

ソフトウェアプロジェクトデータに基づく運用信頼性の予測に関する研究

荻原, 佑実 / OGIWARA, Yumi

(出版者 / Publisher)

法政大学大学院理工学研究科

(雑誌名 / Journal or Publication Title)

法政大学大学院紀要. 理工学・工学研究科編

(巻 / Volume)

59

(開始ページ / Start Page)

1

(終了ページ / End Page)

6

(発行年 / Year)

2018-03-31

(URL)

<https://doi.org/10.15002/00021620>

ソフトウェアプロジェクトデータに基づく運用信頼性の予測に関する研究

A STUDY ON OPERATIONAL SOFTWARE RELIABILITY PREDICTION BASED ON DEVELOPMENT PROJECT DATA

荻原 佑実

Yumi OGIWARA

指導教員 木村光宏

法政大学大学院理工学研究科システム理工学専攻修士課程

This study discusses prediction methods of software failure occurrences which are caused after software development. We analyze the actual data sets collected from the real software development companies by applying several regression methods and a neural network model. The data sets consist of quantitative data and qualitative ones. We compare the prediction performance of the operational software reliability by using these methods and data sets. In particular, we consider whether qualitative data sets can be used for the software reliability prediction.

Key Words : *Software reliability, software project data, multivariate analysis, qualitative data*

1. はじめに

(1) 研究背景

一般的に、比較的大規模なソフトウェアの開発プロジェクトは、開発期間、投入工数、費用、納期への制約などから、うまく実施することが難しいものとして知られている。さらに、近年、情報化社会が発展し、あらゆるものがソフトウェアで制御されているため、開発件数が増えざるを得ない、つまり、あまり効率的なソフトウェア開発ができない企業体でも、なんとか完成させて出荷しなければならないということもあり得る。また、QCD(品質・コスト・納期)の観点から、いわゆる、ソフトウェアの世界で言う失敗プロジェクトの発生は避けなければならない、中でも、品質が低いまま(不具合が運用時に出てしまうような状態で)客先に引き渡さないようにすることは特に重要である。このように、ソフトウェア開発の管理者はどのようにうまく開発すべきか、という問題がある。

ソフトウェアの開発プロジェクトから、様々な指標を開発者が自ら計測することは理論的には可能で、それらの値を説明変数として、出荷後に顕在化した不具合の数を目的変数としたとき、もしうまく多変量解析のモデルが構築できれば、ソフトウェア開発の管理に応用することができると考えられ、以前から研究されている[1]。また、独立行政法人情報処理推進機構技術本部ソフトウェア高信頼化センター(以下、略して“IPA/SEC”と記す)のような機関が、実際にソフトウェアプロジェクトを実施

している企業などからデータ[2]を集め、なんらかの手立てにしようともしている。しかし、実のところ、開発者は忙しく、アンケートや社内でのデータ収集・整理作業に割ける時間はないため、データの質が悪いまま蓄積されているという問題がある。例えば、答えを空欄のまま返してしまう(データが欠測する)、適当な数字を入れて(あるいは入れ間違えて)答えてしまうような問題である。

この問題に対して、欠測値を補完する方法について研究されている[3]。一方、適当なあるいは誤った数字を入れて答えてしまうような、データ蓄積上の問題については、そもそも数値で答えさせる質問ではなく、いくつかの選択肢を選ばせる方式にすれば、アンケートに答える労力がかなり減り、無理して数値データを用意させ、入力させる必要がないことから、集まったデータの質は向上すると想定できる。

したがって、そのような選択肢から選ぶタイプの定性的データを説明変数として、目的変数を説明する多変量解析モデルの性能向上が期待されるとして、Random Forestsを使った方法論も提案されている[4]。

(2) 研究目的

本研究の目的は大きく2つある。1つ目は、ソフトウェア開発プロジェクトにおいて、稼働後一定期間における不具合の発生有無の予測である。定性的データをダミー変数変換し、ロジスティック回帰分析にてどの程度の的中率が得られるのか、その後、他の方法を模索してその

精度を上げること目標としている。

2 つ目は、先行研究[4, 5]で行われなかった、説明変数を定性的データ、目的変数を稼働後一定期間に発生した不具合現象数とした場合、どのように多変量解析モデルを作ればよいのか、また、そのモデルの予測精度の評価である。もし、ソフトウェアの設計段階の定性的データで、稼働後の不具合現象数を精度よく見積もることができれば、プロジェクトマネージャーは、プロジェクト初期以降のプロジェクト実施の各種パラメータの逐次的見直しができるかもしれないと考える。

2. 分析対象データ

本研究では、IPA/SEC が提供しているデータ、「ソフトウェア開発データ白書 2014-2015」[2]の元データを用いた。このデータは、2004 年から 2013 年の間に国内 29 社の企業から収集した 3,541 件のプロジェクトデータに対し、611 項目の回答で構成されている。

(1) データセット

本研究の目的は、ソフトウェア開発プロジェクトにおける稼働後一定期間に発生した不具合現象に関する予測なので、目的変数は、プロジェクトデータ数の多い「5267 発生不具合現象数(合計)1 ヶ月」とプロジェクトデータ数の少ない「5269 発生不具合現象数(合計)6 ヶ月」の 2 種類とした。不具合には重大、中度、軽微の 3 段階の重大度があり、合計とは 3 段階それぞれの不具合現象数の合計を意味する。

本研究では、最も一般的な開発モデルである“ウォーターフォールモデル”で開発されたプロジェクトのみを研究対象とした。また、事務局が評価した信頼度が「A(データに合理性があり、完全に整合していると認められる。)」であるプロジェクトを抽出し、数値データは JMP の外れ値検出機能を使い、対象の外れ値は除外した。以下の表 1~3 にデータセット A~C の説明変数項目一覧を示す。表の左側の数字は、元データにおける項目の順番を意味するものである。なお、目的変数が「5267 発生不具合現象数(合計)1 ヶ月」のデータセットをそれぞれデータセット A1, B1, C1 とし、「5269 発生不具合現象数(合計)6 ヶ月」のデータセットをそれぞれデータセット A6, B6, C6 とする。

a) 定量的データ

データセット A1 は 191 件のプロジェクト、データセット A6 は 59 件のプロジェクトを含む。

表 1 データセット A の説明変数項目一覧

5004	SLOC 実測値 SLOC
5007	同言語別 SLOC 1
5179	実績工数(開発)基本設計
5180	実績工数(開発)詳細設計
5181	実績工数(開発)製作
5182	実績工数(開発)結合テスト
5183	実績工数(開発)総合テスト(ベンダ確認)
5223	平均要員数プロジェクト全体
5232	ピーク要員数プロジェクト全体
5251	テストケース数結合テスト
5253	検出バグ現象数結合テスト
11004	SLOC 実績値(追加・新規)
11015	プロジェクト開発工数計画値(基本設計開始時点)

b) 定性的データ

データセット B1 は 477 件のプロジェクト、データセット B6 は 86 件のプロジェクトを含む。

表 2 データセット B の説明変数項目一覧

103	開発プロジェクト種別
123	実績の評価(コスト)
124	実績の評価(品質)
125	実績の評価(工期)
201	業種 1
308	アーキテクチャ 1
312	主開発言語 1

c) ソフトウェアの設計段階の定性的データ

データセット B からソフトウェアの設計段階でわかる項目のみを取り出し、説明変数とした。データセット C1 は 477 件のプロジェクト、データセット C6 は 86 件のプロジェクトを含む。

表 3 データセット C の説明変数項目一覧

103	開発プロジェクト種別
201	業種 1
308	アーキテクチャ 1
312	主開発言語 1

3. 分析手法

データセット B1, B6, C1, C6 は定性的データ、つまり質的変数で構成されているため、変数間に距離の概念が考慮されておらず、定量的データと同等に分析することができない。そのため、0 または 1 で構成されたデータセットとなるようダミー変数変換を施した。

(1) 一般化線形モデル

本研究では、一般化線形モデル[6]であるロジスティック回帰分析、ポアソン回帰分析、負の二項回帰分析を用いて分析した。これらの手法において、5分割交差検証を用い、Rでパラメータ推定を行い、それぞれの手法の適当な式にあてはめ、予測値を算出した。5分割交差検証とは、全データを5個に分割し、そのうちの4個を学習データ、残りの1個をテストデータとし、標本群それぞれをテストデータとし、5回検証するものである。なお、5個の標本群それぞれの結果を平均したものが評価指標となる。本研究ではExcelのRAND関数を使って、無作為に5分割した。以下の式(1)~(3)は、予測値の算出に用いた式である。なお、 X を説明変数ベクトル、 β をパラメータベクトルとする。

ロジスティック回帰分析

$$p = \frac{1}{1+e^{-\beta X}} \quad (1)$$

ポアソン回帰分析(期待値 λ)

$$\lambda = e^{\beta X} \quad (2)$$

負の二項回帰分析(期待値 μ)

$$\mu = e^{\beta X} \quad (3)$$

(2) ニューラルネットワーク

ニューラルネットワークにおいては、SAS社が視覚的に分析できるよう開発した統計解析ソフトウェアであるJMPを使って分析した。JMPのニューラルネットワークは無作為 k 分割方式であり、 $k=5$ とした。この機能は検証するたびに予測結果が異なる。また、隠れ層は1つで、活性化関数はTanH関数(双曲線正節関数)である。TanH関数は、入力値が-1~1の範囲に変換されるように尺度化されたロジスティック関数である。 x を説明変数 X の線形結合とすると、TanH関数は

$$\frac{e^{2x}-1}{e^{2x}+1} \quad (4)$$

で表される。

4. 分析結果

ニューラルネットワークを除いた各分析では、的中率と予測値 \geq 実測値(安全側の予測)の割合を算出し、モデルの評価を行った。予測値 \geq 実測値の割合を算出した理由は、実際は不具合が発生しないのに発生すると予測することは、実際は発生するのに発生しないと予測するよりも問題はないと考えたからである。また、ポアソン回帰分析と負の二項回帰分析では、的中した割合と実測値より1だけ多く予測してしまった割合の合計値も算出し

た。割合の数値は、小数点第二位を四捨五入した。

(1) ロジスティック回帰分析による予測結果

ロジスティック回帰分析は2種類しかできないので、目的変数の値が0(不具合発生なし)なら0,1以上(不具合発生あり)なら1と置き換えて分析した。

表4 ロジスティック回帰分析による予測結果

データセット	的中率	予測値 \geq 実測値
A1	70.1%	79.5%
A6	61.0%	80.0%
B1	68.8%	82.0%
B6	60.4%	82.4%
C1	58.9%	82.6%
C6	40.7%	60.3%

(2) ニューラルネットワークによる予測結果

ニューラルネットワークは、JMPの無作為 k 分割($k=5$)という方式で行うため、検証するたびに結果が変わるので、隠れ層のノード数2~10に対して、それぞれ5回ずつ検証し、最も予測結果が良かったもの(実際は不具合が発生するのに、発生しないと予測した割合が少ないものとする)をそのノード数における結果とした。

a) 2種類

データセットB1, C1を除いて、いずれかのノード数での的中率100%の予測結果を得ることができた。表5, 6にデータセットB1, C1の最も予測結果が良かったものを示す。表の()内の数字は個数を表す。

表5 データセットB1におけるニューラルネットワーク(隠れ層のノード数【5】)の予測結果

実測値	予測値	割合
データセットB1	不具合発生あり	不具合発生なし
不具合発生あり	90.9% (30)	9.1% (3)
不具合発生なし	1.6% (1)	98.4% (61)

データセットB1に関して、的中率のみに着目すると、隠れ層のノード数5において最大95.8%であった。

表6 データセットC1におけるニューラルネットワーク(隠れ層のノード数【8】)の予測結果

実測値	予測値	割合
データセットC1	不具合発生あり	不具合発生なし
不具合発生あり	75.8% (25)	24.2% (8)
不具合発生なし	16.1% (10)	83.9% (52)

データセットC1に関して、的中率のみに着目すると、隠れ層のノード数7において最大85.3%であった。

b) 3 種分類

目的変数の値が 0(不具合発生なし)なら“不具合発生なし”，1 以上 9 以下なら“不具合数 1~9”，10 以上なら“不具合数 10 以上”と置き換え，JMP を使って 3 種分類を行った．その結果，データセット *B6*(隠れ層のノード数 7, 9)とデータセット *C6*(隠れ層のノード数 5)において 100%の的中率が得られた．他のデータセットにおいても最低でも 81.1%の的中率が得られた．表 7~10 にデータセット *A1*, *A6*, *B1*, *C1* の最も予測結果が良かったものを示す．表の () 内の数字は個数を表す．

表 7 データセット *A1* におけるニューラルネットワーク (隠れ層のノード数 【7】，【10】) の予測結果

実測値	予測値 割合		
	不具合数 1~9	不具合数 10 以上	不具合発生なし
データセット <i>A1</i>			
不具合数 1~9	90.9% (10)	0.0%	9.1% (1)
不具合数 10 以上	0.0%	100.0% (2)	0.0%
不具合発生なし	0.0%	0.0%	100.0% (25)

表 8 データセット *A6* におけるニューラルネットワーク (隠れ層のノード数 【3】) の予測結果

実測値	予測値 割合		
	不具合数 1~9	不具合数 10 以上	不具合発生なし
データセット <i>A6</i>			
不具合数 1~9	100.0% (5)	0.0%	0.0%
不具合数 10 以上	0.0%	100.0% (2)	0.0%
不具合発生なし	25.0% (1)	0.0%	75.0% (3)

表 9 データセット *B1* におけるニューラルネットワーク (隠れ層のノード数 【10】) の予測結果

実測値	予測値 割合		
	不具合数 1~9	不具合数 10 以上	不具合発生なし
データセット <i>B1</i>			
不具合数 1~9	88.5% (23)	0.0%	11.5% (3)
不具合数 10 以上	14.3% (1)	57.1% (4)	28.6% (2)
不具合発生なし	0.0%	1.6% (1)	98.4% (61)

表 10 データセット *C1* におけるニューラルネットワーク (隠れ層のノード数 【10】) の予測結果

実測値	予測値 割合		
	不具合数 1~9	不具合数 10 以上	不具合発生なし
データセット <i>C1</i>			
不具合数 1~9	65.4% (17)	3.8% (1)	30.8% (8)
不具合数 10 以上	14.3% (1)	57.1% (4)	28.6% (2)
不具合発生なし	9.7% (6)	0.0%	90.3% (56)

(3) ポアソン回帰分析による予測結果

表 11 ポアソン回帰分析による予測結果

データセット	的中率	$0 \leq (\text{予測値} - \text{実測値}) \leq 1$	予測値 \geq 実測値
<i>A1</i>	18.9%	67.1%	81.7%
<i>A6</i>	6.7%	28.5%	66.2%
<i>B1</i>	31.0%	53.5%	80.1%
<i>B6</i>	43.0%	51.8%	74.4%
<i>C1</i>	25.6%	51.6%	80.7%
<i>C6</i>	26.8%	34.8%	67.3%

的中率と $0 \leq (\text{予測値} - \text{実測値}) \leq 1$ を比べると，最低でも 8%，最大 48.2%上昇している．また，不具合現象数が 0 または 1 のプロジェクトが的中率の大部分を占めていた．

(4) 負の二項回帰分析による予測結果

表 12 負の二項回帰分析による予測結果

データセット	的中率	$0 \leq (\text{予測値} - \text{実測値}) \leq 1$	予測値 \geq 実測値
<i>A1</i>	39.8%	66.0%	79.0%
<i>A6</i>	8.3%	28.5%	64.4%
<i>B1</i>	35.9%	51.8%	80.3%
<i>B6</i>	44.2%	49.4%	74.4%
<i>C1</i>	31.9%	52.6%	79.5%
<i>C6</i>	22.2%	35.0%	67.3%

的中率と $0 \leq (\text{予測値} - \text{実測値}) \leq 1$ を比べると，最低でも 5.2%，最大 26.2%上昇している．また，ポアソン回帰分析と同様，不具合現象数が 0 または 1 のプロジェクトが的中率の大部分を占めていた．

(5) ポアソン回帰分析と負の二項回帰分析の比較

表 11, 12 を見ると，データセット *C6* を除いて，負の二項回帰分析による予測の方が的中率が高いことがわかる． $0 \leq (\text{予測値} - \text{実測値}) \leq 1$ と予測値 \geq 実測値に関しては，僅かな差はあるが，同じくらいの予測結果となっている．しかし，予測値 \geq 実測値の割合には，予測値が実測値よ

り異常に多く見積もってしまった数も含まれている。そこで、それぞれのデータセットにおいて、ポアソン回帰分析と負の二項回帰分析による予測値と実測値の関係を図1, 2のように散布図を作成し、視覚化した。極端な異常値がある場合は、図から除外し、図の下部に記した。なお、()内の数字は実測値を示す。図1, 図2に例としてデータセット BI における予測値と実測値の関係を示す。

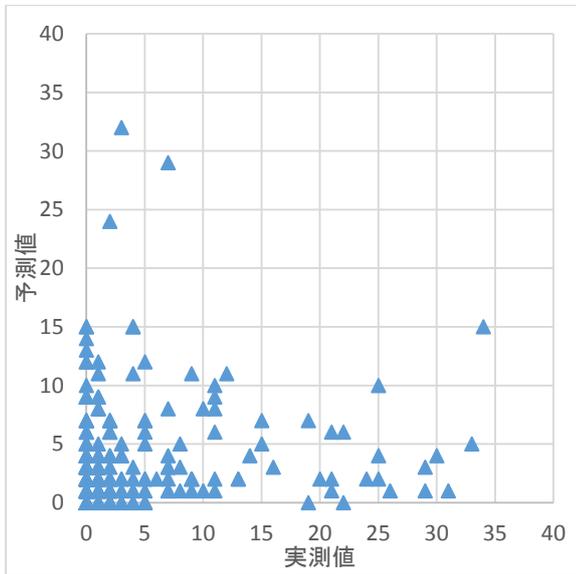


図1 データセット BI におけるポアソン回帰分析による予測値と実測値の関係

除外した異常値はなし。

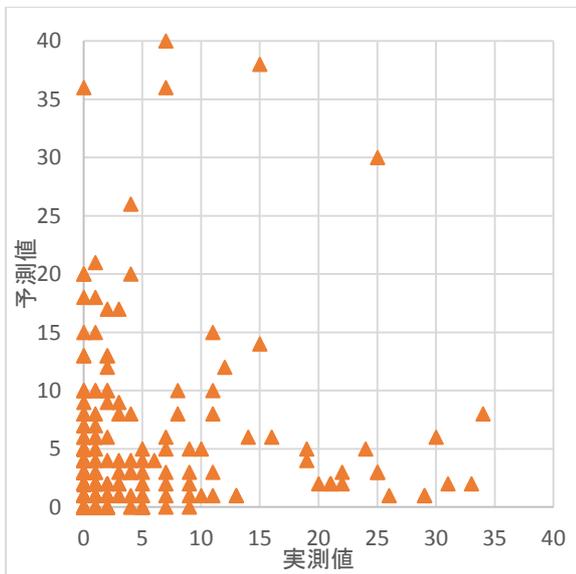


図2 データセット BI における負の二項回帰分析による予測値と実測値の関係

除外した異常値は、43(0), 44(1), 48(11), 51(0), 51(1), 72(0), 80(5), 90(0), 99(1), 144(8), 528(0)。

結果として、負の二項回帰分析の方が実測値とかなり

異なる予測を多くすることがわかった。

5. 考察

不具合が発生するか否かの2種類に関して、ニューラルネットワークによる予測結果がすべてのデータセットに対して、ロジスティック回帰分析による予測結果を上回ることが確認できた。これはニューラルネットワークの特徴である、隠れ層で活性化関数によって非線形変換がうまく作用し、幅広くモデル表現ができることが起因であると考えられる。しかし、5分割する方法が異なるため、ニューラルネットワークでも同じ学習データとテストデータで検証する必要があると考えられる。また、3種類に関して、最大100%的中する予測結果を得ることができたので、目的変数の数値を更に細かく分類して予測することも可能であると考えられる。ニューラルネットワークは少ないデータでも精度良く予測することが可能であり、正確な数値を予測できなくても、目安として大まかな数値範囲を予測できるので、不具合発生有無の予測手法として有用であると考えられる。

また、稼働後一定期間に発生する不具合現象数の予測に関して、負の二項回帰分析による予測結果的中率がデータセット C6を除いて、ポアソン回帰分析による予測結果的中率を上回ることが確認できた。ポアソン分布は期待値と分散が一致していることが前提の分布なので、分散が大きくなる場合は当てはまりにくくなるため、あまり高い中率を得られなかったと考えられる。しかし、負の二項回帰分析はポアソン回帰分析と比べて、実際に発生する不具合現象数よりも極端に多く予測してしまうプロジェクトを多く含むことがわかった。 $0 \leq (\text{予測値} - \text{実測値}) \leq 1$ と $\text{予測値} \geq \text{実測値}$ に関しては、僅かな差はあるが、同じくらいの予測結果となっており、一概に優劣を決めることはできない。

また、説明変数がソフトウェア設計段階の定性的データのデータセットは、ステップワイズ法による AICc と BIC の値を意識したデータセットに比べて予測値的中率は劣ったが、 $\text{予測値} \geq \text{実測値}$ の割合はさほど変わらなかった。説明変数の効果量を測る方法を模索し、目的変数に効果の大きい説明変数のみで学習を行うことで改善されると考える。

6. おわりに

不具合現象数の予測に関して、的中率は最大44.2%、1だけ多く予測してしまった割合を含めても最大67.1%と、まだ実際に開発ソフトウェアの運用信頼性予測の手法としては導入できるような予測結果は得られず、結果として手法の優劣を決めることはできなかったが、ゼロの多い計数データに負の二項回帰分析を当てはめると、過大に推定してしまう傾向がある[7]ので、目的変数がゼロを多く含む場合の解析手法を考慮することで、改善が期待

されると考える。

今後の課題として、定性的データと計測が自動でできる定量的データ(つまり、いい加減な数値を従業員が入力することがない評価値)を混ぜたものを説明変数とした場合、予測精度はどうなるのかを検証するなどが挙げられよう。

参考文献

- [1] S. Yamada and J. Yamakawa, "An Empirical Study on Statistical Analysis Based on Software Process Monitoring Data with Initial Project Risks", Proc. of ISSAT Intern. Conf. on Reliab. and Quality in Design, pp. 384-388 (2011).
- [2] IPA/SEC, 独立行政法人情報処理推進機構技術本部ソフトウェア高信頼化センター, ソフトウェア開発データ白書 2014-2015 (2015).
- [3] 森田 貴之, 「欠測値のあるソフトウェアプロジェクトデータに基づく信頼性解析」, 法政大学大学院理工学研究科修士論文(2015).
- [4] 新井 雄一郎, 「定性的なソフトウェアプロジェクトデータに基づくプロダクト品質予測に関する研究」, 法政大学大学院理工学研究科修士論文(2016).
- [5] 寒河江 徹, 「ソフトウェア開発プロジェクトにおける不具合発生の予測に関する研究」, 法政大学大学院理工学研究科修士論文(2017).
- [6] A. J. Dobson, (田中 豊, 森川 敏彦, 山中 竹春, 富田 誠訳), 一般化線形モデル入門, 共立出版(2011).
- [7] 南 美穂子, 「ゼロの多いデータの解析：負の2項回帰モデルによる傾向の過大推定」, 特集「環境リスクと統計解析ーデータ基盤構築と解析ー」, 統計数理, 第61巻 第2号 271-287(2013).