# Assessment of Bug Prediction Approaches

Prudhvi Ratna Badri Satya
*A02057243*
*Computer Science*
*Utah State University*

***Abstract*—As a part of modern era in technology, software systems have turned into the essential parts of web. Their prosperity has been trailed by a developing convergence of defects. Defect predictors are useful apparatuses for researchers and developers. Exact defect predictions may help in decreasing test times and prompts higher quality codes. Dependably foreseeing programming defects is one of the heavenly chalices of software Engineering. Analysts have formulated and executed a plenty of defect/bug prediction methodologies fluctuating regarding exactness, intricacy and the input they require. Notwithstanding, the nonappearance of a created benchmark makes it hard, if not incomprehensible, to stand up in comparison approaches. I assessed five datasets, to be specific Equinox, Jdt, Lucene, Mylyn and Pde and each of these have three documents which consists of metrics. I have performed information examinations on the defect information utilizing a few defect prediction routines. In view of the outcomes, I talk about in the forthcoming subjects about the most astounding connection to the quantity of post-release defects, best metric in every document and how they correlate with each other. Underneath, I have demonstrated the prediction performance on several regression models. In light of the outcomes, I have examined underneath about the execution and soundness of the methodologies regarding our examination and conclude various bits of knowledge on bug prediction model.**

***Keywords*—Bug Prediction, Bug Analysis, Bug Metrics, Regression Models**

## I. Introduction

Software frameworks will be getting to be progressively complex and are being utilized all over. In this way, the nature of framework matters and there are numerous troublesome issues to be tended to. Taking after two issue viewpoints are the principle issues which are to be determined. Foreseeing the quantity of defects in the framework and assessing the unwavering quality of the framework regarding time to failure. The greater part of them have been assessed in disconnection, or were contrasted with just few different methodologies.

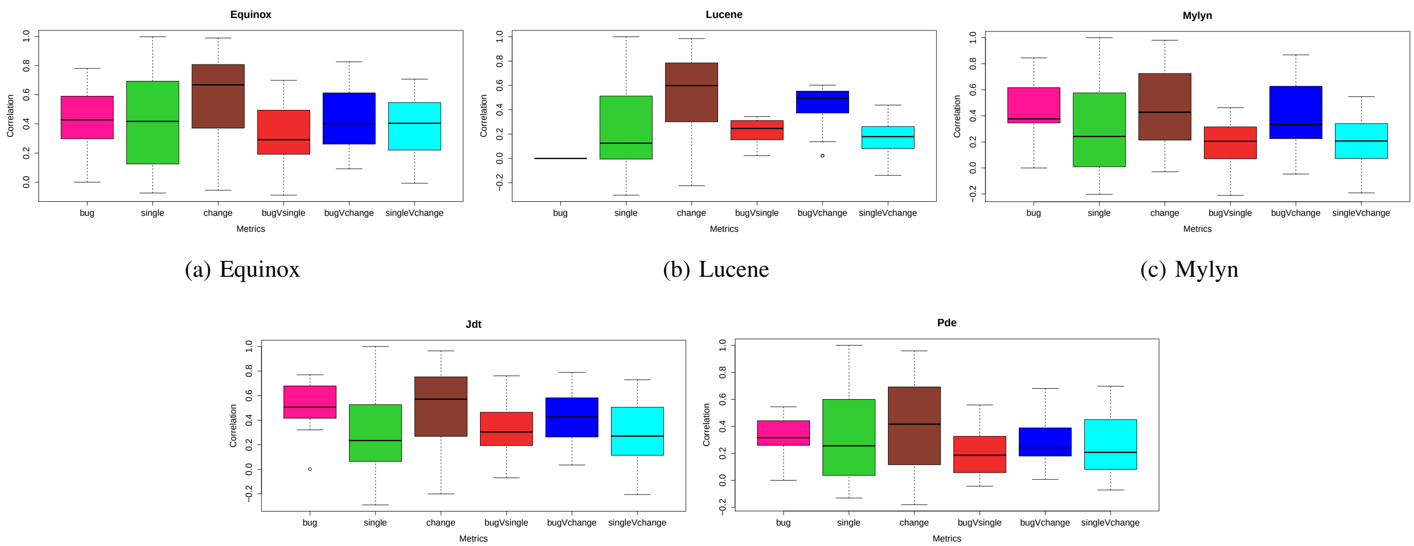
Several approaches to have proposed to defect prediction, the sort of information they require and the different data sets on which they were approved. All methodologies require a deformity chronicle to be approved, however don't essentially oblige it to really perform their analysis.In view of these, I performed data analysis and machine learning for each dataset. With the information assessment I discovered the most elevated connection to the quantity of post-release defects and afterward distinguished best metrics. With these outcomes, I have finished up how these metrics correlate to one another. The prediction performance of these measurements on a few regression models specifically linear regression model, tree regression model and negative binomial regression model are been clarified. To measure these, I have utilized cross-validation system by giving the training data and taking into account these, the performance is predicted for the remaining data. Executed the mean absolute error and assessed overall prediction performance. From the outcomes, the highest prediction performance is shown using the combined metrics and finally shown the highest performance model.

Structure of the paper: The association of the paper is as per the following: Section 2 clarifies about the methodology. Section 2A examines the metrics which have the best correlation to defects? Section 2B examines how the metrics correlate to each other? Segment 2C examines how well each of the best metrics found in section 2A predicts defects? Section 2D examines about the prediction performance of the metrics when combined with one another and examine about the performances of each of that models. Section 3 examines about the results and discussions of this paper. Section 4 finishes up with the conclusion.

## II. Methodology

### A. Generation of metrics that have best correlation to defects.

In this section we are going to find the best metrics which have high correlation with the defects. The approach followed in order to find the best metrics is to compute the linear (i.e. Pearson) and ranked (i.e. Spearman) correlations of each metric to the number of post-release defects (provided in column "bugs") and compare them. We compute the correlation in terms of the Spearmans correlation coefficient and Pearson correlation coefficient. The Spearmans rank correlation test is a non-parametric test that uses ranks of sample data consisting of matched pairs. The correlation coefficient varies from 1, i.e., ranks are identical, to -1, i.e., ranks are the opposite, where 0 indicates no correlation. The Pearson linear correlation is a measure of the linear correlation (dependence) between two variables X and Y, giving a value between +1 and 1 inclusive, where 1 is total positive correlation, 0 is no correlation, and 1 is total negative correlation. It is widely used in the sciences as a measure of the degree of linear dependence between two variables. In order to find the best metrics present in metrics files like single-version-ck-oo, change metrics and bug metrics of the five datasets the average correlation for various metrics are computed. Then paired T-tests are performed for comparing the top two metrics as computed using Spearman and Pearson correlation. The p-values that are obtained from the paired T-tests are used to validate our results.




(a) Equinox (b) Lucene (c) Mylyn

(d) JDT (e) PDE

Fig. 1: Correlation between metrics

### B. *Correlation between the metrics*

In this section, the correlation between the various metrics is analyzed. The metrics from each of the five datasets containing the files single-version.ck-oo, change metrics and bug metrics are taken into account. Then the different aspects within the same metric files and the aspects within one metric file with other metric files in dataset are analyzed using box plots. This is done to verify the assumption whether the metrics for the same aspects (e.g. metrics on single-versions code) are more correlated than ones for different aspects.

### C. *Assessment of best metrics to predict defects*

In this section, the discussion will be based on to check how will the best metrics found in the section 2A will be able to predict the bugs. In this context we will perform the cross-validation, for every random split, by using the training set (80% of the dataset) to build the regression model, and after that we apply the acquired model on the validation set (20% of the dataset), Using the best metrics as predictors which were computed in section 2A. At that point, to assess the performance of the performed performance, we process Spearman's correlation, and the mean absolute error between the actual value and the predicted value, on the validation set, between the lists of classes positioned by predicted and real number of post-release defects. Since we perform 30 folds cross-validation, the last estimations of the Spearman's correlation averages more than 30 folds. (7243 is used as the set of random seed in this paper).The prediction performance of each of the best metrics are verified on each of the accompanying regression models: 1) Linear regression model. 2) Tree regression model. 3) Negative binomial regression model. Then boxplots are plotted for the visual illustrations of the results obtained.

### D. *Analysis with Combination of metrics as predictors*

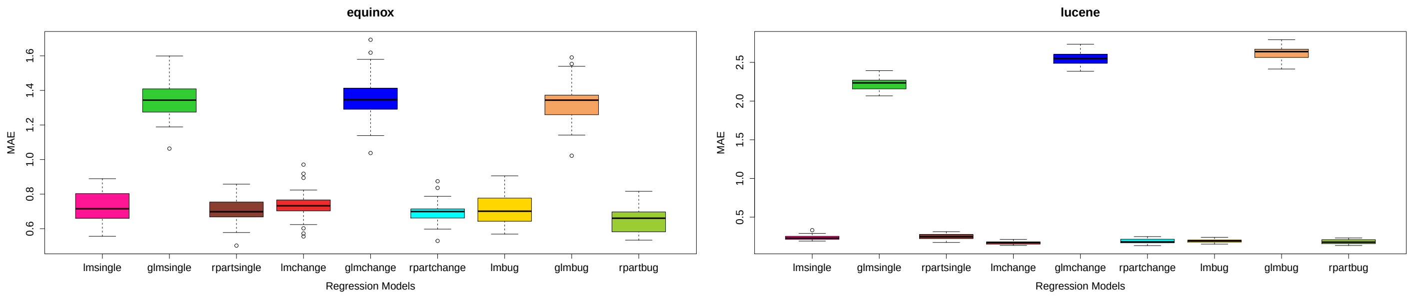
In this section, we are going to discuss whether the combination of the all available metrics as predictors computed over all the three regression models will enhance the prediction performance of the defect prediction system that has been developed. Since the metrics are highly correlated, PCA (Principal Component Analysis) is used to address this situation.

In the next section the various results are analysed and discussed.

## III. Results and Discussion

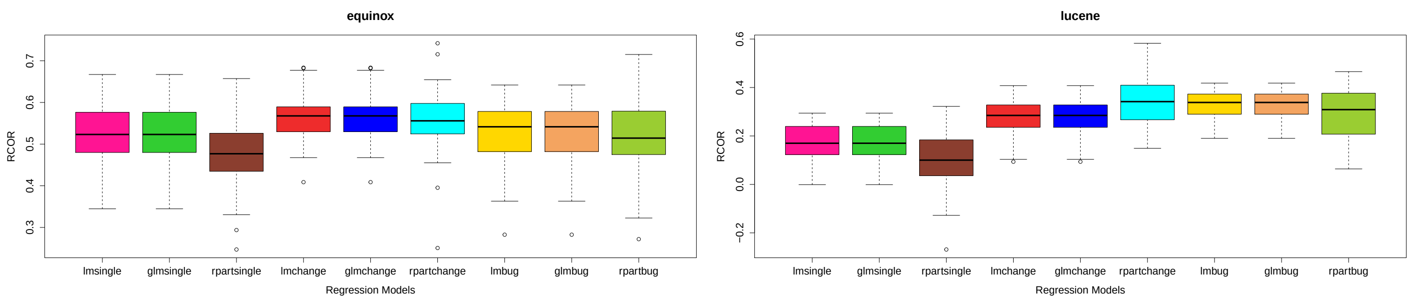
The results in the Table-1 shows the best metrics obtained after calculating the average correlation for single-version-ck-oo, change-metrics and bug-metrics files on all the five data sets using the Spearman correlation and the Pearson correlation. The linesAddedUntil, numberOfLinesOfCode and numberOfNonTrivialBugsFoundUntil are the best metrics that are obtained after calculating the average correlation for the change-metrics, single-version-ck-oo and bug-metrics on all five data sets using spearman method. The numberOfVersionsUntil, rfc and numberOfBugsFoundUntil are the best metrics that are obtained after calculating the average correlation for the change-metrics, single-version-ck-oo and bug-metrics on all five data sets using Pearson method.

Paired T-tests are performed in order to validate the results obtained in Table 1. The resulting p-values are present in Table-2. It is very clear from Table-2 that metrics computed and obtained from spearman method and Pearson method are valid




(a) Equinox (b) Lucene

Fig. 2: Mean absolute error for various Regression models of best metrics in Equinox and Lucene datasets




(a) Equinox (b) Lucene

Fig. 3: Ranked Correlation for various Regression models of best metrics in Equinox and Lucene datasets

| **Metrics** | **Spearman** | **Pearson** |
|---|---|---|
| **change-metrics** | linesAddedUntil | numberOfVersionsUntil |
| **single-version-ck-oo** | numberOfLinesOfCode | rfc |
| **bug-metrics** | numberOfNonTrivialBugsFoundUntil | numberOfBugsFoundUntil |

TABLE I: Best metrics obtained using spearman and pearson method

| | *Spearman* | *Pearson* |
|---|---|---|
| **Metrics** | **P-Value** | **P-Value** |
| Single-Version-ck-oo | 0.6665 | 0.6935 |
| Change-metrics | 0.3359 | 0.07031 |
| Bug-metrics | 0.3908 | 0.3763 |

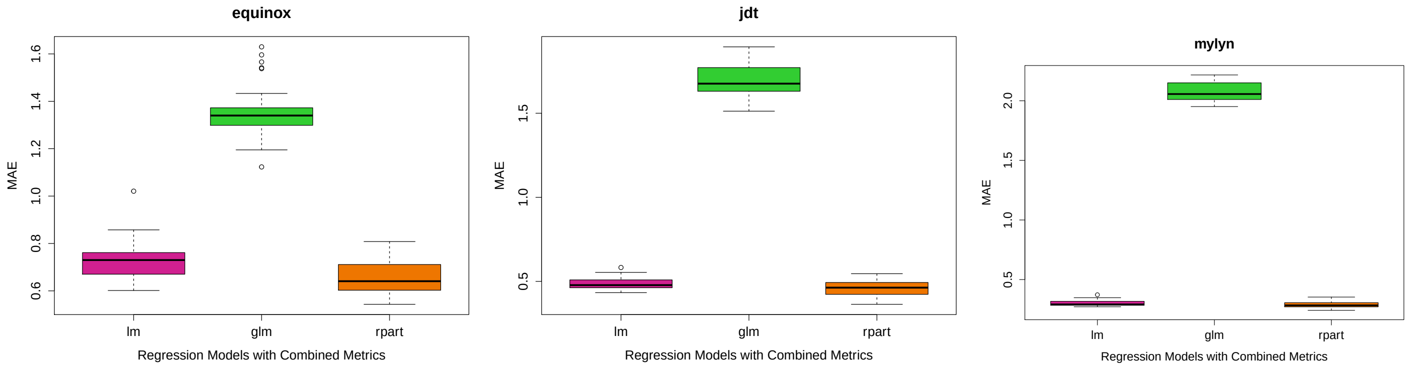
TABLE II: P-values obtained from paired T-tests

since the p-values are greater than 5% as computed for best two metrics in files of single-version-ck-oo, change-metrics and bug-metrics.

*Correlation between metrics* The Fig.[1] shows the box plots for the correlation betweeen metrics of all the five data sets.This boxplots clearly verifies the assumption that metrics for the same aspects (e.g. bug-metrics Vs bug-metrics) are more correlated than ones for different aspects (e.g. bug-metrics Vs single-version-ck-oo) since the values are very closer in the range of median and the boxplot size.

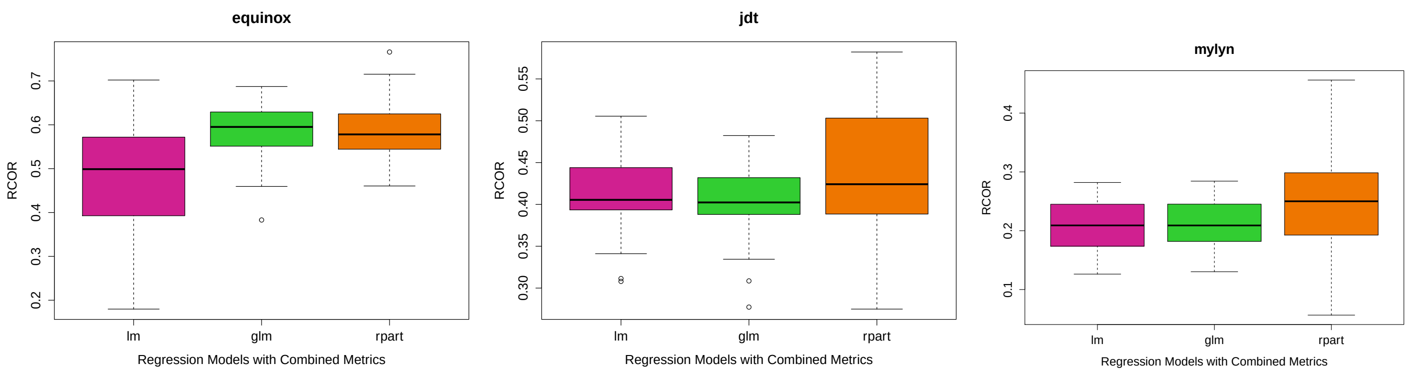
*Assessment of best metrics to predict defects* : The Fig.[2] and Fig.[3] shows the boxplots for the mean absolute error and ranked correlation of best metrics in Equinox and Lucene data sets for various regression models namely linear regression model, tree regression model and negative binomial regression model. From the Fig[2] , it is clear that the mean absolute error for linear regression model and tree regression model is low for the metrics in these data sets and the values for the negative binomial regression model is very high for the metrics. From the Fig[3] , it is clear that the ranked correlation for tree regression model is low for the metrics in these data sets and the values of correaltion for the linear and negative binomial regression model is high for the metrics. This shows that when the best metrics are used the prediction model does not show much major improvement in the prediction of defects.

*Analysis with Combination of metrics as predictors*: In the




(a) Equinox (b) JDT (c) Mylyn

Fig. 4: Mean Absolute Error for various Regression models of combined metrics in Equinox, Jdt and Mylyn




(a) Equinox (b) JDT (c) Mylyn

Fig. 5: Ranked Correlation for various Regression models of combined metrics in Equinox, Jdt and Mylyn

Fig.[4] the boxplots represents the mean absolute error for the combination of the metrics and in Fig.[5] the boxplots represents the ranked correlation using the linear regression model, tree regression model and negative binomial generalized regression model for Equinox, Jdt and Mylyn datasets. The Fig.[4] clearly shows that the linear and tree regression model have low mean absolute error when compared to that of the negative binomial regression model. The Fig.[5] clearly shows that the tree regression model have high ranked correlation when compared to that of the other two regression models. This shows that the combination of the metrics has an advantage of predicting the bugs since the correlation is enhanced when compared to using the best metrics.

## IV. Conclusion

This paper discusses about the prediction performance of the best metrics using various regresion models such as linear regression model, tree regression model and negative binomial generalized regression model.In the wake of experiencing the entire software analysis, I have inferred that we can discover best fit classifier just by considering the combinations of metrics.

*Acknowledgements*

I sincerely thank Dr.Tung Nguyen for his valuable support through out this project. I also acknowledge other students in this class for their extended help in this project.

## References

[1] Dongsung Kim, Sunghun Kim, *Where Should We Fix This Bug? A two Phase Recommendation Model*

[2] Alberto Bacchelli , Marco DAmbros and Michele Lanza, *Are Popular Classes More Defect Prone?*

[3] Macro D Ambros, Michele Lanza, Romain Robbes *An Extensive Comparision Of Bug Predicting Approaches*