

ソフトウェア開発工数見積もりにおける外れ値の実験的評価

小野 健一
奈良先端科学技術大学院大学
ono.kenichi.ob4@is.naist.jp

門田 暁人
岡山大学
monden@okayama-u.ac.jp

角田 雅照
近畿大学
tsunoda@info.kindai.ac.jp

松本 健一
奈良先端科学技術大学院大学
matumoto@is.naist.jp

要旨

大規模なソフトウェア開発においてプロジェクト管理は必要不可欠であり、そのためにはソフトウェア開発工数を正確に見積もることが重要となる。工数見積もりにおいて外れ値は除去されることが多い。ただし、外れ値がどの程度見積もり精度に影響するのかは明らかではない。そこで本研究では、データセットに外れ値を実験的に追加し、外れ値の見積もり精度に対する影響を分析した。外れ値の割合、外れ値の程度、外れ値の存在する変数、外れ値の存在するデータセットの 4 つを変化させて見積もり精度を評価した結果、開発規模に外れ値が含まれていても、外れ値の割合が 10%で、外れ値の程度が 100%の場合、あまり見積もり精度が低下しないことなどがわかった。

1. はじめに

近年、ソフトウェアはより高機能が求められており、その要求にこたえるためにソフトウェアの規模が大規模化している。大規模なソフトウェアを開発するためには、多くの工数(コスト)が必要となる。このようなソフトウェアを開発するプロジェクトでは、スケジュールやコストに関する管理を行わずにプロジェクトを成功させることは困難である。よって、大規模な開発プロジェクトにおいてプロジェクトの失敗を避けるためには、スケジュールやコストの管理が必要不可欠となる。ソフトウェア開発工数の見積もりは、それらの管理の基礎となるものである。

スケジュールやコスト管理を正確に行うためには、ソフトウェア開発の工数を高い精度で見積もることは非常に重要であり、そのため、これまでさまざまな定量的工数見積もり手法が提案されてきた[1][14][17]。定量的工数見積もり方法として、重回帰分析やプロジェクト類似性に基

づく工数見積もり[13]などがあげられる。定量的に工数を見積もるためには、過去のプロジェクトで収集されたデータを見積もりの根拠データとし、見積もり対象のプロジェクトにおいて既知のデータ、例えば開発言語や開発規模などを用いて、開発工数を見積もる。

ただし定量的工数見積もりを行う場合に、見積もりの根拠データなどに外れ値と呼ばれるデータが存在する可能性があり、それにより、工数見積もりの精度が低下する可能性がある。例えば、非常に大量の作業手戻りが発生したプロジェクトの場合、同規模の他のプロジェクトに比べると、開発工数が極端に大きくなる。また、データの収集や集計時にミスが発生したプロジェクトの場合、実際のデータと異なるものが記録される。

外れ値により工数見積もりの精度が低下することを防ぐため、外れ値除去法が適用される場合がある。外れ値除去法とは、見積もりの根拠データに含まれるプロジェクトが外れ値であるか検証し、外れ値と判断されたものをデータから除去する手法である。例えば、重回帰分析によりソフトウェア開発工数を見積もる場合、Cook の距離を用いて外れ値を除去すること多い([7]など)。類似性に基づく見積もり方法に適した外れ値除去法もいくつか提案されている[6]。

本研究では、定量的にソフトウェア開発工数を見積もる場合に、データに含まれる外れ値がどの程度影響を与えるのかを分析する。そのために、本研究ではデータセットに外れ値を実験的に追加することを行う。具体的には、目的変数である工数と、説明変数として最も重要な開発工数に外れ値を含める。実験では以下の 2 つのパラメータを変更した場合の見積もり精度の変化を確かめる。

- 外れ値が含まれる割合
- 外れ値の程度

外れ値が含まれる割合とは、例えばこのパラメータを

10%とした場合、根拠データ 100 件のうち 10 件に外れ値を含めることである。外れ値の程度とは、例えばこのパラメータを 100%とした場合、工数が 100 人時のデータを 200 人時とすることにより外れ値を含めることである。これにより、工数見積もりを行う際、各変数における外れ値の影響をどの程度考慮すべきかの参考になると考える。

さらに、データに外れ値が含まれているパターンについても変更して分析する。具体的には以下のパターンでデータに外れ値を含める。

- 見積もり根拠データ:外れ値あり, 見積もり対象データ:外れ値なし
- 見積もり根拠データ:外れ値あり, 見積もり対象データ:外れ値あり
- 見積もり根拠データ:外れ値なし, 見積もり対象データ:外れ値なし

外れ値は見積もり根拠データだけではなく、見積もり対象プロジェクトにも含まれている場合がある。一般に、見積もり対象プロジェクトの外れ値を除去することは難しいため、これまでで見積もり対象プロジェクトに外れ値が含まれている場合の見積もり精度への影響は考慮されていない。ただし、実際に定量的見積もりを行う場合、見積もり対象データに外れ値が含まれることがあるため、これを考慮せずに見積もりモデルを評価することは、モデルの見積もり精度を過大に評価している可能性がある。この分析により、より現実的な見積もり精度を評価するためにはどうすべきかを考察することができると考える。

実験では、定量的工数見積もり方法として重回帰分析とプロジェクト類似性に基づく工数見積もりを採用する。

2. 定量的工数見積もり

本研究では、定量的工数見積もり方法として用いられる重回帰分析と類似性に基づく工数見積もりにおける、外れ値の影響を評価する。以降では、それぞれの見積もり方法について説明する。

2.1. 重回帰分析に基づく工数見積もり

重回帰分析による見積もりは、ソフトウェア開発工数を定量的に見積もる際に広く用いられる方法である。重回帰分析では、過去のプロジェクトにおいて記録されたデータを見積もりの根拠データとして、最小二乗法により見積もりモデルが 1 つ構築される。開発工数を y , 開発規模などの説明変数を x_1, x_2, \dots, x_k とすると、重回帰分析による積モデルは以下のようになる。

表 1 Analogy 法による工数見積もりにおいて用いるデータセット

Table 1 A dataset used on analogy based effort estimation

	変数 1	変数 2	...	変数 j	...	変数 l
p_1	m_{11}	m_{12}	...	m_{1j}	...	m_{1l}
p_2	m_{21}	m_{22}	...	m_{2j}	...	m_{2l}
...
p_i	m_{i1}	m_{i2}	...	m_{ij}	...	m_{il}
...
p_k	m_{k1}	m_{k2}	...	m_{kj}	...	m_{kl}

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k + \varepsilon \quad (1)$$

ここで、 β_0 は回帰定数、 $\beta_1, \beta_2, \dots, \beta_k$ は偏回帰係数、 ε は誤差項である。重回帰分析により見積もりモデルを構築する場合、経験的に説明変数の 5 から 10 倍のプロジェクト数が必要であるといわれている。

重回帰分析を用いてソフトウェア開発工数を見積もる場合、モデルの構築前に比例尺度の説明変数、目的変数に対し、対数変換が適用される場合がある。対数変換を適用した重回帰分析はソフトウェア開発のデータの特徴を表すのに適していることが指摘されているため[10]、本研究でも対数変換を適用した重回帰分析により見積もりモデルを構築する。

2.2. プロジェクト類似性に基づく工数見積もり

プロジェクト類似性に基づく工数見積もり (Analogy 法) は、事例ベース推論 (Case Based Reasoning; CBR) に基づいた見積もり方法である。CBR は人工知能の分野で研究されてきた問題解決のための方法であり、Shepperd ら[13]が CBR をソフトウェアプロジェクトの開発工数見積もりに適用することを提案した。CBR では、蓄積された過去の事例の中から、問題を解決したい現在の事例と類似したもの抽出し、その解決方法を適用する。CBR の基礎となる考え方は「類似した問題は類似した解決方法を持っている」というものである。Analogy 法では、類似するプロジェクト (開発規模や対象業種などの特徴が互いに似たプロジェクト) は、工数も互いに似た値を取るであろうと仮定し、類似プロジェクトの工数に基づいて対象プロジェクトの工数を見積もる。

Analogy 法による工数見積もりでは、表 1 に示す $k \times l$ 行列で表されるデータセットを入力として用いる。図中、 p_i は i 番目のプロジェクトを表し、 m_{ij} はプロジェクト p_i の j 番目の変数を表す。すなわち、行がプロジェクト、列が変数を表している。ここで p_a を見積もり対象のプロジェクト、

\hat{m}_{ab} を m_{ab} の見積もり値とする。Analogy 法による工数見積もりは、以下の手順に従って行われる。

- 互いに異なる各変数の値域を統一する。本研究では変数 m_{ij} の正規化された値 m'_{ij} を計算する際、以下の式を用いて、変数の値域を $[0, 1]$ に揃える。

$$m'_{ij} = \frac{m_{ij} - \min(m_j)}{\max(m_j) - \min(m_j)} \quad (2)$$

ここで、 $\max(m_j)$ と $\min(m_j)$ はそれぞれ j 番目の変数の最大値、最小値を表す。この計算方法は、値域を変換する際に広く用いられる方法の 1 つである [15]。

- 見積もり対象プロジェクト p_a と他のプロジェクト p_i との類似度 $\text{sim}(p_a, p_i)$ を求める。 p_a と p_i が持つ変数を要素とする 2 つのベクトルを作成し、ベクトルのなす角のコサインを用いて類似度を計算する。見積もり対象プロジェクト p_a と他のプロジェクト p_i との類似度 $\text{sim}(p_a, p_i)$ を次式によって計算する。

$$\text{sim}(p_a, p_i) = \frac{\sum_{j \in M_a \cap M_i} (m'_{aj} - \text{avg}(m'_j))(m'_{ij} - \text{avg}(m'_j))}{\sqrt{\sum_{j \in M_a \cap M_i} (m'_{aj} - \text{avg}(m'_j))^2} \sqrt{\sum_{j \in M_a \cap M_i} (m'_{ij} - \text{avg}(m'_j))^2}} \quad (3)$$

ここで M_a と M_i はそれぞれプロジェクト p_a と p_i で計測された (未欠損の) 変数の集合を表し、 $\text{avg}(m'_j)$ は j 番目の変数の平均値を表す。 $\text{sim}(p_a, p_i)$ の値域は $[-1, 1]$ である。

- 類似度 $\text{sim}(p_a, p_i)$ を用いて、プロジェクト p_a の変数の見積もり値 \hat{m}_{ab} を計算する。本研究では、見積もり値計算時に、プロジェクトの規模を補正する $\text{amp}(p_a, p_i)$ を乗じた値で加重平均を行う [17]。

$$\hat{m}_{ab} = \frac{\sum_{i \in k\text{-nearestProjects}} (m_{ib} \times \text{amp}(p_a, p_i) \times \text{sim}(p_a, p_i))}{\sum_{i \in k\text{-nearestProjects}} \text{sim}(p_a, p_i)} \quad (4)$$

$$\text{amp}(p_a, p_i) = \frac{\hat{fp}_a}{\hat{fp}_i} \quad (5)$$

ここで、 \hat{fp}_a , \hat{fp}_i はそれぞれプロジェクト p_a , p_i におけるソフトウェアの開発規模を示す。

3. 外れ値除去法

外れ値除去法とは、データセット中のプロジェクトが外れ値であるか検証し、外れ値と判断されたプロジェクトをデータセットから除去する処理である。これまで、ソフトウェア開発工数見積もりにおいて、重回帰分析適用時に Cook の距離を用いて外れ値を除去することが行われている ([7] など)。Cook の距離を用いた除去法は、重回帰分析を行い、モデルの係数に大きな影響を与えるプロジェクトを外れ値とみなす。Cook の距離は、あるプロジェクトをモデルの推定の計算から除外した場合に、すべてのプロジ

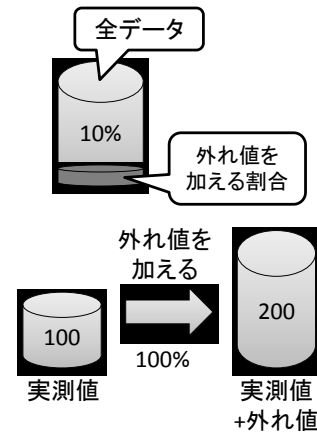


図 1 外れ値の割合, 外れ値の程度
Figure 1 Ratio of Outliers, Degree of Outliers

エクトの残差がどの程度変化するかを示す距離であり、Cook の距離が $4/n$ (n はデータ件数) 以上となるプロジェクトを外れ値とみなし [16], データセットから除外する。Cook の距離が大きいときは、回帰統計量の計算からプロジェクトを除外したことが係数を実質的に変化させたことを示す [16]。

Keung ら [6] は、Analogy 法への適用を前提とした、Mantel 相関を用いた除去法を提案している。Mantel 相関を用いた除去法は、説明変数が類似しているにもかかわらず、目的変数が類似していないケースを外れ値とみなす手法である。通常的相关係数は 2 つの変数の関連の強さを表すのに対し、Mantel 相関は (説明変数のセットと目的変数など) 2 つの変数セット間の関連の強さを表す。Mantel 相関により「プロジェクトの開発規模や開発期間 (説明変数) が似ていると、開発工数 (目的変数) も似ているかどうか」を調べることができる。Mantel 相関は、各プロジェクトのペアについて、説明変数に基づくユークリッド距離と目的変数に基づくユークリッド距離を計算し、それらの相関係数を計算することにより求められる。

本研究では、これらの外れ値除去法を適用しない場合における、外れ値の影響を評価する。

4. 実験

4.1. 外れ値に関する 4 つの観点

本研究では、(1) 外れ値の割合、(2) 外れ値の程度、(3) 外れ値の存在する変数、(4) 外れ値の存在

するデータセットの4つに着目し、これらが異なると工数見積もり精度に対する外れ値の影響も異なると仮定した。そこで、これらを変化させて外れ値の影響を分析した。

(1) 外れ値の割合：全データ件数に対する外れ値の割合を示す。例えば、データセットに含まれるプロジェクトの件数が100件あり、うち10件を実験的に外れ値に置換する場合、外れ値の割合は10%となる(図1)。

(2) 外れ値の程度：変数に記録されている値が、実際の値よりもどの程度ずれているかを示す。例えば、あるプロジェクトの工数が200人月と記録されており、これを実験的に400人月に置換した場合、外れ値の程度は100% ($|400-200|/200$)となる(図1)。

(3) 外れ値の存在する変数：外れ値は開発工数などの説明変数と目的変数である開発工数の両方に含まれると考えられる。そこで実験では、目的変数と、目的変数に最も影響が強いと考えられる説明変数である、開発規模に外れ値を加える。

(4) 外れ値の存在するデータセット：一般に、外れ値除去法は工数見積もりの根拠データに含まれる外れ値を除去することを想定しているが、見積もり対象データにも外れ値が含まれる(見積もり対象データが外れ値となる)可能性がある。見積もりの根拠データに外れ値除去法を適用する場合などを考慮すると、外れ値の存在するパターンは以下の3つに分類されると考えられる。

- 見積もり根拠データ:外れ値あり, 見積もり対象データ:外れ値なし
- 見積もり根拠データ:外れ値あり, 見積もり対象データ:外れ値あり

データ:外れ値あり

- 見積もり根拠データ:外れ値なし, 見積もり対象データ:外れ値なし

4.2. 実験手順

データセットに含まれる外れ値が工数見積もりの精度に与える影響を、以下の手順により分析した。

(1) データセットを無作為に半数ずつ分け、一方をフィットデータ、もう一方をテストデータとする。フィットデータは見積もりの根拠とするデータセット(過去プロジェクトに該当する)、テストデータは見積もり対象とするデータセット(開発中のプロジェクトに該当する)である。

(2) 実験的にデータセットに外れ値を加える。外れ値は4.1節で述べた方法により追加する。予備分析の結果より、開発規模に対しては外れ値の割合を10%及び20%、外れ値の程度を100%として分析した。工数に対しては外れ値の割合を10%、外れ値の程度を50%及び100%として分析した。外れ値の割合を10%から20%とした理由は、一般的に想定される外れ値の割合よりも若干多めに設定することにより、最悪の場合における外れ値の影響を確かめるためである。外れ値の程度を50%から100%とした理由は、これよりも大きい外れ値はデータの転記ミス以外では発生する可能性が低いと想定したためである。

(3) 外れ値が含まれたデータを用いて、見積もりモデルを構築し、工数を見積もり、見積もり精度の評価指標を算出する。

(4) (1) ~ (3)を10回繰り返し行い、評価指標の平均値と中央値を算出し、外れ値を全く加えなか

表2 重回帰分析における、外れ値と見積もり精度の関係(工数, 10%, 50%)

Table 2 Relationships between outliers and estimation accuracy on multiple regression analysis (effort, 10%, 50%)

フィットデータ	テストデータ	AE 平均	AE 中央値	MRE 平均	MRE 中央値	BRE 平均	BRE 中央値
外れ値あり	外れ値なし	-169.99	-173.69	-5.11%	-1.90%	-6.83%	-4.46%
外れ値なし	外れ値あり	-179.67	-86.01	0.17%	-0.69%	-3.51%	-1.70%
外れ値あり	外れ値あり	-342.93	-231.78	-4.54%	-2.19%	-10.02%	-5.94%

表3 重回帰分析における、外れ値と見積もり精度の関係(工数, 10%, 100%)

Table 3 Relationships between outliers and estimation accuracy on multiple regression analysis (effort, 10%, 100%)

フィットデータ	テストデータ	AE 平均	AE 中央値	MRE 平均	MRE 中央値	BRE 平均	BRE 中央値
外れ値あり	外れ値なし	-260.37	-207.64	-10.80%	-3.74%	-13.55%	-7.86%
外れ値なし	外れ値あり	-469.14	-151.18	0.82%	-1.68%	-7.55%	-2.58%
外れ値あり	外れ値あり	-711.48	-430.79	-8.82%	-6.50%	-19.77%	-10.98%

表 4 重回帰分析における、外れ値と見積もり精度の関係 (FP, 10%, 100%)

Table 4 Relationships between outliers and estimation accuracy on multiple regression analysis (FP, 10%, 100%)

フィット データ	テスト データ	<i>AE</i> 平均	<i>AE</i> 中央値	<i>MRE</i> 平均	<i>MRE</i> 中央値	<i>BRE</i> 平均	<i>BRE</i> 中央値
外れ値あり	外れ値なし	-18.03	19.90	4.00%	-1.23%	-0.09%	-0.70%
外れ値なし	外れ値あり	-110.67	-109.37	-3.94%	-2.00%	-2.58%	-3.59%
外れ値あり	外れ値あり	-107.91	-60.70	1.28%	-2.40%	-1.97%	-2.17%

表 5 重回帰分析における、外れ値と見積もり精度の関係 (FP, 20%, 100%)

Table 5 Relationships between outliers and estimation accuracy on multiple regression analysis (FP, 20%, 100%)

フィット データ	テスト データ	<i>AE</i> 平均	<i>AE</i> 中央値	<i>MRE</i> 平均	<i>MRE</i> 中央値	<i>BRE</i> 平均	<i>BRE</i> 中央値
外れ値あり	外れ値なし	-101.19	-99.18	1.03%	-2.75%	-6.97%	-3.53%
外れ値なし	外れ値あり	-135.70	-176.39	-8.09%	-1.55%	-5.48%	0.65%
外れ値あり	外れ値あり	-100.37	-138.22	-0.17%	-2.52%	-7.00%	-4.61%

った場合と比べて、どの程度評価指標の値が変化したのかを確かめる。

4.3. データセット

Desharnais データ[3]を用いて工数見積もりの精度評価した。Desharnais データは、Desharnais によって収集されたカナダのソフトウェア開発企業における 80 年代のデータセットである。このデータセットは、ソフトウェア開発工数見積もりの研究において広く用いられており[6][13]、かつ無料で入手できるデータセットの中ではデータ件数、説明変数の数ともに多いため利用した。データの収集時期は古いですが、例えば近年に発表された研究[6]でも用いられている。

データセットに含まれる 81 件のプロジェクトのうち、欠損値が含まれるプロジェクトを除いた 77 件を用いた。説明変数は、開発年度と調整済み FP (ファンクションポイント)、開発期間を除いた 7 つの変数 (開発チームの経験年数、プロジェクトマネージャの経験年数、トランザクション数、未調整 FP (以降 FP と略す)、調整係数、開発言語、エンティティ数) である。このうち開発言語はカテゴリ変数のため、ダミー変数化して用いた。

4.4. 評価尺度

工数見積もりの精度評価指標として、*AE*、*MRE* (Magnitude of Relative Error)[2]、*BRE* (Balanced Relative Error)[8]の 3 つの指標の平均値と中央値を用いた。工数の実測値を x 、見積もり値を \hat{x} とするとき、それぞれの指標は以下の式により求められる。

$$AE = |x - \hat{x}| \quad (6)$$

$$MRE = \frac{|x - \hat{x}|}{x} \quad (7)$$

$$BRE = \begin{cases} \frac{\hat{x} - x}{\hat{x}}, & \hat{x} - x \geq 0 \\ \frac{x - \hat{x}}{\hat{x}}, & \hat{x} - x < 0 \end{cases} \quad (8)$$

それぞれの指標の値が小さいほど、工数見積もりの精度が高いことを示す。直感的には *MRE* は実測値との相対誤差であるといえる。ただし、*MRE* は過大見積もりに対し、アンバランスな評価になるという問題がある。(見積もり工数が 0 以上の)過少見積もりの場合、*MRE* は最大でも 1 にしかならない(例えば実測値が 1000 人時、見積もり値が 0 人時の場合、*MRE* は 1 となる)。そこで本研究では、*MRE* に加え、過大見積もりと過少見積もりをバランスよく評価する指標[9]である *BRE* を評価指標に用いた。

フィットデータ、テストデータともに外れ値を含んでいないモデルを基準モデルとし、基準モデルと評価対象のモデルの評価指標の差により μ 値の影響を評価した。評価指標が負の値の場合、見積もり精度が基準モデルよりも低下したことを示す。

5. 実験結果

表 2 から表 9 に、各評価指標の基準モデルとの差を示す。表見出しのカッコ内に、外れ値が含まれる変数、外れ値の割合、外れ値の程度を示す。表における評価指標の値は 10 個のテストデータから得られた値を平均したものである。基準モデルよりも *MRE*、*BRE* が 5% 以上悪化した場合は無視できない見積もり精度の悪化であると

表 6 Analogy 法における、外れ値と見積もり精度の関係 (工数, 10%, 50%)

Table 6 Relationships between outliers and estimation accuracy on analogy based estimation (effort, 10%, 50%)

フィット データ	テスト データ	<i>AE</i> 平均	<i>AE</i> 中央値	<i>MRE</i> 平均	<i>MRE</i> 中央値	<i>BRE</i> 平均	<i>BRE</i> 中央値
外れ値あり	外れ値なし	-111.65	-25.49	-5.67%	-2.07%	-5.00%	-2.16%
外れ値なし	外れ値あり	-106.61	26.30	2.80%	1.32%	0.91%	1.32%
外れ値あり	外れ値あり	-199.98	-13.79	-2.41%	0.20%	-3.52%	-0.39%

表 7 Analogy 法における、外れ値と見積もり精度の関係 (工数, 10%, 100%)

Table 7 Relationships between outliers and estimation accuracy on analogy based estimation (effort, 10%, 100%)

フィット データ	テスト データ	<i>AE</i> 平均	<i>AE</i> 中央値	<i>MRE</i> 平均	<i>MRE</i> 中央値	<i>BRE</i> 平均	<i>BRE</i> 中央値
外れ値あり	外れ値なし	-395.67	-342.97	-14.26%	-8.23%	-12.93%	-8.07%
外れ値なし	外れ値あり	-410.08	-102.81	2.26%	-1.11%	-3.71%	-1.99%
外れ値あり	外れ値あり	-738.22	-393.27	-10.59%	-8.55%	-13.95%	-10.66%

考え、これらを太字で示した。なお、ここで 5%を基準とした理由は、ソフトウェア開発における利益(売上 - コストで計算される)を考慮した場合、コスト(工数)の誤差が 5%以上存在すると、利益が赤字になるリスクが無視できない程度に高まると想定したためである。

5.1. 重回帰分析における外れ値の影響

目的変数に外れ値を加えた場合

外れ値の程度が 10%の場合(表 3)、見積もり精度が全体的に悪化していた。外れ値の程度が 50%の場合(表 2)でも、フィットデータに外れ値あり、テストデータに外れ値なしで *MRE* 平均と *BRE* 平均が 5%以上悪化していた。フィットデータ、テストデータ両方に外れ値ありだと、*BRE* 平均が 10%、*BRE* 中央値が 5%以上悪化していた。よって、外れ値の影響は決して無視できないといえる。

当然の結果ではあるが、フィットデータ、テストデータ両方に外れ値ありの場合の見積もり精度が最も低くなっており、外れ値の程度に関わらず、*BRE* 平均と *BRE* 中央値が -5%を越えて悪化していた。

説明変数に外れ値を加えた場合

全体的に見積もり精度が悪化しているものの、工数に外れ値を加えた場合よりは精度の低下が低かった。外れ値の割合を 10%、外れ値の程度を 100%にした場合(表 4)、*MRE*、*BRE* の各評価値で 5%を超える悪化はなかった。ただし、外れ値の割合を 20%にした場合(表 5)は各評価値の悪化が見られた。特に *BRE* 平均が、どのデータセットに外れ値が存在する場合においても 5%以上悪化していた。

このことから、FP の誤差が大きいプロジェクトが見積も

り根拠データに多数含まれている場合、外れ値に注意する必要があるが、外れ値の割合が 10%、外れ値の程度が 100%程度であるならば、あまり外れ値の影響を考慮しなくてもよいといえる。

目的変数に外れ値を加えた場合では、フィットデータに外れ値があり、かつテストデータに外れ値があると、テストデータに外れ値がない時と比べて精度の低下が大きかった。説明変数に外れ値を加えた場合、フィットデータにその外れ値があれば、テストデータにおける外れ値の有無は見積もり精度にそれほど影響していなかった。

5.2. 類似性に基づく見積もりにおける外れ値の影響

目的変数に外れ値を加えた場合

外れ値の割合が 10%、外れ値の程度が 100%の場合(表 7)、見積もり精度が大きく低下しており、フィットデータに外れ値なし、テストデータに外れ値ありの場合を除き、*MRE*、*BRE* の平均と中央値が 5%以上悪化していた。ただし、外れ値の割合が 10%、外れ値の程度が 50%の場合(表 6)、フィットデータに外れ値あり、テストデータに外れ値なしの場合を除き、*MRE*、*BRE* の平均と中央値が 5%以上悪化していなかった。

説明変数に外れ値を加えた場合

全体的に評価指標の値が大きく悪化していなかった(表 8, 表 9)。外れ値の割合が 20%の場合(表 9)においても、*MRE*、*BRE* の平均と中央値が 5%以上悪化することがなかった。よって、工数見積もりに Analogy 法を用いる場合、開発規模に外れ値が含まれていても、あまり影響を考慮しなくてもよいと考えられる。

表 8 Analogy 法における, 外れ値と見積もり精度の関係 (FP, 10%, 100%)

Table 8 Relationships between outliers and estimation accuracy on analogy based estimation (FP, 10%, 100%)

フィット データ	テスト データ	<i>AE</i> 平均	<i>AE</i> 中央値	<i>MRE</i> 平均	<i>MRE</i> 中央値	<i>BRE</i> 平均	<i>BRE</i> 中央値
外れ値あり	外れ値なし	7.98	10.59	0.99%	0.49%	0.86%	0.43%
外れ値なし	外れ値あり	-38.52	-9.64	-1.18%	-0.89%	-0.78%	0.40%
外れ値あり	外れ値あり	-32.39	32.21	-0.15%	-0.91%	0.05%	0.64%

表 9 Analogy 法における, 外れ値と見積もり精度の関係 (FP, 20%, 100%)

Table 9 Relationships between outliers and estimation accuracy on analogy based estimation (FP, 20%, 100%)

フィット データ	テスト データ	<i>AE</i> 平均	<i>AE</i> 中央値	<i>MRE</i> 平均	<i>MRE</i> 中央値	<i>BRE</i> 平均	<i>BRE</i> 中央値
外れ値あり	外れ値なし	-16.57	3.15	0.99%	-1.02%	0.41%	-0.88%
外れ値なし	外れ値あり	-75.97	-55.83	-4.10%	-1.71%	-3.20%	-2.18%
外れ値あり	外れ値あり	-69.30	-77.98	-1.99%	-2.24%	-2.03%	-3.13%

5.3. 重回帰分析と類似性に基づく見積もりとの比較

目的変数に外れ値を加えた場合において, 重回帰分析と Analogy 法を比較すると, 見積もり精度の悪化は, 重回帰分析のほうが大きかった. 特にフィットデータとテストデータに外れ値が含まれる場合の差が大きかった. 例えば, 外れ値の程度が 50% の場合, Analogy 法では *MRE*, *BRE* の平均と中央値が 5% 以上悪化していなかったが, 重回帰分析では 5% 以上悪化していた.

Analogy 法のほうが, テストデータに外れ値が含まれていた場合の影響が小さい可能性がある. 外れ値の程度が 100% でフィットデータに外れ値がない場合, *BRE* 平均値の悪化は -3.71% であり, 重回帰分析の同じ場合よりも精度の悪化が小さかった. ただし, どの実験結果でも同様の傾向であるとまではいえないため, 結論を下すためには, 今後さらなる分析が必要である.

説明変数に外れ値を加えた場合においても, Analogy 法のほうが重回帰分析よりも見積もり精度の低下が小さかった. Analogy 法の場合, 外れ値の割合が 20% の場合においても, *MRE*, *BRE* の平均と中央値が 5% 以上悪化することがなかった.

これらの結果から, 重回帰分析よりも Analogy 法のほうが外れ値の影響を受けにくいと考えられる. ただし, フィットデータとテストデータに外れ値が含まれていない基準モデル同士を比較すると, Analogy 法のほうが重回帰分析よりも見積もり精度が低かった. Analogy 法の基準モデルの見積もり精度が低かったために, 外れ値によるさらなる精度低下が起きにくかった可能性もあることに留意する必要がある.

5.4. 考察

実験結果を, ソフトウェア開発の現場でどのように活用すべきかについて考察する. ソフトウェア開発工数の見積もりモデルを構築する際, ソフトウェア開発データセットに含まれる外れ値の影響を避けるためには, 以下の 2 つの方法がある.

- データセットに外れ値が含まれないように, データ計測及び収集を正確に行う.
- 外れ値と考えられるデータを除去する.

ただし, 上記 2 つの方法を厳格に適用しなくとも, すなわちデータを正確に収集することにコストを掛け過ぎたり, 外れ値を完全に除去したりすることをしない場合でも, 外れ値の影響が極端に大きくならない可能性がある. 実験に用いたデータセットにおいては, 例えば開発規模の計測値に 50% の誤差が含まれているケースが 10% 存在した場合でも, 誤差 (*BRE* 平均など) が 50% 以上変化するなどといった, 極端に見積もり精度が悪化するという事はなかった.

本研究の分析結果は 1 つのデータセットを用いただけであり, 今後さらに他のデータセットで同様の実験を行い, 結果の信頼性を高める必要がある. また, 外れ値の除去を行うことにより見積もり精度は改善されるため, 外れ値を除去する必要性がないというわけではない. ただし, 開発現場においてデータを収集, 活用する際には, 外れ値に対して厳格に対応する必要性は必ずしも高くない可能性がある.

なお, ソフトウェア工数見積もりの研究において, Desharnais データは広く用いられており [6][13],

Desharnais データ 1 つだけを用いて実験を行っている研究も存在する[6]. 本研究の実験結果を過度に一般的することは危険であるが, 他のデータセットで同様の実験を行った場合でも, 外れ値の工数見積り精度に対する影響が非常に大きいなど, 結果が極端に異なる可能性は低いと考えられる. ただし, 外れ値の程度と割合が見積り精度にどの程度影響を与えるか(*BRE* 平均値が何%悪化するか)は, データセットにより異なると考えられるため, 今後さらに別のデータセットにおいて同様の実験を行い, 詳細に両者の関係を分析する必要がある.

6. 関連研究

Seo ら[11]は, ソフトウェア開発工数見積り時に LTS 法を用いた除去法と k-means 法を用いた除去法を適用することを提案し, 重回帰分析, ニューラルネットワーク, ベイジアンネットワークそれぞれによる見積り時に外れ値除去法を適用し, 外れ値除去の効果を確かめている. ただし, 本研究のように実験的に外れ値を追加することは行っておらず, 外れ値の程度, 外れ値の割合などと見積り精度との関係は明らかにしていない.

Shepperd ら[12]は, 重回帰分析や類似性に基づく見積りなどの複数の見積り手法の性能を評価するために, 外れ値や多重共線性を持つデータセットを実験的に生成し, 分析に用いている. Shepperd らは説明変数に外れ値を加えているが, 外れ値の存在するデータセットを変えるなどはしておらず, 本研究のように複数の観点から外れ値の影響を評価することは行っていない.

7. おわりに

本研究では, 工数見積りにおいて外れ値が与える影響について明らかにするために, データセットに外れ値を実験的に追加して分析を行った. 分析では, 外れ値の割合, 外れ値の程度, 外れ値の存在する変数, 外れ値の存在するデータセットの 4 つを変化させて見積り精度を評価した. 工数見積り手法として, 重回帰分析と類似性に基づく見積り方法を採用した. その結果, 開発規模に外れ値が含まれていても, 外れ値の割合が 10%で, 外れ値の程度が 100%の場合, あまり影響が大きくないことなどがわかった. また, 類似性に基づく見積り方法の場合, テストデータに外れ値が含まれていても, 比較的影響が小さい可能性があることがわかった.

今後の課題は, さらにデータセットを増やすとともに, 外れ値の割合や外れ値の程度をさまざまに変化させて

実験することにより, 分析結果の信頼性を高めることである. 特に, 一般に広く知られている IPA/SEC が収集したデータセット[5]や, 工数見積りモデルの評価で広く用いられる ISBSG データセット[4]などを用いて同様の評価を行い, 結果の信頼性を高める必要がある.

謝辞 本研究の一部は, 文部科学省科学研究補助費 (基盤 C:課題番号 25330090)による助成を受けた.

参考文献

- [1] Boehm, B.: *Software Engineering Economics*, Prentice Hall (1981).
- [2] Conte, S., Dunsmore, H., and Shen, V.: *Software Engineering, Metrics and Models*, Benjamin Cummings (1986).
- [3] Desharnais, J.: *Analyse Statistique de la Productivité des Projets Informatique a Partie de la Technique des Point des Fonction*, Master Thesis, University of Montreal (1989).
- [4] International Software Benchmarking Standards Group (ISBSG): *ISBSG Estimating: Benchmarking and research suite*, ISBSG, 2004.
- [5] 情報処理推進機構 ソフトウェア・エンジニアリング・センター: ソフトウェア開発データ白書 2010-2011, 情報処理推進機構 (2010).
- [6] Keung, J., Kitchenham, B., and Jeffery, R.: Analogy-X: Providing Statistical Inference to Analogy-Based Software Cost Estimation, *IEEE Trans. on Software Eng.*, Vol.34, No.4, pp.471-484 (2008).
- [7] Mendes, E., Martino, S., Ferrucci, F., and Gravino, C.: Cross-company vs. single-company web effort models using the Tukutuku database: An extended study, *Journal of Systems and Software*, Vol.81, No.5, pp.673-690 (2008).
- [8] Miyazaki, Y., Terakado, M., Ozaki, K., and Nozaki, H.: Robust Regression for Developing Software Estimation Models, *Journal of Systems and Software*, Vol.27, No.1, pp.3-16 (1994).
- [9] Mølokken-Østfold, K., and Jørgensen, M: A Comparison of Software Project Overruns-Flexible versus Sequential Development Models, *IEEE Trans. on Software Eng.*, Vol.31, No.9, pp.754-766 (2005).
- [10] 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).

[11] Seo, Y., Yoon, K., and Bae, D.: An Empirical Analysis of Software Effort Estimation with Outlier Elimination, *Proc. the international workshop on Predictor models in software engineering (PROMISE)*, pp.25-32, Leipzig, Germany (2008).

[12] Shepperd, M., and Kadoda, G.: Comparing Software Prediction Techniques Using Simulation, *IEEE Trans. on Software Eng.*, Vol.27, No.11, pp.1014-1022 (2001).

[13] Shepperd, M. and Schofield, C.: Estimating software project effort using analogies, *IEEE Trans. on Software Eng.*, Vol.23, No.12, pp.736-743 (1997).

[14] Srinivasan, K., and Fisher, D.: Machine Learning Approaches to Estimating Software Development Effort, *IEEE Trans. on Software Eng.*, Vol.21, No.2, pp.126-137 (1995).

[15] Strike, K., El Eman, K., and Madhavji, N.: Software Cost Estimation with Incomplete Data, *IEEE Trans. on Software Eng.*, Vol.27, No.10, pp.890-908 (2001).

[16] 田中豊, 垂水共之 (編) : Windows 版 統計解析ハンドブック 多変量解析, 共立出版 (1995).

[17] 角田雅照, 大杉直樹, 門田暁人, 松本健一, 佐藤慎一 : 協調フィルタリングを用いたソフトウェア開発工数予測方法, 情報処理学会論文誌, Vol.46, No.5, pp.1155-1164 (2005).