

ソフトウェア開発工数見積りにおける時間情報の扱い

角田 雅照^{†1} 天寄 聡介^{†2}

本稿では、ソフトウェア開発工数見積りモデルにおいて、時間情報(プロジェクトの実施時期)をどう扱うべきかを議論するために、Moving Window 法をはじめとする4つの手法を適用し、それぞれの見積り精度を比較する。

Handling Timing Information for Software Development Effort Estimation

MASATERU TSUNODA^{†1} and SOUSUKE AMASAKI^{†2}

The aim of this paper is to discuss how to handle timing information (performed timing of project) in the software effort estimation. For the discussion, we applied four handling methods such as Moving Window method to effort estimation models, and compared their accuracies.

1. はじめに

ソフトウェア開発工数の見積りは過去の開発データに基づいて行うことが望ましい。しかし、開発プロセスや利用ツールなどは時間とともに変わるため、直近のプロジェクトとそれ以前のプロジェクトでは、説明変数(開発規模など)と目的変数(工数)との関係も変化している可能性がある。このため、一定期間より過去のデータを使わず、直近のデータのみを用いることで工数見積りの精度が向上する可能性がある。Locanら[1]はMoving Window 法(以降 MW 法とする)を用いて、直近のデータのみによるモデルを構築することにより、見積り精度が高まることを示している。

本稿では、プロジェクトの時間情報(実施時期)をモデルで扱うための複数の手法の適用を試み、手法によって見積り精度に差が出るかどうかを確かめる。

2. 時間情報の扱い

本稿では以下の4つの手法で時間情報を扱った。

- **Moving Window 法:** 直近の n 件のプロジェクトのみを用いてモデルを構築する(表 1の例では、 $n=2$ の場合 P005 と P006 を用いる)。
- **Moving Window ダミー変数化:** MW 法はデータの層別とみなすことができるが、層別よりもダミー変数化のほうが工数見積りの精度が高くなる

^{†1} 東洋大学
Toyo University

^{†2} 岡山県立大学
Okayama Prefectural University

場合がある[2]。MW 法のアイデアをベースとして時間情報をダミー変数化することとし、実施時期順に n 件のプロジェクトごとにダミー変数を設定する(表 1(a)。 $n=2$ としている)。

- **実施年ダミー変数化:** プロジェクトの実施年に基づいたダミー変数を作成し、説明変数とする(表 1(b))。
- **シリアル値:** ある基準日からの日数の増加分を数値とし、説明変数として用いる。例えば基準日が2000年1月1日の場合、2000年1月2日のシリアル値は2となる(表 1(c))。

交互作用項(2つの説明変数の積を用いた新たな説明変数)を用いることにより、工数見積りの精度が高くなる場合がある[2]。そこで MW 法以外の手法については、時間情報に基づく変数と開発規模の交互作用項を追加した場合の効果についても確かめる。

3. 実験

ISBSG (International Software Benchmarking

表 1 時間情報の扱い

Project ID	実施年月	(a)		(b)		シリアル値
		Win 1	Win 2	2012	2011	
P001	2010-4	0	0	0	0	40275
P002	2011-12	0	0	0	1	40899
P003	2011-3	0	1	0	1	40610
P004	2012-1	0	1	1	0	40922
P005	2012-6	1	0	1	0	41080
P006	2012-7	1	0	1	0	41098

Standards Group)が収集したデータセットを用いて実験を行った(バージョンは Release 9 である). プロジェクトの条件を整えるため, Locan ら[1]が示した条件(データ品質が A または B など. 一部条件が異なる)を参考に 502 件のプロジェクトを抽出した. 用いた説明変数は未調整 FP, 開発種別, 開発プラットフォーム, 開発言語, 業種である. カテゴリ変数はダミー変数化し, 工数と未調整 FP については対数変換を行った.

時間情報を扱う各手法について, 以下の手順により見積もりモデルを構築し, 精度を評価した.

1. プロジェクトを開発完了日(開発開始日+開発期間により算出)で並べ替える.
2. 先頭(最古のプロジェクト)から数えて k 件目のプロジェクトをテストデータ(見積もり精度評価用), $k - 1$ 件目までをフィットデータ(モデル構築用)とし, モデルの構築及び精度の評価を行う.
3. $k <$ データ件数なら, k を 1 増やして 2. に戻る.

MW 法及び Moving Window ダミー変数化(以降 MW ダミー変数化とする)については, 直近 n 件(window size)を 20 から 100 まで 10 刻みで変化させて実験した. k は 101 を初期値とした(n に関わらずテストデータを同一にするため). モデル構築時には AIC による変数選択をし, 見積もり精度の評価指標として BRE(Balanced Relative Error)の平均値と中央値を用いた.

各手法適用時の工数見積もりモデルの精度を表 2 に示す. 実験では以下の結果となった.

- MW 法, シリアル値(交互作用項あり/なし)では, 時間情報を使わないモデルよりも工数見積もり精度が悪化した.
- MW ダミー変数化(交互作用項あり/なし)では, 精度が高い場合($n = 20$ など)があったが, 有意水準 5% で差があるといえなかった.
- 実施年ダミー変数化については, 交互作用項を追加した場合のみ有意に改善していたが($p = 0.04$), 改善の度合いは大きくなかった.

4. おわりに

本稿では, 工数見積もりモデルにおける時間情報の扱いの違いが見積もり精度に及ぼす影響を確かめた. 実験において精度の改善度合いは小さかったが, これは企業横断的なデータを用いたことが影響している可能性がある. よって, 単一の企業データを用いることで手法の違いによる差が大きくなる可能性もある. ワークショップでは工数見積もりモデルにおいて時間情報をどのように扱うべきか(プロセス改善などの実施タイ

表 2 工数見積もりの精度

	BRE 平均値	BRE 中央値
時間情報なし	153%	84%
実施年ダミー変数化	158%	80%
実施年ダミー変数化 交互作用	153%	80%
シリアル値	155%	85%
シリアル値 交互作用	156%	85%
MW 法		
$n = 20$	237%	92%
$n = 30$	214%	85%
$n = 40$	215%	94%
$n = 50$	188%	95%
$n = 60$	166%	88%
$n = 70$	160%	84%
$n = 80$	161%	87%
$n = 90$	163%	92%
$n = 100$	157%	87%
MW ダミー変数化		
$n = 20$	149%	82%
$n = 30$	153%	80%
$n = 40$	152%	85%
$n = 50$	154%	86%
$n = 60$	157%	88%
$n = 70$	154%	86%
$n = 80$	153%	84%
$n = 90$	155%	85%
$n = 100$	151%	83%
MW ダミー変数化 交互作用		
$n = 20$	150%	81%
$n = 30$	158%	82%
$n = 40$	154%	85%
$n = 50$	156%	83%
$n = 60$	156%	86%
$n = 70$	152%	85%
$n = 80$	154%	84%
$n = 90$	154%	85%
$n = 100$	153%	86%

ミングに応じて, ウィンドウサイズを動的に変化させるべきかなど)について議論したい.

参考文献

- [1] 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).
- [2] Tsunoda, M., Amasaki, S., and Monden, A: Handling Categorical Variables in Effort Estimation, Proc. International Symposium on Empirical Software Engineering and Measurement (ESEM) (to appear).