推薦論文● FOSE 2009

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

門田 暁人　小林 健一

ソフトウェア開発工数予測において，線形重回帰モデルは最も基本的な予測モデルとして多くの採用実績がある．その適用の前処理として，変数変換 (特に，対数変換) が有効であるが，その理論的根拠は必ずしも明らかでなかった．本論文では，対数変換を行った線形回帰モデルは，指数曲線モデルと等価であり，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を示す．ただし，対数変換を行ってから線形回帰を行うと，逆変換する際に過小予測するバイアスが発生してしまうことは見過ごされがちである．本論文ではその補正方法も示す．

Multivariate linear regression models have been commonly used as software effort prediction models. To improve the prediction accuracy, it is a common practice to transform (especially, log-transform) the data before building a model, although its theoretical basis is not necessarily clear. This paper reveals that the log-transformed linear regression model (log-log regression model) is equal to the exponential model, which is suitable to characterize various relationships among software related metrics. However, when using a log-log regression model, the result of inverse transformation tends to under-estimate the effort. This paper also introduces a method to correct such bias.

1　はじめに

本論文では，ソフトウェア開発工数 (人月または人時) の予測に線形重回帰モデルを用いる場合を取り扱う．一般に，線形重回帰モデルは式 (1) の形式を持つ．

$$\hat{Y} = \sum_{j=1}^{n} k_j N_j + C \qquad (1)$$

$\hat{Y}$: 目的変数 (開発工数) の予測値

N_j : 説明変数 (プロジェクト特性)

k_j : 偏回帰係数

C : 定数項

この式では，ソフトウェア開発工数が，プロジェクト特性 (開発規模，期間，言語，アーキテクチャなど) の線形結合により表現される．

線形回帰モデルの予測性能を高める方法として，前処理に変数変換 (特に，対数変換) を行うことが知られているが，その意義や理論的根拠は必ずしも明確ではない．近年，Kitchenham と Mendes [6] は，変数変換の重要性を指摘し，目的変数である開発工数に加えて，規模の尺度 (ファンクションポイント) を対数変換すると，値の分布が正規分布に近づき，予測性能のよい線形回帰モデルができることを示している．ただし，線形回帰の適用条件として，説明変数が正規分布していることが求められているわけではない [8]．また，対数変換を行ってから線形回帰を行うと，各プロジェクト特性は工数に対して (加法的でなく) 乗法的に作用するようになる．この点についても，従来，明確な説明が行われていない．

そこで，本論文では，線形回帰モデルを構築するにあたって，対数変換を行うことの意義や理論的根拠を明らかにする．以降，2 節では，対数変換を行った

The Effect of Log Transformation in Multivariate Liner Regression Models for Software Effort Prediction.

Akito Monden, 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.

Kenichi Kobayashi, 富士通研究所ソフトウェアイノベーション研究部, Software Innovation Laboratory, Fujitsu Laboratories Ltd.

コンピュータソフトウェア, Vol.27, No.4 (2010), pp.234–239.
[研究論文 (レター)] 2010 年 5 月 31 日受付.

線形回帰モデルは，指数曲線モデルとほぼ等価であり，現実をより正しく表すモデルとなっていることを示す．3 節では，プロジェクトデータの特徴から，対数変換の妥当性を論じる．4 節では，ケーススタディを通して，対数変換の有効性を示す．5 節はまとめである．

2 対数変換を伴う線形重回帰モデル

2.1 工数見積もりモデルの一般形

対数変換の意義を明らかにするために，本節では，多くの企業で用いられている工数見積もりモデルの一般形を紹介し，対数変換してから線形回帰を行った式と同じ形となることを示す．

工数見積もりモデル (コストモデルとも呼ばれる) の一般形は，式 (2) のようなべき関数で表される [11]．

$$E = sL^c \tag{2}$$

E : 開発工数
L : 開発規模
s : 生産性調整係数

COCOMO などの代表的なコストモデルにおいてもべき関数が採用されており，適用事例も多い．初級 COCOMO では，開発形態によって s，c の値が決まり，小規模開発を想定した organic モードでは $s = 2.4$，$c = 1.05$ である [1]．

ここで，式 (2) の両辺を対数変換[†1]すると，

$$\log(E) = c \cdot \log(L) + \log(s) \tag{3}$$

となり，実は，規模 E と工数 L を対数変換してから線形回帰を行った場合と同じ形となる．このことから，対数変換してから線形回帰を行うことは，工数と規模の関係をモデル化するのに都合がよい．

同様に，開発期間と工数の関係についても，べき関数への当てはまりが良いことが知られており，開発期間は工数の概ね 3 乗根に比例するという経験則がある [1]．従って，対数変換してから線形回帰を行うことで，工数と開発期間の関係についてもうまくモデル化できる．逆に，対数変換を行わない naïve な線形回帰モデルは，規模，工数，開発期間の関係を表すには本来向いていないといえる．

2.2 対数変換による指数曲線モデルの構築

本論文では，線形回帰の前処理として，全ての説明変数および目的変数を対数変換する場合を想定する．得られるモデル式は，式 (4) の通りである．このモデル式を本論文では，log-log 重回帰モデルと呼ぶことにする．

$$\log(\hat{Y}) = \sum_{j=1}^{n} k_j \log(N_j) + C \tag{4}$$

ここで，式 (4) の両辺について exp を取ると，式 (5) の指数曲線モデルが得られる．従って，log-log 重回帰モデルは，指数曲線モデルであるとも言える．

$$\hat{Y} = \exp(C) \prod_{j=1}^{n} {N_j}^{k_j} \tag{5}$$

式 (5) で明らかとなったように，一見，非線形回帰に見える指数曲線モデルが，対数変換によって単純な線形回帰で得られるのである．式 (4) の log-log 重回帰モデルから予測工数 $\hat{Y}$ を得るためには，出力値 $\log(\hat{Y})$ の exp を取る必要があり，このことは，式 (5) の指数曲線モデルから予測値を得ていることと等価である．

指数曲線モデルでは，各プロジェクト特性は工数に対して (加法的でなく) 乗法的に作用する．このことは，3.4 節で後述するように，開発の生産性に寄与する説明変数を含む場合にモデルの当てはまりが良い．

2.3 指数曲線モデルの残差

式 (5) の指数曲線モデルは，残差にバイアスを生じる (過小予測傾向となる) という落とし穴がある．以下，指数曲線モデルの残差について述べる．

プロジェクト i の工数を Y_i，その残差を ϵ_i とおくと，log-log 重回帰モデルにおける各変数と残差の関係は，次式で表される．

$$\log(Y_i) = \sum_{j=1}^{n} k_j \log(N_j) + C + \epsilon_i \tag{6}$$

線形回帰モデルが妥当となるためには，この残差が以下の性質を満たすことが仮定される [5] [8] [10]．

- 独立性：ϵ_i と ϵ_j は互いに独立である ($i \neq j; i, j =$

†1 本論文では，log は，e を底とする自然対数を用いる．

$1, 2, \cdots, r$． r はプロジェクト件数)．

- 不偏性：ϵ_i の期待値はゼロである．
- 等分散性：ϵ_i の分散は全て等しい．
- 正規性：ϵ_i は正規分布に従う．

ここでは，この ϵ_i が従う正規分布を $N(0, \sigma)$ とする．

一方，式 (6) を変形すると，指数曲線モデルにおける測定値と予測値の関係は，

$$\begin{aligned} Y_i &= \exp(\epsilon_i)\exp(C)\prod_{j=1}^{n} {N_j}^{k_j} \\ &= \exp(\epsilon_i)\hat{Y}_i \end{aligned} \tag{7}$$

となり，その残差 ζ_i は，

$$\begin{aligned} \zeta_i &= Y_i - \hat{Y}_i \\ &= \hat{Y}_i(\exp(\epsilon_i) - 1) \end{aligned} \tag{8}$$

となる．以降では，log-log 重回帰モデルでの残差 ϵ_i の性質が，指数曲線モデルでの残差 ζ_i にどのように現れるかを述べる．

まず，独立性が保たれることは，$i \neq j$ のとき ϵ_i, ϵ_j が独立ならば，$\log(\epsilon_i)$，$\log(\epsilon_j)$ も独立であることから明らかである．

次に，等分散性については，残差 ζ_i が $\hat{Y}$ に比例する値となり，これは工数見積もりモデルとしてむしろ好都合である．式 (8) は，工数が小さなプロジェクトは予測誤差が小さく，大きなプロジェクトになるほど予測誤差が大きくなることを意味する．一般に，大規模プロジェクトほど工数の見積もり誤差も大きくなるため，このような残差の振る舞いをする指数曲線モデルは，現実をモデル化するのにより都合がよい．一方，naïve な線形回帰モデルでは，プロジェクトサイズに応じた分散の変化をうまくモデル化できない．

次に，正規性については，残差 ζ_i は正規分布ではなく -1 だけシフトした対数正規分布となり，プラス側に裾野の広い分布となる．これは，工数の見積もり誤差は超過方向に大きく外れることが多いという現実に即していると言える．

最後に，不偏性についてであるが，式 (8) における $\exp(\epsilon_i)$ は対数正規分布に従うため，その期待値は $\exp(\sigma^2/2)$ となる．従って，残差 ζ_i の期待値は $\hat{Y}_i(\exp(\sigma^2/2) - 1)$ となり，これは常に 0 よりも大きくなる．すなわち，このモデルは平均残差が常に正となる過小予測を起こしやすい偏ったモデルであり，これは実用上の問題となる．

2.4 残差のバイアスの補正

前節で述べたように，log-log 重回帰モデルを変形して得られる指数曲線モデルは，過小見積もりとなる傾向を持つ．ところが，従来，ソフトウェア工学分野では，残差にバイアスが生じることも，その補正方法についても，ほとんど論じられていない．本節では，Finney らの提案する補正方法[3][7][9] を述べる．

$\mathrm{E}(\bullet_i)$ を $\bullet_i$ の期待値を表す関数とする．式 (7) より，$Y_i/\hat{Y}_i = \exp(\epsilon_i)$ であるため，次式が成り立つ．

$$\mathrm{E}(Y_i/\hat{Y}_i) = \mathrm{E}(\exp(\epsilon_i)) = \exp\left(\frac{\sigma^2}{2}\right) \tag{9}$$

σ は残差 ϵ_i が従う正規分布の標準偏差であるが，それ自体は未知であるため，式 (10) に示す σ の不偏推定量である SEE (standard error of estimate; 推定値標準誤差) で代替する．

$$\mathrm{SEE} = \sqrt{\frac{\sum(\log Y_i - \widehat{\log Y_i})^2}{r - p - 1}} \tag{10}$$

式 (10) の分母の $r - p - 1$ はモデルの自由度である．r はモデル構築に用いたプロジェクトの件数であり，p は説明変数の数である．この自由度の調整はしばしば見落とされがちであるため[9]，注意が必要である．

式 (9) と式 (10) から，補正係数 CF(correction factor) が次式で与えられる．

$$\mathrm{CF} = \exp\left(\frac{(\mathrm{SEE})^2}{2}\right) \tag{11}$$

バイアスが補正された予測値 $\hat{Y}^*$ は次式で与えられる．

$$\hat{Y}^* = \mathrm{CF} \cdot \hat{Y} \tag{12}$$

3 プロジェクトデータの特徴からの考察

本節では，プロジェクトデータの特徴から，対数変換の妥当性を論じる．Kitchenham と Mendes[6] は，ソフトウェア開発プロジェクトデータの多くは，工数と規模の関係に着目すると，(1) 規模が大きくなるほ

ど工数のばらつきが大きくなる，(2) 規模の小さい部分にプロジェクトが集中している，(3) 外れ値がある，(4) 異なるタイプのプロジェクトが混在している，といった特徴があると述べている．以降では，これらの特徴と対数変換の関係について考察する．

3.1 規模が大きくなるほど工数のばらつきが大きくなる

多くのプロジェクトデータは，規模が大きくなるほど工数が大きくなり，それに伴って工数のばらつきも大きくなる．2.3 節で述べたように，log-log 重回帰モデルを式変形して得られる指数曲線モデルでは，残差が目的変数に比例して大きくなるため，この特徴をうまくモデル化でき，予測性能の向上が期待できる．

3.2 規模の小さい部分にプロジェクトが集中している

多くのプロジェクトデータは，規模の小さい部分にプロジェクトが集中しており，すそ野の広い分布となる．この分布は，対数正規分布に近いため，対数変換を行うことで，正規分布に近くなる．またその結果，線形回帰モデルの予測性能が向上することが多い．

ただし，線形回帰の適用条件として，説明変数が正規分布していることが求められているわけではない [8]．あくまでも，経験的な Tips として，すそ野の広い分布を持つ変数は，対数変換を試してみると良い，ということである．

3.3 外れ値が存在する

一般に，外れ値とは，分布の中心から大きく外れた値のことを指す．3.1 節で述べたように，多くのプロジェクトデータは，規模が大きくなるほど工数のばらつきが大きくなり，外れ値も増える．そのため，残差が目的変数に比例して大きくなる指数曲線モデルの方が，対数変換を行わない naïve な線形回帰モデルよりも，現実をうまくモデル化できると考えられる．

3.4 異なるタイプのプロジェクトが混在している

ここでは典型的な例として，図 1 のように，規模と工数の関係が異なる 2 つのタイプのプロジェクトが混在している場合を考える．

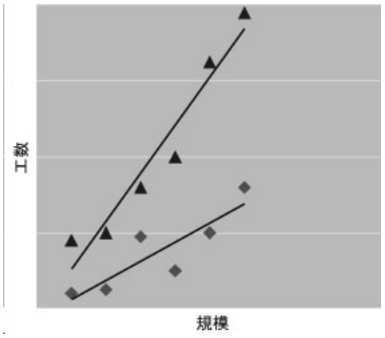

図 1 異質集団を含むデータセット

このような場合，プロジェクトのタイプが 2 値変数で与えられており，かつ，指数曲線モデルを用いた場合に，予測精度の向上が見込まれると考えられる．2.2 節で述べたように，指数曲線モデルの 1 つの特徴は，各説明変数が，加法的ではなく乗法的に結合されるため，回帰曲線の傾きに影響する要因をうまくモデル化できるためである．例えば，アーキテクチャや開発言語など，ソフトウェア開発の生産性に影響を与えている要因を説明変数に用いる場合に，特にうまくモデル化できる．

4 ケーススタディ

4.1 概要

線形回帰における対数変換の効果を分かりやすく示すケーススタディとして，実データ (Desharnais データセット [2]) を用いた残差分析と工数予測の例を示す．Desharnais データセットは，カナダのあるソフトウェア企業の開発実績データであり，無償で一般公開されているため，追実験が可能である．

本ケーススタディでは，過去の実績データを用いて将来のプロジェクトの工数予測を行うことを想定し，モデル構築用のフィットデータとして 1986 年以前のプロジェクト 58 件を用い，モデル評価用のテストデータとして 1987 年以降の 19 件を用いた．目的変数は開発工数である．説明変数は，調整済みファンクションポイント，開発期間，開発チーム経験年数，プロジェクトマネージャ経験年数，開発言語を用いた．開発言語については，2 値変数化した．

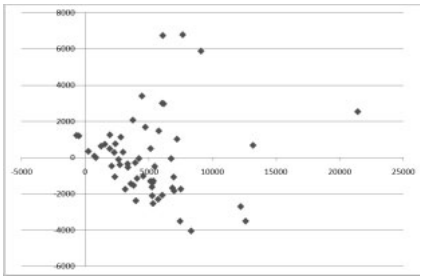

図 2　線形重回帰モデルの残差分布

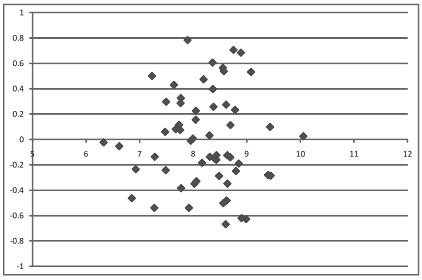

図 4　Log-log 重回帰モデルの残差分布

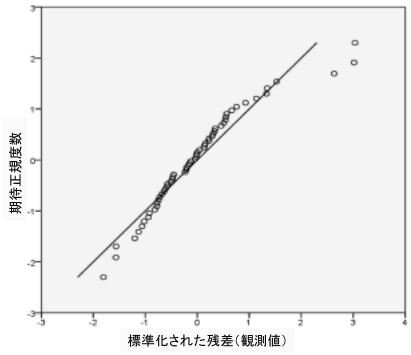

図 3　線形重回帰モデルの残差の正規 Q-Q プロット

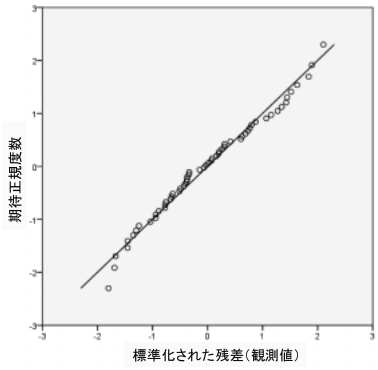

図 5　Log-log 重回帰モデルの残差の正規 Q-Q プロット

4.2　モデルの残差

Naïve な線形回帰モデルの残差の散布図を図 2 に，残差の正規 Q-Q プロットを図 3 に示す．図 2 では，回帰式による開発工数の予測値 $\hat{Y}$ を横軸に，対応する残差の値を縦軸にとっている．図 2 から，開発工数が 0 に近いところでは残差が小さく，開発工数が大きくなるにつれ残差が大きくなっており，残差の等分散性が満たされていないことが分かる．また，図 3 より，各プロットは直線からやや外れており，残差の正規性についても満たされているとはいえない．このことから，この線形回帰モデルは，本データセットをモデル化するのに適していないといえる．

次に，log-log 重回帰モデルの残差の散布図を図 4 に，残差の正規 Q-Q プロットを図 5 に示す．図 4 より，開発工数の大小にかかわらず残差はほぼ同じ幅で分布しており，残差の等分散性がほぼ満たされていることが分かる．また，図 3 と比較すると，図 5 では各プロットは直線上に乗っており，残差の正規性についてもほぼ満たされているといえる．

Naïve な線形回帰モデル，(log-log 重回帰モデルを変形して得られる) 指数曲線モデル，および，2.4 節の方法により残差の補正を行った指数曲線モデルの残差平均を表 1 に示す．表中，MMRE(Mean Magnitude of Relative Error)，MMER(Mean Magnitude of Error Relative) はいずれも相対誤差平均の尺度であり，$\mathrm{MMRE} = \mathrm{E}(|Y_i - \hat{Y}_i|/Y_i)$，$\mathrm{MMER} = \mathrm{E}(|Y_i - \hat{Y}_i|/|\hat{Y}_i|)$ である．また，表中の MR は Mean Residual(残差平均) であり，線形回帰では MR が 0 になるように偏回帰係数が定められる．

表 1 より，MMRE，MMER のいずれの指標においても線形回帰モデルよりも指数曲線モデルが優れており，予測性能がよいことが示唆される．また，MR に着目すると，補正前は MR=291 であり，実測値に対して平均で約 5.8 %の過小予測をするモデルとなっていた．補正後は MR=−88.4 と大幅にバイアスが打ち消されている．相対残差について補正後を補正前と

表 1 各モデルの残差

	MMRE	MMER	MR
線形回帰モデル	0.426	0.404	(0)
指数曲線モデル	0.313	0.324	291
指数曲線モデル (補)	0.347	0.309	−88.4

表 2 各モデルの予測誤差

	MMRE	MMER	MR
線形回帰モデル	0.809	0.784	−242
指数曲線モデル	0.280	0.295	327
指数曲線モデル (補)	0.289	0.284	5.61

比べると，MMRE が少し悪化した一方で，MMER が少し改善している．従来，MMRE は過小予測傾向のモデルを不当に良いと評価してしまうことが知られており [4]，残差の補正によって過小予測傾向が解消されたため，MMRE が増大したと考えられる．

4.3 モデルの予測性能

モデル評価用のテストデータに対する予測を行った結果を表 2 に示す．表 2 より，naïve な線形回帰モデルは，指数曲線モデルと比べて，すべての指標において予測精度が大きく低下していることが分かる．また，MR に着目すると，補正前は明らかなバイアスを持つ問題があるが，補正後はそのバイアスがほとんど打ち消されている．予測誤差について補正後を補正前と比べると，MMRE，MMER は大差ないのでバイアス補正のデメリットはほぼ無い．

モデル構築時の残差平均 (表 1) と誤差平均 (表 2) を比べると，表 2 では，指数曲線モデルの優位性は，さらに大きくなった．このことは，誤ったモデル化を行うことの危険性を示唆している．データセットが当該モデルの仮定に合わない場合，構築したモデルが一定の残差平均を示したとしても，予測時の誤差が顕著となる場合があることを示唆している．

5 まとめ

本論文では，対数変換を行った線形回帰モデルは，指数曲線モデルと等価であり，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を示した．また，得られた指数曲線モデルは，残差にバイアスを生じ，過小見積もりとなる傾向を持つため，その補正方法も示した．ケーススタディでは，構築したモデルの残差分析，及び，残差平均の導出により，naïve な線形回帰モデルよりも指数曲線モデルがより現実をうまく表しており，予測精度も良いことを示した．

対数変換の手続きはごく簡単で効果も大きいため，線形回帰を行う場合には，対数変換を試してみることが望ましいといえる．

謝辞 本研究の一部は，文部科学省「次世代 IT 基盤構築のための研究開発」の委託に基づいて行われた．

参 考 文 献

[1] Boehm, B. W. : *Software Engineering Economics*, Prentice-Hall, 1981.
[2] 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.
[3] Finney, D. J. : On the Distribution of a Variate Whose Logarithm is Normally Distributed, *Journal of the Royal Statistical Society of London*, Series B, No.7 (1941), pp. 155–161.
[4] Foss, T., Stensrud, E., Kitchenham, B. and Myrtveit, I. : A Simulation Study of the Model Evaluation Criterion MMRE, *IEEE Trans. on Software Engineering*, Vol.29, No.11 (2003), pp. 985–995.
[5] Grafen, A., Hails, R.(著), 野間口謙太郎, 野間口眞太郎 (訳) : 一般線形モデルによる生物科学のための現代統計学, 共立出版, 2007.
[6] Kitchenham, B. and Mendes, E. : Why Comparative Effort Prediction Studies May Be Invalid, in *Proc. 5th International Conference on Predictor Models in Software Engineering*, May 2009, Article No.4.
[7] Newman, M. C. : Regression Analysis of Log-transformed Data: Statistical Bias and Its Correction, *Environmental Toxicology and Chemistry*, Vol. 12, No. 6 (1993), pp. 1129–1133.
[8] 奥野忠一, 久米均, 芳賀敏郎, 吉澤正 : 多変量解析法 (改訂版), 日科技連, 1981.
[9] Sprugel, D. G. : Correcting for Bias in Log-transformed Allometric Equations, *Ecology*, Vol. 64, No. 1 (1983), pp. 209–210.
[10] 東京大学教養学部統計学教室編 : 自然科学の統計学, 東京大学出版会, 1992.
[11] 山田茂, 高橋宗雄 : ソフトウェアマネジメントモデル入門, 共立出版, 1993, pp. 36–50.