OSS開発における共進化を定量的に分析するための データマイニング手法

山谷 陽亮[1,a)]　大平 雅雄[1,b)]

概要：本研究の目的は，OSS 開発における共進化の過程を明らかにすることである．そこで本稿では，共進化の過程の定量的な分析を支援するために，遅延相関分析の考え方に基づいたデータマイニング手法を提案する．

1. はじめに

近年オープンソースソフトウェア (OSS) を用いたソフトウェア開発が主流となりつつある．OSS をソフトウェア開発に応用することで，システムの低価格化・短納期化が期待できるが，システム開発を請け負うベンダー企業が OSS を利用したシステム開発に対して不安を抱いているのも事実である．

独立行政法人情報処理推進機構 (IPA) による調査[*1] によると，58%の企業が「利用している OSS がいつまで存続するかわからないこと」や，43%の企業が「バグの改修や顧客からの要請対応に手間がかかること」を OSS を利用したシステム開発に関する懸念事項として挙げている．つまり，OSS をシステム開発に利用する企業は，プロジェクトの開発・保守が盛んに行われる，いわゆる成功したプロジェクトを選択し，活用すると考えられる．

こうした懸念事項を払拭することや，成功したプロジェクトの要因を明らかにするために，OSS の進化に関する研究が盛んに行われてきた．ここでソフトウェアの進化とは，ソフトウェアが外部環境やユーザーのニーズに合わせて変化していくことを指している [3].

OSS の進化の過程を理解することによって，開発者がプロジェクトを成功へと導くために行うべきことや，新規プロジェクトが発展するかどうかなどを明らかにすることができると考えている．

[1] 和歌山大学
Wakayama University
[a)] s151049@sys.wakayama-u.ac.jp
[b)] masao@sys.wakayama-u.ac.jp

[*1] 「第 3 回オープンソースソフトウェア活用ビジネス実態調査」(有効回答数 700 社以上，2009 年度の調査結果)

2. OSS 開発における共進化

OSS の進化に関するこれまでの研究は，アーティファクト（ソースコードやドキュメントなどのプロダクト)，開発者，コミュニティの 3 つの系統が独立して進化すると考えられ，分析がされてきた．こうした研究の例として，Linux カーネルの規模推移をサブシステムごとに調査し，サブシステムの進化について分析した研究 [1] などが挙げられる．

Ye らは，OSS の進化の過程をより正確に捉えるためには，この 3 つの系統が互いに影響を与えながら進化していくことを考慮する必要があると考え，各系統の進化が互いに影響を与えながら進化することを共進化 (co-evolution) と呼んでいる [2].

図 1 は OSS 開発における共進化のモデル [2] を模式的に示したものである．共進化のモデルとは，コミュニティが進化（組織構造が複雑化するなど）することによってアーティファクトが進化（ソースコードの規模が増加するなど）し，アーティファクトが進化することによって開発者が進化（開発者自身のスキルが身につくなど）していくことなどを指す．

しかし，Ye らはこの共進化のモデルを定性的に示したも

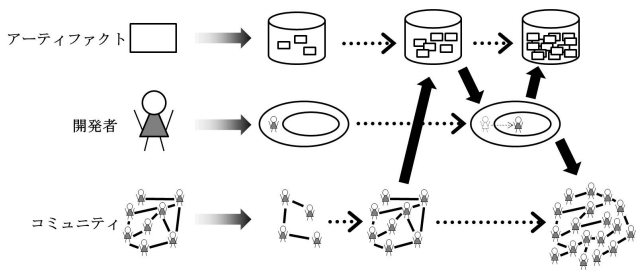

図 1 OSS 開発における共進化の概念（[2] を参考に改編）
Fig. 1 The concept of OSS co-evolution

のの，定量的な分析は行っていない．

そこで共進化の過程を定量的に分析することのできるデータマイニング手法を提案する．共進化という現象は，すぐに現れるわけではなく，一定期間後に現れるものであると考えられる．したがって，一定期間後の相関も抽出することのできる，遅延相関分析を用いることとした．

3. 遅延相関分析に基づくマイニング手法

3.1 遅延相関分析の概要

遅延相関分析とは，竹内ら [4] によって提案され，時系列で表現される説明変数と目的変数との相関関係を明らかにするための手法である．

図 2 は，遅延係数分析の概念図であり [5]，e_i は時刻 i における説明変数の値を表している．ある加算係数 (i　j) における説明変数の値の累積値を e_{ij} とすると，e_{ij} は (1) 式で定義される．

$$e_{ij} = e_i + e_{i-1} + \times\times\times + e_j \tag{1}$$

また，r_n は時刻 n における目的変数の値であり，ある差分係数 (n　m) における目的変数の値の変化値を r_{nm} とすると，r_{nm} は (2) 式で定義される．

$$r_{nm} = r_n \quad r_m \tag{2}$$

相関係数 c_{er} は (3) 式で定義され，S_e および S_r はそれぞれ e_{ij}, r_{nm} の標準偏差，S_{er} は e_{ij}, と r_{nm} の共分散である．

$$c_{er} = \frac{S_{er}}{S_e \quad S_r} \tag{3}$$

このように遅延相関分析は，説明変数の値が累積し，一定期間の遅延をもって目的変数の値の変化に影響を与えると想定した相関分析手法である．

3.2 遅延相関分析を用いた提案手法

提案手法は遅延相関分析の考え方に基づき，説明変数と目的変数に，アーティファクト，開発者．コミュニティの 3 つの系統のいずれかを割り当て．それぞれの系統の進化を表現するメトリクス（例えば，アーティファクトの場合はソースコードの行数など）を用いる．

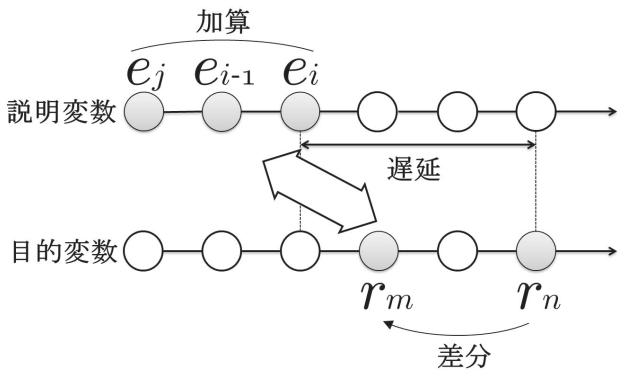

図 2 時系列データ処理の概念（[5] を参考に改編）
Fig. 2 Processing time-series data

また，説明変数の値が累積し，一定期間の遅延をもって目的変数の値の変化に影響を与えるということを表現するために，加算係数（説明変数の値を累積させた期間），差分係数（目的変数の値の変化を考慮する期間），遅延係数（説明変数が目的変数に影響を与えるまでの期間）の 3 つのパラメータを定義する．

提案手法では，この 3 つの各パラメータの最適値を相関係数が最大となるよう自動的に求める処理が含まれるところに特徴がある．そのため，収集したデータを提案手法を用いて解析することで，分析者は遅延の大きさや分析窓の大きさを気にせず，相関係数が大きくなるメトリクスの組み合わせのみを分析対象とすることができる．

このように提案手法を用いることによって，メトリクス同士の関係（例えば，開発者の数が増加するとソースコードの規模が増加するなど）を明らかにすることができ，OSS 開発における共進化の過程を定量的に分析することができると期待できる．

4. おわりに

本研究では，OSS 開発の共進化の過程を定量的に分析するために遅延相関分析に基づくマイニング手法を提案した．今後は提案手法を用いて複数のプロジェクトにおいて分析を行い，提案手法の有用性を確かめるとともに，OSS 開発の共進化の過程を明らかにしていきたい．

謝辞　本研究の一部は，文部科学省科学研究補助金 (基盤 (B):23300009) および (基盤 (C):24500041) による助成を受けた．

参考文献

[1] Godfrey, M. W. and Tu, Q.: Evolution in Open Source Software: A Case Study, *Proceedings of the International Conference on Software Maintenance (ICSM'00)*, pp. 131–142 (2000).
[2] Ye, Y., Nakakoji, K., Yamamoto, Y. and Kishida, K.: *The Co-Evolution of Systems and Communities in Free and Open Source Software Development*, chapter 3, pp. 59–82, Idea Group Publishing (2004).
[3] 大森隆行，丸山勝久，　林晋平，沢田篤史：ソフトウェア進化研究の分類と動向，コンピュータソフトウェア，Vol. 29, No. 3, pp. 168–173 (2012).
[4] 竹内裕之，児玉直樹：生活習慣と健康状態に関する時系列データ解析手法の開発，*Proceedings of the 3th Forum on Data Engineering and Information Management(DEIM'08)* (2008).
[5] 黛　勇気，竹内裕之，児玉直樹：生活習慣と健康状態の時系列データ解析における重み付けの検討 (I) -日毎の任意係数による重みづけ-，*Proceedings of the 3th Forum on Data Engineering and Information Management(DEIM'11)* (2011).