

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

寒河江, 徹 / SAGAE, Tetsu

(出版者 / Publisher)

法政大学大学院理工学・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 / 法政大学大学院紀要. 理工学・工学研究科編

(巻 / Volume)

58

(開始ページ / Start Page)

1

(終了ページ / End Page)

5

(発行年 / Year)

2017-03-31

(URL)

<https://doi.org/10.15002/00014361>

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

THE STUDY ON PREDICTION OF MULFUNCTION IN THE SOFTWARE DEVELOPMENT PROJECT

寒河江徹

Tetsu SAGAE

指導教員 江崎和博

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

Today, system development for problem solution is carried out in various companies and groups as an information-oriented society advance. However, the success propriety of the project is often judged subjectively and qualitatively by a project member, and we cannot expect a future project improvement. Furthermore, a system that the success propriety is vague produces various malfunction after introduction. Therefore, we tried development of the malfunction prediction technique using the statistical technique to distinguish it whether you could release a system quantitatively and objectively in this study. We write down the findings here.

Key Words : *MT system, prediction, success propriety*

1. はじめに

近年、情報化社会が進むにつれて、様々な企業や団体が経営課題の解決に向けてシステム導入のプロジェクトを行っている。しかし、これらのプロジェクトの多くは完了後にその結果をプロジェクトメンバーによる主観的、定性的な判断によって曖昧に評価されることが多く、これではプロジェクトの今後の改善が望めない。

近年、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）により、国内企業におけるソフトウェア開発プロジェクトのデータの収集と蓄積が行われている。そこで、これらのデータを使ってプロジェクトを客観的、定量的に評価することが出来れば、失敗プロジェクトをしっかりと認識し要因を分析することで、次のプロジェクトでは対策を打つことが出来ると考える。

これまでソフトウェア開発プロジェクト成功に向けた管理手法がいくつか提案されているが、それらの多くはそもそもプロジェクト成功の定義が曖昧なものが多く、現状としてはプロジェクト成功の定義は各企業によって異なっているために有用な結果が得られていないのである。そこで、本研究ではプロジェクト成功可否について厳密に定義している先行研究[1]を参考にして研究を進めた。先行研究[1]では、原価、品質、工期の3点における成功可否を判別する手法として重回帰分析、線形判別分析を用いた判別手法を開発することで、プロジェク

トの定量的、客観的な評価手法を提案した。

本研究では、ソフトウェア開発プロジェクトの客観的な評価にあたり、リリース後の不具合発生について注目した。ソフトウェア開発は開発の後に、リリース後のシステムの運用という大事なフェーズがある。実際にテストを重ねたシステムをリリースした後、大きな不具合が発生し業務に支障をきたすといった例も多く存在する。開発企業ではリリース前にカットオーバーのための点検作業を行っているが、この点検についても各企業の主観的などころが多く、重大規模の不具合が実際に発生していることから改善の余地があると考えられる。開発終了時にリリース後の不具合発生が客観的に予測出来れば、カットオーバーに向けた点検時の1つの評価指標となり、プロジェクトの客観的、定量的な評価にもつながる。

不具合発生を予測するにあたってこれまで使っていた線形判別分析に加えて、新たに品質工学で用いられているパターン認識手法MT法を用いた。

本研究ではこのようにソフトウェア開発プロジェクトにおけるリリース後不具合発生予測をテーマとして解析を行った。開発プロジェクト終了時点で不具合発生の有無を予測し、導入前のプロジェクトの客観的評価指標の一つとすることが大きな目的である。

2. 解析対象データ

本研究では、IPA/SECが収集している2004年から2014

年までの国内のソフトウェア開発におけるプロジェクトのデータを使用している。このデータは、3325件のプロジェクトあり、各々のプロジェクトに対して611項目の属性を持っている。

解析データの抽出条件として、必要な属性データに欠損が無く、データの信頼性がB以上で、平均要員数の規模が3人以上のプロジェクトを抽出した。その結果236件のデータを解析対象とすることとした。

3. プロジェクトの成功可否の概念

(1) プロジェクトの成功可否の判定

本研究では、IPA/SECのプロジェクトのQCD(品質, コスト, 工期)の視点から評価する成功可否の評価結果が具体的かつ客観的にプロジェクトの成功可否を示す可能性が高いと考えた。また、データの収集範囲がプロジェクトの計画段階から完了段階までと限られているため、表1、表2に示す成功可否の判定基準を定義した[2]。

表1 プロジェクト計画の成功可否の判定基準

	成功	失敗
品質	規模の属性の「目標値」の根拠が明確で実行可能性を検討済	規模の属性の「目標値」の根拠が不明確又は実行可能性未検討, 計画無し
工期		
コスト		
総合	「品質」「工期」「コスト」計画の全てが成功	「品質」「工期」「コスト」計画のいずれか又は全てが失敗

表2 プロジェクト実績の成功可否の判定基準

計画		自己評価実績	
		失敗	成功
判定	成功	品質, 工期, コスト実績のいずれか, 又は全てが失敗	品質, 工期, コスト実績の全てが成功
	失敗	判定不可能	

(2) 成功度の定量化基準

本研究では、IPA/SECのプロジェクト成功可否のQCDの実績に対して、成功度という定量的な指標を導入した[2]。成功度は、あるプロジェクトがひどい失敗だったのか、成功に近い失敗だったのかなど成功の程度が区別できるようにするために導入している。表3にQCDの実績の成功度の定量化基準を示す。

表3 QCDの成功度の定量化基準

	品質評価_124d	工期評価_125d	コスト評価_123d
優良	a=1.4: 計画値より20%以上少ない $0 \leq x \leq 80$	a=1.4: 納期より前倒し $y < \text{計画納期}$	a=1.4: 計画より10%以上少ないコストで達成 $0 \leq z \leq 90$

成功	b=1.0: 計画値以下 $80 < x \leq 100$	b=1.0: 納期通り $y = \text{計画納期}$	b=1.0: 計画通り(±10%) $90 < z \leq 110$
失敗	c=0.6: 計画値の50%以内の超過 $100 < x \leq 150$	c=0.6: 納期10日未満遅延 計画納期 $< y <$ 計画納期+10	c=0.6: 計画の30%以内の超過 $110 < z \leq 130$
	d=0.2: 計画値の100%以内の超過 $150 < x \leq 200$	d=0.2: 納期30日未満遅延 計画納期+10 $\leq y <$ 計画納期+30	d=0.2: 計画の50%以内の超過 $130 < z \leq 150$
	e=-0.1: 計画値の100%を超える超過 $200 < x$	e=-0.1: 納期を30日以上遅延 計画納期+30 $\leq y$	e=-0.1: 計画値の100%を超える超過 $150 < z$

プロジェクトの属性データにおいて品質, 工期, コストのデータは成果を直接的に表す重要な要因であるため、定量化し不具合発生の予測に活用した。さらに、QCDの視点を取り入れたプロジェクトの総合的な成功度(12oc)を変数として(1)式で求められるように定義した。

$$12oc = 124d + 125d + 123d \quad (1)$$

(3) 不具合発生の定義

本研究ではリリース後6か月間に起こった不具合現象数を用いて予測を行った。IPA/SECの収集したデータでは不具合現象はその規模によって「軽度」「中度」「重大」の3ランクに分けているが、今回は「中度規模以上の不具合発生有無」と「重大規模のみの不具合発生有無」の2パターンの予測を行った。次に、不具合発生予測にあたり不具合発生の有無を表3のように2パターン定義し、以下の値を定義した。

表3 不具合発生有無の定義

	不具合発生なし	不具合発生あり
予測 中度・重大	5257_発生不具合現象数(重大)6ヶ月 5261_発生不具合現象数(中度)6ヶ月 が共に0	5257_発生不具合現象数(重大)6ヶ月 5261_発生不具合現象数(中度)6ヶ月 のどちらかが1以上
重大予測	5257_発生不具合現象数(重大)6ヶ月 が0	5257_発生不具合現象数(重大)6ヶ月 が1以上

- 21bg3_中度・重大不具合発生有無(判別変換): 中度重大不具合発生有無を式(4.1)に基づいて変換
- 22bg3_重大発生不具合有無(判別変換): 重大不具合発生有無を式(4.1)に基づいて変換

4. 解析手法

本研究では不具合発生を予測するために線形判別分析と品質工学で用いられるパターン認識の手法であるMT法を用いた。

5. 線形判別分析

(1) 線形判別分析とは

線形判別分析とは、2つのA群、B群に対して、サンプルがA群に属するか、またはB群に属するかを判別してくれる手法である。したがって、サンプルの所属する群を判別するためには、図1のように1本の線で2つの群に分離しておく必要がある。

図1に示されている直線を与える式 z を線形判別関数という。そして、この線形判別関数 z が2つの群に分ける最も良い境界線となる。線形判別関数 z の式に値を代入して、正の値か負の値かによって、そのサンプルを判別する。線形判別関数の求め方は、A群とB群の相関比を最大とするように係数を決める方法や重回帰分析を利用する方法など様々な方法がある。

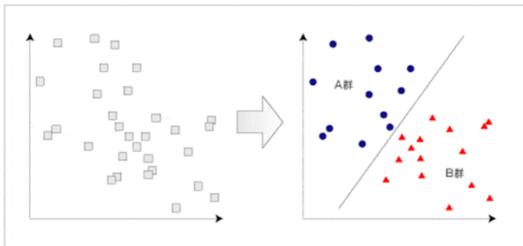


図1 線形判別関数 z における判別のイメージ

本研究では、重回帰分析を利用して線形判別関数を求めることにした。その理由としては、分析に使用するデータの変数が多く、判別分析でも適切な変数を選択できるからである。

(2) 分析の手順

1. 目的変数の値を変換する。
2. 回帰分析を用いて分析し、求めた回帰式を線形判別式とする。
3. 線形判別式を用いて推定値を求め、判別率を求め。
4. 判別率をもとに線形判別式の有効性を検討する。

以上の手順で分析しモデルを構築してそのモデルの有効性を検討する。

6. MT法

(1) MT法とは

MT法は、マハラノビスの距離を品質工学の体系に取り入れたパターン認識の手法である。目的に対して均質な集団の多次元情報を総合して1つのものさしを作り、この均質な集団に属さない個々の対象がこのものさしからどれくらい離れているかマハラノビスの距離を求める

という手法だ。

現在は文字認識や工場の不良予測、また健康診断など幅広い分野で応用されている。対象が2つの集団からどちらに属するかを判別する判別分析に対して、MT法では定義する集団は1つであり、対象データと単位空間の中心との距離を計算するというアプローチとなっている。多次元のデータを統合することで対象の判別や予測が可能であるという点からソフトウェア開発プロジェクトにも応用できると考えた。

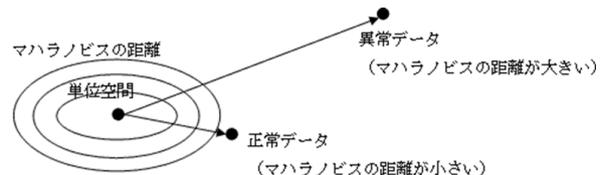


図2 MT法の概念

(2) 分析手順

1. 学習データの中の不具合発生なしのデータで単位空間をつくる。
2. テストデータに適応して、それぞれの馬原のビス距離を測る。
3. X二乗分布に基づいて不具合発生有無の予測と項目データの有効性を検証する。

7. 分析結果

(1) 線形判別分析による分析結果

a) 中度以上の不具合発生予測

リリース後6か月の間に中度以上の不具合が発生するかどうかを、線形判別分析を用いて予測した。

まず、中度・重大不具合発生予測結果を示す。

10050_実績工数(総計人時)_プロジェクト全体

10055_実績工数(総計人時)_製作

10057_実績工数(総計人時)_総合テスト(ベンダ確認)

判別得点はこれらの属性データにより導出した。

学習データ判別率 0.737288136

テストデータ予測率 0.737288136

学習データ判別率もテストデータ予測率も等しく73%となった。

b) 重大のみの不具合発生予測

重大のみの不具合発生予測結果を以下に示す。選択された属性データは以下の通りである。

10055_実績工数(総計人時)_製作

10056_実績工数(総計人時)_結合テスト

10149_実績工数(総計人時)_工程配分不可

学習データ判別率 0.830508475

テストデータ予測率 0.779661017

学習データ判別率は83%、テストデータ予測率は77%であった。

(2) MT法における分析結果

MT法についても線形判別分析と同じデータセットで分析を行った。MT法では判別の良否を測るためにSN比が使われている。動特性の0点比例式のSN比と望大特性のSN比についてそれぞれが1番高くなる項目選択を行った。

a) 中度以上の不具合発生予測

リリース後6か月の間に中度以上の不具合が発生するかどうかを、MT法を用いて予測した。その結果を図3に示す。

動特性SN比	テストデータ予測的中率	0.7118
	動特性SN比	-24.47
	望大特性SN比	-6.876
	しきい値	2.603
望大特性SN比	テストデータ予測的中率	0.7711
	動特性SN比	-30.44
	望大特性SN比	-4.923
	しきい値	2.098

図3 中度以上の不具合予測の結果

単位空間をつくるのに用いた属性データは以下の通りである。

動特性SN比重視

- ・5167_月数(実績)プロジェクト全体
- ・10050_実績工数(総計人時)_プロジェクト全体
- ・125_実績の評価(工期)

望大特性SN比重視

- ・123_実績の評価(コスト)
- ・10052_実績工数(総計人時)_要件定義
- ・10053_実績工数(総計人時)_基本設計
- ・10054_実績工数(総計人時)_詳細設計
- ・10055_実績工数(総計人時)_製作
- ・10149_実績工数(総計人時)_工程配分不可

b) 重大のみの不具合発生予測

リリース後6か月の間に重大のみ不具合が発生するかどうかを、MT法を用いて予測した。その結果を図4に示す。

動特性SN比	テストデータ予測的中率	0.8389
	動特性SN比	-23.86
	望大特性SN比	-7.172
	しきい値	2.098
望大特性SN比	テストデータ予測的中率	0.8644
	動特性SN比	-24.01
	望大特性SN比	-6.653
	しきい値	2.214

図4 重大のみの不具合予測の結果

単位空間をつくるのに用いた属性データは以下の通りである。

動特性SN比重視

- ・10053_実績工数(総計人時)_基本設計
- ・10054_実績工数(総計人時)_詳細設計
- ・10055_実績工数(総計人時)_製作
- ・10056_実績工数(総計人時)_結合テスト
- ・10057_実績工数(総計人時)_総合テスト(ベンダ確認)
- ・10058_実績工数(総計人時)_総合テスト(ユーザ確認)

望大特性SN比重視

- ・10053_実績工数(総計人時)_基本設計
- ・10054_実績工数(総計人時)_詳細設計
- ・10055_実績工数(総計人時)_製作
- ・10056_実績工数(総計人時)_結合テスト
- ・10057_実績工数(総計人時)_総合テスト(ベンダ確認)

(3) 分析結果のまとめ

線形判別分析とMT法それぞれの不具合発生予測の的中率について表4に示す。

表4 各モデルにおける的中率と閾値

	線形判別分析	MT法	
中度以上不具合予測	0.737288136	動特性	0.711864407
		望大特性	0.771186441
重大のみ不具合予測	0.779661017	動特性	0.838983051
		望大特性	0.86440678

表4より中度以上の不具合よりも重大のみの不具合予測の方がより高い的中率となった。手法で比べると、MT法の望大特性のSN比重視の予測で最大の的中率が見

られた。

8. 考察

解析の結果、中度以上の不具合発生予測については最大 77%、重大のみの不具合発生予測については最大 86%の的中率を見ることが出来た。

線形判別分析に関しては、MT 法に比べて不具合発生なしのプロジェクトをありと判別するケースが多く見られた。中度以上の不具合発生予測においては学習データ判別的中率が 73%と高い値とは言えず、線形判別では今回の分布データをうまく分けることが出来なかったことが原因と思われる。重大のみの不具合発生予測については、学習データ判別の中率は 83%と中度以上の場合よりも高かったものの、テストデータ予測的中率にはうまく反映されなかった。

MT 法については、中度以上の予測、重大のみの予測のどちらに関しても望大特性の SN 比を最大にする単位空間の方が高い中率を見ることが出来た。理由として、今回は発生不具合件数の有無についての予測であり、発生不具合件数を推定しているわけではないために動特性の SN 比は判別という結果にはあまり反映されなかったと思われる。また、今回のデータセットには大小様々な規模のプロジェクトデータが使われていて、単位空間の均質性に少し欠けていたことが誤判別の原因と思われる。線形判別分析と MT 法の比較という視点では、線形判別に比べ、望大特性 SN 比重視の予測から 1 番高い中率を見ることが出来た。しかし一概に優劣を決めることは出来ない。

9. おわりに

結果的に、ソフトウェア開発プロジェクトにおいてシステム開発終了時に 70%~80%の確率でリリース後の不具合発生有無を予測することができた。まだ実際に導入できるような高確率とは言えないが、まだ改善の余地があると考えられる。

本研究では不具合発生を予測するにあたって、発生不具合件数と工数に関するデータとの間に相関がみられたため、主に工数に関する属性データを使用した。全体工数や参加人数の多いプロジェクトで一概に不具合件数が多いという訳ではなく、中度・重大など大規模の不具合に関しては全体の工数の中でどの工程にどのくらいの時間と人を割いているかが大きく関係していると思われる。その点では、項目間の相関が距離に反映される MT 法は今回の研究対象に対して適切であり、的中率にも表れている。

今後の課題としては、不具合件数の発生有無だけではなく、高い精度で軽度・中度・重大それぞれの不具合発生件数を予測することであると思う。今回用いた MT 法は 2 種類の「判別」に適しているといわれているが、数値の「予測」に適している MTA 法など新たな MT システム

の手法を使うことでより高い精度を実現できると考えている。

参考文献

- [1] 桑原希尽, ソフトウェア開発プロジェクトの工期と成功可否の関係の研究, 研究報告, 情報処理学会研究報告, pp1-20, 2016
- [2] IPA/SEC: ソフトウェア開発データ白書, 2013-2014, 2013
- [3] 江崎和博: 情報システム開発プロジェクトの品質に影響を及ぼす要因の研究, 情報システムと社会環境研究会第 129 回研究発表会, pp1-8(2014).
- [4] 長谷川良子: マハラノビス・タグチシステムのはなし, 日科技連出版社(2004).
- [5] 鈴木真人: 試して極める, 品質工学 MT システム解析法入門, 日刊工業新聞社(2012).
- [6] 田口玄一: 確率・統計, 日本規格協会(1981).
- [7] 手島昌一 長谷川良子: 入門 MT システム, 日科技連, 2008