Vol.2015-SE-190 No.7
2015/12/15

非公開データからの類似データ生成

佐々木健太郎[†1]　門田暁人[†1,†2]　松本健一[†2]　大岩佐和子[†3]　押野智樹[†3]

概要：実証的ソフトウェア工学の研究では，多数のソフトウェア開発プロジェクトのデータが必要となる．ところが，研究者が利用できる公開データは古いものが多く，研究を進める上で大きな障害となっている．本稿では，企業における非公開データを用いた学術研究を可能とするために，非公開データの特徴量のみをデータ収集企業から受け取ることを想定し，特徴量の似た研究用データを人工的に作成する方法を提案する．

キーワード：実証的ソフトウェア工学，ソフトウェア開発プロジェクトデータセット，実証的研究

On Creating an Artificial Data Set from a Secret Data Set

Kentaro SASAKI[†1]　Akito MONDEN[†1]　Kenichi MATSUMOTO[†1]
Sawako OHIWA[†2]　Tomoki OSHINO[†3]

Abstract: Many empirical software engineering studies require a large software project data set to conduct an empirical research. However, publicly available project data sets are mostly old, and this lowers the validity and usefulness of research outcomes. To enable doing a research using a recent not-public (secret) industry data set owned by a software company, this paper explores the creation of an artificial project data set from the secret data set, assuming that a researcher obtains only small amount of information about the secret data set from a data owner company.

Keywords: Empirical software engineering, software development project data set, empirical study

1. はじめに

実証的ソフトウェア工学の研究分野では，多数のソフトウェア開発プロジェクトの実績データが必要となる．ところが，一般公開されているデータは古いものが多く，研究の妥当性や信頼性を確保する上で大きな問題となっている．例えば，Promise Repository[11]にて公開されているDesharnais[4]，COCOMO'81[3]，Kemerer[6]，Albrecht[1]といったプロジェクトデータセットは，1980年代のソフトウェア開発のデータである．

一方，多くの企業では最新のソフトウェア開発のデータを計測・蓄積しているが，大学の研究者がそれを使った研究を行うことはますます困難となりつつある．昨今では，個人情報保護，及び，コンプライアンス重視のために，企業における機密保持がより厳格となり，データを社外に出すことがより困難となっているためである．

本稿では，データ収集企業における非公開データを用いた学術研究を可能とするために，非公開データの特徴量のみをデータ収集企業から受け取ることを想定し，特徴量の似た研究用データを人工的に作成する方法を提案する．基本的なアイデアとしては，各変数値の分布や，変数間の相関係数といったデータの特徴量をデータ収集企業から受け取り，類似する分布を持った変数値の集合をランダムに生成する．そして，変数間の関係が保たれるように，変数値の入れ替えを行う．本方法では，データそのものは全くランダムに生成したものであるので，データの持ち出しや公開への道が開けると期待される．また，データ件数を増やすことも可能なため，データ分析結果の信頼性を高める効果も期待される．

本稿では，一般財団法人経済調査会において2007～2011年に収集されたソフトウェア開発プロジェクトデータを用い，提案方法の有効性を確認する．

以降，2章では，従来研究について述べ，3章では，提案方法を説明する．4章では，ケーススタディを紹介し，5章はまとめと今後の課題である．

2. 従来研究

2.1 ソフトウェア開発データを用いた研究の現状

ソフトウェア開発工数予測の研究では，評価実験用の標準データセット群として，Promise Repository[11]にて公開されているMaxwell[10], Desharnais[4], COCOMO'81[3], Kemerer[6], Albrecht[1]といったプロジェクトデータセットが（近年のトップジャーナルの論文においても）よく用いられている2)7)9).

ところが，これらの公開データは古いものが多く，また，プロジェクト件数が少ないものもあり，研究の妥当性や信頼性を確保する上で大きな問題となっている．例えば，前

†1 岡山大学
Okayama University
†2 奈良先端科学技術大学院大学
Nara Institute of Science and Technology
†3 一般財団法人経済調査会
Economic Research Association

記の，Desharnais，COCOMO’81，Kemerer，Albrecht はいずれも 1980 年代のソフトウェア開発のデータであり，現代のソフトウェア開発とは，開発環境やプロセスが大きく異なる可能性がある．そのため，これらのデータセットを用いた研究の妥当性には疑問が残る．また，Kemerer は 15 件，Albrecht は 24 件とプロジェクト件数が少なく，これらのデータセットを用いた実験結果の信頼性には疑問が残る．

一方，最新のソフトウェア開発のデータを用いた研究や分析も実施されているが，分析結果のみが公開され，データ自体は公開されていない．例えば，独立行政法人情報処理推進機構が発行するソフトウェア開発データ白書 2014-2015 では，日本のソフトウェア開発企業 29 社において 2000 年～2013 年にかけて実施された 3541 プロジェクトのデータ分析結果を掲載している．このデータは，多くの研究者にとって垂涎の的ともいえるが，データそのものは一般公開されていない．また，筆者らも，企業との共同研究を通してデータの貸与を受け，研究を進めてきたが（文献 14)など），共同研究の終了後はデータの返却／破棄が義務付けられ，他社のデータとの比較を行ったり，複数の企業のデータを用いたベンチマーキングを行ったりすることはできない．

以上のように，多くの企業では最新のソフトウェア開発のデータを計測・蓄積しているが，大学の研究者がそれを使った研究を行うことは困難，もしくは，大きな制約がある．さらに，昨今では，個人情報保護，及び，コンプライアンス重視のために，企業における機密保持がより厳格となり，データを社外に出すことがより困難となっている．そのため，多くの研究者は Promise Repository 等に公開されている古いデータを使って研究を進めているのが現状である．

2.2 データの匿名化

Peters ら [12]は，ソフトウェア開発組織のデータ共有を妨げている要因として，プライバシーの問題に着目し，データ匿名化手法 MORPH を提案している．この MORPH は，バグ予測研究を対象としており，ソースコードの特性値（行数，変更行数，サイクロクロマティック数，）とバグの有無に関する情報を含むデータセットにおいて，データセットに含まれるそれぞれの値を少しずつ変化させることで，各ケース（個体）の同定を妨げることを目的としている．さらに，Peters らは，バグ予測に役立たない一部の個体を削除する方法 CLIFF を提案し，MORPH と組み合わせることで，データ匿名化とバグ予測精度の確保を両立させている 13).

ただし，MORPH では，データセット中の各値に変更が加えられるものの，元データより派生したデータであることに変わりはなく，データの持ち出しや一般公開は依然として敷居が高いと考えられる．一方，本稿では，データそのものは全くランダムに生成したものであるので，データの持ち出しや公開に対して，合意が得られやすいと期待される．

表 1. ソフトウェア開発プロジェクトデータセットの例

FP	Duration	Sector	Manufactu	Schedule	…	Effort
266	4	1	1	3	…	2
1000	6	1	0	5	…	10
46	0.75	0	0	4	…	0.5
8857	17	0	0	3	…	110
396	3	0	0	4	…	5
3204	9	1	1	3	…	44
…	…	…	…	…	…	…
100	12	1	0	3	…	60

3. 提案方法

3.1 基本方針

対象とするデータセットは本稿ではソフトウェア開発プロジェクトに関するデータセットのことを指す．例となるデータセットの一部を表 1 に示す．ソフトウェア開発プロジェクトを一例に話を進めるが，前提条件さえ整っていれば応用可能であるためこの手法が幅広く活用できることが期待される．公開不可データから類似データを生成するにあたって，以下の 2 点を満たすことで類似データの生成を実現させる．

・量的変数において似た分布を持つ変数値の集合をランダムに生成する．
・変数間の関係が保たれるように変数値の入れ替えを行う．

3.2 前提条件

ソフトウェア開発プロジェクトデータセットには．2 種類の変数が存在している．一つは，規模や工数等を表す量的変数が含まれている．もう一つは，環境や条件等を表す量的変数（カテゴリ変数）である．この 2 種類はそれぞれの特徴に応じた方法を使ってデータ生成する必要がある．

一般に，プロジェクトデータセットには，量的変数として，規模尺度（ファンクションポイントやソースコード行数），開発期間，開発工数といったメトリクスが含まれる．これらの変数は経験的に確率分布が対数正規分布にある程度従うことが知られている 8)．このように，量的変数が対数正規分布で近似できることを前提条件とする．

また欠損値については現段階では考慮していないため，欠損値がないことも前提条件の一つとなる．

3.3 量的変数の生成

量的変数の生成に，ボックスミュラー法を用いる．ボックスミュラー法とは，正規分布の標準偏差σと平均値μからランダムに新たな正規分布を生成できるアルゴリズムである．式は次の通りである．

$$N = \sigma\sqrt{-2logR_1}cos2\pi R_2 + \mu$$

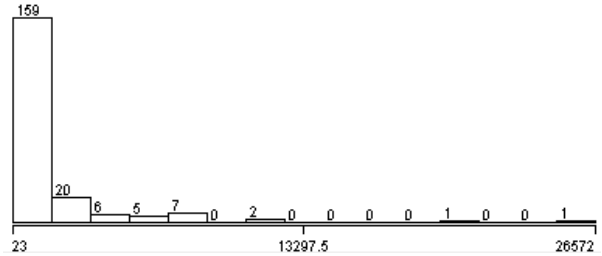

図 1.　FP の値の分布

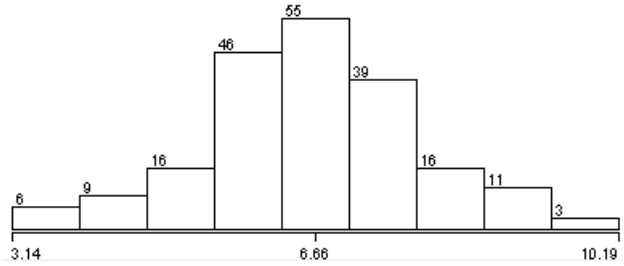

図 2.　FP の対数変換後の値の分布

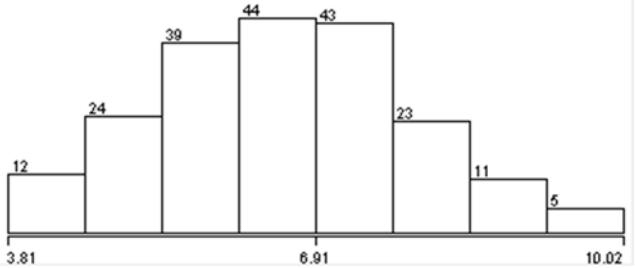

図 3 ボックスミュラー法で生成した値の分布

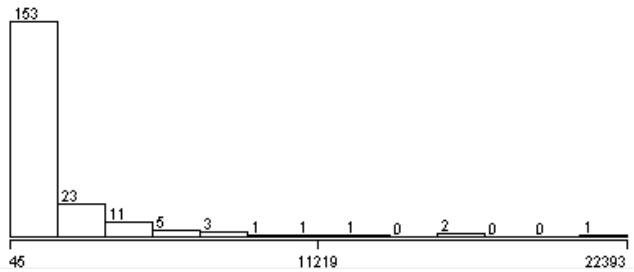

図 4　逆変換後の値の分布

ただしR_1，R_2は$(0,1]$の範囲のランダムな値である．R_1，R_2を複数個用意することで生成した値の集合の分布が正規分布に従う．

前述の通り，量的変数は対数正規分布に従う．従って，量的変数のそれぞれの値に対数変換を行うことで正規分布に変換することができる．図 1 にファンクションポイントの値の分布の例を示す．図 2 はそれを対数変換した後の値の分布である．図 2 から，対数変換後の値の分布が概ね正規分布に近いことが見て取れる．対数変換後の標準偏差と平均値を使い，ボックスミュラー法を利用すると正規分布に従う乱数の集合を生成する．図 2 の平均値と標準偏差から生成した値の集合を図 3 に示す．図 2 と図 3 は，よく似た分布となっていることが分かる．その後，対数変換の逆変換を行うことで，目的とする量的変数のデータ生成が実現できる（図 4）．

また以下の式を使うことで対数正規分布の標準偏差σ_xと平均値μ_xから直接正規分布の標準偏差σと平均値μを求めることができる．

$$\sigma^2 = \ln\{1 + (\sigma_x/\mu_x)^2\}$$
$$\mu = \ln(\mu_x) - \sigma^2/2$$

これにより元のデータの量的変数の標準偏差と平均値を入手するだけで量的変数のデータ生成が可能となる．

3.4 カテゴリ変数生成

カテゴリ変数はそれぞれの変数に対して各々の値が一定の確率で現れるとの考えのもと，ソースデータと同じ割合でそれぞれの値を生成する．

3.5 変数間の結び付け

ソースデータのプロジェクトの変数間に（表 1 行方向）は何らかの関係があるはずであり，生成後のデータもその関係を保つことが望ましい．以下はその関係を保つための手順である．

1. ソースデータの変数間の関係を順位相関として数値化し記録する．
2. 生成後のデータからランダムに一つ選びその変数と同じ変数の中からもう一つランダムに選出し，値を入れ替える．
3. 入れ替えた後の順位相関とソースデータの順位相関との差が，入れ替える前の順位相関とソースデータの順位相関との差より小さければ入れ替えは成功とし，手順 2 に戻る．もし入れ替えが失敗なら入れ替えをなかったこととして手順 2 に戻る．
4. 生成したデータの順位相関とソースデータの順位相関との差が収束したら入れ替えを終了する．

3.6 生成後の値の丸め

生成した量的変数の値は乱数を使って生成しているため，精度がばらばらな実数になっている．そのためソースデータの精度（例えば，整数値とするか，小数第何位までの値とするか等）に合わせてそれぞれの値を適当な精度に丸める．

例えば FP は量的変数であるがソースデータでは整数となっているため生成後のデータもそれに合わせて，実数とし

表 2 ソースデータの基本統計量

	平均値	標準偏差	最大値	最小値
FP	1672.697	3021.734	26572	23
Duration	11.34204	9.122927	80	0.75
Effort	132.0655	239.9036	1954	0.5

表 3 生成データの基本統計量

	平均値	標準偏差	最大値	最小値
FP	1676.995	2828.904	22393	45
Duration	11.40378	8.521531	71.2	1.32
Effort	148.5162	310.1336	2645.14	1.87

表 3 生成データからみた各変数の相対誤差

	平均値	標準偏差	最大値	最小値
FP	0.00257	0.063814	0.157271	0.956522
Duration	0.005444	0.065921	0.11	0.76
Effort	0.124565	0.292743	0.353705	2.74

て生成された値を整数になるように丸める．

4. ケーススタディ

4.1 題材

本ケーススタディでは，一般財団法人経済調査会において 2007～2011 年に収集されたソフトウェア開発プロジェクトデータを用いる．欠損値のあるプロジェクトを除いた結果，今回対象とするプロジェクト数は 202 である．本データセットには，量的変数は FP（ファンクションポイント），Duration（開発期間），Effort（開発工数）の 3 つが含まれている．またカテゴリ変数（質的変数）は，ダミー変数（2 値変数）8 つと順序変数 5 つが含まれる．2 値変数としては，Sector（発注者分類が民間企業），Manufacturing（適用業種が製造業），Electric（適用業種が電気・ガス・熱提供・水道業），ICT（適用業種が情報通信業），Transportation（適用業種が運輸業），Sales（適用業種が卸売・小売行），Finance（適用業種が金融・保険業），Government（適用業種が公務）が含まれる．また，順序変数はいずれも 5 段階の変数であり，Functionality（生産性変動要因：機能性），Reliability（生産性変動要因：信頼性），Platform（生産性変動要因：プラットフォーム），Schedule（生産性変動要因：開発スケジュール），Req.Clarity（生産性変動要因：発注要件の明確度・安定度）の 13 個が含まれている．これらの変数の定義については，文献 15)に記載されている．

4.2 量的変数

ソースデータと生成データのそれぞれの量的変数の平均値，標準偏差，最大値，最小値を表 2，3 に，それらの相対誤差を表 4 まとめる．どの変数値に注目してもある程度類似した値が生成されていることがわかる．平均値に関してはかなりの精度が確認できる．相対誤差で確かめてみると最大値，最小値に関しては大きな誤差が見受けられるが，平均値，標準偏差に関しては誤差が小さいことがうかがえる．

量的変数，Duration，Effort のソースデータ，生成データの分布を図 5，6，7，8 に示す．FP の分布については前項図 1，4 を参照されたい．FP，Duration，Effort いずれもグラフの形が類似している事が見て取れる．値，グラフの形の誤差はボックスミュラー法が正規分布に沿った値の集合を生成するものであることに対して，ソースデータの対数変換後の値の集合があくまで正規分布で近似できる分布であることが原因だと考えられる．

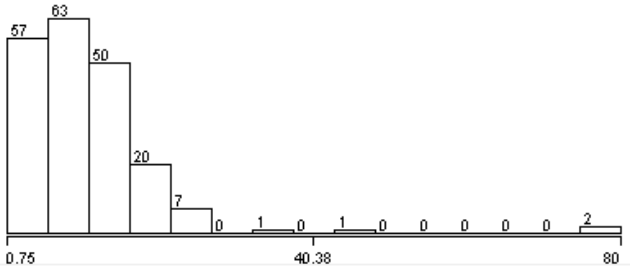

図 5 ソースデータの Duration の分布

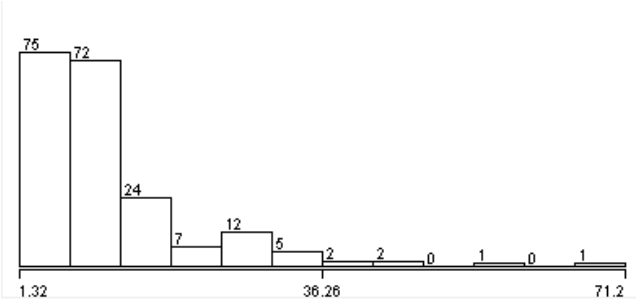

図 6 生成データの Duration の分布

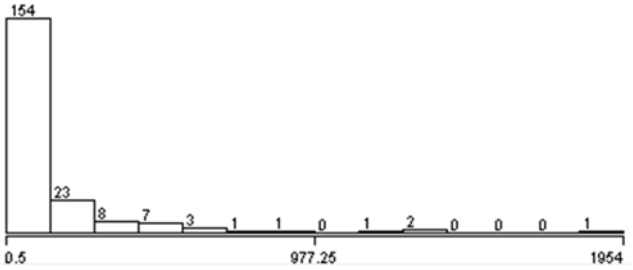

図 7 ソースデータの Effort の分布

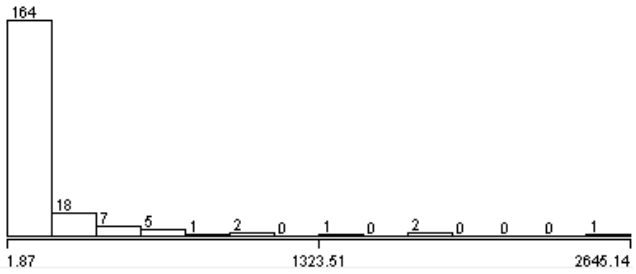

図 8 生成データの Effort の分布

4.3 順位相関

各データセットの変数間の順位相関の一部を表 5，表 6 に示す．最も差が大きいところでも 0.061 であり，全体での差の二乗和は 0.007399 である．以上のことから変数間の関係は大きく再現できていると考えられる．変数の入れ替えによる順位相関の差の収束の様子を図 9 に示す．

表 5　ソースデータの順位相関行列

	FP	Duration	Sector	Manufactu	Electric	ICT
FP	1					
Duration	0.665	1				
Sector	−0.16	−0.015	1			
Manufactu	−0.067	0.048	0.252	1		
Electric	−0.003	0.145	0.134	−0.185	1	
ICT	−0.213	−0.216	0.009	−0.162	−0.086	1
Transporta	0.044	0.022	0.091	−0.191	−0.101	−0.088
Sales	0.067	0.029	0.092	−0.128	−0.068	−0.059
Finance	0.009	0.06	0.175	−0.29	−0.154	−0.134
Governmer	0.14	−0.001	−0.842	−0.212	−0.113	−0.098
Functionali	−0.139	−0.194	−0.024	0.208	−0.087	−0.102
Reliability	−0.277	−0.161	0.003	0.089	0.072	0.006

表 6　生成データの順位相関行列

	FP	Duration	Sector	Manufactu	Electric	ICT
FP	1					
Duration	0.667	1				
Sector	−0.163	−0.013	1			
Manufactu	−0.066	0.048	0.252	1		
Electric	−0.003	0.144	0.134	−0.185	1	
ICT	−0.211	−0.216	0.009	−0.162	−0.086	1
Transporta	0.043	0.023	0.091	−0.191	−0.101	−0.088
Sales	0.067	0.028	0.092	−0.128	−0.068	−0.059
Finance	0.01	0.063	0.175	−0.261	−0.154	−0.134
Governmer	0.14	0.001	−0.842	−0.212	−0.113	−0.037
Functionali	−0.139	−0.192	−0.019	0.208	−0.086	−0.104
Reliability	−0.277	−0.159	−0.002	0.098	0.069	0.007

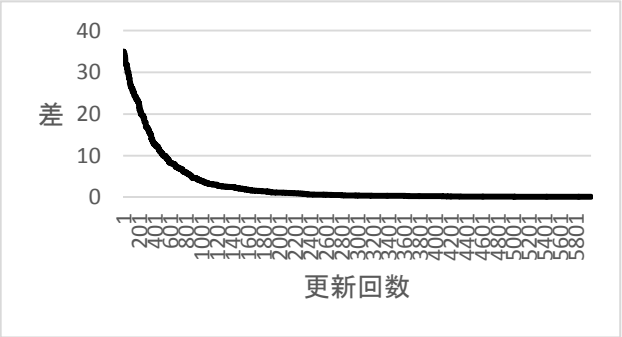

図 9.　順位相関の差の収束

4.4 Log-log 重回帰分析の比較

ソースデータ, 生成データそれぞれで log-log 重回帰分析を行い，Effort の予測モデルを比較する．重回帰分析を行った結果の一部を表 7，8 に示す．重決定係数 R^2 について注目するとどちらも 0.79 以上とかなり高い相関があることが確認できる．双方を比べてみてもその差は 0.0519 である．次に各変数の有意確率である p 値を比較する．p 値が有意水準 0.01 未満の変数（表 7，8 網掛け部）に注目すると FP, Duration, Finance, Reliability の 4 変数で一致していることが分かる．その係数に関してもある程度近い値が生成されている．このことから Effort を予測する信頼度の高い変数が生成データで再現できていることが確認できる．

表 7 ソースデータの重回帰分析

回帰統計			係数	P−値
重相関 R	0.919949	切片	−1.72293	0.001102
重決定 R2	0.846307	lnFP	0.755343	5.56E−35
補正 R2	0.833845	lnDuration	0.615655	4.65E−10
標準誤差	0.604653	Sector	0.415279	0.076732
観測数	201	Manufactu	−0.40914	0.010111
		Electric	0.257148	0.1912
		ICT	0.356103	0.08355
		Transporta	0.200759	0.284465
		Sales	0.13744	0.568654
		Finance	0.489847	0.002307
		Governmer	0.304174	0.249612
		Functionali	0.015038	0.80664
		Reliability	−0.18078	0.001453
		Platform	0.029883	0.594067
		Schedule	−0.15065	0.020209
		Req.Clarity	−0.10044	0.057645

表 8 生成データの重回帰分析

回帰統計			係数	P−値
重相関 R	0.891296	切片	−0.42161	0.488511
重決定 R2	0.794409	lnFP	0.666776	1.97E−24
補正 R2	0.77774	lnDuration	0.671824	1.77E−09
標準誤差	0.711793	Sector	0.245555	0.315153
観測数	201	Manufactu	−0.35041	0.035091
		lnElectric	0.246596	0.25572
		ICT	0.44218	0.056645
		Transporta	0.333274	0.114099
		Sales	0.211454	0.441098
		Finance	0.538533	0.002253
		Governmer	0.129797	0.636649
		Functionali	0.020935	0.769423
		Reliability	−0.22694	0.000654
		Platform	0.014423	0.825748
		Schedule	−0.26001	0.000808
		Req.Clarity	−0.17398	0.00537

5. おわりに

本稿では量的変数の標準偏差と平均値，カテゴリ変数のそれぞれの値の割合，及び，変数間の順位相関だけから元のデータの特徴を保った類似データを生成する手法を提案した．提案方法を 202 件，16 変数のプロジェクトデータセットに適用した結果, 得られた主な知見は次の通りである．

- 生成データの各量的変数は，平均値，標準偏差ともにソースデータに近い値が得られた．
- 生成データの各量的変数の値の分布は，ソースデータと類似している．
- 生成データの各変数間の順位相関係数は，ソースデータにおける値と極めて近く，変数間の関係を再現できている．
- ソースデータ，生成データのそれぞれに対し log-log 重回帰分析を適用し Effort の予測モデルを構築した結

果， 4 つの変数が両データセットにおいて有意水準 0.01 で有意となった．重決定係数 R^2 においてもどちらの予測モデルも 0.79 を上回った．

謝辞 本研究の一部は，JSPS 科研費基盤研究（C）課題番号 26330086 の補助を受けた．

参考文献

1) Albrecht, A. J., Gaffney, J.: Software function, source lines of code, and development effort prediction, IEEE Trans. on Software Engineering, Vol. 9, pp.639-648 (1983).
2) Azzeh, M.: A replicated assessment and comparison of adaptation techniques for analogy-based effort estimation. Empirical Softw Eng 17(1-2), 90-127 (2012).
3) Boehm, B.: Software Engineering Economics, Prentice-Hall, NY (1981).
4) Desharnais, J. M. : Analyse Statistique de la Productivité des Projets de Développement en Informatique à Partir de la Technique des Points de Fonction in Program de maîtrise en informatique de gestion, Université du Québec à Montréal (1988).
5) 独立行政法人情報処理推進機構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, SEC BOOKS (2014).
6) Kemerer, C. F.: An Empirical Validation of Software Cost Estimation Models, Communications of the ACM, Vol. 30, No. 5, pp. 416-429 (1987).
7) Keung, J., Kocaguneli, E., Menzies, T.: Finding conclusion stability for selecting the best effort predictor in software effort estimation. Automated Software Eng 20(4), 543-567 (2013).
8) Kitchenham, B. and Mendes, E. : Why Comparative Effort Prediction Studies May Be Invalid, Proc. 5th International Conference on Predictor Models in Software Engineering, Article No.4 (2009).
9) Kocaguneli, E., Menzies, T., Keung, J.: On the value of ensemble effort estimation. IEEE Trans Softw Eng 38(6), 1403-1416 (2012).
10) Maxwell, K.: Applied statistics for software managers. Englewood Cliffs, NJ, Prentice-Hall (2002).
11) Menzies, T., Rees-Jones, M., Krishna, R., Pape, C.: tera-promise: one of the largest repositories of se research data. "http://openscience.us/repo/index.html" (2015).
12) Peters, F. and Menzies, T.: Privacy and utility for defect prediction: experiments with MORPH, Proc. Int'l Conf. Soft. Eng., pp.189-199 (2012).
13) Peters, F., Menzies, T., Gong, L., Zhang, H.: Balancing Privacy and Utility in Cross-Company Defect Prediction, IEEE Trans. Software Eng., Vol.39, No.8, pp.1054-1068 (2013).
14) 角田 雅照, 門田 暁人, 松本 健一, ``組込みソフトウェア開発における設計関連メトリクスに基づく下流試験欠陥数の予測," SEC journal, Vol.11, No.2, pp.16-23 (2015).
15) 財団法人経済調査会: 平成 22 年度ソフトウェア開発に関する調査票（受託者向け）集計結果その 1 (2011).