.3 = H$	$R = \text{Runaway}$	0.91
$Q1.4 = H \wedge Q1.5 = H \wedge Q2.3 = H$	$R = \text{Runaway}$	0.77
$Q1.4 = H \wedge Q3.3 = H$	$R = \text{Runaway}$	0.77
$Q1.1 = H$	$R = \text{Runaway}$	0.71
$Q1.4 = H \wedge Q2.3 = H$	$R = \text{Runaway}$	0.69
$Q1.2 = H \wedge Q1.4 = H \wedge Q2.3 = H$	$R = \text{Runaway}$	0.67
$Q1.5 = H \wedge Q2.3 = H$	$R = \text{Runaway}$	0.67
$Q3.3 = H$	$R = \text{Runaway}$	0.65
$Q1.2 = H \wedge Q2.3 = H$	$R = \text{Runaway}$	0.63
$Q1.2 = H \wedge Q1.5 = H$	$R = \text{Runaway}$	0.63
$Q1.4 = H \wedge Q1.5 = H$	$R = \text{Runaway}$	0.59
$Q2.3 = H$	$R = \text{Runaway}$	0.55
$Q3.5 = H$	$R = \text{Runaway}$	0.53
$Q1.4 = H$	$R = \text{Runaway}$	0.50
$Q1.2 = H \wedge Q1.4 = H$	$R = \text{Runaway}$	0.50

この観察結果をより詳細に検討するために次の分析を行う。予測に用いるルールの信頼度の下限が、それぞれ 0.9, 0.8, 0.7, 0.65, 0.60 の時の、含まれるルールの総数を図 3, 予測精度を図 4 に示す。

本研究と従来研究では予測精度はほぼ同等であることが分かる。また予測に用いたルール数 (図 3) に関して、半分近くにまで減少している信頼度区間もあることが分かる。これにより予測精度を保ちながらも注目すべき関係をより明確にできたと考える。

自己適用結果のグラフ (図 4) を見ると、信頼度 0.70 未満のルールを用いると予測精度が大きく下がっていることが分かる。よって、最も注目すべきルールが信頼度 0.70 以上の 3 つのルール

- $Q2.3 = H \wedge Q3.3 = H \implies R = \text{Runaway}$
- $Q1.5 = H \wedge Q3.3 = H \implies R = \text{Runaway}$
- $Q1.1 = H \implies R = \text{Runaway}$

である。

この 3 つのルールに含まれるアンケートの質問項目を以下に列挙する。

- Q1.1: ソフトによる実現を要求する側が、何を要求したいか分かっていなかった
- Q1.5: 頻繁な仕様変更
- Q2.3: 例外処理などの見積もり評価項目抜け
- Q3.3: 各作業分担の工程毎成果物の定義が不十分

これらの項目は対象企業において、特に重要なリスク要因であると考えられる。

## 5 まとめ

ランダムサンプリングを行ったアンケートデータに対して相関ルールマイニングを適用することで、プロジェクトの混乱とリスク要因の間の関係を表すルールを抽出することが出来た。また、冗長なルー

表 3: 提案手法により得られたルール

前提	結論	出現回数	平均信頼度	平均支持度
$Q2.3 = H \wedge Q3.3 = H$	$R = \text{Runaway}$	912	0.90	0.25
$Q1.5 = H \wedge Q3.3 = H$	$R = \text{Runaway}$	886	0.82	0.22
$Q1.1 = H$	$R = \text{Runaway}$	913	0.71	0.25
$Q1.2 = H \wedge Q3.5 = H$	$R = \text{Runaway}$	891	0.69	0.22
$Q1.4 = H \wedge Q2.3 = H$	$R = \text{Runaway}$	914	0.68	0.27
$Q1.5 = H \wedge Q2.3 = H$	$R = \text{Runaway}$	913	0.66	0.25
$Q3.3 = H$	$R = \text{Runaway}$	914	0.64	0.27
$Q3.6 = H$	$R = \text{Runaway}$	887	0.64	0.22
$Q1.2 = H \wedge Q1.5 = H$	$R = \text{Runaway}$	911	0.62	0.25
$Q5.2 = H$	$R = \text{Runaway}$	888	0.60	0.22
$Q1.4 = H \wedge Q1.5 = H$	$R = \text{Runaway}$	913	0.59	0.25
$Q2.3 = H$	$R = \text{Runaway}$	882	0.55	0.27

表 4: リスク要因

信頼度区間	本研究	従来研究
1.0 - 0.9	Q2.3, Q3.3	Q1.4, Q2.3, Q3.3
0.9 - 0.8	Q1.5	-
0.8 - 0.7	Q1.1	Q1.5, Q1.1
0.7 - 0.6	Q1.2, Q1.4, Q3.5, Q3.6, Q5.2	Q1.2
0.6 - 0.5	-	Q3.5

ルを除去することで混乱とリスク要因との関係で注目すべきルールを大きく絞り込むことが出来た。

今後も、注目すべき関係をより絞り込んで、混乱プロジェクトに影響を与える問題点をより明確にすることは重要である。

## 参考文献

- [1] B. W. Boehm. Industrial software metrics top 10 list. *IEEE Software*, 4(5):84-85, 1987.
- [2] E. H. Conrow and P. S. Shishido. Implementing risk management on software intensive projects. *IEEE Software*, 14(3):83-89, 1997.
- [3] R. Fairley and P. Rook. Risk management for software development. In *Software Engineering*, pp. 387-400. IEEE CS Press, 1997.
- [4] J. Han and M. Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers, 2001.
- [5] W. S. Humphrey. *Winning with Software: An Executive Strategy*. Addison-Wesley, 2001.
- [6] C. Jones. *Assessment and control of software risks*. Prentice Hall, Inc., 1993.
- [7] D. W. Karolak. *Software Engineering Risk Management*. IEEE CS Press, CA, 1996.

表 5: 自己適用による予測精度比較

最低信頼度	本研究			従来研究		
	適用ルール数	カバレッジ	予測精度	適用ルール数	カバレッジ	予測精度
0.90 以上	1	77%	90%	2	77%	90%
0.80 以上	2	77%	88%	2	77%	90%
0.70 以上	3	85%	85%	5	92%	83%
0.65 以上	6	92%	67%	9	100%	70%
0.60 以上	10	100%	60%	11	100%	63%

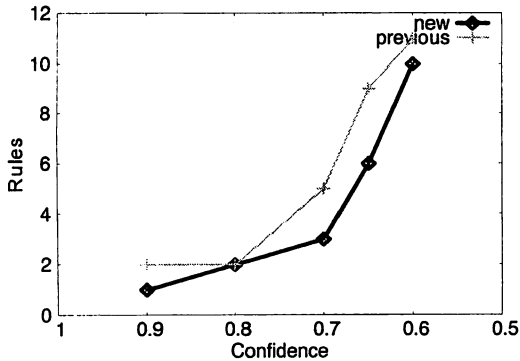


図 3: 信頼度別ルール数

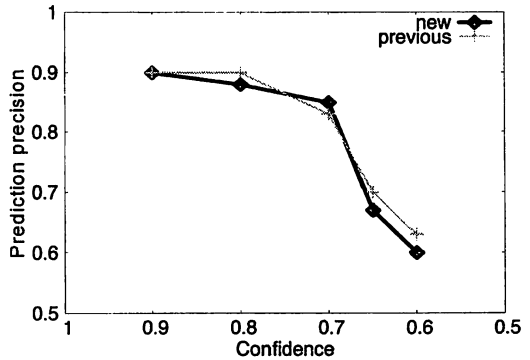


図 4: 信頼度別予測精度

- [8] J. Kasser and V. R. Williams. What do you mean you can't tell me if my project is in trouble? In *Proc. of 1st European Conference on Software Metrics (FESMA'98)*, 1998.
- [9] A. Michail. Data mining library reuse patterns using generalized association rules. In *Proc. of the 22nd International Conference on Software Engineering*, pp. 167-176, 2000.
- [10] Y. Takagi, O. Mizuno, and T. Kikuno. An empirical approach to characterizing risky software projects based on logistic regression analysis. *Empirical Software Engineering*, 10(4):495-515, 2005.
- [11] R. C. Williams, G. J. Pandelios, and S. G. Behrens. Software risk evaluation (SRE) method description (version 2.0). Technical Report CMU/SEI-99-TR-029, Software Engineering Institute, 1999.
- [12] 浜野康裕, 天寄聡介, 水野修, 菊野亨. 相関ルールマイニングによるソフトウェア開発プロジェクト中のリスク要因の分析. *コンピュータソフトウェア*. to appear.