Vol.2011-SE-173 No.9
2011/7/21

単一組織における windowing approach による工数見積もりモデルの有効性評価

高原洋平[†1]　天嵜聡介[†1]

工数見積もりモデルの構築において，直近の開発終了プロジェクトのみを訓練に用いる windowing が有効であると考えられている．実証的に，過去の研究では ISBSG R10 データを用いて windowing approach の有効性が考察されている．しかしながら，この研究では単一の組織のデータセットを用いた検証が行われていない．本研究では，単一の組織でのデータセットにおける windowing approach の有用性及び過去の研究結果の一般性について調査を行った．この調査では CSC と Maxwell の 2 つのデータセットを用意し，過去の研究と同様の方法により実験を行った．調査の結果，windowing approach による予測性能の改善が単一の組織のデータセットにおいても見られた．この結果は実際的な状況下での windowing approach の有効性の理解に貢献すると考えられる．

Performance Evaluation of windowing Approach on Single Company Effort Estimation

YOHEI TAKAHARA[†1] and SOUSUKE AMASAKI[†1]

Background: In effort estimation model construction, it seems effective to window training project data so that recently finished projects are only used. The past study examined this windowing approach with ISBSG R10 Data. However, this approach has not been validated with single company dataset. Aim: To investigate effects of windowing approach at a company and generality of observations in the past study. Method: We replicated the past study with two other datasets: CSC and Maxwell datasets. Results: windowing approach improved predictive performance. Conclusions: This result contributes to understand the effects of windowing approach under practical situation.

†1 岡山県立大学
Okayama Prefectural University

1. はじめに

ソフトウェア工数見積もりモデルの研究では，leave-one-out 法や random holdout 法などの交差検証法によって工数見積もりモデルの予測性能を評価することが一般的である．しかしながら，いくつかの研究では現実のソフトウェア開発組織の成長過程を模した形で評価を行うため，プロジェクトデータのリポジトリの成長過程に沿って性能評価を行うことがある．この評価方法では，まずプロジェクトデータを時系列に並び替える．そして，ある時点を起点として，古いプロジェクトデータで訓練した工数見積もりモデルで新しいプロジェクトの工数を見積もる．見積もり対象のプロジェクトをより新しいプロジェクトに変更しつつ，このような見積もりを繰り返す．このとき，古いプロジェクトデータ全てを工数見積もりモデルの訓練に用いる方法と，一定以上古いプロジェクトは除いて訓練を行う方法の二通りのアプローチが考えられる．本論文では前者を growing approach，後者を windowing approach と呼ぶ．

図 1 に growing approach 及び windowing approach による工数見積もりモデルの構築法の違いを示す．図 1 の各矢印はソフトウェア開発組織における時間軸を表す．矢印上の黒丸は，ある時点における工数見積り対象のプロジェクトを表す．また，矢印上の白丸は，その時点においてすでに完了しているプロジェクトを表す．growing approach（**図 1(a)**）では，工数見積もりの対象プロジェクトの開始前に完了した全てのプロジェクトを用いて工数見積もりモデルを構築する．対照的に，windowing approach では，過去に完了したプロジェクトを，最近完了したプロジェクトと古いプロジェクトに分類する．**図 1(b)** では×印が古いプロジェクトに対応する．工数見積もりモデルの構築には白丸のプロジェクトデータのみを用いる．このとき，白丸の数を規定するのが Window 幅である．

growing approach はソフトウェア開発組織を取り巻く環境や組織自体が安定している場合に合理的であると言える．一方，windowing approach は，ソフトウェア開発組織を取り巻く環境が頻繁に変化しており，古いプロジェクトが組織の代表的なデータと言えない場合に合理的だと考えられる．windowing approach が有効な状況がある一方で，このアプローチは古いプロジェクトデータを利用しないため，モデル構築に利用できるプロジェクト数が限られるという特徴がある．工数見積もりモデルとして線形回帰モデルを使用した場合などは，プロジェクトデータ数が少ない場合に複雑なモデルを作成することが困難となる．しかし，単純なモデルは見積が不正確となる可能性も存在する．

Lokan[1)] らは windowing approach の有効性について評価を行った．彼らの研究結果では，

growing approach より windowing approach の方が予測性能が良くなるような Window 幅を「スイートスポット」と呼んでいる．ISBSG R10 データを用いた評価実験により，スイートスポットが存在することが示されていた．これは，適切に古いプロジェクトを取り除くことができれば windowing approach が有効な場面があることを示す．

この研究結果は興味深いものであるが，ISBSG Data という複数の組織から収集されたデータセットに基づいて評価実験が行われている点で，その結論の一般性及び実用性について議論の余地がある．まず，ソフトウェア開発組織は自社のプロジェクトデータから工数見積もりモデルの構築を行うことが一般的である．複数組織のデータセットを用いた場合と自社のデータセットを用いた場合との比較評価の研究は行われているものの，その優劣についての評価は定まっていないのが現状である．次に，単一の組織のデータに注目した場合，複数組織から収集したデータセットに比べてプロジェクト間の差異は少なくなる傾向がある．また，一定の期間内に実施されるプロジェクトの数が相対的に少ないため，同程度の複雑さのモデルの構築に必要なデータの収集期間がより長くなる可能性がある．この場合，windowing approach によってソフトウェア開発組織を取り巻く環境の変化に追随したモデルの構築が困難になる可能性がある．

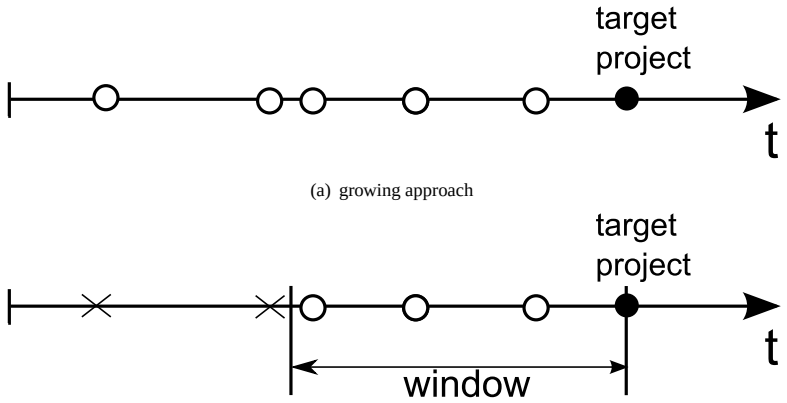
以上の理由より，我々は単一の組織における windowing approach の有効性の評価した．この目的のため，CSC と Maxwell の二つのデータセットを用いて Lokan らの研究と同じ手順に従って評価実験を行った．

2. データセット

本研究では PROMISE repository[2] やサーベイ論文[3] を元に，一般公開されているデータセットのうちから単一組織でプロジェクトデータを収集しているものを選別した．最終的に，CSC[4]，Maxwell[5] の 2 つのデータセットを windowing approach の有効性評価に用いた．これらのデータセットは既存の工数見積もりモデル研究において頻繁に利用されているデータセットであり，いずれも単一の組織によって収集されている．

2.1 データセット CSC

データセット CSC は 1994 年から 2002 年にかけて Computer Science Corporation 社において収集されたデータセットである．データセット CSC は 145 のプロジェクトデータを含む．各プロジェクトは新規開発か既存アプリケーションの保守開発いずれかに分類できる．製品の規模は調整済みファンクションポイント法で計測されており，工数は人時で計測されている．このデータセットにはクライアント 1 からクライアント 6 まで 6 種類のクライアント向けプロジェクトが混在している．



(a) growing approach

(b) windowing approach

図 1 各アプローチのモデル構築法
Fig. 1 growing and windowing approaches

文献 4) の著者らは評価実験にクライアント 1 とクライアント 2 向けのプロジェクトを評価実験に用いている．しかし，各クライアント向けに作成された工数見積もりモデルのパラメータは異なっており，また，クライアント 1 向けのプロジェクト数は 16 個と少ない．以上の点から，クライアントに関してさらに層別化することにより，評価実験に用いるデータセットがより均質な集合になると期待できる．

本研究ではクライアント 2 向けのプロジェクトのみを評価実験に用いた．ただし，クライアント 2 向けのプロジェクトのうち明らかな外れ値 1 つを除いた．このプロジェクトは様々な意味で通常のプロジェクトとは異なっている．例えば，このプロジェクトは CSC 社におけるクライアント-サーバ構成による初めてのプロジェクトであり，同種のプロジェクトの経験がないにもかかわらず比較的大規模であったことが文献 4) で指摘されている．

最終的に選択されたプロジェクトは 38 の新規開発プロジェクトと 67 の保守プロジェクトの合計 105 プロジェクトである．ただし，予測因子として，ファンクションポイント及びプロジェクトタイプ (新規あるいは保守) を使用した．文献 4) によると，このデータセットでは生産性が時期によって変化していることが確認されている．つまり，ソフトウェア開発組織を取り巻く環境もしくは開発組織自身が期間内に変化しており，windowing approach

が growing approach よりよい選択である可能性がある．

2.2 データセット Maxwell

データセット Maxwell は文献 5) に記載されているものである．このデータセットは 1985 年から 1994 年にかけてフィンランドで最も大きな商業銀行の一つで収集された．このデータセットには 63 個のプロジェクトが含まれている．各プロジェクトではアプリケーションの種類やスタッフの技能レベルなどを示す 22 個の情報が記録されている．製品規模はファンクションポイント法により計測され，工数は人時で計測されている．データセット CSC とは対照的に，このデータセットには新規や保守といったプロジェクトの種別は記録されていない．また，予測因子の候補となる変数が多いため，windowing approach を採用した際に backward のステップワイズ回帰によるモデル選択[6] を行うことは困難である．そこで，あらかじめデータセット全てを用いて予測因子を選択した．

最終的に，本研究ではデータセットから外れ値 1 つと GUI 製品のプロジェクト 4 つを除いた 58 のプロジェクトを評価実験用に選択した．そして，予測因子にはファンクションポイント，要求の変動の度合い (T08)，要求品質 (T09) の 3 つを選択した．T08 及び T09 は順序尺度である為，これらを連続変数に変換して調査に利用した．文献 4) と同じく，このデータセットを大きさの等しい 4 つのサブセットに分割してそれぞれで回帰モデルを作成した結果，回帰モデルのパラメータの推定値がサブセット間で変化していることを確認した．このことから windowing approach は growing approach より優れている可能性が考えられる．

表 1 は CSC，Maxwell，及び文献 1) で使用された ISBSG データセットの各メトリクスの統計要約量である．工数の単位は全て人時であり，また，規模は全てファンクションポイント法によって測定されている．データセット CSC は全体的に ISBSG 10 データより小さなプロジェクトで構成されている．一方でデータセット Maxwell は全体的に ISBSG 10 データより大きなプロジェクトで構成されている．規模と工数の標準偏差を平均値で割った値で定義される変動係数を見ると，これらのデータセットは ISBSG 10 データよりばらつきが小さかった．また，データセット Maxwell はデータセット CSC よりデータにばらつきが大きかった．

3. 実　　験

3.1 モデル構築

本研究では，文献 1) と同様に自動的に工数見積もりモデルの構築を行った．構築した全てのモデルは予測因子として規模を含んでおり，最も単純なモデルは以下のようになる．

表 1 データセット毎のメトリクス統計要約量
Table 1 Summary statistics for datasets

Datasets	Vals.	Mean	Median	StDev	Min	Max
ISBSG[1)]	規模	496	266	699	10	6294
(N=228)	工数	4553	2408	6212	62	57749
CSC	規模	395	258	388	30.0	2076
(N=105)	工数	2461	1650	2575	219	15670
Maxwell	規模	655	377	797	48	3643
(N=58)	工数	8426	5190	10794	583	63694

$$\log(\text{Effort}) = \beta_0 + \beta_1 \log(\text{Size}) \ . \tag{1}$$

訓練に N 個のデータが与えられると，$N/10$ 個の予測因子の全ての組み合わせで backward のステップワイズ回帰を行い，最も当てはまりが良い予測因子の組み合わせを採用する．ただし，必ず規模を予測因子に含む．規模以外の予測因子の候補は本質的には名義尺度か順序尺度であったため，対数変換などは行わずにモデルに追加した．また，外れ値については文献 5) と同様にクックの距離に基づいて判断して訓練データから取り除いた．具体的には，クックの距離が $3 \times 4/N$ 以上のプロジェクトを取り除いた．また，$4/N$ 以上 $3/N$ 以下の距離を備えたプロジェクトについては一時的に取り除いた上でモデルをいったん構築する．その結果，調整済み R^2 を基準として性能がより悪かった場合には，取り除かれたプロジェクトを元に戻した．

3.2 性 能 指 標

性能の測定には文献 1) と同様に MMAE，MMRE 及び PRED(25) を用いた．MMRE 及び MMAE はそれぞれ MRE(相対誤差の大きさ) と MAE(絶対誤差の大きさ) の算術平均である．MRE には，最小値を過大に見積もる偏りが存在する．一方で MAE は過大評価や過小評価が存在しない．また，PRED(25) は，0.25 未満の MRE となるプロジェクトの割合として定義される．さらに，評価実験では Wilcoxon の順位和検定を用いて比較を行った．これは，各プロジェクトに対する見積もりが growing approach と windowing approach との間で対応があるためである．ただし，テストプロジェクトのサイズが ISBSG データと比較して小さいため，有意水準には $\alpha = 0.1$ を採用した．

3.3 Moving Window による評価実験の方法

評価実験には文献 1) と同様の手順を用いた．以下にその手順を示す．

(1)　全てのプロジェクトを開発開始日時でソートする．
(2)　Window 幅を w とした場合に，少なくとも w 個のプロジェクトが完了した時点の直

近に開始されるプロジェクト p_0 を見つける．

(3) プロジェクト p_i の工数見積もりを windowing approach と growing approach で行う．

(4) 次に早く完了したプロジェクト p_i を手順 2 と同様の順で探す．

(5) 最新のプロジェクトまで手順 3，4 を繰り返す．

(6) 性能指標と統計的検定を用いて見積もり結果の評価を行う．

文献 1) と同様に Window 幅はプロジェクト数を基準として設定する．この定義では一定期間で行われるプロジェクトの数によって期間の長さに違いが生じるが，一方で訓練に用いるデータセットの数が安定しやすい利点がある．

文献 1) と同様にスイートスポットを見つけるために Window 幅を様々に変化させて上記の実験を繰り返した．このとき最小の Window 幅は，実験手順中に作成される回帰モデルが全て統計的に有意であるよう最小のサイズを採用した．対照的に，最大の Window 幅は統計的検定に必要となるテストプロジェクトの必要数から決定した．その結果，データセット CSC と Maxwell の Window 幅の最小値と最大値はそれぞれ 10-84 と 10-37 となった．

4. 結　　果

4.1 データセット CSC

表 2 に Window 幅毎の MMAE の一部を示している．今回の実験では $20 \leq w \leq 41$ と 84 において統計的有意性が見られた．**図 2(a)** は横軸に Window 幅，縦軸に growing approach と windowing approach による見積もりの絶対誤差の平均値の差をプロットしたものである．縦軸における位置が 0 より大きい場合は growing approach の方がよい性能を示しており，0 より小さくなると windowing approach の方がよい性能を示す．図 2(a) より，Window 幅が 84 の時が最も windowing approach にとって性能が良いスイートスポットを表しているように見える．しかしながら, MAE の平均値だけでなく中央値も考慮すると，$23 \leq w \leq 41$ もスイートスポットの候補を含むと考えられる．統計的検定は両方の Window 幅の範囲にスイートスポットが存在している事を示していた．ただし，growing approach が統計的に良い性能を示す Window 幅も同じ範囲にあった．

表 4 は Window 幅毎の MMRE を表す．MMRE の場合も $20 \leq w \leq 41$ と 84 において windowing approach が統計的に良い性能を示していた．**図 2(b)** は横軸に Window 幅，縦軸に growing approach と windowing approach による見積もりの MMRE の差をプロットしたものである．図 2(b) からは windowing approach は $w = 83, 84$ でのみより良い性能を示しているように見える．しかし，PRED(25) について同様のプロットを行うと，$30 \leq w \leq 32$ もスイートスポットを含む可能性が高い．

表 2　Window 幅別の MMAE (CSC)
Table 2　Mean absolute residuals with different window sizes (CSC)

Window size (w)	Testing projects	With window	Without window	p.value
20	75	1006	953	0.08
30	68	1005	926	0.01
40	53	1145	1092	0.09
50	31	1000	907	0.61
60	29	865	841	0.68
70	20	831	769	0.14
80	15	929	929	0.43

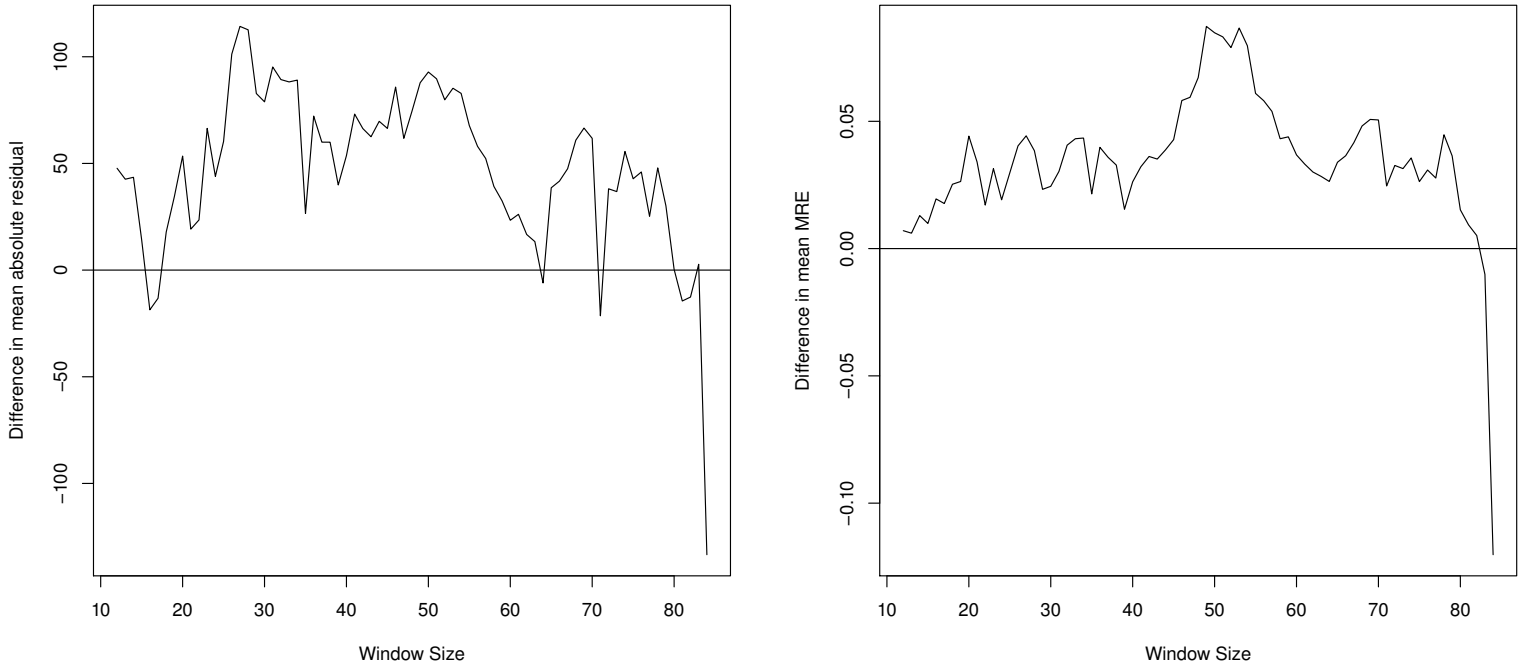
表 4　Window 幅別の MMRE (CSC)
Table 4　Mean MRE with different window sizes (CSC)

Window size (w)	Testing projects	With window	Without window	p.value
20	75	0.50	0.45	0.04
30	68	0.42	0.40	0.10
40	53	0.42	0.39	0.10
50	31	0.48	0.40	0.57
60	29	0.43	0.40	1.00
70	20	0.49	0.44	0.20
80	15	0.54	0.53	0.50

結局，データセット CSC におけるスイートスポットは $w = 30$ であると推測した．しかしながら，この場合の回帰エラー特性曲線 (REC) を描いた場合でも明確な違いを示さなかった．

4.2 データセット Maxwell

表 5 は Window 幅毎の MMAE の結果を示している．今回の実験では windowing approach は $w = 20$ の周辺で悪く，$w = 18$ 及び $w = 35$ の周辺で良い結果が得られた．統計的検定の結果によると，有意水準 $\alpha = 0.1$ で有意な差を示す Window 幅が 2 つ存在したが，いずれも growing approach を支持するものであった．**図 3** は，横軸に Window 幅，縦軸に growing approach と windowing approach による見積もりの MMAE の差をプロットしたものである．文献 1) と同様に，Window 幅が大きくなるにつれて growing approach と windowing approach の性能差が小さくなっている．MAE の中央値による同様のプロットの傾向を考慮すると，統計的な差はみられなかったものの，windowing approach は $35 \leq w \leq 37$ の範囲において growing approach より良い結果を示すと考えられる．



(a) Absolute residuals

(b) MMRE

図 2 Window 幅毎の測定精度の差 (CSC)
Fig. 2 Difference in accuracy measures, by window size (CSC)

表 6 は，Window 幅毎の MMRE の結果を表している．今回の実験では，MMRE に関して windowing approach は全て growing approach よりも悪い結果となった．しかし，同様のプロットを PRED(25) で行った場合，windowing approach は growing approach より良い性能を示す場合が多かった．このことから，一方のアプローチがもう一方のアプローチより良い性能を示すとは言い難い．しかし，もし windowing approach がデータセット Maxwell においてスイートスポットを持つと仮定すると，$w = 36$ に存在すると言える．MMRE を除く評価指標において growing approach より多少ながら良い性能を示しているためである．$w = 36$ における平均期間はおよそ 4 年であった．

図 4 は Window 幅が 36 の時の MAE の REC 曲線を表している．REC 曲線の場合，線の軌跡が左上端に近いほど良い．windowing approach の曲線は必ずしも growing approach の曲線より左上に接近するとはいえない．つまり，windowing approach を採用することが明確な差となるとは言えない．MRE についての REC 曲線も同様の結果を示していた．しかし

表 5 Window 幅別の MMAE (Maxwell)
Table 5 Mean absolute residuals with different window sizes (Maxwell)

Window size (w)	Testing projects	With window	Without window	p.value
15	29	4094	3437	0.30
20	24	4398	2749	0.20
25	21	2839	2773	0.81
30	14	1467	1235	0.67
35	10	1201	1259	0.75

表 6 Window 幅別の MMRE (Maxwell)
Table 6 Mean MRE with different window sizes (Maxwell)

Window size (w)	Testing projects	With window	Without window	p.value
15	29	0.53	0.44	0.20
20	24	0.53	0.41	0.17
25	21	0.41	0.37	0.86
30	14	0.34	0.29	0.77
35	10	0.43	0.37	0.90

ながら，一定の効果はあると言える．

5. 考　　察

いずれのデータセットにおいても windowing approach が growing approach よりも良い性能を示す Window 幅があった．また，統計的な有意差はなかったものの，Maxwell データセットを用いた実験の方がスイートスポットの効果がより明確であった．Window 幅の変化に対する growing approach と windowing approach の性能の差の傾向はデータセット間で対照的であった．データセット CSC においてはより小さい Window 幅の場合に，スイートスポットが見つかり，データセット Maxwell においてはより大きい Window 幅の場合にスイートスポットが見つかった．

文献 1) では ISBSG 10 データにおけるスイートスポットに対応する Window 幅のプロジェクトデータ収集期間はおよそ 1 年である．データセット CSC におけるスイートスポットに対応するプロジェクトデータの収集期間もおよそ 1 年である．対照的に，データセット Maxwell ではおよそ 4 年となる．

データセット CSC とデータセット Maxwell の主な違いはサンプルサイズの大きさと予測因子の数である．データセットはどちらもほとんど同じ期間に収集されている．しかしながらデータセット CSC はデータセット Maxwell のおよそ 2 倍の数のプロジェクトデータを含んでいる．予測因子に関しては，データセット Maxwell では 2 つの有効な予測因子を含むことが可能なのに対し，データセット CSC では，有効でない予測因子 1 つしか含むことができない．有効な予測因子は見積もり性能を向上させる．しかし，モデル構築にはより多くの訓練データが必要となる．これはデータセット Maxwell においてスイートスポットとなる Window 幅が大きい原因と考えられる．そこで，データセット Maxwell を式 (1) に示す単純なモデルのみを用いて再度実験を行った．

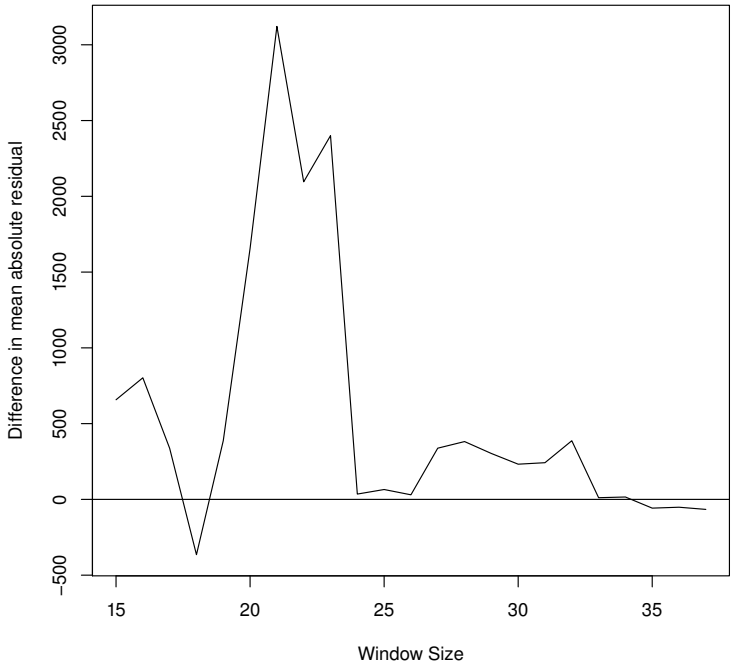


図 3 Window 幅毎の MMAE の差 (Maxwell)
Fig. 3 Difference in absolute residuals, by window size (Maxwell)

Vol.2011-SE-173 No.9
2011/7/21

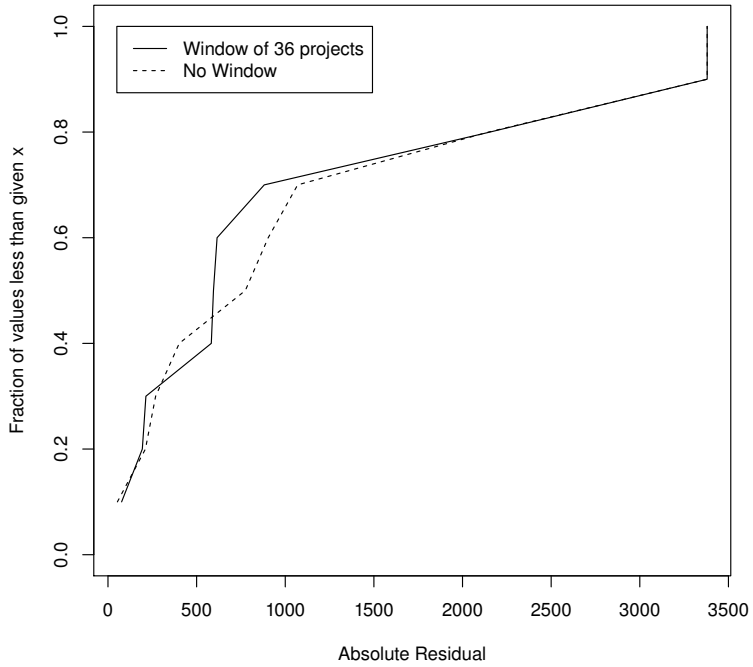


図 4 Window 幅 36 における MMAE の REC 曲線 (Maxwell)
Fig. 4 REC curves of absolute residuals, window of 36 projects (Maxwell)

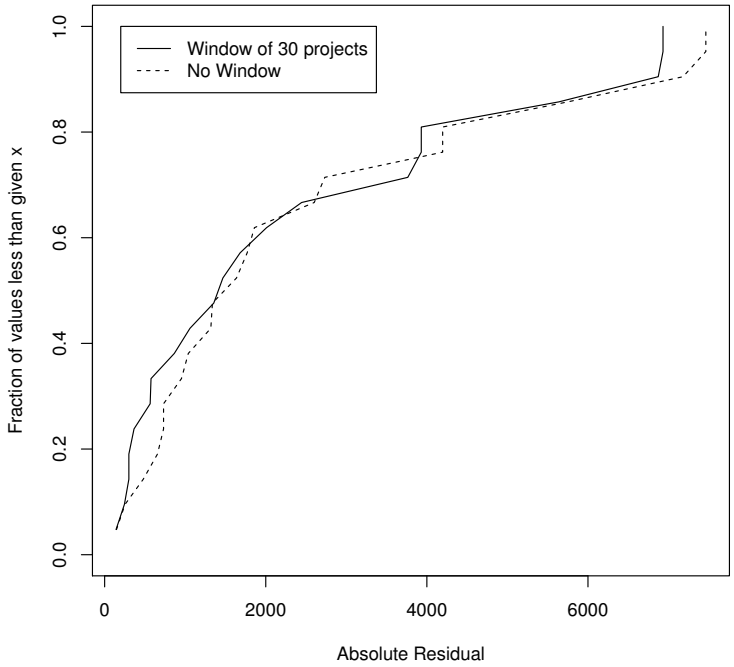


図 5 Window 幅 30 における単純なモデルを用いた場合の MMAE の REC 曲線 (Maxwell)
Fig. 5 REC curves of absolute residuals, simple model, window of 30 projects (Maxwell)

実験の結果，スイートスポットは MAE では $w = 30$，MRE では $w = 32$ という結論に至った．また，予測因子を複数含みうる場合と比べて，windowing approach は統計的により良い性能を示した．$w = 30$ 及び $w = 32$ の場合，プロジェクトデータの収集に必要な期間は平均で約 3.5 年となる．したがって，予測因子の追加は最適なスイートスポットとなる Window 幅の大きさに影響を及ぼしていたと言える．しかし，4 年と 3.5 年では大きな差とは言えないため，予測因子の追加がデータセット間の対照的な結果の主要な原因とは言えない．この点については他の原因を調査する必要がある．

図 5 に $w = 30$ のときの MAE の REC 曲線を示す．全体的に windowing approach の方が性能が高いことから，growing approach より良い方法であると言える．ただし，図 4 と図 5 の横軸が示すように，単純なモデルでは誤差も大きくなるため，性能の面では予測因子を含むモデルの方が良いと言える．

6. 結　　論

windowing approach を採用することで単一の組織向けの工数見積もりモデルでも予測精度を改善させることができることが明らかとなった．より多くのプロジェクトデータを必要とする複雑なモデルではスイートスポットと呼べる Window 幅は大きくなる傾向がわかっ

Vol.2011-SE-173 No.9
2011/7/21

た．一方で，より単純なモデルを採用した場合は予測性能が低下するというトレードオフの関係が見られた．この事より，組織が有用なメトリクスを収集する一定の期間内に比較的多くのプロジェクトを行う場合において，windowing approach が有効であると考えられる．一定の期間内に行われるプロジェクトが少ない場合ではその効果は小さなものとなる．

この結果より，実際的な状況下で windowing approach は一定の効果をもたらす場合があると言える．今後は，類推法などプロジェクトデータの数がモデルの精度の決定的な要因ではない工数見積もりモデルによる同様の分析が必要であると言える．

参 考 文 献

1) Lokan, C. and Mendes, E.: Applying moving windows to software effort estimation, *Proc.of 3rd International Symposium on Empirical Software Engineering and Measurement (ESEM'09)*, pp.111–122 (2009).
2) Boetticher, G., Menzies, T. and Ostrand, T.: PROMISE Repository of empirical software engineering data, `http://promisedata.org/repository`, West Virginia University, Department of Computer Science (2007).
3) Mair, C., Shepperd, M. and Jørgensen, M.: An Analysis of Data Sets Used to Train and Validate Cost, *Proc.of 1st International Workshop on Predictor Models in Software Engineering(PROMISE'05)* (2005).
4) Kitchenham, B., Pfleeger, S., McColl, B. and Eagan, S.: An empirical study of maintenance and development estimation accuracy, *Journal of Systems and Software*, Vol.64, pp.57–77 (2002).
5) Maxwell, K.D.: *Applied Statistics for Software Managers*, Prentice Hall, Inc. (2002).
6) Harrel, F.E.: *Regression Modeling Strategies*, Springer (2001).