Assignment 1 – Paper 1
Mining Software Repositories (IN4334)
2015-2016

Luís Filipe Correia Cleto
4495101

Tiago Almeida Fernandes
4486722

November 15, 2015

Paper Analysed

Śliwerski, J., Zimmermann, T., & Zeller, A. (2005). When do changes induce fixes?. ACM sigsoft software engineering notes, 30(4), 1-5.

Research Question(s)

The main research question presented in the paper is: Which properties of changes may induce fixes?

Dataset Used

For conducting their research, the authors of the paper chose to use log messages from CVS and BUGZILLA archives relative to the Mozilla and Eclipse projects, containing all changes up to January 20, 2005. This accounts for 78954 and 109658 transactions for each project, respectively.

Sampling Mechanism

The sampling mechanism is random since the analysis was conducted on bugs for which the cause and the fix were detected through the developed detection mechanism. This accounts for 47% of fixed bugs in Eclipse and 55.30% for Mozilla.

Type of Study

The study conducted is observational and retrospective since the data is collected in a way that does not interfere with how the data arises and the study uses only past data. This does not allow causal connections to be made (only correlations) but does allow the conclusions to be generalized (random sampling but no random assignment).

Main Strengths

- The underlying concepts are thoroughly explained or referenced (CVS, BUGZILLA, …)
- The paper is original in providing a method for detecting fix-inducing changes
- The methods used for detection of fixes or fix-inducing changes are thoroughly explained, allowing the study to be reproduced.
- The conclusions are relevant and allow to correlate transaction size, nature (fix or new feature) and day of the week with the probability for inducing fixes.

Main Weaknesses

- The accuracy rate for detecting fix-inducing changes is not measured.

Assignment 1 – Paper 2
Mining Software Repositories (IN4334)
2015-2016

Luís Filipe Correia Cleto
4495101

Tiago Almeida Fernandes
4486722

November 15, 2015

Paper Analysed

Bacchelli, A., D'Ambros, M., & Lanza, M. (2010). Are popular classes more defect prone?. In Fundamental Approaches to Software Engineering (pp. 59-73). Springer Berlin Heidelberg.

Research Question(s)

The research questions presented in this paper are:

Q1. Does the popularity of software components in discussions correlate with soft- ware defects?
Q2. Is a regression model based on the popularity metrics a good predictor for software defects?
Q3. Does the addition of popularity metrics improve the prediction performance of existing defect prediction techniques?

Dataset Used

For conducting their research, the authors of the paper chose to use code from SVN/CVS repositories, mails and bugs from the associated mailing lists and bug tracking systems relative to the projects Equinox, Jackrabbit, Lucene ands Maven.

Sampling Mechanism

The sampling mechanism was random with the intention to create 50 random 90%-10% splits of the data in order to divide it in training and validation set.

Type of Study

The study conducted is observational and retrospective since the data is collected in a way that does not interfere with how the data arises and the study uses only past data. This does not allow causal connections to be made (only correlations) but does allow the conclusions to be generalized (random sampling but no random assignment).

Main Strengths

- General organization is excellent.
- The paper is original trying to find correlation between mailing lists and number of bugs in classes.
- Through the whole paper there is a logical line around the research questions.
- Every steps of the observation are thoroughly explained.

Main Weaknesses

- A formula explaining how Pearson's and the Spearman's correlation coefficients are calculated would be benefic to comprehend what their values represent.