

---

# 協調フィルタリングに基づく工数見積もりのロバスト性評価

Experimental Evaluation for Robustness of CF-Based Effort Estimation

柿元 健\* 角田 雅照† 大杉 直樹‡ 門田 暁人§ 松本 健一¶

**Summary.** Effort estimation is important for software development. Effort estimation uses past project's data including product and process metrics. Generally, project data contain significant amount of missing values, which cause the degradation of the estimation performance. We focus on a Collaborative Filtering based (CF-based) effort estimation technique to solve this problem. In this paper, we evaluate an estimation accuracy of CF-based estimation technique by using various types of dataset, i.e. varieties of missing value rate and distribution. The results showed that the CF-based estimation technique was more effective than conventional regression models for all types of data.

## 1 はじめに

ソフトウェア開発プロジェクトにおいて、工数の見積もりは適切な資源の配置、及び開発コストや納期を決定する上で非常に重要であり、多くの工数見積もり方法が提案されている [1] [3] [9]。工数見積もりは、過去のプロジェクトで計測されたデータを用いて数学的な見積もりモデルを作成し、現行のプロジェクトで計測されたデータをモデルに代入することで見積もり値を算出する。モデルを作成するために用いるデータは、ソフトウェアメトリクス (または、単にメトリクス) と呼ばれる尺度の下で計測される。メトリクスには、開発するソフトウェアの規模や複雑さなどを表すメトリクスや、開発工程で消費された工数や、作業の進捗度合いを表すメトリクスがある。

しかし、過去に計測されたメトリクスには、記録されていない値 (欠損値) が含まれる可能性が高く、工数見積もりを行う上で見積もりモデルの作成が行えないという問題を引き起こす。欠損値が発生する原因には、プロジェクトによって異なるメトリクスを記録している、同じプロジェクトであっても時間経過とともに記録するメトリクスに変化が生じる、不注意などにより記録漏れが生じる、あるいは記録を紛失するといったことが挙げられる。特に工数や作業の進捗度合いなど、作業担当者などによって手動で計測されるメトリクスは後から再計測することが困難であるため欠損値が含まれやすい。

この問題を解決するための手法として欠損値処理がある [4] [5] [10]。欠損値処理とは、欠損値を含むプロジェクトやメトリクスをデータから除去する、あるいは、欠損

---

\*Takeshi Kakimoto, 奈良先端科学技術大学院大学 情報科学研究科

†Masateru Tsunoda, 奈良先端科学技術大学院大学 情報科学研究科

‡Naoki Ohsugi, 奈良先端科学技術大学院大学 情報科学研究科

§Akito Monden, 奈良先端科学技術大学院大学 情報科学研究科

¶Ken-ichi Matsumoto, 奈良先端科学技術大学院大学 情報科学研究科

値に何らかの値を補完する，などして欠損値を含まないデータを作成する手法である．欠損値処理により見積もりモデルの作成が可能となるが，データ全体に対する欠損値の割合（欠損率）が30%以上の場合や，欠損値分布に偏りがある場合には，見積もり精度が著しく低下するなどの問題が生じる[4]．一方，実際のソフトウェア開発から得られるデータは欠損率が30%以上，あるいは，欠損値分布に偏りがあることも少なくない[2]．

欠損率が高く，欠損値分布に偏りがある場合に，従来手法よりも高い精度の見積もりを行う手法として協調フィルタリングに基づく見積もり法（CF-based 見積もり法）が提案されており，企業のソフトウェア開発で得られた，欠損率が約60%，欠損値分布に偏りがあるデータを用いた場合，CF-based 見積もり法の精度が，従来手法である欠損値処理を適用したステップワイズ重回帰分析の精度よりも高いことが報告されている[6][11]．しかし，実際に得られるデータにおいて欠損率や，欠損値分布は様々であり，どの程度の欠損率まで，あるいは，どのような欠損値分布に対してCF-based 見積もり法が有効であるのかは明らかではない．

本論文では，欠損率や欠損値分布の偏りに対するCF-based 見積もり法のロバスト性（どの程度精度が低下しないか）を実験的に評価し，明らかにする．しかし，欠損率や欠損値分布の偏りが様々であるソフトウェア開発の過程で得られたデータを用意することは難しい．そこで，まず，企業におけるソフトウェア開発の過程で得られたデータから欠損値が含まれるプロジェクトやメトリクスを除去し，欠損値を含まないデータ（基礎データ）を用意した．次に，基礎データをもとに欠損率や欠損値の分布を変化させて欠損値を与え，実験に用いるためのデータ（実験データ）を作成した．各実験データを用いてCF-based 見積もり法と従来手法の精度を評価した．実験データの作成に当たっては，三種類の欠損値分布（Missing Completely At Random, Missing At Random, nonignorable missingness）を考え，各分布について欠損率を変化させて欠損値を作成した．実験データに含まれる実際の工数を，仮にその工数が欠損値であると考えて見積もりを行い，見積もり値と実測値の差から各見積もり法を評価した．

## 2 工数見積もり法

工数見積もり法として，ソフトウェア開発見積もりモデルとして従来から数多く提案されている手法であるステップワイズ重回帰分析と，欠損値に対してロバストな手法である協調フィルタリングがある．それぞれの手法について以下で述べる．

### 2.1 ステップワイズ重回帰分析

多変量解析の手法の一つである重回帰分析は，ソフトウェア開発見積もりモデル作成において数多く用いられている．重回帰分析は，ある変数（目的変数）と，それに対して影響すると考えられる複数の変数（説明変数）間の関係式を求め，その関係式に基づいて説明変数から目的変数を見積もる手法である．工数見積もりにおいては，見積もる工数が目的変数であり，それ以外のメトリクスが説明変数となり得る．

目的変数に対して強く影響する変数を説明変数として選択する手法の一つがステップワイズ変数選択法である．ステップワイズ変数選択法は以下の手順で行われる．(1) 初期モデルを作成する，(2) 作成されたモデルに対して，各説明変数の係数が0でないかの検定を行い，有意水準5%で棄却されなかった場合に変数を選択し，有意水準

10%で棄却される場合には変数を除去する，(3)手順(2)ができなくなるまで繰り返す．

欠損値を含むデータに対してステップワイズ重回帰分析を行うことはできない．そこで，欠損値処理を用いて欠損値を含まないデータを作成し，作成したデータに対してステップワイズ重回帰分析を行う．以下の三種類の欠損値処理が広く用いられている [4] [10]．

リストワイズ除去法 欠損値を一つでも含んでいるプロジェクトは除去し，欠損値を全く含まないプロジェクトのみで回帰分析を行う．

ペアワイズ除去法 欠損値でない値を可能な限り用い，かつメトリクス間の相関係数が高い組み合わせを選択してデータを作成し回帰分析を行う，

中間値挿入法 欠損値に対して，その変数の全ての欠損値を除いた値の平均値を計算し，欠損値を計算された平均値で補完することで作成された欠損値を含まないデータを用いて回帰分析を行う．

しかし，含まれる欠損値が多い場合，及び欠損値分布に偏りが生じている場合には，欠損値処理を用いてもステップワイズ重回帰分析は有効に働かない．

## 2.2 協調フィルタリング

協調フィルタリングは，データに欠損値が含まれることを前提とした見積もり法である．協調フィルタリングは，情報検索の分野において，非常に多くの情報からユーザの好みそうな情報を推薦するシステムを実現させるために発展してきた．協調フィルタリングは以下の手順で推薦を行う．

1. あらかじめユーザがアイテム（書籍，映画，楽曲など）を評価しておく．
2. 推薦対象のユーザと評価が似ているユーザ（類似ユーザ）を選び出す．
3. 類似ユーザの対象アイテムの評価を用いて推薦対象のユーザの対象アイテムの評価を見積る．
4. 推薦対象のユーザの対象アイテムの評価が高いと見積もられた場合，推薦対象ユーザに対して対象アイテムを推薦する．

Resnick ら [7]は，非常に多くのアイテムの中から個々のユーザの好むアイテムを推薦するシステムを開発した．また，Sarwar ら [8]は，アイテム数が非常に多く，ユーザによる評価が全体の1%以下しか評価が行われていないデータにおいてユーザの好みを見積もるアルゴリズムを提案した．このアルゴリズムは Amazon.com<sup>1</sup>の本推薦システムにおいても利用されている．

協調フィルタリングは，欠損値を大量に含む場合においても評価値を見積り推薦を行うことができる．そこで，同様に欠損値を含むデータを用いて見積りを行う工数見積りに対して適用することで高い精度で見積りを行うことができると考えられる．しかし，ユーザの評価を用いた推薦システムでは，ユーザの評価という値域が一定なデータを用いるのに対して，工数見積りにおいては，値域が一定でないデータであるメトリクスを用いる．文献 [6]や [11]において，値域が一定でない場合においても有効な協調フィルタリングのアルゴリズムが提案されており，工数及び信頼性を見積りにおける有効性が確認されている．

<sup>1</sup><http://www.amazon.com/exec/obidos/subst/home/home.html>

工数見積もりにおける協調フィルタリングを用いた見積もりは以下の手順で行われる．(1) メトリクス間の値域の違いを小さくするためにメトリクスの値を正規化する(メトリクス正規化)，(2) 正規化されたメトリクスの値を用いて，見積もり対象プロジェクトと類似したプロジェクトを算出する(類似度計算)，(3) 類似したプロジェクトの見積もり対象メトリクスを用いて見積もり対象プロジェクトの見積もり値を算出する(見積もり計算)．

上記手順のメトリクス正規化，類似度計算，見積もり計算のそれぞれの手法には，いくつかの手法が存在する．事前実験において最も高い見積り精度が得られた手法を実験で利用した．それぞれの手法について以下で述べる．

**メトリクス正規化** メトリクス正規化には，Z-score 法を用いた [10]．プロジェクト  $i$  のメトリクス  $j$  の値  $m_{i,j}$  の正規化された値  $Z(m_{i,j})$  は式 (1) で求められる．

$$Z(m_{i,j}) = \frac{m_{i,j} - \mu_j}{\sigma_j} \quad (1)$$

ここで， $\mu_j$  はメトリクス  $j$  の値の平均値， $\sigma_j$  はメトリクス  $j$  の値の標準偏差である．

**類似度計算** 類似度計算には，Cosine Similarity 法を用いた [8]．プロジェクト  $i$  とプロジェクト  $a$  の類似度  $sim_{i,a}$  は式 (2) で求められる．

$$sim_{i,a} = \frac{\sum_{j \in M_i \cap M_a} Z(m_{i,j}) \times Z(m_{a,j})}{\sqrt{\sum_{j \in M_i \cap M_a} (Z(m_{i,j}))^2} \sqrt{\sum_{j \in M_i \cap M_a} (Z(m_{a,j}))^2}} \quad (2)$$

ここで， $M_i$ ， $M_a$  はプロジェクト  $i$ ， $a$  の欠損値でないメトリクスの集合である．

**見積もり計算** 見積もり計算には，Weighted Sum 法を用いた [8]．プロジェクト  $i$  のメトリクス  $j$  の値  $m_{i,j}$  の予測値  $\hat{m}_{i,j}$  は式 (3) で求められる．

$$\hat{m}_{i,j} = \frac{\sum_{k \in k_{nearestProjects}} (sim_{i,k} \times m_{k,j})}{\sum_{k \in k_{nearestProjects}} sim_{i,k}} \quad (3)$$

ここで， $k_{nearestProjects}$  は，プロジェクト  $i$  との類似度が高い  $k$  個のプロジェクトの集合である．

### 3 欠損値作成法

欠損値作成法とは，データに対して任意の欠損率や欠損値分布の偏りを持った欠損値を与える手法である．ソフトウェア開発において実際に得られた欠損率や欠損値分布の偏りの異なるデータを用意することは難しいが，欠損値作成法を用いることにより，擬似的なデータを作成することが可能である．欠損値作成法には，作成する欠損値分布の偏りによって Missing Completely At Random (MCAR)，Missing At Random (MAR)，nonignorable missingness の三種類がある [10]．これらの手法は，当初は欠損値処理において提案された欠損のメカニズムである [5]．以下，それぞれの手

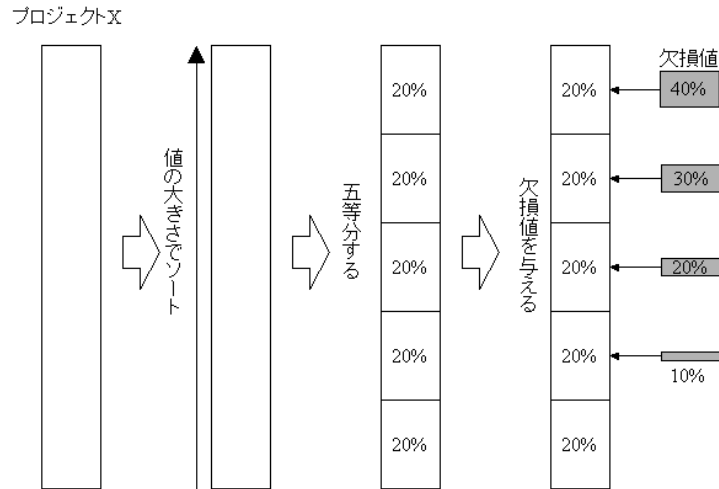


図1 MARの欠損値作成手順

法による欠損値の作成法について述べる。

**Missing Completely At Random (MCAR)** Missing Completely At Random (MCAR) は、分布の偏りが全くない欠損値を作成する手法である。つまり、全くランダムに欠損値を作成する手法である。MCARは、最も容易に欠損値が作成でき、任意の欠損率で欠損値の作成が可能である。

**Missing At Random (MAR)** Missing At Random (MAR) は、ある変数(依存変数)の大きさに従って、他の変数に対して偏りを持った欠損値を作成する手法である。作成の手順は、(1) 依存変数の大きさに従って他の変数をソートする、(2) ソートされたデータを5つに等分する、(3) 5つに等分されたセットに対して順に作成する欠損値の40%、30%、20%、10%、0%の欠損値をランダムに作成する。図1に作成手順を示す。例として、100プロジェクトのデータ中の、ある変数に対して10個の欠損値を与える場合を考えると、ソートされた100個のうち、最初の20個に4個の欠損値を作成し、以下2番目に3個、3番目に2個、4番目に1個、5番目に0個の欠損値を作成する。MARでは、最大50%の欠損率まで任意に欠損値の作成が可能である。

**nonignorable missingness** nonignorable missingness は、欠損する変数自身の大きさに従って、偏りを持った欠損値を作成する手法である。作成手順は、それぞれの変数自身の大きさに従って変数をソートすること以外はMARと同じである。nonignorable missingnessでも、最大50%の欠損率まで任意に欠損値の作成が可能である。

#### 4 実験

実験は、欠損率や欠損値分布の偏りが様々であるデータにおける、CF-based見積もり法の見積もり精度を確かめるために行った。見積もり精度を確かめるために、見積もり値と実際に得られた値(実測値)の差を使って、ステップワイズ重回帰分析で見積もりを行った結果との比較を行った。以下で実験の詳細について述べる。

表 1 実験データの各メトリクスの値の特徴

	設計工数	製造工数	試験工数	開発規模	設計工数 (正社員)
平均値	1.153	2.0613	1.446	233.047	0.134
中間値	0.085	0.134	0.0703	43.9	0.0171
分散	47.455	174.62	101.076	1092691.039	0.608
最大値	67.458	144.036	111.876	11301	8.724
最小値	0.001	0.001	0.001	0.4	0
	設計工数 (派遣)	設計工数 (外注)	製造工数 (正社員)	製造工数 (派遣)	製造工数 (外注)
平均値	0.696	0.0096	0.034	0.117	1.507
中間値	0.0421	0	0.0025	0.026	0
分散	25.808	0.002	0.012	0.067	152.447
最大値	58.734	0.488	0.981	1.586	144.036
最小値	0	0	0	0	0

#### 4.1 実験データ

実験で利用したデータは、プロジェクト数 140 件、メトリクス数 10 個の欠損値を含まないデータである。実験データに含まれる 10 個のメトリクスとは、設計工数、製造工数、試験工数、開発規模、設計工数(正社員分)、設計工数(派遣社員分)、設計工数(外注分)、製造工数(正社員分)、製造工数(派遣社員分)、製造工数(外注分)である。実験データは、ある企業で得られたプロジェクト数 1081 件、メトリクス数 14 件、欠損率 59.83% の基礎データから欠損値を含まないように抽出したものである。また、全ての値が、あらかじめ特定の値で除算されているデータである。実験データの各メトリクスの値の特徴を表 1 に示す。

#### 4.2 評価基準

見積もり精度として、絶対誤差平均、絶対誤差中間値の二種類の評価基準を用いて評価を行った。絶対誤差平均、絶対誤差中間値はともに値が小さいほど見積もり精度が高いことを表す。

それぞれの評価基準は以下の式(4)、(5)で計算される。ここで、あるメトリクスに  $M$  個のデータがあるとする。また、実測値と見積もり値をそれぞれ  $X_i$ 、 $\hat{X}_i$  ( $i = 1 \sim M$ ) とし、 $A_i = |\hat{X}_i - X_i|$  とおく。

絶対誤差平均

$$\text{絶対誤差平均} = \frac{\sum_{i=1}^M A_i}{M} \quad (4)$$

絶対誤差中間値

$$\text{絶対誤差中間値} = \begin{cases} A_n & M = \text{奇数}(2n - 1) \\ & (A_1 \leq A_2 \leq \dots \leq A_n \leq \dots \leq A_{2n-1}) \\ \frac{A_n + A_{n+1}}{2} & M = \text{偶数}(2n) \\ & (A_1 \leq \dots \leq A_n \leq A_{n+1} \leq \dots \leq A_{2n}) \end{cases} \quad (5)$$

4.3 実験手順

実験手順を以下に示し、実験におけるデータの加工を示した図を図2に示す。以後、見積もりモデルを作成するためのデータをフィットデータ、作成した見積もりモデルを使って実際に見積もりを行うデータをテストデータと定義する。

1. 4.1節で述べた実験データを無作為にフィットデータとテストデータにプロジェクト70件ずつに均等に分けた。この工程を10回繰り返し、異なる10組のフィットデータとテストデータの組を作成した。
2. 作成した10個のフィットデータに対して3章で述べた欠損値作成法を用いて欠損値を与えた。MCARでは欠損率0～90%で、MARとnonignorable missingnessでは欠損率0～50%で、それぞれ10%刻みで欠損値を作成した。
3. CF-based 見積もり法による見積もりを行うために必要な、類似プロジェクト数(式(3)の  $k - \text{nearestProjects}$ )を決定するために、一部のフィットデータとテストデータの組を使って類似プロジェクト数を変化させて見積もりを行った。
4. 3で決定した類似プロジェクト数を使って、全てのフィットデータとテストデータの組に対して2.2節で述べたCF-based 見積もり法により見積もりを行い、4.2節で述べた評価基準をそれぞれ求めた。見積もりを行ったメトリクスは試験工数であり、テストデータの試験工数は存在しないものとして見積もりを行った。
5. 同様にして、CF-based 見積もり法と同じフィットデータとテストデータの組を用いて、2.1節で述べたステップワイズ重回帰分析により見積もりを行った。リス

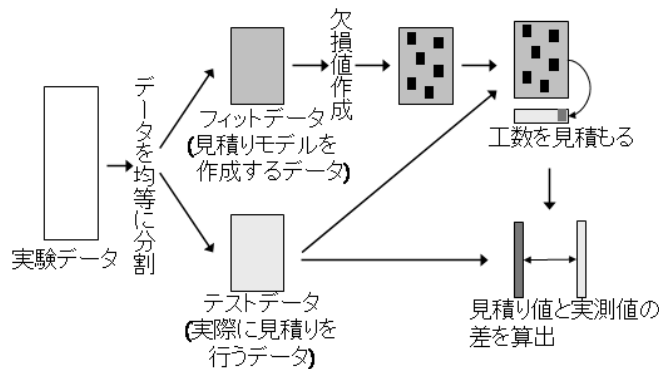


図2 実験におけるデータの加工

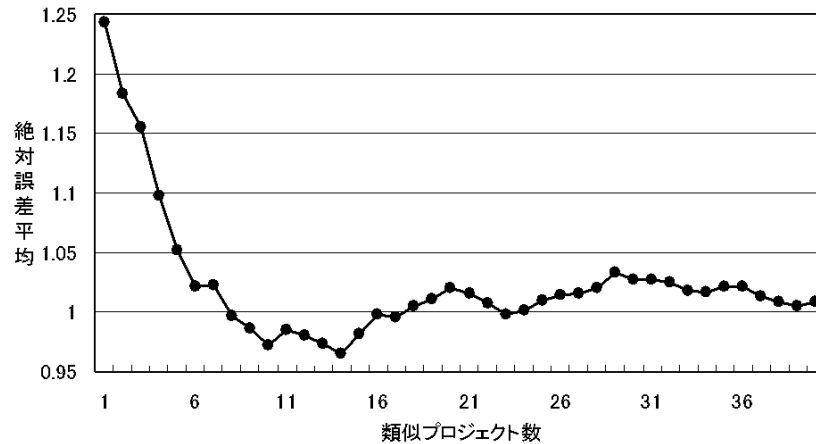


図3 各類似プロジェクト数における絶対誤差平均

トワイズ除去法，ステップワイズ除去法，平均値挿入法の三種類の欠損値処理それぞれにおいてステップワイズ重回帰分析を用いて試験工数の見積もり，4.2節で述べた評価基準をそれぞれ求めた．

- それぞれの欠損値作成法において，4，5で求めた評価基準を用いて比較し，CF-based 見積もり法の有効性を確認した．

## 5 結果と考察

実験結果を，最初に類似プロジェクト数の決定について述べ，その後，それぞれの欠損値作成法における評価基準の比較について述べる，最後に欠損率や欠損値分布の偏りが様々であるデータにおける CF-based 見積もり法の精度についての考察を行う．

### 5.1 類似プロジェクト数の決定

各類似プロジェクト数における絶対誤差平均を表したグラフを図3に示す．グラフの縦軸は絶対誤差平均を，横軸は類似プロジェクト数を示す．類似プロジェクト数を決定するために，欠損値作成法 MCAR，欠損率 50% で 5 組のフィットデータとテストデータを用いて類似プロジェクト数を 1 ~ 40 個まで変化させて見積もりを行い絶対誤差平均を算出した．この結果から類似プロジェクト数 14 で最も見積もり精度が高いことがわかる．そこで，類似プロジェクト数を 14 として以後の実験手順を行った．

### 5.2 実験結果

実験結果として，MCAR，MAR，nonignorable missingness のそれぞれの欠損値作成法ごとに評価基準の比較を行う．評価基準は，10 組のフィットデータとテストデータにおいて CF-based 見積もり法とステップワイズ重回帰分析，それぞれで見積もりを行って算出された，絶対誤差の平均値と中間値である．

MCAR，MAR，nonignorable missingness で欠損値を作成したときの，各欠損率における CF-based 見積もり法とステップワイズ重回帰分析による評価基準を表した



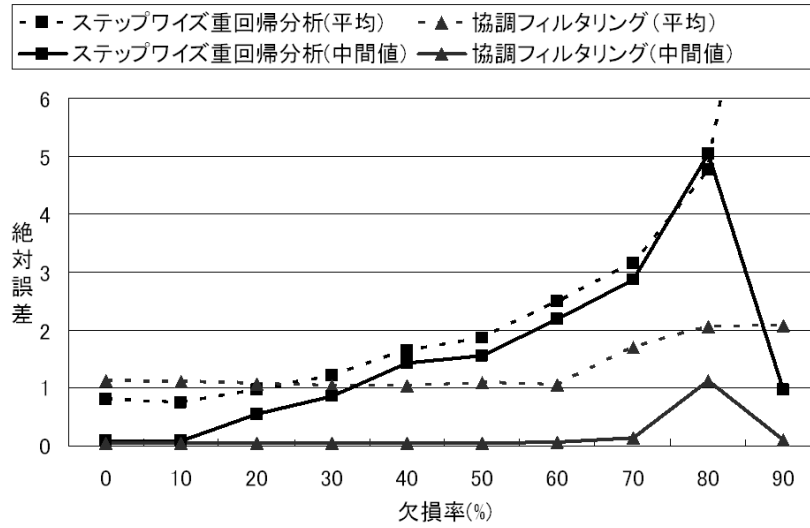


図4 各欠損率における評価基準 (MCAR)

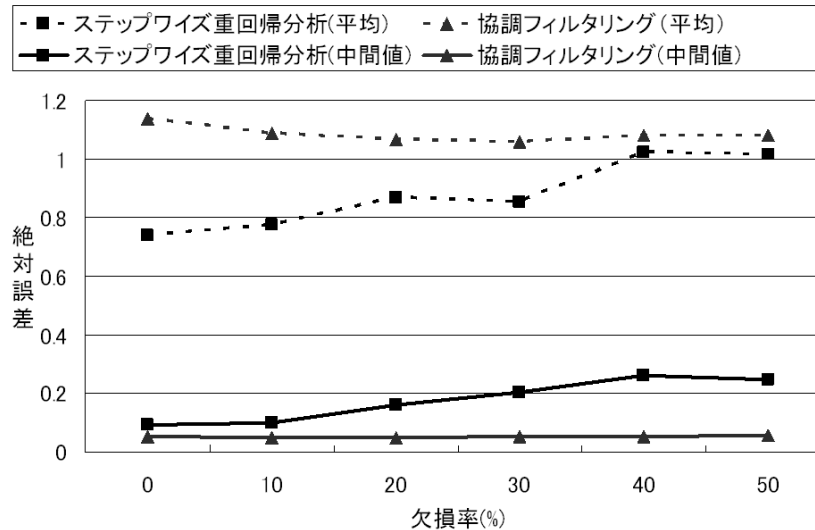


図5 各欠損率における評価基準 (MAR)

グラフを図4～6に示す．それぞれのグラフの縦軸は絶対誤差を，横軸は欠損率を示す．ステップワイズ重回帰分析の結果は，三種類の欠損値処理の結果のうち最も精度が高い，つまり絶対誤差が小さい手法の結果を選択してプロットした結果である．

各グラフは，CF-based 見積もり法は，ステップワイズ重回帰分析よりも，欠損率，欠損値作成法に関わらず高い精度を絶対誤差中間値において示している．これより，CF-based 見積もり法を用いた見積りを使って，欠損率や欠損値分布の偏りが様々であるデータに対して高い見積り精度が得られることがわかる．また，CF-based 見積もり法では，絶対誤差平均，絶対誤差中間値ともに，欠損率に関わらずほ

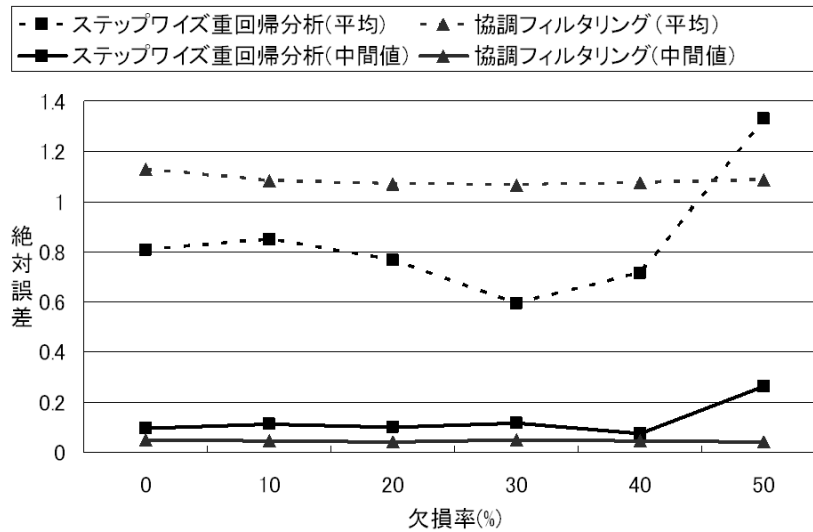


図6 各欠損率における評価基準 (nonignorable missingness)

ば一定した精度が得られた。これより、CF-based 見積もり法は、欠損率による影響を受けずに見積りを行えていることがわかる。

しかし、絶対誤差平均では、特に欠損率が低い場合においてステップワイズ重回帰分析よりも低い精度を示している。また、CF-based 見積もり法、ステップワイズ重回帰分析ともに平均値が中間値よりも大きな値となっており、両者の間には大きな差が生じている。これより、見積り精度が著しく悪い結果となる場合が、CF-based 見積り手法、ステップワイズ重回帰分析ともにあることを示している。CF-based 見積り手法の方が精度がより低下しているため、CF-based 見積り手法では見積り精度が著しく低下する場合が多い、もしくは、見積り精度が著しく低下する場合はステップワイズ重回帰分析以上に見積り精度が低下することが考えられる。しかし、MCAR では欠損率40%以上と、nonignorable missingness の欠損率50%ではステップワイズ重回帰分析よりも高い精度を示しており、MARにおいても、欠損率が高くなるほどステップワイズ重回帰分析との差は小さくなっている。そこで、絶対誤差平均値においても高い欠損率においては、CF-based 見積もり法が、ステップワイズ重回帰分析よりも精度の高い見積りが行えると考えられる。

また、CF-based 見積もり法、ステップワイズ重回帰分析ともに、MCARにおける見積り精度が、MAR、nonignorable missingness の見積り精度と比べて低い結果を示している。これは、欠損値作成法の過程においてMAR、nonignorable missingness では値域が狭くなるのに対して、MCARでは値域はほぼ変わらないために見積り精度が低下していると考えられる。

### 5.3 考 察

実験結果では、CF-based 見積もり法は、欠損率、欠損値分布の偏りに関わらずステップワイズ重回帰分析よりも高い精度が得られることを示した。しかし、見積りにくいデータにおける見積りでは、低い欠損率においてステップワイズ重回帰分析より

も低い精度しか得られず、また、正規化を行っているにも関わらず値域が広い場合において見積り精度が低下した。そのため、このようなデータに対しても高い見積り精度で見積りが行えるように、現在は2.2節で述べた各計算法一種類のみ使用して見積りを行っているが、より適切なアルゴリズムを選択して見積りを行えるように今後改善する必要がある。

本論文の限界のひとつとして、MCAR, MAR, nonignorable missingness の三種類の欠損値作成法を、それぞれ独立に使用してのみ欠損値を作成していることである。実際に企業で得られるデータにおいては、様々な要因で欠損値が発生するために、欠損値が発生するメカニズムも様々である。また、欠損値分布の偏りがプロジェクト間のみで、メトリクス間には偏りが生じていない。そこで、より実際のデータに類似したデータを作成する必要がある。三種類の欠損値作成法を混在させて欠損値を作成して見積りを行うことや、メトリクス間に欠損値分布の偏りを生じさせることによって、より多様なデータを作成できる可能性がある。さらに、欠損値を含まないデータとして一種類のデータしか用いていないので、複数の別のデータを用いて欠損値を含まないデータを作成して実験を行うことで、より信頼性のある結果を得ることができる。

## 6 まとめ

本論文では、協調フィルタリングに基づいた見積り法 (CF-based 見積もり法) の欠損率、欠損値分布の偏りといった、様々な欠損値に対するロバスト (堅牢) 性を確かめた。実験の結果、従来手法であるステップワイズ重回帰分析よりも高い精度の見積り結果が得られ、CF-based 見積もり法は従来手法よりも様々な欠損値に対してロバストであると言える。

今後は、欠損値作成法をより工夫して適用することで多様なデータを作成し実験を行う予定である。また、複数の欠損値を含まないデータを用いて実験を行うことで結果の信頼性を向上させる。さらに、見積りを行うデータに対してより適切なアルゴリズムを選択して見積りを行って見積もり精度の改善を行う予定である。

謝辞 本研究の一部は、文部科学省「e-Society 基盤ソフトウェアの総合開発」の委託に基づいて行われた。

## 参考文献

- [1] Boehm, B.W.: Software Engineering Economics. IEEE Trans. on Software Eng., vol.10, no.1, 4-21 (1984)
- [2] Briand, L., El Eman, K., and Wiczorek, I.: Explaining the Cost of European Space and Military Projects. In Proc. Int'l Conf. Software Eng., vol.1, no.1, pp.61-88 (1996)
- [3] Conte, S.D., Dunsmore, H.E., and Shen, V.Y.: Software Engineering Metrics and Models. The Benjamin/Cummings Publishing Company, Inc., Menlo Park, California (1986)
- [4] Kromrey, J., and Hines, C.: Nonrandomly Missing Data in Multiple Regression: An Empirical Comparison of Common Missing-Data Treatments. Educational and Psychological Measurement, vo.54, no.3, pp.573-593 (1994)
- [5] Myrtveit, I., Stensrud, E., and Olsson, U. H.: Analyzing data sets with missing data: An empirical evaluation of imputation methods and likelihood-based methods, IEEE Trans. on Software Eng., vol.27, no.11, pp.999-1013 (2001)
- [6] Ohsugi, N., Tsunoda, M., Monden, A., Matsumoto, K., "Applying collaborative filtering for effort estimation with process metrics," 5th International Conference on Product Focused Soft-

- ware Process Improvement (Profes2004), Kyoto, Japan, Lecture Notes in Computer Science, Vol. 3009, pp.274-286, (2004)
- [7] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J.: GroupLens: An Open Architecture for Collaborative Filtering of Netnews. In Proc. ACM Conf. on Computer Supported Cooperative Work (CSCW'94), Chapel Hill, North Carolina, United States, pp.175-186 (1994)
- [8] Sarwar, B.M., Karypis, G., Konstan, J.A., and Riedl, J.: Item-Based Collaborative Filtering Recommendation Algorithms, In Proc. 10th International World Wide Web Conference (WWW10), Hong Kong, pp.285-295 (2001)
- [9] Shepperd, M., and Schofield, C.: Estimating Software Project Effort Using Analogies. IEEE Trans. on Software Eng., vol.23, no.12, pp.76-743 (1997)
- [10] Strike, K., El Eman, K., and Madhavji, N.: Software Cost Estimation with Incomplete Data. IEEE Trans. on Software Eng., vol.27, no.10, pp.890-908 (2001)
- [11] 角田 雅照, 大杉 直樹, 門田 暁人, 松本 健一: 協調フィルタリングに基づくソフトウェア信頼性予測方法. 電子情報通信学会技術報告, ソフトウェアサイエンス研究会, 福岡, No.SS2003-27, pp.19-24, (2003)