

## プロジェクト類似性に基づく 工数見積りに適した変数選択法

瀧 進也<sup>†1,\*1</sup> 戸田 航史<sup>†1</sup> 門田 暁人<sup>†1</sup>  
柿 元 健<sup>†2</sup> 角 田 雅 照<sup>†1</sup>  
大 杉 直 樹<sup>†3</sup> 松 本 健 一<sup>†1</sup>

ソフトウェア開発における多変数の実績データを用いた工数見積りでは、その精度を高めるうえで、見積りに不要な変数を除去することが必須となる。しかし、近年注目されているプロジェクト間の類似性に基づく工数見積り方法 (analogy-based 法) においては、その特性を活かした変数選択法が提案されていない。本論文では、「説明変数 (プロジェクト特性) の値が似ているプロジェクトどうしは工数も似た値をとる」という analogy-based 法の考え方を踏襲し、目的変数である工数との相関係数が低い説明変数を除去することで、説明変数の値が似ていれば目的変数の値も似ているという状態を保つように変数選択する方法 Correlation Threshold based Selection (CTS) を提案する。ソフトウェア開発企業で収集された 3 種類の開発実績データセットを用いた評価実験の結果、CTS は、従来手法であるステップワイス変数選択法と比べて計算量が小さく、特に、カテゴリ変数の数が少ない場合に、安定して精度を改善することができた。

### A Variable Selection Method for Analogy-based Effort Estimation

SHINYA TAKI,<sup>†1,\*1</sup> KOJI TODA,<sup>†1</sup> AKITO MONDEN,<sup>†1</sup>  
TAKESHI KAKIMOTO,<sup>†2</sup> MASATERU TSUNODA,<sup>†1</sup>  
NAOKI OHSUGI<sup>†3</sup> and KEN-ICHI MATSUMOTO<sup>†1</sup>

Analogy-based estimation methods have been studied for deriving an accurate effort estimate required to complete a project; however, few studies have focused on their variable selection methods. Here, variables to be selected are predictor variables (e.g. development size and development duration, etc.) related to an objective variable (effort). In this paper, we propose Correlation Threshold based Selection (CTS) method that exploits the basic concept of analogy-based

estimation where efforts are similar if predictor variables are similar. The proposed method requires few amount of calculation while it can accurately exclude variables that do not correlate with the effort. This paper experimentally evaluated the effectiveness of the proposed method by using three data sets collected from software development companies. The result showed that the proposed method provided high estimation accuracy for projects having larger number of predictor variables.

#### 1. はじめに

ソフトウェア開発プロジェクトの完遂に要する工数 (人月) を見積もる手法として、近年、プロジェクト類似性に基づく工数見積り方法が注目されている<sup>(1),(2)</sup>。この方法では、見積り対象プロジェクトに類似するプロジェクトを実績データの中から探し出し、それら類似プロジェクトの実績工数をもとに工数の見積り値の算出を行う。プロジェクト管理者はしばしば、過去に手がけた類似の案件をもとに、いわゆる類推見積りを行うため、この方法は管理者の直感によく合う<sup>(1)</sup>。また、工数算出の根拠となる類似プロジェクトを参照できるため、管理者に受け入れられやすい。

類似性に基づく工数見積りの課題の 1 つは、変数選択法が確立されていないことである。重回帰分析などのモデルベースの工数見積り手法と同様、類似性に基づく工数見積りにおいても、プロジェクトの特性値 (規模、開発期間など) を説明変数として、目的変数である工数を算出する。その際に、見積りに有用でない説明変数を取り除くための変数選択を行うことが、見積り精度を高めるうえで必須となる。

従来の変数選択方法を類似性に基づく工数見積りに適用する場合には、次に述べる問題がある。一般に、変数選択法は、(1) Filter 手法: 予測アルゴリズムとは独立に、変数間の関係のみに着目して選択する変数を決定する手法、(2) Wrapper 手法: 予測アルゴリズムを実績データのサンプルに繰り返し適用・評価することで、選択する変数を探索的に決定する

†1 奈良先端科学技術大学院大学情報科学研究科

Graduate School of Information Science, Nara Institute of Science and Technology

†2 大阪大学大学院情報科学研究科

Graduate School of Information Science and Technology, Osaka University

†3 株式会社 NTT データ

NTT DATA Corporation

\*1 現在、株式会社日立製作所

Presently with Hitachi, Ltd.

手法, の 2 つに大別される<sup>3)</sup>。(1) の filter 手法としては「相関係数の高い説明変数ペアの片方を削除する」方法や, 2 章で述べる Correlation based Feature Selection<sup>4)</sup> などがあげられる。これらの手法は, 説明変数間の独立性を高めることにより, 多重共線性, すなわち 2 つ以上の説明変数の間に強い相関が発生している状態を回避できる。多重共線性が発生すると, 説明変数それぞれの従属変数への効果が分離しにくくなり, 重回帰式での係数が一意に決定できなくなるという問題が現れる<sup>5)</sup>。しかし, 類似性に基づく工数見積りでは, そもそも重回帰式の導出を行わないため, 多重共線性を避ける必要がない。そのため, filter 手法の狙いである多重共線性を避けるように変数選択を行うことは, 類似性に基づく工数見積りに対してミスマッチである。一方, (2) の wrapper 手法としては, 全探索法, ステップワイズ法, 山登り法などが知られている。これらの手法では, 予測アルゴリズムに応じた統計量 (尤度比検定量, Wald 統計量, AIC (Akaike's Information Criteria) など) を用い, その統計量の改善が最も適したものになるように変数を選択する。ただし, 事前に予測モデルを構築しない類似性に基づく工数見積りでは, これらの統計量が算出できないため, これらの統計量は利用できない。詳しくは 2 章で述べるが, 代案として, 交差検証法を用いて仮の見積りを行い, その精度 (絶対誤差平均など) を統計量の代わりに用いることが可能である。しかし, 仮の見積りを行うために, 大きな計算量が必要となることや, 仮の見積りにおいて予測モデルの構築に用いるデータに対して予測モデルが過剰に適合 (オーバーフィッティング<sup>6)</sup>) し, そのデータ以外のデータを予測した場合に, 高い予測精度が得られない可能性があることが問題となる。

本論文では, プロジェクト類似性に基づく工数見積りの特性を生かした変数選択法として, Correlation Threshold based Selection (CTS) を提案する。CTS では, 目的変数である工数との相関係数が低い (ある閾値以下の) 説明変数を除去することで, 「説明変数の値が似ていれば目的変数の値も似ている」という状態を保つように変数選択を行う。CTS では, 前述の wrapper 手法を用いる場合と同様, 相関係数の (最適な) 閾値の決定に交差検証法を用いるが, 必要となる計算量は小さい。

本論文では, CTS の有効性を確認するために, 無償もしくは有償で著者らが入手・利用可能であり, 多くの論文で使われており, プロジェクト数が多い, ソフトウェア開発企業で収集された 3 種類の実績データ (Desharnais データ<sup>7),8)</sup>, Kitchenham データ<sup>9)</sup>, ISBSG データ<sup>10)</sup>) を用いた評価実験を行う。評価実験では, CTS, ステップワイズ変数選択, および, 変数選択を行わない場合の見積り精度を比較する。

以降, 2 章では変数選択に関する関連研究を紹介し, 3 章では研究の背景として, プロジェ

クト類似性に基づく工数見積りについて説明する。4 章で提案手法, および, 評価実験で用いた従来手法 (ステップワイズ法) について説明する。5 章では評価実験の方法として, 用いたデータ, 精度評価指標, 実験手順を説明する。6 章で実験結果を述べ, 7 章で結果に対する考察を述べる。8 章でまとめと今後の課題を述べる。

## 2. 関連研究

従来, filter 手法は, 類似性に基づく工数見積りにはほとんど用いられておらず, 主にクラス分類問題に対して適用されてきた。1 つの有力なアイデアは, Ghiselli によって提唱された仮説「良い説明変数の集合は, (各個体が属する) クラスと高い相関を持ち, かつ, 互いに相関を持たない」である<sup>11)</sup>。Hall は, このアイデアを応用し, Correlation based Feature Selection (CFS) を提案した<sup>4)</sup>。CFS では, ある説明変数の組合せに対して, 各説明変数と目的変数との間の相関係数が大きいほど, また説明変数間の相関係数が小さいほど大きな値をとる評価指標 (*Merit*) を算出し, *Merit* が最大となった変数の組合せを変数選択結果として採用する。CFS は, 多重共線性を避ける効果が期待できる。ただし, 類似性に基づく工数見積りでは, 回帰分析特有の問題である多重共線性の問題はそもそも生じないため, 説明変数が互いに独立であることを必要としない。そこで, 本論文では, 「説明変数間の相関係数が小さい」という条件を設けず, 各説明変数と目的変数との間のみに着目することにした。

一方, wrapper 手法を類似性に基づく工数見積りに適用した事例は存在する<sup>12)</sup>。1 章でも述べたように, 類似性に基づく工数見積りでは, 重回帰分析などのモデルベース手法における尤度比検定量, Wald 統計量, AIC といった統計量が存在しない。そこで, Kirsopp ら<sup>12)</sup> は, 交差検証法の一種である leave-one-out 法を用いて仮の見積りを行い, 絶対誤差平均が最も低くなる説明変数の組合せを特定した。組合せの探索方法として, ランダム変数選択, 山登り法, 変数増加法の 3 つを実施した結果, これらすべての変数選択法において, 変数選択前よりも予測精度が向上した。しかしながら, 提案手法の適用対象のデータセットが 1 種類に限定されており, 異なる特性を持つデータセットにおいても, これらの変数選択法が有効であるかは不明である。また, wrapper 手法と比べると, 仮の見積りを行うために, 大きな計算量が必要となることが問題である。

## 3. プロジェクト類似性に基づく工数見積り

ソフトウェア開発プロジェクトにおいて, より直感的かつ正確な工数見積りを目指して,

プロジェクト類似性に基づく工数見積りの研究が行われてきた<sup>1),2)</sup>。この見積り方法は、類似するプロジェクト（開発規模や対象業種などの特徴が互いに似たプロジェクト）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いる。たとえば、開発規模に着目すると、開発規模が小さいプロジェクトの工数を見積もる場合には、過去のプロジェクトの中から開発規模が小さいプロジェクトを探索し、その実績工数から見積り対象プロジェクトの工数を見積もる。反対に、開発規模が大きいプロジェクトの工数を見積もる場合には、開発規模が大きいプロジェクトの実績工数から見積りを行う。

プロジェクト類似性に基づく工数見積りでは、プロジェクト間の類似度の算出に先立って、まず、データに含まれる値の値域を  $[0, 1]$  に正規化する。この正規化は、値域のばらつきによる、変数ごとの類似度に対する影響度合いの差を排除するために行われる。 $i$  番目のプロジェクトの項目  $j$  における変数の値を  $v_{i,j}$ 、 $j$  番目の変数の最大値を  $\max(P_j)$ 、最小値を  $\min(P_j)$  としたとき、 $v_{i,j}$  の正規化後の値  $v'_{i,j}$  は式 (1) によって求めるのが一般的である。

$$v'_{i,j} = \frac{v_{i,j} - \min(P_j)}{\max(P_j) - \min(P_j)} \quad (1)$$

次に、見積り対象の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、すでに収集された変数の値から計算する。類似度計算の方法として、特性値の系列をベクトルと見なし、ベクトル間の角度を用いる手法<sup>2),13)</sup> や、ベクトル間のユークリッド距離を用いる手法<sup>1)</sup> などが提案されている。いずれの手法においても、変数の値が似ているほど、より類似したプロジェクトであると見なされるが、欠損値（データに含まれる未記録の値）の取り扱い方に違いがある。前者は欠損値を含んだまま類似度を計算できるのに対し、後者は欠損値を含まない実績データが前提となる。本論文では、変数選択の効果を評価するにあたって、欠損値の存在が結果に影響することを避けるため、欠損値のない実績データ、および、ユークリッド距離を用いる手法を用いた。

プロジェクト  $p_a$  とプロジェクト  $p_i$  の間のユークリッド距離  $dist(p_a, p_i)$  は式 (2) によって求められる。

$$dist(p_a, p_i) = \sqrt{\sum_j (v'_{a,j} - v'_{i,j})^2} \quad (2)$$

最後に、類似度が高いプロジェクトの工数の加重平均により、見積り対象プロジェクトの見積り工数の算出（予測値計算）を行う。この際に、“類似度の高いプロジェクト”として、見積り対象プロジェクトよりも規模の小さい（もしくは大きい）プロジェクトが多数選ばれ

た場合、それらの工数の単純平均をとると、過少見積り（もしくは過大見積り）が生じることになる。そこで、予測値計算時に、プロジェクトの規模を補正する  $amp(p_a, p_i)$  を重みとする加重平均をとることとする<sup>14)</sup>。ここで、 $amp(p_a, p_i)$  を式 (3) のように定義し、 $i$  番目のプロジェクトの実績工数を  $effort_i$ 、 $i$  番目のプロジェクトの正規化後の開発規模を  $FP'_i$  としたとき、見積り対象である  $a$  番目のプロジェクトの工数の見積り値  $effort'_a$  は式 (4) のように定義される。

$$amp(p_a, p_i) = \frac{FP'_a}{FP'_i} \quad (3)$$

$$effort'_a = \frac{\sum_{i \in k\text{-nearestprojects}} (effort_i \times amp(p_a, p_i))}{k} \quad (4)$$

ここで  $k\text{-nearestprojects}$  は、プロジェクト  $p_a$  との類似度が高い  $k$  個のプロジェクトの集合を表す。本論文では  $k$  の値を決定するために、 $k$  を 3 ~ 20 まで変化させた予備実験を事前に行っている。最も高い精度が得られた  $k = 5$  を実験中で用いる値として採用した。

## 4. 提案方法

### 4.1 Correlation Threshold based Selection (CTS)

プロジェクト類似性に基づく工数見積りに適用可能な変数選択法として Correlation Threshold based Selection (CTS) を提案する。この方法は、変数の値が似ているプロジェクトどいうしは工数も似た値をとる、という類似性に基づく工数見積りの考え方を変数選択にも適用した手法であり、目的変数との間の相関係数の絶対値が低い説明変数を除去する。相関係数の算出には工数の予測値計算を必要としないため、計算量を抑えることができる（ただし、除去のための最適な閾値を算出するうえで、予測値計算が必要となる）。CTS による変数選択は、以下の手順で行われる。

1. すべての説明変数に関して、目的変数との間の相関係数を算出する。
2. 閾値を 0.0 に設定する。
3. 現在の閾値よりも相関係数の絶対値が小さい変数をすべて削除し、見積り精度（5.3 節で詳述）を算出する。
4. 閾値が 1.0 未満ならば、現在の閾値にあらかじめ設定しておいた値を加算し、3 に戻る。
5. 見積り精度が最も高かったときの閾値を最適な閾値であると判断し、その閾値より相関係数の絶対値が小さい変数をすべて削除する。

本論文では、閾値を 0.05 刻みで動かして最適な閾値を特定した。そのため、説明変数の個数にかかわらず、見積り精度を 20 回計算するだけで変数選択が完了する。この計算回数  
は変数の個数  $n$  に非依存であるので、計算量は  $O(1)$  である。これに対し、総当りによる変  
数選択は  $O(2^n)$ 、後述する擬似ステップワイズ法は  $O(n^2)$  である。

また、5 章の実験では、CTS の手順の中で見積り精度を評価する指標としては、後述す  
る相対誤差平均<sup>\*1</sup>を用い、相関係数にはピアソンの積率相関係数を用いた（相対誤差平均の  
代わりに、絶対誤差平均や誤差分散などの指標を用いることも可能である）。

#### 4.2 擬似ステップワイズ法

2 章で述べたように、プロジェクト類似性に基づく工数見積りは、回帰係数の偏 F 値や  
AIC などに該当する統計量が存在しないため、従来のステップワイズ法をそのまま用いる  
ことはできない。そこで、Kirsopp ら<sup>12)</sup> が行ったように、leave-one-out 法（過去に実施さ  
れた全プロジェクトの中からプロジェクトを 1 つ抽出して見積り対象の現行プロジェクト  
と見なし、残りを過去プロジェクトと見なして見積もることを過去の全プロジェクトに対  
して繰り返す方法）を用いて、過去に実施された全プロジェクトに対して仮の見積りを行い、  
その精度（絶対誤差平均、および、相対誤差平均）を統計量の代わりに用いて探索を行う。  
本論文では、便宜上、このような探索による変数選択を、擬似的なステップワイズ（擬似ス  
テップワイズ）と呼ぶことにする。

一般に、ステップワイズ法は、変数を追加または削除していく手順の違いによって、さら  
に 4 つの方法（変数増加法、変数増減法、変数減少法、変数減増法）に分類される<sup>15)</sup>。プ  
ロジェクト類似性に基づく工数見積りでは、類似度計算に最小でも 2 変数が必要な場合があ  
る<sup>2),13)</sup> ため、説明変数の数を 1 個から増やしていく変数増加法、変数増減法よりも、最大  
数から減らしていく変数減少法、および、変数減増法の方が汎用性が高いと考えられる。そ  
こで、本論文では、擬似的にステップワイズを行うにあたって、変数減少法、および、変数  
減増法を採用し、それぞれ、便宜上、擬似変数減少法、擬似変数減増法と呼ぶこととする。

擬似変数減少法における変数選択手順は、次のとおりである。

1. 説明変数を 1 つ仮削除する。
2. 削除および仮削除されていない説明変数を用いて、leave-one-out により予測を行い、精  
度評価指標（絶対誤差平均、または、相対誤差平均）を算出する。

\*1 相対誤差は、実工数に比して誤差がどの程度の大きさであるかを評価する（たとえば、5 人月の誤差となった場  
合、100 人月のプロジェクトでは見積り精度が高いと見なし、10 人月のプロジェクトでは見積り精度が低いと  
見なす）指標であり、見積り精度を評価する際に広く用いられている。

3. 1~2 を現在削除されていないすべての説明変数に対して行う。
4. 3 において、仮削除時に精度評価指標の値が最も良くなった説明変数を実際に削除する。  
ただし、精度評価指標が（削除しない場合と比べて）向上しない場合は、削除せずに停  
止する。
5. 1 に戻る。

擬似変数減少法では、変数の数を  $n$  とすると精度評価指標の計算回数は最大で  $n(n+1)/2$   
となる。

一方、擬似変数減増法に基づく変数選択手順は、次のとおりである。

1. 説明変数を 1 つ仮削除する。
2. 削除および仮削除されていない説明変数を用いて、leave-one-out により予測を行い、精  
度評価指標（絶対誤差平均、または、相対誤差平均）を算出する。
3. 1~2 を現在削除されていないすべての説明変数に対して行う。
4. 現在削除されている説明変数を 1 つ仮追加する。
5. 4 で仮追加された説明変数と現在削除されていない説明変数を用いて、leave-one-out に  
より予測を行い、精度評価指標を算出する。
6. 4~5 を現在削除されているすべての説明変数に対して行う。
7. 3 および 6 において精度評価指標の値が最も改善された操作を実際に行う。ただし、精  
度評価指標が（操作を行う前と比べて）向上しない場合は、操作を行わずに停止する。
8. 1 に戻る。

擬似変数減増法では、変数の数を  $n$  とすると精度評価指標の計算回数は最大で  $n^2$  となる。

## 5. 評価実験

### 5.1 目的

CTS、擬似変数減少法、擬似変数減増法の 3 つの変数選択法を用いて変数選択した場合  
のそれぞれの見積り精度を確認し、各変数選択法の性能を比較することを目的として評価実  
験を行った。以降、実験に用いたデータ、用いた精度評価尺度、精度評価のための実験手順  
について説明する。

### 5.2 実験に用いたデータ

本研究では、Desharnais データ<sup>7),8)</sup>、Kitchenham データ<sup>9)</sup>、ISBSG データ<sup>10)</sup> の 3 つの  
プロジェクトデータセットを用いた。これらのデータセットに共通して含まれているデータ  
項目には、目的変数である総工数、規模を表す FP 数などがある。以下では、それぞれの

データセットについて、より詳細に述べる。

Desharnais データは、Desharnais によって収集されたカナダのソフトウェア開発企業における 80 年代のデータである<sup>7),8)</sup>。データには 77 件のプロジェクトについて、プロジェクトマネージャの経験年数やチームの経験年数などの 9 種類の変数が記録されている。また、このデータには欠損値は含まれていない。

Kitchenham データは、Kitchenham らが 2002 年に文献の中で公開したソフトウェア開発企業のデータである<sup>9)</sup>。データには 145 件のプロジェクトについて、9 種類の変数が記録されている。しかし、欠損値や工数見積りに用いるには不適切な説明変数（プロジェクト管理者による工数の見積り値など）が含まれていたため、そのサブセットとしてプロジェクト数 135 件、顧客名（匿名化のために番号に置き換えられている）やプロジェクトタイプ（開発（development）、適応保守（adaptive）、是正保守（corrective）など）を含む変数 5 種類のデータを作成し、実験に用いた。

ISBSG データは、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実績データである<sup>10)</sup>。データには 1989 年から 2004 年までの 3,026 件のプロジェクトについて、それぞれ 99 種類の変数が記録されている。ただし、このデータには数多くの欠損値が含まれているため、欠損を含むプロジェクト、変数を削除し、プロジェクト数 133 件、FP 計測手法（IFPUG, SPR, NESMA など）や開発言語を含む変数 11 種類の欠損値のないサブセットを作成し、実験に用いた。

これら 3 つのデータセットでは、説明変数に量的変数と質的変数が混在しており、工数を見積もる前に、質的変数をダミー変数化（2 値変数化）し、量的変数に変換する必要がある。量的変数とは開発規模や開発期間などの、連続値として表現される変数であり、質的変数とは開発言語などが、どのカテゴリ（C 言語, COBOL, Java など）に属するかという情報が不連続値として表現される変数である。ダミー変数化する際には、変数に含まれる各カテゴリに対してそのカテゴリに属するかどうかを表現する 2 値の変数に変換する。ある質的変数に  $p$  個のカテゴリがあるなら、その質的変数は  $p$  個のダミー変数になる。上記の 3 つのデータに対してこの変換を行った結果、Desharnais データの変数は 16 個、Kitchenham データの変数は 15 個、ISBSG データの変数は 27 個となった。

### 5.3 精度評価指標

評価実験で見積り性能を比較するための精度評価指標として、絶対誤差平均、絶対誤差中央値、絶対誤差分散、相対誤差平均、相対誤差中央値、相対誤差分散、 $PRED(0.25)$  の 7 種類を用いた。以降で各評価指標について説明する。

(1) 絶対誤差平均:  $i$  番目のプロジェクトの絶対誤差  $MAE_i$  (Magnitude of Absolute Error) を、実績工数を  $effort_i$ 、見積り工数を  $effort'_i$  として式 (5) のように定義すると、 $n$  件の見積りを行った場合の絶対誤差平均  $MMAE$  (Mean Magnitude of Absolute Error) は式 (6) で定義される。

$$MAE_i = |effort_i - effort'_i| \quad (5)$$

$$MMAE = \frac{1}{n} \sum_{i=1}^n MAE_i \quad (6)$$

絶対誤差平均の値が小さいほど全体として誤差が小さい、すなわち、見積り精度が高いことを示す。

(2) 絶対誤差中央値: 絶対誤差中央値は絶対誤差  $MAE_i$  の中央値であり、絶対誤差平均と同様に、値が小さいほど見積り精度が高いことを示す。平均値の場合は極端に大きい値が存在するとその影響を強く受けてしまう傾向があるが、中央値はそのような値が存在しても影響を受けないため、平均値とあわせて見ることで見積り精度の高さをより正確に判断することができる。

(3) 絶対誤差分散: 絶対誤差分散  $VMAE$  (Variance Magnitude of Absolute Error) は式 (7) で定義される。

$$VMAE = \frac{1}{n-1} \sum_{i=1}^n (MAE_i - MMAE)^2 \quad (7)$$

(4) 相対誤差平均:  $i$  番目のプロジェクトの相対誤差  $MRE_i$  (Magnitude of Relative Error) を、実績工数を  $effort_i$ 、見積り工数を  $effort'_i$  として式 (8) のように定義すると、 $n$  件の見積りを行った場合の相対誤差平均  $MMRE$  (Mean Magnitude of Relative Error) は式 (9) で定義される。

$$MRE_i = \frac{|effort_i - effort'_i|}{effort_i} \quad (8)$$

$$MMRE = \frac{1}{n} \sum_{i=1}^n MRE_i \quad (9)$$

相対誤差平均の値が小さいほど全体として誤差が小さい、すなわち見積り精度が高いことを示す。

(5) 相対誤差中央値: 相対誤差中央値は相対誤差  $MRE_i$  の中央値であり、相対誤差平均と

同様に、値が小さいほど見積り精度が高いことを示す。また、絶対誤差中央値と同様に、相対誤差平均とあわせて見ることで、見積り精度の高さをより正確に判断することができる。

(6) 相対誤差分散：相対誤差分散  $VMRE$  (Variance Magnitude of Relative Error) は式 (10) で定義される。

$$VMRE = \frac{1}{n-1} \sum_{i=1}^n (MRE_i - MMRE)^2 \quad (10)$$

相対誤差分散も絶対誤差分散と同様に誤差のばらつきの大きさを示しており、値が小さいほど誤差のばらつきが小さく、安定して見積もることができたことを示す。

(7)  $PRED(0.25)$ :  $MRE_i$  が 0.25 以下であったプロジェクト件数を  $m$  とすると、 $PRED(0.25)$  は式 (11) で定義される。

$$PRED(0.25) = \frac{m}{n} \quad (11)$$

$PRED(0.25)$  は全プロジェクト中で、相対誤差 0.25 以下で見積も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、 $PRED(0.25)$  の値が大きいほど正しく見積もることができたプロジェクトの割合が大きかったことを示す。

#### 5.4 実験手順

本論文では leave-one-out 法に基づいて、各プロジェクトの工数を見積もり、見積もった工数を実績値と比較することで精度評価尺度を算出した。実験で用いた leave-one-out 法の手順を以下に示す。

1. データ中のプロジェクトを 1 つ選ぶ。
2. 選んだプロジェクトにおける工数の実績値を隠し、見積り対象のデータ (テストデータ) とする。
3. その他のプロジェクトを見積りの根拠として使用するデータ (フィットデータ) とする。
4. フィットデータを用いて変数選択を行う。
5. 変数選択したフィットデータを用い、テストデータの工数を見積もる。
6. 見積もった工数と実績値を比較する。
7. 上記、1 から 6 をデータに含まれる全プロジェクトに対して行い、精度評価指標を算出する。

## 6. 実験結果

### 6.1 精度

各変数選択法を適用した場合と、変数選択を行わなかった場合の各精度評価指標の値を表 1、表 2、表 3 に示す。それぞれ、表 1 は Desharnais データによる実験の結果を、表 2 は Kitchenham データによる実験の結果を、表 3 は ISBSG データによる実験の結果を表しており、各列は左から順に、適用した変数選択法、絶対誤差平均、絶対誤差中央値、絶対誤差分散、相対誤差平均、相対誤差中央値、相対誤差分散、 $PRED(0.25)$  を表す。

表 1 を見ると、Desharnais データによる実験では、変数選択をしない場合と比べて、いずれの変数選択法を適用した場合でも、絶対誤差分散を除くすべての尺度で見積り精度が同等もしくは向上した。また、CTS とその他の変数選択法を比較すると、CTS は半数以上の指標 (絶対誤差平均、絶対誤差分散、相対誤差平均、相対誤差分散) について最も精度が高くなっており、特に誤差の分散が大きく改善されていた。相対誤差中央値と  $PRED(0.25)$  については、他の変数選択法よりも精度が低い、その差はごくわずかであった。なお、 $PRED(0.25)$  は値が大きいほど精度が良いことを示すことに注意されたい。

表 2 を見ると、Kitchenham データによる実験では、変数減少法、変数増加法を適用したときは見積り精度が向上したが、CTS を適用したときはほとんど効果がなかったといえる。CTS を適用したとき、いずれの精度評価指標の値についてもほとんど変化が見られず、見積り精度が向上しなかった。また、変数減少法、変数増加法についても、絶対誤差平均、絶対誤差中央値、相対誤差平均などについてある程度改善が見られるが、Desharnais データによる実験ほどの精度改善効果は得られなかった。

表 3 を見ると、ISBSG データによる実験では、いずれの変数選択法を適用した場合でも見積り精度が向上しており、CTS を適用したときの効果が最も高かったといえる。いずれの手法を適用した場合でも、すべての精度評価指標の値が改善されており、特に、CTS を適用したときの効果は多くの精度評価指標において最も高いものであった。

### 6.2 計算所要時間

計算に要した時間 (秒) を表 4 に示す。この表から、Desharnais データでは CTS に比べて変数増加法は 6~7 倍、変数減少法は 9~10 倍の時間を要していたことが分かる。同様に Kitchenham データでは変数減少法は 5~6 倍、変数増加法は 6~7 倍、ISBSG データでは変数減少法は 8~12 倍、変数増加法は 9~17 倍の時間を要していたことが分かる。それぞれの計算量は、CTS は  $O(1)$ 、変数減少法と変数増加法は  $O(n^2)$  であることから、説明

表 1 Desharnais データでの実験結果  
Table 1 Experimental result for Desharnais dataset.

変数選択法	絶対誤差平均	絶対誤差中央値	絶対誤差分散	相対誤差平均	相対誤差中央値	相対誤差分散	PRED (0.25)
変数選択せず	2385.11	1393.42	5982818.17	0.75	0.37	0.87	0.30
CTS	2012.16	1222.03	4348573.91	0.43	0.37	0.10	0.34
変数減少法(MMRE)	2094.86	1029.00	6145171.35	0.47	0.35	0.24	0.32
変数減少法(MMAE)	2159.07	1294.57	5149198.60	0.56	0.36	0.31	0.38
変数増加法(MMRE)	2164.15	1029.00	6259651.61	0.49	0.35	0.27	0.32
変数増加法(MMAE)	2136.06	1292.32	5192108.40	0.55	0.36	0.32	0.39

表 2 Kitchenham データでの実験結果  
Table 2 Experiment result for Kitchenham dataset.

変数選択法	絶対誤差平均	絶対誤差中央値	絶対誤差分散	相対誤差平均	相対誤差中央値	相対誤差分散	PRED (0.25)
変数選択せず	1626.60	700.04	19586516.49	0.91	0.43	3.19	0.28
CTS	1497.01	843.27	11904038.05	0.89	0.45	3.36	0.25
変数減少法(MMRE)	1181.30	590.66	3922570.15	0.75	0.43	1.50	0.33
変数減少法(MMAE)	1510.00	634.40	19434357.22	0.83	0.43	2.29	0.33
変数増加法(MMRE)	1182.01	634.40	3921707.68	0.75	0.43	1.50	0.33
変数増加法(MMAE)	1519.22	634.40	19427260.14	0.85	0.43	2.50	0.33

表 3 ISBSG データでの実験結果  
Table 3 Experimental result for ISBSG dataset.

変数選択法	絶対誤差平均	絶対誤差中央値	絶対誤差分散	相対誤差平均	相対誤差中央値	相対誤差分散	PRED (0.25)
変数選択せず	1392.99	439.32	7091284.25	8.85	0.72	5726.33	0.14
CTS	1145.25	270.92	5681154.57	1.75	0.60	12.43	0.20
変数減少法(MMRE)	1222.61	393.73	6221720.50	2.11	0.67	19.76	0.23
変数減少法(MMAE)	1149.78	423.64	3558615.69	2.72	0.69	52.52	0.19
変数増加法(MMRE)	1230.73	397.73	6210590.87	2.12	0.68	19.75	0.22
変数増加法(MMAE)	1156.38	423.64	3545994.30	2.73	0.70	52.45	0.20

表 5 Desharnais データでの変数選択結果  
Table 5 Variable selection result for Desharnais dataset.

変数選択法	Duration	ExpEquip	ExpProjMan	Transactions	FPS	Dev Env1	Dev Env2	Dev Env3
CTS	98.3%	89.4%	1.3%	89.6%	100.0%	2.6%	0.0%	89.4%
変数減少法(MMRE)	50.6%	49.4%	5.2%	61.0%	70.1%	100.0%	100.0%	100.0%
変数減少法(MMAE)	41.6%	15.6%	20.8%	81.8%	72.7%	74.0%	63.6%	97.4%
変数減増法(MMRE)	54.5%	49.4%	5.2%	63.6%	83.1%	100.0%	100.0%	100.0%
変数減増法(MMAE)	42.9%	15.6%	20.8%	81.8%	75.3%	81.8%	67.5%	97.4%

  

変数選択法	Year Fin_83	Year Fin_84	Year Fin_85	Year Fin_86	Year Fin_87	Year Fin_88	Entities
CTS	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	89.6%
変数減少法(MMRE)	3.9%	32.5%	11.7%	51.9%	0.0%	0.0%	49.4%
変数減少法(MMAE)	31.2%	42.9%	22.1%	83.1%	5.2%	20.8%	48.1%
変数減増法(MMRE)	3.9%	36.4%	11.7%	55.8%	0.0%	0.0%	49.4%
変数減増法(MMAE)	32.5%	45.5%	22.1%	89.6%	13.0%	22.1%	51.9%

表 4 計算総所要時間結果  
Table 4 Required time for estimation.

変数選択法	Desharnais	Kitchenham	ISBSG
CTS	5.73	24.70	22.10
変数減少法(MMRE)	35.75	114.59	185.81
変数減少法(MMAE)	34.10	108.54	253.10
変数減増法(MMRE)	50.35	167.46	206.59
変数減増法(MMAE)	46.84	152.70	300.58

変数の多いデータセットでは計算時間の差はさらに広がると予想される。

### 7. 考 察

Desharnais データおよび ISBSG データに変数選択法を適用した場合、CTS は変数減少

法や変数減増法に比べて高い工数見積り精度を示した。一方、Kitchenham データでは変数減少法や変数減増法の方が CTS に比べて高い精度を示した。そこで変数選択結果の違いを、各変数が選択された割合から調査した。その結果を表 5、表 6 に示す。ISBSG データについては、変数が多く、Desharnais データと同様の傾向を示したので割愛した。

表 5 より、Desharnais データでは、CTS を適用したときは、各説明変数が 0%または 100%に近い割合で選択されており、予測対象プロジェクトにかかわらず変数選択結果はほぼ一定であるといえる。一方、変数減少法や変数減増法を適用したときは選択される割合が 50%程度の変数が含まれており、試行（予測対象プロジェクト）ごとに変数選択結果が大きく異なっている。そのため、試行によっては、見積り精度が極端に悪くなった可能性がある。表 1 を見ると、相対誤差中央値については、CTS、変数減少法、変数減増法の間で大差がないのに対し、相対誤差平均値については、変数減少法と変数減増法は CTS に劣ることから、予測が大きく外れたプロジェクトが存在することがうかがえる。

一方、表 6 より、Kitchenham データでは、全 14 個の説明変数のうち Client code 1~6、Project type A~U の 12 個がカテゴリ変数であり、CTS を適用したときにこれらのカ



表 6 Kitchenham データでの変数選択結果  
Table 6 Variable selection result for Kitchenham dataset.

変数選択法	Client code1	Client code2	Client code3	Client code4	Client code5	Client code6	Project type_A
CTS	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
変数減少法(MMRE)	98.5%	98.5%	0.0%	0.7%	0.7%	2.2%	0.0%
変数減少法(MMAE)	100.0%	100.0%	1.5%	91.1%	3.0%	2.7%	4.2%
変数減増法(MMRE)	98.5%	98.5%	0.7%	0.7%	0.7%	3.0%	0.0%
変数減増法(MMAE)	100.0%	100.0%	7.4%	94.1%	3.0%	9.6%	5.2%
変数選択法	Project type_D	Project type_P	Project type_C	Project type_Pr	Project type_U	Duration	FPs
CTS	1.5%	1.5%	0.0%	0.0%	0.0%	39.3%	100.0%
変数減少法(MMRE)	0.0%	100.0%	1.5%	4.4%	98.5%	0.0%	100.0%
変数減少法(MMAE)	6.7%	98.5%	5.9%	6.7%	83.0%	0.7%	100.0%
変数減増法(MMRE)	0.0%	100.0%	1.5%	5.2%	99.3%	0.0%	100.0%
変数減増法(MMAE)	6.7%	98.5%	6.7%	7.4%	80.7%	0.7%	100.0%

カテゴリ変数がほとんど選択されなかった。このことが、CTS の効果が小さかった原因と考えられる。

このような結果となった理由としては、多くのカテゴリ変数は工数との相関が低いため CTS では選ばれなかったと考えられる。さらに調査を行った結果、いくつかのカテゴリ変数は、開発の生産性（工数を規模で割ったもの）との相関が高いことが分かった。類似プロジェクトの選択時に、「工数が類似する」だけでなく「生産性が類似する」ことも考慮するならば、今後、CTS を拡張し、カテゴリ変数については生産性との相関係数を基準にして変数選択を行うことで、予測精度が改善される可能性がある。

以上のことから、現時点では、カテゴリ変数が少ない場合には CTS を、多い場合には変数減少法が変数減増法を適用するのが望ましいと考える。CTS におけるカテゴリ変数の扱いは今後の重要な課題である。

## 8. おわりに

本論文では、プロジェクト類似性に基づく工数見積りに対する効率的な変数選択法とし

て、Correlation Threshold based Selection (CTS) を提案した。ソフトウェア開発企業で収集された 3 種類の実績データを用いた評価実験の結果、特に、カテゴリ変数の数が少ない場合に、CTS の有効性を確認できた。一方、カテゴリ変数の多い（量的変数が 2 つしかない）データセットでは CTS の効果は小さかった。計算量については、CTS は  $O(1)$  であるのに対し、擬似ステップワイズは  $O(n^2)$  であり、実験においても、擬似ステップワイズは CTS よりも 5~17 倍の計算時間を要した。

今後の課題としては、より多くのデータセットを用いて同様の実験を繰り返すことにより、CTS が有用となるコンテキストをより明確にしていくことが重要となる。

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」および「次世代 IT 基盤構築のための研究開発」の委託に基づいて行われた。

## 参 考 文 献

- 1) Shepperd, M. and Schofield, C.: Estimating Software Project Effort Using Analogies, *IEEE Trans. Softw. Eng.*, Vol.23, No.12, pp.736-743 (1997).

- 2) 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).
- 3) Kohavi, R. and John, G.H.: Wrappers for feature subset selection, *Artificial Intelligence*, Vol.97, Issue 1-2, pp.273-324 (1997).
- 4) Hall, M.: Correlation-Based Feature Selection for Discrete and Numeric Class Machine Learning, *17th International Conference on Machine Learning*, pp.359-366 (2000).
- 5) 高橋 一 (編): 計量経済学, 八千代出版 (1993).
- 6) Hawkins, D.M.: The Problem of Overfitting, *Journal of Chemical Information and Modeling*, Vol.44, No.1, pp.1-12 (2004).
- 7) Desharnais, J.M.: Analyse Statistique de la Productivite des Projets Informatique a Partie de la Technique des Point des Function, Masters Thesis, University of Montreal (1989).
- 8) Mair, C., Kadoda, G., Lefley, M., Phalp, K., Schofield, C., Shepperd, M. and Webster, S.: An Investigation of Machine Learning Based Prediction Systems, *Journal of Systems and Software*, Vol.53, Issue 1, pp.23-29 (2000).
- 9) Kitchenham, B., Pfleeger, S.L., McColl, B. and Eagan, S.: An Empirical Study of Maintenance and Development Estimation Accuracy, *Journal of Systems and Software*, Vol.64, Issue 1, pp.57-77 (2002).
- 10) International Software Benchmarking Standards Group (ISBSG): ISBSG Estimating, Benchmarking and Research Suite Release 9, ISBSG (2004).
- 11) Ghiselli, E.E.: *Theory of psychological measurement*, McGraw-Hill (1964).
- 12) Kirsopp, C., Shepperd, M. and Hart, J.: Search Heuristics, Case-based Reasoning and Software Project Effort Prediction, *Genetic and Evolutionary Computation Conference*, New York, USA (2002).
- 13) 柿元 健, 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工数見積手法のデータの欠損に対するロバスト性の評価, *電子情報通信学会論文誌 D*, Vol.J89-D, No.12, pp.2602-2611 (2006).
- 14) Walkerden, F. and Jeffery, R.: An Empirical Study of Analogy-based Software Effort Estimation, *Empirical Software Engineering*, Vol.4, Issue 4, pp.135-158 (1999).
- 15) 奥野忠一, 久米 均, 芳賀敏郎, 吉澤 正: 多変量解析法, *日科技連* (1971).

(平成 19 年 10 月 17 日受付)

(平成 20 年 4 月 8 日採録)



瀧 進也

平成 17 年京都工芸繊維大学電子情報工学科卒業。平成 19 年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修了。同年株式会社日立製作所入社。修士(工学)。ソフトウェア開発工数予測等の研究に興味を持つ。



戸田 航史(学生会員)

平成 16 年大阪大学基礎工学部卒業。平成 18 年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修了。現在, 同大学博士後期課程在籍。ソフトウェアメトリクスの研究に従事。電子情報通信学会, IEEE 各会員。



門田 暁人(正会員)

平成 6 年名古屋大学工学部電気学科卒業。平成 10 年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修了。同年同大学同研究科助手。平成 16 年同大学助教授。平成 19 年同大学准教授。平成 15~16 年 Auckland 大学客員研究員。博士(工学)。ソフトウェアメトリクス, ソフトウェアセキュリティ, ヒューマンファクタ等の研究に従事。電子情報通信学会, 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。



柿元 健(正会員)

平成 15 年神戸市立工業高等専門学校専攻科電気電子工学専攻修了。平成 20 年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修了。同年大阪大学大学院情報科学研究科特任助教。博士(工学)。ソフトウェア信頼性/開発工数予測等の研究に従事。電子情報通信学会, IEEE 各会員。



角田 雅照 (正会員)

平成 9 年和歌山大学経済学部卒業。平成 19 年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修了。同年同大学同研究科特任助教。博士 (工学)。ソフトウェアメトリクス, ユビキタスコンピューティングの研究に従事。電子情報通信学会, ヒューマンインタフェース学会, IEEE 各会員。



大杉 直樹 (正会員)

平成 13 年奈良工業高等専門学校専攻科・電子情報専攻修了。平成 16 年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修了。同年同大学研究員。平成 19 年株式会社 NTT データ入社, 技術開発本部勤務。博士 (工学)。エンピリカルソフトウェア工学, 生産性/品質データ収集/分析, メトリクスの研究に従事。電子情報通信学会, IEEE 各会員。



松本 健一 (正会員)

昭和 60 年大阪大学基礎工学部情報工学科卒業。平成元年同大学大学院博士課程中退。同年同大学基礎工学部情報工学科助手。平成 5 年奈良先端科学技術大学院大学助教授。平成 13 年同大学教授。工学博士。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。電子情報通信学会, 日本ソフトウェア科学会, ACM 各会員,

IEEE Senior Member .