論　文

協調フィルタリングに基づく工数見積り手法のデータの欠損に対するロバスト性の評価

柿元　健[†]　角田　雅照[†]　大杉　直樹[†]　門田　暁人[†]
松本　健一[†]

Evaluating the Robustness of Collaborative Filtering Based Effort Estimation Method against Data Missing

Takeshi KAKIMOTO[†], Masateru TSUNODA[†], Naoki OHSUGI[†], Akito MONDEN[†], and Ken-ichi MATSUMOTO[†]

あらまし　本論文では，協調フィルタリングに基づいた工数見積り手法（CF-based 見積り手法）が，データの欠損に対してどの程度ロバストであるかを実験により評価する．CF-based 見積り手法は，ソフトウェア開発プロジェクトの特性値（規模，工期など）を説明変数とし，目的変数である工数を見積もる手法であり，説明変数に未記録の値（欠損値）が含まれる場合にも適用できることが特長である．ただし，欠損値の生じるメカニズムの違い，及び，欠損率の変化が，見積り精度に与える影響（ロバスト性）は従来明らかでなかった．本論文では，欠損値が生じる三つのメカニズムを想定し，それぞれについて欠損率を変化させたデータセットを多数作成し，各データセットを用いて工数見積りを行うことで，ロバスト性を実験的に評価した．実験の結果，CF-based 見積り手法が，従来手法であるステップワイズ重回帰分析と欠損値処理を併用する手法よりもロバスト性が高い，すなわち，欠損のメカニズムにかかわらず，欠損率が増大しても見積り精度が大きく低下しないことが示された．

キーワード　特性値，MDT，欠損メカニズム，欠損率，ステップワイズ重回帰分析

1. まえがき

ソフトウェア開発プロジェクトにおいて，工数の見積りは，適切な資源の配置，及び，スケジュール管理を行う上で非常に重要であり，多くの工数見積り手法が提案されている [1], [2], [4], [15], [16]．これらの工数見積り手法では，プロジェクトの特性値（規模，開発期間など）を説明変数，工数を目的変数とする見積りモデルを過去プロジェクトの実績データに基づいて構築し，現行プロジェクトで計測した特性値をモデルに代入することで見積り値を算出する．

見積りモデルの構築は，多くの場合，欠損値（未記録の値）を含んだ実績データを用いて行うことが要求される．一般に，多数のプロジェクトから多種類の特性値を計測して得られるデータセットには，欠損値が数多く含まれるためである．例えば，情報処理推進機構ソフトウェア・エンジニアリング・センターが 15 社から収集した 1,009 件のプロジェクト，約 400 種類の特性値の実績データでは，全体の 87.7%が欠損値であり，記入を必須とした項目に限っても 36.9%が欠損値である [5]．また，一つの企業内で収集された 1,081 件のプロジェクト，14 種類の特性値の実績データにおいても，60%の欠損値を含んでいたことが報告されている [18]．海外においても，International Software Benchmarking Standards Group（ISBSG）において収集された，20 か国，3024 件のプロジェクト，99 種類の特性値の実績データは，57.6%の欠損値を含んでいる [6]．データ欠損の原因は様々であり，時間的制約や不注意による記録漏れ，組織内で統一的な取決めがないためにプロジェクトごとに計測した特性値の種類が異なる，などが挙げられる．

欠損値を含むデータセットを用いて見積りモデルを構築する一つの方法は，欠損値処理（MDT：Missing Data Techniques）を併用することである

[†] 奈良先端科学技術大学院大学情報科学研究科，生駒市
Graduate School of Information Science, Nara Institute of Science and Technology, 8916–5 Takayama, Ikoma-shi, 630–0192 Japan

[7], [10], [17]．MDT とは，与えられたデータセットから欠損値を含むプロジェクトを除外したり，欠損値を何らかの値で補完したりすることによって，欠損値を含まないデータセットを作成する方法のことである．典型的な例として，欠損値を含むデータセットに対し，MDT を適用後，ステップワイズ重回帰分析により（回帰モデルを構築して）工数を見積もることが行われている [17]．MDT は，データ全体に対する欠損値の割合（欠損率）が 30%以下で，かつ，欠損値が均一に生じている場合に有効である [7]．しかし，欠損率が 30%を超える場合，及び，欠損値が均一に生じていない場合には見積り精度が著しく低下することが報告されている [7]．現実には，データの欠損率が 30%を超えることも多く，欠損のメカニズムも多様であるため，MDT の適用範囲は限られる．

もう一つの方法は，欠損していない値のみを用いて予測を行うことであり，その具体的な手法として，協調フィルタリングに基づく工数見積り手法 [12], [18]（以降，CF-based 見積り手法と呼ぶ）や Optimized Set Reduction（OSR）法 [3] が提案されている．CF-based 見積り手法は，約 60%の欠損値を含むデータセットを用いた事例において，MDT 適用後にステップワイズ重回帰分析を行った場合と比較して，見積り値の相対誤差が，22.11 から 0.79 に改善されたことが示されている [18]．このことから，データの欠損率が 30%を超える場合の見積り手法として有望であるといえる．しかし，適用事例はこの一つのみであり，欠損率や欠損値の分布が異なるデータセットに対しても高い見積り精度が得られるかどうかは不明である．CF-based 見積り手法を開発現場で採用するためには，欠損値の生じるメカニズムの違い，及び，欠損率の変化が見積り精度に与える影響（ロバスト性）を明らかにすることが望ましい．

本論文では，欠損値が生じる三つのメカニズムを想定し，それぞれについて欠損率の異なるデータセット（欠損率 10%，20%，30%，…）を用意し，各データセットを用いて工数見積りを行うことで，CF-based 見積り手法のロバスト性を実験的に評価する．評価にあたっては，MDT 適用後にステップワイズ重回帰分析を行った場合と比較する．これは，データ欠損を含むデータセットに対して一般的に用いられ，見積り手法の評価においても，比較対象としてしばしば用いられる手法である [9]．三つの欠損メカニズムは，ソフトウェア工学分野において従来想定されている Missing Completely At Random（MCAR），Missing At Random（MAR），Nonignorable Missingness（NM）を用いる [8], [17]（**3. 1** 参照）．ただし，これらの欠損のメカニズムにそれぞれ合致し，かつ，欠損率 10%，20%，30%，… を満たすような評価用のデータセット群を，現実のプロジェクトから直接得ることは難しい．そこで，本論文では，従来から使用されている欠損値を与える方法 [17] を用いて，それぞれの欠損メカニズム（MCAR，MAR，NM）に合うように意図的にデータを欠損させることで，評価用のデータセット群を作成することにした．本論文の評価の方法は，欠損値を与える方法以外も Strike らの方法 [17] を踏襲している．ただし，Strike らの評価対象は本論文とは異なり，MDT の各手法の性能である．Strike らは，欠損メカニズムと欠損率が異なるデータセット群を作成し，それらを用いて MDT を併用した重回帰分析を行い，MDT の各手法の性能比較をしている．

以降，**2.** で欠損値を考慮した工数見積り手法を説明し，**3.** でロバスト性の評価方法について述べる．**4.** では実験の方法と手順について説明し，**5.** で実験結果と結果に対する考察を述べる．最後に **6.** で本論文の結論について述べる．

2. 欠損値を考慮した工数見積り手法

2. 1 CF-based 見積り手法

協調フィルタリングは，多くの欠損値を含むデータセットを用いることを前提として，ユーザ間の類似性に基づいて各ユーザにアイテムの推薦を行うための基盤技術である [13], [14]．CF-based 見積り手法は，協調フィルタリングをソフトウェア開発分野での見積りに応用し，プロジェクト間の類似性に基づいて工数見積りを行う手法である [12], [18]．類似したプロジェクトを用いて見積りを行う手法として，CF-based 見積り手法のほかに事例ベース推論（CBR）[15] があるが，欠損を含まないデータセットの使用を前提としているため，本論文では評価の対象外とする．

CF-based 見積り手法は四つの手順（ダミー変数化，標準化，類似度計算，見積り値計算）から構成され，各手順で用いるアルゴリズムは複数の中から選択可能である．我々は，各種アルゴリズムを実装した CF-based 見積りソフトウェア（NCFE）をオープンソースとして公開している [11]．本論文では，NCFE を用いた予備実験で最も高い見積り精度が得られたアルゴリズムを採用した．以降では，各手順の詳細と採用したアル

ゴリズムについて述べる（他のアルゴリズムについては [11] を参照されたい）．

手順 1．ダミー変数化：データセットに名義尺度の特性値が含まれる場合，カテゴリーごとにダミー変数に置き換える．プロジェクト p_i の特性値 m_j のカテゴリー k のダミー変数 $d_{ij}(k)$ は式 (1) で定義される．

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

手順 2．特性値の標準化：各特性値は値域に大きなばらつきがあるため，値域をそろえるための標準化を行う．本論文では，標準化のアルゴリズムとしては Z-score [17] を用いた．Z-score では平均値が 0，分散が 1 となるように標準化され，データの分布が正規分布あるいは正規分布に近い場合 $[-2, 2]$ にデータの約 95%が含まれるように標準化する．本論文の評価実験で用いたデータセット（基礎データセット（**4.1** 参照））では 97.9%が $[-2, 2]$ に含まれた．プロジェクト p_i の特性値 m_j の値 v_{ij} を標準化した値 v'_{ij} は式 (2) で定義される．

$$v'_{ij} = \frac{v_{ij} - \mu_j}{\sigma_j} \quad (2)$$

ここで，μ_j は特性値 m_j の平均値，σ_j は特性値 m_j の標準偏差を表す．

手順 3．プロジェクト間の類似度計算：見積り対象のプロジェクトと類似した他のプロジェクトを見つけるため，プロジェクト間の類似度を算出する．類似度計算のアルゴリズムとしては，Cosine Similarity [14] を用いた．見積り対象のプロジェクト p_a と他の各プロジェクト p_i との類似度 $sim(p_a, p_i)$ は式 (3) で定義される．

$$sim(p_a, p_i) = \frac{\sum_{j \in M_a \cap M_i} v'_{aj} \times v'_{ij}}{\sqrt{\sum_{j \in M_a \cap M_i} {v'_{aj}}^2} \sqrt{\sum_{j \in M_a \cap M_i} {v'_{ij}}^2}} \quad (3)$$

ここで，M_a と M_i はそれぞれプロジェクト p_a と p_i に関して記録されている（欠損していない）特性値の集合を表す．

CF-based 見積り手法では，類似度を求める二つのプロジェクトにともに記録されている特性値の集合を用いて類似度を算出するため，欠損値を含むデータセットに対しても適用できる．

手順 4．類似度に基づく見積り値の算出：類似したプロジェクトの実測値を用いて，見積り対象のプロジェクトの目的変数（工数）を算出する．見積り値算出のアルゴリズムとしては，Weighted Sum [14] を用いた．プロジェクト p_a の目的変数 m_b の見積り値 $\hat{v}_{ab}$ は式 (4) で定義される．

$$\hat{v}_{ab} = \frac{\sum_{i \in k \text{ - } nearestProjects} (v_{ib} \times sim(p_a, p_i))}{\sum_{i \in k \text{ - } nearestProjects} sim(p_a, p_i)} \quad (4)$$

ここで，*k-nearestProjects* は，特性値 m_b が欠損しておらず，かつ，プロジェクト p_a と類似度の高い上位 k 個のプロジェクトの集合を表す．k の値は実験的に別途求める必要がある．

2.2 MDT とステップワイズ重回帰分析の併用

2.2.1 ステップワイズ重回帰分析

重回帰分析は多変量解析の一手法であり，ソフトウェア開発に要する工数を見積もるために広く用いられており，見積り手法の評価において，比較対照としてしばしば用いられる [9]．本論文の評価実験においても，比較対象として重回帰分析の一手法であるステップワイズ重回帰分析を用いた．

重回帰分析では，見積り対象の変数（目的変数）と，目的変数に影響を与える複数の変数（説明変数）との関係を表した一次式（回帰式）を作成する．回帰式中の各係数と定数は，見積り値の絶対誤差（残差）の二乗和が最小になるように決定される．作成された回帰式に，現行プロジェクトで計測した説明変数を与えることで，目的変数を見積もることが可能となる．

重回帰分析では，見積り精度を向上させるために，多数の説明変数候補の中から，見積り精度の向上に寄与すると予測される変数を選択して回帰式を作成する方法がとられる．ステップワイズ重回帰分析は，ステップワイズ変数選択法により採用する変数を決定し，重回帰分析を行う手法である．ステップワイズ変数選択は次の手順で行われる．

手順 1． 変数を全く含まないモデルを初期モデルとして作成する．

手順 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，指定した有意水準（本論文の評価実験では，偏 F 値の有意水準を $p_{in} = 0.05$, $p_{out} = 0.1$ とした）で棄却されない場合に変数を採択する．ただし，多重共線性を回避するために，採択す

る変数の分散拡大要因（VIF）が一定値（本論文の評価実験では 10 とした）以上の場合，またはその変数を採択することによって，他の変数の VIF が一定値以上となる場合，その変数は採択しない．

手順 3. 検定により適切な変数が選択されたと判断されるまで手順 2 を繰り返す．

2.2.2 欠損値処理（MDT：Missing Data Techniques）

欠損値を含むデータセットに対してステップワイズ重回帰分析を適用する場合には，MDT を併用する必要がある．MDT とは，多変量解析を可能とするために，与えられたデータセットから欠損値を含むプロジェクトを除外したり，欠損値を何らかの値で補完する，といった前処理を行う方法である．重回帰分析に対しては，次の 3 種類の MDT の手法が広く用いられる [7], [17].

リストワイズ除去法：欠損値を一つでも含むプロジェクトをすべて除去する．

ペアワイズ除去法：重回帰分析に特化した手法で，重回帰分析の過程において特性値間の相関を求める際に，相関を求める特性値のいずれかが欠損しているプロジェクトを除外して相関を求める [17].

平均値挿入法：欠損値に対して，当該特性値の平均値を挿入することで，欠損値を補完する．

これら 3 手法のうち，リストワイズ除去法が最も見積り精度を低下させない手法である [17]．ただし，リストワイズ除去法は，欠損率が高い場合にはすべてのプロジェクトが除去され見積り不可能となる．そこで，本論文の評価実験においては，上記の 3 手法すべてを用いてステップワイズ重回帰分析を行い，見積り可能な手法の中で最も高い精度を示した手法を採用し，ステップワイズ重回帰分析の結果として評価に用いた．

3. ロバスト性の評価方法

本論文では，Little ら [8] が定義した 3 種類（MCAR, MAR，NM）の欠損メカニズムを仮定し，それぞれについて欠損率の異なるデータセット（欠損率 10%, 20%, 30%, ···）を用意し，各データセットを用いて工数見積りを行うことで，CF-based 見積り手法のロバスト性を実験的に評価する．欠損値を含んだデータセットの構築は，Strike ら [17] の方法を踏襲する．

3.1 欠損メカニズム

3 種類の欠損メカニズムの詳細とデータ欠損の一例を次に示す．

Missing Completely At Random（MCAR）：欠損値が生じる確率が，データ中のどの変数にも依存しない．例えば，時間的制約や不注意によって特性値の記録漏れが生じた場合である．

Missing At Random（MAR）：欠損値が生じる確率が，欠損する値以外の変数に依存する．例えば，ソフトウェアの規模を表す尺度（ファンクションポイントなど）とコードレビューにおける発見バグ数が変数として含まれるデータを考える．規模が小さいプロジェクトではコードレビューが省略され，発見バグ数が欠損値となったとする．この場合，コードレビューにおける発見バグ数の欠損は規模に依存しているため，欠損メカニズムは MAR である．

Non-ignorable Missingness（NM）：欠損値が生じる確率が，欠損する値そのものに依存する．例えば，ソフトウェアの規模を表す尺度が変数として含まれるデータを考える．規模が小さいプロジェクトで，人員的余裕がないため規模自身が記録されなかったとする．この場合，欠損メカニズムは NM である．

これらの欠損メカニズムは，元来，統計の分野で考案されたものであるが，ソフトウェア開発データの欠損を表す手段としても用いられている [17].

3.2 実験データセット作成手順

指定された欠損率に基づいて，実験データセットを作成する手順（基礎データセットに欠損を与える手順）を，欠損メカニズムごとに示す．

MCAR

手順 1：全プロジェクトから，欠損値を与えるプロジェクトをランダムに選択する．

手順 2：手順 1 で選択したプロジェクトで欠損していない特性値の中から，一つをランダムに選択する．ただし，見積り対象の特性値（目的変数）は選択対象としない．

手順 3：手順 1 で選択したプロジェクトの，手順 2 で選択した特性値を欠損させる．

手順 4：指定された欠損率に達するまで手順 1〜3 を繰り返す．

MAR

手順 1：依存変数の値が小さい順にプロジェクトをソートし，ソートしたプロジェクトを先頭から五つのグループに均等に分ける（図 1）．本論文では Strike ら [17] と同様に依存変数として開発規模を用いた．すなわち，開発規模が小さなプロジェクトほど，データ収集の必要性が低かったり，データ収集に多くの時間

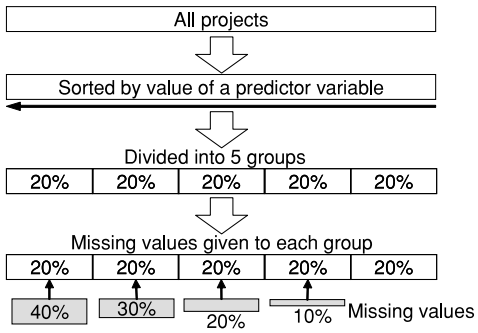

図 1　MAR と NM の欠損値生成手順
Fig. 1　Process of inducing missing data by MAR and NM.

を割くことができずデータに欠損が生じやすいというモデルを仮定している．

手順 2：上位 4 グループについて，各グループ内の欠損値の量が，上位のグループから順に，総欠損値数（指定した欠損率に達するのに必要な欠損値の個数）の 40%，30%，20%，10%になるまで手順 3〜5 を繰り返す（図 1）．

手順 3：グループ内のプロジェクトから，欠損値を与えるプロジェクトをランダムで選択する．

手順 4：手順 3 で選択したプロジェクトで欠損していない特性値の中から，一つをランダムで選択する．ただし，目的変数と依存変数は選択対象としない．

手順 5：手順 3 で選択したプロジェクトの，手順 4 で選択した特性値を欠損させる．

NM

手順 1：目的変数以外の全変数の中から一つを選択する．

手順 2：選択された変数の値が小さい順にプロジェクトをソートし，ソートしたプロジェクトを先頭から五つのグループに均等に分ける（図 1）．

手順 3：上位 4 グループについて，各グループ内の当該特性値についての欠損値の量が，上位のグループから順に，総欠損値数の 40%，30%，20%，10%になるまで手順 4，手順 5 を繰り返す．

手順 4：グループ内のプロジェクトから，欠損値を与えるプロジェクトをランダムに選択する．

手順 5：手順 4 で選択したプロジェクトの当該特性値を欠損させる．

手順 6：全変数について手順 1〜5 により欠損値を与える．

上記手順の MAR 及び NM において，全プロジェクトの 20%に対して総欠損値数の 40%の欠損値を与える必要があるため，欠損率 50%以下でしか欠損値を与えることはできない．

どの変数がどの欠損メカニズムに従うかは，データ計測体制やソフトウェア開発方法論に依存すると考えられるため，一意に決定することは難しい．そこで，本論文では，欠損メカニズムごとにデータセットを作成することとし，各データセットにおいて，目的変数と依存変数以外の全変数が当該メカニズムに従うと仮定した [17]．

3.3　ロバスト性の評価尺度

本論文では，データの欠損率，及び，欠損メカニズムの違い，のそれぞれに関して，CF-based 見積り手法とステップワイズ重回帰分析のロバスト性を比較する．

ある見積り手法が「欠損率に関してロバストである」とは，見積りモデルを作成するためのデータ（以降フィットデータと呼ぶ）の欠損率を増大させた場合に，見積り精度を評価するためのデータ（以降テストデータと呼ぶ）の見積り精度が大きく低下しないことと考える．本論文では，欠損率に関するロバスト性の評価尺度として，フィットデータの欠損率を 0%から 50%へと増大させたときの見積り精度（絶対誤差，相対誤差）の変化率を用いる．変化率が小さいほど，欠損率に関してロバスト性が高いとみなす．欠損率を 50%とした理由は，欠損メカニズムのうち MAR 及び NM では，欠損率 50%以下でしか欠損値を与えることができないためである．

また，ある見積り手法が「欠損メカニズムの違いに関してロバストである」とは，異なる欠損メカニズムを用いて作成されたフィットデータが複数存在する場合に，どのフィットデータを用いてもテストデータの見積り精度に大差がないと考える．本論文では，欠損メカニズムの違いに関するロバスト性の評価尺度として，欠損率 50%のフィットデータを三つの欠損メカニズムをそれぞれ用いて作成し，それらのフィットデータを使って見積りを行ったときの，最良の見積り精度と最低の見積り精度の差を用いる．この差が小さいほど，欠損メカニズムの違いに関してロバスト性が高いとみなす．

4.　評 価 実 験

4.1　基礎データセット

実験で利用した基礎データセットは，プロジェクト

数 140 件，特性値数 10 個の，欠損値を含まないデータセットである．10 個の特性値の名称，及び，各特性値の統計量を表 1 に示す．これらの特性値のうち試験工数を目的変数とし，試験工数以外の特性値を説明変数の候補とした．基礎データセットは，企業のソフトウェア開発で得られたプロジェクトのデータセットから欠損値を含まない部分を取り出したデータセットである．基礎データに含まれるプロジェクトは，新規開発，機能拡張，パッケージのカスタマイズなど様々な開発形態のプロジェクトが含まれており，実施期間が数週間から数年までのプロジェクトが含まれている．また，守秘義務契約により，特性値ごとに何らかの定数であらかじめ除算された値が企業から提供されている．

4.2 実験手順

実験手順は次のとおりである（図 2）．

（1） **4.1** で述べた基礎データセットを 10 回無作為に 2 等分し，見積りモデルを作成するためのデータ（フィットデータ）と見積り精度を評価するためのデータ（テストデータ）の組を 10 組作成した．

（2） 作成した 10 個のフィットデータに対して **3.1** で述べた方法を用いて欠損値を与えた．MCAR で欠損率 0〜90%，MAR と NM で欠損率 0〜50%の範囲で，10%刻みで欠損値を与え，テストデータ 1 個に対して 22 個のフィットデータ，計 220 個のフィットデータを作成した．

（3） CF-based 見積り手法のパラメータである類似プロジェクト数（式 (4) の $k - nearestProjects$）の最適な値を決定した．類似プロジェクト数 k は，予備実験で，最も高い見積り精度が得られた $k = 14$ を採用した．

（4）（3）で決定した類似プロジェクト数を使い，（2）で作成したフィットデータとテストデータの組に対して **2.1** で述べた CF-based 見積り手法により見積りを行い，見積り値の絶対誤差と相対誤差を求めた．

（5）（2）で作成したフィットデータとテストデータの組に対して，**2.2** で述べた MDT を併用したステップワイズ重回帰分析により見積りを行い，見積り値の絶対誤差と相対誤差を求めた．3 種類の MDT の結果のうち，最も見積り誤差が小さい結果をステップワイズ重回帰分析の結果として採用した．採用した結

表 1　実験データセットの各特性値の統計量
Table 1　Statistics of each metrics in the experiment dataset.

	設計工数（総計）	設計工数（正社員）	設計工数（派遣）	設計工数（外注）
平均値	1.15	0.13	0.7	0.01
中央値	0.09	0.02	0.04	0
分散	47.5	0.61	25.8	0.002
最大値	67.5	8.72	58.7	0.42
最小値	0.001	0	0	0
	製造工数（総計）	製造工数（正社員）	製造工数（派遣）	製造工数（外注）
平均値	2.06	0.03	0.12	1.51
中央値	0.13	0.003	0.03	0
分散	175	0.01	0.07	152
最大値	144	0.98	1.59	144
最小値	0.001	0	0	0
	開発規模	試験工数		
平均値	233	1.45		
中央値	43.9	0.07		
分散	1×10^6	101		
最大値	1×10^4	112		
最小値	0.4	0.001		

値はあらかじめ特性値ごとに何らかの定数で除算されている

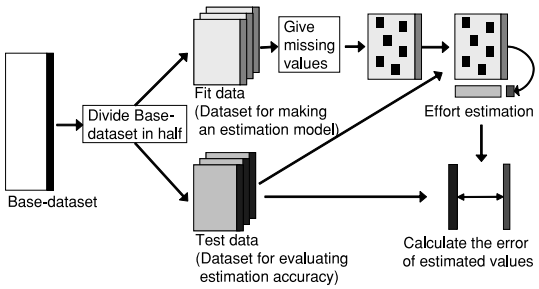

図 2　実験におけるデータ加工
Fig. 2　Data handling process in the experiment.

表 2　変数選択率
Table 2　Ratio of variable selection.

特性値		設計工数（総計）	設計工数（正社員）	設計工数（派遣）	設計工数（外注）	製造工数（総計）	製造工数（正社員）	製造工数（派遣）	製造工数（外注）	開発規模
絶対誤差	MCAR	7.00%	13.30%	12.30%	7.00%	7.90%	0%	52.40%	0%	5.00%
	MAR	49.50%	42.00%	41.90%	0%	62.10%	35.70%	45.30%	36.30%	1.50%
	NM	48.70%	38.20%	39.70%	0%	38.30%	23.50%	43.00%	23.20%	39.30%
相対誤差	MCAR	6.50%	12.80%	11.50%	6.70%	7.90%	0%	50.40%	0%	5.00%
	MAR	49.50%	42.00%	41.90%	0%	62.10%	35.70%	45.30%	36.30%	1.50%
	NM	48.70%	38.20%	39.70%	0%	38.30%	23.50%	43.00%	23.20%	39.30%

果において，ステップワイズ変数選択で各特性値が選択された割合を表 2 に示す．

（6）（4），（5）で得られた結果を比較し，CF-based 見積り手法のロバスト性を確認した．

5. 結果と考察

5.1 欠損率に関するロバスト性

工数見積り結果として，データ欠損率が 0%と 50%のときの絶対誤差，相対誤差の中央値を表 3 に示す．また，表 3 の右端の欄には，欠損率が 0%から 50%へと増大したときのそれら誤差の中央値の変化率を示す．ここで誤差の評価に平均値でなく中央値を用いた理由は，誤差の大きな少数のプロジェクトが存在し，誤差の分布が正規分布となっていなかったためである．データ欠損のない場合の見積り精度の評価は本論文の対象外ではあるが，表 3 に示されるように，データ欠損率 0%では，CF-based 手法の方がステップワイズ重回帰分析よりも絶対誤差，相対誤差ともに小さかった．

表 3 において，データ欠損率を 0%から 50%へ増大させたときの絶対誤差の変化率に着目すると，CF-based 手法では変化率が 7.8〜28.1%であるのに対し，ステップワイズ重回帰分析では 107.5〜905.7%となっており，14〜48 倍大きな値を取っている．このことから，見積り値の絶対誤差について，CF-based 手法は欠損率の変化に関してよりロバストであるといえる．

相対誤差についても，同様に，データ欠損率を 50%へ増大させたときの誤差の変化率に着目すると，CF-based 手法では変化率が 1.0〜6.8%であるのに対し，ステップワイズ重回帰分析では 43.0%〜743.8%となっており，33〜109 倍大きな値をとっている．このことから，相対誤差についても，CF-based 手法は欠損率の変化に関してよりロバストであるといえる．

表 3 見積り誤差の中央値
Table 3 Median of estimation error.

欠損率			0%	50%	変化率 (%)
絶対誤差	CF-based 手法	MCAR	0.064	0.076	18.8
		MAR		0.059	7.8
		NM		0.048	28.1
	SW 重回帰分析	MCAR	0.106	1.066	905.7
		MAR		0.220	107.5
		NM		0.510	381.1
相対誤差	CF-based 手法	MCAR	0.770	0.718	6.8
		MAR		0.762	1.0
		NM		0.757	5.1
	SW 重回帰分析	MCAR	1.383	11.67	743.8
		MAR		1.977	43.0
		NM		3.731	169.8

それぞれの欠損メカニズムについて，欠損率を 10%刻みで変化させたときの誤差の分布を示す箱ひげ図を図 3〜図 5 に示す．グラフの縦軸は見積り誤差を，横軸は欠損率を示し，箱の下端は第 1 四分位，上端は第 3 四分位，箱中の横線は中央値，線分（ひげ）の下端の横線は最小値を表す．グラフの上方を省略しているため最大値は示されていない．

図 3〜図 5 において，ステップワイズ重回帰分析の結果は，MCAR では欠損率 30%以上で見積り精度が低下しており，MAR では欠損率 40%以上で，NM では欠損率 50%以上で見積り精度が低下している．一方，CF-based 見積り手法では，欠損率 60%以下では見積り精度は大きく低下していない．ステップワイズ重回帰分析と比較すると，高い精度で見積り可能な欠損率の範囲が大きく，許容される欠損率の範囲という観点からも，ロバスト性は高いといえる．ただし，

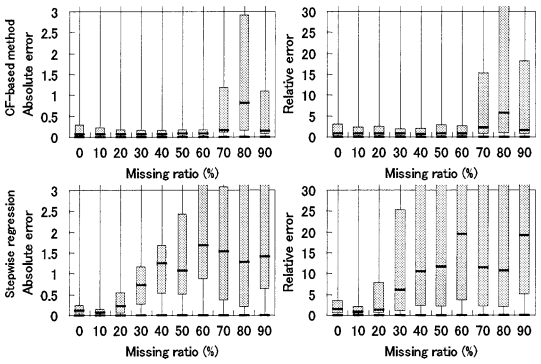

図 3 各欠損率における見積り精度（MCAR）
Fig. 3 Estimation accuracy in each missing ratio (MCAR).

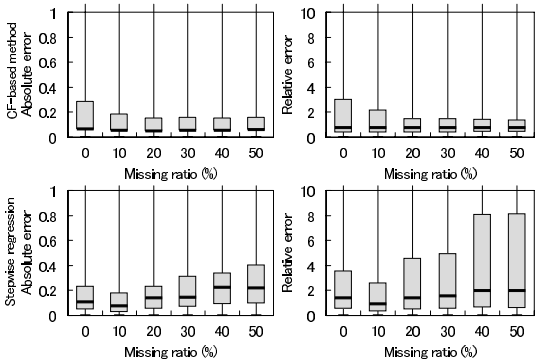

図 4 各欠損率における見積り精度（MAR）
Fig. 4 Estimation accuracy in each missing ratio (MAR).

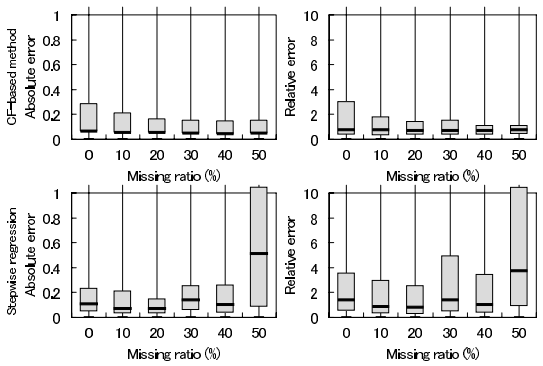

図 5　各欠損率における見積り精度（NM）
Fig. 5　Estimation accuracy in each missing ratio (NM).

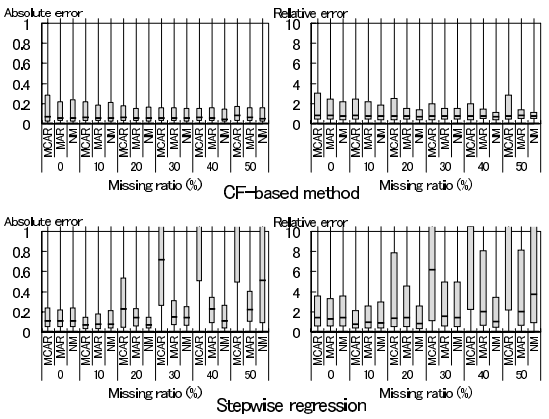

図 6　各欠損メカニズムの見積り精度
Fig. 6　Estimation accuracy in each missing mechanism.

CF-based 見積り手法も，欠損率が 70%以上では見積り精度が大きく低下しており，見積りを行うべきではないといえる．

また，図 3〜図 5 の箱の大きさ（第 1 四分位と第 3 四分位の差）に着目すると，CF-based 見積り手法の方が小さく，ばらつきが小さいといえる．ステップワイズ重回帰分析は，特に MCAR において，欠損率の増大に伴って誤差のばらつきが著しく増大した．

5.2　欠損メカニズムの違いに関するロバスト性

表 3 において，欠損率 50%の絶対誤差に着目すると，CF-based 見積り手法では，絶対誤差中央値の最大が MCAR の 0.076，最小が NM の 0.048 であり，その差は 0.028 であったのに対し，ステップワイズ重回帰分析では最大が MCAR の 1.066，最小が MCAR の 0.220 であり，その差は 0.846 となり，これは CF-based 見積り手法と比べて約 31 倍大きい．このことから，見積り値の絶対誤差について，CF-based 手法は欠損メカニズムの違いに関してよりロバストであるといえる．

相対誤差についても，同様に，CF-based 見積り手法では，最大が MAR の 0.762，最小が MCAR の 0.718 であり，その差は 0.044 であったのに対し，ステップワイズ重回帰分析では最大が MCAR の 11.67，最小が MAR の 1.977 であり，その差は 9.7 となり，これは CF-based 見積り手法と比べて約 220 倍大きい．このことから，相対誤差についても，CF-based 手法は欠損メカニズムの違いに関してよりロバストであるといえる．

また，欠損メカニズム，欠損率，及び，見積り誤差の関係を一つの図で表したものを図 6 に示す．グラフの縦軸は見積り誤差を，横軸は欠損メカニズムと欠損率を示す．図 6 の箱ひげ図の中央値に着目すると，CF-based 見積り手法は欠損値の大小にかかわらず，欠損メカニズムの違いによる誤差のばらつきは小さい．一方，ステップワイズ重回帰分析では，欠損率が 20%を超えると欠損メカニズムの違いによる誤差のばらつきが大きくなり，特に，欠損値が均一に生じている MCAR と均一に生じていない MAR，NM の間で差が大きくなった．このことからも，CF-based 見積り手法は，より広い欠損率の範囲において，欠損メカニズムの違いに関してよりロバストであるといえる．

6. むすび

本論文では，協調フィルタリングに基づく工数見積り手法（CF-based 見積り手法）のデータの欠損に対するロバスト性を評価した．評価においては，欠損率に関するロバスト性と欠損メカニズム（MCAR，MAR，NM）の違いに関するロバスト性の二つの観点から評価を行った．140 件のプロジェクト，10 種類の特性値の実績値データを用いた評価実験の結果，CF-based 見積り手法は MDT を併用したステップワイズ重回帰分析よりも，欠損率に関しても，欠損メカニズムの違いに関してもロバストであった．

ただし，CF-based 見積り手法でも，欠損率 70%以上では見積り精度が大きく低下した．したがって，欠損率 70%以上では工数見積りを行うべきではないといえる．また，本論文では，全変数が同一の欠損メカニズムに従うと仮定してロバスト性の評価実験を行ったが，実際には，変数ごとに従う欠損メカニズムは異な

ると考えられる．そこで，3種類の欠損メカニズムを組み合わせてデータ欠損させたデータセットによるロバスト性の評価が今後の課題として挙げられる．

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．

文　献

[1] A. Albrecht and J. Gaffney, “Software function, source lines of code, and development effort prediction,” IEEE Trans. Softw. Eng., vol.9, no.6, pp.83–92, 1979.

[2] B.W. Boehm, Software engineering economics, Prentice Hall, New Jersey, 1981.

[3] L. Briand, V. Basili, and C. Hetmanski, “Developing interpretable models with optimized set reduction for identifying high-risk software components,” IEEE Trans. Softw. Eng., vol.19, no.11, pp.1028–1024, 1993.

[4] S.D. Conte, H.E. Dunsmore, and V.Y. Shen, Software engineering metrics and models, The Benjamin/Cummings Publishing Company, Inc., California, 1986.

[5] （独）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，ソフトウェア開発データ白書 2005—IT 企業 1000 プロジェクトの定量データを徹底分析，日経 BP 社，東京，2005.

[6] “ISBSG Estimating, Benchmarking and Research Suite Release 9,” International Software Benchmarking Standards Group, 2004, http://www.isbsg.org/

[7] J. Kromrey and C. Hines, “Nonrandomly missing data in multiple regression: An empirical comparison of common missing-data treatments,” Educational and Psychological Measurement, vol.54, no.3, pp.573–593, 1994.

[8] R.J.A. Little and D.B. Rubin, Statistical Analysis with Missing Data, 2nd ed., John Wiley & Sons, New York, 2002.

[9] E. Mendes, I. Watson, C. Triggs, N. Mosley, and S. Counsell, “A comparative study of cost estimation models for web hypermedia applications,” Empir. Softw. Eng., vol.8, no.2, pp.163–196, 2003.

[10] I. Myrtveit, E. Stensrud, and U.H. Olsson, “Analyzing data sets with missing data: An empirical evaluation of imputation methods and likelihood-based methods,” IEEE Trans. Softw. Eng., vol.27, no.11, pp.999–1013, 2001.

[11] “Naist collaborative filtering engines,” http://sourceforge.jp/projects/ncfe/

[12] N. Ohsugi, M. Tsunoda, A. Monden, and K. Matsumoto, “Applying collaborative filtering for effort estimation with process metrics,” 5th Int’l Conf. on Product Focused Software Process Improvement (Profes2004), Lecture Notes in Computer Science, vol.3009, pp.274–286, Kyoto, Japan, 2004.

[13] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, “GroupLens: An open architecture for collaborative filtering of netnews,” Proc. ACM Conf. Computer Supported Cooperative Work (CSCW’94), pp.175–186, Chapel Hill, North Carolina, United States, 1994.

[14] B.M. Sarwar, G. Karypis, J.A. Konstan, and J. Riedl, “Item-based collaborative filtering recommendation algorithms,” Proc. 10th International World Wide Web Conference (WWW10), pp.285–295, Hong Kong, 2001.

[15] M. Shepperd and C. Schofield, “Estimating software project effort using analogies,” IEEE Trans. Softw. Eng., vol.23, no.12, pp.736–743, 1997.

[16] K. Srinivasan and D. Fisher, “Machine learning approaches to estimating software development effort,” IEEE Trans. Softw. Eng., vol.21, no.2, pp.126–137, 1995.

[17] K. Strike, K. El Eman, and N. Madhavji, “Software cost estimation with incomplete data,” IEEE Trans. Softw. Eng., vol.27, no.10, pp.890–908, 2001.

[18] 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，” 情処学論，vol.46, no.5, pp.1156–1164, 2005.

（平成 18 年 1 月 4 日受付，5 月 16 日再受付）

柿元　健　（学生員）

平 15 神戸市立高専専攻科・電気電子了．平 17 奈良先端科学技術大学院大学情報科学研究科博士前期課程了．現在，同大学院博士後期課程在学中．ソフトウェア信頼性/開発工数予測などの研究に従事．情報処理学会，IEEE 各会員．

角田　雅照　（学生員）

平 9 和歌山大・経済卒．平 16 奈良先端科学技術大学院大・情報科学研究科博士前期課程了．現在，同大博士後期課程在学中．ソフトウェアプロジェクトのデータ分析，ソフトウェア開発工数見積りなどの研究に従事．IEEE 会員．

大杉　直樹　（正員）

平 13 奈良高専専攻科・電子情報了．平 16 奈良先端科学技術大学院大学情報科学研究科博士後期課程了．同年同大学研究員．博士（工学）．エンピリカルソフトウェア工学，プロジェクトデータ収集/分析，及び，開発工数/品質/リスク見積りの研究などに従事．情報処理学会，IEEE 各会員．

門田　暁人　（正員）

平 6 名大・工・電気卒．平 10 奈良先端科学技術大学院大学情報科学研究科博士後期課程了．同年同大・情報科学・助手．平 16 同大助教授．平 15〜16 オークランド大客員研究員．博士（工学）．定量的ソフトウェア開発支援，ソフトウェアセキュリティなどの研究に従事．情報処理学会，日本ソフトウェア科学会，教育システム情報学会，IEEE，ACM 各会員．

松本　健一　（正員）

昭 60 阪大・基礎工・情報卒．平元同大大学院博士課程中退．同年同大・基礎工・情報工学科・助手．平 5 奈良先端科学技術大学院大学情報科学研究科・助教授．平 13 同大教授．工博．エンピリカルソフトウェア工学，特に，プロジェクトデータ収集/利用支援の研究に従事．情報処理学会，ACM 各会員，IEEE Senior Member．