修士学位論文

題目

見積もり研究における外的妥当性の調査を目的とした系統的レビューと追試

指導教員
楠本 真二 教授

報告者
江川　翔太

平成 28 年 2 月 9 日

大阪大学 大学院情報科学研究科
コンピュータサイエンス専攻

平成 27 年度 修士学位論文

見積もり研究における外的妥当性の
調査を目的とした系統的レビューと追試

江川　翔太

内容梗概

ソフトウェア工学における見積もりとは，ソフトウェア開発管理において，開発に必要な工数や費用，期間を予測することを指す．見積もりを行い開発計画を立案することで，円滑にプロジェクトを進行させることが可能になる．そのため，見積もりはプロジェクトの成功を左右する重要な要素として知られており，見積もりの精度を向上させるための研究が盛んに行われている．

一方で，ソフトウェア工学研究においては，得られた研究成果に外的妥当性が求められる．あるコンテキストで有意な結果を確認できたとしても，他のコンテキストにおいて同様の結果が確認できない場合，得られた結果には一般性がないと判断される．研究成果の外的妥当性を検証する方法として追試（replication study）がある．追試とは，研究成果の一般性を確認するため，実験の設定を部分的に変更して実験を再現することを指す．

以上の背景から本論文では，見積もり研究において，研究成果の外的妥当性がどの程度意識されているかを調査するため，過去 6 年間に発表された研究論文を対象に系統的レビュー（systematic review）を実施した．系統的レビューとは，網羅的で再現性のある文献調査手法である．系統的レビューを実施した結果，主要国際会議の論文集と主要論文誌に採録された論文 89 本の内，26 本が研究成果の妥当性について議論を行っていなかった．また，系統的レビューの結果から，外的妥当性検証の余地があると判断した既存研究について追試を実施した．追試の結果，既存研究の結果とは一部異なった結果が得られた．

主な用語

ソフトウェア工学
ソフトウェア見積もり
外的妥当性
系統的レビュー（systematic review）
追試（replication study）

目 次

図 目 次

表 目 次

1　まえがき

ソフトウェア工学における見積もりとは，ソフトウェア開発管理において，開発に必要な工数や費用，期間を予測することを指す．見積もりを行い開発計画を立案することで，円滑にプロジェクトを進行させることが可能になる．そのため，見積もりはプロジェクトの成功を左右する重要な要素として知られており [1–3]，見積もりの精度を向上させるための研究が盛んに行われている [4]．

一方で，ソフトウェア工学研究においては，得られた研究成果に外的妥当性が求められる．外的妥当性とは，ある研究から得られた成果を，違った母集団，状況，条件へ一般化し得る程度を指す [5]．外的妥当性は，一般化可能性とも言い換えられる．あるコンテキストで有意な結果を確認できたとしても，他のコンテキストにおいて同様の結果が確認できない場合，得られた結果には一般性がないと判断される．そのため，研究成果には高い外的妥当性が求められる [6]．

研究成果の外的妥当性を検証する方法として追試（replication study）がある．追試とは，研究成果の一般性を確認するため，実験の設定を部分的に変更して実験を再現することを指す [7]．追試を行い研究成果を再現することで，その研究成果が偶然の結果ではないということが証明できるため，ソフトウェア工学研究分野において追試は重要であると言える [8]．

以上の背景から本研究では，見積もり研究において，研究成果の外的妥当性がどの程度意識されているかを調査するため，過去 6 年間に発表された研究論文を対象に系統的レビュー（systematic review）を実施した．系統的レビューとは，網羅的で再現性のある文献調査手法である [9–11]．系統的レビューは主に，提案した仮説の検証や，若手研究者の補助を目的とした既存研究の要約を行うために実施される．本研究では，レビューを行うに当たって明らかにする Research Question を 3 つ設定した．1 つ目（RQ1）は，実験対象が単一である研究はどの程度存在するか，である．2 つ目（RQ2）は，研究成果の外的妥当性について議論を行っていない研究はどの程度存在するか，である．3 つ目（RQ3）は，外的妥当性の検証を目的とした追試はどの程度行われているか，である．主要国際会議の論文集と主要論文誌に採録された論文 89 本を対象に系統的レビューを実施した．RQ1 への回答として，実験対象として単一のデータセットのみを用いている研究論文は 10 本存在するという結果が得られた．RQ2 への回答として，研究成果の外的妥当性についての議論を行っていない研究論文が 26 本存在するという結果が得られた．RQ3 への回答として，外的妥当性の検証を目的として追試を実施した研究論文はわずか 2 本であるという結果が得られた．

更に本研究では，系統的レビューの結果から，研究成果の外的妥当性について検証の余地があると判断した既存研究に対する追試を行った．既存研究では，実験対象として単一のデータセットのみが用いられていたため，研究成果の外的妥当性について検証の余地がある

と判断した．追試の結果，既存研究の結果とは一部異なった結果が得られた．

以降，2 章では本研究において前提知識となる用語の解説を行う．3 章では研究動機についての説明を行う．4 章では系統的レビューの手順について詳細に説明を行う．5 章では系統的レビューを実施して得られた結果の提示と結果に対する考察を行う．6 章では追試の詳細についての説明を行う．7 章では本研究の妥当性への脅威についての説明を行う．最後に 8 章で本研究のまとめと今後の課題について述べる．

2　準備

本章では，本研究において前提知識となる用語の解説を行う．

2.1　ソフトウェア見積もり

ソフトウェア工学における見積もりとは，ソフトウェア開発管理において，開発に必要な工数や費用，期間を予測することを指す．見積もりの一般的な流れとしてはまず，開発対象ソフトウェアの規模計測を行う．規模計測の代表的な手法として，ファンクションポイント法（FP 法）[12] が挙げられる．次に，計測した規模と，開発を行うチームの生産性を基に，開発に必要な工数の見積もりを行う．最後に，見積もった工数を基に，開発にどの程度の人員を割り当てるかを決定し，人員数から費用を見積もる．以上が見積もりの一般的な流れである．

見積もりを行い開発計画を立案することで，円滑にプロジェクトを進行させることが可能になる．失敗した開発プロジェクトの内，31%が見積もりの誤りによるものであるということが過去の調査で判明している [1]．そのため，見積もりはプロジェクトの成功を左右する重要な要素として知られており，見積もりの精度を向上させるための研究が盛んに行われている [4]．

2.2　外的妥当性

外的妥当性とは，ある研究から得られた成果を，違った母集団，状況，条件へ一般化し得る程度を指す [5]．外的妥当性は，一般化可能性とも言い換えられる．例えば，ある手法をある実験対象 A に適用して得られた結果と，別の実験対象 B に適用して得られた結果が等しく有用である場合，その手法の外的妥当性は高いと言える逆に，あるコンテキストで有用な結果を確認できたとしても，他のコンテキストにおいて同様の結果が確認できない場合，得られた結果には一般性がないと判断される．そのため，研究成果には高い外的妥当性が求められる [6]．

2.3　系統的レビュー（systematic review）

系統的レビュー（systematic review）とは，網羅的で再現性のある文献調査手法である [9–11]． 系統的レビューでは明確な手順を踏んで調査を行うため，レビュー自体の再現性は高くなる．つまり，レビューの実行者に関わらす同様の結果が得られるはずであるため，レビュー結果の信頼性は高い．

元来，系統的レビューは医療の研究分野において，情報収集や調査を行う際によく用いられる手法である．ソフトウェア工学研究分野においては系統的レビューは主に，提案した仮

説の検証や既存研究の要約を行うために実施される．ソフトウェア工学研究分野における系統的レビューの手法は Kitchenham らにより提案されており [9]，この手法に従った系統的レビューが盛んに実施されている [13]．本研究で実施する系統的レビューも概ね Kitchenham らが提案した手法に従っている．見積もりの研究分野においては，機械学習を利用した見積もり手法に関する系統的レビュー [14] や類推見積もり手法に関する系統的レビュー [15] が行われている．

系統的レビューの具体的な手順については，4 章で詳細な説明を行う．

2.4 追試（replication study）

追試（replication study）とは，研究成果の一般性を確認するため，実験の設定を部分的に変更して実験を再現することを指す [7]．追試は，研究成果の外的妥当性検証を目的として実施される．追試を行い，研究のコンテクスト（実験の設定など）によって研究成果がどのように変動するのかを確かめることは重要な研究活動の 1 つである．また，追試を行い研究成果を再現することで，その研究成果が偶然の結果ではないということが証明できるため，ソフトウェア工学研究分野において追試は重要であると言える [8]．また，ソフトウェア工学研究に携わる研究者も追試の重要性を認識している [16]．

しかし，過去の調査から，ソフトウェア工学研究に携わる研究者は追試を敬遠しがちであることが判明している [16]．研究者が追試を敬遠する理由の 1 つに，追試を実施するために必要なコストが大きいことが挙げられる．実験手順の詳細が不明であったり，実験に用いられたツールやデータセットが入手困難であることが多々あるため，追試の準備と実施にはコストが必要になる．また，追試は画期的な結果を報告しないため，追試をまとめた論文は論文誌に採択されづらいという考えが研究者の間で根付いていることも理由として挙げられる．

3　研究動機

本章では，既存研究の紹介と研究動機についての説明を行う．

3.1　既存研究

既存研究として，ソフトウェア工学研究全般に対し，研究成果の妥当性に関する調査が行われている [16]．調査対象は国際会議である ICSE（International Conference on Software Engineering），ESEC/FSE（the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering） の論文集と，論文誌である EMSE（EMpirical Software Engineering）に採録された論文全 405 本である．論文の出版年代は ICSE が 2012-2013，FSE が 2011-2013，EMSE が 2011-2013 となっている．調査の結果，ソフトウェア工学研究に関する論文 381 本の内，177 本の論文が研究成果の妥当性についての議論を行っていないという事が判明した．また，追試を実施した論文が 32 本のみであるという事も判明した．この調査結果を受け Siegmund らは追加調査として，ソフトウェア工学研究者 79 人に対し，研究成果の妥当性に関する意識調査を行っている．調査の結果，ソフトウェア工学研究分野において追試をより実施すべきであるという考えを持つ研究者は 79 人中 69 人存在する事が判明した．

また，ソフトウェア工学研究に関する国際会議で発表された研究に対しても同様に，研究成果の妥当性に関する調査が行われている [6]．調査対象は国際会議である ESEM（Empirical Software Engineering and Measurement）2009 の論文集に採録された論文 43 本である．調査の結果，43 本の論文の内，9 本の論文が研究成果の妥当性についての議論を行っていないという事が判明した．

上述した既存研究の結果から，ソフトウェア工学研究に携わる研究者の一部は，研究成果の外的妥当性に対する意識が希薄であることがわかる．そこで私は，現在盛んに研究が行われている見積もりの研究分野について，研究成果の外的妥当性に関する系統的レビューを行い，近年の動向，つまり，見積もり研究に携わる研究者は研究成果の外的妥当性をどの程度意識しているかを明らかにしようと考えた．

3.2　既存研究との相違点

既存研究と本研究の相違点は 2 つ存在する．

まず，既存研究の調査対象はいずれもソフトウェア工学全般に関する研究論文である事が相違点として挙げられる．本研究では，対象を見積もりに関する研究に絞り，見積もりの分野において研究成果の外的妥当性がどのように意識されているのかを明らかにする．

また，既存研究はいずれも系統的レビューを行っていないという事が相違点として挙げられる．本研究では Kitchenham らが提案した系統的レビューの手法 [9] に従い，見積もりに関する研究論文を網羅的に調査することを目的とする．系統的レビューの手法に従うことでレビューの再現性が高まり，調査結果の信頼性も高まる．

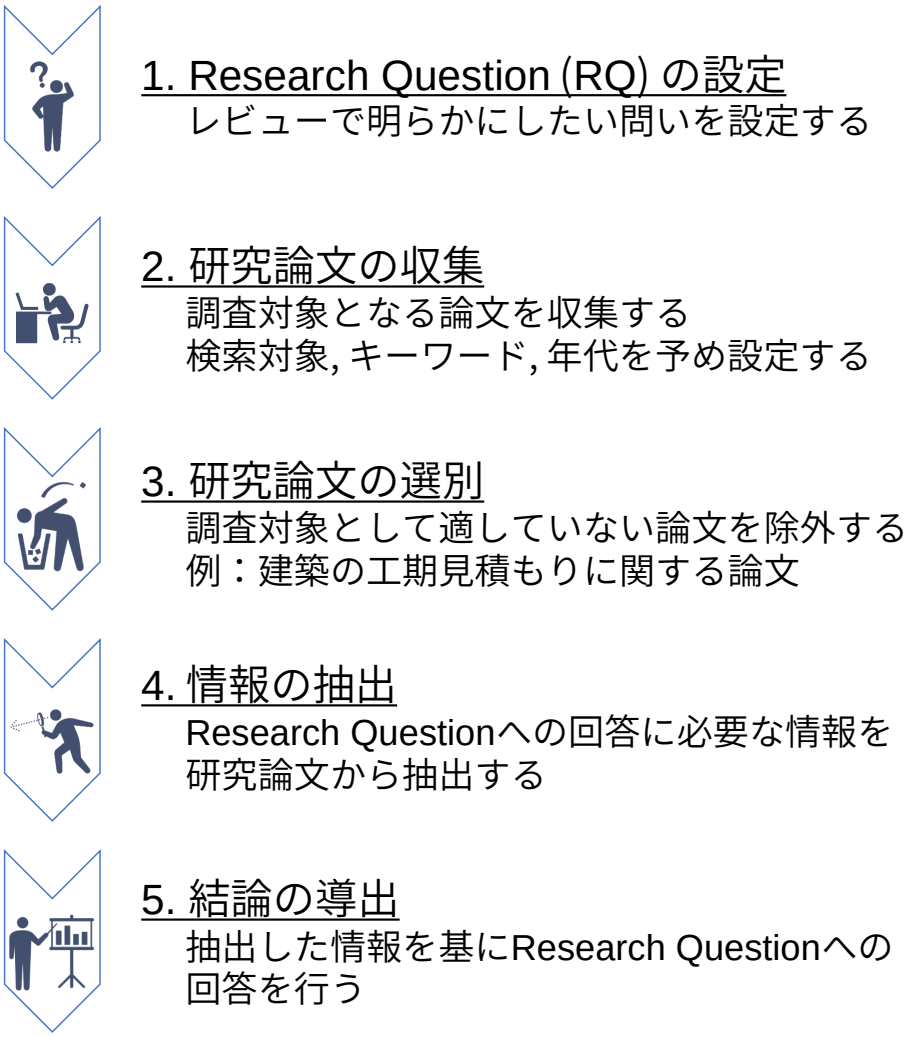

図 1: レビュー手順

4　レビュー手法

本章では，系統的レビューの手順について詳細に説明を行う．

本研究では，Kitchenham らが提案した手法に従い系統的レビューを行う．レビュー手順を図 1 に示す．

本研究で行う系統的レビューは，Research question の設定，研究論文の収集，研究論文の選別，情報の抽出，結論の導出の 5 つの手順から成る．以降，各手順について説明を行う．

4.1 Research question の設定

この手順では，系統的レビューを行うにあたって明らかにしたい問いを Research question（RQ）として設定する．本研究では，研究成果の外的妥当性がどのように意識されているのかを明らかにするために，以下 3 つの Research question を設定した．

- RQ1：実験対象が単一である研究はどの程度存在するか
- RQ2：研究成果の外的妥当性について議論を行っていない研究はどの程度存在するか
- RQ3：外的妥当性の検証を目的とした追試はどの程度行われているか

4.1.1 節で RQ1 についての説明を，4.1.2 節で RQ1 についての説明を，4.1.3 節で RQ1 についての説明を行う．

4.1.1 RQ1

実験対象が単一の研究とは，実験のために用いたデータセットが 1 種類のみである研究を指す．見積もり研究では通常，新たな見積もり手法を考案した場合，その性能を評価するため，実際の開発のデータを収集したデータセットに手法を適用する．手法により予測した工数と実際の工数の差が小さいほど，精度が高い見積もり手法であると言える．手法の外的妥当性を高めるためには，手法を複数のデータセットに適用することが望まれる．本研究では，実験対象が単一である研究を外的妥当性検証の余地がある研究と見なし，そのような研究がどの程度存在するかを調査する．

4.1.2 RQ2

研究成果の外的妥当性についての議論とは，実験対象についての分析や，提案手法の外的妥当性検証の必要性に関する記述を指す．この Research Question により，研究成果の外的妥当性を意識していない研究者がどの程度存在するかが明らかとなる．

4.1.3 RQ3

外的妥当性の検証を目的とした追試とは，既存研究の実験設定を一部変更し，実験の再現を行った研究を指す．この Research Question により，研究成果の外的妥当性の検証がどの程度行われているかが明らかとなる．

4.2 研究論文の収集

この手順では，調査対象となる研究論文の収集を行い，調査対象論文の候補とする．収集の際には予め，検索する期間と検索対象データベース，検索に用いるキーワードを設定する．検索する期間は過去 6 年間（2010 年 1 月から 2015 年 10 月まで）とした．以降の小節で検索対象データベースと検索に用いるキーワードについて説明を行う．

4.2.1 検索対象データベース

検索対象としたデータベースは以下の 5 つの電子データベースである．

- IEEE xplore [17]
- ACM Digital library [18]
- Science Direct [19]
- SpringerLink [20]
- Google Scholar [21]

見積もりの研究分野において著名な論文誌や国際会議の論文集に採録されている研究論文が網羅されているため，上記データベースを検索対象として選択した．

4.2.2 検索キーワード

Kitchenham らの提案手法 [9] と過去の系統的レビュー [22] に倣い，以下の手順を踏んで検索キーワードを設定した．

1. 見積もりの研究に関連する単語を選別する
2. 選別した単語の類義語（同意語）を確認する
3. OR 演算子で類義語を連結する
4. 類義語を連結した節を AND 演算子で連結する

設定したキーワードを以下に示す．

```
(software OR system OR application OR product OR project OR development)
 AND (effort OR cost OR resource) AND (estimation OR prediction)
```

Google Scholar 以外の 4 データベースに対しては，タイトル，内容梗概，キーワードのいずれかに上記のキーワードが含まれる論文を出力するよう検索オプションを設定したが，Google Scholar に対してはタイトルにのみ上記キーワードが含まれる論文を出力するよう検索オプションを設定した．これは，内容梗概とキーワードに上記キーワードが含まれる論文を出力するよう設定すると，膨大な量の無関係な論文が出力されるためである [14].

4.3 研究論文の選別

この手順では，各検索対象データベースから収集した候補の選別を行い，調査対象とする研究論文を決定する．選別の手順を以下に示す.

1. 各検索対象データベースから出力された論文のタイトルを確認し，重複している論文を候補から取り除く

2. 各論文のタイトル，内容梗概，キーワードを確認し，ソフトウェアの見積もりに関連しない論文（建築の工期見積もりに関する論文など）を候補から取り除く

3. 各論文の出典を確認し，CORE RankingsPortal [23] においてランクが B 以下である論文誌または国際会議の論文集に採録されている論文を候補から取り除く

4. 候補に含まれる各論文の参考文献を確認し，ソフトウェアの見積もりに関連し，出典が CORE Rankings Portal においてランクが A 以上である論文誌または 国際会議の論文集に採録されている論文であり，かつ，研究論文の収集の段階で各データベースから出力されなかった論文を候補に加える

5. 候補を調査対象論文として選択する

Kitchenham らの提案手法 [9] では本来，上記手順 3 の段階において候補となっている論文の本文を確認し，論文の質の評価を行う.

質の評価は，評価項目を設定し，評価項目ごとに 3 段階で評価し点数を付けることで行われる．評価項目を完全に満たす場合は yes として 1 点，部分的に満たす場合は Partly として 0.5 点，満たさない場合は No として 0 点を付ける．このようにして付けられた点数を合計し，合計点数が項目数× 0.5 点以下となった論文を取り除く．例えば論文 A と B を，評価項目を 4 つ設けて質を評価した場合を考える．A の質の評価合計点が 2.0 点，B の質の評価合計点が 3.0 点となったと仮定する．この場合，評価項目は 4 つなので，4*0.5 = 2.0 点以下の論文が候補から取り除かれる．よって論文 A は候補から除かれ，論文 B は候補に残る．

Kitchenham らの提案手法 [9] に従った系統的レビューでは，上記のように質の評価を行っている系統的レビューが主だが，質の評価を上記手順 3 のように，論文の出典で質の評価を

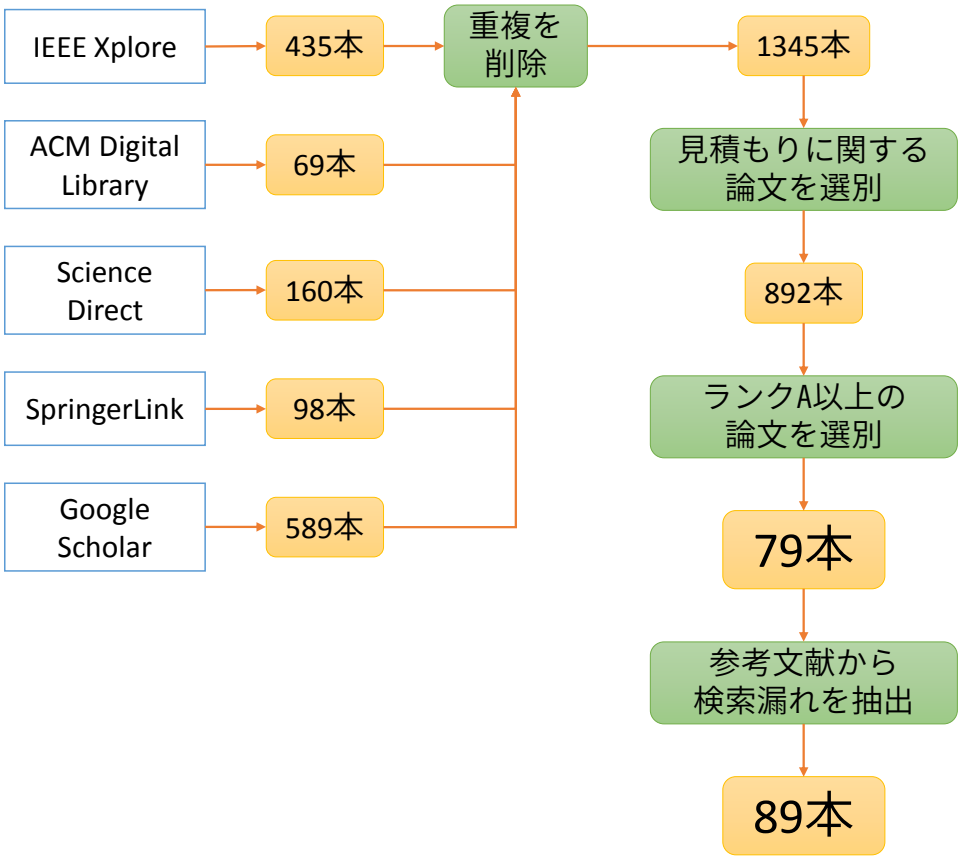

図 2: 収集と選別の結果

行っている系統的レビューもいくつか存在する [4]．本研究ではそのような論文に倣い，質の評価を論文の出典で行っている．

研究論文を収集し，上記選別手順を実行した結果を図 2 に示す．

収集した候補から重複を取り除いたところ，1345 本の論文が候補として得られた．1345 本の候補からソフトウェア見積もりに関連しない論文を取り除いたところ，892 本が候補として残った．892 本の候補から CORE RankingsPortal [23] においてランクが B 以下である論文誌または国際会議の論文集に採録されている論文を候補から取り除いたところ，79 本が候補として残った．79 本の候補に含まれる各論文の参考文献を確認し，ソフトウェアの見積もりに関連し，出典が CORE Rankings Portal においてランクが A 以上である論文誌または国際会議の論文集に採録されている論文であり，かつ，研究論文の収集の段階で各データベースから出力されなかった論文を候補に加えたところ，89 本が調査対象として選択された．調査対象とした論文のタイトルと著者名，出典名，出版年を「付録 A　調査対象論

文」に示す.

4.4 情報の抽出

この手順では，調査対象とした研究論文の本文を確認し，論文に関する基本的な情報とResearch Question への回答に必要な情報を抽出する．論文に関する基本的な情報とは，タイトル，著者名，出典名，出版年である．抽出する情報を表 1 に示す.

すべての調査対象論文からすべての情報を抽出できるわけではない．例えば，実験対象にデータベースを用いていない論文（被験者実験やインタビューを行っている論文）や，研究成果の外的妥当性について議論を行っていない論文の存在が考えられる．このような論文に関しては，情報が抽出できなかった旨を記録する.

4.5 結論の導出

この手順では，抽出した情報を整理・考察した上で，Research Question への回答を行う．個々の調査対象論文から抽出した情報を整理して組み合わせることで，Research Question への回答に説得力を持たせることが可能となる [24]．Research Question 1 への回答のために

表 1: 抽出する情報

・タイトル
・著者名
・出典名
・出版年
・RQ1 への回答に必要な情報
- 実験対象にデータセットを用いているか
- 実験対象データセット数
- 実験対象
・RQ2 への回答に必要な情報
- 研究成果の外的妥当性に関する記述の有無
- 研究成果の外的妥当性が独立した章で議論されているか
・RQ3 への回答に必要な情報
- 研究が追試であることを示す記述

は，実験対象データベース数が 1 である研究論文の数を集計する必要がある．また，Research Question 2 への回答のためには，研究成果の外的妥当性に関する議論が行われていない研究論文の数を集計する必要がある．そして，Research Question 3 への回答のためには，研究が追試であることをしめす記述が含まれている研究論文の数を集計する必要がある．各集計結果と Research Question への回答，また，その結果を受けた考察については 5 章で説明を行う．

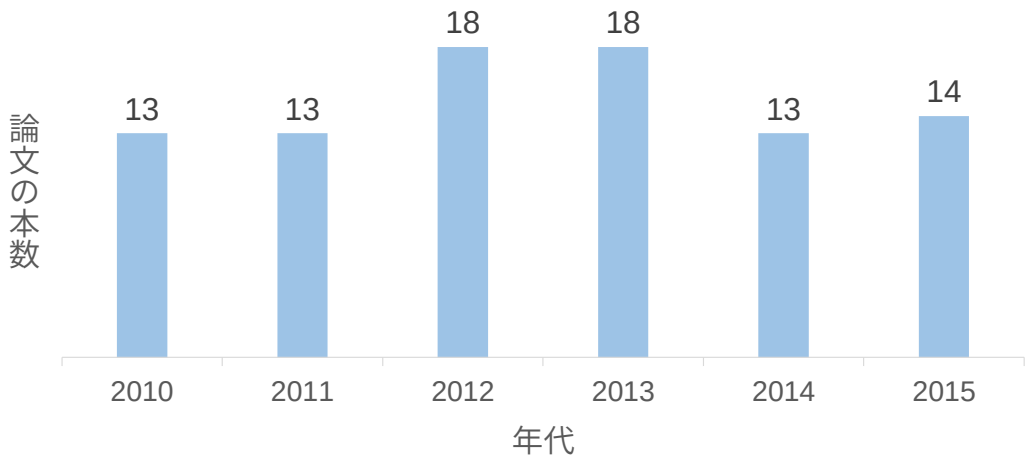

図 3: 年代ごとの論文数

5　レビュー結果と考察

本章では，系統的レビューを実施して得られた結果の提示と結果に対する考察を行う．

5.1　調査対象論文の概観

4 章で説明した通り，論文の収集と選別を行った結果 89 本が調査対象論文として選択された．各調査対象論文のタイトル，著者名，出典名，出版年を「付録 A　調査対象論文」に示す．また，各調査対象論文から抽出した情報を「付録 B　調査対象論文から抽出した情報」に示す．出版された年代ごとの調査対象論文数を図 3 に示す．年代ごとの出版数に大きな偏りは見られず，一定数の見積もり研究に関する論文が出版されていることから，見積もりに関する研究が盛んに行われていることがわかる．

以降の小節で抽出した情報の分析と，各 Research Question に対する回答を行う．

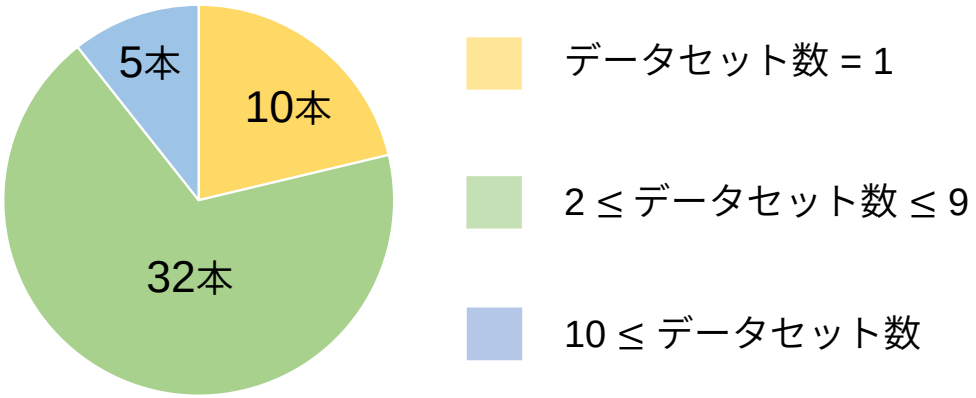

図 4: 実験対象データセット数の分布

5.2 RQ1 への回答

調査対象論文が実験対象としたデータセット数の分布を図 4 に示す．集計は以下に示す区分を用いて行った．

- 実験対象としたデータセット数が 1
- 実験対象としたデータセット数が 2〜9
- 実験対象としたデータセット数が 10 以上

実験対象としてデータセットを用いている研究論文は 47 本存在した．集計の結果，実験対象としたデータセット数が 1 である論文は 10 本，実験対象としたデータセット数が 2〜9 である論文は 32 本，実験対象としたデータセット数が 10 以上である論文は 5 本であった．実験対象としたデータセット数は最大で 12，実験対象としたデータセット数の平均は約 4.5 であった．

上記より，RQ1 : 実験対象が単一である研究はどの程度存在するか，への回答は 47 本中 10 本（21%）となる．つまり，研究成果に外的妥当性検証の余地がある研究は 10 件存在する．

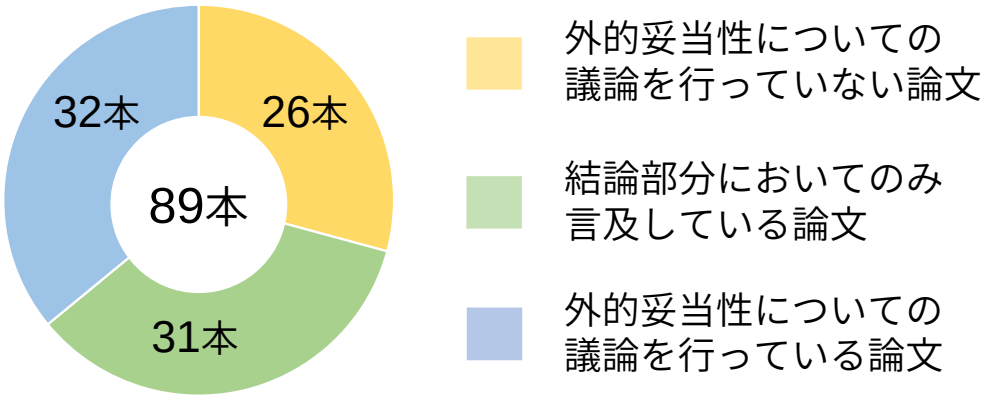

図 5: 外的妥当性についての議論の有無

5.3 RQ2 への回答

調査対象論文中における外的妥当性についての議論の有無の分布を図 5 に示す．集計は以下に示す区分を用いて行った．

- 研究成果の外的妥当性について議論を行っていない
- 研究成果の外的妥当性について，論文中の結論部分においてのみ言及している
- 研究成果の外的妥当性について議論を行っている

集計の結果，研究成果の外的妥当性について議論を行っていない論文は 26 本，研究成果の外的妥当性について，論文中の結論部分においてのみ言及している論文は 31 本研究成果の外的妥当性について議論を行っている論文は 32 本であった．

上記より，RQ2 : 研究成果の外的妥当性について議論を行っていない研究はどの程度存在するか，への回答は 89 本中 26 本（29%）となる．

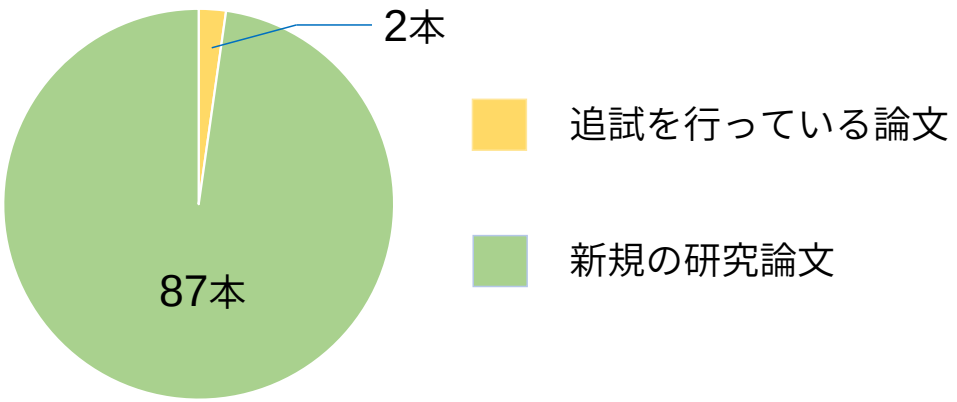

図 6: 新規研究の論文数と追試を行っている論文数の分布

5.4 RQ3 への回答

新規研究の論文数と追試を行っている論文数の分布を図 6 に示す．集計は以下に示す区分を用いて行った．

- 追試を行っている論文
- 新規の研究を行っている論文

集計の結果，追試を行っている論文は 2 本，新規の研究を行っている論文は 87 本であった．

上記より，RQ3 : 外的妥当性の検証を目的とした追試はどの程度行われているか，への回答は 89 本中 2 本（2%）となる．

5.5 見積もり研究における外的妥当性への対応

この小節では，各 Research Question への回答結果から，見積もりの分野において研究成果の妥当性を高めることを目的とした，研究者に対する勧告を考察する．

5.5.1 多様なデータセットの使用

実験を行う際には，可能な限り多様なデータセットを用いるべきである．RQ1 に対し，実験対象が単一である研究は 10 件という回答が得られた．10 件中 3 件については，提案手法が適用可能なドメインが限定されている（Web アプリケーションの開発プロジェクト）ため，複数のデータセットを用意して実験を行うのは困難であると予想される．しかし，残り 7 件についてはドメインが特に限定されていない．見積もりの研究分野においては，実験対象として自由に利用可能なデータセット [25] がインターネット上で公開されている．また，有料であるデータセット [26] も存在する．上記データセットを用いて可能な限り多様なデータセットに手法を適用し，手法の外的妥当性を高めるべきである．実験対象に複数のデータセットを用いている研究についても同様である．

また，ソフトウェア工学の研究分野において，実験対象の多様性を評価する手法が提案されている [27]．この手法を利用することで，実験対象の多様性を保持したまま，提案手法を適用するデータセット数を絞ることができるため，より少ないコストで研究成果の外的妥当性を高めることが可能になると考えられる．

5.5.2 外的妥当性についての詳細な議論

手法の外的妥当性についての議論はより詳細に行うべきである．RQ2 に対し，研究成果の外的妥当性について，議論を行っていない論文は 26 本存在し，論文中の結論部分においてのみ言及している論文は 31 本存在するという回答が得られた．論文中の結論部分のみにおいて言及している論文の中には，「更なるデータセットへの適用が必要になる」とだけ議論し，実験対象についての分析や提案手法が適用可能なドメインについての記述を行っていない論文が存在する．研究成果の外的妥当性についての議論は後続の研究者が追試を行う上での助けになるため，詳細に議論するべきであると考える．

5.5.3 積極的な追試の実施

研究成果の外的妥当性検証を目的とした追試は積極的に行うべきである．過去の調査 [16] から，ソフトウェア工学研究に携わる研究者は，追試の重要性を認識してはいるものの，追試を敬遠しがちであることが判明している．これは，追試を行っている論文が 89 本中 2 本しか存在しないという，RQ3 の回答からも明らかである．追試は基本的に画期的・新規的な結果を報告しないため，追試についての論文は論文誌や国際会議の論文集に採択されにくいという考え研究者の間で根付いており，その考えが，研究者が追試を敬遠する主な理由となっている．また，実験結果の詳細が論文中に記述されていなかったり，実験で用いられたツールやデータセットが入手困難であるなどの理由で，追試を実施するために必要となるコ

ストが高いことも理由として挙げられる．しかし，追試を行い研究成果を再現することで，その研究成果が偶然の結果ではないということが証明できる．つまり，研究成果の外的妥当性を高めることが可能であるため，見積もりの分野においても，追試は非常に重要な研究活動であると言える．

6 追試

本章では，追試の詳細についての説明を行う．

6.1 対象とする研究

追試の対象とした論文 [28] のタイトル，著者名，出典名，実験対象，「付録 A　調査対象論文」における番号を表 2 に示す．以降の小節で研究の概要，実験手順，実験結果について説明を行う．

6.1.1 概要

Myrtveit らは，見積もり手法の精度を評価する指標の信頼性を調査するために実験を行った．調査対象とした評価指標は MRE [30]，MER [31]，BRE [32]，IBRE [32] である．各評価指標の詳細については次小節で説明を行う．実験の結果，上記の評価指標は精度が低い手法を，精度が高い手法よりも優れていると評価する場合が存在した．この結果から Myrtveit らは，上記の評価指標を用いて見積もり手法の精度の評価を行うのは妥当ではないと結論付けている．

Myrtveit らの研究を追試の対象とした理由は 2 つある．1 つ目は，実験対象が単一のデータセットであり，研究成果に外的妥当性検証の余地があると判断したためである．Myrtveit らは実験対象として COCOMO81 データセット [29]（63 プロジェクト）のみを用いていたため，研究成果に外的妥当性検証の余地があると判断した．2 つ目は，実験手順の詳細が明らかであり，かつ，実験に必要なツールの入手が容易なためである．つまり，追試にかかるコストが比較的小さい．Myrtveit らの実験手順は Foss らの先行研究 [33] に倣っており，Foss らの論文には実験手順が詳細に記述されている．

表 2: 追試の対象とした論文

タイトル	Validity and reliability of evaluation procedures in comparative studies of effort prediction models
著者名	Ingunn Myrtveit Erik Stensrud
出典名	Empirical Software Engineering Vol.17, No. 1-2, pp.23-33, 2012
実験対象	COCOMO81 データセット [29]
付録 A における番号	[73]

6.1.2 評価指標

式中の実測値は工数の予測対象としたプロジェクトが実際に必要となった工数を指し，予測値は見積もり手法によって予測された工数の値を表す．評価指標の値が小さい，つまり，手法が予測した工数と実際の工数の差が小さいほど，その手法は精度が高い．

MRE（Magnitude of Relative Error）

MRE（Magnitude of Relative Error）は見積もり手法の精度を評価する際に最も一般的に用いられている評価手法である [30]．MRE は以下の式で求められる．

$$MRE = \frac{|\text{実測値} - \text{予測値}|}{\text{実測値}} \tag{1}$$

Foss らの研究 [33] により，MRE は過小見積もりを行う手法，つまり，実測値よりも小さい予測値を出力する手法の精度を高く評価する傾向があると判明している．例えば，実工数の 2 倍の値を予測値として出力する手法 A と，実工数の $\frac{1}{20}$ の値を予測値として出力する手法 B という 2 つの手法が存在すると仮定する．直感的に，手法 A の見積もり精度は手法 B の見積もり精度よりも高い．予測された工数の値を用いてプロジェクトに必要な費用を見積もることを考慮すると，手法 B の見積もり精度は手法 A の見積もり精度よりも低いことは自明である．しかし，各手法を MRE で評価した場合，手法 A の MRE の値は 1.0，手法 B の MRE の値は 0.95 となり，手法 A の見積もり精度よりも手法 B の見積もり精度が高いと判定される．このように，MRE は過小見積もりを行う手法の精度を高く評価する傾向がある．

MER（Magnitude of Error Relative to the estimate）

MER（Magnitude of Error Relative to the estimate）は Kitchenham らが提案した見積もり精度評価指標である [31]．MER は以下の式で求められる．

$$MER = \frac{|\text{実測値} - \text{予測値}|}{\text{予測値}} \tag{2}$$

MER は実測値と予測値の比の散布度を示す．

BRE（Balanced Relative Error）

BRE（Balanced Relative Error）は Miyazaki らが提案した見積もり精度評価指標である [32]．BRE は以下の式で求められる．

$$BRE = \begin{cases} \frac{(\text{予測値}-\text{実測値})}{\text{実測値}} & \text{予測値} - \text{実測値} \geq 0 \\ \frac{|\text{予測値}-\text{実測値}|}{\text{予測値}} & \text{予測値} - \text{実測値} < 0 \end{cases}$$

予測値が実測値より大きい場合と小さい場合で求める式が異なる．予測値が実測値より大きい場合はMREを求める式と等価だが，予測値が実測値より小さい場合は実測値と予測値の差の絶対値を予測値で除する．式の場合分けにより，MREよりも過小見積もり手法を正確に評価できるとされる．

IBRE（Inverted Balanced Relative Error）

IBRE（Inverted Balanced Relative Error）はMiyazakiらが提案した見積もり精度評価指標である [32]． IBREは以下の式で求められる．

$$IBRE = \begin{cases} \frac{|\text{予測値}-\text{実測値}|}{\text{実測値}} & \text{予測値} - \text{実測値} < 0 \\ \frac{(\text{予測値}-\text{実測値})}{\text{予測値}} & \text{予測値} - \text{実測値} \geq 0 \end{cases}$$

IBREは値が0から1の範囲に収まるため，平均を取る際に外れ値，つまり，大幅に誤った予測値の影響を受けにくいとされる．

6.1.3 実験手順

Myrtveitらが行った実験の手順は，Fossらの先行研究 [33] に倣っている．

まず，COCOMO81データセットに収録された63データの工数とソフトウェアの規模の値に対し，底を自然対数のと底して対数変換を行う．次に，工数を対数変換した値を説明変数，規模を対数変換した値を従属変数として回帰分析を行う．回帰分析手法は最小二乗法である．その後，回帰分析によって得られた式に対し指数変換を行い，工数の予測モデルを得る．予測モデルは以下のような式となる．

$$\text{工数} = \mathrm{e}^{a} * (\text{規模})^{b} \tag{3}$$

Myrtveitらは，以上の式を正確な見積もりを行うモデル（精度が高いモデル）と定義した．

次に，上記のモデルに規模の値を30個与え，工数の値を30個出力させる．その後，出力された工数の値に対し乱数を付加（加算もしくは減算）し，30個の工数と規模の値の組を作成する．この30個の組を擬似的に，30プロジェクト分のデータを持つ仮想データセットとみなす．Myrtveitらは，付加する乱数を変更して同様の作業を繰り返し，仮想データセットを1000個作成した．この仮想データセットはCOCOMO81データセットと似た特徴を持つ．データセットを直接の実験対象とせず仮想データセットを作成した理由は，1000個という多数のデータセットを実験対象とすることで，得られる研究成果の信頼性を高めるためである．ここでMyrtveitらは，式(3)中のaとbの値を操作し，4つのモデルを作成した．具体的には，aの値を小さくし，大幅な過小見積もりを行うモデルと，bの値を小さくし，軽

度な過小見積もりを行うモデルと，a の値を大きくし，大幅な過大見積もりを行うモデルと，b の値を大きくし，軽度な過大見積もりを行うモデルを作成した．

Myrtveit らは以上の 4 つのモデルを，誤った見積もりを行うモデル（精度が低いモデル）として定義した．以降，大幅な過小見積もりを行うモデルを誤りモデル 1，軽度な過小見積もりを行うモデルを誤りモデル 2，軽度な過大見積もりを行うモデルを謝りモデル 3，大幅な過大見積もりを行うモデルを誤りモデル 4 と表記する．

その後，仮想データセットが持つ工数の値を実測値，実工数と組になった規模の値を各モデルに与え出力された値を予測値とし，前述した 4 つの評価指標を用いて，正確なモデルと各誤りモデルの精度を比較し，どちらが精度が高いモデルであるかを判定する．30 個のデータについてそれぞれ評価指標の値を求め，それらの平均をとった値が低い方のモデルを精度が高いモデルとみなす．判定は 1000 個のデータセットごとに行い，より多くのデータセットで精度が高いと判定されたモデルを，その評価指標によって精度が高いと判定されたモデルとする．

6.1.4 実験結果

正確なモデルと誤りモデル 1 について，各評価指標を用いてどちらの精度が高いかを判定した結果を表 3 に，正確なモデルと誤りモデル 2 について，各評価指標を用いてどちらの精度が高いかを判定した結果を表 4 に，正確なモデルと誤りモデル 3 について，各評価指標を用いてどちらの精度が高いかを判定した結果を表 5 に，正確なモデルと誤りモデル 4 について，各評価指標を用いてどちらの精度が高いかを判定した結果を表 6 に示す．

表 3: 誤りモデル 1 との精度比較

評価指標	正確なモデル	誤りモデル 1
MRE		○
MER	○	
BRE	○	
IBRE	○	

表 4: 誤りモデル 2 との精度比較

評価指標	正確なモデル	誤りモデル 2
MRE		○
MER	○	
BRE		○
IBRE		○

表 5: 誤りモデル 3 との精度比較

評価指標	正確なモデル	誤りモデル 3
MRE	○	
MER		○
BRE	○	
IBRE	○	

表 6: 誤りモデル 4 との精度比較

評価指標	正確なモデル	誤りモデル 4
MRE	○	
MER		○
BRE	○	
IBRE	○	

表中の ◯ は，評価指標が，そのモデルの精度が他方より高いと判定したことを示す．表 3 より，MRE は正確なモデルよりも誤りモデル 1，つまり，大幅な過小見積もりを行うモデルの精度が高いと判定することがわかる．また，表 4 より，MRE，BRE，IBRE は正確なモデルよりも誤りモデル 2，つまり，軽度な過小見積もりを行うモデルの精度が高いと判定することがわかる．一方で表 5 と表 6 より，MER は正確なモデルよりも誤りモデル 3 と誤りモデル 4，つまり，過大見積もりを行うモデルの精度が高いと判定することがわかる．

以上の結果から Myrtveit らは，上記 4 つの評価指標を用いて見積もり手法の精度の評価を行うのは妥当ではないと結論付けている．

6.2 使用するデータ

本節では，追試に用いるデータセットについての説明を行う．

6.2.1 ISBSG データセット

ISBSG データセット [26] とは，ISBSG（The International Software Benchmarking Standards Group）が世界 24 か国に存在する組織・企業から実開発のデータを収集し，整理したデータセットである．開発工数やソフトウェアの規模，開発言語等のデータが 5,052 プロジェクト分収録されている．このデータセットに収録されたデータの選別を行い，追試に用いるデータの抽出を行った．データの選別基準を表 7 に示す．

表中の Data Rating は，データセットに収録されたデータの信頼性を示す．ISBSG データセットにおいて，各データの信頼性が A,B,C,D の 4 段階で評価される．A と評価されたデータは最も信頼性が高く，D と評価されたデータは最も信頼性が低い．見積もりの研究分野においては ISBSG データを利用する際，データの信頼性が A または B と評価されたデータを利用するべきであるとされる [34, 35]．

表中の UFP Rating は，データセットに収録されたデータ中の，ソフトウェアの規模を表す指標である未調整ファンクションポイント（Unadjusted Function Point）[12] の値の信頼性を示す．各値の信頼性は A,B,C,D の 4 段階で評価される．A と評価されたデータは最も信頼性が高く，D と評価されたデータは最も信頼性が低い．ISBSG データ中の未調整ファ

表 7: ISBSG データの選別基準

Data Rating（データの信頼性）	A もしくは B
UFP Rating（ソフトウェア規模の信頼性 ）	A もしくは B
Function Size Metric Used（規模計測手法）	IFPUG3 もしくは IFPUG4
Resource Level（工数の計測対象部分）	1（開発自体に必要とした工数）

ンクションポイントの値を利用する際，信頼性が A または B と評価されたデータを利用するべきであるとされる [34, 35].

表中の Function Size Metric Used は，未調整ファンクションポイントを計測する手法を示す．ファクションポイントの計測方法には IFPUG 法，COSMIC 法などが存在する．追試では，Myrtveit らが選択した実験対象のファンクションポイント計測手法と同様に，IFPUG 法（Version 3 もしくは Version 4）で計測されたデータを用いる．

表中の Resource Level は，工数を計測する際に対象とした作業区分を示す．Resource Level には 1（開発に必要とした工数），2（開発と品質向上作業に必要とした工数），3（開発，品質向上作業，保守作業に必要とした工数），4（開発，品質向上作業，保守作業，ユーザの教育に必要とした工数）という 4 つの区分が存在する．追試では，Myrtveit らが選択した実験対象の工数計測対象部分と同様に，Resource Level が 1 となるデータを用いる．

以上の選別基準に従いデータの選別を行い，追試の対象とする研究が用いたデータ項目（工数とソフトウェアの規模）が欠損していないデータを抽出した結果，追試に利用可能なデータとして 2,399 プロジェクトデータが抽出された．

6.2.2 IPA/SEC データセット

IPA/SEC データセット [36] とは，独立行政法人情報処理推進機構が日本に存在する組織・企業から実開発のデータを収集し，整理したデータセットである．開発工数やソフトウェアの規模，開発言語等のデータが 3,541 プロジェクト分（2014-2015 年版）収録されている．データの信頼性による選別を除き，ISBSG データセットと同様の選別を行った結果，追試に利用可能なデータとして 99 プロジェクトデータが抽出された．

6.2.3 某社データセット

某社データセットとは，金融保険業で用いるソフトウェアの開発を行う某社から提供を受けたデータセットである．開発工数やソフトウェアの規模，開発言語等のデータが 113 プロジェクト分収録されている．データの信頼性による選別を除き，ISBSG データセットと同様の選別を行った結果，追試に利用可能なデータとして 48 プロジェクトデータが抽出された．

6.3 追試の結果

この小節では，各データセットに対して，前述した手順に従い実験を行い得られた結果について説明する．

6.3.1 ISBSG データセット

正確なモデルと誤りモデル 1 について，各評価指標を用いてどちらの精度が高いかを判定した結果を表 8 に，正確なモデルと誤りモデル 2 について，各評価指標を用いてどちらの精度が高いかを判定した結果を表 9 に，正確なモデルと誤りモデル 3 について，各評価指標を用いてどちらの精度が高いかを判定した結果を表 10 に，正確なモデルと誤りモデル 4 について，各評価指標を用いてどちらの精度が高いかを判定した結果を表 11 に示す．

表中の数字は，そのモデルの精度が他方より高いと評価指標が判定した回数を示す．表 8 より，MRE は正確なモデルよりも誤りモデル 1，つまり，大幅な過小見積もりを行うモデルの精度が高いと判定することがわかる．また，表 9 より，MRE，BRE，IBRE は正確なモデルよりも誤りモデル 2，つまり，軽度な過小見積もりを行うモデルの精度が高いと判定することがわかる．一方で表 10 と表 11 より，MER は正確なモデルよりも誤りモデル 3 と誤りモデル 4，つまり，過大見積もりを行うモデルの精度が高いと判定することがわかる．つまり，ISBSG データセットを用いて Myrtveit らの研究の追試を行ったところ，Myrtveit らの研究と同様の結果が得られた．

表 8: 誤りモデル 1 との精度比較

評価指標	正確なモデル	誤りモデル 1
MRE	0	1,000
MER	998	2
BRE	110	890
IBRE	222	778

表 9: 誤りモデル 2 との精度比較

評価指標	正確なモデル	誤りモデル 2
MRE	0	1,000
MER	959	41
BRE	8	992
IBRE	133	867

表 10: 誤りモデル 3 との精度比較

評価指標	正確なモデル	誤りモデル 3
MRE	1,000	0
MER	235	765
BRE	1,000	0
IBRE	983	17

表 11: 誤りモデル 4 との精度比較

評価指標	正確なモデル	誤りモデル 4
MRE	1,000	0
MER	115	885
BRE	999	1
IBRE	939	61

6.3.2 IPA/SEC データセット

正確なモデルと誤りモデル 1 について，各評価指標を用いてどちらの精度が高いかを判定した結果を表 12 に，正確なモデルと誤りモデル 2 について，各評価指標を用いてどちらの精度が高いかを判定した結果を表 13 に，正確なモデルと誤りモデル 3 について，各評価指標を用いてどちらの精度が高いかを判定した結果を表 14 に，正確なモデルと誤りモデル 4 について，各評価指標を用いてどちらの精度が高いかを判定した結果を表 15 に示す．

表中の数字は，そのモデルの精度が他方より高いと評価指標が判定した回数を示す．表 12 と表 13 より，MRE，BRE，IBRE は正確なモデルよりも誤りモデル 1 と誤りモデル 2，つまり，過小見積もりを行うモデルの精度が高いと判定することがわかる．一方で表 14 と表 15 より，MER は正確なモデルよりも誤りモデル 3 と誤りモデル 4，つまり，過大見積もりを行うモデルの精度が高いと判定することがわかる．つまり，IPA/SEC データセットを用いて Myrtveit らの研究の追試を行ったところ，Myrtveit らの研究とほぼ同様の結果が得られた（Myrtveit らの研究結果と異なる点は，IPA/SEC データセットと似た特徴を持つデータセットにおいて，BRE と IBRE が大幅な過小見積もりを行うモデルの精度が高いと判定する点である）．

表 12: 誤りモデル 1 との精度比較

評価指標	正確なモデル	誤りモデル 1
MRE	0	1,000
MER	1,000	0
BRE	53	947
IBRE	184	816

表 13: 誤りモデル 2 との精度比較

評価指標	正確なモデル	誤りモデル 2
MRE	0	1,000
MER	992	8
BRE	11	989
IBRE	124	876

表 14: 誤りモデル 3 との精度比較

評価指標	正確なモデル	誤りモデル 3
MRE	1,000	0
MER	96	904
BRE	1,000	0
IBRE	975	25

表 15: 誤りモデル 4 との精度比較

評価指標	正確なモデル	誤りモデル 4
MRE	1,000	0
MER	44	956
BRE	1,000	0
IBRE	917	83

6.3.3 某社データセット

正確なモデルと誤りモデル 1 について，各評価指標を用いてどちらの精度が高いかを判定した結果を表 16 に，正確なモデルと誤りモデル 2 について，各評価指標を用いてどちらの精度が高いかを判定した結果を表 17 に，正確なモデルと誤りモデル 3 について，各評価指標を用いてどちらの精度が高いかを判定した結果を表 18 に，正確なモデルと誤りモデル 4 について，各評価指標を用いてどちらの精度が高いかを判定した結果を表 19 に示す．

表中の数字は，そのモデルの精度が他方より高いと評価指標が判定した回数を示す．表 17 より，MRE，BRE，IBRE は正確なモデルよりも誤りモデル 2，つまり，軽度な過小見積もりを行うモデルの精度が高いと判定することがわかる．この結果は Myrtveit らの研究結果と同様である．しかし一方で，表 18 と表 19 より，MER は誤りモデル 3 と誤りモデル 4，つまり，過大見積もりを行うモデルよりも，正確なモデルの精度が高いと判定することがわかる．つまり，某社データセットを用いて Myrtveit らの研究の追試を行ったところ，Myrtveit らの研究とは異なる結果が得られた．結果が異なった原因についての詳細は現時点では不明であるが，誤りモデルを作成する際の a と b の操作量が不適当であり，誤りモデル 3 と誤りモデル 4 の精度が著しく低くなったためであると考えられる（Myrtveit らの論文には a と b の操作量が明記されていなかった）．

表 16: 誤りモデル 1 との精度比較

評価指標	正確なモデル	誤りモデル 1
MRE	873	127
MER	1,000	0
BRE	1,000	0
IBRE	999	1

表 17: 誤りモデル 2 との精度比較

評価指標	正確なモデル	誤りモデル 2
MRE	39	961
MER	662	338
BRE	249	751
IBRE	264	736

表 18: 誤りモデル 3 との精度比較

評価指標	正確なモデル	誤りモデル 3
MRE	1,000	0
MER	1,000	0
BRE	1,000	0
IBRE	1,000	0

表 19: 誤りモデル 4 との精度比較

評価指標	正確なモデル	誤りモデル 4
MRE	1,000	0
MER	997	3
BRE	1,000	0
IBRE	1,000	0

7　妥当性への脅威

本章では，本研究の結果の妥当性に影響を及ぼす恐れのある問題について，Yin らの分類 [5,37] に基づいて議論を行う．

7.1　系統的レビュー

7.1.1　内的妥当性

内的妥当性とは，独立変数と従属変数の因果関係について，その因果関係が存在する程度を指す．つまり，研究結果が研究の際に操作した要因から影響を受けている程度を指す．系統的レビューにおいては，結果は調査対象論文以外の影響を受けないため，内的妥当性に関する議論は省略する．

7.1.2　外的妥当性

外的妥当性とは，ある研究から得られた成果を，違った母集団，状況，条件へ一般化し得る程度を指す．系統的レビューの調査対象とした論文は，出典を CORE Ranking Portal [23] においてランク A 以上に認定されている論文誌または国際会議の論文集に収録された論文に絞り，質の評価を行っている．質の評価方法が異なる場合，調査対象論文も異なり，また得られる結果も異なる恐れが存在する．しかし，CORE Ranking Portal [23] においてランク A 以上に認定されている論文誌または国際会議は，見積もりの研究分野において重要度が高い論文誌と国際会議である．よって，結果の外的妥当性は高いと言える．

また，調査対象論文は 2010 年 1 月から 2015 年 10 月の間に出版されたソフトウェア見積もりに関する研究論文である．検索年代が異なる場合，調査対象論文も異なり，また得られる結果も異なる恐れが存在する．結果の外的妥当性を高めるためには，検索年代を広げ，改めて系統的レビューを行う必要がある．

7.1.3　構成概念妥当性

構成概念妥当性とは，結果を得るために行った操作が適切である程度を指す．調査対象論文の収集と選別及び情報の抽出は手動で行ったため，漏れが生じる恐れは否定できない．しかし，4 章に記述したレビュー手順に基づき，細心の注意を払ってレビューを実施したため，結果の構成概念妥当性は高いと言える．

7.1.4 信頼性

信頼性は，他者が同様の手順で研究を行った場合，研究結果が再現可能となる程度を指す．本研究で行った系統的レビューの手順は Kitchenham らの提案手法 [9] に従っており，かつ論文中に系統的レビューの手順を明記したため，そちらに従えば本研究の結果が再現可能である．よって，結果の信頼性は高いと言える．

7.2 追試

7.2.1 内的妥当性

内的妥当性とは，独立変数と従属変数の因果関係について，その因果関係が存在する程度を指す．つまり，研究結果が研究の際に操作した要因から影響を受けている程度を指す．Myrtveit らは各評価指標の値について有意差検定を行っており，検定の結果，各評価指標の値については有意差が認められた．追試においても同様に有意差検定を行ったところ，各評価指標の値について有意差が認められた．よって，結果の内的妥当性は高いと言える．

7.2.2 外的妥当性

外的妥当性とは，ある研究から得られた成果を，違った母集団，状況，条件へ一般化し得る程度を指す．追試では ISBSG データセット，IPA/SEC データセット，某社データセットを対象に Myrtveit らの研究結果の再現を行った．ISBSG データセットは世界 24 か国の組織・企業のデータ，IPA/SEC データセットは日本国内の組織・企業のデータ，某社データセットは金融保険業で用いるソフトウェアの開発を行っている企業のデータを収録しており，実験対象の多様である．そのため，結果の外的妥当性は高いと言える．しかし，他のデータセットを対象に同様の実験を行った場合，結果が変動する恐れは存在する．開発に必要な工数と開発対象ソフトウェアの規模データがあれば実験は再現可能なため，異なるデータセットに対しても追試を実施する必要がある．

7.2.3 構成概念妥当性

構成概念妥当性とは，結果を得るために行った操作が適切である程度を指す．Myrtveit らは COCOMO81 データセットから，COCOMO81 データセットに似た特徴を持ち，30 プロジェクトデータを収録した仮想データセットを 1000 個作成し，その全てを対象に各評価指標の信頼性を調査した．追試はこの操作方法に準じているため，結果の構成概念妥当性は高いと言える．

7.2.4 信頼性

信頼性は，他者が同様の手順で研究を行った場合，研究結果が再現可能となる程度を指す．実験手順は Foss らの論文 [33] に明記されているため，そちらに従えば本研究の結果が再現可能である．よって，結果の信頼性は高いと言える．

8　あとがき

本章では，本研究のまとめと今後の課題について述べる．

本研究では，見積もり研究における研究成果の外的妥当性についての調査を目的とし，過去 6 年間に出版された見積もりに関する論文を対象とした系統的レビューを実施した．89 本の論文を調査した結果，研究成果の外的妥当性についての議論を行っていない研究論文が 26 本存在することが判明した．また，外的妥当性の検証を目的として追試を実施した研究論文はわずか 2 本であった．以上の結果から，見積もりの研究に携わる研究者は，研究成果の外的妥当性に関して注意を払うべきであると主張する．

更に本研究では，系統的レビューの結果から，研究成果の外的妥当性について検証の余地があると判断した既存研究に対する追試を行った．既存研究では，実験対象として単一のデータセットのみが用いられていたため，研究成果の外的妥当性について検証の余地があると判断した．追試の結果，既存研究の結果とは一部異なった結果が得られた．この結果は，追試によって研究成果の外的妥当性を検証することに意義が認められるという，過去研究の主張 [8] を補強するものであると考える．

今後の課題として，他の既存研究に対する追試が考えられる．系統的レビューの結果，実験対象として単一のデータセットのみを用いている研究論文は 10 本存在した．本研究で追試の対象とした既存研究はこの 10 本の研究論文の中から選択したが，残り 9 本についても外的妥当性を検証するべきである．また，見積もり研究における外的妥当性の検証方法や議論の方法を示すガイドラインの提案も今後の課題として考えられる．系統的レビューの結果，研究成果の外的妥当性についての議論を論文の結論部分でのみ行っている研究論文は 31 本存在した．つまり，研究成果の外的妥当性について意識はしているものの，詳細に議論を行う必要性を認識していない研究者が存在すると言える．このような研究者に対し，外的妥当性に関する議論の記述方法を示したガイドラインの提案が望まれる．

謝辞

本研究の全過程を通し，大変理解ある親身なご指導を賜り，研究の完遂に多大なるご協力を頂きました 楠本 真二 教授に心より感謝申し上げます．

本研究に至るまでに，講義やミーティングにおいて丁寧なご指導を賜りました信州大学情報工学科 岡野 浩三 准教授に深く感謝申し上げます．

本研究に関して，適切なご指導を賜り，また日常の議論の中で多数のご助言を頂きました肥後 芳樹 准教授に深く感謝申し上げます．

本研究に関して，親切なご指導を賜り，また日常の中で常に励まして頂きました 柗本 真佑 助教に深く感謝申し上げます．

本研究に関して，ミーティングにおける議論の中で貴重なご助言を頂きました，近畿大学理工学部情報学科講師の 角田 雅照 氏に深く感謝申し上げます．

本研究を行うにあたり，データを提供して頂くとともに多大なご助言を頂きました，独立行政法人情報処理推進機構技術本部ソフトウェア高信頼化センターの関係各位に深く感謝申し上げます．

本研究を行うにあたり，様々な点でご協力頂きました 鈴鹿 久佳 様に深謝致します．

本研究に関して，多くのご助力を頂きました大阪大学大学院情報科学研究科コンピュータサイエンス専攻特任研究員の櫻井 浩子 氏に深く感謝申し上げます．

研究室生活の中で，事務作業を行う際に多大なるご支援を頂きました大阪大学大学院情報科学研究科コンピュータサイエンス専攻楠本研究室事務員の 神谷 智子 氏，同 藤野 香 氏，同 中埜 由美 氏に深く感謝申し上げます．

本研究を行うにあたりご指導，ご協力を頂き，さらに日常でも声をかけて頂きました大阪大学大学院情報科学研究科コンピュータサイエンス専攻博士後期課程 3 年の 村上 寛明 氏，同 楊 嘉晨 氏に深く感謝申し上げます．

研究室生活の中で相談に乗って頂き，また励まして頂きました大阪大学大学院情報科学研究科コンピュータサイエンス専攻博士前期課程 2 年の 大谷 明央 氏，同 高 良多朗 氏に深く感謝申し上げます．

研究室生活を大変豊かにして頂きました大阪大学大学院情報科学研究科コンピュータサイエンス専攻博士前期課程 1 年の小倉 直徒 氏，同 佐飛 祐介 氏，同 鷲見 創一 氏，同 古田 雄基 氏，同 幸 佑亮 氏，同 横山 晴樹 氏に深く感謝申し上げます．

研究室の環境維持に多くのご助力を頂きました，大阪大学基礎工学部情報科学科 4 年の下仲 健斗 氏，同 中島 弘貴 氏，同 山田 悠斗 氏，同 山本 将弘 氏に深く感謝申し上げます．

最後に，本研究に至るまでに，講義，演習，実験等でお世話になりました大阪大学大学院情報科学研究科の諸先生方に，この場を借りて心から御礼申し上げます．

付録 A　調査対象論文

[1] Pekka Abrahamsson, Ilenia Fronza, Raimund Moser, Jelena Vlasenko, and Witold Pedrycz. Predicting development effort from user stories. In Empirical Software Engineering and Measurement (ESEM), 2011 International Symposium on, pp. 400-403, 2011.

[2] Rama Akkiraju and Hendrik Van Geel. Estimating the cost of developing customizations to packaged application software using service oriented architecture. In Web Services (ICWS), 2010 IEEE International Conference on, pp. 433-440, 2010.

[3] Fatima Azzahra Amazal, Ali Idri, and Alain Abran. Software development effort estimation using classical and fuzzy analogy: a cross-validation comparative study. International Journal of Computational Intelligence and Applications, Vol. 13, No. 03, p. 1450013, 2014.

[4] Leandro Antonelli, Gustavo Rossi, Julio Cesar Sampaio do Prado Leite, and Alejandro Oliveros. Language extended lexicon points: Estimating the size of an application using its language. In Requirements Engineering Conference (RE), 2014 IEEE 22nd International, pp. 263-272, 2014.

[5] De A Araujo, Adriano LI Oliveira, Sergio Soares, Silvio Meira, et al. Gradient-based morphological approach for software development cost estimation. In Neural Networks (IJCNN), The 2011 International Joint Conference on, pp. 588-594, 2011.

[6] Ricardo de A Ara újo, Adriano LI Oliveira, Sergio Soares, and Silvio Meira. An evolutionary morphological approach for software development cost estimation. Neural Networks, Vol. 32, pp. 285-291, 2012.

[7] Iman Attarzadeh and Siew Hock Ow. Improving estimation accuracy of the cocomo ii using an adaptive fuzzy logic model. In Fuzzy Systems (FUZZ), 2011 IEEE International Conference on, pp. 2458-2464, 2011.

[8] Damir Azhar, Patricia Riddle, Eduardo Mendes, Nikolaos Mittas, and Lefteris Angelis. Using ensembles for web effort estimation. In Empirical Software Engineering and Measurement, 2013 ACM/IEEE International Symposium on, pp. 173-182, 2013.

[9] Mohammad Azzeh. A replicated assessment and comparison of adaptation techniques for analogy-based effort estimation. Empirical Software Engineering, Vol. 17, No. 1, pp. 90-127, 2012.

[10] Mohammad Azzeh, Ali Bou Nassif, and Leandro L Minku. An empirical evaluation of ensemble adjustment methods for analogy-based effort estimation. Journal of Systems and Software, Vol. 103, pp. 36-52, 2015.

[11] Mohammad Azzeh, Daniel Neagu, and Peter I Cowling. Fuzzy grey relational analysis for software effort estimation. Empirical Software Engineering, Vol. 15, No. 1, pp. 60-90, 2010.

[12] Mohammad Azzeh, Daniel Neagu, and Peter I Cowling. Analogy-based software effort estimation using fuzzy numbers. Journal of Systems and Software, Vol. 84, No. 2, pp. 270-284, 2011.

[13] Tibor Bakota, Péter Hegedus, Gergely Ladányi, Peter Kortvelyesi, Rudolf Ferenc, and Tibor Gyimóthy. A cost model based on software maintainability. In Software Maintenance (ICSM), 2012 28th IEEE International Conference on, pp. 316-325, 2012.

[14] Vahid Khatibi Bardsiri, Dayang Norhayati Abang Jawawi, Siti Zaiton Mohd Hashim, and Elham Khatibi. A flexible method to estimate the software development effort based on the classification of projects and localization of comparisons. Empirical Software Engineering, Vol. 19, No. 4, pp. 857-884, 2014.

[15] Rodrigo C Barros, Márcio P Basgalupp, Ricardo Cerri, Tiago S da Silva, and André CPLF de Carvalho. A grammatical evolution approach for software effort estimation. In Proceedings of the 15th annual conference on Genetic and evolutionary computation, pp. 1413-1420, 2013.

[16] Dirk Basten and Werner Mellis. A current assessment of software development effort estimation. In Empirical Software Engineering and Measurement (ESEM), 2011 International Symposium on, pp. 235-244, 2011.

[17] Dirk Basten and Ali Sunyaev. A systematic mapping of factors affecting accuracy of software development effort estimation. Communications of the Association for Information Systems, Vol. 34, No. 1, p. 4, 2014.

[18] Barry Boehm and Ricardo Valerdi. Impact of software resource estimation research on practice: a preliminary report on achievements, synergies, and challenges. In Software Engineering (ICSE), 2011 33rd International Conference on, pp. 1057-1065, 2011.

[19] Kristin Børte and Monika Nerland. Software effort estimation as collective accomplishment: An analysis of estimation practice in a multi-specialist team. Scandinavian Journal of Information Systems, Vol. 22, No. 2, pp. 71-104, 2010.

[20] Andrea Capiluppi, Daniel Izquierdo-Cortázar. Effort estimation of FLOSS projects: a study of the linux kernel. Empirical Software Engineering, Vol. 18, No. 1, pp. 60-88, 2013.

[21] Denis Čeke and Boris Milašinović. Early effort estimation in web application development. Journal of Systems and Software, Vol. 103, pp. 219-237, 2015.

[22] Anna Corazza, Sergio Di Martino, Filomena Ferrucci, Carmine Gravino, and Emilia Mendes. Investigating the use of support vector regression for web effort estimation. Empirical Software Engineering, Vol. 16, No. 2, pp. 211-243, 2011.

[23] Anna Corazza, Sergio Di Martino, Filomena Ferrucci, Carmine Gravino, Federica Sarro, and Emilia Mendes. Using tabu search to configure support vector regression for effort estimation. Empirical Software Engineering, Vol. 18, No. 3, pp. 506-546, 2013.

[24] Adler Diniz De Souza. A proposal for the improvement of project's cost predictability using evm and historical data of cost. In Proceedings of the 2013 International Conference on Software Engineering, pp. 1447-1449, 2013.

[25] Karel Dejaeger, Wouter Verbeke, David Martens, and Bart Baesens. Data mining techniques for software effort estimation: a comparative study. Software Engineering, IEEE Transactions on, Vol. 38, No. 2, pp. 375-397, 2012.

[26] Marta Fernández-Diego and José-María Torralba-Martínez. Discretization methods for nbc in effort estimation: An empirical comparison based on isbsg projects. In Proceedings of the ACM-IEEE international symposium on Empirical software engineering and measurement, pp. 103-106, 2012.

[27] Antonio Girasella and Filippo Pagin. An uml-based approach to software development cost estimation. In Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, p. 16, 2014.

[28] Fernando González-Ladrón-de-Guevara and Marta Fernández-Diego. Isbsg variables most frequently used for software effort estimation: A mapping review. In Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, p. 42, 2014.

[29] Magne Jórgensen. Selection of strategies in judgment-based effort estimation. Journal of Systems and Software, Vol. 83, No. 6, pp. 1039-1050, 2010.

[30] Magne Jórgensen. A strong focus on low price when selecting software providers in- creases the likelihood of failure in software outsourcing projects. In Proceedings of the 17th International Conference on Evaluation and Assessment in Software Engineering, pp. 220-227, 2013.

[31] Magne Jórgensen. Communication of software cost estimates. In Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering, p. 28, 2014.

[32] Magne Jórgensen and Stein Grimstad. The impact of irrelevant and misleading information on software development effort estimates: A randomized controlled field experiment. Software Engineering, IEEE Transactions on, Vol. 37, No. 5, pp. 695-707, 2011.

[33] Magne Jórgensen and Stein Grimstad. Software development estimation biases: The role of interdependence. Software Engineering, IEEE Transactions on, Vol. 38, No. 3, pp. 677-693, 2012.

[34] Magne Jórgensen, Torleif Halkjelsvik. The effects of request formats on judgment-based effort estimation. Journal of Systems and Software, Vol. 83, No. 1, pp. 29-36, 2010.

[35] Magne Jórgensen and Erik Løhre. First impressions in software development effort estimation: Easy to create and difficult to neutralize. In Evaluation & Assessment in Software Engineering (EASE 2012), 16th International Conference on, pp. 216-222, 2012.

[36] Jacky Keung, Ekrem Kocaguneli, and Tim Menzies. Finding conclusion stability for selecting the best effort predictor in software effort estimation. Automated Software Engineering, Vol. 20, No. 4, pp. 543-567, 2013.

[37] Michael Kläs, Adam Trendowicz, Yasushi Ishigai, and Haruka Nakao. Handling estimation uncertainty with bootstrapping: Empirical evaluation in the context of hybrid prediction methods. In Empirical Software Engineering and Measurement (ESEM), 2011 International Symposium on, pp. 245-254, 2011.

[38] Ekrem Kocaguneli, Gregory Gay, Tim Menzies, Ye Yang, and Jacky W Keung. When to use data from other projects for effort estimation. In Proceedings of the IEEE/ACM international conference on Automated software engineering, pp. 321-324, 2010.

[39] Ekrem Kocaguneli and Tim Menzies. How to find relevant data for effort estimation? In Empirical Software Engineering and Measurement (ESEM), 2011 International Symposium on, pp. 255-264, 2011.

[40] Ekrem Kocaguneli and Tim Menzies. Software effort models should be assessed via leave-one-out validation. Journal of Systems and Software, Vol. 86, No. 7, pp. 1879-1890, 2013.

[41] Ekrem Kocaguneli, Tim Menzies, Ayse Basar Bener, and Jacky W Keung. Exploiting the essential assumptions of analogy-based effort estimation. Software Engineering, IEEE Transactions on, Vol. 38, No. 2, pp. 425-438, 2012.

[42] Ekrem Kocaguneli, Tim Menzies, and Jacky W Keung. On the value of ensemble effort estimation. Software Engineering, IEEE Transactions on, Vol. 38, No. 6, pp. 1403-1416, 2012.

[43] Ekrem Kocaguneli, Tim Menzies, and Jacky W Keung. Kernel methods for software effort estimation. Empirical Software Engineering, Vol. 18, No. 1, pp. 1-24, 2013.

[44] Ekrem Kocaguneli, Tim Menzies, Jacky Keung, David Cok, and Ray Madachy. Active learning and effort estimation: Finding the essential content of software effort estimation data. Software Engineering, IEEE Transactions on, Vol. 39, No. 8, pp. 1040-1053, 2013.

[45] Ekrem Kocaguneli, Tim Menzies, and Emilia Mendes. Transfer learning in effort estimation. Empirical Software Engineering, Vol. 20, No. 3, pp. 813-843, 2015.

[46] Luigi Lavazza and Sandro Morasca. Software effort estimation with a generalized robust linear regression technique. In Evaluation & Assessment in Software Engineering (EASE 2012), 16th International Conference on, pp. 206-215, 2012.

[47] Jonathan Lee, Wen-Tin Lee, and Jong-Yih Kuo. Fuzzy logic as a basic for use case point estimation. In Fuzzy Systems (FUZZ), 2011 IEEE International Conference on, pp. 2702-2707, 2011.

[48] Taeho Lee, Taewan Gu, and Jongmoon Baik. Mnd-scemp: an empirical study of a software cost estimation modeling process in the defense domain. Empirical Software Engineering, Vol. 19, No. 1, pp. 213-240, 2014.

[49] Yan-Fu Li, Min Xie, and Thong-Ngee Goh. Adaptive ridge regression system for software cost estimating on multi-collinear datasets. Journal of Systems and Software, Vol. 83, No. 11, pp. 2332-2343, 2010.

[50] Jin-Cherng Lin and Han-Yuan Tzeng. Applying particle swarm optimization to estimate software effort by multiple factors software project clustering. In Computer Symposium (ICS), 2010 International, pp. 1039-1044, 2010.

[51] Kenneth Lind and Rogardt Heldal. Categorization of real-time software components for code size estimation. In Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement, p. 26, 2010.

[52] Kenneth Lind and Rogardt Heldal. A practical approach to size estimation of embedded software components. Software Engineering, IEEE Transactions on, Vol. 38, No. 5, pp. 993-1007, 2012.

[53] Erik Løhre and Magne Jórgensen. Numerical anchors and their strong effects on software development effort estimates. Journal of Systems and Software, 2015.

[54] Cuauhtémoc López-Martín and Alain Abran. Neural networks for predicting the duration of new software projects. Journal of Systems and Software, Vol. 101, pp. 127-135, 2015.

[55] Cuauhtemoc Lopez-Martin, Claudia Isaza, and Arturo Chavoya. Software development effort prediction of industrial projects applying a general regression neural network. Empirical Software Engineering, Vol. 17, No. 6, pp. 738-756, 2012.

[56] Stephen G MacDonell and Martin Shepperd. Data accumulation and software effort prediction. In Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement, p. 31, 2010.

[57] Raymond Madachy, Barry Boehm, Brad Clark, Thomas Tan, and Wilson Rosa. Us dod application domain empirical software cost analysis. In Empirical Software Engineering and Measurement (ESEM), 2011 International Symposium on, pp. 392-395, 2011.

[58] Ana Magazinius, Sofia Börjesson, and Robert Feldt. Investigating intentional distortions in software cost estimation-an exploratory study. Journal of Systems and Software, Vol. 85, No. 8, pp. 1770-1781, 2012.

[59] Viljan Mahnič and Tomaž Hovelja. On using planning poker for estimating user stories. Journal of Systems and Software, Vol. 85, No. 9, pp. 2086-2095, 2012.

[60] Emilia Mendes, Marcos Kalinowski, Daves Martins, Filomena Ferrucci, and Federica Sarro. Cross-vs. within-company cost estimation studies revisited: an extended systematic review. In Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering, p. 12, 2014.

[61] Tim Menzies, Andrew Butcher, David Cok, Andrian Marcus, Lucas Layman, Forrest Shull, Burak Turhan, and Thomas Zimmermann. Local versus global lessons for defect prediction and effort estimation. Software Engineering, IEEE Transactions on, Vol. 39, No. 6, pp. 822-834, 2013.

[62] Tim Menzies, Andrew Butcher, Andrian Marcus, Thomas Zimmermann, and David Cok. Local vs. global models for effort estimation and defect prediction. In Proceedings of the 2011 26th IEEE/ACM International Conference on Automated Software Engineering, pp. 343-351, 2011.

[63] Tim Menzies, Omid Jalali, Jairus Hihn, Dan Baker, and Karen Lum. Stable rankings for different effort models. Automated Software Engineering, Vol. 17, No. 4, pp. 409- 437, 2010.

[64] Leandro L Minku and Xin Yao. Software effort estimation as a multiobjective learning problem. ACM Transactions on Software Engineering and Methodology (TOSEM), Vol. 22, No. 4, p. 35, 2013.

[65] Leandro L Minku and Xin Yao. How to make best use of cross-company data in software effort estimation? In Proceedings of the 36th International Conference on Software Engineering, pp. 446-456, 2014.

[66] Leandro Minku, Federica Sarro, Emilia Mendes, and Filomena Ferrucci. How to make best use of cross-company data for web effort estimation? In Empirical Software Engineering and Measurement (ESEM), 2015 ACM/IEEE International Symposium on, pp. 1-10, 2015.

[67] Nikolaos Mittas and Lefteris Angelis. Lseba: least squares regression and estimation by analogy in a semi-parametric model for software cost estimation. Empirical Software Engineering, Vol. 15, No. 5, pp. 523-555, 2010.

[68] Nikolaos Mittas and Lefteris Angelis. Visual comparison of software cost estimation models by regression error characteristic analysis. Journal of Systems and Software, Vol. 83, No. 4, pp. 621-637, 2010.

[69] Nikolaos Mittas and Lefteris Angelis. A permutation test based on regression error characteristic curves for software cost estimation models. Empirical Software Engineering, Vol. 17, No. 1-2, pp. 34-61, 2012.

[70] Nikolaos Mittas and Lefteris Angelis. Ranking and clustering software cost estimation models through a multiple comparisons algorithm. Software Engineering, IEEE Transactions on, Vol. 39, No. 4, pp. 537-551, 2013.

[71] Nikolaos Mittas, Efi Papatheocharous, Lefteris Angelis, and Andreas S Andreou. Integrating non-parametric models with linear components for producing software cost estimations. Journal of Systems and Software, Vol. 99, pp. 120-134, 2015.

[72] Julie Moeyersoms, Enric Junqué de Fortuny, Karel Dejaeger, Bart Baesens, and David Martens. Comprehensible software fault and effort prediction: A data mining approach. Journal of Systems and Software, Vol. 100, pp. 80-90, 2015.

[73] Ingunn Myrtveit and Erik Stensrud. Validity and reliability of evaluation procedures in comparative studies of effort prediction models. Empirical software engineering,

Vol. 17, No. 1-2, pp. 23-33, 2012.

[74] Ali Bou Nassif, Luiz Fernando Capretz, and Danny Ho. Calibrating use case points. In Companion Proceedings of the 36th International Conference on Software Engineering, pp. 612-613, 2014.

[75] Ali Bou Nassif, Danny Ho, and Luiz Fernando Capretz. Towards an early software estimation using log-linear regression and a multilayer perceptron model. Journal of Systems and Software, Vol. 86, No. 1, pp. 144-160, 2013.

[76] Dinesh R Pai, Kevin S McFall, and Girish H Subramanian. Software effort estimation using a neural network ensemble. Journal of Computer Information Systems, Vol. 53, No. 4, pp. 49-58, 2013.

[77] Passakorn Phannachitta, Akito Monden, Jacky Keung, and Kenichi Matsumoto. Case consistency: a necessary data quality property for software engineering data sets. In Proceedings of the 19th International Conference on Evaluation and Assessment in Software Engineering, p. 19, 2015.

[78] Eltjo Poort and Eric Van Der Vliet. Architecting in a solution costing context: Early experiences with solution-based estimating. In Software Architecture (WICSA), 2015 12th Working IEEE/IFIP Conference on, pp. 127-130, 2015.

[79] Narayan Ramasubbu and Rajesh Krishna Balan. Overcoming the challenges in cost estimation for distributed software projects. In Proceedings of the 34th International Conference on Software Engineering, pp. 91-101, 2012.

[80] Wilson Rosa, Ray Madachy, Barry Boehm, and Brad Clark. Simple empirical software effort estimation model. In Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, p. 43, 2014.

[81] Wilson Rosa, Travis Packard, Abishek Krupanand, James W Bilbro, and Max M Hodal. Cots integration and estimation for ERP. Journal of Systems and Software, Vol. 86, No. 2, pp. 538-550, 2013.

[82] Yeong-Seok Seo and Doo-Hwan Bae. On the value of outlier elimination on software effort estimation research. Empirical Software Engineering, Vol. 18, No. 4, pp. 659-698, 2013.

[83] Boyce Sigweni. Feature weighting for case-based reasoning software project effort estimation. In Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering, p. 54, 2014.

[84] Boyce Sigweni and Martin Shepperd. Using blind analysis for software engineering experiments. In Proceedings of the 19th International Conference on Evaluation and Assessment in Software Engineering, p. 32, 2015.

[85] Masateru Tsunoda, Sousuke Amasaki, and Chris Lokan. How to treat timing information for software effort estimation? In Proceedings of the 2013 International Conference on Software and System Process, pp. 10-19, 2013.

[86] Burak Turhan. On the dataset shift problem in software engineering prediction models. Empirical Software Engineering, Vol. 17, No. 1-2, pp. 62-74, 2012.

[87] Muhammad Usman, Emilia Mendes, and Jürgen Börstler. Effort estimation in agile software development: A survey on the state of the practice. In Proceedings of the 19th International Conference on Evaluation and Assessment in Software Engineering, p. 12, 2015.

[88] Corinne C Wallshein and Andrew G Loerch. Software cost estimating for cmmi level 5 developers. Journal of Systems and Software, Vol. 105, pp. 72-78, 2015.

[89] Peter A Whigham, Caitlin A Owen, and Stephen G Macdonell. A baseline model for software effort estimation. ACM Transactions on Software Engineering and Methodology (TOSEM), Vol. 24, No. 3, p. 20, 2015.

付録 B　調査対象論文から抽出した情報

付録 A における番号	RQ1:実験対象がデータセット	RQ1:データセット数	RQ1:実験対象	RQ2:議論の有無	RQ2:独立した章	RQ3:新規研究 or 追試
[1]	No		ユーザーストーリー	有り	No	新規研究
[2]	No		開発シナリオ	有り	No	新規研究
[3]	Yes	1	ISBSG	有り	No	新規研究
[4]	No		学生	有り	Yes	新規研究
[5]	Yes	5	Desharnais Cocomo Albrecht Kemerer KotenGray	無し	No	新規研究
[6]	Yes	5	Desharnais Cocomo Albrecht Kemerer KotenGray	有り	No	新規研究
[7]	Yes	2	COCOMO I Nasa93	無し	No	新規研究
[8]	Yes	1	Tukutuku	無し	No	追試
[9]	Yes	7	Maxwell Desharnais COCOMO Albrecht Telecom China Kemerer	有り	Yes	追試
[10]	Yes	8	Desharnais Kemerer Albrecht Cocomo Maxwell China Telecom Nasa	有り	Yes	新規研究
[11]	Yes	5	ISBSG Desharnais Kemerer Albrecht & Gaffney COCOMO81	有り	Yes	新規研究
[12]	Yes	5	ISBSG Desharnais Kemerer Albrecht COCOMO	有り	No	新規研究
[13]	No		Java プロジェクト	有り	No	新規研究
[14]	Yes	3	ISBSG COCOMO8 Maxwell	有り	No	新規研究
[15]	Yes	10	Albrecht China Coc81 Coc81lnh Cocomo-sdr Cocomo-nasa Desharnais Kemerer Maxwell Nasa93	無し	No	新規研究
[16]	No		開発者	有り	Yes	新規研究
[17]	No		研究論文	有り	Yes	新規研究
[18]	No		開発者	無し	No	新規研究
[19]	No		専門家	有り	No	新規研究
[20]	No		linux git repository	有り	Yes	新規研究
[21]	No		web プロジェクト	有り	Yes	新規研究
[22]	Yes	1	Tukutuku	有り	Yes	新規研究
[23]	Yes	12	COCOMO81 Nasa93 Desharnais Albrecht Finnish SDR Maxwell Telecom Miyazaki Kemerer China Tukutuku dataset	有り	Yes	新規研究
[24]	No		実プロジェクト	無し	No	新規研究

付録 A における番号	RQ1:実験対象がデータセット	RQ1:データセット数	RQ1:実験対象	RQ2:議論の有無	RQ2:独立した章	RQ3:新規研究 or 追試
[25]	Yes	9	Coc81 Maxwell Experience ESA ISBSG USP05 Cocnasa Euroclear Desharnais	無し	No	新規研究
[26]	Yes	1	ISBSG	無し	No	新規研究
[27]	No		実プロジェクト	無し	No	新規研究
[28]	No		研究論文	有り	No	新規研究
[29]	No		開発者	有り	No	新規研究
[30]	No		実プロジェクト	有り	No	新規研究
[31]	No		開発者	無し	No	新規研究
[32]	No		開発会社	無し	No	新規研究
[33]	No		開発者	有り	No	新規研究
[34]	No		専門家	有り	No	新規研究
[35]	No		開発者	無し	No	新規研究
[36]	Yes	11	COCOMO81 Nasa93 Desharnais Albrecht Finnish SDR Maxwell Telecom Miyazaki Kemerer China	有り	Yes	新規研究
[37]	No		見積もり手法	有り	Yes	新規研究
[38]	Yes	3	COCOMO81 Desharnais Nasa93	無し	No	新規研究
[39]	Yes	6	COCOMO81 Desharnais Finnish Nasa93 Kemerer Maxwell	有り	Yes	新規研究
[40]	Yes	11	COCOMO81 Nasa93 Desharnais Albrecht Finnish SDR Maxwell Telecom Miyazaki Kemerer China	有り	No	新規研究
[41]	Yes	6	COCOMO81 Nasa93 Desharnais Albrecht ISBSG SDR	有り	Yes	新規研究
[42]	Yes	11	COCOMO81 Nasa93 Desharnais Albrecht Finnish SDR Maxwell Telecom Miyazaki Kemerer China	有り	Yes	新規研究
[43]	Yes	10	COCOMO81 Nasa93 Desharnais Albrecht Finnish SDR Maxwell Telecom Miyazaki Kemerer	有り	Yes	新規研究

付録 A における番号	RQ1:実験対象がデータセット	RQ1:データセット数	RQ1:実験対象	RQ2:議論の有無	RQ2:独立した章	RQ3:新規研究 or 追試
[44]	Yes	9	COCOMO81 Nasa93 Desharnais SDR Maxwell Miyazaki Kemerer Finnsh Albrecht	無し	No	新規研究
[45]	Yes	3	Tukutuku Nasa93 COCOMO81	有り	Yes	新規研究
[46]	Yes	4	Desharnais Nasa93 Albrecht Qqdefects	有り	Yes	新規研究
[47]	Yes		ユースケース	無し	No	新規研究
[48]	No		実プロジェクト	有り	No	新規研究
[49]	Yes	2	Ablrecht Desharnais	有り	Yes	新規研究
[50]	Yes	1	COCOMO81	無し	No	新規研究
[51]	No		ソフトウェア	有り	Yes	新規研究
[52]	No		開発者	有り	Yes	新規研究
[53]	No		開発者	有り	No	新規研究
[54]	Yes	1	ISBSG	有り	No	新規研究
[55]	Yes	1	ISBSG	無し	No	新規研究
[56]	No		実プロジェクト	有り	No	新規研究
[57]	No		ドキュメント	無し	No	新規研究
[58]	No		開発者	有り	No	新規研究
[59]	No		学生と専門家	有り	Yes	新規研究
[60]	No		研究論文	無し	No	新規研究
[61]	Yes	2	China NasaCoc	有り	Yes	新規研究
[62]	Yes	2	China NasaCoc	有り	No	新規研究
[63]	Yes	2	Coc81 Nasa93	有り	Yes	新規研究
[64]	Yes	5	SDR Nasa Nasa93 COCOMO81 Desharnais ISBSG	無し	No	新規研究
[65]	Yes	5	Maxwell COCOMO81 ISBSG2000 ISBSG2001 ISBSG	有り	Yes	新規研究
[66]	Yes	1	Tukutuku	有り	Yes	新規研究
[67]	Yes	3	ISBSG Nasa93 ISBSG	有り	No	新規研究
[68]	Yes	1	ISBSG	有り	No	新規研究
[69]	Yes	2	Nasa93 ISBSG	有り	No	新規研究
[70]	Yes	6	ISBSG Desharnais Albrecht Kemerer Miyazaki Telecom	有り	No	新規研究
[71]	Yes	2	ISBSG Nasa93	無し	No	新規研究
[72]	Yes	5	COCOMO81 Cocomonasa2 European Space Agency Maxwell Desharnais	有り	No	新規研究
[73]	Yes	1	COCOMO81	有り	No	新規研究
[74]	No		なし	無し	No	新規研究
[75]	Yes	3	ISBSG Western University CompuTop	有り	Yes	新規研究
[76]	No		実プロジェクト	有り	No	新規研究
[77]	Yes	4	Kemerer Miyazaki China Nasa93	有り	Yes	新規研究
[78]	No		開発者	無し	No	新規研究

付録 A における番号	RQ1:実験対象がデータセット	RQ1:データセット数	RQ1:実験対象	RQ2:議論の有無	RQ2:独立した章	RQ3:新規研究 or 追試
[79]	No		実プロジェクト	有り	No	新規研究
[80]	No		実プロジェクト	有り	Yes	新規研究
[81]	No		実プロジェクト	有り	No	新規研究
[82]	Yes	4	ISBSG Desharnais Bank dataset Stock data	有り	No	新規研究
[83]	No		なし	有り	Yes	新規研究
[84]	Yes	1	Finnish	無し	No	新規研究
[85]	Yes	3	ISBSG Maxwell Kitchenham	有り	No	新規研究
[86]	No		なし	無し	No	新規研究
[87]	No		開発者	有り	Yes	新規研究
[88]	No		実プロジェクト	有り	Yes	新規研究
[89]	Yes	4	Cocomo81 Desharnais Maxwell ISBSG	無し	No	新規研究

参考文献

[1] Narciso Cerpa and June M Verner. Why did your project fail? *Communications of the ACM*, Vol. 52, No. 12, pp. 130–134, 2009.

[2] Barry W Boehm and Ricardo Valerdi. Achievements and challenges in software resource estimation. *development*, Vol. 4, p. 30, 2006.

[3] Saleem Basha and Dhavachelvan Ponnurangam. Analysis of empirical software effort estimation models. *arXiv preprint arXiv:1004.1239*, 2010.

[4] Magne Jórgensen and Martin Shepperd. A systematic review of software development cost estimation studies. *Software Engineering, IEEE Transactions on*, Vol. 33, No. 1, pp. 33–53, 2007.

[5] Per Runeson and Martin Höst. Guidelines for conducting and reporting case study research in software engineering. *Empirical software engineering*, Vol. 14, No. 2, pp. 131–164, 2009.

[6] Robert Feldt and Ana Magazinius. Validity threats in empirical software engineering research-an initial survey. In *SEKE*, pp. 374–379, 2010.

[7] Cleyton VC de Magalhães, Fabio QB da Silva, and Ronnie ES Santos. Investigations about replication of empirical studies in software engineering: preliminary findings from a mapping study. In *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering*, p. 37, 2014.

[8] Fabio QB Da Silva, Marcos Suassuna, A César C França, Alicia M Grubb, Tatiana B Gouveia, Cleviton VF Monteiro, and Igor Ebrahim dos Santos. Replication of empirical studies in software engineering research: a systematic mapping study. *Empirical Software Engineering*, Vol. 19, No. 3, pp. 501–557, 2014.

[9] Barbara Kitchenham. Procedures for performing systematic reviews. *Keele, UK, Keele University*, Vol. 33, No. 2004, pp. 1–26, 2004.

[10] Pearl Brereton, Barbara A Kitchenham, David Budgen, Mark Turner, and Mohamed Khalil. Lessons from applying the systematic literature review process within the software engineering domain. *Journal of systems and software*, Vol. 80, No. 4, pp. 571–583, 2007.

[11] Barbara Kitchenham, Pearl Brereton, Zhi Li, David Budgen, and Andrew Burn. Repeatability of systematic literature reviews. In *Evaluation & Assessment in Software Engineering (EASE 2011), 15th Annual Conference on*, pp. 46–55, 2011.

[12] Allan J Albrecht and John E Gaffney Jr. Software function, source lines of code, and development effort prediction: a software science validation. *Software Engineering, IEEE Transactions on*, No. 6, pp. 639–648, 1983.

[13] Barbara Kitchenham and Pearl Brereton. A systematic review of systematic review process research in software engineering. *Information and Software Technology*, Vol. 55, No. 12, pp. 2049–2075, 2013.

[14] Jianfeng Wen, Shixian Li, Zhiyong Lin, Yong Hu, and Changqin Huang. Systematic literature review of machine learning based software development effort estimation models. *Information and Software Technology*, Vol. 54, No. 1, pp. 41–59, 2012.

[15] Ali Idri, Fatima azzahra Amazal, and Alain Abran. Analogy-based software development effort estimation: A systematic mapping and review. *Information and Software Technology*, Vol. 58, pp. 206–230, 2015.

[16] Janet Siegmund, Norbert Siegmund, and Sven Apel. Views on internal and external validity in empirical software engineering. In *Software Engineering (ICSE), 2015 IEEE/ACM 37th IEEE International Conference on*, Vol. 1, pp. 9–19, 2015.

[17] IEEE xplore. `http://ieeexplore.ieee.org/Xplore/home.jsp`.

[18] ACM Digital library. `http://dl.acm.org/`.

[19] Science Direct. `http://www.sciencedirect.com/`.

[20] SpringerLink. `http://link.springer.com/`.

[21] Google Scholar. `https://scholar.google.co.jp/`.

[22] Barbara A Kitchenham, Emilia Mendes, and Guilherme H Travassos. Cross versus within-company cost estimation studies: A systematic review. *Software Engineering, IEEE Transactions on*, Vol. 33, No. 5, pp. 316–329, 2007.

[23] Computing Research & Education; CORE Rankings Portal. `http://www.core.edu.au/index.php/conference-portal`（閲覧日:2015 年 12 月 3 日）.

[24] Shari Lawrence Pfleeger. Soup or art? the role of evidential force in empirical software engineering. *Software, IEEE*, Vol. 22, No. 1, pp. 66–73, 2005.

[25] PROMISE SOFTWARE ENGINEERING REPOSITORY. `http://promise.site.uottawa.ca/SERepository/datasets-page.html`.

[26] International Software Benchmarking Standards Group (ISBSG). `http://www.isbsg.org`.

[27] Meiyappan Nagappan, Thomas Zimmermann, and Christian Bird. Diversity in software engineering research. In *Proceedings of the 2013 9th joint meeting on foundations of software engineering*, pp. 466–476, 2013.

[28] Ingunn Myrtveit and Erik Stensrud. Validity and reliability of evaluation procedures in comparative studies of effort prediction models. *Empirical software engineering*, Vol. 17, No. 1-2, pp. 23–33, 2012.

[29] Barry W Boehm, et al. *Software engineering economics*, Vol. 197. Prentice-hall Englewood Cliffs (NJ), 1981.

[30] LC Briand and I Wieczorek. Resource modeling in software engineering. *J. J. Marciniak (Ed.), Encyclopaedia of software engineering*, 2002.

[31] Barbara A Kitchenham, Lesley M Pickard, Stephen G. MacDonell, and Martin J. Shepperd. What accuracy statistics really measure. In *Software, IEE Proceedings*, Vol. 148, pp. 81–85, 2001.

[32] Y Miyazaki, M Terakado, K Ozaki, and H Nozaki. Robust regression for developing software estimation models. *Journal of Systems and Software*, Vol. 27, No. 1, pp. 3–16, 1994.

[33] Tron Foss, Erik Stensrud, Barbara Kitchenham, and Ingunn Myrtveit. A simulation study of the model evaluation criterion mmre. *Software Engineering, IEEE Transactions on*, Vol. 29, No. 11, pp. 985–995, 2003.

[34] Chris Lokan and Emilia Mendes. Cross-company and single-company effort models using the isbsg database: a further replicated study. In *Proceedings of the 2006 ACM/IEEE international symposium on Empirical software engineering*, pp. 75–84, 2006.

[35] Fernando González-Ladrón-de Guevara, Marta Fernández-Diego, and Chris Lokan. The usage of isbsg data fields in software effort estimation: A systematic mapping study. *Journal of Systems and Software*, Vol. 113, pp. 188–215, 2016.

[36] IPA 独立行政法人 情報処理推進機構. `https://www.ipa.go.jp`.

[37] Robert K Yin. *Case study research: Design and methods*. Sage publications, 2013.