7E-4

重回帰分析による潜在バグの予測

野崎 瑞　池亀 一　遠山 正裕　田畑 実　蓮 重晴

(富士通株式会社)

1．はじめに

ソフトウェアの品質管理のひとつとして，バグ数の予測がある。『予測バグ数』を知ることによって，テスト工程において，これをひとつの指標とする事ができる。この『予測バグ数』を知る技術として，いくつか方法が考えられるが，我々は今回，重回帰分析を用いた。

工程管理や予測を行う手法のひとつとして挙げられるこの重回帰分析は，複数の特性値間の関係を明らかにする手法である。この分析を行うことによって，バグ数の予測をすることができ，更に，どのような要因がバグの出現に影響しているかを知ることができる。この影響の大きい要因を管理することによって，バグの作り込みを最少に防ぐこともできる。そこで，ソフト開発時に取られているデータから，重回帰分析によるプログラムの潜在バグの予測を試みた。

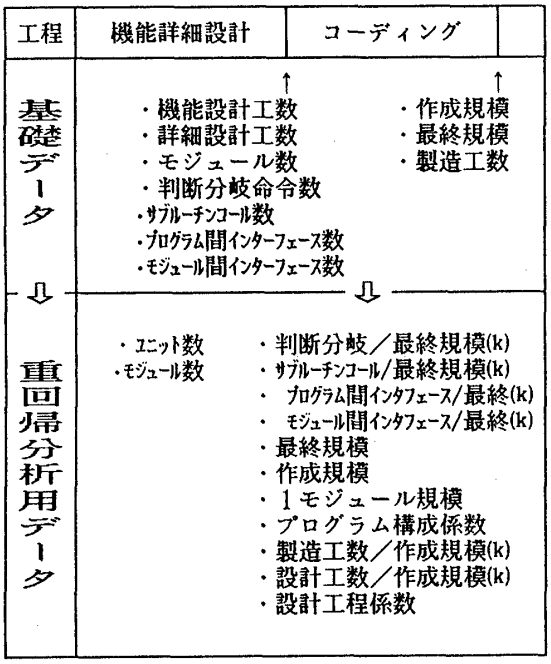

図1　分析用データ項目の抽出

2．分析の手順

ソフト開発時に収集されているデータを基礎データとし，それらのデータをもとにバク数の変動に影響を与えていると考えられる項目を作成し，重回帰分析用データとして，データベース化した。設計及び製造工程で収集される基礎データと，それに加工を施した重回帰分析用データ項目の対応は〔図1〕のとおりである。今回は，yの目的変数に“結合テストで検出されるバグ数”を取り上げ，それを予測するための説明変数として，工数，規模，複雑さ等の項目を取り上げた。すなわち，『工数，規模，複雑さ等の要因が変化すると，結合テストで検出されるバグがどう変化するか？』を予測しようとした事になる。

重回帰分析用データ項目として，基礎データを基に，３５項目を抽出した。そのうちコーディング工程終了時までに得られるデータは〔図1〕に示した１３項目となる。さらに重回帰分析を行うのに適した項目に絞りこんだ。たとえば，説明変数同士の相関が高いような場合，重回帰分析の理論の上で前提条件に反することになる。このような不都合を起こさないために項目の絞り込みを行い，〔表1〕に示す１０個の説明変数で分析を行った。

表1　項目一覧

		設定項目
目的変数		結合テスト検出バグ数
説明変数	複雑さ	①判断分岐命令数／最終規模(k) ② サブルーチンコール数／最終規模(k) ③ プログラム間インターフェース数／最終規模(k) ④ モジュール間インターフェース数／最終規模(k)
	規模	⑤最終規模 ⑥作成規模 ⑦プログラム構成係数*
	工数	⑧製造工数／作成規模(k) ⑨設計工数／作成規模(k) ⑩設計工程係数**

* ：1－（(母体－差替－削除）／最終規模)
** ：1－（機能設計工数／機能＋詳細設計工数)

Estimation of The Latent Software Errors by Multiple Regression Analysis
Mizuho NOZAKI, Hajime IKEGAME, Masahiro TOHYAMA, Minoru TABATA, Shigeharu HASU
FUJITSU LIMITED

分析を行うにあたって，なるべく環境・条件等が同じになるよう層別した。層別の観点としては，『言語』と『新規プログラム／改造プログラム』を考えた。それぞれに分類されたサンプル数は〔表2〕のとおりである。

まず，はじめに全体の分析を行い，つぎに言語で層別を行い，さらにＨＬＬで書かれたプログラムを新規－改造という観点で層別し，５種類の分析を行った。

表2　データの分類

言　語	サ　ン　プ　ル　数		
ＡＳＭ	１２		合計
ＨＬＬ	１４７		１５９
	新規	改造	
	２９	１１８	

表3　分析結果一覧

	サンプル数	回　帰　式	重相関係数	寄与率(%)	自由度修正済寄与率%
全　体	１５９	$\hat{y}=-0.02173+0.054X_6$	0.7234	52.3	52.0
Ｈ　Ｌ　Ｌ	１４７	$\hat{y}=-0.22073+0.0057X_6$	0.7288	53.1	52.8
アセンブラ	１２	$\hat{y}=-0.01353+0.0027X_6$	0.9742	94.9	94.4
ＨＬＬ－新規	２９	$\hat{y}=-17.8671+0.00787X_5+0.02346X_8$	0.6694	44.8	40.6
ＨＬＬ－改造	１１８	$\hat{y}=-0.89799+0.00288X_6+0.00068X_5$	0.7249	52.6	51.7

3．分析結果

それぞれ分類されたグループ毎の結果は〔表３〕に示すとおりである。それぞれの回帰式のｘに添えてある数字は，〔表１〕に対応している。

全体的に見て，変数選択を行った後の説明変数は１～２個で，寄与率は５０数%という，変数の数が少ない割りには高い寄与率を示している。

サンプル数などの分析条件が整っている"ＨＬＬ－改造"（図２参照）の結果を見ると，変数選択後の説明変数に『作成規模』と『最終規模』が残っている。この場合の『作成規模』とは，改造するに当たって実際にコーディングを行った規模を指している。規模が品質に影響することは一般に言われているが，今回の分析結果を見て問題なのは，『作成規模』と共に『最終規模』が残ったという事である。言い換えれば「流用できる所は流用し，『作成規模』を小さく抑えたとしても，できあがったソフトウェアの最終的な規模が大きくなってしまうようならば，バグは出やすくなる。」ということである。

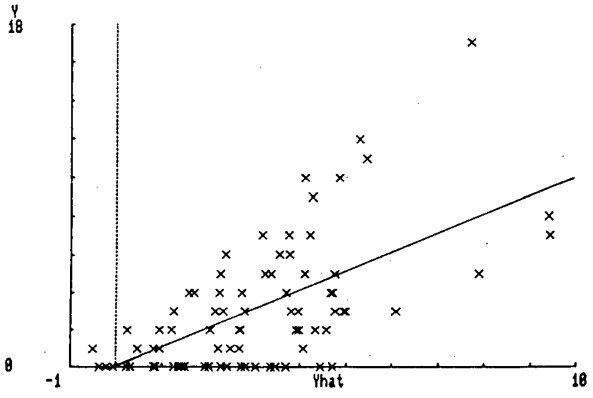

図２　予測値と実績値の散布図（HLL-改造）

4．おわりに

今回の試行は，一つのプロジェクトのデータを使っての分析であったが，今後は，多くの事例について分析を行い，分析の精度を上げるよう努めるつもりである。

《参考文献》

・奥野忠一他：多変量解析法，日科技連出版
・小林龍一：数量化理論入門，日科技連出版
・田中豊他編：パソコン統計解析ハンドブックⅡ，共立出版㈱