

ソフトウェア保守改善に向けたデータ分析

角田 雅照

奈良先端科学技術大学院大学／近畿大学

門田 暁人 松本 健一

奈良先端科学技術大学院大学

大岩 佐和子 押野 智樹

一般財団法人経済調査会

1. はじめに

近年、受託開発ソフトウェアは、新規の開発が頭打ちの状況に対して、開発済みのソフトウェア資産が増えることに伴い保守の比率が高まり、つれて保守コスト分析の重要性が高まっている。保守とは、単にソフトウェア出荷後に発見された欠陥を除去することだけを指すのではない。ソフトウェアは利用されているうちに、ビジネス環境の変化により、機能の拡張、修正が必要となる。保守とは、それらの修正も含んだ概念であり、JIS X 0161[1]では以下の4つに分類している。

- (1) 是正保守: ソフトウェア製品の引渡し後に発見された問題を訂正するためにおこなう受身の修正
- (2) 予防保守: 引渡し後のソフトウェア製品の潜在的な障害が顕在化する前に発見し、是正を行うための修正
- (3) 適応保守: 引渡し後、変化した又は変化している環境において、ソフトウェア製品を使用できるように保ち続けるために実施するソフトウェア製品の修正
- (4) 完全化保守: 引渡し後のソフトウェア製品の性能又は保守性を改善するための修正

本稿の目的は、ソフトウェア保守の業務改善を行うにあたり、有用な情報を提供することである。そのために、一般財団法人経済調査会によって平成24年度に収集された134件の事例に対し、統計的な分析を行った。

2. ソフトウェア保守データの分析

2.1 ソフトウェア保守費用の分析

ソフトウェア保守の費用を決定している主要な要因について分析した。ここでソフトウェア保守の費用とは、あるソフトウェアに対する1年間の保守費用である（契約金額であり、原価ではない）。ソフトウェア保守の費用は人件費に基づく、すなわち、以下の項目によって費用が決定していると考えられる。

- 受託側作業時間：ソフトウェア保守の受託側の年間作業時間
- 受託側作業要員数：ソフトウェア保守の受託側の作業要員数（分析の都合上、派遣、専任、非専任など全て含めて合計した）

表1に上記の項目のデータ数と平均値、中央値（値を大きい順に並べた場合に真ん中の順位となる値）を示す。それぞれの項目が保守費用に影響を与えているかどうか確かめるために、保守費用との関連の強さを分析した。関連の強さの分析には、スピアマンの順位相関係数を用いた。スピアマンの順位相関係数は-1から1の値を取り、絶対値が1に近いほど関連が強いことを示す。以降では単に相関係数と表す。

表2に保守費用との相関係数を示す。受託側作業時間のほうが、受託側作業要員数よりも相関係数大きい。受託側作業時間の相関係数の大きさは0.8に近く、保守費用との関連が強いことを示している。各項目と保守費用との関係を図1、図2に示す（値の分布が広いので、本稿の散布図では軸を対数とした）。なお、図1、図2では保守費用の値は示さなかった。図からも保守費用と受託側作業時間との関連が強いことがわかる。

ここで受託側作業時間と受託側作業要員数との関係を相関係数で確かめると、0.56と中程度の関連の強さがあった。これは、受託側作業時間と受託側作業要員数に相互の関係があることを示しており、例えば作業要員数が多くなるほど作業時間が多くなり、その結果保守費用が多くなっている、つまり保守費用に影響する要因は作業要員数である可能性もある。

そこで、受託側作業時間と受託側作業要員数の相互の関係を考慮して保守費用との関係を分析するため、重回帰分析を用いた。重回帰分析を用いることにより、項目間の相互の関係を除外して、それぞれの項目の保守費用に対する関連の強さを分析することができる。

重回帰分析により構築されたモデルの標準化偏回帰係数を表 3 に示す。標準化偏回帰係数が大きいほど、保守費用との関連が大きいことを示す。受託側作業時間のほうが係数も大きく、また有意確率も 5%を下回っていた。有意確率は分析結果の確からしさを示し、一般に 5%を下回る場合、結果が信頼できるといえる。

さらに、調整済み R^2 は 0.73 であった。調整済み R^2 は 0 から 1 の値を取り、一般に 0.5 を超え 1 に近いほど、説明変数によって目的変数が決定している割合が大きくなる。本稿の場合は、この重回帰分析の結果から、保守費用はおおむね受託側作業時間で決定しているといえる。これは当然の結果ではあるが、分析の誤りを防ぐために、統計的に関係を確認しておくことは重要である。

本節のまとめ：保守費用は受託側作業時間に基づいて決まる。受託側作業要員数の保守費用に対する影響は大きくない。

表 1 受託側作業員時間，受託側作業要員数の統計量

	データ数	平均値	中央値
受託側作業時間	88	2585	240
受託側作業要員数	91	4.1	2.0

表 2 保守費用との相関係数

項目	データ数	相関係数	有意確率
受託側作業要員数	76	0.6	0%
受託側作業時間	81	0.79	0%

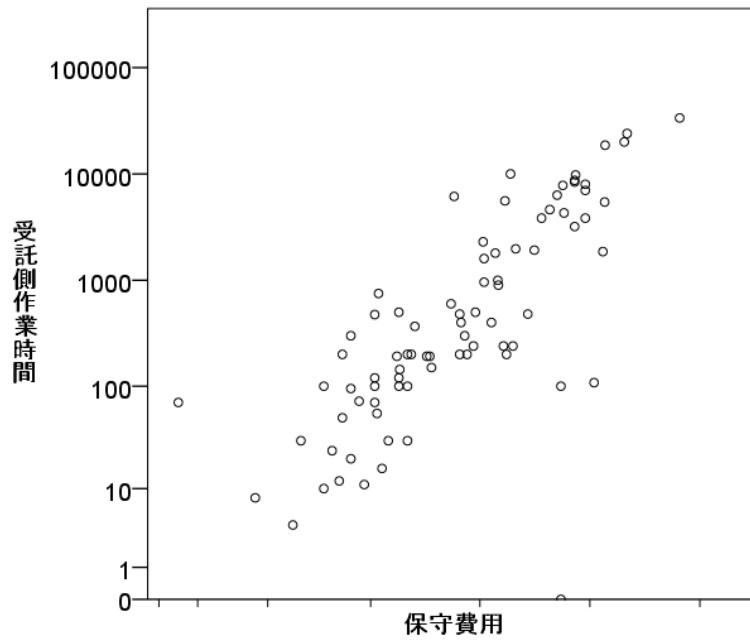


図 1 保守費用と受託側作業時間との関係

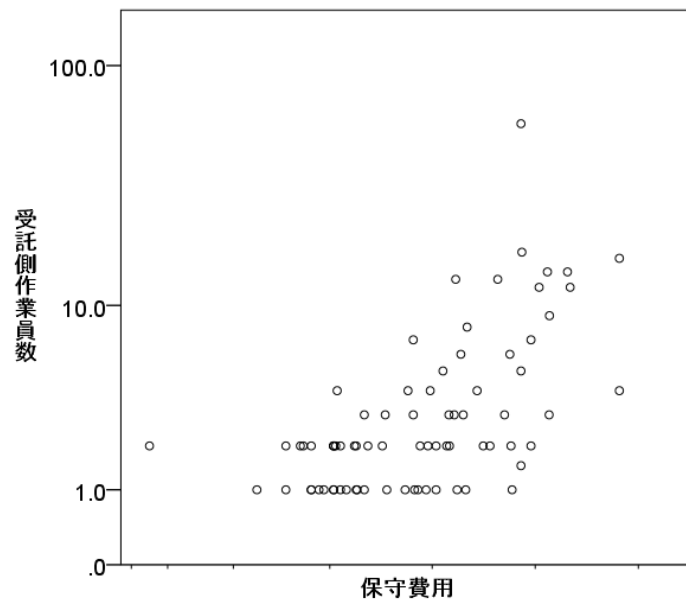


図 2 保守費用と受託側作業要員数との関係

表 3 各項目の標準化偏回帰係数

項目	標準化偏回帰係数	有意確率
受託側作業時間	0.76	0%
受託側作業要員数	0.15	7%

2.2 ソフトウェア保守の修正量の分析

ソフトウェア保守の修正量を表す項目について分析した。修正量は、ソフトウェア保守の効率について議論する場合、欠くことのできない要素である。修正量を表す指標として、以下の項目が収集されている。

- 実施 FP (Function Point) : 保守対象のソフトウェアに対して、修正を加えた機能量
- 実施 SLOC (Source Lines of Code) : 保守対象のソフトウェアにおいて、ソースコードに修正を加えた行数
- 実施 PG 本数: 保守対象のソフトウェアにおいて、修正を加えたプログラムの本数
- 実施画面数: 保守対象のソフトウェアにおいて、修正を加えた画面の数
- 実施帳票数: 保守対象のソフトウェアにおいて、修正を加えた帳票の数
- 実施ファイル数: 保守対象のソフトウェアにおいて、修正を加えたデータファイルの数
- 実施 BAT 数: 保守対象のソフトウェアにおいて、修正を加えたバッチファイルの数

ソフトウェア保守の修正量と作業量の関係を分析するにあたり、上記の項目のなかから修正量を表す指標を検討した。ソフトウェア開発データリポジトリの分析結果[3]では、開発工数に最も影響を与える規模の指標は FP である。そのため、ソフトウェア保守でも実施 FP が修正量を表す指標として最も適していると考えられる。ただし、表 4 に示すように実施 FP のデータ数は多くないため、実施 FP を用いて分析すると、結果に偏りが生じる可能性がある。

そこで本稿では、実施 FP よりもデータ件数が多く、実施 FP との関連が最も強い項目を修正量として採用する。表 5 に実施 FP との相関係数 (2.1 節参照) を示す。実施 PG 本数の相関係数が最も大きく、また、有意確率 (2.1 節参照) も 4%であった。そこで以降の分析では、実施 PG 本数を修正量として採用する。

なお、相関係数 0.54 は関連の強さが中程度であることを示している。すなわち、実施 PG 本数と実施 FP はほとんど一致しているとまではいえないため、実施 PG 本数を用いた場合の分析結果は、実施 FP を用いた場合と異なる可能性があることに留意されたい。

本節のまとめ：実施 PG 本数をソフトウェア保守における修正量とみなす。

表 4 修正量を示す項目の統計量

項目	データ数	平均値	中央値
実施PG本数	48	514.8	44.0
実施LOC	40	1938.8	42.0
実施FP	23	889.9	600.0
実施画面数	52	42.0	16.0
実施帳票数	39	37.1	10.0
実施ファイル数	36	119.9	25.0
実施BAT数	31	110.6	14.0

表 5 実施 FP との相関係数

項目	データ数	相関係数	有意確率
実施 PG 本数	14	0.54	4%
実施 LOC	22	0.37	9%
実施画面数	17	0.28	28%
実施帳票数	13	0.23	45%
実施ファイル数	14	0.46	10%
実施 BAT 数	14	-0.06	84%

2.3 作業量と修正量との関連分析

2.1 節の分析結果より、保守費用（コスト）と最も関連の強い項目は受託側作業時間であった。作業時間（投入された人的資源）と最も関連の強いものは修正量（産出量）であると考えられる。そこで、作業量と修正量の関係について分析した。修正量については、2.2 節の分析結果より実施 PG 本数を用いる。

まず、総作業時間と実施 PG 本数の関係に加えて、総作業要員数、実施 FP との関係も分析し、総作業時間と実施 PG 本数を用いた分析が妥当かどうかを確かめる。総作業時間と総作業要員数の定義を以下に示す。表 6 にそれぞれの平均値と中央値を示す。

- 総作業時間：ソフトウェア保守の受託側と委託側の年間作業時間の合計
- 総作業要員数：ソフトウェア保守の受託側の委託側の作業要員数（派遣，専任，非専任など全て含める）の合計

なお、分析では、作業量の指標に受託側作業時間（作業要員数）ではなく総作業時間（作業要員数）を用いている。これは、保守費用は受託側の作業量のみが影響するのに対し、修正量は委託側と受託側の両方の作業量が影響するためである。

作業量と修正量の関係を

表 7 に示す。また、実施 PG 本数と作業量との関係を

図 3，図 4 に図示する。実施 PG 本数と総作業時間の関連の強さは中程度であり、その他の作業量を示す項目と、修正量を項目との関連の強さも同様に中程度であった。仮に、実施 PG 本数と総作業時間を用いるよりも、他の項目を用いるほうが作業量と修正量の関連の強さが明確になる場合、それらの項目を用いるべきであるが、どの項目を用いた場合でも中程度の関連の強さであったため、実施 PG 本数と総作業時間を分析に用いても、特に問題はないと考えられる。

ここで、総作業時間と実施 PG 本数との関係を単回帰分析により確かめる。単回帰分析と重回帰分析の異なる点は、後者は複数の項目の関係を分析することに対し、前者は総作業時間と実施 PG 本数の 1 対 1 の関係を分析することである。以降の節において、実施 PG 本数以外に総作業時間に関連する要因を分析するが、その前に本節において、実施 PG 本数のみによって総作業時間がどの程度決定しているのかを確かめる。

重回帰分析により構築されたモデルの標準化偏回帰係数は 0.58（有意確率 0%）であり、調整済み R^2 （2.1 節参照）は 0.32 であった。調整済み R^2 の大きさは、実施 PG 本数だけでは総作業時間を決定するには不十分であることを示している。総作業時間を決定するために必要な、その他の項目については以降

の節で検討する。

本節のまとめ：総作業時間（作業量）は、実施 PG 本数（修正量）と関連がある。ただし、総作業時間を決定するためには、実施 PG 本数だけでは不十分である。

表 6 総作業員時間，総作業要員数の統計量

項目	データ数	平均値	中央値
総作業時間	87	10934	300
総作業要員数	98	9.7	3.0

表 7 作業量と修正量の関係

項目	総作業時間			作業要員計		
	データ数	相関係数	有意確率	データ数	相関係数	有意確率
実施 PG 本数	46	0.47	0%	45	0.58	0%
実施 FP	14	0.65	1%	14	0.53	5%

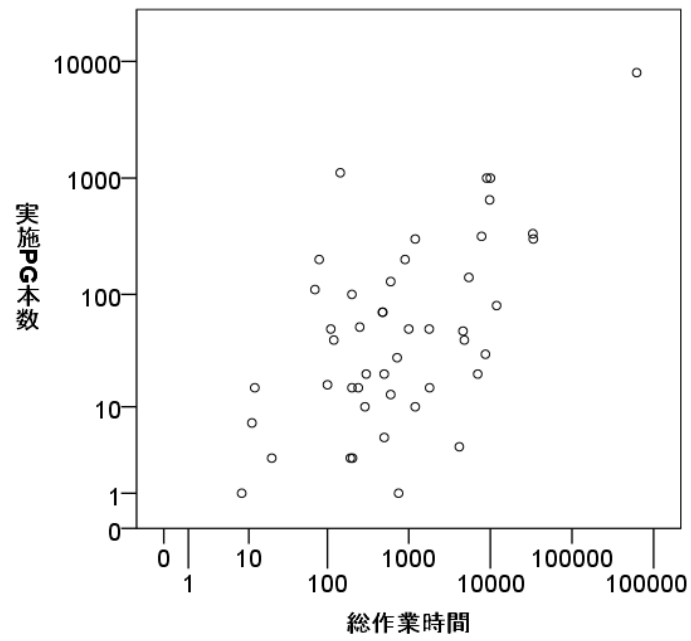


図 3 実施 PG 本数と総作業時間との関係

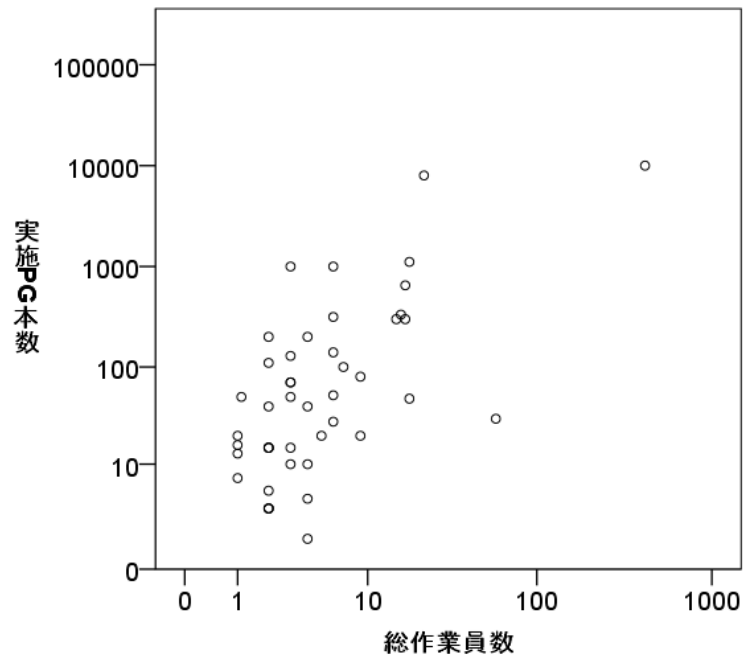


図 4 実施 PG 本数と総作業要員数との関係

2.4 システム構成と作業効率との関連分析

2.4.1 重回帰分析による分析

システム構成が作業量（総作業時間）に与える影響について分析した。システム構成とは、保守対象のソフトウェアが稼働するシステムの構成を指す。1つのソフトウェアが複数のシステム上で稼働している場合もある。種別とデータ数を図 5 に示す。

2.3 節で述べたように、実施 PG 本数も総作業時間に影響しているため、重回帰分析を用いて総作業時間に対するシステム構成と実施 PG 本数の影響を分析する。システム構成は、各種別を1つの項目によって表現する。重回帰モデル適用時には、変数選択を行った。変数選択とは、総作業時間に関連の弱い項目を除外する方法である。

重回帰分析により構築されたモデルの調整済み R^2 (2.1 節参照) は 0.43 であった。Web 系を加えることにより、実施 PG 本数のみの場合に比べ、調整済み R^2 は改善していたが、0.5 は超えておらず、総作業時間を決定するには十分であるとはいえない。

構築されたモデルの標準化偏回帰係数を表 8 に示す。変数選択の結果、Web 系以外のシステム構成の項目がモデルに残らなかった。これは、Web 系以外のシステム構成は総作業時間に与える影響が小さいことを示す。また、実施 PG 本数の標準化偏回帰係数が、Web 系よりも大きいことから、前者のほうが総作業時間に与える影響が大きいといえる。Web 系の標準化偏回帰係数が正の値であり、これは、実施 PG 本数（作業量）が同じ ~~2つの~~ソフトウェアがある場合、Web 系のほうが、総作業時間が多くなる傾向があることを示している。

本節のまとめ：システム構成が Web 系である場合、総作業時間が多くなる傾向がある。ただし、総作業時間を決定するためには、システム構成と実施 PG 本数だけでは不十分である。

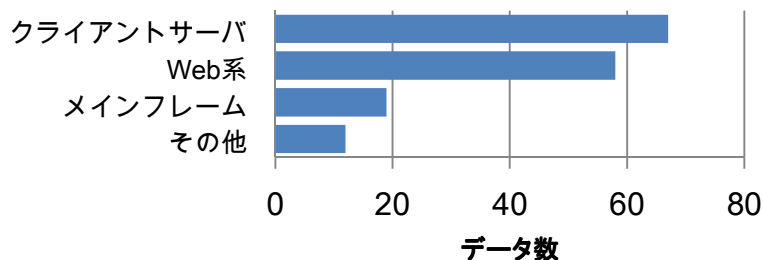


図 5 システム構成の種別とデータ数

表 8 システム構成の標準化偏回帰係数

	標準化偏回帰係数	有意確率
実施 PG 本数	0.51	0%
Web 系 (システム構成)	0.35	0%

2.4.2 作業効率に基づくベンチマーク

2.4.1 節の分析結果を別の観点で表すために、作業効率を新たに定義した。作業効率とは、作業時間（投入された人的資源）に対する修正量（産出量）であり、前者を総作業時間、後者を実施 PG 本数とする時、以下のように定義される。

作業効率: 実施 PG 本数 ÷ 総作業時間

作業効率を用いると、前述の「実施 PG 本数が同じ ~~2-5-4~~ ソフトウェアがある場合、Web 系のほうが、総作業時間が多くなる」は、「Web 系の作業効率（総作業時間あたりの実施 PG 本数）が低い」と言い換えることができる。

システム種別と作業効率の関係を箱ひげ図を用いて確かめた。これらの箱ひげ図を図 6 から図 8 に示す。誌面の都合上、その他のシステム構成の箱ひげ図は省略する。箱ひげ図は、データの分布を表す図であり、箱の中の太線は中央値を示す。箱の下辺は、例えば 100 個の値を小さい順に並べた場合に 25 番目に現れる値を示し、上辺は 75 番目に現れる値を示す。図中のひげの部分は箱から少し離れた値、丸印は箱から大きく離れた値、星印は丸印よりもさらに大きく離れた値を示す。箱の部分に、全体の 50% のデータが含まれる。図では、

箱ひげ図を見やすくするために、作業効率が極端に大きいデータ（外れ値）の一部を除外した。

図より、Web系のシステム構成の場合、明確に作業効率が低くなっていることがわかる。また、クライアントサーバ、メインフレームについては、それらのシステム構成による作業効率の差が小さい（箱の位置の差が少ない）ことがわかる。すなわち、これらの図からも「Web系の作業効率（総作業時間あたりの実施PG本数）が低い」ことがわかる。

これらの箱ひげ図は、簡易的な作業効率のベンチマーク（基準値）として参照することができる。例えば図6は、Web系のシステム構成の場合、作業効率（実施PG本数÷総作業時間）は0.1を下回っているプロジェクトが多いことを示しており、自社のソフトウェア保守がWeb系の場合、業界標準と比較して高いか低いかを簡易的に知ることができる。業界標準よりも低い場合、業務改善の余地がある可能性がある。

本節のまとめ：作業効率の箱ひげ図は、簡易的なベンチマークとして用いることができる。

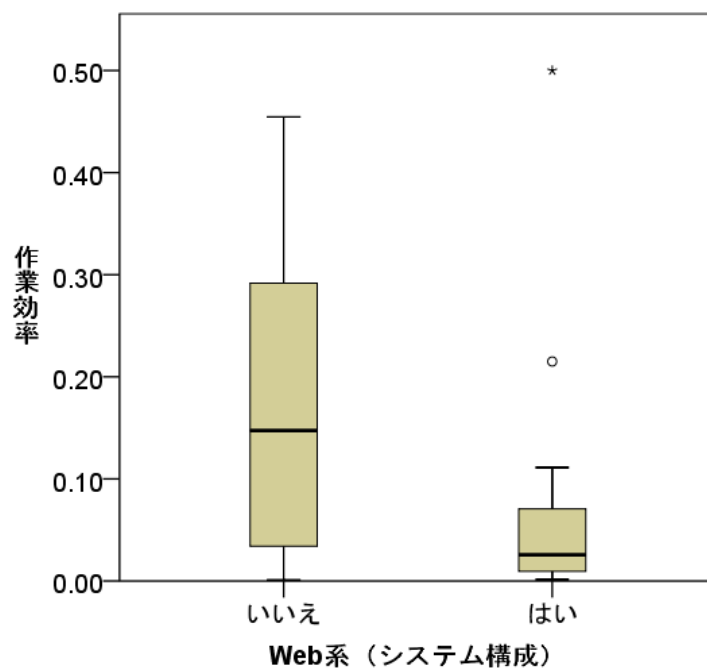


図6 Web系と作業効率との関係

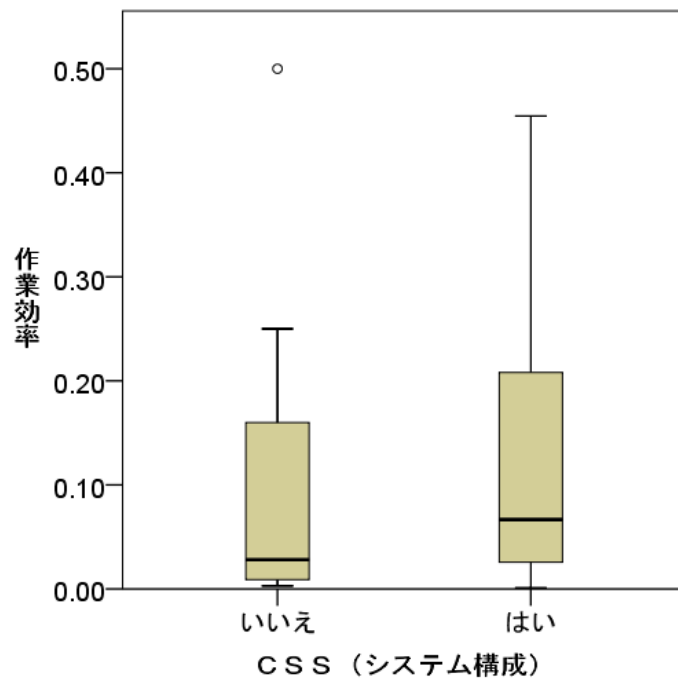


図 7 クライアントサーバ (CSS) と作業効率との関係

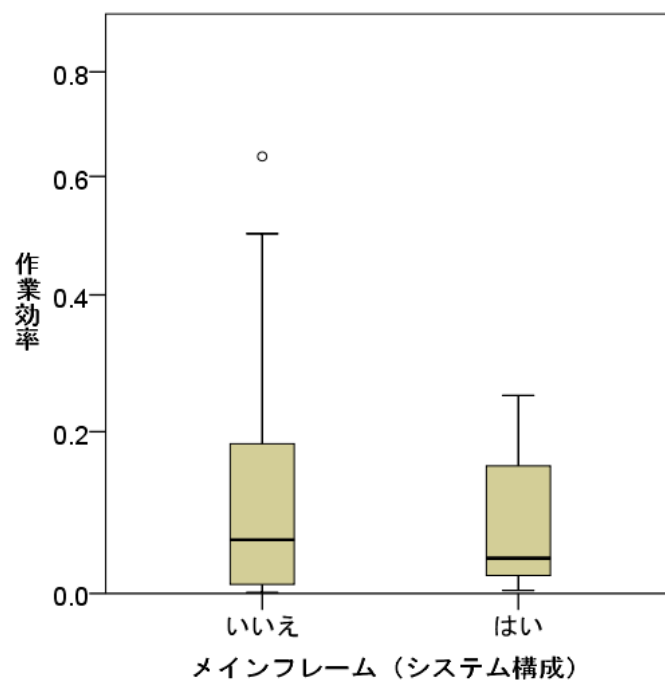


図 8 メインフレームと作業効率との関係

2.5 開発言語と作業効率との関連分析

開発言語が総作業時間に与える影響について分析した。開発言語とは、保守対象のソフトウェアで用いられているプログラミング言語を指す。1つのソフトウェアで複数のプログラミング言語が用いられている場合もある。種別とデータ数を図 9 に示す。

2.4.1 節と同様にして、重回帰分析を用いて総作業時間に対する開発言語と実施 PG 本数の影響を分析した。重回帰分析により構築されたモデルの調整済み R^2 は 0.52 であった。調整済み R^2 が 0.5 を超えていることから(2.1 節参照)、Java と実施 PG 本数により、総作業時間を決定することができるといえる。

重回帰分析により構築されたモデルの標準化偏回帰係数を表 9 に示す。変数選択の結果、Java 以外の開発言語の項目がモデルに残らなかった。Java の標準化偏回帰係数が実施 PG 本数よりも小さく、Java の標準化偏回帰係数は正の値であった。すなわち、開発言語が Java の場合、総作業時間を増加させる傾向があるが、その影響の大きさは実施 PG 本数よりも小さいといえる。

開発言語と作業効率の関係を示す箱ひげ図を図 10 から図 12 に示す。誌面の都合上、Java と一部の開発言語の箱ひげ図のみを示す。それぞれの開発言語の使用、不使用で開発効率が異なっているが、特に Java の場合、明確に開発効率が異なっている（箱の位置がほとんど重なっていない）ことがわかる。

本節のまとめ：開発言語が Java である場合、総作業時間が多くなる傾向がある。

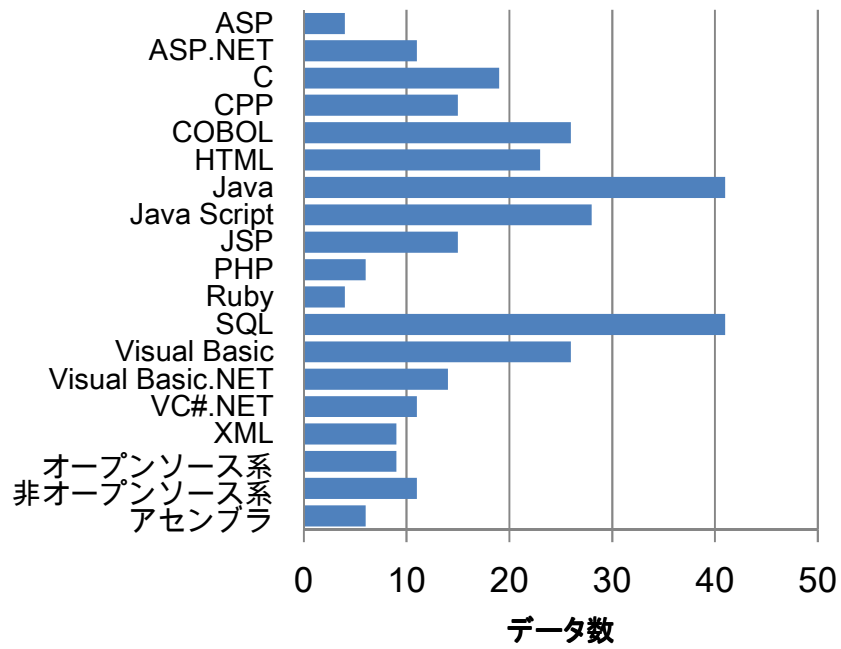


図 9 開発言語の種別とデータ数

表 9 開発言語の標準化偏回帰係数

	標準化偏回帰係数	有意確率
実施 PG 本数	0.58	0%
Java (開発言語)	0.44	0%

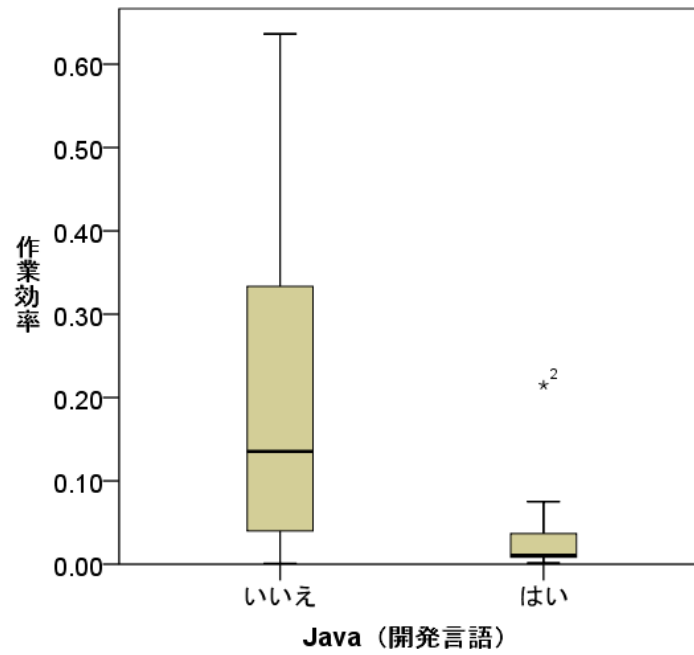


図 10 Java と作業効率との関係

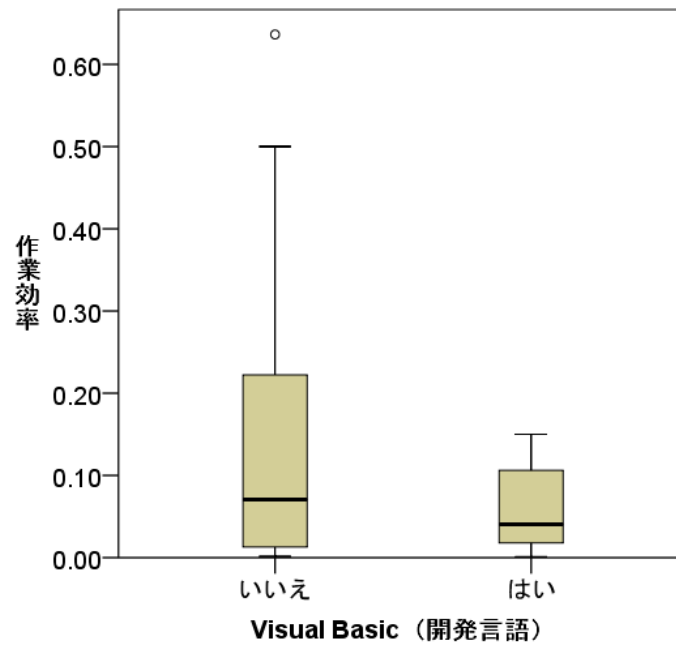


図 11 Visual Basic と作業効率との関係

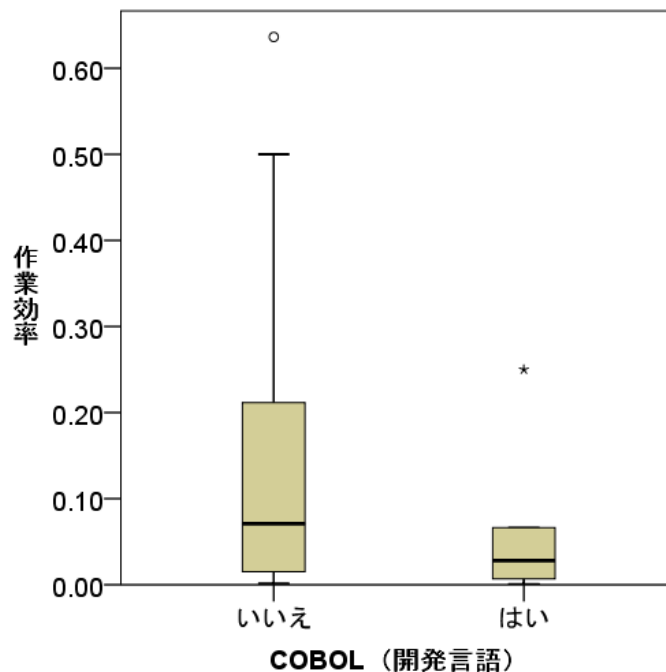


図 12 COBOL と作業効率との関係

2.6 業種と作業効率との関連分析

業種が総作業時間に与える影響について分析した。業種とは、保守対象のソフトウェアの利用者（ソフトウェア保守の委託者）の業種を指す。種別とデータ数を図 13 に示す。2.4.1 節と同様にして、重回帰分析を用いて業種の影響を分析した。重回帰分析により構築されたモデルの調整済み R^2 は 0.37 であった。調整済み R^2 が 0.5 を下回っていることから（2.1 節参照）、業種と実施 PG 本数だけでは、総作業時間を決定できないといえる。

重回帰分析により構築されたモデルの標準化偏回帰係数を表 10 に示す。変数選択の結果、金融業、保険業以外の業種の項目がモデルに残らなかった。金融業、保険業の係数は実施 PG 本数よりも小さく、正の値であった。よって、業種が金融業、保険業の場合、総作業時間を増加させる傾向があり、影響の大きさは実施 PG 本数よりも小さいといえる。

業種と作業効率の関係を示す箱ひげ図を図 14 から図 16 に示す。誌面の都合上、金融業、保険業に加えて、データ数が多い業種の箱ひげ図のみを示す。製造業や卸売業、小売業に該当する場合、該当しない場合と比べて開発効率が同程度、もしくは若干高い傾向があったが、金融業、保険業に該当する場合、

開発効率が明確に低かった。

本節のまとめ：業種が金融業、保険業である場合、総作業時間が多くなる傾向がある。ただし、総作業時間を決定するためには、業種と実施 PG 本数だけでは不十分である。

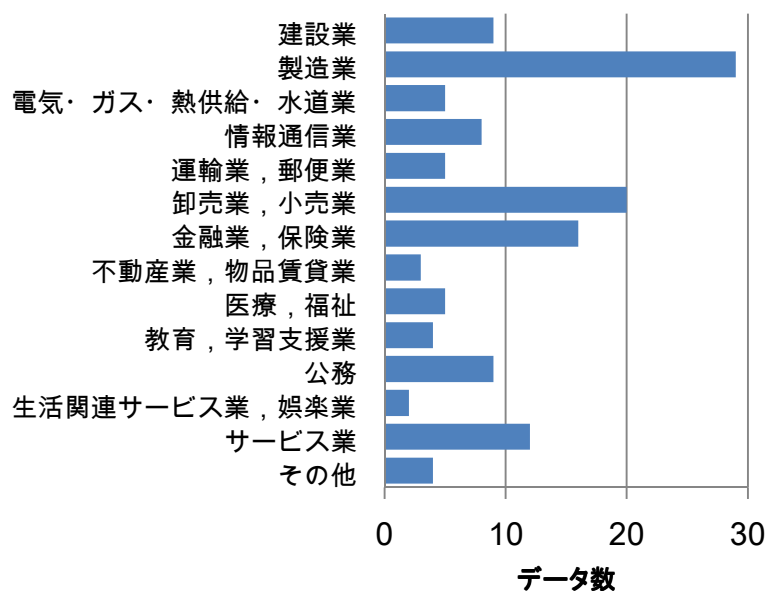


図 13 業種の種別とデータ数

表 10 業種の標準化偏回帰係数

項目	標準化偏回帰係数	有意確率
実施 PG 本数	0.58	0%
金融業, 保険業 (業種)	0.24	5%

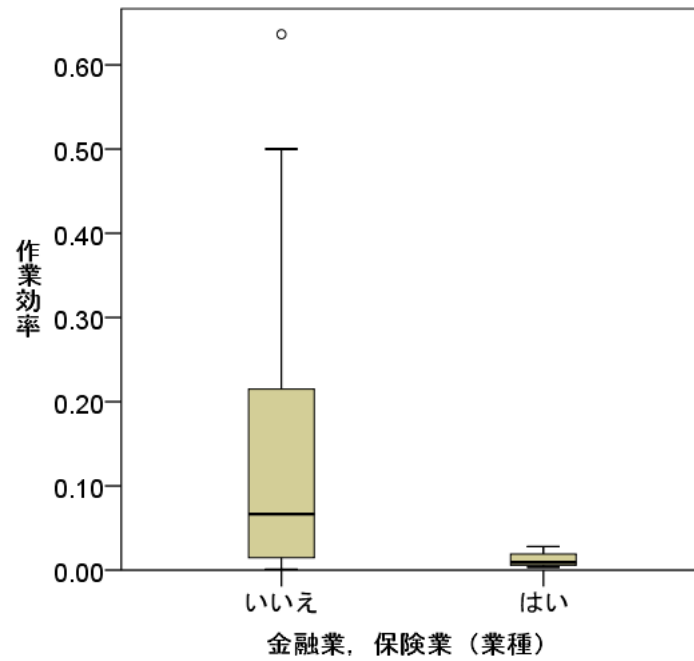


図 14 金融業, 保険業と作業効率との関係

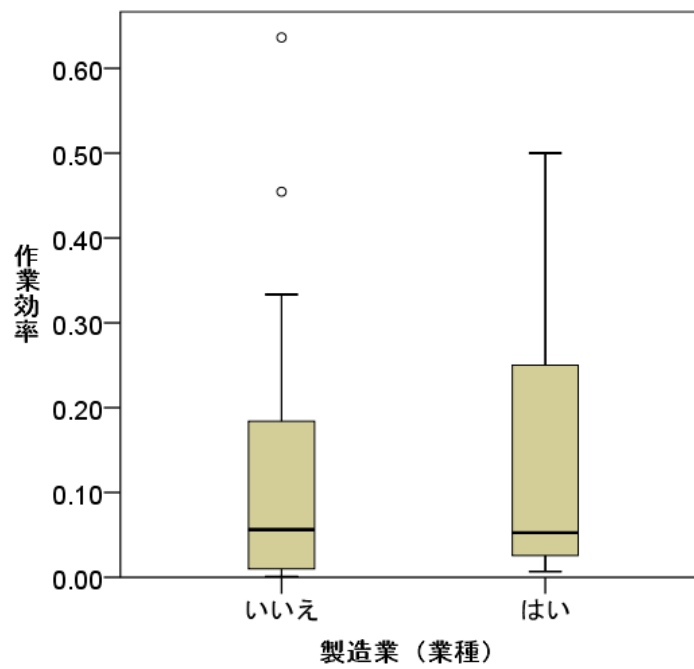


図 15 製造業と作業効率との関係

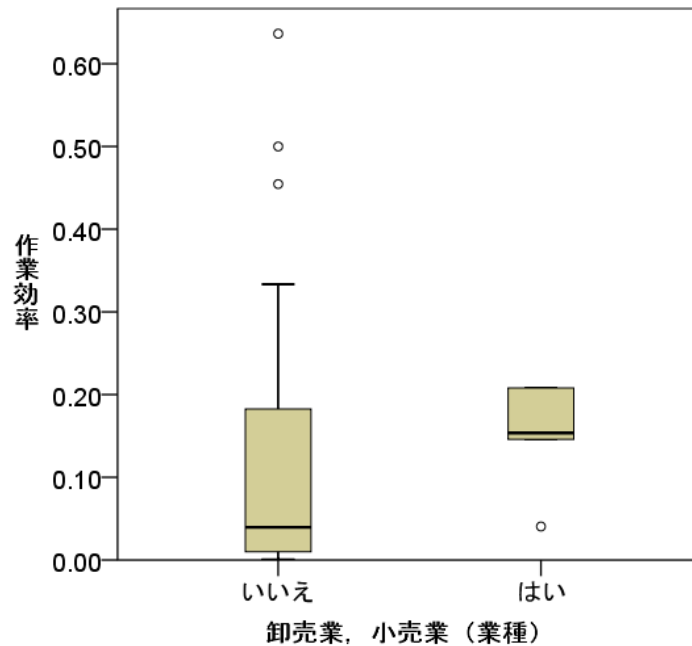


図 16 卸売業，小売業と作業効率との関係

2.7 生産性変動要因と作業量との関連分析

生産性変動要因が総作業時間に与える影響について分析した。生産性変動要因とは、ソフトウェア保守の生産性に影響すると考えられる要因を3段階（1：かなり厳しい，2：少し厳しい，3：厳しくない）で評価した項目[3]である。各項目の詳細を表 11 に示す。

2.4.1 節と同様にして，重回帰分析を用いて分析した。構築されたモデルの調整済み R^2 は 0.41 であった。調整済み R^2 が 0.5 を下回っていることから（2.1 節参照），生産性変動要因と実施 PG 本数だけでは，総作業時間を決定できないといえる。

表 12 に生産性変動要因の標準化偏回帰係数を示す。変数選択の結果，ツール要因以外の項目がモデルに残らなかった。ツール要因の標準化偏回帰係数は負の値であった。これは，ツール要因の値が大きい（要因が厳しくない）場合，総作業時間を減少させる傾向があることを示している。また，係数の絶対値が実施 PG 本数よりも小さく，影響の大きさは実施 PG 本数よりも小さいといえる。

生産性変動要因と作業効率の関係を示す箱ひげ図を図 17 と図 18 に示す。

誌面の都合上，ツール要因と人的要因の箱ひげ図のみを示す．どの生産性変動要因の場合も，値が3の場合，開発効率が高かったが，ツール要員の場合，特に開発効率の差が大きかった（値が2の場合と3の場合で，箱の位置が異なっていた）．

本節のまとめ：ツール要因が厳しくない場合，総作業時間が少なくなる傾向がある．ただし，総作業時間を決定するためには，生産性変動要因と実施 PG 本数だけでは不十分である．

表 11 生産性変動要因の詳細

生産性変動要因	詳細	平均値
人的要因	保守プロジェクトあるいは保守組織の大きさと熟練度	2.0
問題要因	対象とする問題の型と重要度，要求仕様の構成，問題解の制約，波及度合い	1.9
プロセス要因	要求仕様言語，設計/プログラミング言語，開発方法論	2.5
プロダクト要因	対象システムの信頼性，規模，効率，制御構造，複雑度	2.0
資源要因	対象とするハードウェア，期間，予算	2.1
ツール要因	ライブラリ，コンパイラ，テストツール，保守ツール，リバースエンジニアリングツール	2.5

表 12 生産性変動要因の標準化偏回帰係数

項目	標準化偏回帰係数	有意確率
実施 PG 本数	0.44	0%
ツール要因	-0.39	0%

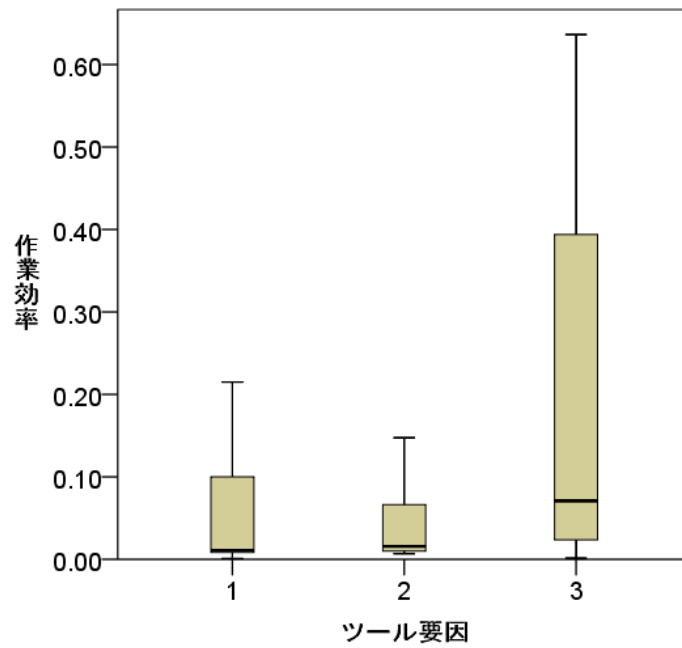


図 17 ツール要因と作業効率との関係

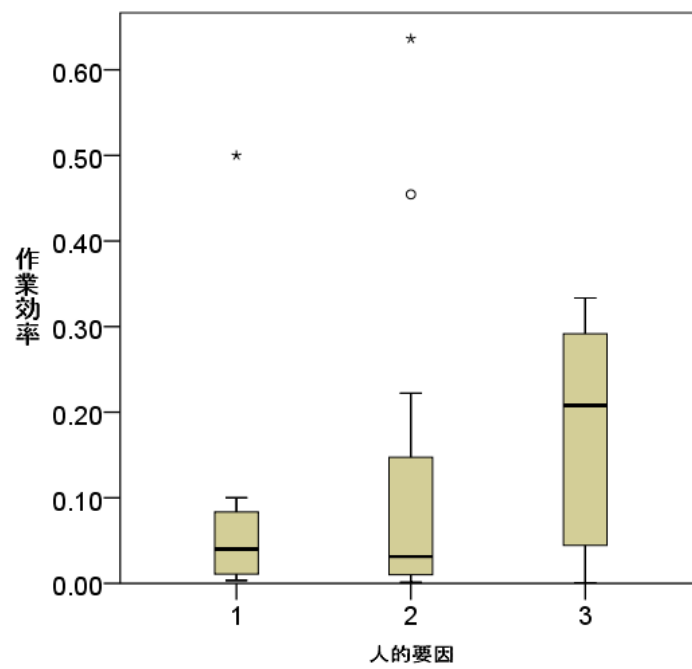


図 18 人的要因と作業効率との関係

2.8 保守種別と作業量との関連分析

保守種別が総作業時間に与える影響について分析した。保守種別は、是正、予防、適応、完全化保守（1章参照）のいずれかの作業割合が、全体の50%以上を占める場合を取り上げ、それらを分類したものである。保守種別とデータ数を図19に示す。2.4.1節と同様にして、重回帰分析を用いて保守種別の影響を分析したが、保守種別がモデルに残らなかった。そこで変数選択を行わずに、データ数の多い是正保守と適応保守を用いてモデルを構築した。

重回帰分析により構築されたモデルの調整済み R^2 は0.41であった。調整済み R^2 が0.5を下回っていることから（2.1節参照）、保守種別と実施PG本数だけでは、総作業時間を決定できないといえる。

重回帰分析により構築されたモデルの標準化偏回帰係数を表13に示す。適応保守の有意確率は5%に近く、係数の絶対値は実施PG本数よりも小さく、負の値であった。適応保守の場合、総作業時間を減少させる可能性があり、影響の大きさは実施PG本数よりも小さい。ただし、有意確率が5%を上回っているため、結論を下すためには今後のさらなる分析が必要である。

保守種別と作業効率の関係を示す箱ひげ図を図20に示す。適応保守の場合、是正保守と比べて開発効率が高い傾向があることが読み取れる。

本節のまとめ：適応保守が全体の保守作業の半分以上を占める場合、総作業時間が少なくなる可能性があるが、さらなる分析が必要である。総作業時間を決定するためには、保守種別と実施PG本数だけでは不十分である。

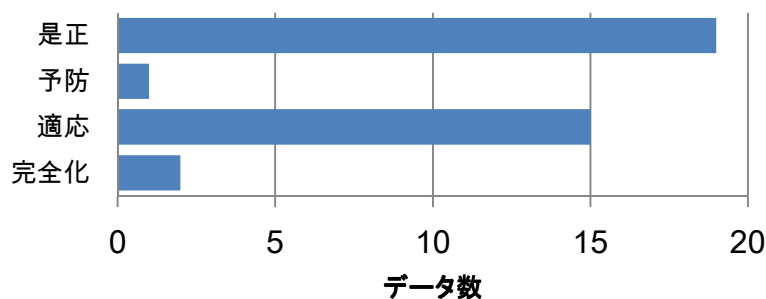


図19 保守種別とデータ数

表 13 保守種別の標準化偏回帰係数

項目	標準化偏回帰係数	有意確率
実施 PG 本数	0.60	0%
是正保守（保守種別）	-0.13	38%
適応保守（保守種別）	-0.27	7%

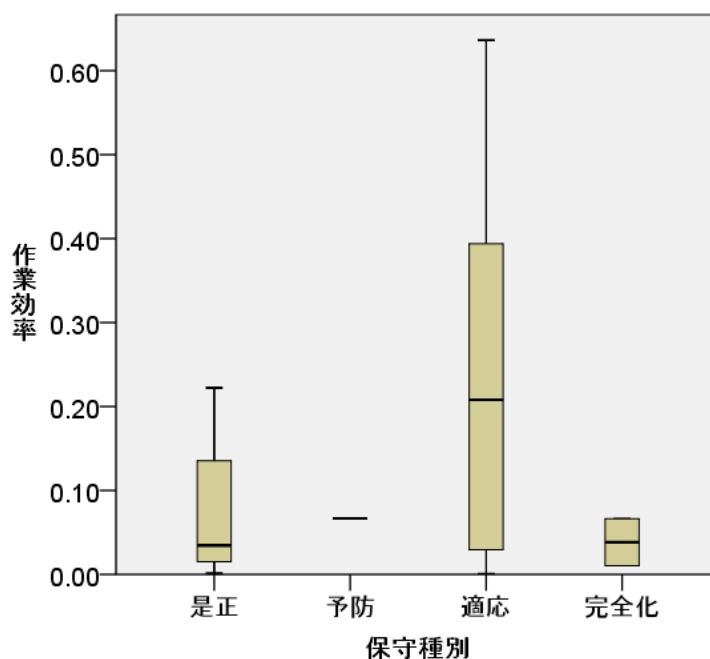


図 20 保守種別と作業効率との関係

2.9 複数の要因と作業量との関連分析

2.4節から2.7節の分析において、総作業時間に影響を与えていた項目それぞれにも、関連があると考えられる。例えば、Web系のシステムでは、開発言語にJavaを用いている可能性が高い。そこで、これらの項目間の関係を考慮しつつ、総作業時間に与える影響を分析するために、これらの項目全てと修正PG本数を用いて重回帰分析を行った。

用いた項目は、Web系（システム構成）、Java（開発言語）、ツール要因（生産性変動要因）、金融業、保険業（業種）、適応保守（保守種別）、修正PG本数である。重回帰分析により構築されたモデルの調整済み R^2 は0.56であった。調整済み

R^2 が0.5を上回っていることから（2.1節参照）、開発言語、生産性変動要因、実施PG本数により、総作業時間を決定できるといえる。

モデルの標準化偏回帰係数を表 14 に示す。変数選択の結果、Web系（システム構成）と金融業、保険業（業種）、適応保守（保守種別）はモデルに残らなかった。これらの項目は、項目間の関係を考慮した場合、総作業時間に影響を与えているとはいえなかった。ツール要因の標準化偏回帰係数は負の値であり、項目間の関係を考慮した場合でも、ツール要因が厳しくない場合、総作業時間は小さくなる傾向があるといえる。

係数の絶対値の大きさから、実施 PG 本数が総作業時間に最も影響を与えており、Java を利用しているかどうかは次に影響を与えるといえる。ツール要因は最も総作業時間に与える影響が小さいが、係数の有意確率は 5%を下回っており、ツール要因は総作業時間に影響しているといえる。

本節のまとめ：総作業時間に対し、実施 PG 本数、開発言語（Java であるかどうか）、ツール要因が影響している。複数の要因の関係を考慮した場合、システム構成（Web 系）、業種（金融業、保険業）、適応保守（保守種別）は総作業時間に影響しているとはいえなかった。

表 14 各項目の標準化偏回帰係数

項目	標準化偏回帰係数	有意確率
実施 PG 本数	0.47	0%
Java（開発言語）	0.37	0%
ツール要因	-0.29	1%

3. おわりに

最後に、ソフトウェア保守の業務改善に向け、どのように本稿の分析結果を利用すればよいかを述べる。2.1節の分析結果より、保守費用を下げるためには、保守の作業時間を減らす必要があるといえる。作業時間を減らすためには、保守業務の改善が必要となる。保守業務を改善する余地があるかどうかの判断基準のひとつとして、本稿の分析結果を参照することができる。具体的には以下の手順で判断を行うとよい。

1. 保守対象のソフトウェアのシステム構成，開発言語，業種，ツール要因，保守種別を確かめる．
2. システム構成がWeb系，開発言語がJava，業種が保険業，金融業，ツール要因が3（厳しくない）未満，保守種別が適応保守以外のいずれかに当てはまっているかを確かめる．
3. 以下に基づき，開発効率を比較する．開発効率の中央値よりも低い場合，改善の余地がある可能性がある．
 - A) 当てはまっている場合，それらの開発効率の中央値（表 15）と，対象の保守ソフトウェアの開発効率を比較する．
 - B) 複数の項目が当てはまっている場合，最も低い開発効率と比較する．
 - C) 当てはまっていない場合，全体の開発効率と比較する．

表 15 要因別の開発効率

	保険業， 金融業	Web 系	Java	ツール要 因 < 3	適応 保守 以外	全体
開発効率 (中央値)	0.0092	0.0257	0.0104	0.0129	0.0367	0.0563
データ数	4	23	15	15	22	43

例えば，自組織の保守対象ソフトウェアの開発言語がJavaであり，ツール要因が3未満である場合，Javaの開発効率の中央値のほうが低いので，この数値と自組織のソフトウェアの開発効率を比較し，中央値（業界標準）よりも低ければ，何らかの業務改善の余地がある可能性がある．ただし，データ数が十分に多いとはいえないので，比較結果を絶対視するのではなく，判断材料のひとつとすべきである．

なお，システム構成，開発言語，業種，ツール要因，保守種別は変更することができない要素である．開発効率と関連の強い，変更することが可能な要素（保守の体制，手順など）を明らかにすることは今後の課題である．

参考文献

- [1] 日本規格協会：JIS X0161 ソフトウェア技術－ソフトウェアライフサイクルプロセス－保守，日本規格協会，2008.
- [2] 経済調査会：ソフトウェア開発データリポジトリの分析，経済調査会，2010.
- [3] ソフトウェア・メンテナンス研究会：第4年度 ソフトウェア・メンテナンス研究会報告書，ソフトウェア・メンテナンス研究会，1995.