wakayama univ.

Are 20% of Files Responsible for 80% of Defects?

Neil Walkinshaw, Leandro Minku

発表者： 和歌山大学 池内未来

背景

- 先行研究でパレートの法則に従うことが示されている[1]
  - 少数のソース非公開ソフトウェア
  - 複数のファイルにおいて同じ欠陥を修正した場合，別々の欠陥としてカウント

[1] Carina Andersson and Per Runeson. 2007. A replicated quantitative analysis of fault distributions in complex software systems. IEEE Transactions on Software Engineering 33, 5 (2007), 273–286.

目的・アプローチ

- 目的
  - 多数のシステムでパレートの法則が適用されるかの分析
- アプローチ
  - 100のGitHubリポジトリを分析
  - ファイル，メトリクス（LOC，Code churn）および欠陥修正の関係の調査

Research Questions

- RQ1：パレートの法則はソフトウェアの欠陥に適用されるか？
- RQ2：欠陥が混入しやすいファイルは，基本的なメトリクスによって容易に識別できるか？
- RQ3：１つの欠陥を修正するのに複数のファイルを変更しなければならないとしたら，それらの変更は欠陥が発生しやすい20%のファイルに集中しているか？

RQ1の結果

- パレートの法則は適用される

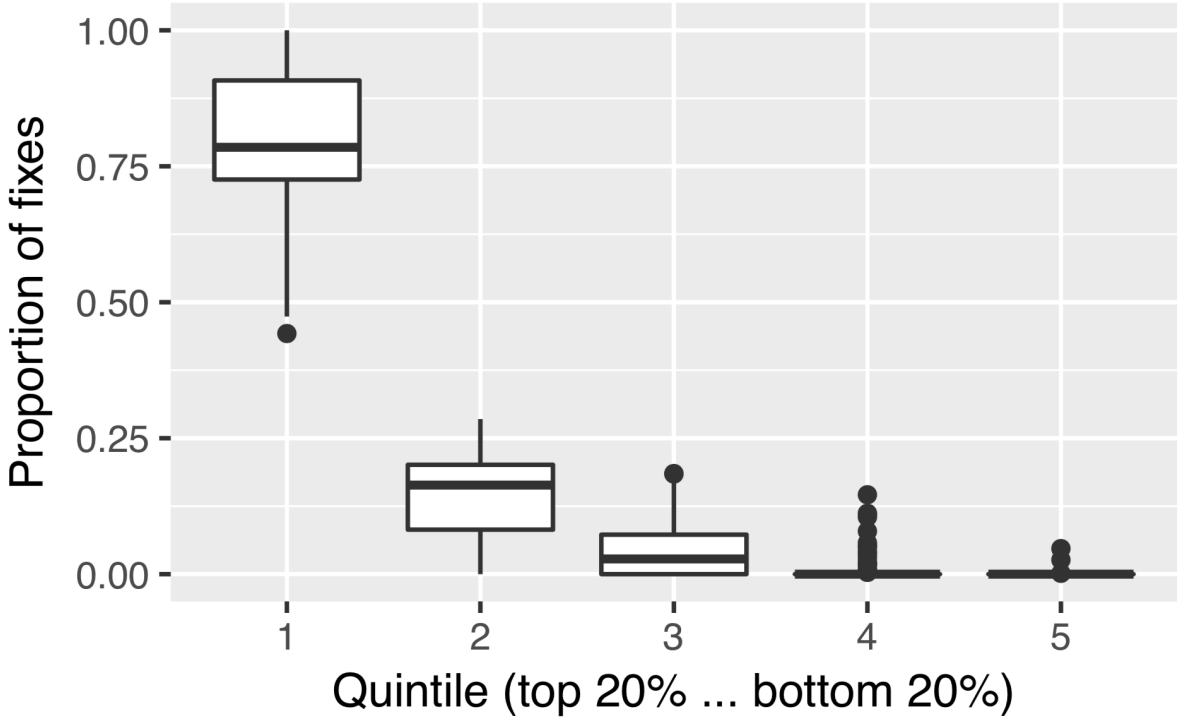


Figure 2: Proportion of fixes involving files in each quintile (where quintile 1 represents the top 20% of the files etc.).

RQ2の結果

- LOCよりもChurnは信頼性が高い傾向にある

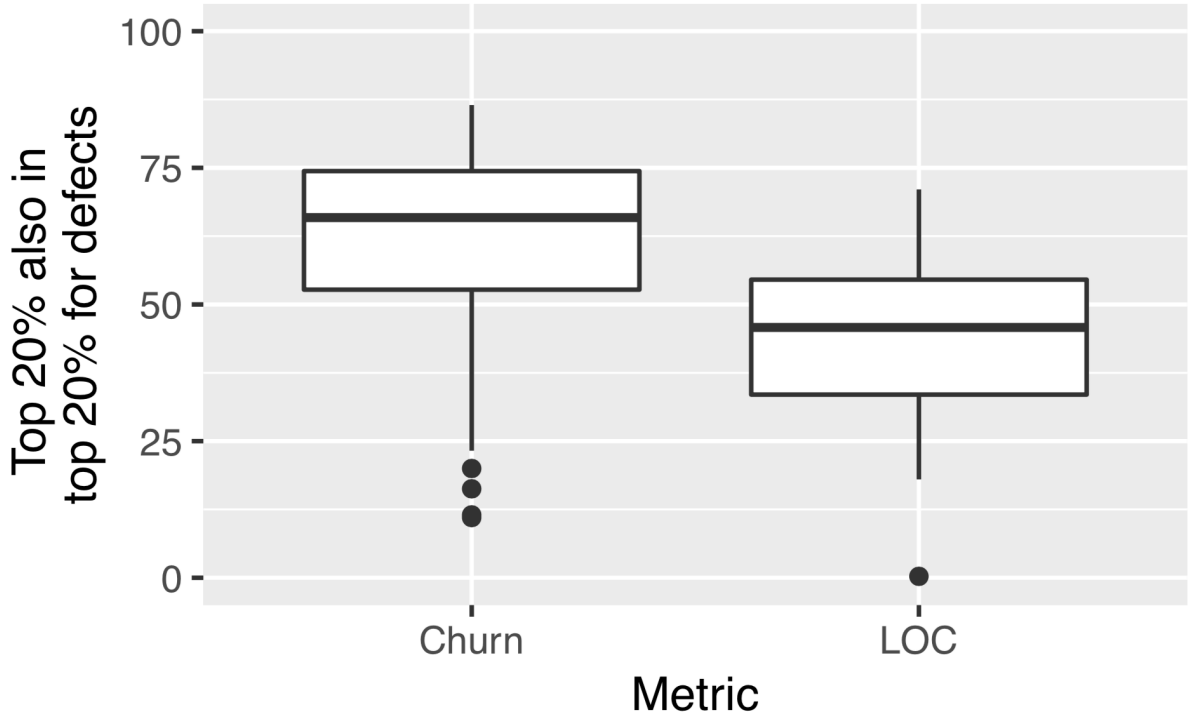


Figure 4: Proportion of fixes belonging to top quintiles that also belong to top quintile of defect-prone files.

RQ3の結果

- 上位20%のバグを含むファイルを修正したとき，それに伴い修正されるファイルの半分以下が上位20%に含まれる

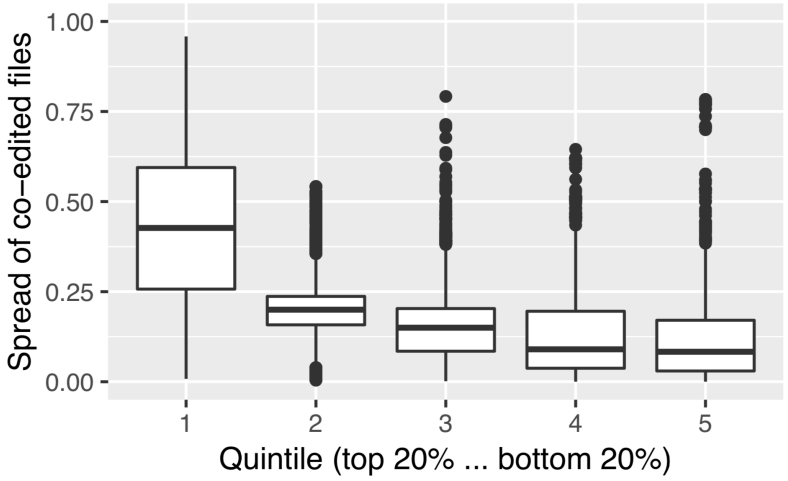
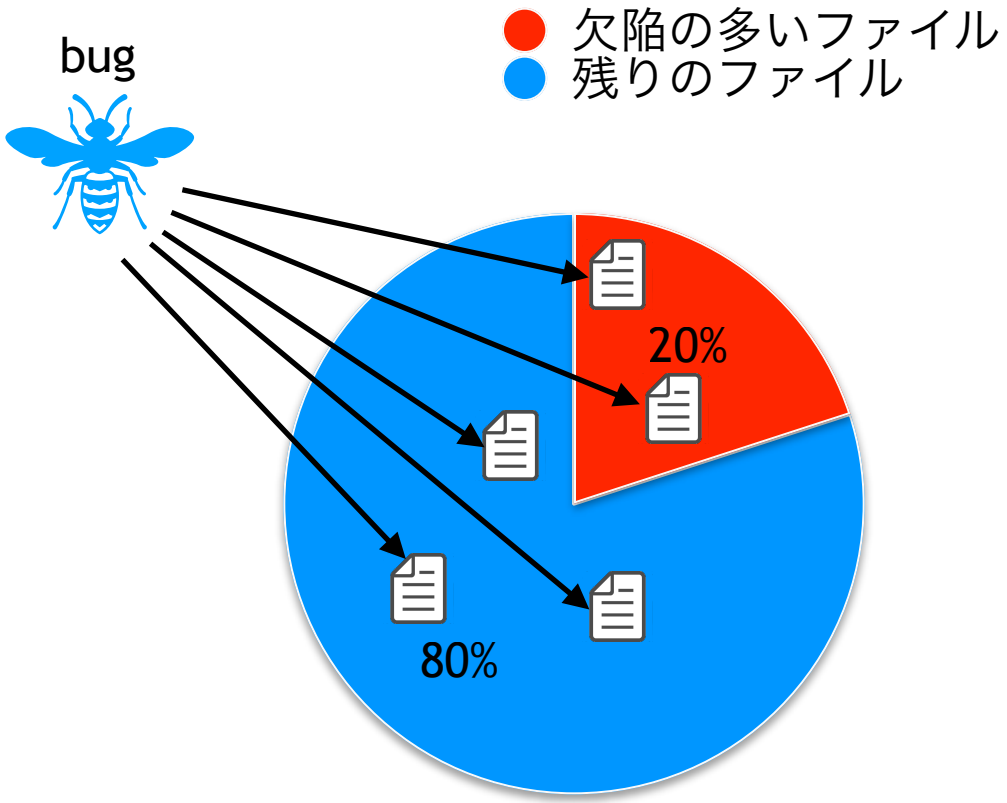


Figure 6: Spread of files co-edited with files in the top quintile of defect-prone files.



結論・所感

- 結論
  - 多数のシステムでもパレートの法則は適用される
  - メトリクスから上位20%を識別するのは容易ではないが，Code churnの方がより信頼性が高い
- 所感
  - 図の詳しい説明がなくて読むのが難しかった
  - RQの結果がわかりにくかった