

説明変数のランダムサンプリングを用いた 対数重回帰モデルによる ソフトウェア開発工数予測の試み

小西 文章[†] 内田 眞司^{††}
戸田 航史^{†††} 門田 暁人^{†††}

ソフトウェア開発における工数予測は、プロジェクト完遂に必要な人的資源やスケジュール管理を行う上で重要である。工数予測手法には、過去のソフトウェア開発プロジェクトの実績データに基づいて、工数予測モデルを構築する手法がある。我々は、工数予測モデルの中で広く用いられている対数重回帰分析によるモデル（対数重回帰モデル）に着目した。対数重回帰モデルは、オーバーフィッティングが発生する可能性があるという問題がある。そこで本稿では、オーバーフィッティングに考慮した説明変数のランダムサンプリングを用いた対数重回帰モデルによるソフトウェア開発工数予測手法を提案する。

An Attempt of Software Effort Prediction

using Multivariate Log-log Regression

by Random Sampling of Explanatory Variables

Fumiaki Konishi,[†] Shinji Uchida,^{††}
Koji Toda^{†††} and Akito Monden^{†††}

Effort estimation is important for project managers in controlling human resources in a software project. There is the method which builds an effort estimation model based on software measurement data. We focused the model by multivariate log-log regression analysis, multivariate log-log regression model that is widely used to predict effort. Multivariate log-log regression model have the annoying problem of over-fitting. In this paper, We propose the technique of performing effort prediction using multivariate log-log regression model which takes the problem of over-fitting into consideration.

1. はじめに

ソフトウェア開発プロジェクトの開発工数を予測することは、開発に必要な資源の割り当て、スケジュール管理を行う上で重要である。必要な開発工数を過不足なく定期的に予測することで、納期遅れやコストの超過といったプロジェクトの失敗を防ぐことが可能となる。そのため、様々な工数予測手法に関する研究が行われている。

工数予測手法には、過去のソフトウェア開発プロジェクトの実績データに基づいて、開発工数を予測するモデル（工数予測モデル）を構築する手法がある 8)6)。この手法は、過去のソフトウェア開発プロジェクトの実績データを用いて工数予測モデルを構築し、工数を予測したい現在進行中のプロジェクト（現行プロジェクト）で計測した特性値をモデルに代入することで開発工数を予測する手法である。工数予測モデルとは、開発工数を除くプロジェクトの特性（開発規模、開発期間など）を説明変数とし、目的変数である開発工数との関係を数学的に表わすものである。その工数予測モデルには、従来、ニューラルネット 8)、対数重回帰モデル 6)など様々な種類が提案されている。特に、対数重回帰モデルは重回帰モデルを拡張した工数予測モデルであり 12)9)、対数変換データを用いて重回帰モデルを構築する。対数変換を行いデータセットの分布を正規分布に近づけることにより、予測精度の向上が期待される。

対数重回帰モデルは、最小二乗法によりモデル構築に使われるデータ（以降、モデル構築用データ）に最も適合するようにモデルが構築される。モデル構築用データと予測対象のデータ（以降、評価用データ）が類似している場合、モデル構築用データで構築された回帰モデルは評価用データに対しても高い精度で予測できる。ただし一般的に、評価用データに対して、モデル構築用データには多かれ少なかれ偏りが生じる。データの偏りとは、例えば開発工数が少ないプロジェクトが集中して集まっていることを指す。このようなデータによって構築されたモデルは、開発工数が少ないプロジェクトに対しての精度は高いが、開発工数が多いプロジェクトに対しての精度は低くなるといったオーバーフィッティングが発生する。オーバーフィッティングとは、構築したモデルがモデル構築用データに過度に適合し、評価用データに対して予測精度が低下してしまうことである。オーバーフィッティングの問題に対して、標本の復元抽出（ブートストラップ法）による集団学習の適用例がある 11)5)。集団学習を用いるとデータの恣意性に左右されにくくなり、予測精度が安定することが報告されている。

[†]奈良工業高等専門学校 専攻科 電子情報工学専攻

^{††}Advanced Course of Electronic and Information Engineering, Nara National College of Technology

^{†††}奈良工業高等専門学校 情報工学科

Information Engineering, Nara National College of Technology

^{††††}奈良先端科学技術大学院大学 情報科学研究科

Graduate School of Information Science, Nara Institute of Science and Technology

本稿では、ソフトウェア開発工数予測において、オーバーフィッティングを軽減するためランダムフォレスト法2)の特長である説明変数のランダムサンプリングを用いた工数予測手法を提案する。ランダムサンプリングとは無作為標本抽出のことで、調査対象をある母集団（本稿ではモデル構築データの全説明変数）からランダムにサンプリングすることである。提案手法によりオーバーフィッティングを軽減することで、予測精度の向上を目指す。

以降、2章でソフトウェア開発工数予測について説明し、3章で提案手法について説明する。4章では評価実験の方法と結果を示し、5章では考察について述べる。最後に6章で本稿の結論を述べる。

2. 原理

2.1 対数重回帰分析

本稿では、工数予測モデルを構築する手法の中でも、一般的に用いられている対数重回帰分析について着目する。対数重回帰分析は、多変数解析の一手法である重回帰分析を拡張した手法である。重回帰分析は、予測対象の変数（目的変数）と、目的変数に影響を与える複数の変数（説明変数）との関係を表した回帰式を作成する。回帰式中の各回帰係数は、予測値と実際の値（実測値）との誤差の二乗和が最小になるように決定される。作成された回帰式に、説明変数を与えることで、目的変数が得られる。対数重回帰分析は、対数変換したデータから回帰式を作成する手法である。作成された回帰式に対して、対数変換した説明変数を与え、得られた値を逆対数変換することで予測値が得られる。この対数重回帰分析は、ソフトウェア開発における実績データのような等分散ではないデータセットから予測を行う場合に広く用いられている。対数変換を行いデータセットの分布を正規分布に近づけることにより、予測精度の向上が期待される。

ここで、対数重回帰式の一般的な形を式(1)に示す。説明変数の数を n とすると、式中の A_i ($i=0\sim n$)は回帰係数、 X_i ($i=1\sim n$)は説明変数、 E は目的変数である。

$$\text{Log}(E) = A_0 + A_1 \times \text{Log}(X_1) + A_2 \times \text{Log}(X_2) + \dots + A_n \times \text{Log}(X_n) \quad (1)$$

2.2 オーバーフィッティング

我々は、説明変数のランダムサンプリングにより複数の対数重回帰モデルを構築することで、オーバーフィッティングの軽減を目指す。対数重回帰モデルでは、予測値と実測値の残差の二乗和が最も小さくなるように、すなわちモデル構築用データに最も適合するようにモデルが構築される。ただし、収集されるモデル構築用データには

偏りが生じる場合がある。そのような偏ったデータによって構築された回帰モデルは、オーバーフィッティングが発生する可能性がある。オーバーフィッティングは過学習とも呼ばれ、回帰モデルがモデル構築用データに過度に適合し、評価用データに対して予測精度が低下してしまうことである。つまり、構築された対数重回帰モデルが、未知データに対しては適合しない、汎化ではない状態に陥る可能性があるという問題である。

オーバーフィッティングの一例を示す。表1は、あるモデル構築用データで“全説明変数”を用いて対数重回帰モデル(モデルA)を構築したときの結果と、“全説明変数の半分(目的変数との相関の高い変数を選択)”を用いて対数重回帰モデル(モデルB)を構築したときの結果である。評価指標として用いている絶対残差(または誤差)平均は小さいほど測定の精度が高いと評価できる。表1の“モデル構築用データ”の項目では、モデル構築用データを予測(自己予測)した時の予測値と実測値の絶対残差平均を示す。“評価用データ”の項目では、評価用データを予測した時の予測値と実測値の絶対誤差平均を示す。表1より、モデルAは、自己予測した時の精度に比べて、評価用データを予測した時の精度が著しく低下している。よって、このモデルは汎化ではなくオーバーフィッティングしていると言える。一方、モデルBは、モデルAと比べて自己予測した場合の精度は低いが、評価用データを予測した時の精度は高い。モデルBは、モデル構築用データだけではなく、評価用データに対しても安定して予測が行える、信頼性の高いモデルであると言える。

表1 オーバーフィッティングの一例

	モデルA		モデルB	
	モデル構築用 データ (自己予測)	評価用 データ	モデル構築用 データ (自己予測)	評価用 データ
絶対残差(または誤差)平均	1644.7	3132.5	2041.2	2293.0

3. 提案手法

3.1 概要

本稿ではオーバーフィッティングを軽減するために、ランダムフォレスト法の特長である説明変数のランダムサンプリングを用いる。ランダムフォレスト法において説明変数のランダムサンプリングを行う意味の1つとしては、標本の復元抽出によって得られたサブセットで構築されたモデル間の差異は限定的で、説明変数の選択において

は非常に類似したものになることが挙げられる。個々のモデルがお互いに非常に類似している場合には、複数のモデルを組み合わせて平均化を行っても際立った精度の改善が見られない。そこで、2つのランダム化プロセスにより（標本の復元抽出 + 説明変数のランダムサンプリング）モデル間の差異を明確にする。

3.2 提案モデル

提案手法は、対数重回帰分析を用いた集団学習である。提案手法のアルゴリズムを以下に示す。また、提案手法による工数予測の流れを図1に示す。

- 手順 1.** モデル構築用データが N 件の場合、復元抽出により N 件で構成されるサブセットを B 組作成する。ここで、手順2で構築する対数重回帰モデルに常に（固定で）使用する説明変数 F を全説明変数の M から決定する。
- 手順 2.** 作成したサブセットを用いて対数重回帰モデルを構築する。ただし、固定で使用する説明変数以外の説明変数 ($M-F$) から R の割合だけモデル構築に使用する説明変数 m をランダムサンプリングする。説明変数 F と m を対数重回帰モデル構築に用いる。
- 手順 3.** 手順2の作業を B 回繰り返し、得られた B 個の対数重回帰モデルに対して評価用データを適用し、その平均値を予測値とする

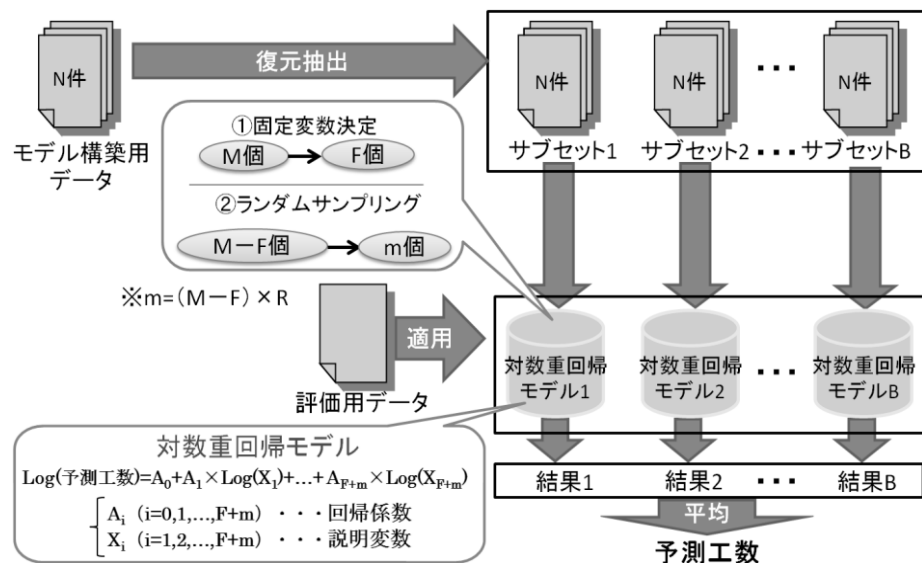


図1 提案手法による工数予測の流れ

提案手法におけるパラメータ (F, R, B) について説明する。パラメータ F は、各サブセットから構築される対数重回帰モデルに必ず使用する説明変数の個数である。工数に影響を大きく与える変数（例えば、工数との相関が著しく高い変数など）が F として選択される。なお、 F はランダムフォレスト法には存在しない。ランダムフォレスト法は構築する各決定木のノードごとに説明変数のランダムサンプリングを行っているため、決定木全体としては全変数を使用している。提案手法は各対数重回帰モデル全体に使用する説明変数としてランダムサンプリングを行っている。従って工数に影響を大きく与える変数がランダムサンプリングによって選ばれなかった場合、予測精度が低下する恐れがあるため、固定変数として F を決定する。パラメータ R は、各対数重回帰モデル構築に使用する説明変数を残りの説明変数 ($M-F$) 個からランダムサンプリングする割合である。実際にランダムサンプリングする説明変数の個数 m は式(2)により与えられる。例えば、 R が2分の1なら、 $(M-F)/2$ 個の説明変数をランダムサンプリングする。なお、割り切れない場合は切り捨てとする。パラメータ B は、復元抽出によって作成するサブセットの個数すなわち対数重回帰モデルを構築する個数である。 B を大きくするほど予測結果が正規化され、精度が安定すると考えられる。

$$m = (M - F) \times R \quad (2)$$

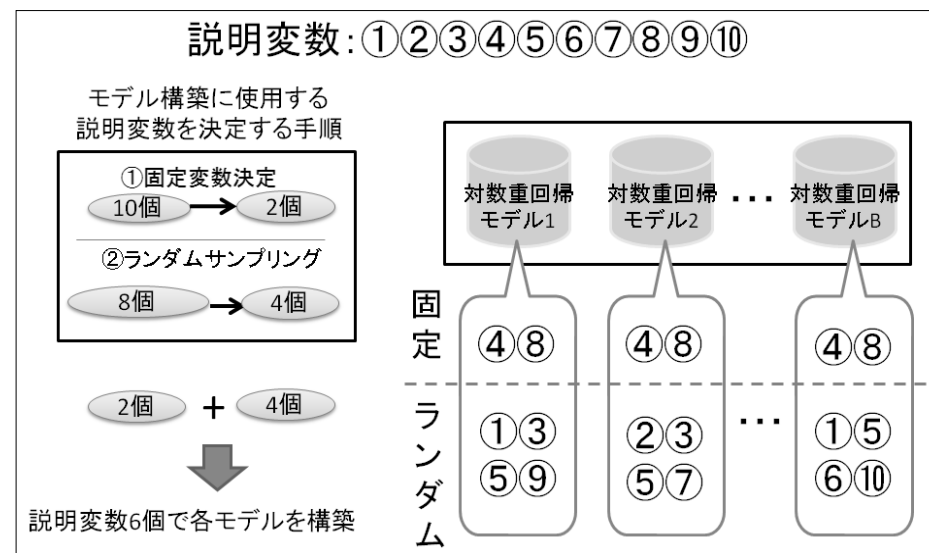


図2 提案手法における各対数重回帰モデルで使用する説明変数の例

提案手法において、モデル構築に使用する説明変数の例を図2に示す。M=10, F=2, R=2分の1とする。式(2)より、ランダムサンプリングする説明変数の個数 $m=4$ となる。よって例では、各対数重回帰モデルは説明変数6個(F+m)で構築される。パラメータFの変数は目的変数との相関が高い説明変数が選択され、全モデルで使用する。パラメータRの変数は各モデル構築時にランダムサンプリングにより選択される。

4. 評価実験

4.1 概要

評価実験の目的は、提案手法と対数重回帰分析（以降、従来手法）の予測精度を比較し、提案手法の有効性を確認することである。実験はテストサンプル法で行った。提案手法におけるパラメータBは、予備実験によって得た、予測精度が安定する50を採用した。パラメータFとRについては、7パターンの組合せで実験を行った。7パターンの組合せとは、(F,R) = (全変数,0), (0,3分の1), (0,3分の2), (1,3分の1), (1,3分の2), (2,3分の1), (2,3分の2)である。(全変数,0)とは、説明変数をランダムサンプリングしないで各対数重回帰モデルで全変数を用いてモデルを構築することを示す。その他の6パターンはF=0,1,2とR=3分の1,3分の2の組合せである。まずFとして0~2を採用したのは実験で用いたデータセットに、工数に影響を大きく与えると考えられる（工数との相関が強い）変数が2つあったため、それらを固定でモデルに導入した場合の精度を評価するためである。R=3分の1を採用したのは、提案手法の基となったランダムフォレスト法において、ランダムサンプリングする説明変数の割合の標準値として3分の1が使われていたためである。さらに、全変数とR=3分の1の間であるR=3分の2も評価する。評価実験では、工数予測を行うタイミングとして、開発対象ソフトウェアのファンクションポイント数（FP数）の計測が終了した時点、つまり、基本設計の完了時を想定している。また、開発期間は、開発初期に予め決定されていることを想定している。

4.2 データセット

本実験では、ソフトウェア開発企業の実績データであるISBSGデータセット3)とKitchenhamデータセット4)を用いた。それらには欠損値や工数予測に用いるのに不適切な説明変数が含まれていたため、それぞれのサブセットを実験では用いた。

ISBSGデータセットは3026件のプロジェクトについて、それぞれ99種類の変数が記録されている。そのサブセットとして欠損を含まないプロジェクト数232件、特性8種類のデータを作成し、実験に用いた10)。以降、ISBSGデータセットとはこのサブセットを指す。ISBSGデータセットの詳細を表2に示す。表2において、総工数を目的変数とし、それ以下のFP数から工数算入業務までを説明変数とした。

Kitchenhamデータセットには145件のプロジェクトについて、それぞれ9種類の変数が記録されている。そのサブセットとして欠損を含まないプロジェクト数135件、特性5種類のデータを作成し、実験に用いた。以降、Kitchenhamデータセットとはこのサブセットを指す。Kitchenhamデータセットの詳細を表3に示す。表3において、総工数を目的変数とし、それ以下のFP数から顧客コードまでを説明変数とした。

表2 ISBSGデータセットに含まれるプロジェクトの特性

特性名	特性の種類	含まれる特性名	特性の種類数
総工数	量的変数	数値データ	—————
FP数	量的変数	数値データ	—————
計画工数	量的変数	数値データ	—————
要件定義工数	量的変数	数値データ	—————
開発タイプ	質的変数	新規開発, 派生開発, 再開発	3
開発言語	質的変数	Java, C, COBOL等	24
工数計測手法	質的変数	実作業時間集計等	7
工数算入業務	質的変数	収集レベル1~4	4

表3 Kitchenhamデータセットに含まれるプロジェクトの特性

特性名	特性の種類	含まれる特性値	特性の種類数
総工数	量的変数	数値データ	—————
FP数	量的変数	数値データ	—————
開発期間	量的変数	数値データ	—————
開発タイプ	質的変数	開発, 適応保守, 是正保守等	6
顧客コード	質的変数	1~6	6

これら2つのデータセットの説明変数には量的変数と質的変数が混在している。このうち質的変数は、それぞれ0または1の値をとる複数の2値変数（ダミー変数）に変換したものを実験に用いた。この変換を行った結果、ISBSGデータセットの変数は38個、Kitchenhamデータセットの変数は13個となった。

4.3 評価指標

評価実験における予測精度の評価指標として、絶対誤差平均 (MMAE) と相対誤差平均 (MMRE) の 2 種類を用いた。それぞれの評価指標は次の式 (3) ~ (4) により算出される。各評価値は低いほうが、予測精度が高いことを示す。ここで、 Z 件のプロジェクトについて、その実測値と予測値を E_i, \hat{E}_i ($i = 1 \sim Z$) と定義する。そして、

$$A_i = |\hat{E}_i - E_i|, R_i = \frac{|\hat{E}_i - E_i|}{E_i} \text{ とおく.}$$

$$\text{MMAE} = \frac{\sum_{i=1}^M A_i}{Z} \quad (3)$$

$$\text{MMRE} = \frac{\sum_{i=1}^M R_i}{Z} \quad (4)$$

4.4 実験手順

実験はテストサンプル法により行う。実験手順を以下に示す。

- 手順 1. データセットに含まれるデータ L 件をランダムに 2 分割し、一方をモデル構築用データ $L/2$ 件、他方を評価用データ $L/2$ 件とする。評価用データのプロジェクトは総工数が未知と仮定され、工数予測の対象となる。
- 手順 2. モデル構築用データから、提案手法と従来手法でモデルを構築する。
- 手順 3. 構築した予測モデルを用いて評価用データの開発工数を予測し、各評価基準を求める。
- 手順 4. 結果の妥当性を確保するために、手順 1~手順 3 を 10 回繰り返し、各評価基準について平均値と標準偏差を算出する。

4.5 実験結果

ISBSG データセットの実験結果を箱ひげ図で図 3 と図 4 に示す。MMAE, MMRE とともに従来手法と比較して、7 パターン中 6 パターンで提案手法の予測精度が高くなった。予測精度が低くなった提案手法(0,3 分の 1)では、ランダムサンプリングにより工数に影響を大きく与える (工数との相関が強い) 変数が選択されなかったために予測精度が低下したと考えられる。

Kitchenham データセットの実験結果を箱ひげ図で図 5 と図 6 に示す。従来手法と比較して、7 パターン中 6 パターンで提案手法の予測精度が低くなっている。提案手法 (全変数,0) のみ従来手法と同等の予測精度を得た。提案手法 (全変数,0) 以外では、2 個の変数を固定 (F=2) したとき精度が高い。Kitchenham データセットに対しては、提案手法は有効ではないことが明らかとなった。理由については次章で考察する。

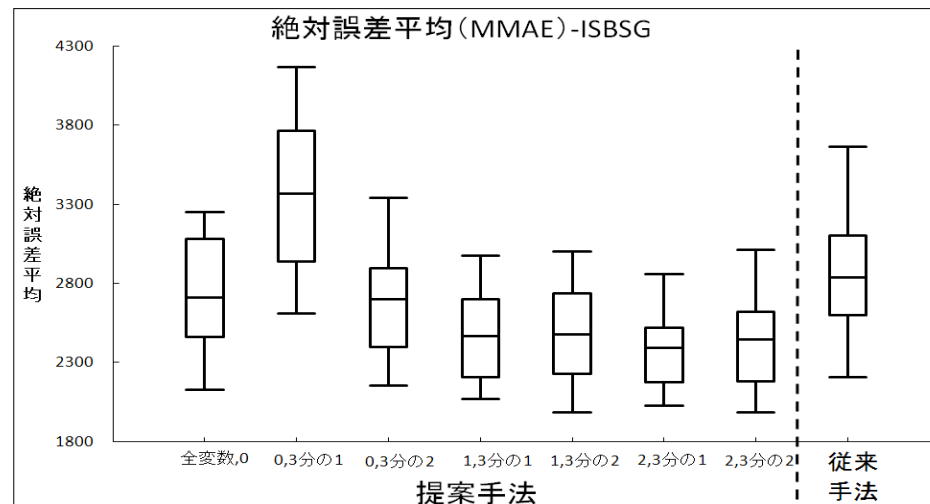


図 3 評価実験の絶対誤差平均の箱ひげ図 (ISBSG)

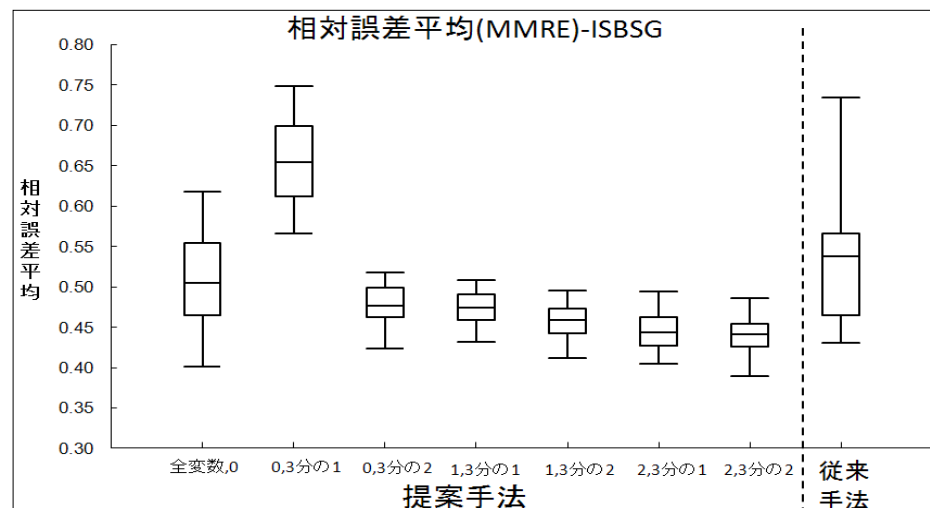


図 4 評価実験の相対誤差平均の箱ひげ図 (ISBSG)

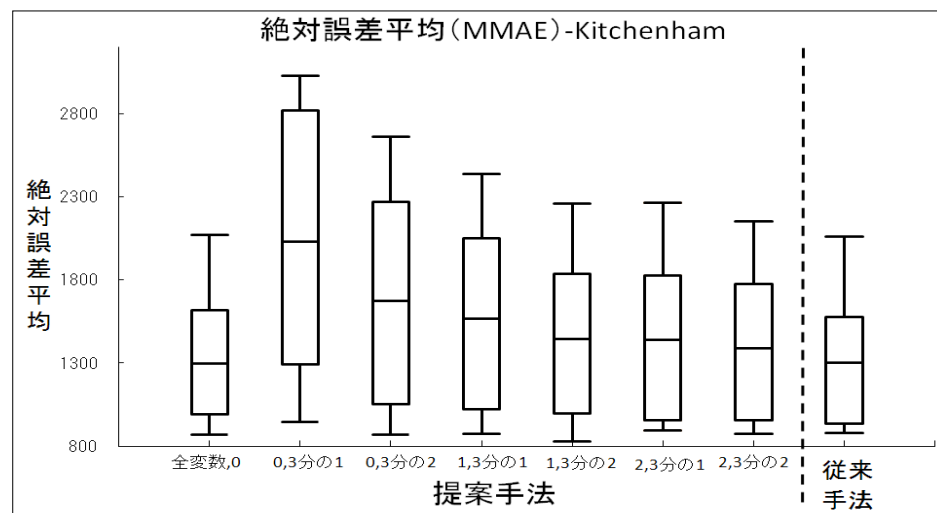


図5 評価実験の絶対誤差平均の箱ひげ図 (Kitchenham)

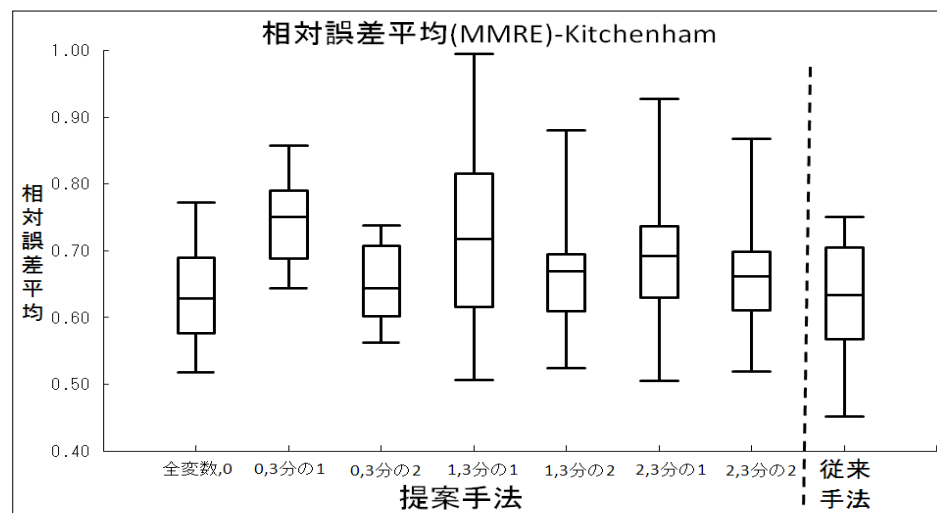


図6 評価実験の相対誤差平均の箱ひげ図 (Kitchenham)

5. 考察

5.1 実験結果について

ISBSG データセットにおいて予測精度が向上した. 一方で Kitchenham データセットに対しては予測精度が向上しなかった. その理由について考察する.

表4に各データセットの生産性 (=FP 数/総工数) の平均値, 標準偏差, 変動係数を示す. 変動係数とは標準偏差を平均で割ったもので相対的なバラツキを表す. Kitchenham データセットと比較して ISBSG データセットの変動係数が大きく生産性にばらつき(外れ値)がある. 外れ値近辺のデータによりモデルを構築した場合オーバーフィッティングを発生しやすくなる. 従って提案手法が ISBSG データセットに対して効果があったと考えられる.

表4 各データセットの生産性について

	生産性	
	ISBSG データセット	Kitchenham データセット
平均	0.725	0.758
標準偏差	0.119	0.104
変動係数	0.165	0.137

Kitchenham データセットに対して提案手法では全変数を用いた構築モデルの予測精度が高い. また変数選択を行った構築モデルの予測精度は低い. Kitchenham データセットは説明変数の数が少なく(12 個)ほぼ全ての説明変数が開発工数との関係が高いため, ランダムサンプリングの効果が発揮できなかったと考えられる.

5.2 実験結果とオーバーフィッティングの関係について

提案手法の予測精度とオーバーフィッティングとの関係について ISBSG データセットの実験結果を用いて考察する. まずオーバーフィッティングの大きさ (D) を式(5)で定義する. 式(5)において, TE は評価用データに対する予測誤差, FE はモデル構築用データに対する予測 (自己予測) 残差とする. オーバーフィッティングとは『モデル構築用データに過度に適合し, 評価用データに対して予測精度が低下してしまうこと』である. 言い換えれば, ある構築したモデルにおいて, モデル構築用データに対する予測 (自己予測) 残差は少ないが, 評価用データに対する予測誤差は大きくなることである. よって, D の値が大きいほど, オーバーフィッティングの影響を受けていると言える.

$$D = TE - FE \quad (5)$$

ここで、評価実験における 10 回の従来手法のオーバーフィッティングの大きさ(D)と、従来手法に対する提案手法の向上値を表 5 と表 6 に示す。なお提案手法は MMAE,MMRE で最も向上値の高かった提案手法 (2,3 分の 1) とする。表 5 および表 6 中の“相関係数”とは、オーバーフィッティングの大きさ D と提案手法 (2,3 分の 1) の向上値との相関を示す。いずれの指標でも相関は高い(MMAE=0.723, MMRE=0.914)。従って、従来手法におけるオーバーフィッティングの影響が大きいほど、提案手法の向上値が大きく、効果的であると言える。

6. おわりに

本研究では、ソフトウェア開発工数予測において、オーバーフィッティングの問題の解消を目的として説明変数のランダムサンプリングを用いた手法を提案した。提案手法では、説明変数のランダムサンプリングを対数重回帰分析に適用し、複数の対数重回帰モデルを構築する。それらのモデルから得られた値の平均値を予測値とすることで工数予測を行う。

ISBSG データセットと Kitchenham データセットを用いて評価実験を行った結果、ISBSG データセットでは提案手法の予測精度が向上した。

今後は、本稿の実験結果の信頼性を高めるためにさらに多くのデータセットで同様の実験を行うこと、提案手法のパラメータ決定法を確立しその評価をすることが必要である。

参考文献

- 1) Boehm,B.W.: Software engineering economics, Prentice Hall, NewJersey(1981)
- 2) Breiman,L.: Random Forests, Machine Learning, Vol.45, pp.5-32 (2001)
- 3) International Software Benchmarking Standards Group: ISBSG Estimating, Benchmarking and research suite release 9 (2004)
- 4) Kitchenham,B., Pfleeger,S., McColl,B. and Eagan,S.: An empirical study of maintenance and development estimation accuracy,” Journal of Systems and Software, Vol. 64, No. 1, pp.57-77 (2002)
- 5) Mittas,N., Athanasiades,M. and Angelis,L.: Improving analogy-based software cost estimation by a resampling method,” Information and Software Technology, Vol.50, No.3, pp.221-230(2008)
- 6) 門田 暁人, 小林 健一: 線形重回帰モデルを用いたソフトウェア開発工数予測における対数変換の効果, コンピュータソフトウェア, Vol.27, No.4, pp.234-239 (2010)

表 5 オーバーフィッティングの大きさと提案手法の向上値との関係 (MMAE)

MMAE 実験回数	従来手法			提案手法(2,3 分の 1)の 向上値
	FE	TE	D(=TE-FE)	
1 回目	1697.6	3012.7	1315.1	152.4
2 回目	1644.7	3132.5	1487.8	631.0
3 回目	2366.0	2252.6	-113.4	110.0
4 回目	1607.8	3249.9	1642.1	429.6
5 回目	2221.5	2586.2	364.8	58.5
6 回目	2045.2	2631.0	585.7	606.9
7 回目	2301.3	2203.8	-97.6	97.9
8 回目	1660.6	3662.9	2002.3	1384.3
9 回目	1902.2	2972.2	1070.0	704.0
10 回目	2002.7	2644.2	641.5	271.7
平均	1945.0	2834.8	889.8	444.6
相関係数				0.723

表 6 オーバーフィッティングの大きさと提案手法の向上値との関係 (MMRE)

MMRE 実験回数	従来手法			提案手法(2,3 分の 1)の 向上値
	FE	TE	D(=TE-FE)	
1 回目	0.350	0.524	0.174	0.029
2 回目	0.331	0.535	0.204	0.105
3 回目	0.399	0.485	0.086	0.018
4 回目	0.390	0.459	0.069	0.032
5 回目	0.434	0.430	-0.004	0.026
6 回目	0.369	0.627	0.257	0.210
7 回目	0.383	0.458	0.075	0.004
8 回目	0.308	0.735	0.427	0.294
9 回目	0.352	0.568	0.215	0.102
10 回目	0.313	0.561	0.167	0.122
平均	0.363	0.538	0.167	0.094
相関係数				0.914

- 7) Shulani,S., Boehm,B. and Steece,B.: Bayesian analysis of empirical software engineering cost models, IEEE Transactions on Software Engineering, Vol.25, No.4, pp.573-583(1999)
- 8) Srinivasan,K. and Fisher,D.: Machine learning approaches to estimating software development effort, IEEE Transactions on Software Engineering, Vol.21, No.2, pp.126-137 (1995)
- 9) 戸田航史, 角田雅照, 門田暁人, 松本健一: 工数見積もりモデルで予測できないソフトウェアプロジェクトの特徴分析, 電子情報通信学会技術報告, ソフトウェアサイエンス研究会, SS2005-74, pp.67-72 (2005).
- 10) 戸田航史, 門田暁人, 松本健一: ソフトウェア開発工数予測のためのフィットデータ選定方法, 情報処理学会論文誌, Vol.50, No.11, pp.2699-2709(2009)
- 11) 角田雅照, 門田暁人, 松本健一: 予測モデル構築・評価におけるリサンプリング法の活用, ウィンターワークショップ 2011・イン・修善寺, pp.105-106 (2011)
- 12) 渡邊瑞穂, 柿元健, 戸田航史, 門田暁人, 松本健一: ソフトウェア開発工数予測における特異プロジェクト除去の効果, 日本ソフトウェア科学会第 24 回大会論文集 CD-ROM(講演番号 7B-2)(2007)