

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

Effect of Selecting Software Projects Likely to Fail in Effort Estimation

戸田 航史* 門田 暁人† 松本 健一‡

あらまし ソフトウェア開発工数予測において、全ての開発プロジェクトが精度良く予測できるとは限らず、予測誤差が100%を超える場合も少なくない。そのようなプロジェクトについては、予測自体を行わない(予測回避)ことが必要であると考えられる。本稿では、類似性に基づく工数予測手法において、類似プロジェクトのばらつきに基づいて予測回避プロジェクトを選定し、予測対象から外す方法について検討する。また、選定結果に基づいて予測回避を行った結果、予測を大きく外すプロジェクトが減少することを実験により示す。

1 はじめに

ソフトウェア開発において、工数は重要な指標のひとつである。工数とは開発要員数×時間で算出される、のべ作業時間をあらわす指標であり、単位は多くの場合、人月または人時である。ソフトウェア開発にかかるコストの多くは人件費に由来するため、コストに直結する工数の予測は、プロジェクトを予定通りに完遂させるために必要な人的資源やスケジュール管理を行う上で重要である。必要な工数を過不足無く予測することは、納期の遅れやコストの超過を防ぐことができるためである。このことから、高い精度で工数を予測することを目的として数多くの研究が行われてきた [1] [2] [3] [4]。

図1は筆者らが以前に行った工数予測の結果である [5]、ISBSG [6] のデータセットから作成した、欠損のないサブセットに対してステップワイズ重回帰モデルを用いて予測を行った。ISBSGは実際のソフトウェア開発プロジェクトで得られる工数や開発言語などのプロジェクトデータの収集を行っている組織である。横軸は相対誤差、縦軸はプロジェクトの件数を表す。図からわかるように予測結果には相対誤差が30%以下のプロジェクトが多数含まれており、これらのプロジェクトでは予測が成功したといえる。しかし、相対誤差が100%を超える、いわば予測に失敗したプロジェクトも成功したプロジェクトほどではないにせよ含まれている。この結果は全体としての予測精度がどれほど高くとも、いくつかのプロジェクトでは予測に失敗する可能性があることを示しており、平均的に精度の高い手法を採用しても、予測に失敗することがありえる。そして、予測に失敗する可能性があるならば、予測結果を得ることができてもその結果を用いることは難しい。これまでの工数予測手法に関する研究では、大きく予測が外れる可能性、すなわち予測の失敗が目目されることはほとんどなかった。

そこで本研究では「類似性に基づく工数予測において、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、類似するプロジェクトのばらつきを利用して予測に失敗しやすいプロジェクトを選定する。そして、予測に失敗すると判定されたプロジェクトは予測自体を回避すべきプロジェクト(予測回避プロジェクト)として、そもそも予測対象から外すことを考える。仮説の検証のために、類似性に基づく工数予測において、予測に失敗したプロジェクトとその類似プロジェクトのばらつきを示す値と

*Koji Toda, 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

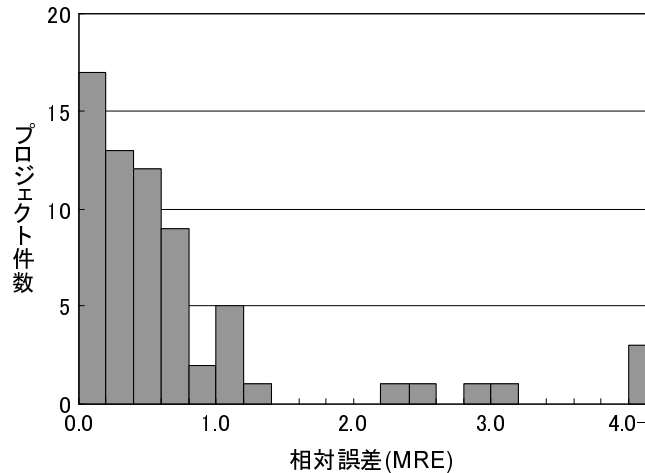


図1 重回帰分析による予測結果の一例

の関係を実際のソフトウェア開発プロジェクトで得られたデータを用いて検証する。予測に失敗したプロジェクトの定義は、予測の相対誤差が100%を超えるプロジェクトとした。

以降、2章では仮説について詳しく述べ、3章では仮説の検証で用いた類似性に基づく工数予測手法について述べる。続く4章では仮説を検証する実験の概要について述べ、5章では実験の結果を、6章では結果に対する考察を述べる。最後に7章で本稿の結論について述べる。

2 仮説

本稿では、「類似性に基づく工数予測において、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、予測に失敗しやすいプロジェクトを選定し、予測対象から外す方法について検討する。

仮説では「プロジェクト群のばらつき」として2つのばらつきを想定している。1つは類似しているプロジェクト群と予測対象との間の類似度のばらつき、もう1つは、類似しているプロジェクト群の中での予測対象の変数、本稿の場合は工数のばらつきである。前者をばらつきとして採用した場合を仮説1、後者を採用した場合を仮説2として、詳細を以下に述べる。

仮説1 仮説1は、「予測対象に類似していると判定されたプロジェクト群の類似度のばらつきが大きい場合、その予測は失敗しやすい」というものである。これは予測対象のプロジェクトに似ていると判定されたからといって、類似度にばらつきがないとは限らないことに由来する。例として、類似度として予測対象プロジェクトとのユークリッド距離を用い、予測に用いたデータセットの上位3件のプロジェクトを類似していると判定する場合を考える。このとき3件それぞれと予測対象プロジェクトとのユークリッド距離がほとんど0、すなわち全てのプロジェクトが予測対象プロジェクトと十分類似している場合には問題はないが、3件のうち2件がほとんど0だが、残りの1件の値(ユークリッド距離)が極めて大きければ、本当に類似しているプロジェクトとそうでないプロジェクト混在させて予測に用いていることになり、予測精度が低くなる可能性がある。

仮説2 仮説2は、「予測対象に類似していると判定されたプロジェクト群の工数の

ばらつきが大きい場合、その予測は失敗しやすい」というものである。類似性に基づく工数予測では、類似プロジェクトの工数を加重平均して予測工数を得る。その算出過程において、類似プロジェクトの工数が信頼できるかどうかは考慮されない。そのため、類似プロジェクトの工数の信頼性に問題があったとしても、その値がそのまま予測工数の算出に用いられる。この類似プロジェクトの問題がある場合の例として、類似プロジェクトの工数のばらつきが大きい場合を考える。例として、先ほどと同様に上位3件を類似していると判定する場合を考える。その上位3件が非常に類似度が高いと判定されたとしても、その工数が、例えば10人月、500人月、1000人月であった場合、記録ミスや特異プロジェクトを含む可能性が高く、算出される工数は実際のものとはかけ離れた値となるおそれがある。

このように、2つのばらつきのどちらが大きい場合にも、その精度は低くなる可能性がある。1章でも述べたとおり、どれほど全体として精度が高くとも、予測に失敗する可能性があるならば、その結果を採用することは難しく、また予測に失敗する可能性が高いプロジェクトは予測対象から除外すべきである。それゆえ、仮説が正しいならば、ばらつきが大きいプロジェクトでは予測を回避すべきであると考えられる。

3 類似性に基づく工数予測

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予測を目指してプロジェクト類似性に基づく工数予測の研究が行われてきた [4] [7]。この予測方法は、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予測する場合には、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、その実績工数から予測対象プロジェクトの工数を予測する。反対に、開発規模が大きいプロジェクトの工数を予測する場合には、開発規模が大きいプロジェクトの実績工数から予測を行う。プロジェクト類似性に基づく工数予測では、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範囲を $[0,1]$ に正規化する。この世紀か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、予測対象のプロジェクトと過去に行われたプロジェクトとの類似度を、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、特性地の系列をベクトルとみなし、ベクトル間の角度を用いる方法や、ベクトル間のユークリッド距離を用いる方法が提案されている。いずれの方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みなされるが、欠損値（データセット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、欠損値の存在が結果に影響することを避けるため、欠損値のないデータおよびユークリッド距離を用いる方法を採用した。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、予測対象プロジェクトの工数の算出（予測値計算）を行う。この際に「類似度の高いプロジェクト」として予測対象プロジェクトよりも規模が大きい、または小さ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な平均では予測が外れる可能性が高くなる。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にする。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 とすると、予測対象である a 番目のプロジェクトの予測値 $effort'_a$ は式(4)のように定義される。

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

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

ここで $k - nearestprojects$ 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。

4 検証実験

4.1 概要

検証実験の目的は、2章で述べた「類似性に基づく工数予測において、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」という仮定が、現実のソフトウェア開発プロジェクトで得られた定量的データを用いた場合に成立するかどうかを確かめ、ばらつきによる選定が予測に失敗するプロジェクトを特定可能かを検証することである。具体的には、実験では3種類の実際の開発プロジェクトから得られたデータを用いて、要件定義および設計が終了した時点を想定して総工数を予測し、その結果から得られるばらつきを表す指標と予測精度の関係についての分析を行った。以降では、実験に用いた、データ、精度評価指標について述べ、続いて実験手順について述べる。

4.2 利用したデータセット

本研究では、Desharnais データ [8] [9]、Kitchenham データ [10]、ISBSG データ [11] 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。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11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。ただし、このデータセットには開発の終了年などの工数予測に用いるには不適切な説明変数が含まれていたため、変数8種類を含むサブセットを作成し、実験に用いた。Kitchenham データは、Kitchenham 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。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、このデータセットにも Desharnais データと同様に、工数予測に用いるには不適切な説明変数や欠損値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 7 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。データには 1989 年から 2004 年までの 3026 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 232 件、開発種別や開発言語を含む変数 8 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、説明変数に質的変数と量的変数が混在しており、工数を予測する前に、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 $p-1$ 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 8 個、Kitchenham データの変数は 15 個、ISBSG データの変数は 41 個となった。実験においてステップワイズ法に代表される、計算による変数選択は行わなかった。

4.3 精度評価指標

評価実験では、予測誤差を評価するために相対誤差 MRE (Magnitude of Relative Error) [12] を利用し、ここから導くことのできる MdMRE および $\text{Pred}(\overline{100})$ を予測に失敗したかどうかを評価する基準として用いる。MdMRE は MRE の中央値を表し、 $\text{Pred}(\overline{100})$ は、相対誤差が 100% を越えるプロジェクトが全体に占める割合を表す。MRE および $\text{Pred}(\overline{100})$ の算出式を以下に示す。式中、実測値を E で表し、モデルから算出される予測値を \hat{E} で示す。

MRE

$$MRE = \left| \frac{E - \hat{E}}{E} \right| \quad (5)$$

$\text{Pred}(\overline{100})$

$$\text{Pred}(\overline{100}) = \frac{\sum_{i=1}^M \text{isAccurate}(R_i)}{M} \quad (6)$$

$$\text{isAccurate}(R) = \begin{cases} 1 & R \geq 1 \\ 0 & R < 1 \end{cases} \quad (7)$$

4.4 手順

実験では各データセットに対して leave-one-out 法を適用し、予測を行った。実験手順を以下に示す。

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、予測対象のデータ（テストデータ）とし、それ以外のプロジェクトを予測の根拠として使用するデータ（フィットデータ）とする。
3. フィットデータを用いて、テストデータの工数を類似性に基づいた予測手法で予測する。このとき、類似度の算出にはプロジェクト間のユークリッド距離を、類似ケース数は 5 件とした。

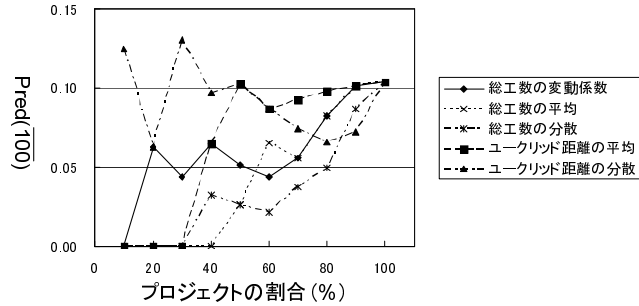


図2 Desharnais データにおけるばらつきの指標ごとの $\text{Pred}(\overline{100})$ の変化

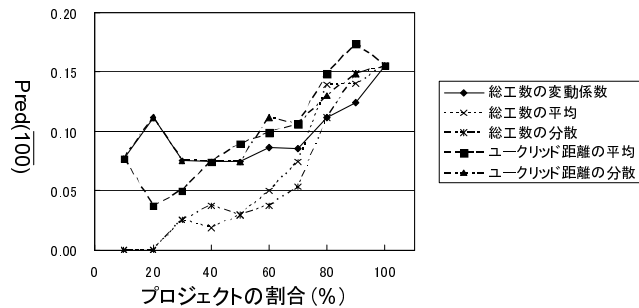


図3 Kitchenham データにおけるばらつきの指標ごとの $\text{Pred}(\overline{100})$ の変化

4. 実験後の分析のために、類似度計算の結果、類似していると判定されたプロジェクト（以降類似プロジェクトと呼ぶ）の予測値からばらつきを表す指標を算出する。
5. 工数の予測値と実績値を比較し、MRE を算出する。
6. 上記、1 から 5 をデータに含まれる全プロジェクトに対して行う。
7. 全プロジェクトの結果から得られた MRE を用いて $\text{Pred}(\overline{100})$ を算出する

本実験では類似度の算出にはユークリッド距離を採用した。工数のばらつきを表す指標として、類似プロジェクトの総工数の変動係数、平均および分散を、類似度のばらつきを表す指標として類似プロジェクトと予測対象プロジェクトとの間のユークリッド距離の平均および分散を用いた。本実験の場合、変動係数は 5 件の類似プロジェクトの総工数の標準偏差を平均値で割ったものとなる。

5 実験結果

実験から得られた $\text{Pred}(\overline{100})$ を図 2～図 4 に、MdMRE を図 5～図 7 に示す。図中では横軸はばらつきの指標として用いた値の下位（ばらつきの小さい）何パーセントが結果として採用されたかを示す。縦軸は $\text{Pred}(\overline{100})$ では相対誤差が 100% を超えるプロジェクトが採用された結果に占める割合を示し、MdMRE では採用された結果の MRE の中央値を示す。例えば $\text{Pred}(\overline{100})$ の図で、総工数の変動係数の縦軸の値が 0.05、横軸の値が 50 ならば、変動係数の値が下位 50% のプロジェクト中に相対誤差が 100% を超えているプロジェクトが占める割合が 0.05 すなわち 5% であることを示す。

まず $\text{Pred}(\overline{100})$ について述べる。 $\text{Pred}(\overline{100})$ は、総じて平均および分散の値との

Effect of Selecting Software Projects Likely to Fail in Effort Estimation

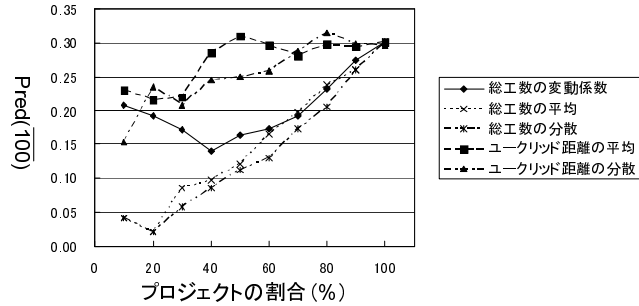


図 4 ISBSG データにおけるばらつきの指標ごとの $\text{Pred}(\overline{100})$ の変化

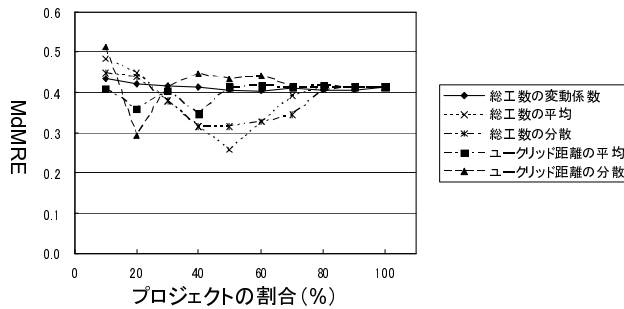


図 5 Desharnais データにおけるばらつきの指標ごとの MdMRE の変化

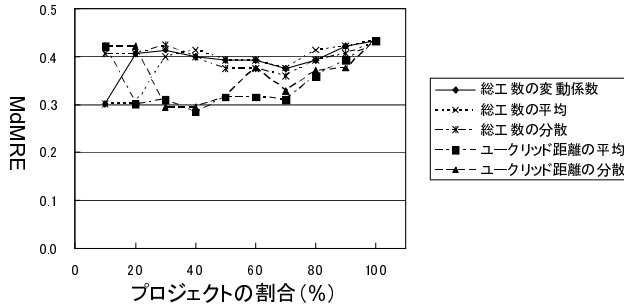


図 6 Kitchenham データにおけるばらつきの指標ごとの MdMRE の変化

関係が深いことが分かった。データセットごとの結果では、図 2 から Desharnais のデータではばらつきの指標として総工数の平均を用いた場合、平均値の下位 40% を結果として採用すれば、相対誤差が 100% を超えることがなくなるということが分かる。このことから類似プロジェクトの総工数の平均値が最も $\text{Pred}(\overline{100})$ との関係が深いことが分かり、続いて総工数の分散、ユークリッド距離の分散と平均がそれに続いていることが分かる。同様に図 3 から、Kitchenham データでは、総工数の平均および総工数の分散をばらつきの指標として採用すると、下位 20% のプロジェクトには相対誤差 100% 以上のプロジェクトが含まれていないことが分かり、類似プロジェクトの総工数の平均と総工数の分散が $\text{Pred}(\overline{100})$ との関係が深く、それ以外のものは、それほど深い関係ではないといえる。最後に図 4 から、ISBSG データでは相対誤差が 100% を超えるプロジェクトが含まれなくなる状況はないものの、総工数の

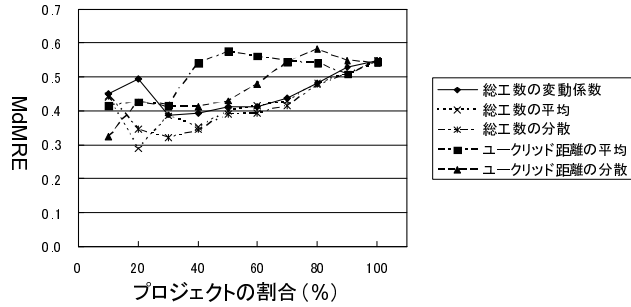


図7 ISBSG データにおけるばらつきの指標ごとの MdmRE の変化

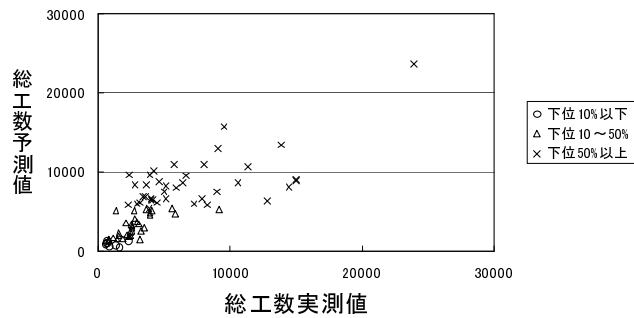


図8 類似プロジェクトの工数平均値と予測対象の工数実績・予測値との関係

分散が、続いて総工数の平均，総工数の変動係数の順に $\text{Pred}(\overline{100})$ との関係が深いことが分かる。

次に MdmRE について述べる．MdmRE とばらつきをあらわす指標との間に明確な関係は見られなかった，データセットごとの結果では，Desharnais データセットでは，図5から下位50%前後では総工数の平均，および分散の場合に限り一時的に精度が向上するものの，それ以外では類似プロジェクトの総工数のばらつきはそれほど精度に影響を与えていないことが分かる．Kitchenham データセットでは，図6から，一旦はばらつきが小さくなるにつれて精度が向上していくが，下位10%の場合のみ精度が悪くなっているユークリッド距離の平均や，それまでは精度がほとんど変わっていないのに下位20%から急激に精度が向上する総工数の平均など，精度の変化とばらつきを表す指標の間に相関がみられないことがわかる．ISBSG データセットでは，図7から，おおむねどの指標でもばらつきが小さくなるにつれて精度も向上している．しかしながら下位30%以下では精度が悪くなっていく総工数の平均，下位10%を結果に採用した場合のみ精度が急激に向上しているユークリッド距離の分散，下位20%を採用した場合のみ精度が極端に低い総工数の平均など，どの指標が MdmRE と関係があるかを図から読み取ることは難しいである．

6 考察

実験結果から，ばらつきを表す指標の一部について， $\text{Pred}(\overline{100})$ すなわち予測に失敗したプロジェクトが占める割合との間に関係があることが分かった．Desharnais データでは類似プロジェクトの平均値が下位40%のものには相対誤差が100%を超えるプロジェクトが含まれていなかった．同様に Kitchenham データでは類似プロ

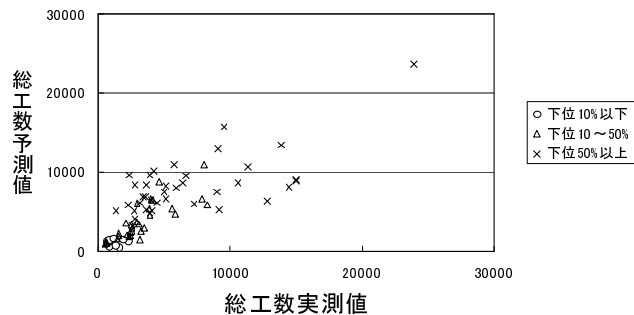


図9 類似プロジェクトの工数分散値と予測対象の工数実績・予測値との関係

プロジェクトの総工数の平均値および分散が下位 20%に含まれているものでは予測に失敗したプロジェクトが存在していなかった。ISBSG データセットでは予測に失敗したプロジェクトが存在しないような区分は得られなかったものの、類似プロジェクトの総工数の平均値と分散では常に、総工数の変動係数では中盤まではばらつきが小さくなるに従い、予測に失敗したプロジェクトが含まれる割合は単調減少していた。全てのデータセットについて平均値と分散の値と $\text{Pred}(\overline{100})$ の間に関係があることが示されたことを踏まえ、以下ではそのような関係が表れた理由について考察する。平均、分散に共通している事の 1 つに、計算式の都合上、元の値が大きければこれらの算出値も大きくなる傾向にあるということが挙げられる。これに対して変動係数は算出元の値に左右されず、ユークリッド距離も、本稿では標準化を行ったため同様である。このことから、そもそも類似性に基づく工数予測では、元の値すなわち工数が大きいプロジェクトの予測では相対誤差が大きくなる傾向にあり、そのために類似プロジェクトの平均値や分散が大きい場合には予測結果を採用しないほうが望ましいという結果が導かれた可能性がある。例として図 8 に Desharnais データにおける予測値と実績値と類似プロジェクトの平均値、図 9 に予測値と実績値と類似プロジェクトの分散の関係を表すグラフを示す。横軸は実績値、縦軸は予測値を表している。また図中「下位 10%」とは、その点が表すプロジェクトの類似プロジェクトの平均（分散）の値が、データセットに含まれる全プロジェクトの平均（分散）の値の下位 10%以下に含まれているということを示す。図から明らかのように、平均、分散の値が小さくなるに従って工数の規模も小さくなっており、このことから、類似性に基づく工数予測では工数の大きい、すなわち規模の大きいプロジェクトを予測する場合、予測に失敗しやすいということが明らかとなった。また、類似性に基づく工数予測では、大規模なプロジェクトを予測する場合には、予測に失敗する可能性が小規模なプロジェクトを予測する場合よりも高いことが分かった。そして大規模なプロジェクトは、小規模なプロジェクトに比べて予測に失敗した場合の損害が大きくなるため、大規模なプロジェクトの予測には類似性に基づく工数予測手法を用いることは避けるべきである。

7 おわりに

本稿では「類似性に基づく工数予測において、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、予測に失敗しやすいプロジェクトを選定し、予測対象から外す方法について検討した。そして選定結果に基づいて予測回避を行った結果、予測を大きく外すプロジェクトが減少することを実験から検証した。検証実験では、ばらつきの指標として類似プロジェクトの目的変数（総工数）の分散および平均の値が大きい場

合には予測対象から外すべきであることが分かった。実験結果から、規模の大きいプロジェクトは類似度に基づく工数予測手法での予測には向かないことが分かった。このような結果になった原因の1つとして、規模が大きいプロジェクトは数が少ないということが考えられる。今後、規模の大きいプロジェクトを予測するのに適した手法と、類似度に基づく手法を組み合わせることで予測対象から外すべきプロジェクトを減らし、より外れる可能性の低い予測手法を開発することが考えられる。

謝辞 本研究の一部は、文部科学省「次世代IT基盤構築のための研究開発」および「先導的ITスペシャリスト育成推進プログラム」の委託に基づいて行われた

参考文献

- [1] B. Boehm. *Software Engineering Economics*. Prentice Hall, 1981.
- [2] M. Shepperd and C. Schofield. Estimating software project effort using analogies. *IEEE Trans. on Software Eng.*, Vol. 23, No. 12, pp. 736–743, 1997.
- [3] K. Srinivasan and D. Fisher. Machine learning approaches to estimation software development effort. *IEEE Trans. on Software Eng.*, Vol. 21, No. 2, pp. 126–137, 1995.
- [4] 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.
- [5] 戸田航史, 角田雅照, 門田暁人, 松本健一. 工数見積もりモデルで予測できないソフトウェアプロジェクトの特徴分析. *電子情報通信学会技術研究報告. SS, ソフトウェアサイエンス*, Vol. 105, No. 491, pp. 67–72, 2005.
- [6] ISBSG. International software benchmarking standards group. <http://www.isbsg.org/>.
- [7] M. Shepperd and C. Schofield. Estimating software project effort using analogies. *IEEE Trans. on Software Engineering*, Vol. 23, No. 12, pp. 736–743, 1997.
- [8] J. Desharnais. *Analyse Statistique de la Productivite des Projets Informatique a Partie de la Technique des Point des Fonction*. PhD thesis, University of Montreal, 1989.
- [9] C. Mair, G. Kadoda, M. Lefley, K. Phalp, C. Schofield, M. Shepperd, and S. Webster. An investigation of machine learning based prediction systems. *Journal of Systems and Software*, Vol. 53, No. 1, pp. 23–29, 2000.
- [10] B. Kitchenham, S. Pfleeger, B. McColl, and S. Eagan. An empirical study of maintenance and development estimation accuracy. *Journal of Systems and Software*, Vol. 64, No. 1, pp. 57–77, 2002.
- [11] ISBSG Estimating. *Benchmarking and Research Suite Release 9*. International Software Benchmarking Standards Group (ISBSG), 2004.
- [12] S. Conte, H. Dunsmore, and Y. Shen. *Software Engineering Metrics and Models*. The Benjamin-Cummings Pub. Co., 1986.