データ欠損を考慮したプロジェクトコスト超過予測

本村拓也* 瀧進也* 星幸雄** 米光哲哉** 福地豊** 松本健一*

Predicting Cost Overrun Projects with Missing Data

Takuya Motomura* Shinya Taki* Yukio Hoshi** Tetsuya Yonemitsu**Yutaka Fukuchi** Ken-ichi Matsumoto*

要旨　本研究では，データ欠損を考慮したプロジェクトのコスト超過予測方法を提案する．本研究では，コスト超過を予測するためにリスク項目の評価値を用いる．しかし，リスク項目には，様々な理由により未記録の値（欠損値）が多数含まれる．従来の手法である線形判別分析を用いる場合，多数の欠損値は予測の正確さ（精度）を低下させる原因となる．そこで本研究では，欠損値を含むデータを用いても高い精度で予測できる協調フィルタリングを応用し，コスト超過の有無を予測する方法を提案する．評価実験の結果，従来手法である線形判別分析よりも，提案手法の予測精度が高いことが確認できた．

キーワード　リスク管理，欠損値，コスト超過，判別分析

Abstract　In this paper, we propose a method of predicting cost overrun projects with missing data. We use evaluated values of risk factors to predict the cost overrun. However, the risk factors often include missing values for various reasons. The missing values affect the accuracy of the prediction when using linear discriminant analysis. In this paper, we propose a method of predicting cost overrun projects using collaborative filtering, which is robust against a missing value. As a result of our experiment, the prediction accuracy of the proposed method is higher than linear discriminant analysis.

Keyword　risk management, missing value, cost overrun, discriminant analysis

1.　まえがき

ソフトウェア開発プロジェクトにおいては，開発コストが予定よりも超過すること(コスト超過)がしばしば発生する[9]．コスト超過を避けるためには，開発プロジェクトの初期にコスト超過を正確に予測することが重要となる[5]．予測が正確であれば，人員を再配置するなどの対策を効率よく施すことができる．また，対策を施す時期が早期であれば，対策の選択肢が多いだけでなく，対策の効果も大きくなる[6]．

本研究ではリスク項目の評価値を用いて，コスト超過を予測する．リスク項目の評価値とは，プロジェクトマネージャーが，リスクを管理するために，開発コストに影響を与える項目(要因)を評価したものであり，数段階の数値で与えられる．過去のプロジェクトのリスク項目の評価値と，コストが超過したか否かのデータを用いて，現在のプロジェクトを予測する．

ただし，過去のデータには数値が未記録（欠損値）であることが多いため，一般的な予測方法である線形判別分析を用いてコスト超過を予測すると，予測の正確さ（精度）が著しく低下する可能性がある．例えば，リスク項目を追加した場合，過去のプロジェクトではそれらの評価値が欠損値になってしまう．また，リスク項目を評価する担当者が一部の項目を評価し忘れた場合にも欠損値が発生する．

そこで本研究では，協調フィルタリング（以降 CF と記す）[3]を用いて，開発プロジェクトのコスト超過を予測することを提案する. CF は過去のプロジェクトのデータに欠損値が多く含まれる場

*奈良先端科学技術大学院大学（Nara Institute of Science and Technology）
**日立製作所（Hitachi, Ltd.）

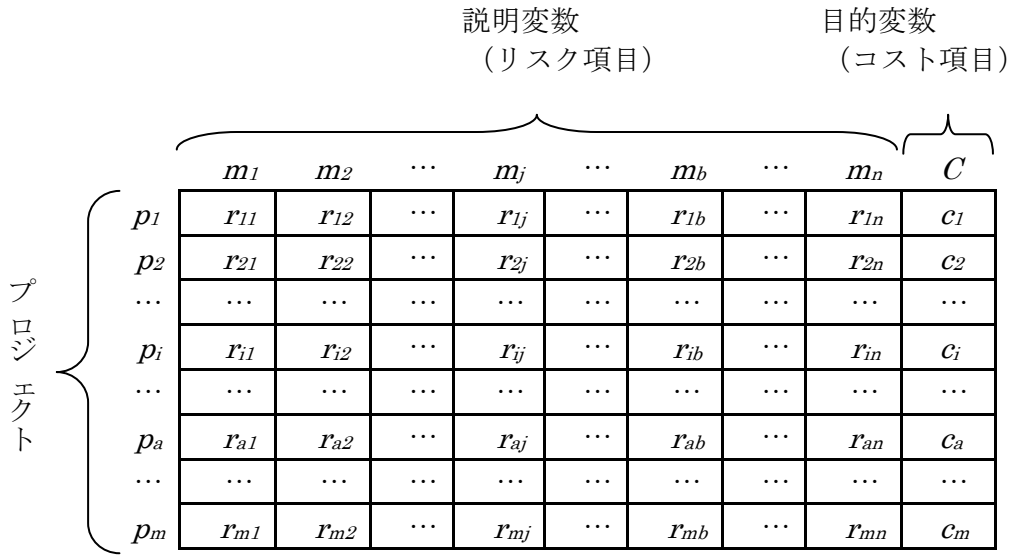

図 1. 予測に用いる行列

合でも，比較的高い精度で予測できるという特徴がある[12]. CF をコスト超過予測に適用することにより，リスク項目に多くの欠損値が含まれている場合でも, 高い精度で予測できると考えられる.

本研究では 2 種類の方法で予測を行った．1 つめの方法では，まず予測対象（実施中）プロジェクトと，過去に行われた各プロジェクトとの間の類似度を，リスク項目の類似性に基づいて計算する．次に，類似するプロジェクトから，予測対象プロジェクトのコストが超過するか否かを予測する．2 つめの方法では，まずコストが超過するか否かと，各リスク項目との間の類似度を，相関係数に基づいて計算する．次に，類似するリスク項目の，予測対象プロジェクトにおける値に基づいて，コストが超過するか否かを予測する．いずれの方法でも，欠損値の部分は計算に使わずに類似度計算を行う.

本研究の新規性は，コスト超過の予測に CF を適用していることと, CF を判別分析に適用していることである．また，有用性は，失敗するプロジェクト同士は似ているという類推が可能であることを明らかにしたことと，失敗するプロジェクトの判別予測に CF が適用可能であることを示したことである.

以降，2 章で CF の予測手順を説明する．3 章で提案方法の予測手順について述べる．4 章で提案方法の有効性を評価するためのケーススタディについて報告し，5 章でケーススタディの結果と考察を述べる. 6 章で関連研究を紹介し, 7 章でまとめと今後の課題について述べる.

2. 協調フィルタリング

協調フィルタリング（CF）は，非常に多くのアイテム（書籍，映画，音楽など）の中からユーザの好みに合うと思われるアイテムを予測し，推薦するシステムに用いられてきた. CF を用いたシステムは，推薦手順に応じてユーザベース手法とアイテムベース手法の 2 つに分類できる.

ユーザベース手法を用いたものとして，Resnick ら[7]のシステムがあげられる．ユーザベース手法においては，次の手順で推薦が行われる.

us1.　データの収集
各ユーザが, 使用したことがある各アイテムに対して，好みの度合を数段階の数値（例えば，5 段階評価）で評価する.

us2.　類似度計算
収集したデータに含まれる推薦対象のユーザ（対象ユーザ）とその他のユーザとの間の類似度を算出し, 評価の傾向が対象ユーザと似ているユーザ（類似ユーザ）を探し出す.

us3.　予測値計算
対象ユーザがまだ評価していないアイテム全てに対する評価を, 類似ユーザが与えた評価に基づいて予測する.

us4.　推薦作業
対象ユーザが高い評価を与えるであろうと予測されたアイテムを推薦する.

アイテムベース手法は Sarwar ら[8]によって提案された手法であり, 次の手順で推薦が行われる.

is1.　データの収集
us1 と同様の処理を行う.

is2.　類似度計算

推薦対象ユーザが未評価のアイテム(対象アイテム)とその他のアイテムとの間の類似度を算出し,評価の傾向が対象アイテムと似ているアイテム（類似アイテム）を探し出す.

is3.　予測値計算

対象ユーザがまだ評価していないアイテム全てに対する評価を,対象ユーザが類似アイテムに与えた評価値に基づいて予測する.

is4.　推薦作業

us4 と同様の処理を行う.

3.　提案方法

提案方法では，図 1 に示すような，m 行 n 列の表を用いて予測する．図 1 で，$p_i \in \{p_1, p_2, \ldots, p_m\}$ は i 番目の開発プロジェクトを表し，$m_j \in \{m_1, m_2, \ldots, m_n\}$ は j 番目のリスク項目を表す．表中の各セル $r_{ij} \in \{r_{11}, r_{12}, \ldots, r_{mn}\}$ は，プロジェクト p_i のリスク項目 m_j に対する評価値を表す．評価値 r_{ij} が未評価の場合，そのセルは欠損値となる．さらに，$c_i \in \{c_1, c_2, \ldots, c_m\}$ はプロジェクト p_i のコストが超過したか否かを表す評価値（コスト評価値）を表す．プロジェクト p_i の実施中に許容できない程度に大きくコストが超過した場合，c_i は 1 とし，超過しなかった場合は 0 とする．コスト評価値 c_i は C（コスト項目）に含まれているとする．以降，提案する各手法による予測手順について説明する．

ユーザベース手法による予測手順

1. 予測対象プロジェクト p_a とその他のプロジェクト p_i との間の類似度 $sim(p_a,p_i)$ を次式で計算する．

$$sim(p_a, p_i) = \frac{\sum_{j \in M_a \cap M_i} \left(w_j r_{aj} - \overline{m_j}\right)\left(w_j r_{ij} - \overline{m_j}\right)}{\sqrt{\sum_{j \in M_a \cap M_i} \left(w_j r_{aj} - \overline{m_j}\right)} \sqrt{\sum_{j \in M_a \cap M_i} \left(w_j r_{ij} - \overline{m_j}\right)}} \quad (1)$$

ただし，M_a と M_i はそれぞれ，プロジェクト p_a と p_i で評価されたリスク項目の集合を現す．$\overline{m_j}$ は，リスク項目 m_j に対する評価値の中央値を表す．w_j はリスク項目 m_j に対する重みであり，次式で計算する．

$$w_j = \frac{\sum_{i \in P_a \cap P_j} \left(r_{jb} \times r_{ij}\right)}{\sqrt{\sum_{i \in P_a \cap P_j} \left(r_{jb}\right)^2} \sqrt{\sum_{i \in P_a \cap P_j} \left(r_{ij}\right)^2}} \quad (2)$$

ただし，P_a と P_j は各々，リスク項目 m_a と m_j を評価したプロジェクトの集合を表す.

2. 予測対象プロジェクト p_a に対するコスト評価値 c_a の予測値 $\hat{c}_a$ は，式 1 によって計算した類似度を用い，次式(3)で計算する．計算結果に対して閾値を設定し，予測値 $\hat{c}_a$ が閾値より大きければ 1（コストが超過する），小さければ 0（コストが超過しない）と予測する．

$$\hat{c}_a = \frac{\sum_{j \in k-nearest\ \Pr ojects} c_j \times sim\left(p_a, p_i\right)}{\sum_{j \in k-nearest\ \Pr ojects} sim\left(p_a, p_i\right)} \quad (3)$$

ただし，*k-nearestProjects* は，予測対象プロジェクト p_a との類似度が高い上位 k 件の類似プロジェクトの集合を表す.類似プロジェクト数 k の値は予測精度に影響を与える．4 章で説明するケーススタディでは,最も精度が高くなる k の値を採用した.

アイテムベース手法による予測手順

1. コスト項目 C と各リスク項目 m_j との間の類似度を次式で計算する．

$$sim(C, m_j) = \frac{\sum_{i \in P_b \cap P_j} \left(r_{ib} - \overline{m_i}\right) \times \left(r_{ij} - \overline{C}\right)}{\sqrt{\sum_{i \in P_a \cap P_j} \left(r_{ib} - \overline{m_i}\right)^2} \sqrt{\sum_{i \in P_b \cap P_j} \left(r_{ij} - \overline{C}\right)^2}} \quad (4)$$

ただし，P_a と P_j は各々，リスク項目 m_a と m_j を評価したプロジェクトの集合を表す．また $\overline{m_i}$ は，リスク項目 m_i に対する評価の平均値を表す.

2. 予測対象プロジェクト p_a に対するコスト評価値 c_a の予測値 $\hat{c}_a$ は，式(4)によって計算した類似度を用いて次式で計算する.計算結果に対して閾値を設定し，予測値 $\hat{c}_a$ が閾値より大きければ 1（コストが超過する），小さければ 0（コストが超過しない）と予測する．

$$\hat{c}_a = \frac{\sum_{j \in k-nearestRisks} \left(m_{ij} - \overline{p_i}\right) \times sim(C, m_j)}{\sum_{j \in k-nearestRisks} sim(C, m_j)} + \overline{p_a} \quad (5)$$

ここで，$\overline{p_i}$ はプロジェクト p_i に対する評価の

表 1. 各フェーズの項目数と欠損率

	基本	見積フェーズ	受注フェーズ	計画フェーズ
項目数	37	144	293	544
欠損率	0.00	0.41	0.40	0.51

平均値を表す．また，*k-nearestRisks* は，コスト項目 C と類似度が高い上位 k 件のリスク項目の集合を表す. 4 章で説明するケーススタディでは, 最も精度が高くなる k の値を採用した.

4. ケーススタディ

4.1. 目的とアプローチ

ケーススタディの目的は，多くの欠損値を含むデータに対する提案方法の有効性を評価することである．そのためのアプローチとして，実際のソフトウェア開発プロジェクトに対するリスク項目の評価データを用い，提案方法と代表的な従来手法（線形判別分析）の予測精度を比較した．ただし，線形判別分析に関してはデータに欠損値が含まれる場合に予測ができないため，欠損値処理法（平均値挿入法[4]）を適用した．精度の評価指標は適合率，再現率，F1 値を用いた.

4.2. 実験データ

実験データとして，日本のあるソフトウェア開発企業で実施されたソフトウェア開発プロジェクトにおけるリスク項目の評価値を用いた．実験データに含まれるプロジェクトは 46 件であった. 十分訓練を積んだプロジェクトマネージャーにより, 各プロジェクトの各リスク項目が 0（コスト超過に全く影響を与えない）から 3（コスト超過に大きな影響を与える）の 4 段階で評価されている. リスク項目には，基幹業務のシステムかどうか，高信頼性を要求されるシステムかどうか，定量的な規模見積もりを行っているかどうか，といったものが含まれる．また，リスク項目以外に，プロジェクトの特性を現す基本項目が含まれている. 基本項目はプロジェクトの開始時に評価される. さらに，コスト項目 C が 0（コストが超過しなかった），あるいは，1（コストが超過した）で評価されている.

プロジェクトのフェーズが進むと，新たなリスク項目が評価されるとともに，以前のフェーズで評価されたリスク項目が再評価される(ただし, 基本項目は再評価されない). フェーズが進むほど, 評価されたリスク項目が増加する．実験では，見積フェーズ，受注フェーズおよび計画フェーズのそれぞれのフェーズの時点でコスト超過を予測することを想定し，3 種類のデータセットを用意した.

表 1 に, 各フェーズが完了した時点での項目数と, 欠損率を示す. 実験データ全体の欠損率は 56%であった．基本項目には欠損値が存在しなかったが，その他の分類に含まれるリスク項目の欠損率の平均値は約 30%から 60%であった．プロジェクトが実施された時期や部署が様々であったため，欠損率やその分布の偏りが大きくなったと考えられる.

4.3. 評価基準

予測精度の評価基準として適合率，再現率，F1 値を用いた．これらの評価基準は, CF の精度評価基準として広く用いられている[1]. 適合率と再現率を求めるために，次の 2 つの集合を定義しておく.

A：コスト評価値の予測値 $\hat{c}_a$ が 1 と計算されたプロジェクトの集合.

R：実際のコスト評価値 c_a が 1 だったプロジェクトの集合.

適合率

集合 ***A*** に含まれる要素のうち, 集合 ***R*** に含まれる要素と一致した割合．すなわち，予測値が 1 のときに実測値も 1 だった割合．予測結果の正確性を表す．適合率（precision）は次式で計算する.

$$precision = \frac{|A \cap R|}{|R|} \quad (7)$$

再現率

集合 ***R*** に含まれる要素のうち, 集合 ***A*** に含まれる要素と一致した割合．すなわち，実測値が 1 のときに予測値も 1 であった割合．予測結果の網羅性を表す．再現率（recall）は次式で計算する.

$$recall = \frac{|A \cap R|}{|A|} \quad (8)$$

表 2. ケース毎の各評価値の値

	見積フェーズ			受注フェーズ			計画フェーズ		
	適合率	再現率	F1 値	適合率	再現率	F1 値	適合率	再現率	F1 値
ユーザベース CF	59.01	55.61	53.87	61.9	71.02	65.84	69.15	64.21	64.29
アイテムベース CF	54.9	49.46	49.14	59.37	58.22	59.46	60.09	63.41	60.5
線形判別分析	54.32	57.56	55.72	57.71	59.63	57.54	60.4	54.31	56.02

F1 値

F1 値は, 適合率と再現率を 1 つの値で表わす指標である．F1 値を用いることにより，適合率と再現率それぞれのバランス（予測結果の網羅性と正確性）を考慮した評価を行うことができる．F1 値は次式で計算する．

$$F1 = \frac{(1+b^2)\times precision\times recall}{b\ \times(precision+recall)} \quad (9)$$

ただし，式 9 中の b は適合率，再現率のどちらを重視するかを指定するためのパラメータである．本ケーススタディでは b=1 とし，同等に扱った．

4.4. 実験手順

ユーザベース CF，アイテムベース CF，線形判別分析[11]，それぞれを用いて予測を行った．予測精度の評価は以下の手順で行った．

1. 実験データを無作為に 2 等分し，一方を過去のプロジェクトとして扱うフィットデータ，もう一方を実施中(予測対象)のプロジェクトとして扱うテストデータとした．
2. フィットデータを用いて，テストデータのコスト評価値の予測値 $\hat{c}_a$ を算出した．
3. 算出した予測値 $\hat{c}_a$ を，0 と 1 に二値化し，各評価基準を算出した(CF の閾値は 0.5 とした).
4. 1 から 4 の手順を，見積フェーズ，受注フェーズ, 計画フェーズのデータごとに 10 回繰り返した．
5. 各データの 10 回分の各評価基準の平均値を計算し，最終的な結果とした．

5. 結果と考察

ケーススタディの結果，得られた各評価基準の値を表 2 に示す．各行は予測方法を，各列は想定したフェーズを表している. いずれの評価基準も，値が大きいほど予測精度が高いことを示す．

見積フェーズを除いて，ユーザベース CF の F1 値は線形判別分析よりも高かった. この結果から，ユーザベース CF による予測手法の有効性が最も高いといえる．これは，CF による予測が欠損値の影響を受けにくいのに対し，線形判別分析では欠損値の影響で予測モデルの作成が適切に行われなかったためであると考えられる．また，見積フェーズ完了時点での精度が低かったことに関しては，この時点でのリスク項目数が，類似度を算出するには十分でなかったことが考えられる．

アイテムベース CF による予測の精度は，ユーザベース CF による予測と比較して全体的に低い．実験データにおいて，コスト項目と高い相関をもつリスク項目は極めて少ないため，多数のリスク項目がコスト超過に対し，複雑に影響を与えていると考えられる．

フェーズ別に F1 値を観察すると，CF による予測では，見積フェーズと比べて，受注フェーズの予測の方が比較的精度が高くなっている．また，計画フェーズの予測精度は，受注フェーズとほぼ同程度となっている．このことから，受注フェーズ中に，コスト超過が起こるプロジェクトが持つ性質を決定する要因が多く含まれていると考えられる．

6. 関連研究

リスク項目から，プロジェクトのコスト超過を予測する研究がいくつか行われている．Takagi ら[10]は，過去に行われた開発プロジェクトを対象にアンケートを行い, 収集したリスク評価値から，ロジスティック回帰モデルを用いて，失敗する危険性が高いプロジェクトを予測するためのモデルを作成し, その精度を評価, 報告している. また，Jorgensen ら[2]は，リスク項目に基づき，見積工数と予測工数の相対誤差を予測する回帰モデルを提案している．これらの研究では，事前に準備された，コスト超過予測用のアンケートを用いて予測している．本研究は，コスト超過予測に特化していないリスク管理用のデータを用いても，コスト超過を予測することが可能であることを明らかに

した．

7. むすび

本研究では，協調フィルタリングによるソフトウェア開発プロジェクトのコスト超過を予測することを提案し，その精度を評価した．特に，予測精度が低下する原因となる欠損値の問題に注目し，提案方法による予測と，従来方法である線形判別分析による予測について，それぞれの精度を評価した．その結果，ユーザベース CF による予測の有効性を確認できた．

今後の課題として, CF における類似度計算の妥当性を向上させ予測精度を高めるために，あらかじめコストの評価値への影響が低いリスク項目を特定し，これらの項目を除去した上で予測を行うことを考えている．また，予測用データに含まれるプロジェクトの件数を増やし，より多数かつ多様なプロジェクトが含まれるデータセットを用いて，予測実験と評価を行うことも今後の課題である．

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研究成果に基づく．

参考文献

[1] J. Herlocker, J.A. Konstan, L.G. Terveen, and J.T. Riedl, “Evaluating collaborative filtering recommender systems,” ACM Trans. on Information Systems, Vol.22, No.1, pp.5-53, 2004.

[2] M. Jørgensen, “Regression Models of Software Development Effort Estimation Accuracy and Bias,” Empirical Software Engineering, Vol.9, No.4, pp.297-314, 2004.

[3] N. Ohsugi, M. Tsunoda, A. Monden, and K. Matsumoto, “Applying Collaborative Filtering for Effort Estimation with Process Metrics,” Proc. of the 5th Int’l Conf. on Product Focused Soft. Process Improvement, Springer, Berlin Heidelberg, pp.274-286, 2004.

[4] 奥野忠一，久米均，芳賀敏郎，吉澤正，多変量解析法（改訂版），日科技連出版社，東京，2005．

[5] J. D. Procaccion, J. M. Verner, S. P. Overmyer, and M. E. Darter, “Case study: factors for early prediction of software development success,” information and software technology, Vol.44, No.1, p53-62, 2002.

[6] Project Management Institute, A Guide to the Project Management Body of Knowledge 2000 Edition, Project Management Institute, 2000.

[7] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, “GroupLens: An Open Architecture for Collaborative Filtering of Netnews,” Proc. ACM Conf. on Computer Supported Cooperative Work, pp.175-186, Chapel Hill, North Carolina, U.S.A, Oct 1994.

[8] B. M. Sarwar, G. Karypis, J. A. Konstan, and J.T. Riedl, “Item-Based Collaborative Filtering Recommendation Algorithms,” Proc. 10th International World Wide Web Conference, pp.285-295, Hong Kong, May 2001.

[9] The Standish Group International, Inc., “2003 CHAOS Chronicles Report,” 2003.

[10] Y. Takagi, O. Mizuno, and T. Kikuno, “An Empirical Approach to Characterizing Risky Software Projects Based on Logistic Regression Analysis,” Empirical Software Engineering, Vol.10, No.4, pp.495-515, 2005.

[11] 田中豊, 垂水共之, 統計解析ハンドブック 多変量解析, 共立出版, 東京, 1998.

[12] 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.1155-1164, 2005.