

NEEDLE説明資料

森崎 修司

EASEプロジェクト/奈良先端科学技術大学院大学



背景 動機

- ソフトウェア開発プロジェクトの特性をまとめたデータ(コスト、プロフィール、バグ)から規則性、傾向、例外を抽出したい。
 - 改修プロジェクトのテスト工数比率は新規プロジェクトのテスト工数比率よりどのくらい大きい？

プロジェクト特性データの例

ID	開発種別	...	アーキテクチャ	...	要件定義工数	結合試験工数	総合試験工数	...	不具合密度	...
001	新規	...	3階層CS	...	80	230	200	...	0.124	...
002	改修	...	スタンドアロン	...	120	200	360	...	0.086	...
003	拡張	...	3階層CS	...	60	260	400	...	0.158	...
...



関連ルール分析

- 対象データに含まれる「AならばB」という規則(関連ルール)を全て列挙する。
- 列挙されたルールから解釈を与えることができるルールを人手により探し、役立てる。
- コンビニの購買履歴から得た関連ルールの例
休日に「レジャーシート」を買う顧客は「おにぎり」と「お茶」も同時に買っている。
「(曜日=土日) and おにぎり and お茶 ⇒ レジャーシート」
→ 休日には、レジャーシートの配置をおにぎりかお茶に近づけ、発見率、併せ買い率を上げる。

3

プロジェクト特性データから得る関連ルールの例

- 「(開発種別=拡張) and (アーキテクチャ=3階層CS) ⇒ テスト工数比率=大」
3階層アーキテクチャの機能拡張プロジェクトではテスト工数比率が高くなる。
→ 3階層アーキテクチャの機能拡張プロジェクトのテスト工数は他よりも大きく見積る。

ID	開発種別	...	アーキテクチャ	...	要件定義工数	結合試験工数	総合試験工数	...	不具合密度	...
001	新規	...	3階層CS	...	80	230	200	...	0.124	...
002	改修	...	スタンドアロン	...	120	200	360	...	0.086	...
003	拡張	...	3階層CS	...	60	260	400	...	0.158	...
...

4

相関ルール分析適用の問題点

- 項目の組合せによっては、利用価値の低いルールが多く含まれる。(開発種別とアーキテクチャ、OSとプログラミング言語など)
- 数値データ(量的変数)を含むソフトウェア特性データにそのまま適用することはできない。

ID	開発種別	...	アーキテクチャ	...	要件定義工数	結合試験工数	総合試験工数	...	不具合密度	...
001	新規	...	3階層CS	...	80	230	200	...	0.124	...
002	改修	...	スタンドアロン	...	120	200	360	...	0.086	...
003	拡張	...	3階層CS	...	60	260	400	...	0.158	...
...

5

プロジェクト特性データへの相関ルール分析適用

- 分析者が結論部を指定し、ルールを抽出する。
例) A かつ B ならば (不具合密度 = 低い)
指定する
- 数値データを扱えるようにする。
 - 質的ルール(通常の間関ルール)
数値データを区間に離散化した後、ルール抽出する。
 - 量的ルール(相関ルールの拡張)
結論部以外は数値データを区間に離散化しておき、結論部は数値データの統計値(平均、標準偏差)とする。

6

抽出ルール 質的ルール

- 表記: $A \Rightarrow B$, 支持度、信頼度、リフト値
 - 支持度: ルールの出現確率
 - 信頼度: Aが起きているときBも同時に起きている確率
 - リフト値: Aがないとき(データセット全体)の信頼度とAがあるときの信頼度の比
- 数値データの離散化
 - あらかじめ数値を区間に置き換え、ルール抽出する。
例) (1, 2, 4, 8, 9) → (小, 小, 中, 大, 大)
小: 1~3, 中: 4~6, 大: 7~9

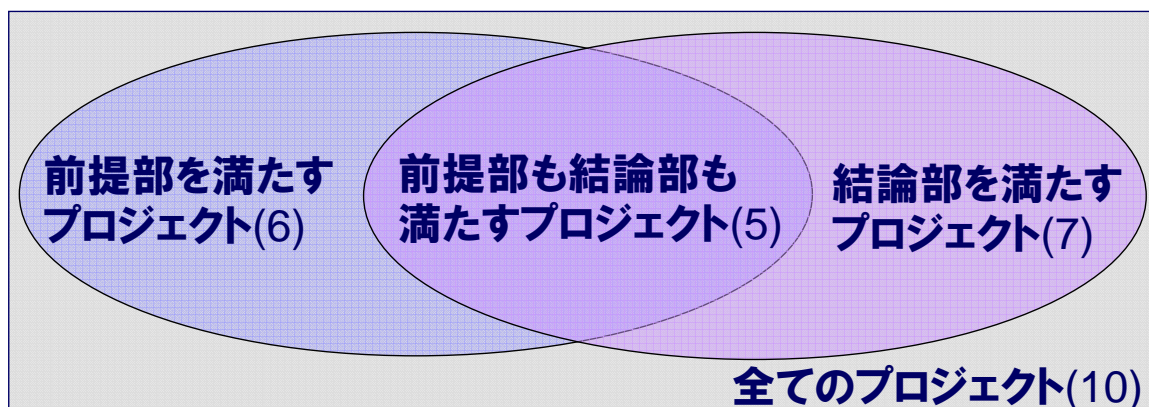
7

Copyright © 2006 Nara Institute of Science and Technology. All Rights Reserved.

ease
EASE PROJECT

抽出ルール 質的ルールの指標値例

- (開発種別=機能拡張) and (アーキテクチャ=3階層CS) ⇒ テスト工数比率=大
支持度: $\frac{5}{10}$ 、信頼度: $\frac{5}{6}$ 、リフト値: $\frac{0.83}{0.7}$



8

Copyright © 2006 Nara Institute of Science and Technology. All Rights Reserved.

ease
EASE PROJECT

抽出ルール 量的ルール

- 表記 : $A \Rightarrow B$ (平均、標準偏差)、支持度、基準化平均、基準化標準偏差
 - 基準化平均 (全体平均に対する倍率)
全プロジェクトの平均と前提Aを含むプロジェクトの平均の比
 - 基準化標準偏差 (全体標準偏差に対する倍率)
基準化平均と同様
- 結論部 (B) 以外の数値データは質的ルールと同様に区間に分割しておく。

9

Copyright © 2006 Nara Institute of Science and Technology. All Rights Reserved.

ease
EASE PROJECT

抽出ルール 量的ルールの指標値(基準化平均)

- (顧客 = 既存) and (アプリケーションサーバ = WebLogic) \Rightarrow 外部委託率 (平均0.32 標準偏差0.23)、支持度: 0.38, 基準化平均1.39, 基準化標準偏差0.8

顧客	...	アプリケーションサーバ	...	外部委託率
既存	...	WebLogic	...	0.32
既存	...	自社プロダクト	...	0.13
新規	...	WebLogic	...	0.26
既存	...	自社プロダクト	...	0.12
既存	...	WebLogic	...	0.35
新規	...	自社プロダクト	...	0.17
既存	...	WebLogic	...	0.28
既存	...	WebSphere	...	0.24

前提部を含むプロジェクト
の外部委託率の平均: 0.32

全てのプロジェクトの外部
委託率の平均: 0.23

基準化平均 = $0.32 / 0.23 = 1.39$

10

Copyright © 2006 Nara Institute of Science and Technology. All Rights Reserved.

ease
EASE PROJECT

ルールの利用例(状況把握)

- 開発規模による外部委託率の違い
 - (顧客=既存) and (開発規模=小) ⇒ 外部委託率(平均 0.28、標準偏差 0.2)、支持度0.23、基準化平均 1.8、基準化標準偏差 0.8
→ 開発規模が小さい既存顧客の案件は外部委託率が平均より1.8倍高い。
- 開発環境によるテスト工数比率の違い
 - (テスト環境=エミュレータあり) and (実機=既存) ⇒ プログラムテスト工数比率=小、支持度 0.16、信頼度 0.8、リフト値 1.4
→ 既存のハードウェアを利用した開発でソフトウェアによるエミュレーション環境がある場合、プログラムテスト工数比率が小さい

値と項目は架空のものです。 11

NEEDLE

NEEDLE入出力データ

- バグ管理表(1行=1バグ)から規則性をA→Bの形式で抽出する。

A	B	C	D	E	F	G
1	バグID 検出できなかった要因	混入工程	再現頻度	検出工程	種別	除去工程
10	9 レビュー指摘もれ	製造/単体テスト	再現頻度大	製造/単体テスト	コード不具合	製造/単体テスト

A	B	C	D	E
1	ルール	支持度	信頼度	リフト値
14	13 (検出工程 = 製造/単体テスト) & (修正工程 = 製造/単体テスト) → (修正工数 = 3.0[75人時以上1.00人時未満])	0.112745	0.496403	3.164568

