バグレポート生成間隔系列を生成するモデル構築に向けた一考察

眞鍋 雄貴[1,a)]

概要：本稿では，オープンソースソフトウェアの開発データ分析により得られた知見の一般化を検討するためのデータ生成を目指し，Eclipse と Netbeans のバグレポート生成間隔について特徴を分析し，線形回帰によるモデル構築を試行した．それらの結果より，課題を示す．

Toward Building a Model for Representing Sequence of Bug Report Generation Interval

1. はじめに

オープンソースソフトウェアの開発ではバグレポートなどのデータ（開発データ）が生成され，それらを分析することにより多くの知見が得られている [1]．しかし，知見を得るために選択されたプロジェクトがオープンソースソフトウェア全体を代表していることを示すのは難しい．また，特性の違うオープンソースソフトウェアプロジェクトでも得られた知見が当てはまるのか検討するのが難しい．

そこで，オープンソースソフトウェアの開発データがどのように生成されるかの統計モデルを構築することによって，これらの問題を解決することを目指す．統計モデルを構築するためには，モデルを選択し，実際のプロジェクトを分析することでモデルのパラメータを決定する．例として，線形回帰モデルの場合，各説明変数に対する重みと切片がパラメータとなる．このパラメータとプロジェクトの特性の関係を結びつけることで，多種多様なプロジェクトを模倣することができる．また，複数のプロジェクトから得られたパラメータから，オープンソースソフトウェア全体でのパラメータの分布を推定し，その分布に従って多数の模擬的なプロジェクトを構築することで，プロジェクトの分析により得られた知見が一般化できるか検討するデータを得ることができる．

そこで本稿では，予備的な考察として，Eclipse と Netbeans のバグレポートを用い，それらのバグレポート生成間隔がどのようになっているのかを調べることでバグレポートの生成間隔が持つ特徴を調べる．また，線形回帰モデルを用いて，生成間隔をモデル化して得られた結果から考察を行う．

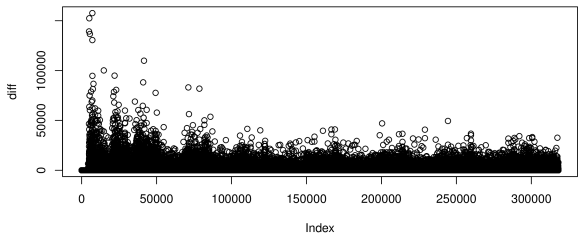
2. ケーススタディ

本章では，Mining Software Repository 2011 で行われた Mining Challenge のために公開されているデータ[*1]のうち，Eclipse(316911 件，2001〜2010 年) と Netbeans（185578 件，1998〜2010 年）のバグレポートを用いてモデルを構築し，考察を行う．各バグレポートについては，id と報告時刻を用い，報告時刻をバグレポート生成時刻とした．なお，欠損している *id* を持つバグレポートの生成時間は，前後のバグレポートの生成時間から線形補間により補完した．

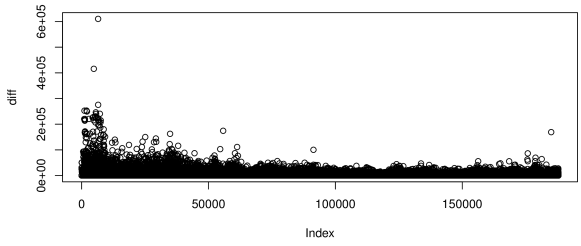
本ケーススタディにおける評価基準を設定するため，本稿で構築することを目的としているモデルに求められる特性について述べる．一つは，異なるプロジェクトに同じモデルを適用し，プロジェクトごとにパラメータを選択することでそのプロジェクトのデータに適合することが望ましい．そのため，各プロジェクトで一定の精度が確保される必要がある．もう一つは，元になったデータの特徴が，構築されたモデルから生成されたデータからも確認できることである．このような特徴は得られる知見とも関係するため，既存研究の評価も難しい．

[1] 熊本大学
2-39-1, Kurokami, Kumamoto, Kumamoto, 860-8555, Japan
[a)] y-manabe@cs.kumamoto-u.ac.jp

[*1] http://2011.msrconf.org/msr-challenge.html

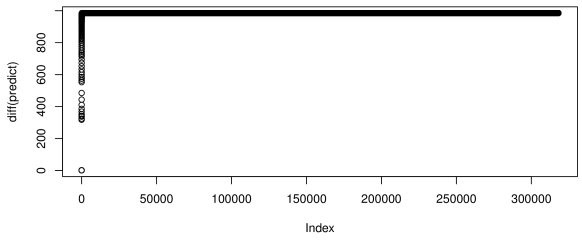


(a) Eclipse

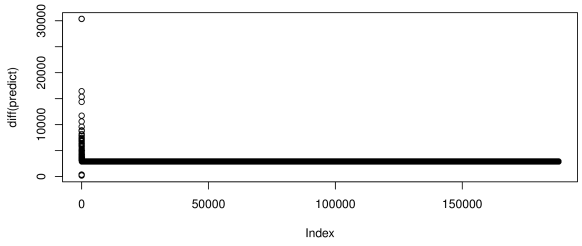


(b) Netbeans

図 1 バグレポート生成間隔



(a) Eclipse



(b) Netbeans

図 2 逐次予測により得られた生成間隔

2.1 データの観察

Eclipse のバグレポート生成時間の間隔を**図 1(a)**，Netbeans の場合を**図 1(b)** に示す．図から，バグレポートの生成間隔には複数のピークがあることがわかる．また，これらのピークは外れ値となるものであった．そのため，このようなピークを再現できるようなモデルが必要となる．

2.2 予測モデル構築の試行

得られた生成間隔データから予測モデルを構築した．予測モデルとして，線形回帰モデルを用い，目的変数として id が t であるバグレポートが生成されてから次の $t+1$ のバグレポートが生成されるまでの時間 T_t とし，説明変数を $T_{t-1}, T_{t-2}, ..., T_{t-10}$ とする．バグレポートの数を m とするとき，$\left(T_1, ..., T_{\lfloor \frac{m-1}{2} \rfloor}\right)$ を学習データ，$\left(T_{\lfloor \frac{m-1}{2} \rfloor+1}, ..., T_{m-1}\right)$ を評価データとした．評価尺度として，予測値と実際の値の差の絶対値が実際の値に占める割合の平均である MMRE を用いた．結果として，Eclipse が MMRE=0.767，Netbeans が MMRE=0.766 であった．同一のモデルで，近い MMRE を出せていることから，様々なプロジェクトで同一のモデルを利用できる可能性がある．一方で，どちらも MMRE が大きいため，線形回帰モデルは不適切であることも示している．

次に，各モデルを用いて，$T_{10}, T_9, ..., T_1$ を初期値として逐次的に id11 以降の生成間隔　を求めたものを**図 2** に示す．これらの結果から，**図 1** にも現れている最初のピークについては，特徴を捉えられている．一方，それ以外の特徴は出ておらず，知見の検討に向いていないモデルであると言える．

2.3 関連研究

Yu ら [2] は，クローズされたバグレポート数等の分析に時系列分析を用いている．本稿が対象とするバグレポートの生成間隔では時系列分析の方が適切である可能性がある．

実証的研究におけるサンプリングの問題を扱った研究として，Nagappan ら [3] は，選択した OSS プロジェクトの coverage を計測する手段と，coverage を最大化するためのプロジェクトの選択アルゴリズムを提案している．本稿の目指すオープンソースソフトウェア全体の推定ができているかを評価するために使える可能性がある．

3. まとめ

本稿では，バグレポートの生成間隔系列の生成について，モデルを構築できるかの考察を行った．今後の課題として，データの特徴をより分析し，モデルを選択すること，プロジェクトの特性との関係がある．

参考文献

[1] 伊原彰紀，大平雅雄：『オープンソースソフトウェア工学』シリーズオープンソースソフトウェア工学，コンピュータソフトウェア， Vol. 33, No. 1, pp. 28–40 (2016).

[2] Yu, L., Ramaswamy, S., Lenin, R. B. and Narasimhan, V. L.: Time Series Analysis of Open-source Software Projects, *Proc. ACM-SE 47*, pp. 64:1–64:6 (2009).

[3] Nagappan, M., Zimmermann, T. and Bird, C.: Diversity in Software Engineering Research, *Proc. ESEC/FSE 2013*, pp. 466–476 (2013).