

ソフトウェア品質の第三者評価における 探索的データ解析ツールの利用とその効果： OSSデータを対象とした検証実験

大平 雅雄^{†1}伊原 彰紀^{†2}中野 大輔^{†2}松本 健一^{†2}

本論文では、ソフトウェア品質の第三者評価における探索的データ解析ツールの利用とその効果について報告する。探索的データ解析は、明確な分析目的を持たない状態で、与えられたデータに潜むモデルや傾向を多角的に分析する手法である。ソフトウェア品質の評価をおこなうためには、まず、ソフトウェアとその品質が実現される過程を理解するところから始めなければならない場合も多い。データの特徴、傾向、規則性、特異点などを「仮説」として生成し、ソフトウェア品質との因果関係を明らかにしていく過程は、探索的データ解析そのものと言える。OSSプロジェクトデータ(Eclipse Platform プロジェクトの課題データ)を対象に、探索的データ解析ツール HCE (Hierarchical Clustering Explore) [Seo2002] を用いた仮説生成実験をおこなった結果、ドメイン知識のない被験者であっても、ソフトウェア工学の研究分野で検証済みの重要な法則につながる仮説を生成できることが分かった。

Exploratory Data Analysis in Independent Assessment of Software Quality: A Case Study of OSS Project Data

Masao Ohira^{†1}, Akinori Ihara^{†2}, Daisuke Nakano^{†2}, Kenichi Matsumoto^{†2}

Abstract

This paper reports the use of exploratory data analysis and its effect in independent assessment of software quality. Exploratory data analysis (EDA) is a method to analyze models or trends hidden in datasets from various perspectives while there is no explicit goal for the analysis. Assessing software quality often requires an understanding of the process of developing software and achieving the quality. In a sense, the early process of the analysis in independent assessment of software quality can be regarded as the process of EDA, since without any specific goals, extracting characteristics, trends, regularity, singularity and so forth from datasets need to uncover causal relationships hidden in the datasets. We have conducted a case study of an EDA tool called HCE (Hierarchical Clustering Explore) [Seo2002] with issue dataset collected from the Eclipse Platform project. As a result, we found that the tool can support subjects to create hypotheses which would result in finding important rules verified in the literature of software engineering, even if our subjects do not have sufficient knowledge on the target domain.

【脚注】

† 1 和歌山大学 Wakayama University

† 2 奈良先端科学技術大学院大学 Nara Institute of Science and Technology

1. はじめに

現在、利用者からは中身が見えないソフトウェアの安全性、信頼性にかかわる品質について、専門的知見に基づいた中立的な第三者の立場で評価し、専門知識を持たない利用者にも理解できる形で提示するための仕組み「ソフトウェア品質監査制度」の整備が急務となっている [IPA]. ソフトウェアシステムの高度化・複雑化とともに、ソフトウェア品質を担保するための考え方が国際的に変化してきており、我が国の産業界の品質説明力を強化する必要性に迫られているためである。

このような背景を受け、我々の研究グループでは、ソフトウェアやその品質が実現される過程を様々な観点から解析・可視化するための表現手法「ソフトウェアプロジェクトトモグラフィ (Software Project Tomography)」を構築し、ソフトウェア品質の第三者評価を支援するための要素技術を構築した [IPA2012]. ソフトウェアトモグラフィとは、医療におけるコンピュータ断層撮影、所謂、CT (Computed Tomography) から着想を得た手法である。ソフトウェア開発プロジェクトを体に見立て、プロジェクトの開始から終了までの幾つかの時点において、プロジェクトの状態を定量的に表すスナップショットとして、プロジェクトを断面画像のように様々な視点で可視化し、その品質が実現される過程を表現することを目的としている。

本論文では、構築した要素技術の内、「仮説の生成・検証」を支援するための探索型解析技術の効果について、既存の探索的データ解析ツール HCE (Hierarchical Clustering Explore) [Seo2002] を用いた被験者実験の結果をもとに報告する。

2. 探索的データ解析

2.1. 第三者評価における探索的データ解析

探索的データ解析は、明確な分析目的を持たない状態で、与えられたデータに潜むモデルや傾向を多角的に分析する手法である [Tukey]. ソフトウェア品質の第三者評価、特にその初期段階では、探索的データ解析が必要になると考えられる。ソフトウェアの品質を評価する第三者は、対象プロジェクトの全容についてあらかじめ詳細に把握している訳ではないため、ソフトウェアやその品質が実現される過程を理解するところから始めなければならない。

そのため、まず、構成管理システム、工程管理システム、不具合管理システム等に蓄積されたソフトウェアプ

ロジェクトのデータから、その特徴、傾向、規則性、特異点などを「仮説」として生成し、ソフトウェア品質との因果関係を明らかにしていく必要がある。さらに、生成した仮説の真偽や適用範囲の確認をおこなうことが、客観的にソフトウェア品質を検証するための「検証型解析」における解釈モデルや判断基準、分析モデル等の構築に必要となる。

2.2. 第三者評価の範囲と探索型解析技術

ソフトウェア品質の第三者評価における探索型解析技術の主たる目的は、文献 [IPA] に示されている第三者評価 (品質監査) に求められる 4 つの範囲の内、①プロセス実施の妥当性 (プロセスが規定通りに実施されているか、プロセスを構成するタスクやアクティビティが確実に実施されているか) と、②採用規格・技術の妥当性 (採用した規格や技術は適切か、採用した技術は適切に利用されているか) を、検証型解析技術を用いて客観的に検証するのに必要なモデルや基準を提供することである。

一般的に、探索的データ解析では、散布図やヒストグラム、箱髭図等の簡易な統計ツールが利用される。しかし、ソフトウェアプロジェクトデータ (あるいは、ソフトウェアプロジェクトトモグラフィにおいて利用されるスナップショットデータ) は、基本的に多変量・多次元データから構成される。データの特徴や傾向を把握するために任意の 2 変数間の関係を調べる、といった作業だけでも多大な労力が必要となることが多いため、簡易な統計ツールのみでは不十分な場合が多い。

近年では、多変量・多次元データに対する探索的データ分析を支援するソフトウェアが多数登場しており、その多くはデータや分析結果をグラフィカルに表示し、ユーザがデータを探索 (試行錯誤) しながらデータを理解したりモデルを構築したりするのを助ける。本研究では、ソフトウェア品質を評価する第三者は、必ずしも

- ・対象プロジェクトの全容をあらかじめ詳細に把握している訳ではない
- ・多変量・多次元データに対する統計手法に精通している訳ではない

ことを前提 (ペルソナ) として、第三者が効率的に探索的データ解析を行えるようにするための探索型解析技術の選定をおこない、rank-by-feature framework [Seo205] に基づく探索型可視化方式を採用した。また、rank-by-feature framework を実装した探索的データ解析ツール HCE (Hierarchical Clustering Explore)[Seo2002]

[Seo2007]を用いた被験者実験をおこない、第三者評価における探索型解析技術の有用性を検証した。

3. 探索的データ解析ツール

3.1. 技術選定方針

探索型解析技術として、rank-by-feature frameworkにもとづく探索型可視化方式を採用するに至った技術選定方針を以下に述べる。

前述したように、本研究では、ソフトウェア品質を評価する第三者は、対象プロジェクトの詳細及び統計手法に十分な知識がない場合があることを想定している。加えて、ソフトウェアプロジェクトのデータ（本研究におけるスナップショットデータ）は、多次元・多変量データの集合体であり、情報可視化における従来の原則「Overview first, zoom and filter, then details on demand」は必ずしもあてはまらないことに着目した。そこで本研究では、多数の変数間の関係を網羅的に表し、かつ、第三者（分析者）の認知的負荷の軽減にもつながる、できるだけシンプルな提示方法が必要なことから、Seoらが提案するフレームワーク rank-by-feature framework[Seo2005]が有用であると考えた。

Rank-by-feature frameworkでは、探索型可視化を行うユーザが、多変量・多次元データから興味深い仮説を導き出せるよう、1次元（ヒストグラムや箱髭図等）、あるいは、2次元（散布図等）でデータが表現される。また、複数の視点からデータの全体像を素早く把握し、系統立った可視化が可能となるよう、データへの順位付け(rank)が行われる。順位付けには、データ分布の正規性や均一性、外れ値や特異値など、データの特徴(feature)を表す統計量が利用される。その一方で、3次元表現など、ユーザの認知的負荷が大きく、解釈やデータ操作がしばしば困難となる表現形式は使用されない。

本研究では、スナップショットの探索型可視化において、ソフトウェアプロジェクトトモグラフィを構成する5つの観点（要求、作業、組織、プロダクト、課題）を特徴(feature)と考え、それらを表す統計量によってスナップショットの構成要素やその解析結果の順位付けを行う。例えば、「プロダクト」の観点であれば、モジュールの複雑さ、変更量、クローン値などにもとづいてスナップショットの構成要素やその解析結果を順位付けすることになる。特徴として用いる5つの観点は、ソフトウェア開発者・管理者にとってなじみ深いものである。従っ

て、可視化結果の直感的な理解や解釈を可能にし、仮説の真偽や適用範囲の確認も容易になると考えた。

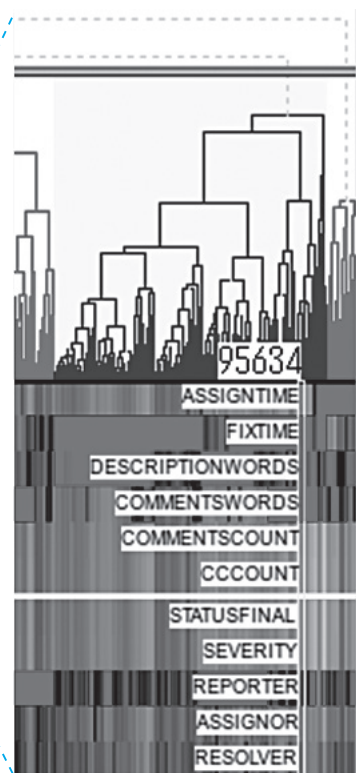
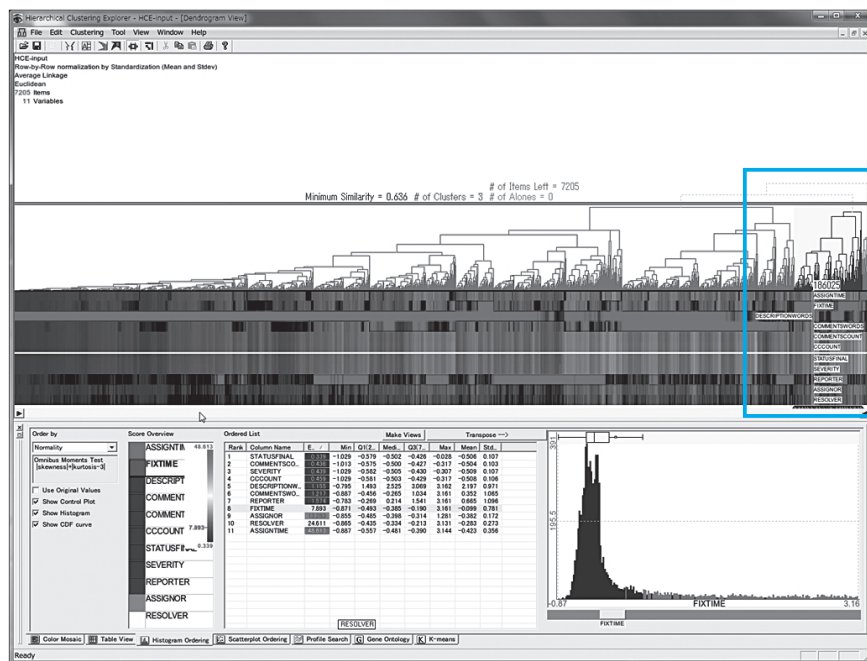
3.2. HCE (Hierarchical Clustering Explore)

本研究では、rank-by-feature framework[Seo2002]を実装した探索的データ解析ツール HCE (Hierarchical Clustering Explore)[Seo2002][Seo2007]を用いた被験者実験をおこない、第三者評価における探索型解析技術の有用性を検証する。以下では、HCE ツール（図1）の概要と利用用法について述べる。

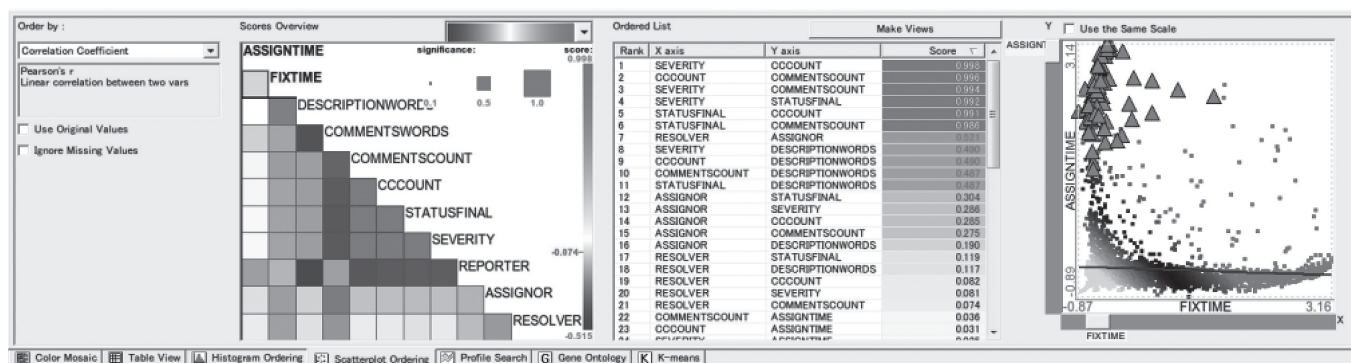
HCE ツールは、入力されたスナップショットデータから、類似するデータ同士をグルーピングする階層的クラスタリング機能（図1-(a)の画面上部）、クラスタ内のデータの分布を把握するヒストグラム作成機能（図1-(a)の画面下部、Histogram Ordering）、分析者が注目したクラスタの任意の2変数間の関係を把握するための散布図作成機能（図1-(b)、Scatterplot Ordering）、クラスタ内のすべての変数間の関係を把握するための平行座標プロット機能、クラスタ内のデータの分布を把握するヒストグラム作成機能（図1-(c)、Profile Search）などから構成されている。これら豊富な探索型可視化分析機能により、HCE ツールは、生物学分野における遺伝子解析やプロテオーム解析（タンパク質の機能・構造の大規模解析）など [Cluzet][Tsai] で数多くの利用実績がある。

分析者はまず、階層的クラスタリング機能を用いていくつのクラスタを作成するかを指定する。分析対象データは通常、多変量データであるが、変数間の類似性が階層図の下側に色の濃淡で表現されている（図1-(a)の画面上部）ため、容易にデータのまとまりを識別できるようになっている。注目するクラスタを選択（図1-(a)の画面上部の3つのクラスタの内、真ん中のクラスタ）すると、選択したクラスタに対応するデータの散布図やヒストグラムが図1-(a)の画面下部に自動的に表示されるため、一画面の中で探索的にデータを分析することができる。特に、相関関係の強い（または弱い）2変数をランク付けする（例えば、図1-(b)の色の濃淡による相関係数の表示方法）ことで、分析対象に精通していない、あるいは、分析のエキスパートでない分析者にも重点的に分析すべき対象が容易に選択できるようにしている点に大きな特徴がある。

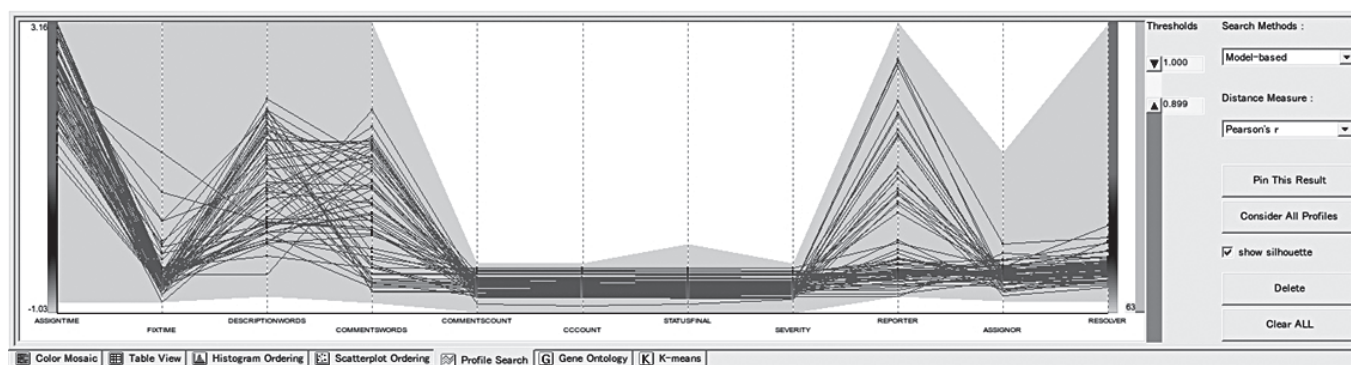
また、HCE ツールでは、ユーザがデータセットを準備するのを助けるために、タブ区切り、コンマ区切りのテキストファイルの他、MS Excel ファイルを入力ファイル



(a) ツールの概観と選択中のクラスタ (拡大図) (画面下部は Histogram Ordering タブ選択時の表示例)



(b) Scatterplot Ordering タブ



(c) Profile Search タブ

図 1 HCE ツール [Seo2002][Seo2005]

として利用できる。一行目に変数名、二行目にデータの型を記述し、三行目以降にデータを配置した入力ファイルを作成すれば分析を始めることができるため、スナップショットデータとの親和性も非常に高い。

4. 検証実験

4.1. 実験概要と目的

探索型解析技術として、rank-by-feature framework にもとづく探索型可視化方式を用いることが、ソフトウェア品質の第三者評価、特にその初期段階における探索的データ解析に効果があるかどうかを調べるための検証実験をおこなった。

検証実験では、2種類の実験をおこなった。まず、OSS 開発を熟知しているグループとそうでないグループの2群に分け、Eclipse Platform プロジェクトの課題（不

表1 実験に用いた課題データに含まれる変数（項目）

項目（変数）	詳細
AssignTime	修正者が決定されるまでの時間
Fixtime	修正されるまでの時間
Resolution	課題の最終状態 (RESOLVED, VERIFIED など)
Severity	重要度
CCCount	メーリングリストの登録者数
Comments	コメントの件数
CommentWords	コメントの総単語数
DescriptionWords	記述情報の単語数
Reporter	報告者の名前
Triager	管理者の名前
Fixer	修正者の名前
Component	不具合が発生されたコンポーネント名

表2 仮説生成実験 I (HCE ツール使用) の結果

被験者群	被験者	仮説生成数	仮説生成数 (平均)	重要な仮説の数	重要な仮説の数 (合計)
グループ A	A-1	5	6	0	4
	A-2	6		1	
	A-3	7		3	
グループ B	B-1	2	2.7	0	2
	B-2	3		1	
	B-3	3		1	

表3 仮説生成実験 II (MS Excel 使用) の結果

被験者群	被験者	仮説生成数	仮説生成数 (平均)	重要な仮説の数	重要な仮説の数 (合計)
グループ C	C-1	3	4.3	0	1
	C-2	3		0	
	C-3	7		1	
グループ D	D-1	2	4.7	0	0
	D-2	4		0	
	D-3	8		0	

具合) データを HCE ツールで分析させた (検証実験 I)。さらに、知識の有無だけではなく、MS Excel 等の一般的な表計算アプリケーションを用いた場合との効果の違いを明らかにするために、検証実験 I と同様の実験をおこなった (検証実験 II)。

実験の目的は、Rank-by feature framework に基づく探索型可視化分析が仮説生成を支援できるかどうかを確認することである。具体的には、(1) 分析対象のドメイン知識の有無が生成される仮説数にどれくらいの影響を与えるのか、(2) 被験者が生成した仮説が国際会議等ですでに発表された知見に含まれるか (重要な知見につながる仮説を導けたか) どうかを、被験者実験により検証した。

4.2. データセットと変数

実験では、Eclipse Platform プロジェクトの課題（不具合) データから、Eclipse 3.2 に関連する約 7,200 件をデータセットとして収集した。本データセットに含まれる 11 の変数を表 1 に示す。

4.3. 被験者

検証実験 I では、OSS 及びデータマイニングに関連する研究を行っている大学院生 3 名をグループ A とし、それらに精通していない大学生 3 名をグループ B として被験者を構成した。

検証実験 II においても同様に、OSS 及びデータマイニングに関連する研究を行っている大学院生 3 名をグループ C とし、それらに精通していない大学生 3 名をグループ D として被験者を構成した。

いずれの実験においても、被験者の重複はない (合計 12 名)。なお、被験者の大学院生は、本実験で対象とする OSS プロジェクトの課題データで用いられる変数の意味について、(個人差はあるものの) 同程度の知識があると見なすことができる。一方、両実験の大学生は、情報科学分野で 3 年以上在籍し、履修科目の 1 つとしてソフトウェア工学を学んだ経験があるものの、OSS 及びデータマイニングに関してはほとんど知識のない被験者である。その意味で、実験課題に対しては同程度の知識量であったと見なすことができる。

4.4. 実験手順とタスク

実験に先立ち、被験者には HCE ツールの利用方法と使用する変数 (メトリクス) について説明した後、食品に含まれる栄養素をまとめたサンプルデータセット (本実験には無関係のドメインにおけるデータ) を与え、HCE

ツールを約1時間試用してもらった。HCE ツールの利用に大きな支障がないことを確認した後、前述したデータセットを与えて実験を開始した。各被験者は隔離され、グループ内の被験者同士で相談等ができない環境でタスクに取り組んだ。実験時間は1時間とした。

被験者には、データセットに含まれる興味深い関係や傾向を事実としてまず抽出し、抽出した事実からプロセス改善や品質改善につながる案を仮説として生成するよう指示した。また、仮説を思いついた時点でメモとして記録しておくよう依頼した。被験者の様子は実験者が記録するとともに、画面キャプチャツールを用いて被験者の画面操作を記録した。

4.5. 実験結果

- ・ 分析対象のドメイン知識の有無が生成される仮説数にどれくらいの影響を与えるか？

表2から、HCE ツールを用いた場合では、ドメイン知識の有無で生成できる仮説の数に違いがあることが見て取れる。一方、表3から、MS Excel を用いた場合では、グループ内での個人差が見られるものの、多くの仮説を生成できる被験者はグループC、グループDのどちらにも1名存在しており、ドメイン知識の有無によって生成できる仮説の数に違いがないことが分かる。また、ドメイン知識のないグループBとグループDが生成した仮説数は、MS Excel を用いたグループDの方が多いことが見て取れる。

- ・ 被験者が生成した仮説が国際会議等ですでに発表された知見に含まれるか？

被験者A-3が3件の重要な仮説を生成しているため、表2の結果のみで結論付ける事は困難であるが、表2の重要な仮説の合計から、HCE ツールを用いた場合では、ドメイン知識の有無が生成できる重要な仮説の数に違いを生むことが確認できる。ただし、ドメイン知識のないグループBは、生成できた仮説数はすべてのグループの中で最も少ないものの、重要な仮説を2件生成できている。表3から、MS Excel を用いた場合では、生成できた仮説数はグループC,Dともに比較的多いものの、重要な仮説はグループCが1件生成できたのみであった。

検証実験I及びIIの結果から、HCE ツールを用いた場合と、MS Excel を用いた場合とで、生成できる仮説数自体には大きな違いはなかったが、仮説の質の点では大きな違いがあることを確認できた。また、HCE を用いたグループ間にはドメイン知識の差が仮説生成の数と質の差として現れることが確認できた。一方、MS Excel を用い

た場合では、ドメイン知識と仮説生成数とに関連は見当たらなかった。次章では、実験中の被験者の様子から本実験結果について考察する。

5. 考察

5.1. HCE ツールの利用

検証実験Iの結果から、ドメイン知識を有した被験者が数多くの仮説を生成できることが分かった。ただし、ドメイン知識のない被験者であっても、一人平均2.7件の仮説を生成でき、さらに、『「課題が報告されてから担当者に割り当てられるまでの時間」と「課題が割り当てられてから解決されるまでの時間」は反比例する。』といった趣旨の、すでにソフトウェア工学の研究分野で検証済みの重要な法則を仮説として生成できることが分かった。

HCE ツールを用いた被験者はまず、幾つかのクラスタを生成するところから分析を開始する。例えば、本論文の実験で用いたEclipse Platformの不具合データを可視化している図1-(a)では、3つのクラスタが生成されていることを示している。生成された個々のクラスタは、データの特徴量が類似するデータの集合を意味する。したがって、被験者が着目(選択)したクラスタを分析する時点で、すでに幾つかの変数の間には何らかの強い関係が存在することになる。図1-(a)で選択中のクラスタ(真ん中のクラスタ)と、左右の2つのクラスタとの明らかな違いは、AssignTimeとFixTimeのスコア(値)の大きさから見て取れる。例えば、現在選択中のクラスタとその右側のクラスタとでは、AssignTimeとFixTimeのスコアの取り方に真逆の関係(一方が大きな値で一方が小さな値)があることを見て取ることができる。すなわち、AssignTimeとFixTimeには負の相関が存在する事を示しており、『「課題が報告されてから担当者に割り当てられるまでの時間」と「課題が割り当てられてから解決されるまでの時間」は反比例する。』という関係[Ohira]を導き出す事ができる。

HCE ツールが提供する(画面下部で利用する)各種分析機能は、注目するクラスタ内の変数間の関係を詳細に調べる過程を支援するため、MS Excel を用いた被験者と比べ、仮説生成の質に違いが現れたものと考えられる。

表4に検証実験Iで被験者が生成した仮説の内、重要な知見につながる仮説を示す。HCE ツールを用いた被験者は、特定のクラスタやクラスタ内の変数間の関係に着目して分析していることが見て取れる。これらの仮説

は、ソフトウェア工学の研究分野においても重要性がすでに確認されており、支援手法についても多数提案されている [Anvik][Breu][Jeong][Ohira][Zimmerman] ものである。これらの結果からは、rank-by-feature framework に基づく探索型可視化方式は、探索的データ解析においてドメイン知識の不足する分析者であっても、課題対応プロセスの実施・適用技術の妥当性評価につながるものと考えることができる。

5.2. MS Excel の利用

HCE ツールを用いた被験者が重要な仮説を合計 6 件生成できたのに対して、MS Excel を用いた被験者は、生成できた仮説数自体は多いものの、重要な仮説を 1 件しか生成できなかった。

被験者の画面操作をキャプチャしたデータから、被験者は主に、SUM, AVERAGE, AVERAGEIF, COUNTIF などの関数、相関係数を求めるアドオン機能を使用し分析を行っていたことが分かった。また、興味のある 1 つの変数に着目し、着目した変数と他の変数との関係を 1 つ 1 つ調べるといった方法で分析を行っていることも分かった。HCE ツールを用いた分析では、選択したクラスタが

表 4 生成された重要な仮説

被験者群	事実	仮説
グループ A	AssignTime が短いときは FixTime にばらつきがあるが、AssignTime が長ければ FixTime が短くなっている。	慎重に不具合を割り当てることが、プロダクトの品質改善につながる [Anvik].
	不具合が修正者に割り当てられるまでの時間 (AssignTime) は、管理者 (Triager) によって異なる。	修正依頼までに要する時間を短縮しようとしているプロジェクトとして、管理者の選出は非常に重要な要素となる [Ohira].
	修正依頼時間 (AssignTime) と修正時間 (FixTime) がともに長いクラスタに着目すると、descriptions の単語数 (DescriptionWords) が多い傾向にある。	不具合をより正確に報告することは、不具合の修正時間を改善するのに役立つ [Breu].
グループ B	コメント数 (Comments) と descriptions の単語数 (DescriptionWords) の散布図から、ほとんどコメントがなく単語数も少ないクラスタに分布する不具合は、コメント回数/単語ともに多い不具合に比べて、修正されるのが遅い (FixTime)。	不具合報告の情報は、不具合修正プロセスを改善するために重要である [Breu].
	多くのコメントが付けられた (Comments) 不具合報告は、Resolution が RESOLVED や VERIFIED となっている。	不具合を確実に修正してもらうためには、開発者がコメントの内容を理解できるよう明瞭に不具合の内容を記述する必要がある [Zimmerman].
グループ C	管理者 (Triager) のヒストグラムを見ると、少数の管理者に不具合を多数割り当てていることが分かる。また、それらは修正されるのが遅くなる (FixTime) 傾向にある。	こうした事態を防ぐためには、適度な負荷分散をおこない、不具合修正プロセスを改善する必要がある [Jeong].
	コメント数が 10 件以上ある案件は AssignTime が長い。コメント数が 10 件以上ある案件は FixTime が短い。	複数人がとりかかる必要のある問題は修正を依頼する人を決定するのに時間がかかるが、活発な議論が繰り広げられる環境では早く問題が解決する [Ohira].

表 5 被験者 C1 が生成した仮説

仮説番号	事実	仮説
C1-1	不具合の深刻度 (Severity) が上がるほど、バグレポート当たりの CommentsCount が増える	開発者は、バグの深刻度が高いレポートに関心を持つ
C1-2	不具合の深刻度が上がるほど、バグレポート当たりの Description words が増える	説明文の長いバグレポートは、深刻度の高いバグを報告している
C1-3	不具合の深刻度と CommentsCount の相関係数は -0.11 であり、ほぼ相関がない。	開発者がバグレポートに興味を持つ要因は、バグの深刻度ではない。

「なぜクラスタとしてまとまりを持つのか？」を考えるために、すべての変数間の関係を俯瞰的に調べる機能 (図 1-(b),(c) など) が利用できるのに対して、MS Excel を使った分析では、まず興味のある変数に着目し、その変数に関係する変数を見つけるのが分析の主眼となっていることが分かった。例えば、多くの被験者が課題の重要度 (Severity) に着目していたが、重要度そのものがデータセット全体の中で重要な変数ではない (課題の多くは重要度が適切に付与されていない) ため、結果的に重要な仮説を生成するのまでには至らなかったものと思われる。

特に、注目した変数と他の変数との関係を機械的に調べる傾向が強く、表 5 に挙げた被験者 C1 の仮説 C1-1 及び C1-2 のように、被験者自身が矛盾する仮説に気付いていないケースが多々見られた。このような傾向は、MS Excel を用いた検証実験 II だけではなく、検証実験 I においても共通して見られた傾向であった。したがって、このような傾向は、探索的データ分析をおこなう際の共通課題であるとも考えられる。探索的データ分析ツールを用いる際に、手本となる仮説生成までの手順をいくつか用意しておくことにより、今回見られた傾向を大きく

改善できる可能性がある。

また、MS Excel を用いた被験者が重要な仮説を導きだせなかったその他の理由として、データの特徴量の類似度を基にデータのまとまりを構成するクラスタリング分析とは異なり、MS Excel を用いて SUM や AVERAGE 関数を用いて分析を行っても、対象プロジェクトにおける問題や課題を抽出するのが本質的に困難であった可能性がある。プロジェクトにおける問題や課題は、プロジェクトに所属する開発者（第一人者）さえも認識しづらいものと考えられる（プロジェクト全体で認識された課題や問題であれば、すでに克服されている可能性が高い）ためである。したがって、SUM や AVERAGE 関数を用いてデータセット全体の傾向から分かる課題や問題は、プロジェクトにとって本質的なものではない場合が多く、MS Excel を用いた被験者のほとんどが重要な仮説につながる事実を抽出することができなかったものと思われる。

6. まとめ

本論文では、ソフトウェア品質の第三者評価における探索的データ解析ツールの利用とその効果について報告した。第三者が効率的に探索的データ解析を行えるようにするための探索型解析技術の選定をおこない、rank-by-feature framework [Seo205] に基づく探索型可視化方式を採用した。また、rank-by-feature framework を実装した探索的データ解析ツール HCE (Hierarchical Clustering Explore)[Seo2002][Seo2007] を用いた被験者実験をおこない、第三者評価における探索型解析技術の有用性を検証した。検証実験の結果から得られた知見は以下の通りである。

- rank-by-feature framework に基づく探索的可視化方式は、ソフトウェア品質の第三者評価、特に初期段階での探索的データ解析を支援することができる
- rank-by-feature framework に基づく探索的可視化方式によるデータ分析は、一般的な表計算アプリケーションでおこなうデータ分析に比べ、プロセスの実施・適用技術の妥当性評価につながる重要な仮説を導きだすのに有用である

本論文における検証実験は、オープンソースプロジェクトの課題データを用いておこなった。今後は、ソフトウェアプロジェクトトモグラフィにおけるスナップショットが提供する5つのペイン（構成画面）のデータすべてを用いて、プロジェクト及びプロダクトの異常検出や品質評価が行えるかどうかを、ソフトウェア開発企

業のデータを用いておこなっていく必要がある。特に、第三者評価の観点から探索型可視化技術の有用性を検証するために、大学の研究者等が生成した仮説と、実際に開発に携わった管理者・開発者などの実務者が生成した仮説がどの程度一致するかを確認することが求められる。

謝辞

本研究は、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 (SEC: Software Reliability Enhancement Center) が実施した「2012年度ソフトウェア工学分野の先導的研究支援事業」の支援を受けたものです。本研究の被験者実験には、奈良先端科学技術大学院大学情報科学研究科の皆様、和歌山大学システム工学部の皆様に御協力頂いた。

【参考文献】

- [Anvik] J. Anvik, L. Hiew, G.C. Murphy, "Who should fix this bug?," Proceedings of the 28th international conference on Software engineering (ICSE'06), pp. 361-370, 2006.
- [Breu] S. Breu, R. Premraj, J. Sillito, T. Zimmermann, "information needs in bug reports: improving cooperation between developers and users," Proceedings of the 13th ACM conference on Computer supported cooperative work (CSCW'10), pp. 301-310, 2010.
- [Cluzet] S. Cluzet, C. Torregrosa, C. Jacquet, C. Lafitte, J. Fournier, L. Mercier, S. Salamagne, X. Briand, M. T. Esquerre-Tugaye, and B. Dumas, "Gene expression profiling and protection of *Medicago truncatula* against a fungal infection in response to an elicitor from green algae *Ulva* spp.," Plant, Cell and Environment, Vol.27, pp.917-928, 2004.
- [IPA] 情報処理推進機構, "ソフトウェアの品質説明 力強化のための制度フレームワークに関する提案 (中間報告)" (平 23-9).
- [IPA2012] 2012年度ソフトウェア工学分野の先導的研究支援事業「ソフトウェア品質の第三者評価のための基盤技術-ソフトウェアプロジェクトトモグラフィの開発-」成果報告書, <http://www.ipa.go.jp/files/000026806.pdf>
- [Jeong] G. Jeong, S. Kim, T. Zimmermann, "Improving bug triage with bug tossing graphs," Proceedings of the 7th joint meeting of the European software engineering conference and the ACM SIGSOFT symposium on The foundations of software engineering (ESEC/FSE'09), pp. 111-120, 2009.
- [Ohira] M. Ohira, A.E. Hassan, N. Osawa, K. Matsumoto, "The Impact on Bug Management Patterns on Bug Fixing: A Case Study of Eclipse Projects," Proceedings of the 28th International Conference on Software Maintenance (ICSM'12), pp. 264-273, 2012.
- [Seo2002] J. Seo, B. Shneiderman, "Interactively Exploring Hierarchical Clustering Results," IEEE Computer, Volume 35, Number 7, pp. 80-86, 2002.
- [Seo2005] J. Seo, B. Shneiderman, "A rank-by-feature framework for interactive exploration of multidimensional data," Information Visualization, Vol.4, No.2, pp.96-113, 2005.
- [Seo2007] J. Seo and H. Gordish-Dressman, "Exploratory Data Analysis with Categorical Variables: An Improved Rank-by-Feature Framework and a Case Study," International Journal of Human-Computer Interaction, Vol.23, No.3, pp.287-314, 2007.
- [Tsai] J.-M. Tsai, H.-C. Wang, J.-H. Leu, H.-H. Hsiao, A. H. J. Wang, G.-H. Kou, C.-F. Lo, "Genomic and proteomic analysis of Thirty-nine structural proteins of shrimp white spot syndrome virus," Journal of Virology, Vol.78, pp.11360-11370, 2004.
- [Tukey] J.W. Tukey, "Exploratory Data Analysis," Addison-Wesley, 1977.
- [Zimmerman] T. Zimmermann, R. Premraj, N. Bettenburg, S. Just, A. Schroter, C. Weiss, "What Makes a Good Bug Report?," IEEE Transactions on Software Engineering (TSE), Vol. 36, No. 5, pp. 618-643, 2010.