

# 修士学位論文

## 題目

類似性に基づく工数予測における予測回避プロジェクト判別手法の  
提案とその評価

## 指導教員

楠本 真二 教授

## 報告者

中村 哲彬

平成 22 年 2 月 8 日

大阪大学 大学院情報科学研究科  
コンピュータサイエンス専攻

類似性に基づく工数予測における予測回避プロジェクト判別手法の  
提案とその評価

中村 哲彬

内容梗概

ソフトウェア開発管理において、開発プロジェクトの計画の立案や管理を円滑に進行、完遂するために、開発工数の予測が行われる。開発工数の予測を行うことにより、開発に必要な要員数や期間が決定され、人的資源の確保や要員の割り当てなどの決定が可能である。工数予測手法の一つに、類似性に基づく工数予測 (Estimation by Analogy, 以下 EbA) 手法がある。EbA 手法では、プロジェクト特性を表すメトリクスが類似しているプロジェクトは、工数も互いに似た値をとるであろうという仮定に基づいて予測する手法である。従って、EbA 手法では、過去プロジェクト中に予測するプロジェクト (予測プロジェクト) と類似したプロジェクトが、数多く存在する程高い精度での予測が期待できる。一方、過去プロジェクト中に類似したプロジェクトが存在しない場合、EbA 手法では高い精度での予測は期待できないため、他の予測手法を用いる方が良い可能性がある。従来、工数予測手法の選択についての明確な基準はなく、EbA 手法を用いるべきかの判断が工数予測における課題の一つである。

本論文では、EbA 手法において高い予測精度が期待できないプロジェクト (予測回避プロジェクト) を判別する手法を提案する。提案手法では、判別の基準として予測プロジェクトが各過去プロジェクトから見て、類似プロジェクトとして選択されるかを採用する。予測プロジェクトが、全ての過去プロジェクトから見て類似プロジェクトとして選択されなかった場合、予測回避プロジェクトと判別する。提案手法による判別効果を評価するために、実際の開発プロジェクトで構成される三種類の公開データセットを用いて実証実験を行った。提案手法を適用した実証実験の結果、提案手法で予測回避プロジェクトと判別されたプロジェクトのみを EbA 手法で予測することで、予測回避しない場合と比較して、最大で相対誤差平均値で 0.19、MER 平均値で 0.26 改善された。

主な用語

予測回避, EbA (Estimation by Analogy), 類似度, MRE (Magnitude of Relative Error), MER (Magnitude of Error Relative)

## 目次

1	まえがき	1
2	類似性に基づく工数予測手法	3
3	提案手法	5
3.1	提案手法の概要	5
3.2	提案手法の手順	5
4	実証実験	8
4.1	実験目的と概要	8
4.2	実験に用いたデータセット	8
4.3	評価指標	9
4.4	実験手順	10
5	判別効果の評価結果	13
6	考察	16
7	関連研究	21
8	あとがき	23
	謝辞	24
	参考文献	25
	付録	28

## 目次

1	P1 の類似プロジェクトの範囲 . . . . .	6
2	予測回避プロジェクトでないと判別する時 . . . . .	6
3	予測回避プロジェクトと判別する時 . . . . .	7
4	実験の流れ . . . . .	12
5	各データセットにおける MRE での判別精度 . . . . .	18
6	各データセットにおける MER での判別精度 . . . . .	19
7	提案手法と関連研究の比較 . . . . .	29

## 表目次

1	Desharnais データセットにおける実験結果 . . . . .	14
2	ISBSG データセットにおける実験結果 . . . . .	14
3	Kitchenham データセットにおける実験結果 . . . . .	15

## 1 まえがき

ソフトウェア開発において、開発プロジェクトの計画立案や開発管理を円滑に進行、完遂するために、開発工数の予測が行われる。開発工数の予測を行うことで、開発に必要な要員数や期間が決まり、人的資源の確保や要員の割り当てなどの決定が可能である。適切に人的資源や要員の割り当てを決定することで、コスト超過や納期遅れ等のプロジェクト失敗のリスクが抑えられる。また、工数を実際よりも小さく予測（過小予測）した場合、ベンダ側において必要なアウトプットの達成が困難になり、一方、工数を実際よりも大きく予測（過大予測）した場合、ユーザ側とベンダ側の信頼関係を損なう原因となる [1]。

これまでに多数の工数予測手法が提案されている [2][3][4][5] が、工数予測手法は大別すると、モデルベースの手法とメモリベースの手法にわけられる。

モデルベースの手法は、プロジェクトの特性を表す値（メトリクス）を説明変数とし、工数を目的変数として予測を行うモデルを構築する手法である。モデルベースの手法はさらに、COCOMO[2] などの定義済みモデルを用いる手法と、重回帰分析やニューラルネット [3] などの過去プロジェクトからモデル式を構築する手法にわけられる。モデルベースの手法では、関係式が明確であり、シミュレーションが可能である反面、特徴的なプロジェクトでは十分な予測精度を得ることが難しい [6]。

メモリベースの手法 [5][7][8] は、予測対象プロジェクト（予測プロジェクト）毎に、類似した過去プロジェクト（類似プロジェクト）を選択し、類似プロジェクトの工数を用いて予測値を得る手法である。メモリベースの手法は、類似性に基づく工数予測 (Estimation by Analogy, 以下 EbA) 手法とも呼ばれ、メトリクスが類似しているプロジェクト同士は工数も互いに似た値をとるであろうという前提に基づいている。本論文ではこの EbA 手法を対象とする。

EbA 手法は、過去にほぼ同様のプロジェクトを実施していれば、非常に高い精度で工数予測が可能であるが、一方で過去プロジェクトの中に類似したプロジェクトが存在しない場合には、高い予測精度は期待できない。従って、EbA 手法を用いて高い予測精度が望めないのであれば、EbA 手法で予測を行うべきではない。しかし、EbA 手法で予測回避すべきかを判断する方法は明らかではない。実際の開発現場では、EbA 手法で予測を行うかは熟練した開発者の経験則により判断されており、系統的な方法はない。

そこで、本論文では EbA 手法における予測回避プロジェクトを判別する手法を提案する。提案手法では、予測プロジェクトが過去プロジェクトの類似プロジェクトとして選ばれるかを予測回避プロジェクトの判別基準に採用する。そして、予測回避プロジェクトの判別による効果を評価するため、三種類のデータセットにおいて、提案手法で予測回避と判別したプロジェクトを除いて予測を行った場合と、予測回避判別を行わなかった場合について比較す

る評価実験を行った。

以降、2章ではEbA手法の説明を行う。続いて3章では提案する予測回避プロジェクト判別手法について説明を行い、4章では提案手法の判別の効果を評価するための実験について説明する。5章では実験結果をまとめ、6章で考察を行う。7章で関連研究とその問題点について述べる。8章ではまとめと今後の課題について記す。

## 2 類似性に基づく工数予測手法

類似性に基づく工数予測 (Estimation by analogy, 以下 EbA) [5][7] 手法は, 予測プロジェクトに類似したプロジェクトを過去プロジェクトから選択し, 類似プロジェクトの工数の実績値に基づいて工数を予測する手法である. EbA 手法ではプロジェクトの特性値 (メトリクス) が類似しているプロジェクト同士は, 工数も互いに似た値をとることを前提としている.

EbA 手法はダミー変数化, 正規化, 類似度計算, 予測値計算から構成される. 各手順の計算アルゴリズムとしては, いくつかの方法が提案されているが, ここでは本論文の実験で採用したアルゴリズムのみを示す. 以下では説明のため,  $P_i$  はプロジェクト  $i$  を,  $M_j$  はメトリクス  $j$  を,  $V_{i,j}$  はプロジェクト  $i$  のメトリクス  $j$  の値を表すとする.

### 手順 1 ダミー変数化

データセットにカテゴリ変数と呼ばれる数値自体に意味をもたない名義尺度のメトリクスが含まれる場合, 便宜上量的なデータとして扱うため, カテゴリごとにダミー変数に置き換える. プロジェクト  $P_i$  のメトリクス  $M_j$  のカテゴリ  $c$  のダミー変数  $d_{ij}(c)$  は式 (1) で定義される. カテゴリ  $d$  はカテゴリ変数のメトリクス名を表す.

$$d_{ij}(c) = \begin{cases} 1 \dots \text{カテゴリ } d \text{ に属する} \\ 0 \dots \text{カテゴリ } d \text{ に属さない} \end{cases} \quad (1)$$

### 手順 2 正規化

データセットに記録されている値は, メトリクス毎に値域が異なるため, 類似度に対するメトリクスごとの影響度に差が生じる. メトリクス毎の類似度への影響度を均等にするために, メトリクスの正規化を行う.

メトリクスの正規化のうち, 以下二つのアルゴリズムである Normalize[5] と Order[9] を用いる.

**normalize:** normalize はメトリクス毎に値域を  $[0,1]$  に揃える正規化方法である. メトリクス  $M_j$  の最大値を  $\max(V_j)$ , 最小値を  $\min(V_j)$  としたとき, プロジェクト  $P_i$  のメトリクス  $M_j$  の値  $V_{i,j}$  を正規化した値  $V'_{i,j}$  は式 (2) で定義される.

$$V'_{i,j} = \frac{V_{i,j} - \min(V_j)}{\max(V_j) - \min(V_j)} \quad (2)$$



**order:** order はメトリクス毎に値の順位付けしたうえで値域を [0,1] に揃える正規化方法である．順位付けで同じ値が複数存在する場合には，最高順位ではなく同順位を用いなかったときの和の平均順位を順位として用いる．例えば，あるメトリクスにおける値が {1, 4, 4, 8,10} のとき，順位付けを行うと {1, 2 ,2 ,4 ,5} であるが，平均順位を用いて {1, 2.5 ,2.5 ,4 ,5} とする．メトリクスの値  $V_{i,j}$  の順位を  $order(V_{i,j})$  とすると，正規化した値  $V'_{i,j}$  は式 (3) で定義される．

$$V'_{i,j} = \frac{order(V_{i,j}) - \min(order(V_j))}{\max(order(V_j)) - \min(order(V_j))} \quad (3)$$

手順 3 類似度計算正規化したメトリクスの値を用いて予測プロジェクトと過去プロジェクトの類似度を計算する．ユークリッド距離に基づく類似度計算方法 [10] は，プロジェクト  $P_a$  とプロジェクト  $P_i$  の類似度  $sim(P_a, P_i)$  は式 (4) で定義される．

$$sim(P_a, P_i) = \frac{1}{dist(P_a, P_i)} \quad (4)$$

ここで， $dist(P_a, P_i)$  はプロジェクト  $P_a$  とプロジェクト  $P_i$  のユークリッド距離であり，式 (5) によって定義される．

$$dist(P_a, P_i) = \sqrt{\sum_{j=1}^y (V'_{a,j} - V'_{i,j})^2} \quad (5)$$

#### 手順 4 予測値計算

類似度が高い過去プロジェクトの工数の実績値から，予測プロジェクトの工数の予測値を計算する．予測値計算には，プロジェクトの規模を補正する  $amp(P_a, P_i)$  を類似プロジェクトの工数に乗じた値の平均 [7] を採用する．過去プロジェクト  $P_i$  の工数の実績値を  $E_i$ ，類似プロジェクト数を  $k$  個としたとき，予測プロジェクト  $P_a$  の予測値  $E_a$  は式 (6) で定義される．

$$E_a = \frac{\sum_{i \in k\text{-nearestProjects}} (E_i \times amp(P_a, P_i))}{k} \quad (6)$$

ここで  $k\text{-nearestProjects}$  は，予測プロジェクト  $P_a$  と類似度の高い上位  $k$  個の過去プロジェクトの集合を表す．また，プロジェクトのメトリクス数を  $l$  個とすると， $amp(P_a, P_i)$  は式 (7) で定義される．

$$amp(P_a, P_i) = \frac{\sum_{m \in l} \left( \frac{V_{a,m}}{V_{i,m}} \right)}{l} \quad (7)$$

$k$  の値は実験的に別途求める必要がある．

### 3 提案手法

#### 3.1 提案手法の概要

提案手法では、予測プロジェクトが過去プロジェクトの類似プロジェクトに選択されるかを判別の基準として予測回避プロジェクトを判別する。

#### 3.2 提案手法の手順

図 1, 図 2, 図 3 を用いて提案手法の手順を説明する。図 1, 図 2, 図 3 において, P1 ~ P4 は過去プロジェクトを表し, Pa, Pb は予測プロジェクトを表す。工数以外のメトリクスは要員数, 開発期間の二種類として, 縦軸, 横軸はそれぞれメトリクスを表す。プロジェクト間の距離が小さい程, 類似度が高いことを表し, 破線の円は, 中心にあるプロジェクトの類似プロジェクト  $t$  個の範囲を表す。 $t$  は予測回避プロジェクト判別の閾値となる。例では,  $t=2$  としている。

手順 1 図 1 において, ある過去プロジェクト(ここでは P1) とその他のプロジェクトの類似度の値を, EbA 手法で用いる類似度計算と同様の方法で計算する。

手順 2 手順 1 で求めた類似度の値を降順にソートし, 過去プロジェクトにおける類似プロジェクトの閾値  $t$  番目以内のプロジェクトを探し,  $t$  番目のプロジェクトが含まれるように破線の円で囲む。

手順 3 手順 1 ~ 手順 2 をすべての過去プロジェクトで行うと図 3, 又は図 2 のようになる。

手順 4 ここで予測回避プロジェクトの判別を行う。

図 2 では, 予測プロジェクト (Pa) が, ある過去プロジェクト (P4) において類似プロジェクトに含まれている。よって Pb は予測回避プロジェクトでないと判別する。

図 3 では, 予測プロジェクト (Pb) が, すべての過去プロジェクトの類似プロジェクトに含まれていない。よって Pb は予測回避プロジェクトと判別する。

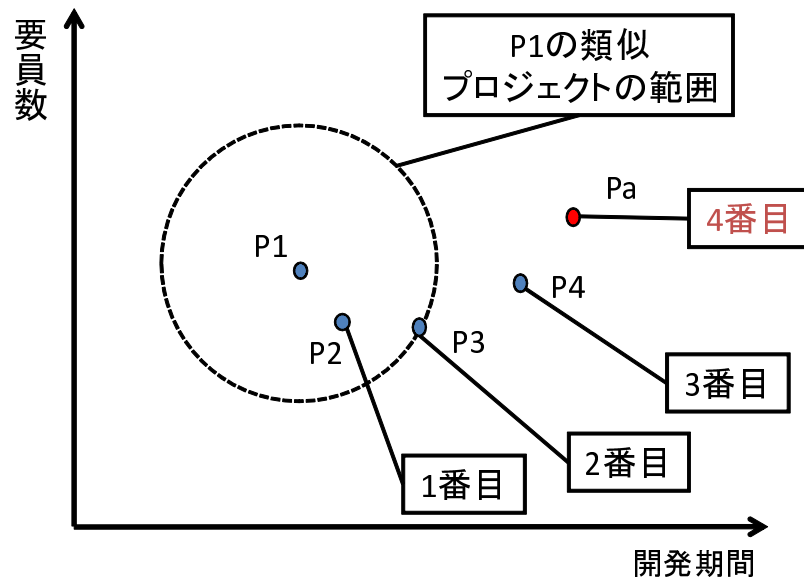


図 1: P1 の類似プロジェクトの範囲

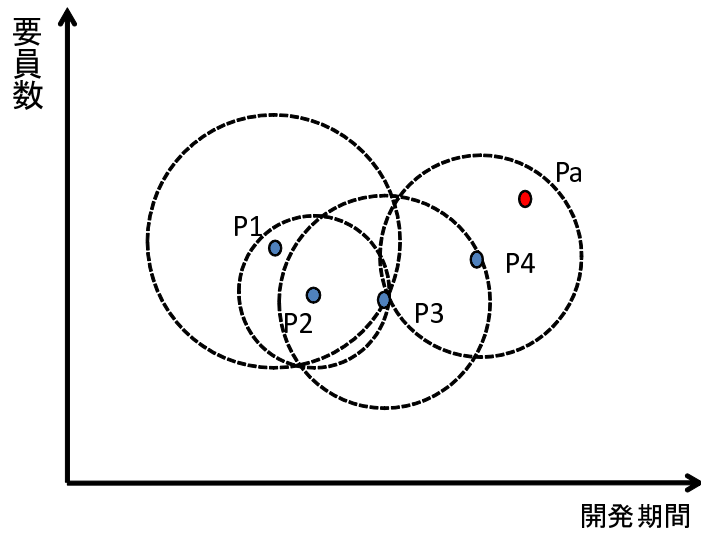


図 2: 予測回避プロジェクトでないと判別する時

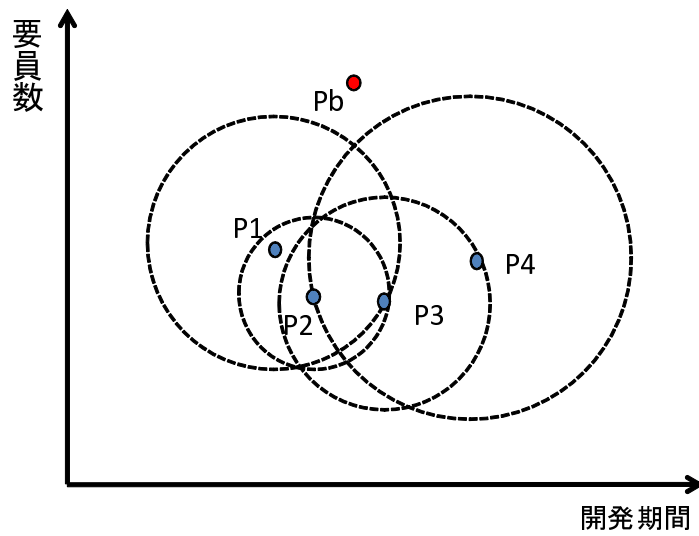


図 3: 予測回避プロジェクトと判別する時

## 4 実証実験

### 4.1 実験目的と概要

提案手法の予測回避プロジェクトの判別効果を評価するために、実際に行われたプロジェクトのデータセットを用いて実験を行った。実験では、Leave-One-Out (以下、LOO) 法 [11] を用いて

- 提案手法で予測回避プロジェクトを除去後に EbA 手法を用いて予測を行う
- 提案手法で予測回避プロジェクトを除去せずに EbA 手法を用いて予測を行う

の二組の比較を行う。

### 4.2 実験に用いたデータセット

提案手法の判別効果を評価するための実験において、Desharnais データセット [12][13]、ISBSG データセット [14]、Kitchenham データセット [15] の 3 種類を用いた。それぞれのデータセットにおいて、以下の理由からサブデータセットの抽出を行った。

- 想定している予測時に収集されていないメトリクス
- 欠損の多く含まれているメトリクス
- 欠損の含まれているプロジェクト

以下では各データセットの詳細について述べる。

Desharnais データセットは、Desharnais の修士学位論文 [12] で発表されたものであり、1980 年代のカナダのソフトウェア開発企業のプロジェクトから構成されるデータセットである。元のデータセットには 81 件のプロジェクト、12 種類のメトリクスが含まれており、抽出の結果、プロジェクト数 77 件、メトリクス数 6 種類のサブデータセットを作成した。

ISBSG データセットは、International Software Benchmark Standard Group (ISBSG) が収集した 20 カ国のソフトウェア開発企業のプロジェクトから構成されるデータセットである。データセットには 1989 年から 2008 年までに行われた 5052 件のプロジェクトがあり、118 種類のメトリクスが含まれており、抽出の結果、プロジェクト数 135 件、メトリクス数 5 種類のサブデータセットを作成した。

Kitchenham データセットは、Kitchenham が 2002 年に文献 [15] の中で公開したソフトウェア開発企業のプロジェクトから構成されるデータセットである。データセットには 1995 年から 1999 年までに行われた 145 件のプロジェクトがあり、9 種類のメトリクスが記録されており、抽出の結果、プロジェクト数 131 件、メトリクス数 5 種類のサブデータセットを

作成した．また Kitchenham データにはカテゴリ変数が含まれており，顧客コード（1～6），プロジェクトタイプ（新規か，開発か etc.）が含まれており，顧客コードは6種類，プロジェクトタイプは7種類の要素を含む．

以下では，それぞれのサブデータセットの各メトリクスの種類を示す．

- Desharnais データセット  
開発総工数（人時），調整済みファンクションポイント，開発期間（月），トランザクション数，エンティティ数
- ISBSG データセット  
開発総工数（人時），調整済みファンクションポイント，開発期間（月），計画工数（人時），要件定義工数（人時）
- Kitchenham データセット  
開発総工数（人時），調整済みファンクションポイント，開発期間（月），顧客コード，プロジェクトタイプ

### 4.3 評価指標

本論文では，提案手法の判別効果を評価するために，予測精度指標として MMAE，MMRE，MMER[16]，PRED25 の4種類を用いた．以下では予測プロジェクト数を  $N$  件，工数の実測値と予測値をそれぞれ  $X_i$ ， $\hat{X}_i$  ( $i = 1 \sim N$ ) とする．

- 絶対誤差平均値（MMAE:Mean Magnitude of Absolute Error）  
実測値と予測値の絶対誤差の平均値を表す．数値が小さいほど予測精度が高いことを示す．MMAE は式 (8) で定義される．

$$\text{MMAE} = \frac{\sum_{i=1}^N (|X_i - \hat{X}_i|)}{N} \quad (8)$$

- 相対誤差平均値（MMRE:Mean Magnitude of Relative Error）  
実測値と絶対誤差の比である MRE の平均値を表す．MMRE は実測値と比べて絶対誤差がどの程度の規模かを評価する指標であり，実測値よりも大きく予測を行う過大予測時に数値が特に大きくなり，数値が小さいほど予測精度が高いことを示す．MMRE は式 (9) で定義される．

$$\text{MMRE} = \frac{\sum_{i=1}^N \left( \frac{|X_i - \hat{X}_i|}{X_i} \right)}{N} \quad (9)$$

- MER 平均値 (MMER:Mean Magnitude Error of Relative)

予測値と絶対誤差の比である MER[16] の平均値を表す。MER は予測値と比べて誤差がどの程度の規模かを評価する指標であり、実測値よりも小さく予測を行う過小予測時に数値が特に大きくなり、数値が小さいほど予測精度が高いことを示す。MMER は式 (10) で定義される。

$$\text{MMER} = \frac{\sum_{i=1}^N \left( \frac{|X_i - \hat{X}_i|}{\hat{X}_i} \right)}{N} \quad (10)$$

- PRED25

全プロジェクトの中で、MRE の値が 0.25 以下で予測することができたプロジェクトの占める割合を表しており、値が大きい程高い精度で予測することができたことを表す。MRE が 0.25 以下であるプロジェクト件数を  $m$  件とすると、PRED25 は式 (11) で定義される。

$$\text{PRED25} = \frac{m}{N} (\%) \quad (11)$$

#### 4.4 実験手順

実験には、Leave-One-Out (以下、LOO) 法 [11] を用いた。説明のため、以下においては実験に用いるデータセットには  $n$  件のプロジェクトが含まれているとする。LOO 法は、予測対象とするプロジェクトをデータセットから 1 件取り出し、取り出したプロジェクトの工数を未知の値として、残りの  $n-1$  個のプロジェクト用いて予測を行うことを、予測プロジェクトを代えて個全てのプロジェクトに対して行う方法である。

以下では、実験手順の流れを図 4 のフローチャートを用いて説明する。

1. データセットから 1 件のプロジェクトを取り出し、工数の値を未知とすることで予測プロジェクトとする。残りの  $(n-1)$  件のプロジェクトを過去プロジェクトとして扱う。
2. 提案手法を用いて予測回避プロジェクトかを判別する。
3. 予測回避プロジェクトの場合、予測を回避する。予測を回避しなかった場合、EbA 手法を用いて予測を行い、MAE, MRE, MER を計算する。
4. 1. における予測プロジェクトの取り出しが  $n$  回目ならば MMAE, MMRE, MMER, PRED25 を計算して終了する。予測プロジェクトの取り出しが  $n$  回目でなければ 1. に戻る。
5. 比較対象の実験を行う。手順 1. 3. 4. の順に予測回避プロジェクトの判別をせずに、 $n$  回繰り返し、MMAE, MMRE, MMER, PRED25 を計算して終了する。

提案手法における予測回避プロジェクト判別の閾値  $t$  は予備実験 [17] の結果,  $t=1$  を用いた. 提案手法において, 類似度計算前の処理であるメトリクスの正規化には `normalize` と `order` のそれぞれの場合を用いた. また本論文では, EbA 手法を用いる際に, Naist Collaborative Filtering Engines[18] と呼ばれる奈良先端科学技術大学院大学のソフトウェア工学講座 定量的データ分析研究グループが作った EbA 手法を計算するシステムを用いた. EbA 手法に用いた正規化および類似度計算のアルゴリズムは, 提案手法の予測回避時に用いたアルゴリズムと同じものを用いた.

予測値計算時に用いる類似プロジェクト数  $k$  の値は予備実験の結果, 提案手法を適用時と適用しなかった時の共に Desharnais データでは  $k=2$ , ISBSG データセットでは  $k=5$ , Kitchenham データセットでは  $k=2$  を用いた.



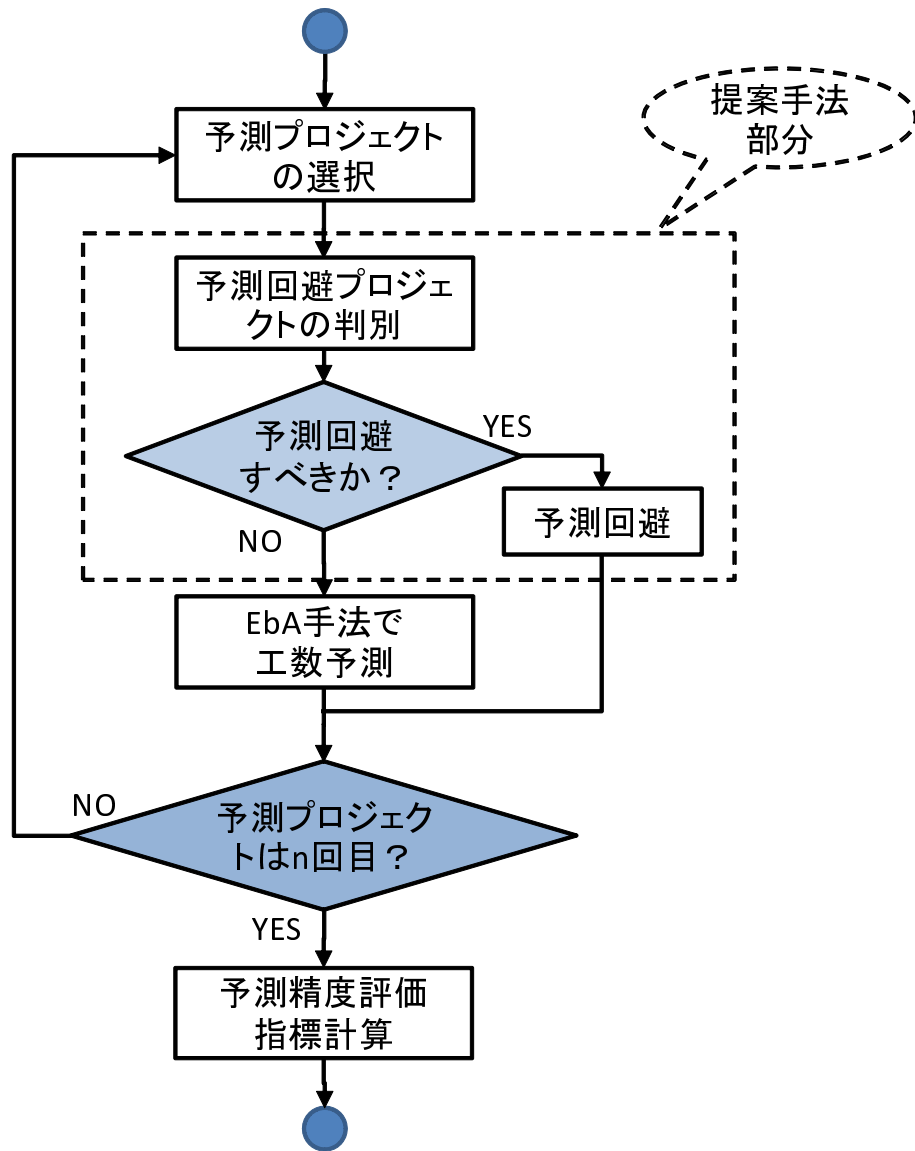


図 4: 実験の流れ

## 5 判別効果の評価結果

提案手法における、予測回避プロジェクト判別効果の評価のために行った実験結果を示す。実験から得られた提案手法適用時と適用しなかった時の各精度評価指標の値を表1、表2、表3に示す。表1、表2、表3はそれぞれ、Desharnais データセット、ISBSG データセット、Kitchenham データセットにおける実験結果を表している。各列は左から順に、提案手法で用いたメトリクス正規化の方法、対象または改善率、MMAE、MMRE、MMER、PRED25の値を表す。改善率とは、提案手法適用時の値を提案手法を適用しなかった時の値で割ったときの予測回避の結果、改善した割合を示す。

表1から、Desharnais データセットにおいて正規化に normalize を用いたとき、MMAE と MMER では改善され、MMRE と PRED25 では予測精度が下がった。また order を用いたとき MMAE、MMRE、MMER、PRED25 のすべてにおいて改善が見られ、改善率でそれぞれにおいて 22.4%、27.6%、42.7%、55.6% と全体的に大きく改善が見られた。また予測回避したプロジェクトのみの予測では、全体的に予測精度が悪くなっていたことから、Desharnais データセットにおいては、normalize では判別の効果は見られなかったが、order では判別の効果があったことがわかる。

表2から、ISBSG データセットにおいて正規化に normalize を用いたとき、MMER では若干の改善されたが、MMAE、MMRE、PRED25 ではそれぞれで若干予測精度が下がった。また order を用いたとき、MMAE、MMRE、PRED25 では若干の改善が見られ、それぞれ 9.4%、3.3%、2.0% の改善が見られたが、MMER では若干予測精度が悪くなり、改善率で-1.1% のだった。また、予測回避プロジェクトのみの予測を行い MMAE、MMRE、PRED25 では予測精度が下がっていたことから、ISBSG データセットにおいては、normalize では判別の効果見られなかったが、order では MMER では若干の改悪が見られたものの、Desharnais データセット程ではないものの判別の効果があったことがわかる。

表3から、Kitchenham データにおいて normalize を用いたとき、MMAE、MMRE、MMER、PRED25 のすべてにおいて改善が見られ、改善率でそれぞれにおいて 41.6%、8.8%、1.5%、12.9% の改善が見られた。特に MMAE で大きく改善が見られたことがわかる。また order を用いたとき MMAE、MMRE、MMER、PRED25 のすべてにおいて精度が下がった。また、予測回避プロジェクトのみの予測において、normalize ですべての評価指標において精度が下がっていたことから、Kitchenham データセットにおいては、order では判別の効果により、予測精度が下がったが、normalize では Desharnais データセット程ではないが、判別の効果があったことがわかる。

表 1: Desharnais データセットにおける実験結果

		MMAE	MMRE	MMER	PRED25
normalize	提案手法適用せず	2834.44	0.73	0.56	40.6
	提案手法適用時	2199.30	0.79	0.49	35.3
	予測回避プロジェクトのみ	3245.76	0.69	0.75	40.6
	改善率	30.9%	-8.9%	12.6%	-13.1%
order	提案手法適用せず	2900.02	0.68	0.62	28.1
	提案手法適用時	2004.81	0.49	0.36	43.7
	予測回避プロジェクトのみ	3435.36	0.84	0.87	32.5
	改善率	22.4%	27.6%	42.7%	55.6%

表 2: ISBSG データセットにおける実験結果

		MMAE	MMRE	MMER	PRED25
normalize	提案手法適用せず	1797.34	0.63	0.52	33.3
	提案手法適用時	2058.83	0.66	0.49	32.6
	予測回避プロジェクトのみ	1603.11	0.56	0.58	34.9
	改善率	-6.6%	-4.8%	6.0%	-1.1%
order	提案手法適用せず	2823.96	0.53	0.61	37.8
	提案手法適用時	2568.97	0.51	0.62	38.5
	予測回避プロジェクトのみ	5420.66	0.54	0.58	32.5
	改善率	9.4%	3.3%	-1.1%	2.0%

表 3: Kitchenham データセットにおける実験結果

		MMAE	MMRE	MMER	PRED25
normalize	提案手法適用せず	1870.88	0.77	0.77	28.0
	提案手法適用時	1093.49	0.70	0.76	31.6
	予測回避プロジェクトのみ	4111.58	0.96	0.80	17.6
	改善率	41.6%	8.8%	1.5%	12.9%
order	提案手法適用せず	2271.40	0.70	0.60	36.3
	提案手法適用時	2761.78	0.72	0.67	34.7
	予測回避プロジェクトのみ	673.73	0.65	0.56	41.9
	改善率	-21.6%	-2.0%	-11.7%	-4.7%

## 6 考察

実証実験から得られた結果を各データセットの各アルゴリズム毎に結果を見ていく。

Desharnais データセットでは、order を用いたとき MMAE, MMRE, MMER, PRED25 すべてにおいて改善が見られ、改善率でそれぞれ 22.4% , 27.6% , 42.7% , 55.6% 改善された。MMRE, MMER と PRED25 のそれぞれにおいて大きく改善されたことから、過大予測時、過小予測時の両方に判別の効果があったことがわかる。また MMAE が改善されたことから、規模の大きいプロジェクトに対しても予測回避の判別の効果があることがわかる。normalize では MMRE と PRED25 においては改善されず判別の効果があったとは言えないが、MMAE, MMER では改善率でそれぞれ 30.9% , 12.6% 改善されたことから、規模の大きいプロジェクトと、過小予測時の提案手法の判別の効果があったことがわかる。

ISBSG データセットでは order を用いたとき、MMER では判別の効果があまり見られず、改善率で 1.1% 悪くなったものの、MMAE, MMRE, PRED25 では改善が見られ、改善率でそれぞれ 9.4% , 3.3% , 2.0% と改善された。このことから、過小予測時では、提案手法の判別の効果が見られなかったが、過大予測時と規模の大きいプロジェクトに対しては提案手法の判別の効果があったことがわかる。改善率が小さい理由の一つとして、ISBSG データセットは、International Software Benchmarking Standards Group が予測に用いるためのプロジェクトを選び出しているため、ISBSG データセットを用いた EbA 手法の予測の精度が高い点が挙げられる。normalize では、MMAE, MMRE と PRED25 においては改善されず判別の効果があったとは言えないが、MMER では改善率で 6.0% 改善された。過小予測時では提案手法の判別の効果が若干得られたが、過大予測時及び規模の大きいプロジェクトに対しては効果が得られなかった。

Kitchenham データでは、order を用いたとき、MMAE, MMRE, MMER, PRED25 すべてにおいて改善されず、提案手法の判別の効果がなかったことがわかる。normalize では MMAE, MMRE, MMER, PRED25 すべてにおいて改善され、改善率で 41.6% , 8.8% , 1.5% , 12.9% 改善された。このことから規模の大きいプロジェクトに対して、特に判別の効果があったことがわかる。過大予測時と過小予測時に対しても、規模の大きいプロジェクトに対して程ではないにせよ判別の効果があったことがわかる。

予測回避プロジェクトの判別の効果に対する評価として、予測回避プロジェクト 1 件ずつに着目するため、Alberg Diagram[19] の AUC (Area Under the Curve) を用いる。AUC とは曲線下面積のことを表し、精度が高いほど左上に凸形状となり、グラフの曲線下面積が大きくなる特徴がある。この Alberg Diagram は、提案手法で予測回避プロジェクトと判別されたプロジェクトの件数のうち、予測結果である MRE の値を降順にソートしたときに予測回避プロジェクトが該当している割合を表す。

各データセットにおける提案手法適用時の判別結果の MRE と MER を表した Alber Diagram を図 5(a) ~ 図 6(c) に表す。図 5(a), 図 5(b), 図 5(c) はそれぞれ Desharnais データセット, ISBSG データセット, Kitchenham データセットでの MRE の実験結果を表し, 図 6(a), 図 6(b), 図 6(c) はそれぞれ Desharnais データセット, ISBSG データセット, Kitchenham データセットでの MER の実験結果を表す。

また, 提案手法の判別の閾値はすべてにおいて  $t=1$  とした。予測の際に用いる MRE, MER の値は, 各データセットにおいて実験で最も提案手法の判別の効果が良かったときのものを採用した。Desharnais データセットは order で予測時に用いる類似プロジェクト数  $k=2$  を, ISBSG データセットは order で予測時に用いる類似プロジェクト数  $k=6$  を, Kitchenham データセットでは normalize で予測時に用いる類似プロジェクト数は  $k=2$  の結果を採用した。

グラフの横軸は予測結果である MRE または MER の値を降順に並べたときに含まれるプロジェクト数の割合を表す。縦軸については, 左軸は提案手法で判別された予測回避プロジェクトのうち, 実際に含まれるプロジェクトの割合を表し, 右軸は MRE または MER の値を表す。凡例“理想判別モデル”は MRE または MER の値が大きな順に予測回避プロジェクトが選ばれた場合のモデルを表す。

図 5(a) では Desharnais データセットにおける, AUC は 0.59 である。提案手法において, MRE の値が大きいプロジェクトについては理想判別モデルに近いことがわかる。判別結果がランダムの場合における原点を通る正比例のモデルと比較すると, 提案手法は, AUC が 0.5 を超えていることから, MRE の大きいプロジェクトを判別している傾向があることがわかり, 過大予測時に判別の効果があることがわかる。

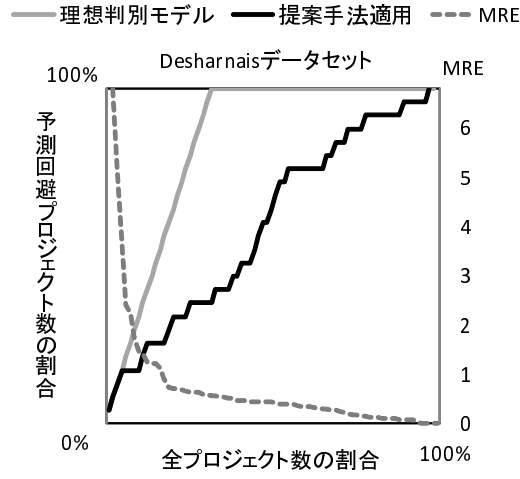
図 5(b) では, ISBSG データセットにおける AUC は 0.53 であり, 理想判別モデルに近いとは言えず, 判別の効果はあまりみられない。

図 5(c) では, Kitchenham データセットにおける AUC は 0.57 であり, 理想判別モデルに近いとは言えないが, 判別の精度があまり良くない。

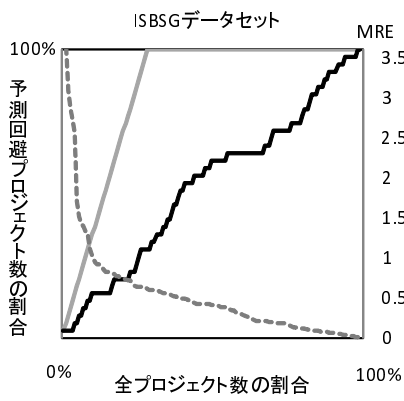
図 6(a) では, Desharnais データセットにおいて, AUC は 0.62 である。提案手法において MER の値が大きいプロジェクトについては理想判別モデルに近いことがわかる。提案手法は AUC が 0.5 を超えていることから, MER の大きいプロジェクトを判別しており, 過小予測時に判別の効果があることわかる。

図 6(b) では, ISBSG データセットにおける AUC は 0.59 であり, MER が大きいプロジェクトにおいて判別できていないことがあるが, 全体的に判別できていないわけではない。

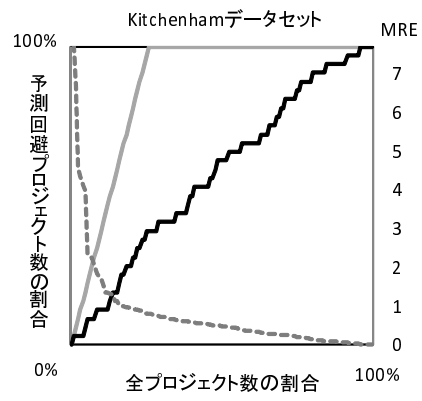
図 6(c) では, Kitchenham データセットにおける AUC は 0.66 であり, AUC の値は比較的大きいが, MER が大きいプロジェクトにおいて判別できていないことがある。MER では三種類のデータセットにおいて AUC の値は MRE のときと比較して大きかった。提案手法では過大予測よりも過小予測を判別できている傾向がある。以上より, 提案手法の正規化



(a) Desharnais データセット

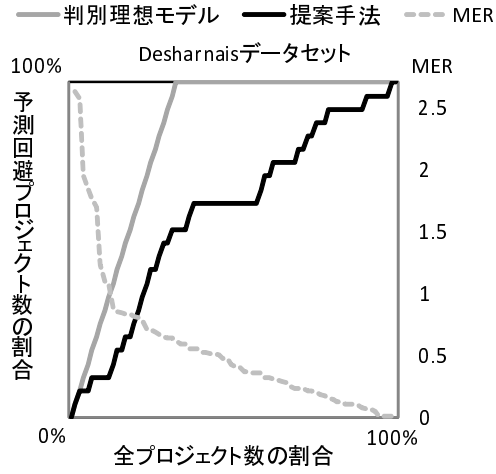


(b) ISBSG データセット

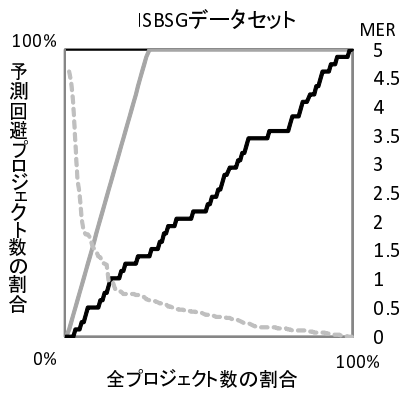


(c) Kitchenham データセット

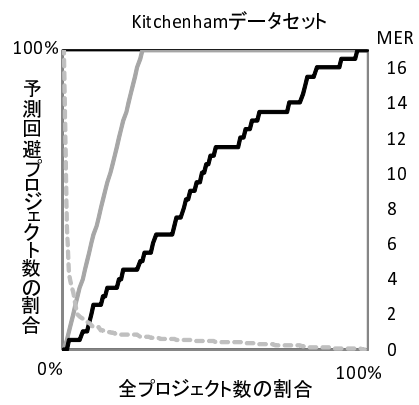
図 5: 各データセットにおける MRE での判別精度



(a) Desharnais データセット



(b) ISBSG データセット



(c) Kitchenham データセット

図 6: 各データセットにおける MER での判別精度



による判別の効果はデータセットの特徴により異なることがわかる。

また、実際の開発現場では、EbA手法の予測回避すべきかの判断は熟練した経験者の判断で行われているため、本研究の結果と経験者の判断を評価することが判別の評価において望ましい。本研究においては、実際の開発現場における情報を得ることができなかつたため、本論文で紹介したような方法をとった。

## 7 関連研究

関連研究として、工数予測モデル構築時に予測失敗につながるプロジェクトを除去する研究として、過去プロジェクトの一部を除去する研究と、予測プロジェクトを除去（予測回避）する研究がある。

予測モデル構築時の過去プロジェクト除去に焦点を当てた研究として Seo ら [20]、渡邊ら [21]、戸田ら [22] の研究がある。

Seo ら [20] は既存の外れ値除去アルゴリズムを工数予測に適用することを提案している。具体的には、複数の予測モデル（最小2乗法、ニューラルネットワーク、ベイジアンネットワーク）構築時に、他と特に異なるプロジェクト（外れ値）を探索する複数のアルゴリズム（Least Trimmed Square[23]、k-means clustering[24]）を、それぞれに用いて実証実験を行っている。複数のデータセットで行い、予測モデル、外れ値除去アルゴリズムとデータセットの特徴の組み合わせによる予測精度への影響を調べる実証実験を行っている。実証実験において外れ値除去アルゴリズムの効果はデータセットの特徴により異なることを示している。

渡邊ら [21] は、開発工数と相関関係が高いメトリクスが、特異な過去プロジェクト（特異プロジェクト）を過去プロジェクトから除去する方法を提案している。具体的には、開発規模が開発工数と一次の、開発期間が開発工数と三乗根の相関関係にある [25] ことから、この関係から外れたプロジェクトを除去し、除去後に予測モデルを構築し、予測を行う手法である。予測モデルにステップワイズ対数重回帰分析を用いた実証実験において、特異プロジェクト除去前と比較して、除去後の方が予測精度で改善したことを示している。

戸田ら [22] は、予測モデルに重回帰モデルを用いて、予測プロジェクトとメトリクスが類似した過去プロジェクトのみを抽出し、予測モデル構築を行う過去プロジェクト選定方法を提案している。例えば、“業種”を表すメトリクスにおいて、予測プロジェクトが“銀行”である場合、過去プロジェクト中に“銀行”以外のものを除去し、予測モデル構築のための候補とする。他のメトリクスにおける候補を選出し、候補の中から最も適した予測モデルを選び工数予測を行う。実証実験において、予測精度で過去プロジェクト選定前と比較して、選定後の方が改善したことを示している。

本論文では、予測プロジェクトを除去の対象としているのに対して、これらの研究では、予測モデル構築における際の過去プロジェクトを除去の対象を対象としている。基本的な考え方は同じであるため、過去プロジェクト除去のアルゴリズムを予測回避に用いることもとも可能である。また両者を併用することで更なる予測精度の改善が見込める。

予測プロジェクトの除去を対象とした研究としては、本論文以外には、戸田ら [26] の研究がある。戸田らは、EbA 手法において予測回避プロジェクトを選定するために、“類似性に

基づく工数予測手法において、予測対象のプロジェクトに似ているプロジェクト群のばらつきが大きいならば、その予測は失敗しやすい。”という仮説を立て、検証実験を行っている。ばらつきを表す指標としては、複数の類似プロジェクトの工数の平均もしくは工数の分散を用いている。検証実験の結果、類似プロジェクトの工数の平均もしくは分散が大きなプロジェクトには、予測精度が低いプロジェクトが多く含まれていることを示している。

戸田らは予測回避すべきプロジェクトの特性は示しているが、予測回避プロジェクトの具体的な判別基準や、判別方法は示していない。戸田らの判別基準が明らかになれば、本論文の提案手法と併用することでさらなる精度改善が期待できる。

## 8 あとがき

本論文では EbA 手法を用いて高い予測精度を期待できないプロジェクト（予測回避プロジェクト）を判別する予測回避プロジェクト判別手法を提案し，公開されているデータセットを用いて，判別の効果を評価する実証実験を行った．予測回避判別手法では，予測プロジェクトが過去プロジェクトの類似プロジェクトに選ばれるかを予測回避の判別基準とした．提案手法を適用した評価実験の結果，提案手法で予測回避プロジェクトと判別されたプロジェクトのみを EbA 手法で予測することで，予測回避しない場合と比較して，最大で相対誤差平均値で 0.19，MER 平均値で 0.26 改善された．

今後の課題として，EbA 手法で，過去プロジェクトにおける特異プロジェクトを除去する方法と提案手法を併用することで，さらなる予測精度での改善が見込まれる．そして，予測回避プロジェクトと判別したプロジェクトを，異なる予測手法を用いて予測を行う必要がある．さらに実際の開発現場では，EbA 手法の予測回避すべきかの判断は熟練した経験者の判断で行われているため，本研究の結果と経験者の判断を比較することも必要である．

## 謝辞

本研究を進めるにあたり，研究方法などに関する多くのアドバイスと共に，研究に対する姿勢など数多くの御指導を頂きました大阪大学大学院情報科学研究科コンピュータサイエンス専攻 楠本 真二 教授に心から深く感謝致します．

本研究を進めるにあたり，本研究を確固たるものとする的確で有益な多くの御助言を頂きました 同 岡野 浩三 准教授に深く感謝致します．

本研究を進めるにあたり，研究方法などに関する多くのアドバイスに加え，的確で有益な多くの御助言を頂きました 同 肥後 芳樹 助教に深く感謝致します．

本研究を進めるにあたり，本研究を確固たるものとする的確で有益な多くの御助言に加え，細部にわたる熱心なご指導を頂きました 同 柿元 健 特任助教に深く感謝致します．

本研究で利用した予測エンジンの作者である大杉直樹氏，角田雅照氏に，深く感謝致します．

最後に，その他様々な御指導，御助言を頂いた大阪大学 大学院情報科学研究科コンピュータサイエンス専攻 楠本研究室の皆様に深く感謝致します．

## 参考文献

- [1] 独立行政法人 情報処理推進機構 ソフトウェア・エンジニアリング・センター編. ソフトウェア開発見積りガイドブック～IT ユーザとベンダにおける定量的見積りの実現. オーム社, 2006.
- [2] B. W. Boehm. *Software Engineering Economics*. Prentice Hall, New Jersey, 1981.
- [3] K. Srinivasan and D. Fisher. Machine learning approaches to estimation software development efforts. *IEEE Transaction on Software Engineering*, Vol. 21, No. 2, pp. 126–137, 1995.
- [4] S. D. Conte, H. E. Dunsmore, and V.Y. Shen. *Software engineering metrics and models*. The Benjamin/Cummings Publishing Company, Inc, 1986.
- [5] M. Shepperd and C. Schofield. Estimating software project effort using analogies. *IEEE Transactions on Software Engineering*, Vol. 23, No. 12, pp. 736–743, 1997.
- [6] B. Steece S. Chulani, B. Boehm. Bayesian analysis of empirical software engineering cost models. *IEEE Transactions on Software Engineering*, Vol. 25, No. 4, pp. 573–583, 1999.
- [7] 角田雅照, 大杉直樹, 門田暁人, 松本健一, 佐藤慎一. 協調フィルタリングを用いたソフトウェア開発工数予測方法. 情報処理学会論文誌, Vol. 46, No. 5, pp. 1156–1164, 1999.
- [8] N. Ohsugi, M. Tsunoda, A. Monden, and K. Matsumoto. Effort estimation based on collaborative filtering. In *Proceeding of 5th International Conference on Product Focused Software Process Improvement (Profes2004)*, pp. 274–286, 2004.
- [9] 角田雅照, 柿元健, 大杉直樹, 門田暁人, 松本健一. 類似プロジェクトを用いた工数予測手法の性能比較. 電子情報通信学会技術報告, Vol. 105, No. 331, pp. 45–50, 2005.
- [10] C. Kirsopp, E. Mendes, R. Premraj, and M. J. Shepperd. An empirical analysis of linear adaptation techniques for case-based prediction. In *Proceedings of 5th International Conference on Case-Based Reasoning (ICCBR2003)*, pp. 231–245, 2003.
- [11] C. Kirsopp, M. Shepperd, and J. Hart. Search heuristics, case-based reasoning and software project effort prediction. In Morgan Kaufmann Publishers Inc., editor, *Genetic and Evolutionary Computation Conference*, pp. 1367–1374, 2002.

- [12] J. M. Desharnais. Analyse statistique de la productivité des projets informatiques à partir de la technique des points de fonction. In *Master Thesis*, University of Montreal, 1989.
- [13] C. Mair, G. Kadoda, M. Lefley, K. Phalp, C. Schofield, M. Sheppard, and S. Webster. An investigation of machine learning based prediction systems. *Journal of Systems and Software*, Vol. 53, No. Issue1, pp. 23–29, 2000.
- [14] ISBSG Estimating. Benchmarking and research suite release 11. International Software Benchmarking Standards Group (ISBSG), 2009.
- [15] B. Kitchenham, S. L. Pfleeger, B. McColl, and S. Eagan. An empirical study of maintenance and development estimation accuracy. *Journal of Systems and Software*, Vol. 64, No. Issue1, pp. 57–77, 2002.
- [16] B. Kitchenham, G. MacDonell, M. Pickard, and M. Sheppard. What accuracy statistics really measure. *IEEE Transactions on Software Engineering*, Vol. 148, No. 3, pp. 85–99, 2003.
- [17] 中村哲彬, 柿元健, 楠本真二. 類似性に基づく工数予測における適切でない予測プロジェクトの判別の試み. 電子情報通信学会技術報告, Vol. 109, No. 170, pp. 83–87, 2009.
- [18] Naist collaborative filtering engines. <http://sourceforge.jp/projects/ncfe/>.
- [19] N. Ohlsson and H. Alberg. Predicting fault-prone software modules in telephone switches. *IEEE Transactions on Software Engineering*, Vol. 22, No. 12, pp. 886–894, 1996.
- [20] Y. S. Seo, K. A. Yoon, and D. H. Bae. An empirical analysis of software effort estimation with outlier elimination. In *1st International Conference on Predictive Models in Software Engineering '08*, pp. 25–32, 2008.
- [21] 渡邊瑞穂, 柿元健, 戸田航史, 門田暁人, 松本健一. ソフトウェア開発工数予測における特異プロジェクト除去の効果. 日本ソフトウェア科学会第24回大会論文集 CD-ROM(講演番号 7B-2), 2007.
- [22] 戸田航史, 門田暁人, 松本健一. ソフトウェア開発工数予測のためのフィットデータ選定方法. 情報処理学会論文誌, Vol. 50, No. 11, pp. 2699–2709, 2009.

- [23] P. Rousseeuw and A. Leroy. *Robust Regression and Outlier Detection*. John Wiley and Sons, Inc, 1987.
- [24] J. Han and M. Kamber. *Data mining: Concepts and Techniques*. Morgan Kaufmann, 2006.
- [25] L. H. Putnam. A general empirical solution to the macro sizing and estimation problem. *IEEE Transactions on Software Engineering*, Vol. 4, No. 4, pp. 345–361, 1978.
- [26] 戸田航史, 門田暁人, 松本健一. ソフトウェア開発工数予測における予測回避プロジェクトの選定とその効果. *ソフトウェア工学の基礎 XVI FOSE 2009*, pp. 229–236, 2009.
- [27] J. Kromrey and C. Hines. Nonrandomly missing data in multiple regression: An empirical comparison of common missing-data treatments. Vol. 54, No. 3, pp. 573–593, 1994.
- [28] R. J. A Little and D. B. Rubin. *Statistical Analysis with Missing Data 2nd edition*. John Willey and Sons, 2002.



## 付録

### 関連研究との比較

7章における戸田ら [26] が提案する予測回避プロジェクト選定における仮定に基づき、実験を行った結果を比較する。戸田らの研究では、予測回避プロジェクトの選定の前提に、“類似性に基づく工数予測手法において、予測対象のプロジェクトに似ている群のばらつきが大きいならば、その予測は失敗しやすい。”ことを挙げている。またばらつきの大きさを表す指標として、類似プロジェクト群の工数における分散を用いている。具体的には、 $n$ 回の予測を行う場合、予測プロジェクトの類似プロジェクト群の分散を  $n$  個準備し、分散の大きさの順に  $n$  個を昇順に並べて下位から全体の 10% 刻みで選んだプロジェクトの予測精度を MMRE が最も高かったときを調べて実験結果とした。

実験手順は 4.4 節で行った内容と同様である。実験に用いたデータセット、EbA 手法における正規化、類似度計算、予測値計算をそれぞれ提案手法と同じ条件で実験を行い、実験結果を MMRE で評価した。類似プロジェクト数の値は、提案手法で最も良かったものを採用し、Desharnais データセットは 2 個、ISBSG データセットは 6 個、Kitchenham データセットは 2 個用いた。予備実験を行った結果、分散の大きさの順に昇順に並べて下位からの全体に占める割合は Desharnais データセット、ISBSG データセット、Kitchenham データセットにおいて、それぞれ 40%、60%、40% であった。図 7 に提案手法と比較対象の実験結果をグラフで表す。左から順に、Desharnais データセットにおける提案手法と戸田ら [26] の手法、ISBSG データセットにおける提案手法と戸田らの手法、Kitchenham データセットにおける提案手法と戸田らの手法を表し、縦軸は MMRE の値を示している。

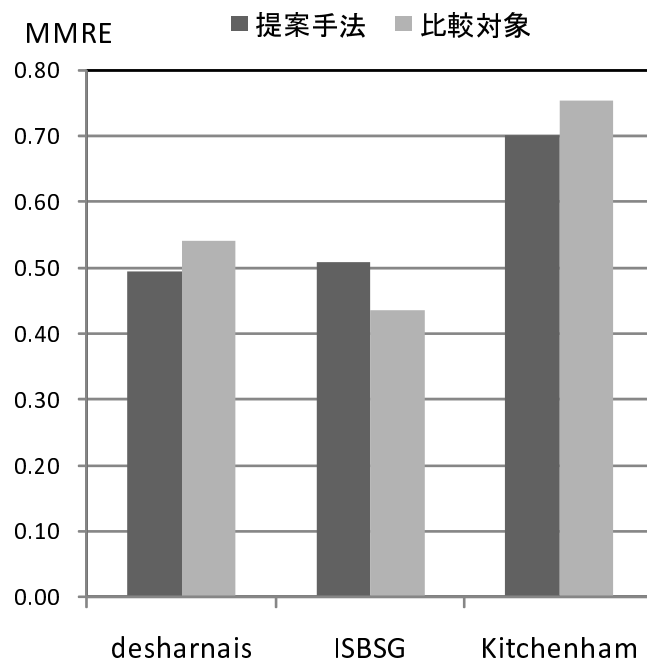


図 7: 提案手法と関連研究の比較