

重回帰モデルによる熟練者見積もり工数の補正の試み

角田 雅 照^{†1}

本研究では、熟練者による工数見積もりの精度が高い場合に、熟練者見積もりを重視しつつ、数学的モデルにより見積りを補正することを試みる。熟練者見積り工数を説明変数とした、見積もり工数補正モデルを重回帰分析により作成して実験した結果、見積もりの偏りが改善された。

Calibration of expert estimated effort using a multiple regression model

MASATERU TSUNODA^{†1}

In this paper, we give weight to expert estimated effort, and calibrate the estimation by a mathematical model when the estimation accuracy is high. We made effort calibration model using expert estimated effort as an independent variable, and the model improved bias of the expert estimation.

1. はじめに

大規模なソフトウェア開発では、スケジュールやコストの管理は必要不可欠であり、ソフトウェア開発工数見積もりは、その管理の基礎となる。高い工数見積もり精度を実現するために、これまで様々な数学的モデルに基づく工数見積もり方法が提案してきた。

ただし、用いるデータセットやモデルによっては、数学的モデルによる見積もりの精度が必ずしも高くない場合がある。一方、数学的モデル以外による見積もり方法として、熟練者による見積もりがある。熟練者が工数を適切に見積もった場合、数学的モデルによる見積もりと比べて見積もり精度が高くなる場合がある。

本研究では、熟練者による見積もりの精度が数学的見積もりモデルよりも精度が高い場合に着目し、その時に熟練者による見積もりを重視しつつ、数学的モデルを用いることにより見積もり精度を高めることを試みる。具体的には、熟練者による見積り工数を説明変数とした工数見積もりモデル（以降見積もり工数補正モデルと呼ぶ）を、重回帰分析を用いて構築し、そのモデルにより見積り精度が改善するかどうかを確かめる。

2. 分析

2.1. データセット

分析で用いたデータセットは、Kitchenham が 2002 年に文献[3]において公開した、あるソフトウェア開発企

業のデータであり、見積もり工数と見積もり方法が記録された 145 件のプロジェクトが含まれている。このデータから、熟練者による見積もり方法を用いた（すなわち、熟練者による見積もり工数が記録されている）プロジェクト 98 件を抽出し、分析に用いた。

分析に用いた変数は、顧客識別子（顧客コード）、開発種別、見積もり工数、実績工数、FP（調整済みファンクションポイント）、見積もり開発期間、開発速度（FP ÷ 見積もり開発期間）、生産性（FP ÷ 見積り工数）である。一部の変数には欠損値が含まれていた。通常、顧客の違いが見積もりに影響するとは考えにくいため、顧客識別子は工数見積もりモデルの説明変数として用いないが、熟練者による見積もりの場合、顧客との関係性が何らかの影響を与える可能性があるため、分析対象の変数に含めた。

2.2. 予備分析

予備分析では、熟練者による見積もりの精度と各変数との関連を分析し、見積もり工数補正モデルの説明変数の候補とすべき変数を絞り込んだ。見積もり精度の評価尺度として、BRE（balanced relative error）を用いた。BRE は見積もり値 > 実測値の場合は|見積もり値 - 実測値|÷ 見積もり値、それ以外の場合は|見積もり値 - 実測値|÷ 実測値により計算される。

関係の分析では、分散分析の調整済み寄与率とスピアマンの順位相関係数を用いた。前者は名義尺度と比例尺度の変数の関連、後者は比例尺度の変数間の関連を分析する際に用いる。前者は値が 1 に近いほど、後者は絶対値が 1 に近いほど関連が強いことを示す。

^{†1} 東洋大学 総合情報学部

Faculty of Information Sciences and Arts, Toyo University

表 1 BRE と各変数との関連

	FP	見積もり 開発期間	開発速度	生産性
相関係数	-0.08	-0.07	-0.01	-0.02
	顧客識別子	開発種別		
寄与率	0.38		0.09	

表 2 各モデルの見積もり精度

	BRE		BRE bias	
	平均値	中央値	平均値	中央値
補正	28.0%	16.4%	0.3%	1.5%
熟練者	28.8%	16.0%	-13.1%	-4.8%
合成	51.9%	28.5%	-22.3%	-5.7%
FP	96.3%	51.6%	-18.3%	5.8%

BRE と各変数との寄与率と相関係数を表 1に示す。顧客識別子のみ BRE と強い関連があつたため、見積もり工数補正モデルの説明変数の候補として、顧客識別子を採用する。

2.3. 見積もり精度の比較

顧客識別子と見積工数を説明変数として、見積もり工数補正モデルを構築した。モデルを構築、評価するために、5-fold cross validation を適用した(試行回数を増やすために4回繰り返して適用した)。顧客識別子はダミー変数化し、AIC(赤池情報量規準)に基づくステップワイズ変数選択を行つた。

精度評価指標として BRE に加え BRE bias を用いた。BRE bias は見積もり値 > 実測値の場合は(見積もり値 - 実測値)÷見積もり値、それ以外の場合は(見積もり値 - 実測値)÷実測値により計算され、見積もりの偏り(過大、過小見積もりに偏っているか)評価に用いる。評価では BRE と BRE bias の平均値と中央値を求めた。

複数の見積もり結果を合成することにより、見積もり精度が改善する可能性がある[2]。そこで、見積もり工数補正モデルとの比較のため、熟練者による見積もり工数と、FP を用いた重回帰モデル(一般的な数学的の見積もりモデル)による見積り工数の平均値を、最終的な見積りとするモデルの精度を評価した。さらに、FP を用いた重回帰モデルについても比較対象とした。

各モデルの見積もり精度を表 2に示す。FP を用いたモデルの精度が最も低く、見積もり結果を合成するモデルの精度も高くなかった。熟練者による見積もりと、見積もり工数補正モデルとの精度を比較すると、BRE にはほとんど差がなかつたが、BRE bias は後者のほうが改善していた。Wilcoxon の符号付順位和検定により両者の差を確かめると、平均値、中央値とも有意水準 5%で差があつた。実験結果より、以下のことがいえる。

- 見積もり工数補正モデルを用いることにより、熟練者による見積もりの精度を改善することはできないが、偏りを改善することはできる可能性がある。
- 熟練者による見積もりの精度が高い場合、単純に複数の見積り結果を合成するだけでは精度が改善しない可能性が高い。

3. 関連研究

Jørgensen[1]は熟練者による見積もりの誤差原因を明らかにするために、見積もり誤差を目的変数とし、説明変数にユーザの優先項目(コスト重視か、など)などを用いて重回帰分析を行なつてゐる。ただし、モデルの説明力は低かった。見積もり工数補正モデルとの最も大きな違いは、Jørgensen は見積り工数を説明変数に用いていない点である。本研究の最も重要な貢献は、熟練者による見積もりの精度が高い場合に、熟練者による見積り工数を説明変数に用いて、見積もり工数が補正できる可能性があることを示したことである。

4. おわりに

見積もり工数補正の結果を熟練者にフィードバックすることにより、熟練者の見積もりが変化する可能性があるため、実際に適用する際には時間的に新しいデータのみを用いてモデルを構築する方法[4]を併用したほうがよい可能性がある。ワークショップでは、熟練者による判断を重視しつつ、数学的モデルによりプロジェクト管理を支援するにはどうすべきかについて議論したい。

参考文献

- [1] Jørgensen, M.: Regression Models of Software Development Effort Estimation Accuracy and Bias, *Empirical Software Engineering*, Vol. 9, Issue 4, pp. 297-314(2004).
- [2] Jørgensen, M.: Practical Guidelines for Expert-Judgment-Based Software Effort Estimation, *IEEE Software*, Vol. 22, No. 3, pp. 57-63 (2005).
- [3] Kitchenham, B., Pfleeger, S., McColl, B., and Eagan, S.: An empirical study of maintenance and development estimation accuracy, *J. of Systems and Software*, Vol. 64, Issue 1, pp. 57-77 (2002).
- [4] Lokan, C. and Mendes, E.: Applying moving windows to software effort estimation, Proc. International Symposium on Empirical Software Engineering and Measurement (ESEM), pp. 111-122, Washington DC, USA (2009).