

プロセス改善と定量化モデルの 進化と発展 III

- 品質予測モデルの構築

日立ソリューションズ
小室 睦



目次

- ❁ 効果説明モデルとその限界 (続き)
- ❁ 予測モデル構築のアイデア
 - 逐次近似
- ❁ 品質予測モデル
 - 反復を避ける工夫
 - 実データによる検証・評価
 - プロセス実績モデルとの関係
- ❁ まとめ



効果説明モデルとその限界

- 効果が生じるメカニズムを明らかにしている
 - 一般化線形モデルによる定式化

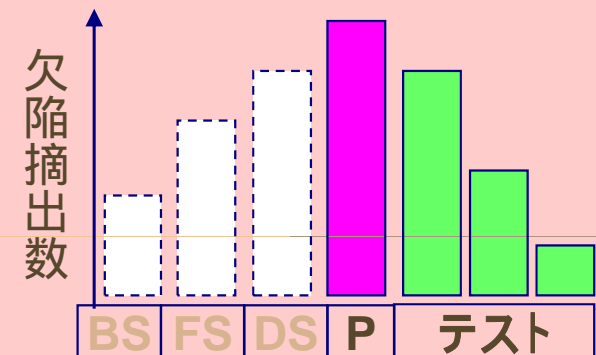
$$\log(test.defect) \sim \log(size) + d(p_j) + (\text{レビュー以外の影響})$$

- レビュー実績とテスト工程の欠陥摘出数 $test.defect$ が直接対応
 - 精度も向上
- 品質向上、原価低減効果を定量的に示す

× 予測には使えない

- 前倒し率 $d(p_j)$ の定義式に $test.defect$ そのものが含まれる
- (例) プログラミング工程を P とする。
 $f(P)$ で工程 P での欠陥摘出数を表わすと

$$d(P) = \frac{f(P)}{f(P) + test.defect}$$



予測モデル構築のアイデア

- ❁ 効果説明モデルを「変形」し、*test.defect*に関する情報を抽出する
- ❁ (実際には近似式であり等式ではないが) 等式と思えば*test.defect*に対する方程式になっている
 - 求めたい情報である*test.defect*が両辺に現われており、「陽には」解けない
 - 方程式を解く常套手段の一つである「逐次近似法」を使う



逐次近似で解ける事の保証

(記法)

関数 $dP(x)$ を $dP(x) = f(P) / (f(P) + x)$ と定める

- この記法によれば、P工程の前倒し率は $d(P) = dP(\text{test.defect})$ と書ける

(仮定1)

効果説明モデルの関係式

$\text{test.defect} = \text{size}^{a_1} \exp(c + a_2 d(P) + \beta_1 + \dots + \beta_n)$
が等式で成り立っているとすると、さらに、 $|a_2| < 4$ とする。

レビュー以外の要因
を表わす項

(仮定2)

td_0 は test.defect に対する十分よい近似値とする

(結論)

$\text{size}^{a_1} \exp(c + a_2 dP(td_0) + \beta_1 + \dots + \beta_n)$ は
 test.defect に対する td_0 よりよい近似値を与える。

レビュー情報を使っていない予測式があったとき、レビュー情報を付与することで精度向上できる

品質予測モデル – 反復を1回ですます

- ❁ 近似の反復を実際実行するのは手続き的にも複雑だし、誤差が累積する可能性もある

- 前スライドでは「等式」を仮定していた

- ❁ 近似操作を1回のみにする方法を考案

- 初期近似: $test.defect = k size$

- 第2近似:

$$a_1 \log(size) + a_2 dP(k size) + \beta_1 + \dots + \beta_n + c$$

$$= a_1 \log(size) + a_2 f(P)/(f(P) + k size) + \beta_1 + \dots + \beta_n + c$$

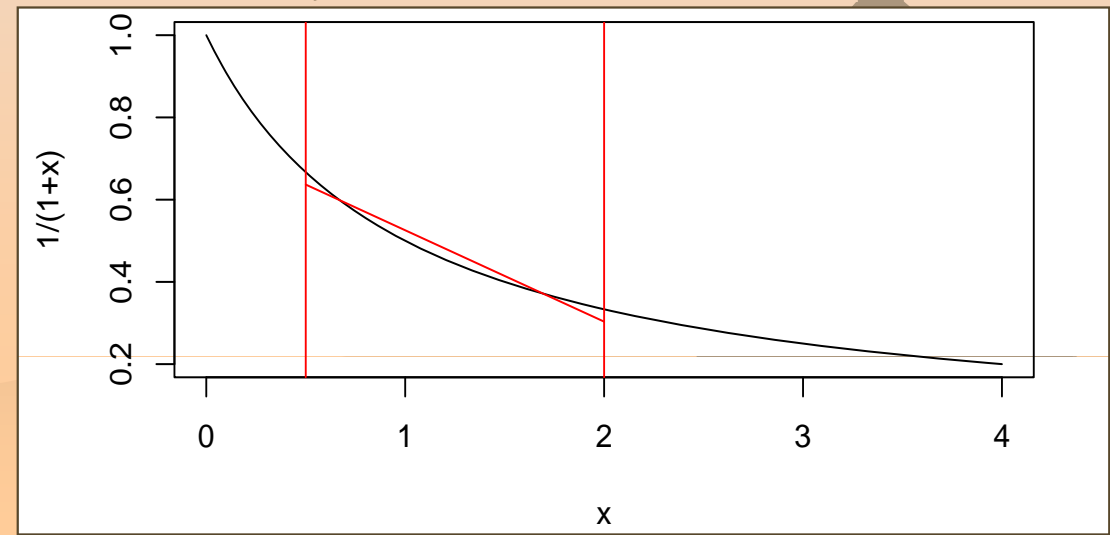
$$= a_1 \log(size) + a_2/(1 + k size/f(P)) + \beta_1 + \dots + \beta_n + c$$

- ❁ (アイデア) 関数 $y=1/(1+x)$ は x の範囲を限定すれば1次関数でよく近似できるので、

$$a_1 \log(size) + a_2' size/f(P) + \beta_1 + \dots + \beta_n + c'$$

で置換え可能

- ❁ 最初の比例定数 k も含めてこの形の線形回帰式を仮定して係数を決める



うらばなし

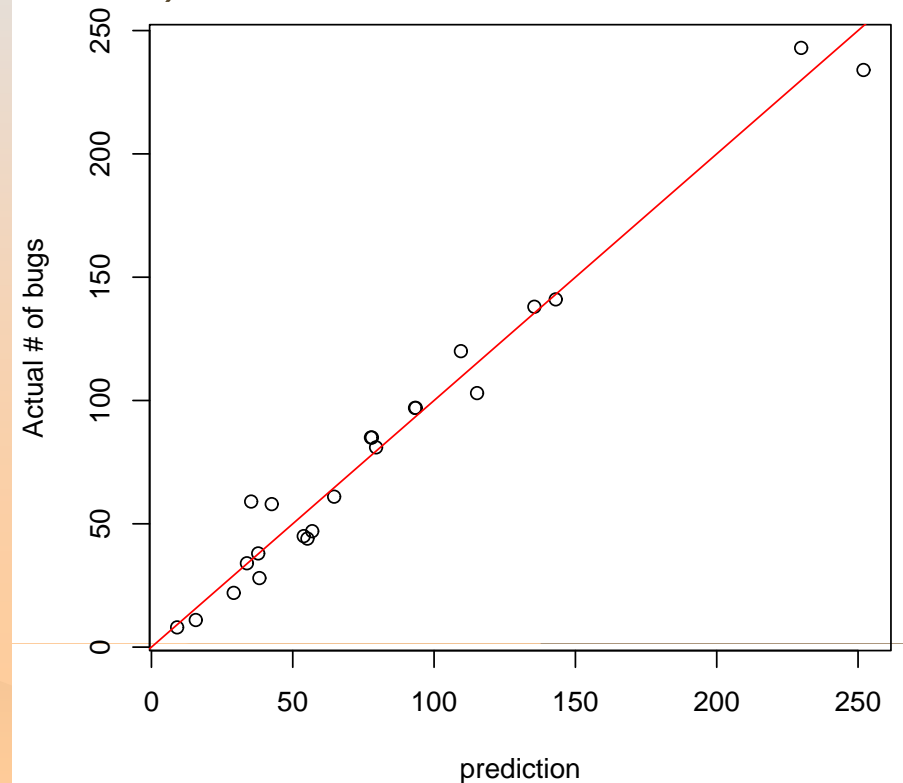
- ❁ 以上、最初からすべてを見通していたかのような話をしたが...
 - 実は、「逐次近似で方程式を解く」のは後から考えた理屈付けである
- ❁ 最初、レビューデータとテスト工程の品質を直接関係付けようという試したが、どれもうまくいかなかった
 - 時間的に離れたデータ間の関係付けの難しさ
 - レビューデータの相反する2つの側面: 品質向上効果、システムの品質状況
- ❁ 効果説明モデルを変形するアプローチ
 - 前倒し率 $d(p_j)$ が $test.defect$ に依存してしまうのが問題
 - これをテスト工程より前にわかる情報で置き換えられないものかと悩んだ
- ❁ JASPIC SPC分科会でP社のIさんから最近の成果について教えていただいた。
 - $f(P)/size$ が品質に影響を与える因子になっているという内容
 - 実はこれは $f(P)/test.defect$ の近似量になっているのではないかと考えた

JASPIC 分科会
がきっかけ

教訓: JASPICは分科会も役に立つ

実データによる検証(1)

- ❁ 同一事業部からのデータ23件を使用
 - Web系の開発で比較的小規模、言語はJava
- ❁ コーディングレビュー結果からの予測式
 - $test.defect = 5.33size^{1.003}exp(-0.0403 \log(base.size+1) + 0.616 size/f(P) - 0.597 PM - 0.496 SF)$
 - base.size: 母体規模, PM: プロジェクトマネージャの経験(ダミー変数)、SF: システム開発の自由度(ダミー変数)
 - 逸脱度: 948.95,
残差逸脱度: 52.196,
残差自由度: 17,
擬似R²: 94.5%
 - プログラミングレビューでの欠陥摘出数f(P)を増やすとバグ数test.defectは減少する

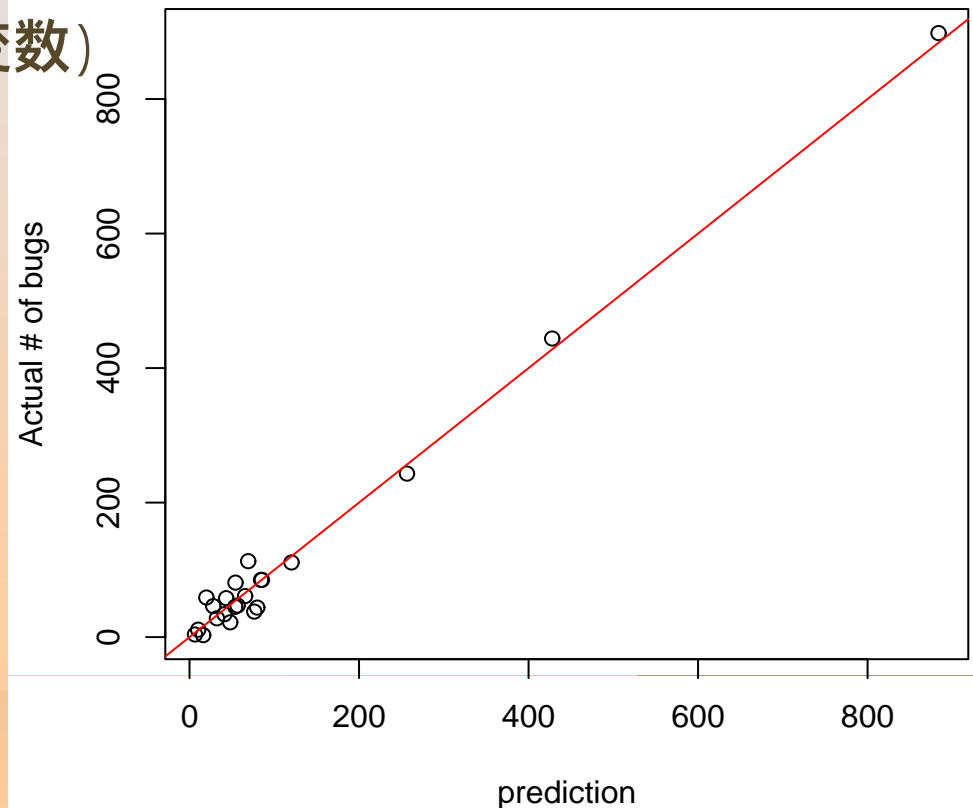


実データによる検証(2)

- ❁ 同一事業部からのデータ22件を使用
 - Web系の開発で比較的小規模、言語はJava
- ❁ ドキュメントレビュー結果からの予測式
 - ドキュメントレビュー(BS,FS,DS)を1つとみなす
 - $test.defect = 6.73size^{1.09}exp(1.81 size/(f(BS)+f(FS)+f(DS)) - 0.81 RC - 0.39 RP)$

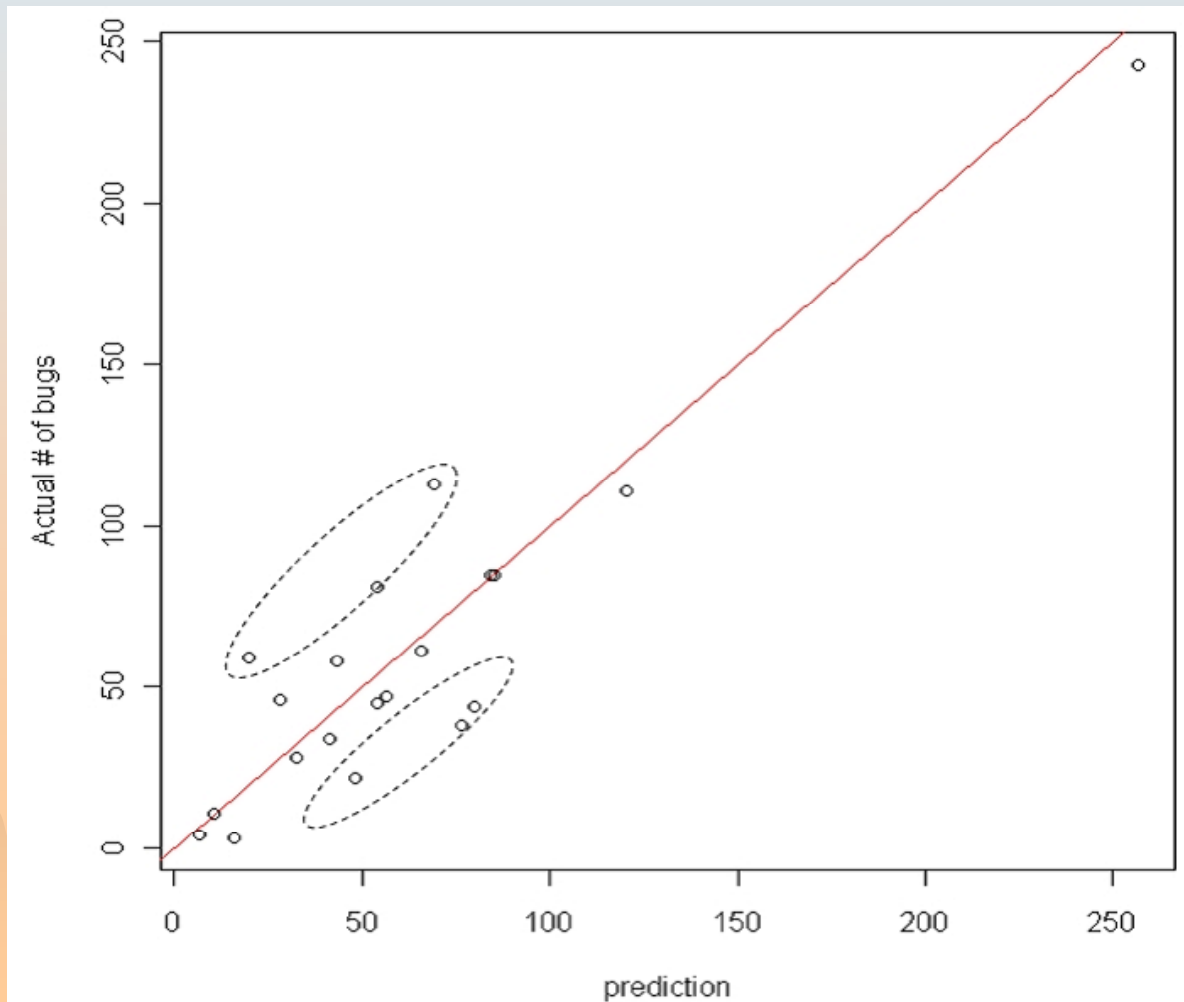
- RC: 要件の明確性(ダミー変数)
RP: リプレースかどうか(ダミー変数)

- 逸脱度: 4030.46,
残差逸脱度: 184.66,
残差自由度: 17,
擬似R²: 95.4%.
- ドキュメントレビューでの
総欠陥摘出数
 $f(BS)+f(FS)+f(DS)$
を増やすと
バグ数 $test.defect$ は減少する



実データによる検証(3)

- ❁ ドキュメントレビューからの予測の方がばらつきは大きい
- ❁ 次工程であるプログラミングとそのレビューの影響が見られる



精度の検討

- ❁ クロスバリデーション法で精度を検討
 - プロジェクトデータを一つ選ぶ
 - それを外した残りのデータでモデル構築を行う
 - そのモデルの予測結果と最初のプロジェクトデータの差異を計測する。
- ❁ コーディングレビューの23件, ドキュメントレビューの22件のデータすべてに実施
- ❁ 相対誤差
 - コーディングレビュー: 平均18.6 %
 - ドキュメントレビュー: 平均24.0 %



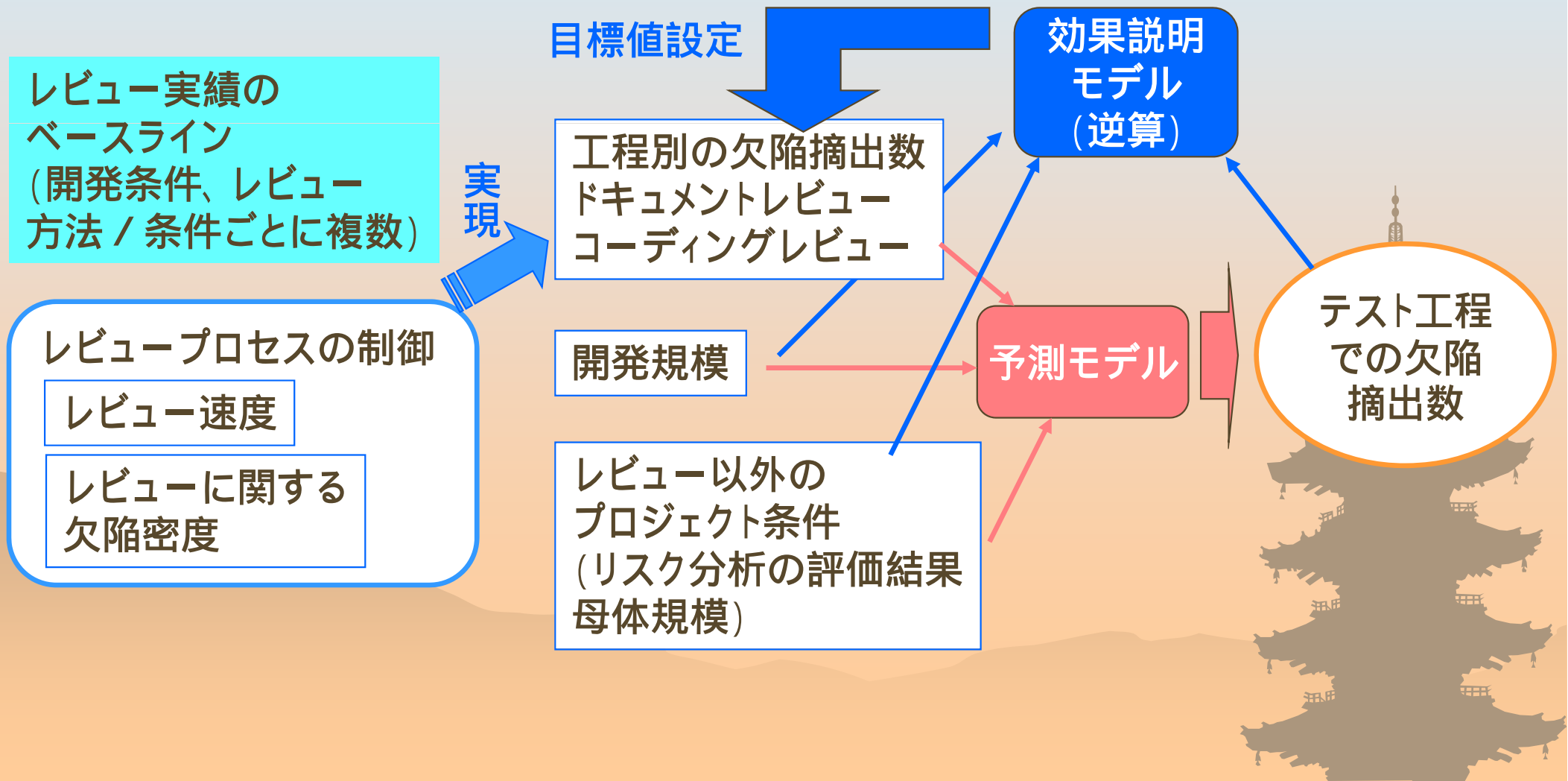
プロセス実績モデルとの関係

- ❁ レビュー実績データからの品質予測モデル
 - 予測モデルではあるが、プロセス実績モデルか？



プロセス実績モデルとの関係

- ❁ 別途、サブプロセスを制御する仕組みがあれば、プロセス実績モデルとして運用可



まとめ

- ❁ レビュー実績からテスト工程での欠陥摘出数を予測するモデルを提案
 - 効果説明モデルを「変形」
 - 一旦、説明モデルを経由することで意味のはっきりしたモデルを構築できた。
- ❁ 実データによる検証
- ❁ プロセス実績モデルとの関係
- ❁ (教訓) JASPICは分科会も役に立つ



END

