資料3

相関ルール分析ツールNEEDLE
– バグ票とプロジェクトデータへの適用事例 –

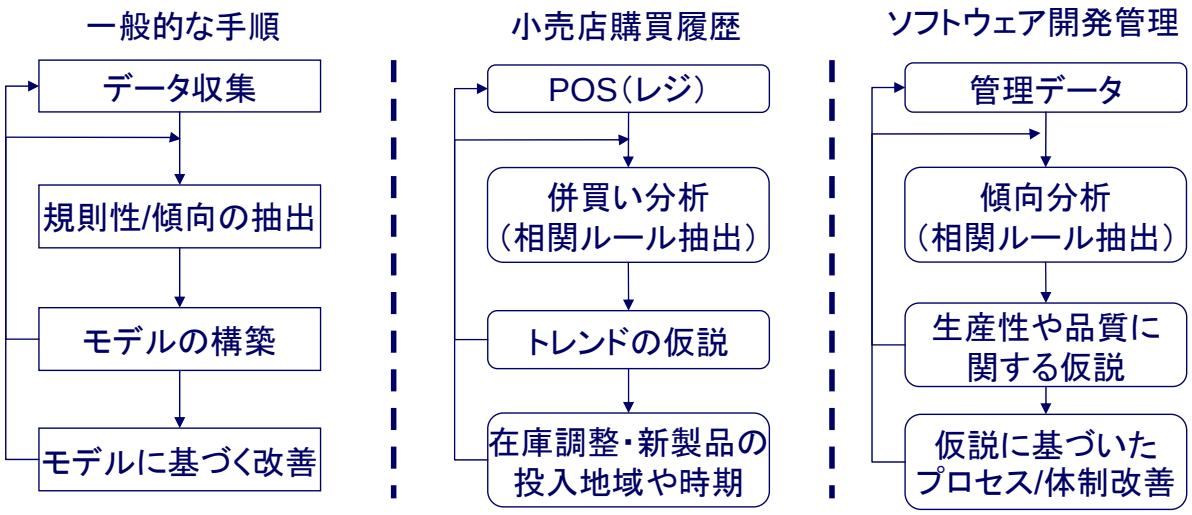
森崎　修司
EASEプロジェクト/奈良先端科学技術大学院大学

これまでの適用事例(公開分)

- 品質データ(プロジェクト毎の不具合密度実績)
  http://empirical.jp/download/past/publicdata/10th_kenkyukai/2-2.pdf
  日立システムアンドサービス 十九川氏
- 障害対応データ
  http://empirical.jp/download/past/publicdata/10th_kenkyukai/2-2.pdf
  日立システムアンドサービス 十九川氏
- プロジェクト規模、品質データ
  http://se.naist.jp/~morisaki/publications/#j-200708
  森崎、門田、玉田、松村、松本:"Mining Quantitative Rules in a Software Project Data Set", 情報処理学会論文誌Vol. 48, No. 8, pp. 2725-2734
- バグ票
  森崎、門田、玉田、松村、松本:"Defect Data Analysis Based on Extended Association Rule Mining", Proceedings of International Workshop on Mining Software Repository, pp.17-24.
  http://se.naist.jp/~morisaki/publications/#i-200705

背景

動機 – 計画/プロセス改善時の仮説立案



背景

小売店販売データへの相関ルール分析

- 対象データに含まれる「AならばB」という規則（相関ルール）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相関ルールの例
  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。

背景

プロジェクト特性データから得る相関ルールの例

- 「(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。

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	…
…	…	…	…	…	…	…	…	…	…	…

背景

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	…
…	…	…	…	…	…	…	…	…	…	…

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。
  (a) 結論部分以外は数値データを区間に離散化しておく。
  (b) 結論部分は数値データの統計値（平均、標準偏差）とする。

（ミドルウェア=開発実績なし）かつ
（設計レビュー指摘件数=低）ならば
(a) 高・中・低の3区間に分割
不具合密度（件/KLOC）=（2.3, 2.5）
(b)平均 (b)標準偏差

抽出ルール

ルールメトリクス

- 表記 ：A⇒B（平均、標準偏差）、支持度、基準化平均、基準化標準偏差
  - 支持度
    出現頻度（ルールにあてはまるプロジェクト件数の割合）
  - 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

抽出ルール

ルール例

- (顧客=既存)かつ(アプリケーションサーバ=WebSphere)⇒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	・・・	WebSphere	・・・	0.32
既存	・・・	自社プロダクト	・・・	0.13
新規	・・・	WebSphere	・・・	0.26
既存	・・・	自社プロダクト	・・・	0.12
既存	・・・	WebSphere	・・・	0.35
新規	・・・	なし	・・・	0.17
既存	・・・	WebSphere	・・・	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

適用事例1

日本ユニシス様 プロジェクトデータへの適用

変数名	取りうる値
開発プロジェクト種別	新規開発, 改修・保守, 再開発
顧客	新規顧客, 既存顧客
業種・業務	新規, 既存
協力会社	初回利用, 2回以上利用
利用形態	特定ユーザの利用, 不特定ユーザの利用
業務パッケージの利用有無	有, 無
処理形態	対話処理, オンライントランザクション処理
アーキテクチャ	スタンドアロン, メインフレーム, 3階層クライアント/サーバ, イントラネット/インターネット
開発対象プラットフォーム	Windows, Windows Server, HP-UX, Solaris, Linux, その他OS
開発対象プラットフォーム数	プラットフォーム数

変数名	取りうる値
Web技術の利用	Java Script, ASP, IIS, Apache, WebLogic, OracleAS, なし, その他
主開発言語	COBOL, Pro*C, VC++, C, VB, Developer2000, PL/SQL, C#, Java, Perl, その他
DBMSの利用	Oracle, SQL Server, なし, その他
類似プロジェクトの有無	有, 無
ピーク要員数プロジェクト全体	プロジェクト中の最大要員数
品質保証体制(結合テスト)	プロジェクトメンバが実施, 専門スタッフが実施
工数比率(要件定義, 基本設計, 詳細設計, 製造, 結合/総合試験)	全体工数に占める各工程の工数の割合(正の実数)
外部委託率	外部委託の割合(正の実数)

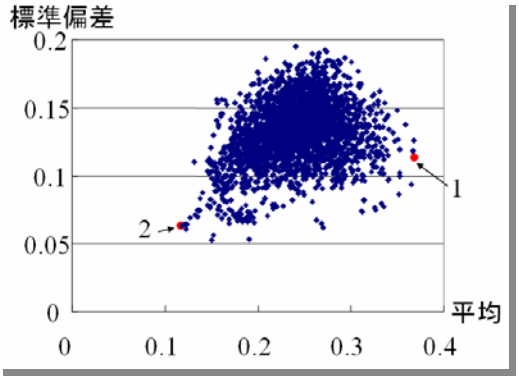
適用事例1

外部委託率を結論部とするルール例

出現頻度0.5%以上の4,000件のルールを抽出

1.（顧客＝既存顧客）and（業種＝開発実績あり）
→ 外部委託比率（平均: 0.37, 標準偏差 0.11）

2.（開発種別＝新規）and（ピーク要員数＝最小）
→ 外部委託率（平均: 0.12, 標準偏差 0.06）



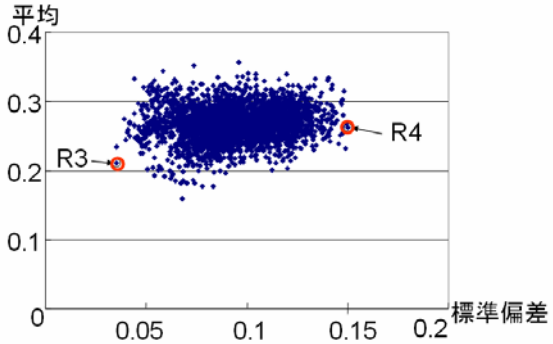
出典: 森崎、門田、玉田、松村、松本:"Mining Quantitative Rules in a Software Project Data Set", 情報処理学会論文誌Vol. 48, No. 8, pp. 2725−2734 http://se.naist.jp/~morisaki/publications/#j−200708

適用事例1

結合/総合試験の工数比率を結論部とするルール例

R3（顧客＝既存顧客）and（商用パッケージ利用＝なし）
and（製造/単体の工数比率＝大）→結合/総合試験工数比率（平均0.21、標準偏差 0.035）

R4（開発種別＝新規）かつ（業種＝開発実績あり）かつ（パートナ＝取引実績あり）かつ（外部委託比率＝大）
→ 外部委託率（平均: 0.12, 標準偏差 0.06）



出典: 森崎、門田、玉田、松村、松本:"Mining Quantitative Rules in a Software Project Data Set", 情報処理学会論文誌Vol. 48, No. 8, pp. 2725−2734 http://se.naist.jp/~morisaki/publications/#j−200708

適用事例2
バグ票への適用

- 交通情報を中心としたセンサネットワークシステムのサーバサイド(ロジック)部分の開発
- C/C++で330KLOC(流用込み)、約1年
- 複数企業によるウォータフォール型開発
- 収集フェーズ:単体～総合試験
  - データ件数 約1300
  - 組織内でバグと承認されたもの以外は入っていない。

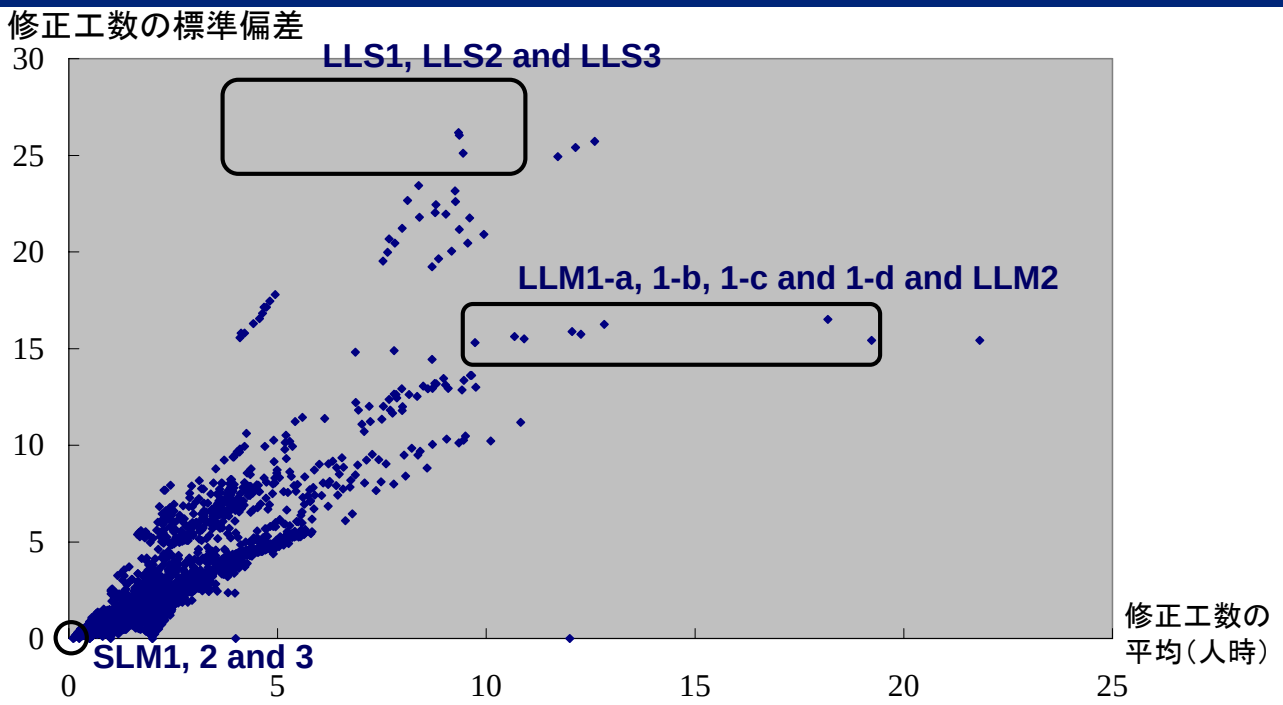
適用事例2
バグ票への適用

変数名	取りうる値
修正工数	修正に必要となった工数(人時)
混入工程	混入，修正，発見された工程 (基本設計，詳細設計，製造/単体試験，結合試験，総合試験)
発見工程	
修正工程	
機能	機能名(9機能のいずれか)
発見が遅れた理由	レビュー未実施，レビュー指摘もれ，修正確認漏れ，工程間引継ぎ，コミュニケーション不足，試験項目抽出もれ，テスト計画に含まれていない，環境が整わずテスト未実施，結果確認ミス
優先度	高，中，低
重要度	高，中，低
再現性	常に，たまに，一度だけ

出典: 森崎、門田、玉田、松村、松本:"Defect Data Analysis Based on Extended Association Rule Mining", Proceedings of International Workshop on Mining Software Repository, pp.17-24. http://se.naist.jp/~morisaki/publications/#i-200705

適用事例2

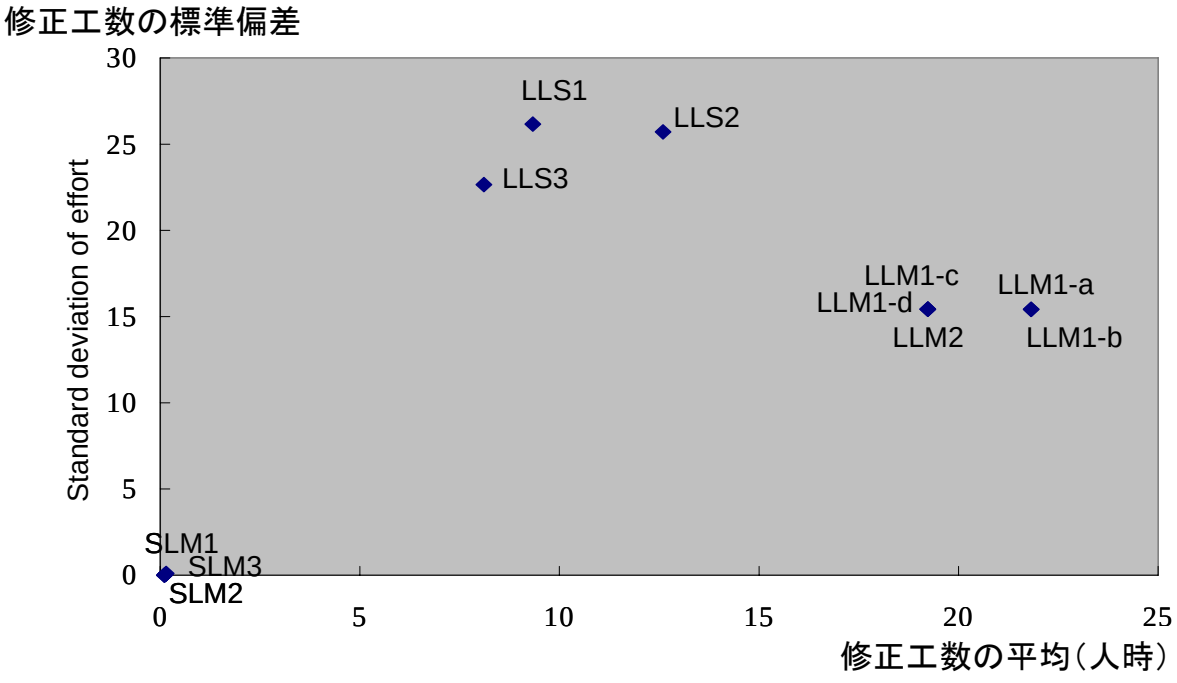
抽出ルールの分布（出現頻度 0.5%以上の17,000ルールを抽出）



出典: 森崎、門田、玉田、松村、松本："Defect Data Analysis Based on Extended Association Rule Mining", Proceedings of International Workshop on Mining Software Repository, pp.17-24. http://se.naist.jp/~morisaki/publications/#i-200705

適用事例2

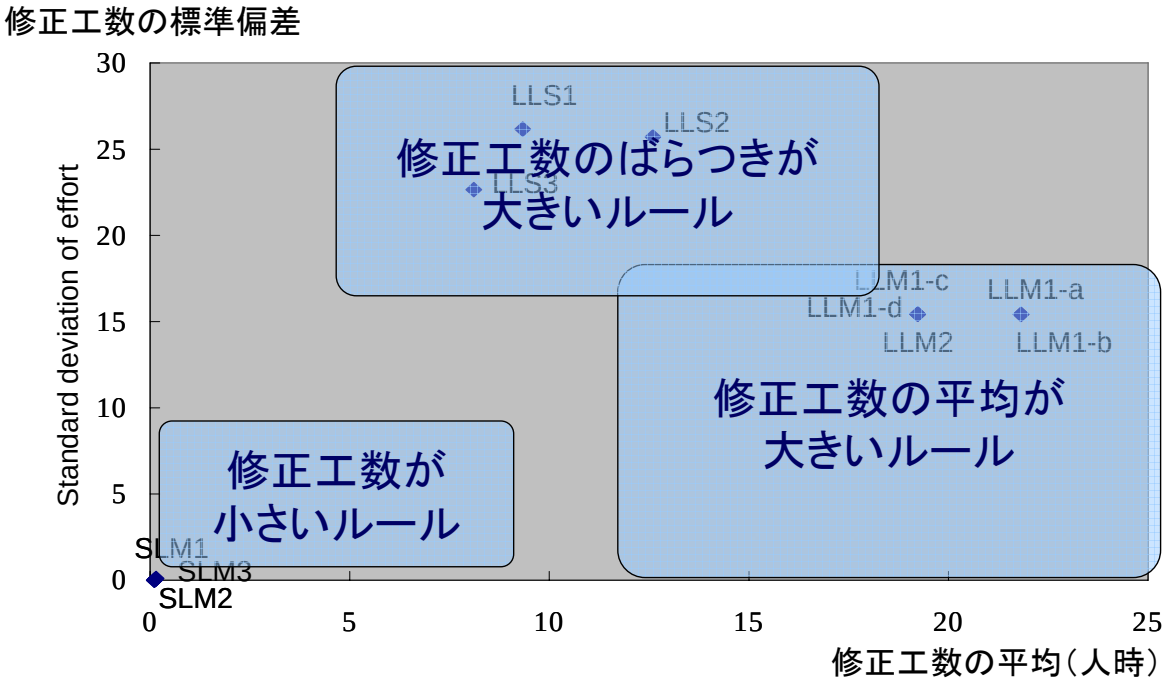
抽出ルールの抜粋（出現頻度 0.5%以上の17,000ルールを抽出）



出典: 森崎、門田、玉田、松村、松本："Defect Data Analysis Based on Extended Association Rule Mining", Proceedings of International Workshop on Mining Software Repository, pp.17-24. http://se.naist.jp/~morisaki/publications/#i-200705

適用事例2

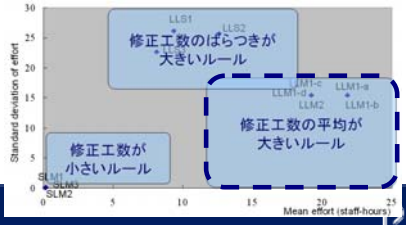
抽出ルールのプロット



適用事例2

ルール例 修正工数の平均が大きいルール

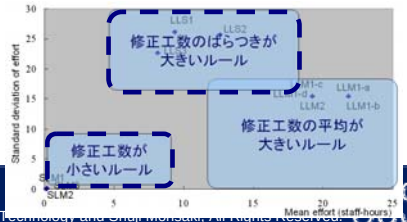
- LLM1-a (a～dは類似のルール)
  （発見工程＝総合テスト）and（優先度＝高）and（重要度＝高） → 修正工数（平均 21.8、標準偏差 15.4）
  - 出現頻度: 1.1%
  - 修正工数の全体平均との比: 10.2
- LLM2
  （混入工程＝製造/単体試験）and（再現性＝常に）and（修正工程＝総合テスト）→修正工数（平均19.2、標準偏差15.4）
  - 出現頻度: 1.1%
  - 修正工数の全体平均との比: 6.0



適用事例2 ルール例

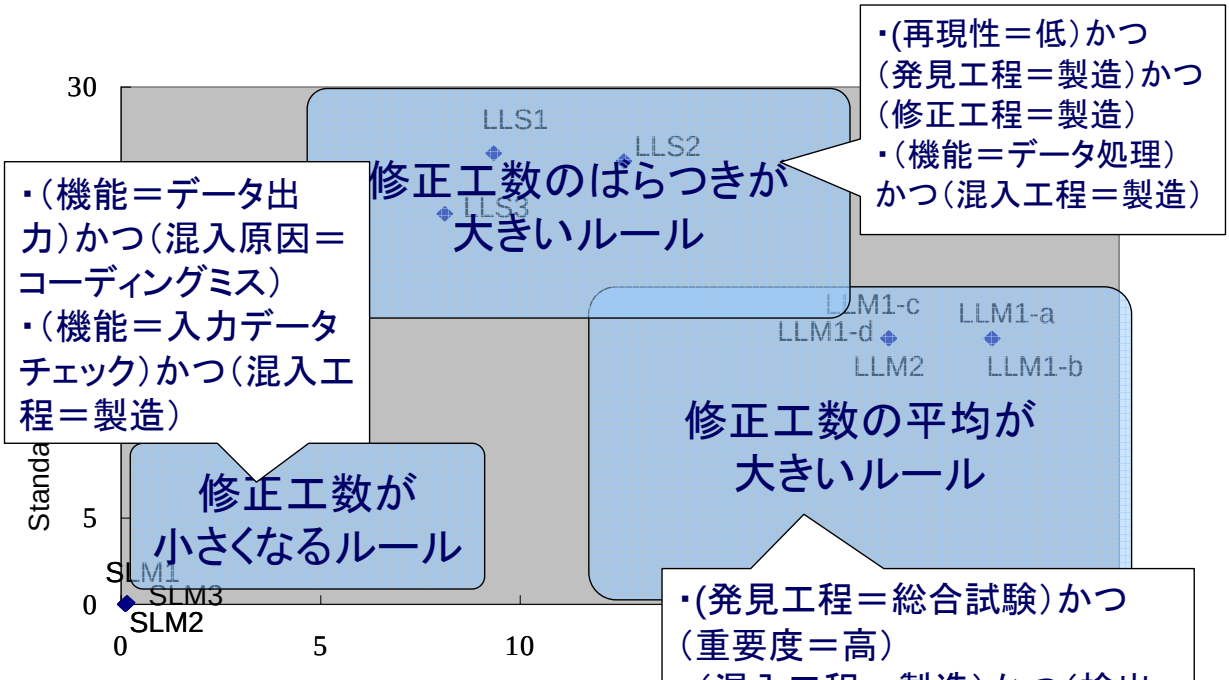
修正工数の分散大、修正工数の平均小のルール

- LLS1
  （発見工程＝製造/単体試験）and（再現性＝一度のみ）and（修正工程＝製造/単体試験）→修正工数（平均 9.3、標準偏差 26.2）
  - 出現頻度: 1.1%
  - 修正工数の全体標準偏差との比: 5.8
- SLM1
  （発見工程＝製造/単体試験）and（機能＝データ出力）and（修正工程＝製造/単体試験）and（優先度＝小）→修正工数（平均 0.1、標準偏差 0.0）
  - 出現頻度1.1%
  - 修正工数の全体平均との比: 0.05



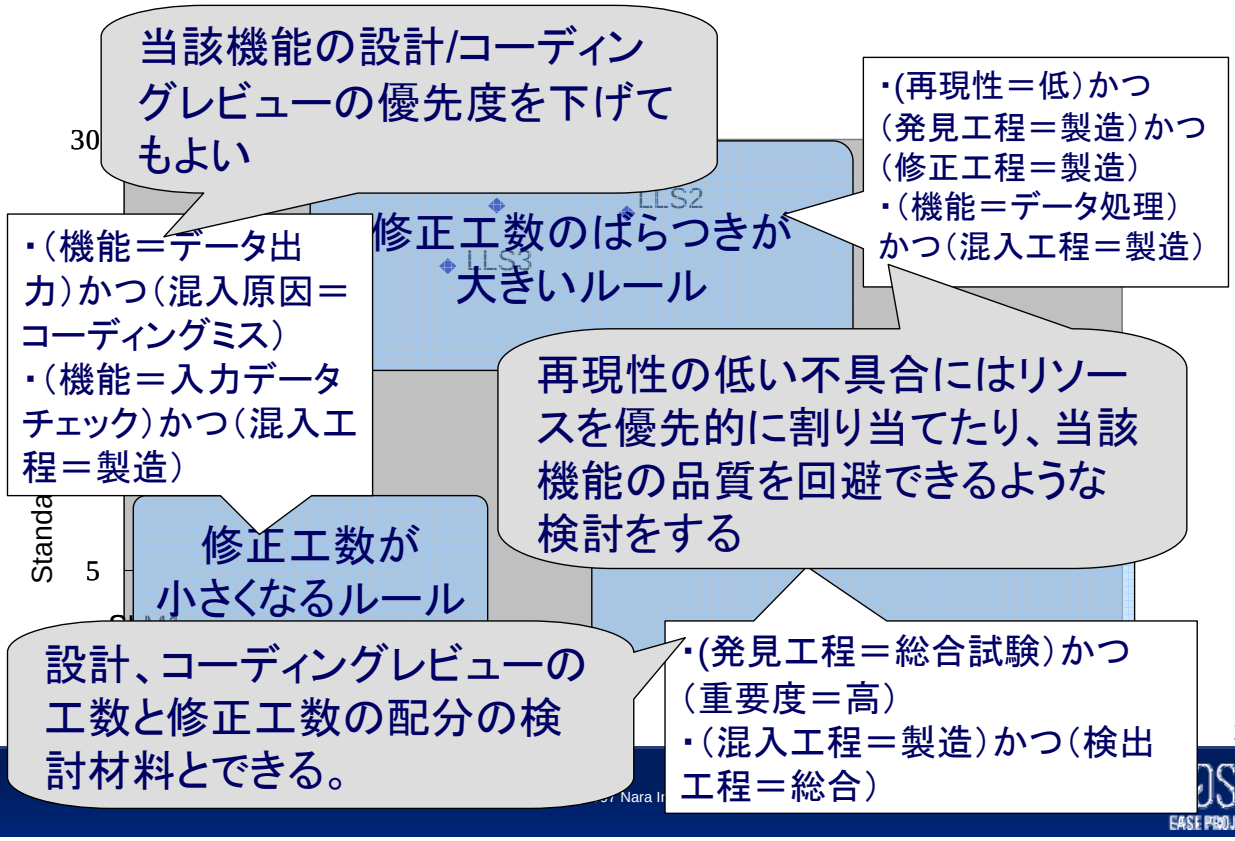
適用事例2

見積り/プロセス改善にむけた分析例



適用事例2

プロセス改善にむけた分析例



今後

- 仮説駆動型/知識駆動型例外ルールの適用中
  - 一般的で正確なルール(常識ルール)と常識ルールを特殊化することで大きく異なる結論部が得られるルール
- バグ票での例
  障害対応履歴の修正時間に基づいて特に修正工数が大きく増える障害や不具合の特徴抽出
  - 常識ルール例
    (発見作業＝単体試験)→修正工数(平均 2時間、標準偏差2.0)
  - 例外ルール例
    (発見作業＝単体試験)かつ(機能名＝A)
    → 修正工数(平均 4時間、標準偏差 4.0)

出典: 仮説駆動型例外ルール: 鈴木英之進, 共通データからの仮説駆動型例外ルール発見, 人工知能学会誌, Vol.15, No.5, (2000年9月)