

トービットモデルに基づくソフトウェア欠陥数予測の試み

村上優佳紗^{†1} 角田雅照^{†1,a)} 戸田航史^{†2,b)}

ソフトウェア開発においてソフトウェアの品質を適切に管理するためには、ソフトウェアリリース後に発見される欠陥数をできるだけ正確に予測する必要がある。欠陥数を数学的モデルにより予測する場合、目的変数となる欠陥数は最小値が0になりうるが、目的変数に0が多く含まれる場合、従来の最小二乗法に基づく重回帰分析では適切なモデルが構築されず、予測値が負の値となる可能性がある。そこで本研究では、トービットモデルを用いてソフトウェア欠陥数予測モデルを構築し、その効果を確認する。トービットモデルは、目的変数の最小値が0であることを前提としたモデルである。ISBSG データセットを用いて、最小二乗法に基づく重回帰モデル、トービットモデル、対数変換を適用した最小二乗法に基づく重回帰モデル、対数変換を適用したトービットモデルの4つの手法の予測精度を評価した。その結果、対数変換を適用したトービットモデルは比較的高い精度を示し、BRE 中央値が93%となった。

An Experiment of Software Defect Prediction Based on the Tobit Model

Yukasa MURAKAMI^{†1} Masateru TSUNODA^{†1,a)} Koji TODA^{†2,b)}

In software development project, it is important to predict the number of defects found after software release accurately, to manage software quality properly. When the number of defects is predicted by a mathematical model, the minimum value of the dependent variable is zero. Ordinary least squares regression (OLS) cannot make an appropriate model when there are many cases where the value of the dependent variable is zero. In the case, predicted values may be smaller than zero. To make the appropriate model, we used Tobit model to build the defect prediction model. Tobit model assumes that there are cases where the value of the dependent variable is zero. Using the ISBSG dataset, we built OLS, Tobit model, OLS with log-transformation, and Tobit model with log-transformation. As a result, the Tobit model with log-transformation showed relatively higher accuracy, and the median of BRE was 93%.

1. はじめに

近年、ソフトウェア開発プロジェクトは大規模化しており、そのようなソフトウェア開発を成功させるためには、コストや品質に関する管理を適切に実施する必要がある。管理を適切に行うためには、開発コストやソフトウェアの品質などを予測し、テストや人員配置などの計画を立案する必要がある。ソフトウェア欠陥数の予測は、ソフトウェアの品質を管理するための基礎となるものである。

ソフトウェアの品質管理を適切に行うためには、ソフトウェア欠陥数を高い精度で予測することが非常に重要である。予測精度が低い場合、テスト計画を適切に作成することができず、それにより納期の遅れ、品質の低下、コストの超過といった深刻な問題が生じる可能性がある。そのため、これまで数学的に欠陥数を予測するモデルが提案されてきた。欠陥数の予測に広く用いられるモデルとして重回帰分析があげられる。数学的にソフトウェア欠陥数を予測するためには、過去のプロジェクトにおいて得られたデータを予測の根拠データとし、予測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を用いて、

ソフトウェア欠陥数を予測する。

ただし、ソフトウェアの欠陥数は最小値が0であるため、そのまま最小二乗法に基づいて重回帰モデルを構築した場合、適切なモデルが構築されない可能性がある。特にソフトウェアのリリース後に発見される欠陥数を予測する場合、無視できない問題となりうる。具体的には、開発規模が一定以下の場合、出荷後の欠陥数は0となることが多いが、重回帰分析では、一定規模以上では欠陥数と規模が比例関係にあり、それ以下では欠陥数が一定となるといったモデルを構築することができない。

本研究では、計量社会学などの他分野で用いられているトービットモデル[12]をソフトウェア欠陥数の予測に適用し、予測精度を改善する効果があるのかを確認する。トービットモデルは、目的変数の最小値が0であることを前提としたモデルである。このモデルを欠陥数予測に適用することより、適切な予測モデルが構築され、予測精度が改善することが期待される。実験において、実際のソフトウェア開発企業から収集されたデータセットを用いて、従来の重回帰分析とトービットモデルによる予測モデルを構築し、各モデルの予測精度を比較することにより、トービットモデルの有用性を評価する。各モデルの構築前に、対数変換を各変数に適用した場合の予測精度についても確認する。トービットモデル自体はかなり以前に提案されたモデルであるが、我々の知る限り、これまでソフトウェア欠陥数の

^{†1} 近畿大学
Kindai University, Japan

^{†2} 福岡工業大学
Fukuoka Institute of Technology, Japan
a) tsunoda@info.kindai.ac.jp
b) toda@fit.ac.jp

予測に適用されておらず、その有効性は明らかでない。

2. 予測モデル

本研究では予測モデルとして、トービットモデルと、ソフトウェア欠陥数の予測モデルとして広く用いられている重回帰分析の2つを用いる。

2.1 重回帰分析

重回帰分析による予測モデルは、ソフトウェア欠陥数を数学的に予測する際に広く用いられる。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予測の根拠データとして、最小二乗法により予測モデルが1つ構築される。総欠陥数を y 、開発規模などの説明変数を x_1, x_2, \dots, x_k とするとき、各プロジェクトの総欠陥数は以下の式で説明される。

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

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。なお、重回帰分析により予測モデルを構築する場合、経験的に説明変数の5~10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欠陥数を予測する場合、モデルの構築前に比例尺度の説明変数と目的変数に対し、対数変換を適用する場合がある。対数変換を適用した重回帰分析はソフトウェア開発データの特徴を表すのに適していることが指摘されており[9]、さらに、対数変換を行うことにより、予測値が負の値となることを避けることができる。そこで本研究では、対数変換を適用した重回帰分析に基づく予測モデルも構築して評価する。

2.2 トービットモデル

トービットモデル[12]とは、目的変数における分布の偏りを考慮したモデルである。分布の偏りは以下の3つに分類される[8]。

- 打ち切り (censored)
- 切断 (truncated)
- 付随的切断 (incidental truncation)

打ち切りとは、目的変数の最小値が0以上など、値域が限定されている場合であり、欠陥数が該当する。切断とは、欠陥数が0のデータが除外されている場合である。付随的切断とは、本来0でない数値が0となる場合である。例えばレビューが省略されたためにレビュー指摘数が0になった場合などがあげられる。打ち切り、切断についてはタイプIトービットモデル、付随的切断についてはタイプIIトービットモデルが用いられる。ソフトウェア欠陥数の最小値は0以上であり、値域が限定されているため、打ち切りデータに該当する。本研究では、ソフトウェア欠陥数を予測するため、タイプIトービットモデルの打ち切り回帰モ

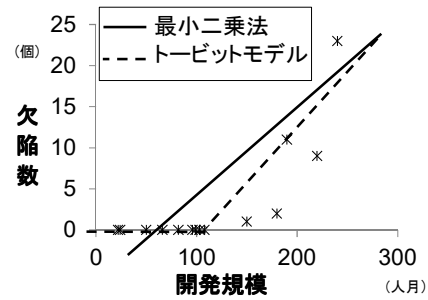


図1 トービットモデルと従来法のモデル
Figure 1 The Tobit model and an ordinary model.

デルを適用する。

タイプIトービットモデルは、打ち切り、切断に有効なモデルである。打ち切り回帰モデルは以下ようになる[8]。

$$y^* = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k + \varepsilon \quad (2)$$

$$y = \begin{cases} y^* & y^* > 0 \\ 0 & y^* \leq 0 \end{cases} \quad (3)$$

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。式(3)により、目的変数の値の分布の偏りを考慮することができる。

切断回帰モデルは以下ようになる[8]。

$$y^* = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k + \varepsilon \quad (4)$$

$$y = y^* \quad y^* > 0 \quad (5)$$

切断データの場合、目的変数が0未満の場合はデータセットに含まれないため、式(5)のようなモデルとなる。この場合、式(5)により目的変数の分布の偏りを考慮できる。トービットモデルはかなり以前に提案されたものであるが、我々の知る限り、これまでソフトウェア欠陥数の予測モデル構築に適用されていない。

図1に、説明変数を開発規模、目的変数をソフトウェア欠陥数として予測モデルを構築した場合のトービットモデルと最小二乗法に基づく重回帰モデルのイメージを示す。グラフ内にプロットされている点は各プロジェクトのケースを表す。図1のように欠陥数が0のデータが多いデータセットの場合、最小二乗法に基づく重回帰モデルと比べて、トービットモデルは欠陥数が0のケースを適切に扱えていることがわかる。

3. 実験

3.1 概要

ソフトウェア欠陥数の予測における、トービットモデル

表 1 実験に用いた変数

Table 1 Variables used in the experiment.

変数名	尺度	説明
欠陥数	比例	ソフトウェアリリース後1ヶ月以内に発見された総欠陥数
FP	比例	ソフトウェアに含まれる機能数 (Function Points). 未調整
開発種別	名義	拡張, 新規, その他
業種	名義	銀行, 金融業, 保険業, 製造業, その他
プラットフォーム	名義	メインフレーム, ミッドレンジ, その他

の有用性を確かめるために、以下の4つのモデルを用いて欠陥数の予測を行い、それぞれの予測精度を比較した。

- 最小二乗法に基づく重回帰モデル(以降 OLS とする)
- トービットモデル (以降 Tobit とする)
- 対数変換を適用した最小二乗法に基づく重回帰モデル (以降 Log-OLS とする)
- 対数変換を適用したトービットモデル (以降 Log-Tobit とする)

目的変数に対数変換を適用して、最小二乗法に基づく重回帰モデルを構築した場合、予測値が負の値となることを避けることができる。また、対数変換を適用した重回帰分析はソフトウェア開発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[1][9]。そこで実験では、トービットモデル、最小二乗法に基づく重回帰モデルの両方に対し、比例尺度の変数を対数変換してモデルを構築し、その場合の予測精度も評価した。

なお、欠陥数には0が含まれており、そのままでは対数変換できない。そこで対数変換を行う場合については、欠陥数にあらかじめ1を加えた上で変換を行った。対数変換時に1を加えてから変換することは一般的に行われる方法である。

3.2 データセット

トービットモデルの有用性を評価するために、ISBSG (International Software Benchmarking Standards Group) が収集したプロジェクトデータ (ISBSGデータ) [3]を用いて分析を行った。ISBSGデータは、20ヶ国のソフトウェア開発企業から収集されたものであり、ソフトウェアプロジェクトの予測モデルの精度評価などに広く用いられている[4][7]。用いたデータセットはRelease 9と呼ばれるバージョンであり、1989年から2004年までに実施されたソフトウェア開発プロジェクトが3026件、変数が99個含まれている。データには欠損値が含まれている。欠損値とは変数に値が

記録されていないことを指す。

実験対象のプロジェクトの条件を整えるため、データ品質評価がAまたはB、FP計測法がIFPUG法のデータを抽出した。また、分析で用いた変数に欠損が含まれているデータを除去した。その結果、221件のプロジェクトが分析対象となった。

実験に用いた変数を表1に示す。表中のFP、開発種別、業種、プラットフォームを説明変数とし、欠陥数を目的変数とした。名義尺度の変数については、それぞれ1つ以上のダミー変数に変換したものをを用いた。ダミー変数は0または1の値を取り、各変数が各カテゴリを表す変数となっている。ダミー変数化により、名義尺度の変数に含まれるカテゴリ数 - 1個のダミー変数が作成される。

予備分析において、最小二乗法による重回帰分析においてAICによる変数選択を行い、予測モデルの説明変数として用いる変数を絞り込んだ。具体的には、FP、新規開発、金融業、メインフレーム、ミッドレンジを説明変数として採用した。なお、FP以外は名義尺度をダミー変数化したものである。

3.3 評価指標

評価指標として、絶対誤差 (AE)、Balanced Relative Error (BRE) それぞれの平均値、中央値とPred25を用いた。ソフトウェア欠陥数の実測値を x 、予測値を \hat{x} とすると、AEとBREは以下の式により求められる。

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

$$BRE = \begin{cases} \frac{\hat{x}-x}{x}, \hat{x} - x \geq 0 \\ \frac{x-\hat{x}}{\hat{x}}, \hat{x} - x < 0 \end{cases} \quad (7)$$

AE、BREは値が小さいほど精度が高いことを示す。直感的にはBREは予測値と実測値との相対誤差であるといえる。Pred25とは、実測値と予測値の相対誤差 (本論文ではBREを用いた) が25%以内のケースの割合を指し、数値が高いほど精度が高いことを示す。

相対誤差としてMRE (Magnitude of Relative Error) が用いられる場合もあるが、MREは過小予測に対し、不適切な評価になるという問題がある。(予測値が0以上の) 過小予測の場合、MREは最大でも1にしかならない。例えば欠陥数の実測値が100個、予測値が0個の場合でも、MREは1としかならない。そこで本研究では、MREの代わりに、過大予測と過小予測をバランスよく評価することが可能であるBREを評価指標として採用した。

3.4 実験手順

モデルの予測精度を評価するために、5-fold cross validation (図 2) を適用した。5-fold cross validation は、予測モデルの評価に一般に用いられる方法の一つである。詳細な手順は以下の通りである。

- (1) データセットを無作為に 5 つに分け、そのうちの 1 つをテストデータ、その他をラーニングデータとする。ラーニングデータは予測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)、テストデータは予測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) テストデータを入れ替え、それぞれにおいて予測を行い、予測指標を計算する。
- (3) 5 回算出された予測指標の平均を算出し、最終的な予測指標の値として採用する。

5-fold cross validation を OLS, Tobit, Log-OLS, Log-Tobit それぞれのモデルに対して適用して予測精度を算出し、各モデルの予測精度を確かめる。

4. 実験結果

実験結果を表 2, 表 3 に示す。従来の最小二乗法に基づく重回帰モデルと比較して、トービットモデルを用いることによりどの程度精度が改善するのかを明確にするため、双方のモデルの評価指標の差分を算出した。差分は、OLS (または Log-OLS) の各評価指標から、Tobit (または Log-Tobit) の各評価指標を減じた値である。ただし、Pred25 のみ逆にして差分を計算した。これは、それぞれの評価指標の差分が正の値の場合、トービットモデルの予測精度が高

い (負の値の場合、最小二乗法に基づく重回帰モデルの予測精度が高い) ことを示すようにしたためである。

4.1 対数変換を適用しない場合の予測精度比較

対数変換を適用しない場合のモデルの予測精度を表 2 に示す。OLS (最小二乗法に基づく重回帰モデル) と Tobit (トービットモデル) を比較すると、5 個中 3 個の評価指標の差分が正の値となっていた。特に BRE 中央値が 50% 程度改善していた。ただし、BRE 平均値と Pred25 の差分が負であった。BRE 平均値については OLS よりも 100% 程度悪化していた。また、Pred25 についても、値が 8% であり、精度が高いとはいえなかった。このことから、対数変換を適用しない場合、予測精度を高めるために、最小二乗法に基づく重回帰モデルとトービットモデルの、どちらのモデルを適用すべきであると結論付けることは難しい。

4.2 対数変換を適用した場合の予測精度比較

対数変換を適用した場合のモデルの予測精度を表 3 に示す。対数変換を適用しない場合と比較すると、最小二乗法に基づく重回帰モデル、トービットモデルとも予測精度が大きく改善していた。Log-OLS と Log-Tobit を比較すると、5 個中 3 個の評価指標の差分が正の値となっていた。AE 平均値の差分は負の値であったが、差は非常に小さかった。Log-OLS と比べ、BRE 平均値は Log-Tobit のほうが 34% 低かったが、対数変換しない場合と比較すると、両者の差は小さくなっていた。BRE 中央値と Pred25 については、Log-OLS と比較してそれぞれ 14%、7% 改善していた。これらの結果より、トービットモデルについても対数変換を適用することにより予測精度が改善するといえる。

表 2 各モデルの予測精度

Table 2 Prediction accuracy of the models.

	AE 平均値	AE 中央値	BRE 平均値	BRE 中央値	Pred25
OLS	10.74	4.11	284%	165%	12%
Tobit	9.66	2.04	378%	115%	8%
差分	1.08	2.06	-94%	49%	-4%

表 3 各モデルの予測精度 (対数変換適用時)

Table 3 Prediction accuracy of the models (Applied log-transformation).

	AE 平均値	AE 中央値	BRE 平均値	BRE 中央値	Pred25
Log-OLS	8.24	1.86	226%	107%	21%
Log-Tobit	8.45	1.53	260%	93%	28%
差分	-0.21	0.33	-34%	14%	7%

表 4 予測結果の選択的採用

Table 4 Results of Additional Experiments

	AE 平均値	AE 中央値	BRE 平均値	BRE 中央値	Pred25
Log-OLS	8.24	1.86	226%	107%	21%
Log-Tobit	8.45	1.53	260%	93%	28%
Log-Merge	8.18	1.57	248%	92%	33%

表 5 各予測方法の適合率, 再現率, F1 値

Table 5 Recall, precision, and F1 value of the prediction methods.

	再現率	適合率	F1 値
Log-OLS	44.2%	64.4%	52.4%
Log-Tobit	74.4%	52.5%	61.5%

4.3 適合率, 再現率, F1 値による評価

欠陥数が0のデータを適切に予測できているかどうかを確かめるために, 適合率, 再現率, F1 値を用いて各モデルの予測精度を評価した. 適合率は欠陥値が0と予測された時に, それが正しい割合, 再現率は欠陥値の実測値が0のデータを, 正しく0と予測できている割合, F1 値は適合率と再現率の両方の指標を同時に評価するための指標である. それぞれの指標の値が大きいくほど, 予測精度が高いことを示す.

結果を表 5 に示す. 4.2 節において予測精度の高かった Log-OLS と Log-Tobit についてのみ示す. 予測値が1未満の場合は予測値は0であるとして各指標を計算した. Log-Tobit の F1 値が大きかったことから, 欠陥数が0のデータについては, Log-Tobit のほうがより適切に扱えているといえる.

本章の実験結果からは, 最小二乗法に基づく重回帰モデルとトービットモデルのどちらの予測精度が明確に高いとは結論付けられない. よって, 予測モデルの構築時には, 最小二乗法に基づく重回帰モデルだけではなく, トービットモデルも予測モデル構築方法の候補に含めるべきであるといえる.

5. 考察

4.3 節の結果より, Log-Tobit の予測値が1未満となっている場合, Log-Tobit のほうが Log-OLS よりも精度が高いと考えられる. この場合に Log-Tobit の予測値を採用し, それ以外の場合に Log-OLS の予測値を採用することにより, Log-Tobit の BRE 平均値が Log-OLS よりも悪化することを防げる可能性がある. そこで, トービットモデルの本来の利用方法から考えるとやや乱暴ではあるが, 2 つのモデルの予測値を選択的に採用した場合の予測精度を確かめた.

2 つのモデルの予測値を選択的に採用する方法 (以下 Log-Merge とする) は以下の手順で行う (図 3).

- (1) Log-Tobit で予測し, 予測値が1未満となった場合, そのまま予測値を採用する.
- (2) (1)以外の場合, Log-OLS の予測値を採用する.

Log-Merge により予測を行い, 5 つの評価指標を算出して Log-OLS と Log-Tobit との精度の比較を行った. 各モデルの評価指標を表 4 に示す. 太字部分は, 3 つのモデルの中で最も評価指標が高かったものを表す. Log-Merge と

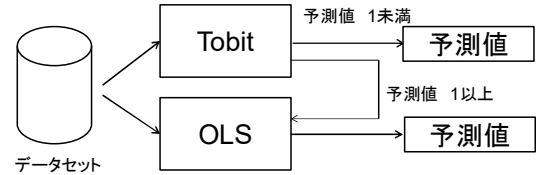


図 3 予測値の選択的採用

Figure 3 Selection of the predicted value.

Log-OLS を比較すると, BRE 平均値以外が改善しており, 特に Pred25 が 10%以上改善していた. Log-Merge と Log-Tobit を比較すると, AE 中央値以外が改善しており, 特に BRE 平均値が 10%以上改善していた.

Log-Tobit を適用する場合と比べて, Log-OLS からの BRE 平均値の悪化が少なくなったため, 予測精度だけに着目するとより適切なモデルとみなすこともできるが, そもそもトービットモデルの本来の利用方法から考えるとかなり無理のある予測方法である. 他のデータセットに適用した場合に, 同様の結果が得られるかどうかは不明であり, 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, 結果を確かめる必要がある.

6. 関連研究

これまで, テスト工程における欠陥数や出荷後の欠陥数を予測した研究は数多く存在し, 様々な数学的モデルが用いられている. 例えば吉村ら[13]は, プロセスメトリクスを説明変数として, ソフトウェアの残存バグ数を一般化線形モデルにより予測している. また, 出荷後に得られるメトリクスを利用し, 残存バグ数を Rayleigh モデルにより予測している. 小室ら[6]は, 上流工程のレビューデータから, テスト工程での欠陥摘出数を一般化線形モデルに基づき予測している. ソフトウェア信頼度成長曲線[10]を用いて欠陥数を予測する場合もある[2]. ただし, 我々の知る限り, 欠陥数の予測にトービットモデルを用いた研究はこれまで存在しない.

トービットモデルと同様に, 目的変数に0のデータが多いことを前提としたモデルとして, ポアソン回帰モデルがあり[5], 欠陥数の予測に用いられることがある. ポアソン回帰モデルでは, 目的変数がポアソン分布に従うことを前提としている. ただし, ポアソン分布は平均と分散が等しいことを前提としているため, トービットモデルよりも欠陥数予測に適していない可能性がある. ポアソン回帰モデルとトービットモデルを比較することは今後の課題の一つである.

7. おわりに

本研究では, ソフトウェア欠陥数の予測にトービットモデルを適用し, 予測精度が従来法と比べて改善するかどうかを実験的に確かめた. 分析では, 最小二乗法に基づく重

回帰モデル, トービットモデル, 対数変換を適用した最小二乗法に基づく重回帰モデル, 対数変換を適用したトービットモデルの4つを用いた. 実験の結果, 対数変換を適用したモデルの予測精度が高くなっており, トービットモデルは最小二乗法に基づく重回帰モデルと比べ, *BRE* 中央値と *Pred25* がそれぞれ 14%, 7%改善した.

今後の課題は, さらに適用するデータセットを増やし, 同様の結果が得られるかを確かめることである. その際, 欠陥数が0のケースの割合と予測精度との関係に着目して分析し, トービットモデルの有効性を確かめる.

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] 古山恒夫: ソフトウェアプロジェクトデータの量的変数に関する分析の一指針と分析事例, *SEC journal*, Vol.7, No.3, pp.105-111 (2011).
- [2] 本田澄, 鷺崎弘宜, 深澤良彰: リポジトリシステムとソフトウェア信頼性モデルを用いた欠陥数予測, *信学技報*, Vol.114, No.271, SS2014-27, pp.13-16 (2014).
- [3] International Software Benchmarking Standards Group: ISBSG Estimating, Benchmarking and Research Suite Release 9, ISBSG (2004).
- [4] Jeffery, R., Ruhe, M. and Wiecek, I.: Using Public Domain Metrics to Estimate Software Development Effort, *Proc. International Software Metrics Symposium (METRICS)*, pp.16-27 (2001).
- [5] Khoshgoftaar, T., Gao, K., and Szabo, R.: An application of zero-inflated poisson regression for software fault prediction, *In Proc. of International Symposium on Software Reliability Engineering (ISSRE)*, pp.66-73 (2001).
- [6] 小室睦, 薦田憲久: ピアレビューデータに基づく品質予測モデル, *電子情報通信学会論文誌 D*, Vol.J94-D, No.2, pp.439-449 (2011).
- [7] Mendes, E., Lokan, C., Harrison, R. and Triggs, C.: A Replicated Comparison of Cross-company and Within-company Effort Estimation Models using the ISBSG Database, *Proc. International Software Metrics Symposium (METRICS)*, p.36 (2005).
- [8] 水落正明: 打ち切り・切断データの分析, 理論と方法, Vol. 24, No. 1, pp. 129-138 (2009).
- [9] 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).
- [10] Musa, J., Iannino, A., and Okumoto, K: *Software Reliability Measurement, Prediction, Application*, McGraw-Hill, New York, 1987.
- [11] Sentas, P., Angelis, L., Stamelos, I. and Bleris, G.: Software productivity and effort prediction with ordinal regression, *Information and Software Technology*, Vol.47, No.1, pp.17-29 (2005).
- [12] Tobin, J.: Estimation of relationships for limited dependent variables, *Econometrica*, Vol .26, No.1, pp.24-36 (1956).
- [13] 吉村博昭, 森岳志, 倉下亮, 野中誠: プロセスおよび出荷後バグメトリクスによる残存バグ予測, *ソフトウェア品質シンポジウム 2010 発表報文集* (2010).