Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 11, Number 1 (2015), pp. 437-462

http://www.ripublication.com

Evaluating The Impact Of Binary And Multinomial LR In Developing Software Fault Prediction Model Using OO-Metrics

Dipti Kumari and Kumar Rajnish

Research Scholar, PhD, BIT, Mesra, Ranchi, Jharkhand-835215 (India)
kumari_dipti0511@yahoo.co.in
Assistant Professor, BIT, Mesra, Ranchi, Jharhand-835215 (India)
krajnish@bitmesra.ac.in

Abstract:

The study used software metrics effectiveness in developing models in two aspects (Binary and Multinomial) and in 2 different ways Bivariate (taking the confounding effect of class size with individual metrics) and Multivariate (combined effect of object-oriented metrics) for finding the classes in different error categories for the three versions of Eclipse, the Java-based open-source software. Bivariate models are not showing the best result for both aspects, but multivariate models are showing the best result for both aspects. Among multivariate models, multinomial aspect is showing the best result. So, during development of a model, we need to include more than two metrics and among that one of the metric must size metric. As we see from the data, distribution of fault among module of a software system is not uniform, in that case multinomial aspect helps the tester to prioritize the tests with the knowledge of error range and category and therefore, work more efficiently. We conclude that prioritizing the test on the basis of different number of errors in different error categories is more efficient than simply knowing the category fault prone and fault free classes for developing fault prediction models.

Key-words:- OO-metrics ,Fault-prone, prediction, logistic regression, multinomial, LR, BBLR, BMLR, MBLR, MMLR,MLE,LL.

1. Introduction

Error-free Software development is the main aim of every software employee (i.e. software developer or tester). But, all their effort for that become in vain because of presence of errors still in the software and are detected after it has been released. Errors need to be found while the software is being developed during the software development process only. If the errors are found late then the corrective action is very expensive and leads to a new release of the software. We can try to locate the occurrence of errors and remove them. Any information that can help to locate the error can increase the testing [1] efficiency. Software metrics can be of great help in this scenario. There are numerous metrics proposed in the literature to capture the quality of object-oriented (OO) design and code, for example [4, 6, 9, 16, 19-20] .These metrics provide ways to evaluate the quality of software and their use in earlier phases of software development can help organizations in assessing a large software development quickly, at a low cost. But how do we know which metrics are useful in capturing the important quality attributes such as fault proneness, effort, productivity or amount of maintenance modifications. An empirical study of real systems only provides relevant answers. There have been empirical studies evaluating the impact of OO metrics on software quality and constructing models that utilize them in predicting quality attributes in the system such as [2-3, 5, 7-8, 10-12, 15 ,17-18] .More data-based empirical studies, which are capable of being verified by observation or experiment are needed. The evidence gathered through these empirical studies are today considered to be the most powerful support possible for testing a given hypothesis, and can then be applied for identifying potentially faulty-classes in future applications or future releases. The usage of design metrics allows the organization to take justifying actions early and consequently avoid costly rework. The study is divided into following part:

- The descriptive statistics[14] for each metric(proposed metric also) is presented to compute low variance metrics that do not differentiate classes very well and, therefore, are unlikely to be useful.
- The principal-component (PC) method [14] is used to determine whether these metrics are independent or are capturing the same underlying property of the object being measured.
- The Correlation of OO-metrics with pre-release error and size is used to determine whether these metrics are able to discriminate between error-free and error-prone and these metrics are related with size.
- After that Bivariate analysis is carried out to answer that the relationship between the object-oriented metric and fault-proneness will be inflated due the effect of size. Without controlling for the confounding effect of class size one obtains results that are systematically optimistic. It is therefore necessary to control class size to get accurate results. We have done this analysis in both aspects binary and multinomial.
- Finally, four prediction model are developed and tested using the bivariate and multivariate logistic regression (LR) technique in both aspects Binary and Multinomial for the Eclipse versions[13][22].

In this paper we report on a study that was performed to construct a model to predict which classes in a future release of a commercial Java application will be faulty. The model uses only object-oriented design metrics. Our empirical validation results indicate that the model has high accuracy in identifying which classes will be faulty and in predicting the overall quality level. On the basis of these results, it is reasonable to claim that such a model could help for planning and executing testing by focusing resources on the fault-prone parts of the design and code.

The article is organized as follows: Section 2 summarizes the metrics studied, describes sources from which data are collected and presents hypothesis to be tested in the study. Section 3 presents the research methodology and model evaluation criteria followed in this article. In Section 4, the results of the study are given. The model is evaluated in Section 5. Conclusions and future scope of the study are presented in Section 6.

2. Research Background

In this section, we present the summary of metrics studied in this article (Section 2.1), empirical data collection (Section 2.2), and hypothesis to be tested in our work (Section 2.3).

2.1. Dependent and Independent Variables

Fault proneness is defined as the probability of fault detection in a class. We used Logistic regression technique, to measure the probability of fault proneness. For LR, independent and dependent variable are required. The dependent variable in this study is fault proneness and independent variables are OO-metrics. The OO-metrics of coupling, cohesion, inheritance and size are the independent variables used in this study, which is usable at early stages of software development.

2.2 Empirical Data Collection

To analyze metrics chosen for this work, their values are computed for the selected Data Source.

The following steps are followed to do so:

i. To select the Data Source.

ii. To select the software metrics.

iii. Associate each Java class with the number of pre-release error data.

2.2.1 To select the Data source

This study makes use of the data collected from three major releases, 2.0, 2.1, and 3.0, of Eclipse. As an open-source Java project, Eclipse is comprised of extensible frameworks, tools and runtimes for building, deploying and managing software across the entire software lifecycle.

Table 1: Data Source information

Version	Released date	# of java files	Total size
Eclipse 2.0	27 June 2002	6751(5686 cd+1065 id)	796 KLOC
Eclipse 2.1	27March2003	7909(6702cd+1207id)	988 KLOC
Eclipse 3.0	25 June 2004	10635(8903cd+1732id)	1306 KLOC

We select Eclipse 2.0, 2.1, and 3.0 as the subjects of our study for two reasons.

1. Their fault data are publicly available [23-24]. Therefore, it is easy to externally validate our empirical results by other researchers.
2. They are major releases of Eclipse and have been widely used for several years.

2.2.2 To select the software metrics

The selection of software metrics was a tedious job because there are many available metrics. We used two criteria in our selection process:

- The set of metrics cover all aspects of OO design.
- We have to be able to collect the metrics by using automated tool.

17 metrics are selected which are discussed in Appendix at the end of the References. These metrics are characterized into coupling, cohesion, inheritance, class complexity and class-size metrics. We used JHAWK [28] automated tool to collect these metrics from the Eclipse source code [27]. JHAWK compiled the source code and give output as each module name and their set of OO metrics. So, we generated three databases for the preprocessed java files.

2.2.3 Associate each Java class with the number of pre-release faults

We collected the fault data from three releases of Eclipse (Versions 2.0, 2.1, and, 3.0) provided by the publicly available data set promise2.0a [23] [24] [29].This data set lists the number of pre-release faults (reported in the last six months before release) for each java file in Eclipse2.0, Eclipse2.1 and Eclipse3.0. We associated each Java class with the number of pre-release faults in the corresponding class definition file in Eclipse 2.0, 2.1, and 3.0 in promise2.0a, respectively. Pre release bug data are used for study and two types of categorization has been done on the pre release error data:

1. Binary Categorization: In this we only used two values 0(means no error) and 1(means with error).
2. Multinomial Categorization: In this we classify the error severity into 4 classes.

For classification our followed steps are as follows:

- We find the descriptive statistics of pre error data. From that we are able to know the min, different number of occurrences of error (nonzero) and max value of error data in all classes of every versions of Eclipse.
- After that, we again find the descriptive statistics of (Min, 25%, 50%, 75% and Max) the different occurrences of number of errors (from min (nonzero) to max).

Based on that we classified class error data into one of five categories that are defined as follows:

- No Error: class containing zero error.
- Nominal: class containing error in the range Min<=error<25%
- Low: class containing error in the range 25% <= error <50%

- Medium: class containing error in the range 50% <=error <75%
- High: class containing error in the range 75% <=error <Max

Table 2: Descriptive statistics of error data for all different occurrences of no of errors in Eclipse (2.0, 2.1&3.0)

Version	Min	0.25	0.50	0.75	Max
Eclipse2.0	1.00	9.00	18.00	29.00	69.00
Eclipse2.1	1.00	6.00	12.00	18.00	24.00
Eclipse3.0	1.00	9.00	18.00	26.00	43.00

2.3 Hypothesis

The objective of the study is to develop a Bivariate and Multivariate Binary as well as Multinomial logistic Regression model using software metrics that can be used to classify modules to different error categories in OO system.

The following are the Null hypotheses for our study:

- Hypothesis 1: A class cannot be categorized into error-prone and error-free categories by a model build with two OO-metric (i.e. one is any OO-metric (except size metric) and another is a size metric) in the binary categorization.
  (Null hypothesis: A class can be categorized into error-prone and error-free categories by a model build with two OO-metric (i.e. one is any OO-metric (except size metric) and another is a size metric) in the binary categorization.
- Hypothesis 2: A class cannot be categorized into error-prone and error-free categories by a model build with more than two OO-metrics in the binary categorization.
  (Null hypothesis: A class can be categorized into error-prone and error-free categories by a model build with more than two OO-metrics in the binary categorization.)
- Hypothesis 3: A class cannot be categorized into any one of the error categories (Nominal, Low, Medium, and High) and the No-error categories by a model build with two OO-metric (i.e. one is any OO-metric (except size metric) and another is a size metric) in the multinomial categorization.
  (Null hypothesis: A class can be categorized into any one of the error categories (Nominal, Low, Medium, and High) and the No-error categories by a model build with two OO-metric (i.e. one is any OO-metric (except size metric) and another is a size metric) in the multinomial categorization for the three releases of Eclipse.
- Hypothesis 4: A class cannot be categorized into any one of the error categories (Nominal, Low, Medium, and High) and the No-error categories by a model build with more than two OO-metric in the multinomial categorization.
  (Null hypothesis: A class can be categorized into any one of the error categories (Nominal, Low, Medium, and High) and the No-error categories by a model build with more than two OO-metric in the multinomial categorization.)

More specifically, we attempt to answer the following questions by appropriate statistical analysis technique:

1. How accurate do the investigated metrics distinguish between fault-prone and not fault-prone classes using Bivariate and Multivariate, Binary Logistic Regression model?
2. How accurate do the investigated metrics classifies classes into four categories: Nominal, Low, Medium, and High based on the error severity level using Bivariate and Multivariate, Multinomial Logistic Regression model?

3. Research Methodology

In this section, the steps taken to analyze OO-metric and size metrics for classes taken for analysis are described in following stages: (i) Descriptive statistics and outlier analysis, (ii) PC method, (iii) correlation to size, (iv)Logistic Regression (LR) analysis and model prediction and (v) model evaluation.

3.1. Descriptive Statistics and Outlier Analysis

Descriptive statistics are helpful for development of certain measures to summarize data using mean, median, quartile range and standard deviation. All data points, which are located in an Empty part of the sample space, are called outliers. Outlier analysis is done to find and remove data points that are over-influential. To identify Bivariate and Multivariate outliers, we calculate for each data point the Mahalanobis Jackknife [21-22] distance. Mahalanobis Jackknife is a measure of the distance in multidimensional space of each observation from the mean center of the observations [21-22]. The influence of Bivariate and Multivariate outliers was tested. If the significance of one or more independent variables in the model depends on the presence or absence of the outlier, then that outlier is to be removed.

3.2. Principal-component Method

PC method is a standard technique used to find the interdependence, among a set of variables. The factors summarize the commonality of these variables and factor loadings represent the correlation between the variables and the factor. PC method maximizes the sum of squared loadings of each factor extracted in turn. The PC method aims at constructing new variable (P_i), called PC out of a given set of variables X_j's (j = 1, 2,.. k)

$$P_1 = b_{11}X_1 + b_{12}X_2 + ... + b_{1k}X_k$$

$$P_2 = b_{21}X_1 + b_{22}X_2 + ... + b_{2k}X_k$$

.....

$$P_k = b_{k1}X_1 + b_{k2}X_k + ... + b_{kk}X_k \qquad (1)$$

All b_{ij}'s called loadings are worked out in such a way that the extracted PCs satisfy the following two conditions:

1. PCs are uncorrelated (orthogonal) and
2. The first PC (P1) has the highest variance; the second PC has the next highest variance and so on.

The variables with high loadings help to identify the dimension, the PC is capturing. For this, we consider the rotated components. There are various strategies to perform such rotation. This includes Quartimax, Varimax and Equimax orthogonal rotation[14][21].We used Varimax method ,which maximizes the sum of variances of required loadings of the factor matrix .Eigen value associated with each PC indicates the relative importance of each dimension for the particular set of variables being analyzed. The PC with Eigen value greater than one is taken for interpretation [14].

3.3. Correlation of Metrics with Size and fault

Correlation analysis studies the variation of two or more variables for determining the amount of correlation between them. In order to analyze the relationship of design metrics to the size and fault data of the class we use Spearman's Rho coefficient of correlation. This is to determine empirically whether the coupling, cohesion or inheritance metric is essentially measuring size (number of lines of code (NLOC)) and fault-proneness.

3.4. Logistic Regression and Model Prediction

LR is a very well known technique in statistics. It is used to predict dependent variable from a set of independent variables [13][32]. Binary LR is used to construct models when the dependent variable is binary and Multinomial LR is used to construct models when the dependent variable has more than two categories. In our study, the dependent variable is fault proneness and the independent variables are OO metrics. LR can be used in two ways: (i) Bivariate LR and (ii) Multivariate LR

In Multivariate LR method, metrics are used in combination (more than two metrics), but two metrics are used in Bivariate LR.

3.4.1. Binary logistic regression model

Binary Logistic regression is a standard statistical modeling method in which the dependent variable Y can take on only one of two different values [25]. In the following, let the values be 0 and 1. Here, Y = 1 represents the corresponding class have at least a fault and Y = 0 represents the corresponding class have no fault. Assume that X_1, X_2, . . . , Xn represents the independent variables (i.e. the metrics in this study) and $\Pr(Y = 1|x_1, x_2, \ldots, x_n)$ represents the probability that Y = 1 when $X_1 = x_1$, $X_2 = x_2$, . . ., and $X_n = x_n$. Then, the logistic regression model assumes that $\Pr(Y = 1|x_1, x_2 \ldots x_n)$ is related to $x_1, x_2, \ldots, x_n$ by the following equation:

$$\Pr(Y = 1 \mid x_1, x_2, \ldots, x_n) = \frac{e^{\alpha+\beta_1 x_1+\ldots+\beta_n x_n}}{1+e^{\alpha+\beta_1 x_1+\ldots+\beta_n x_n}} \quad (2)$$

Where β_is are the regression coefficients and can be estimated through the maximization of a log-likelihood.

3.4.2 Multinomial Logistic regression

Multinomial Logistic regression is modified form of binary logistic regression, it is appropriate when the outcome is a polytomous variable (i.e. categorical with more than two categories) and the predictors are of any type. Y can take on more than two different values depending on the number of different categories [25]. In the following, let the values be 0,1,2,3 and 4.Here, Y = 1,2,3 and 4 represents the corresponding class have fault according to the nominal category, low category, mid category and high category and Y = 0 represents the corresponding class have no fault. Categorization detail is in our previous paper [11].Multinomial logistic regression compares multiple groups through a combination of binary logistic regressions.

As in other forms of linear regression, multinomial logistic regression uses a linear predictor function to predict the probability that observation i has outcome k, of the following form:

$$f(k,i) = \beta_{0,k} + \beta_{1,k} \times x_{1,i} + \beta_{2,k} \times x_{1,i} + \ldots + \beta_{M,k} \times x_{M,i} \tag{3}$$

Where, $\beta_{M,k}$ is a regression coefficient associated with the mth explanatory variable and the kth outcome. The regression coefficients and explanatory variables are normally grouped into vectors of size M+1, so that the predictor function can be written more compactly:

$$f(k,i) = \beta_k \times x_i \tag{4}$$

Where, β_k is the set of regression coefficients associated with outcome k, and x_i(a row vector) is the set of explanatory variables associated with observation i[13].

In Multivariate LR model, we used backward elimination method for OO-metric which are selected in PC. It includes all the independent variables in the model in beginning, after that, Variables are deleted one at a time from the model until stopping criteria are fulfilled.

Multicollinearity refers to the degree to which any variable effect can be predicted by the other variables in the analysis. As multicollinearity rises, the ability to define any variable’s effect is diminished. Thus, the interpretation of the model becomes difficult, as the impact of individual variables on the dependent variable can no longer be judged independently from the other variables [21]. Thus, a test of multicollinearity was performed on the fault-proneness model predicted in Section 4.3. Let X_1, X_2, . . ,X_n be the covariates of the model predicted. PC method is applied on these variables to find maximum eigenvalue, emax and minimum eigenvalue, emin. The conditional number is defined as:

$$\lambda = \sqrt{e_{\max}/e_{\min}} \tag{5}$$

If the value of the conditional number is 30, then multicollinearity is not tolerable [31].

The following statistics are reported for each significant metric:

- Odds ratio: It is the probability of the event divided by the probability of the nonevent. The event in our study is having a fault and nonevent is probability of not having a fault. An odds ratio with value 2 means that dependent variable is multiplied by 2 when the independent variable increases by 1 unit.
- Maximum likelihood estimation (MLE) and coefficients (α_i’s): MLE is a statistical method for estimating the coefficients of a model. The likelihood function (L) measures the probability of observing the set of dependent

variable values (P_1, P_2, . . , P_n). It is written as the probability of the product of the independent variables:

$$L = prob(P_1 \times P_2 \times ... \times P_n) \qquad (6)$$

MLE involves finding the coefficients that make the log of the likelihood function Log (L) as large as possible. The bigger the value of coefficients, the larger the impact of the OO-metrics on the probability of fault detection.

- The statistical significance (sig): It is the significance level of the coefficient; larger the statistical significance, lesser the estimated impact of the independent variables (OO metrics). In our study, we used 0.05 as the significance threshold.
- The R2 statistic: It is the proportion of the variance in the dependent variable that is explained by the variance of the independent variables. The higher the value of R2, the higher the effect of the model's independent variables and more is the accuracy of the model. R2 statistic is defined by the following ratio:

$$R^2 = 1 - \frac{LL(a,B)}{LL(a)} \qquad (7)$$

Where, LL (a, B) is the maximum log-likelihood function with significant independent variables included in the model. LL (a) is the log-likelihood function evaluated with only the constant included.

3.5 Model Evaluations Criteria

The most commonly used measures for predictive effectiveness of classification models are accuracy, sensitivity, specificity, precision, and F-measure [30][32]. In the context of binary fault-proneness and multinomial fault-proneness classification, these five effectiveness measures can be defined as follows:

Accuracy: the number of classes that are correctly classified (in different categories depending on the categorization), divided by the total number of classes. Where TP and TN are the number of correct predictions that an instance is positive and negative respectively. FP and FN are the number of incorrect predictions that an instance is positive and negative respectively.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \qquad (8)$$

- Sensitivity: the number of classes correctly classified as fault-prone (in different categories depending on the categorization), divided by the total number of classes that have faults;
- Specificity: the number of classes correctly classified as not fault-prone (in different categories depending on the categorization), divided by the total number of classes that have no fault.
- The general rule to evaluate the classification performance is to find the area under the curve (AUC)[26]:
  - ➢ AUC=0.5 means no good classification;
  - ➢ 0.5<AUC<0.6 means poor classification;

- $0.6 \leq AUC < 0.7$ means fair classification;
- $0.7 \leq AUC < 0.8$ means acceptable classification;
- $0.8 \leq AUC < 0.9$ means excellent classification;
- $AUC \geq 0.9$ means outstanding classification.

4. Experimental analysis

In this section, we present in detail the experimental results. In section4.1, we report the distribution and section 4.2 correlation analysis results of the investigated metrics on the fault data sets. In section4.3, we do the Principal component analysis. In section 4.4, we show the results of logistic regression in two contexts Binary and Multinomial.

4.1 Descriptive analysis

Table 3, Table 4 and Table 5 presents the common descriptive statistics of the investigated metrics on three data sets Eclipse2.0, Eclipse2.1 and Eclipse3.0 respectively. Columns "Mean","Median","Std.Deviation","Minimum","Maximum","25th Percentile" and "75th Percentile" state for each metric the mean value, median value, standard deviation value, minimum value, maximum value and interquartile ranges. When looking at the lower 25th percentile, the median, the 75th percentile, the mean, and the standard deviation, we find that the distributions of each metric over the three Eclipse versions are similar. This means that a prediction model built on the former version could probably be applied to the latter versions. All metrics showing marginal variation between classes.

4.2 Correlation with Fault and size

There is evidence in Table 7 that object-oriented metrics are associated with size. Spearman rho correlation coefficients go as high as 0.890 for associations between some complexity metrics and 0.849 for association between coupling metrics with size (i.e. NLOC no. of lines of code), and 0.615 for cohesion metrics, and all are statistically significant. Table 6 also shows the results of the Spearman correlation analysis for the three versions of Eclipse with their respective pre-release faults. The correlation values are statistically significant because the p-value of all the metrics are (<.001).Assuming a reasonably sized data set, correlation value less than 0.1 trivial,0.1-0.3 minor,0.3-0.5 moderate,0.5-0.7 large,0.7-0.9 very large, and 0.9-1.0 almost perfect[32].As can be seen, all metrics have minor or moderate correlation with pre-release faults. Of all metrics, RFC, EXT, MPC and PACK have the highest correlation with pre-release faults. Also we can see that INST, CBO and AVCC have lowest (minor, almost trivial) correlation with pre-release faults.

4.3 Principal component analysis

In this section, we present the results from the principal component analysis. All measures with sufficient variance (six or more non-zero data points) were subjected to an orthogonal rotation. A total of 17 measures were used. We identified 4 orthogonal dimensions spanned by the 17 measures, indicating that, as expected, there is a large amount of redundancy present among these measures. Conditional number is less than 30 i.e. $\sqrt{5.081/1.854}$, that's why redundancy is tolerable. The 4 PCs capture 84.923 of the variance in the data set. For each PC, its eigenvalue, the variance and the cumulative variance are arranged in the Table 6. Based on the analysis of the coefficients associated with the measure within each of the rotated components, the PCs are interpreted as follows:

- PC1: UWCS, CC, NOM, LCOM2 and INST, this PC contain different dimensions like complexity, size and cohesion metric.
- PC2: RFC, EXT, MPC and PACK all are coupling metrics.
- PC3: AVCC and MAXCC are average and maximum complexity metrics.
- PC4: CBO and FOUT are coupling metrics respectively.

Hence, we see metrics capturing different properties (i.e. Cohesion, Complexity and size) are included in the same dimension P1. Coupling metrics were included in dimensions P2 and P4.

4.4 Logistic Regression Models

4.4.1 Bivariate Logistic Regression Models

[17] had shown that class size can have a strong confounding effect on the associations between OO-metrics and fault-proneness and therefore recommended that future validation studies should always control for class size. Table 8, Table 9 and Table 10 summarizes the results of the Binary Bivariate logistic regression analysis with regard to pre-release fault-proneness prediction from Eclipse 2.0, Eclipse 2.1 and Eclips3.0 respectively. Table 11, Table 12 and Table 13 summarizes the results of the Multinomial Bivariate logistic regression analysis with regard to pre-release fault-proneness prediction from Eclipse 2.0, 2.1 and 3.0 respectively. The column "Metric", shows the independent variable used. The columns "Const", "Coef.", "S.E." and "p-value" states for each model the constant, the estimated regression coefficient, the standard error of the coefficient, and the statistical significance of the coefficient. The columns "R^2" and "OR" report the R^2 and the odds ratio. OR>1 suggest that metrics are associated with higher odds of outcome (i.e. fault proneness).R^2 is known as the coefficient of determination. R^2 is always between 0 and 100%:0% indicates that the model explains none of the variability of the response data around its mean, 100% indicates that the model explains all the variability of the response data around its mean. The column"% of cases correctly classified" reports the total number of classes which are correctly classified depending on the predicted values for the dependent variable, based upon the regression model, with the actual observed values in the data. We can see that all the metrics have positive regression coefficients and their p-values are very significant. From Table , we can see that: (1) after controlling for class size, some metrics do not have a significant association with fault-proneness in Eclipse 2.0,2.1 and 3.0 (shown in bold); (2) after controlling for class size, the odds ratio for each metric considerably decreases ; and (3) for each metric, the model built with it and NLOC has a slightly higher R^2 than the univariate model built with NLOC and also built with OO-metric .

4.4.2 Multivariate Logistic Regression Models

In this section, we predict models to identify faulty classes. The model is built using backward elimination procedure. All metrics are allowed to enter the model. The variables included in the model and model statistics are shown in Table 14 and Table 15 for all 3 versions of Eclipse in binary and multinomial aspects respectively.

4.5 Hypothesis Testing

In this section, we investigate the applicability of four types of model for predicting fault-proneness in two aspects i.e. binary (BLR and MLR) and multinomial (BLR and MLR).The testing of applicability of hypothesis in all four proposed models are shown in Table 16.

5. Validation Results

Table 17, Table 18 and Table 19 summarizes the TP, TN, FP, FN, Sensitivity, (1-Specificity), AUC and Accuracy results from 17-fold cross-validation for the Bivariate models for Binary categorization of Eclipse 2.0,2.1 and 3.0. The columns from “NOS” to “AVCC” each corresponding to a bivariate model built with a single metric and Number of lines of code(NLOC) respectively report the AUC scores are computed by the SPSS. Table 20, Table 21 and Table 22 summarizes the TP, TN, FP, FN, Sensitivity, (1-Specificity), AUC and Accuracy results from 17-fold cross-validation for the Bivariate models for Multinomial categorization of Eclipse 2.0,2.1 and 3.0 respectively.

Table 23 and Table 24 summarize the TP, TN, FP, FN, Sensitivity, (1-Specificity), AUC and Accuracy results from 17-fold cross-validation for the Multivariate models for Binary and Multinomial aspect respectively for all 3 version of Eclipse.

Following are the result of the statistical analysis:

(1) All metrics showing accuracy in the range 64% to 67% in Eclipse2.0, but in Eclipse2.1 and Eclipse3.0 it is in the range 74% to76% and AUC result comes under the range $0.5<AUC<0.6$ means poor classification in all three version of Eclipse for BBLR.

(2) Accuracy result for nominal category is 58% to 65% and AUC comes in poor category, but accuracy for low, mid and high category is in 50% to 60% and their classification power comes in fair and acceptable classification. PACK is showing best result among all metrics for Eclipse 2.0 in MBLR model.

(3) Accuracy of nominal category is high compare to other but its discriminating power is in poor class. Other 3 categories have accuracy 69% to 71% and low and part of mid class come in acceptable class, but high comes in excellent classification for eclipse 2.1 in MBLR model. PACK is showing the best result among all. Same incidence is found in Eclipse3.0 for MBLR model.

(4) In BMLR model the accuracy model is between 68% and 76% for all 3 version of Eclipse. And AUC comes in poor classification.

(5) In MMLR model the accuracy of nominal category is 62% to 66%, its AUC come in the poor class. But all other 3 categories has accuracy 55% and AUC for low (in Eclipse2.1 and 3.0) and high(in eclipse2.0)comes under acceptable class and except high for eclipse 2.1 comes in excellent class. The AUC result for Eclipse2.1 in high category gives the outstanding classification.

When Software debugger use MMLR model for own Software checking then they got the idea that their classes come in which category. Depending on that they do the debugging of class on that number of error which come in that category range. So, multinomial LR give the idea that applying the same testing effort to all modules of a system is not the optimal approach, because the distribution of bugs among individual parts of a software system is not uniform. So, with such knowledge we would be able to prioritize the tests and therefore, work more efficiently. We conclude that prioritizing the test on the basis different error categories is more effective than the category of fault prone and fault free classes in binary categorization for developing fault prediction models.

6. Conclusion and Future Work

In this paper, we use logistic regression (Binary and Multinomial) to investigate the relationships between selected seventeen traditional metrics and fault-proneness at the class level for three versions of Eclipse. Our results are summarized as follows:

- Bivariate model (in both aspects, Binary and Multinomial) are unable to discriminate between fault-prone and not fault-prone classes, more efficiently. The bivariate model are built with one chosen metric and another one is size metric (i.e. NLOC).All metrics comes in class of poor classification of predictive ability.
- We have developed bivariate model for finding that class size have a strong confounding effect on the associations between OO metrics and fault-proneness. So, we control class size using any size metric and developed model with each metric which are showing better result .But we find that model are not showing best predictive ability. So, the best ideas for developing the model is that develop model using more than two metrics among that one of the metric is size metric.
- When Multivariate logistic regression models built with combined effect of all chosen metrics. This shows better result in Prediction. But, among both aspect (i.e. Binary, Multinomial) multivariate model shows better result in multinomial aspect of fault prediction.
- Of the investigated metrics, PACK is the only metric which is used to develop model in both aspects as well as both way bivariate and multivariate. It shows that Pack metric has the best discrimination ability.
- We conclude that prioritizing the test on the basis of different number of errors in different error categories is more effective than the category of fault prone and fault free classes in binary categorization for developing fault prediction models.

In the future work, we will replicate this study using these investigated metrics and other modeling techniques to draw stronger conclusion for getting best predictor and model also. In this study we have used all those metric which are capable for giving the significant threshold for differentiating the classes in binary categorization (error-free and error-prone) and also multinomial categorization (nominal, low, mid and high) from our previous study [11]. In particular, we will employ data mining technique for developing multivariate logistic regression (i.e. those metrics which show better predictive ability in bivariate LR model, using all those metric we develop the model) in both context (binary and multinomial).Consequently, more general conclusions would be obtained.

References:

[1] G. Myers, T. Badgett, T.Thomas, and C. Sandler, 2004, "The Art of Software Testing", John Wiley & Sons.

[2] Zhou, Y., Leung, H., 2006," Empirical analysis of object-oriented design metrics for predicting high and low severity faults", IEEE Transactions on Software Engineering, 32 (10), pp 771–789.

[3] K. K. Aggarwal, Y. S., Arvinder Kaur and Ruchika Malhotra, 2009," Empirical Analysis for Investigating the Effect of Object-Oriented Metrics on Fault Proneness: Replicated Case Study", Software process improvement and practice, pp 39-62.

[4] T. J. McCabe, 1976, "A complexity measure", IEEE Trans. Software Eng., Volume 2 No. 4, pp. 308–320.

[5] Mittal, P. Singh, S. Kahlon, K.S., 2013," Empirical Model for Fault Prediction using Object-oriented metrics in Mozilla Firefox", International Journal of Computer Technology & Research, 1 (6), pp 151-161.

[6] N. Kayarvizhy, S. K., 2013,"High Precision Cohesion Metric", WSEAS TRANSACTIONS on Information Sciences and Application, Volume 10, pp 79-89.

[7] Yuming, Zhou, Xu Baowen, Leung Hareton, 2010," On the ability of complexity metrics to predict fault-prone classes", The Journal of Systems and Software, pp 660-674.

[8] Olague, H.M., Etzkorn, L.H., Messimer, S.L., Delugach, H.S., 2008, "An empirical validation of object-oriented class complexity metrics and their ability to predict error-prone classes in highly iterative, or agile, software: a case study", Journal of Software Maintenance and Evolution: Research and Practice 20 (3), pp 171–197.

[9] Chidamber S, Kemerer C., 1994, "A metrics suite for object-oriented design", IEEE Transactions on Software Engineering,20(6), pp 476–493.

[10] Raed Shatnawi, Wei Li, James Swain and Tim Newman, 2010, "Finding software metrics threshold values using ROC curves", Journal of Software Maintenance and Evolution: Research and Practice, Volume 22, No. 1, pp.1–16.

[11] Dipti Kumari and Kumar Rajnish, 2013, “Finding error-prone classes at design time using class based Object-Oriented metrics threshold through statistical method”, Infocomp Journal of Computer Science”, Vol. 12 No.1, pp. 49-63.
[12] Olague H, Etzkorn L, Gholston S, Quattlebaum S., 2007,” Empirical validation of three software metrics suites to predict fault-proneness of object-oriented classes developed using highly iterative or agile software development processes”, IEEE Transactions on software Engineering 33(8),pp. 402–419.
[13] Hosmer D, Lemeshow S., 1989, Applied Logistic Regression, John Wiley and Sons.
[14] Kothari CR., 2004,Research Methodology Methods and Techniques, New Age International Limited.
[15] Gyimothy T, Ferenc R, Siket I , 2005,” Empirical validation of object-oriented metrics on open source software for fault prediction.“, IEEE Transactions on Software Engineering 31(10), pp. 897-910.
[16] Lorenz M, Kidd J., 1994, Object-oriented Software Metrics, Prentice-Hall.
[17] El Emam K, Melo W, Machado J., 1999,” The prediction of faulty classes using object-oriented design metrics”, Journal of Systems and Software 56, pp. 63–75.
[18] Yuming Z, Hareton L., 2006, “Empirical analysis of Object- Oriented Design Metrics for predicting high servility faults”, IEEE Transactions on Software Engineering ,32(10), pp. 771–784.
[19] Briand L, Daly J, Wust J., 1998,”Unified framework for cohesion measurement in object-oriented systems”, Empirical Software Engineering, 3, pp. 65–117.
[20] Briand L, Daly J, Wust J. , 1999,”A unified framework for coupling measurement in object-oriented systems”, IEEE Transactions on software Engineering 25, pp. 91–121.
[21] Barnett V, Price T., 1995, Outliers in Statistical Data, John Wiley and Sons; UK.
[22] Hair J, Anderson R, Tatham R, Black W., 2006, Multivariate Data Analysis, Pearson Education; India.
[23] Zimmermann, T., Premrai, R., Zeller, A, 2007,” Predicting defects for Eclipse.”, In: Proceedings of the Third International Workshop on Predictor models in Software Engineering.
[24] Schroter, A., Zimmermann, T., Premraj, R., Zeller, A., 2006, “If your bug database could talk” , In: Proceedings of the Fifth International Symposium on Empirical Software Engineering, vol. 2, pp. 18–20.
[25] Georage, D., Mallery, P., 2011, SPSS for Windows STEP BY STEP, Pearson Education.
[26] Hanley, J.A., McNeil, B.J., 1982,”The meaning and use of the area under a receiver operating characteristic (ROC) curve”, Radiology, 143 (1), pp. 29–36.
[27] Eclipse source code (for archived releases): http://archive.eclipse.org/eclipse/downloads/ (Accessed 3 December 2013)
[28] JHAWK metrics reference http://www.virtualmachinery.com/jhawkreferences.html (Accessed March 2014)

[29] Eclipse bug data (for archived releases): http://www.st.cs.uni-sb.de/softevo/bug-data/eclipse (Accessed 20 November 2013)

[30] Jiang, Y., Cukic, B., Ma, Y., 2008," Techniques for evaluating fault prediction models", Empirical Software Engineering ,13 (5), pp. 561–595.

[31] Belsley D, Kuh E, Welsch R. 1980. Regression Diagnostics: Identifying Influential Data and Sources of Collinearity, John Wiley and Sons; USA.

[32] Reinard, J.C., 2006, Communication Research Statistics, Sage Publications, Inc., Thousand Oaks, California.

APPENDIX A: OBJECT ORIENTED- METRICS DEFINITIONS

Metric	Metric description	Ref.
NOS	Number of statements in the class	[9]
UWCS	Unweighted Class Size (UWCS) of the class	[9]
CC	Class Complexity	[28]
RFC	Response For Class (RFC) metric for the class	[28]
NLOC	Number of lines of code in the class and its methods	[28]
EXT	External methods called by the class and by methods in the class	[28]
MPC	Message passing (MPC) metric for the class	[20]
LMC	Local methods called by the class and by methods in the class	[9]
TCC	Total McCabe's cyclomatic Complexity for the class	[4]
PACK	Packages imported by the class	[9]
NOM	Number of methods in the class	[4]
LCOM2	Lack of Cohesion of Methods (2) (LCOM2) metric for the class	[19]
INST	Instance variables defined by the class	[9]
CBO	Coupling Between Objects (CBO) metric for the class	[20]
MAXCC	Maximum McCabe's cyclomatic Complexity for all of the methods in the class	[4]
FOUT	Fan Out (Efferent coupling (Ce)) metric for the class	[9]
AVCC	Average McCabe's cyclomatic Complexity for all of the methods in the class	[4]

THE APPENDIX B: TABLES RELATED TO VALIDATION AND EVALUATION

Table3:Descriptive statistics of selected metrics of Eclipse2.0

Version	Metric	Mean	Median	Std. Dev.	Min	Max	Percentile	
							25.00	75.00
	NOS	74.18	29.00	150.32	0.00	3582.00	9.00	77.00
	UWCS	15.48	8.00	35.46	0.00	1646.00	4.00	17.00
	RFC	27.47	14.00	39.57	0.00	596.00	5.00	34.00
	NLOC	98.20	40.00	201.20	0.00	5200.00	13.00	101.00
	EXT	17.21	8.00	26.26	0.00	325.00	0.00	22.00
	MPC	17.21	8.00	26.26	0.00	325.00	0.00	22.00
	LMC	2.28	0.00	5.76	0.00	194.00	0.00	2.00
Eclipse2.0	TCC	23.67	10.00	47.99	0.00	1222.00	3.00	24.00
	CC	28.89	12.00	58.81	0.00	1839.00	5.00	31.00
	PACK	7.30	4.00	10.24	0.00	146.00	1.00	9.00
	NOM	10.26	6.00	19.82	0.00	596.00	3.00	12.00
	LCOM2	82.59	5.00	752.85	0.00	41126.00	1.00	22.00
	INST	5.22	2.00	20.12	0.00	1050.00	0.00	5.00
	CBO	3.66	2.00	4.81	0.00	76.00	1.00	4.00
	MAXCC	4.82	3.00	7.76	0.00	229.00	1.00	6.00
	FOUT	1.85	1.00	3.14	0.00	69.00	0.00	2.00
	AVCC	1.89	1.50	1.48	0.00	26.17	1.00	2.40

Table 4: Descriptive statistics of selected metrics of Eclipse2.1

Version	Metric	Mean	Median	Std. Dev.	Min.	Max.	Percentile	
							25	75
	NOS	77.39	30.00	155.97	0.00	3592.00	10.00	79.00
	UWCS	15.85	9.00	36.31	0.00	1740.00	4.00	17.00
	RFC	28.94	15.00	42.02	0.00	613.00	5.00	35.00
	NLOC	102.52	41.00	209.36	0.00	5221.00	13.00	106.00
	EXT	18.46	8.00	28.23	0.00	335.00	0.00	23.00
	MPC	18.46	8.00	28.23	0.00	335.00	0.00	23.00
	LMC	2.45	0.00	6.12	0.00	195.00	0.00	3.00
Eclipse2.1	TCC	24.61	10.00	49.90	0.00	1226.00	3.00	25.00
	CC	29.98	13.00	60.90	0.00	1932.00	5.00	32.00
	PACK	7.59	4.00	10.90	0.00	151.00	1.00	9.00
	NOM	10.48	6.00	20.50	0.00	613.00	3.00	12.00
	LCOM2	89.48	5.00	766.44	0.00	43119.00	1.00	24.00
	INST	5.37	2.00	20.44	0.00	1127.00	0.00	5.00
	CBO	3.64	2.00	4.87	0.00	76.00	1.00	4.00
	MAXCC	4.98	3.00	8.13	0.00	229.00	1.00	6.00
	FOUT	1.82	1.00	3.07	0.00	69.00	0.00	2.00
	AVCC	1.92	1.50	1.54	0.00	30.50	1.00	2.44

Table 5:Descriptive statistics of selected metrics of Eclipse3.0

Version	Metric	Mean	Median	Std. Deviation	Minimum	Maximum	Percentile	
							25.00	75.00
	NOS	74.18	29.00	150.32	0.00	3582.00	9.00	77.00
	UWCS	15.48	8.00	35.46	0.00	1646.00	4.00	17.00
	RFC	27.47	14.00	39.57	0.00	596.00	5.00	34.00
	NLOC	98.20	40.00	201.20	0.00	5200.00	13.00	101.00
	EXT	17.21	8.00	26.26	0.00	325.00	0.00	22.00
	MPC	17.21	8.00	26.26	0.00	325.00	0.00	22.00
	LMC	2.28	0.00	5.76	0.00	194.00	0.00	2.00
Eclipse3.0	TCC	23.67	10.00	47.99	0.00	1222.00	3.00	24.00
	CC	28.89	12.00	58.81	0.00	1839.00	5.00	31.00
	PACK	7.30	4.00	10.24	0.00	146.00	1.00	9.00
	NOM	10.26	6.00	19.82	0.00	596.00	3.00	12.00
	LCOM2	82.59	5.00	752.85	0.00	41126.00	1.00	22.00
	INST	5.22	2.00	20.12	0.00	1050.00	0.00	5.00
	CBO	3.66	2.00	4.81	0.00	76.00	1.00	4.00
	MAXCC	4.82	3.00	7.76	0.00	229.00	1.00	6.00
	FOUT	1.85	1.00	3.14	0.00	69.00	0.00	2.00
	AVCC	1.89	1.50	1.48	0.00	26.17	1.00	2.40

Table 6: Rotated principal component

	Eclipse2.0				Eclipse2.1				Eclipse3.0			
PC	P1	P2	P3	P4	P1	P2	P3	P4	P1	P2	P3	P4
Eigenvalues	5.081	3.864	2.608	1.906	4.96	4.156	2.533	1.854	4.864	4.362	2.484	1.878
Variance %	31.757	24.152	16.297	11.91	31.003	25.974	15.829	11.59	30.398	27.26	15.527	11.738
Cumulative %	31.757	55.909	72.206	84.116	31.003	56.977	72.806	84.396	30.398	57.6582.	73.185	84.923
NOS	0.612	0.463	0.522	0.153	0.608	0.495	0.508	0.138	0.591	0.537	0.486	0.118
UWCS	**0.931**	0.12	0.04	0.199	**0.927**	0.133	0.045	0.214	**0.926**	0.15	0.062	0.21
CC	**0.822**	0.31	0.41	0.183	**0.814**	0.353	0.41	0.177	**0.798**	0.373	0.409	0.162
RFC	0.542	**0.723**	0.269	0.261	0.523	**0.75**	0.245	0.253	0.505	**0.772**	0.23	0.233
EXT	0.19	**0.877**	0.314	0.209	0.181	**0.891**	0.286	0.202	0.173	**0.901**	0.272	0.181
MPC	0.19	**0.877**	0.314	0.209	0.181	**0.891**	0.286	0.202	0.173	**0.901**	0.272	0.181
LMC	0.547	0.604	0.212	0.078	0.507	0.67	0.19	0.056	0.478	0.699	0.187	0.071
TCC	0.66	0.407	0.524	0.177	0.656	0.439	0.512	0.159	0.638	0.48	0.489	0.141
PACK	-0.014	**0.852**	0.025	-0.016	-0.005	**0.856**	0.021	0.014	-0.01	**0.837**	0.016	0.017
NOM	**0.828**	0.281	0.122	0.243	**0.821**	0.309	0.108	0.24	**0.813**	0.337	0.096	0.231
LCOM2	**0.754**	0.121	0.035	0.065	**0.74**	0.168	0.03	0.063	**0.751**	0.169	-0.007	0.087
INST	**0.825**	-0.065	-0.05	0.111	**0.823**	-0.073	-0.028	0.14	**0.838**	-0.072	0.015	0.141
CBO	0.189	0.096	0.043	**0.895**	0.19	0.11	0.046	**0.889**	0.194	0.117	0.043	**0.903**
MAXCC	0.179	0.205	**0.89**	0.052	0.191	0.2	**0.891**	0.051	0.181	0.215	**0.895**	0.047
FOUT	0.238	0.193	0.099	**0.857**	0.263	0.208	0.101	**0.841**	0.266	0.19	0.087	**0.865**
AVCC	-0.036	0.191	**0.868**	0.049	-0.037	0.176	**0.873**	0.059	-0.024	0.171	**0.885**	0.053

Table7: Spearman Correlation metrics with size and pre-release faults

VERSION→	ECLIPSE2.0		ECLIPSE2.1		ECLIPSE2.1	
METRICS↓	NLOC	PRE	NLOC	PRE	NLOC	PRE
NOS	0.993	0.323	0.993	0.364	0.994	0.363
UWCS	0.855	0.245	0.857	0.285	0.862	0.32
CC	0.953	0.291	0.954	0.335	0.958	0.351
RFC	0.929	0.354	0.93	0.373	0.932	0.366
EXT	0.884	0.358	0.887	0.383	0.89	0.36
MPC	0.884	0.358	0.887	0.383	0.89	0.36
LMC	0.792	0.304	0.799	0.324	0.799	0.321
TCC	0.932	0.31	0.934	0.335	0.936	0.345
PACK	0.638	0.346	0.652	0.377	0.663	0.345
NOM	0.836	0.277	0.839	0.287	0.838	0.314
LCOM2	0.598	0.249	0.607	0.218	0.615	0.266
INST	0.629	0.136	0.636	0.199	0.658	0.246
CBO	0.33	0.179	0.845	0.144	0.343	0.157
MAXCC	0.841	0.292	0.341	0.327	0.849	0.315
FOUT	0.424	0.238	0.43	0.196	0.418	0.2
AVCC	0.76	0.267	0.761	0.296	0.764	0.27

Table8: Binary Bivariate logistic regression of Eclipse2.0

Version →	ECLIPSE2.0						
Metrics↓	Const	Coeff.(p-value)	ΔΨ	Size coeff(p-value)	Size ΔΨ	R^2	% of cases correctly classified
NOS	**-0.78**	**.007(.000)**	**1.007**	**-.002(.124)**	**0.998**	**0.073**	**64.4**
UWCS	-0.795	.003(.000)	1.003	.003(.000)	1.003	0.07	64.2
CC	-0.786	.002(.000)	1.002	.003(.000)	1.002	0.069	64.2
RFC	-0.978	.022(.000)	1.022	-.001(.007)	0.999	0.11	66
EXT	**-0.926**	**.025(.000)**	**1.026**	**.000(.349)**	**1**	**0.112**	**66.1**
MPC	**-0.926**	**.025(.000)**	**1.026**	**.000(.349)**	**1**	**0.112**	**66.1**
LMC	-0.786	.011(.000)	1.084	.002(.000)	1.002	0.081	65.3
TCC	**-0.784**	**.004(.147)**	**1.004**	**.002(.000)**	**1.002**	**0.069**	**64.3**
PACK	-1.122	.067(.000)	1.07	.002(.000)	1.002	0.147	67
NOM	-0.812	.010(.005)	1.01	.003(.000)	1.003	0.071	64.2
LCOM2	**-0.773**	**.000(.064)**	**1**	**.003(.000)**	**1.003**	**0.07**	**64.3**
INST	**-0.782**	**.000(.967)**	**1**	**.002(.000)**	**1**	**0.069**	**64.3**
CBO	-0.878	.035(.000)	1.036	.003(.000)	1.003	0.075	64.6
MAXCC	-0.848	.031(.000)	1.031	.002(.000)	1.002	0.073	64.5
FOUT	-0.907	.110(.000)	1.116	.003(.000)	1.002	0.084	65
AVCC	-1.015	.159(.000)	1.172	.003(.000)	1.003	0.079	64.5

Table9: Binary Bivariate logistic regression for Eclipse2.1

Version →	ECLIPSE2.1						
Metrics↓	Const	Coeff.(p-value)	ΔΨ	Size coeff(p-value)	Size ΔΨ	R^2	% of cases correctly classified
NOS	-1.465	.006(.000)	1.006	.000(.729)	1	0.126	75.1
UWCS	-1.467	-.001(.397)	0.999	.005(.000)	1.005	0.124	75.2
CC	-1.471	.000(.904)	1	.004(.000)	1.004	0.124	75.1
RFC	-1.678	.022(.000)	1.022	.000(.959)	1	0.165	76
EXT	-1.65	.028(.000)	1.028	.001(.001)	1.001	0.176	76
MPC	-1.65	.028(.000)	1.028	.001(.001)	1.001	0.176	76
LMC	-1.467	.062(.000)	1.064	.003(.000)	1.003	0.13	75.5
TCC	-1.47	.002(.464)	1.002	.004(.000)	1.004	0.124	75.1
PACK	-1.878	.073(.000)	1.076	.002(.000)	1.002	0.217	76.6
NOM	-1.468	-.001(.598)	0.999	.005(.000)	1.005	0.124	75.1
LCOM2	-1.48	.000(.004)	1	.005(.000)	1.005	0.124	75.1
INST	-1.469	-.001(.656)	0.999	.004(.000)	1.004	0.124	75.2
CBO	-1.484	.005(.383)	1.005	.004(.000)	1.004	0.124	75.1
MAXCC	-1.566	.041(.000)	1.041	.003(.000)	1.003	0.131	75
FOUT	-1.485	.015(.159)	1.015	.004(.000)	1.004	0.124	75.1
AVCC	-1.776	.196(.000)	1.217	.003(.000)	1.003	0.139	74.8

Table10: Binary Bivariate logistic regression for Eclipse3.0

Version →	ECLIPSE3.0						
Metrics↓	Const.	Coeff.(p-value)	ΔΨ	Size coeff.(p-value)	Size ΔΨ	R^2	% of cases correctly classified
NOS	-1.437	.001(.570)	1.001	.004(.003)	1.004	0.124	74.9
UWCS	-1.471	.007(.000)	1.007	.004(.000)	1.004	0.125	74.8
CC	-1.451	.006(.005)	1.006	.003(.000)	1.003	0.125	74.9
RFC	-1.584	.017(.000)	1.017	.001(.007)	1.001	0.148	75
EXT	-1.535	.019(.000)	1.019	.002(.000)	1.002	0.147	75
MPC	-1.535	.019(.000)	1.019	.002(.000)	1.002	0.147	75
LMC	-1.433	.033(.000)	1.034	.004(.000)	1.004	0.125	74.9
TCC	-1.439	.003(.308)	1.003	.004(.000)	1.004	0.124	74.9
PACK	-1.724	.053(.000)	1.055	.003(.000)	1.003	0.18	75.5
NOM	-1.467	.009(.003)	1.01	.004(.000)	1.004	0.125	74.9
LCOM2	-1.45	.000(.000)	1	.005(.000)	1.005	0.125	74.8
INST	-1.446	.004(.035)	1.004	.004(.000)	1.004	0.124	74.8
CBO	-1.501	.023(.000)	1.024	.004(.000)	1.004	0.126	74.9
MAXCC	-1.512	.033(.000)	1.034	.004(.000)	1.004	0.128	74.9
FOUT	-1.511	.068(.000)	1.071	.004(.000)	1.004	0.13	74.8
AVCC	-1.642	.133(.000)	1.143	.004(.000)	1.004	0.131	74.7

Table11: Multinomial Bivariate logistic regression of Eclipse2.0

Metric	Const				Coeff.(p-value)				ΔΨ				size coeff.(p-value)				sizeΔΨ				R^2	% of cases correct
	Nom	Low	Mid	High	Nomi	Low	Mid	High	Nomi	Low	Mid	High	Nomi	Low	Mid	High	Nomin	Low	Mid	High		classifie
NOS	-0.826	-4.090	-5.793	-6.540	0.006 (.000)	0.011 (.000)	0.012 (.000)	0.012 (.000)	1.006	1.011	1.012	1.012	-.002(.158)	-.004(.043)	.004(.088)	-.004(.174)	0.998	0.996	0.996	0.996	0.078	63.300
UWCS	-0.841	-4.104	-5.795	-6.524	0.002 (.194)	0.005 (.018)	0.006 (.016)	0.005 (.096)	0.002	0.000	0.002	0.002	0.003 (.000)	.004(.000)	.004(.000)	.004(.000)	1.021	1.027	1.027	1.027	0.075	63.100
CC	-0.832	-4.088	-5.779	-6.499	0.001 (.477)	0.004 (.073)	0.005 (.051)	0.003 (.426)	1.001	1.004	1.005	1.003	0.003 (.000)	0.003 (.000)	0.003 (.000)	0.004 (.000)	1.003	1.003	1.003	1.004	0.075	63.200
RFC	-0.999	-4.736	-6.685	-7.453	0.020 (.000)	0.038 (.000)	0.043 (.000)	0.041 (.000)	1.021	1.038	1.044)	1.042	-0.001	-0.002	-.003	-.002	0.999	0.998	0.997	0.998	0.129	64.100
EXT	-0.948	-4.678	-6.661	-7.510	0.024 (.000)	0.044 (.000)	0.051 (.000)	0.052 (.000)	1.021	1.039	1.042	1.040	0.000 (.322)	0.000 (.965)	0.000 (.647)	0.000 (.941)	1.025	1.045	1.050	1.052	0.130	64.400
MPC	-0.948	-4.678	-6.661	-7.510	0.024 (.000)	0.044 (.000)	0.051 (.000)	0.052 (.000)	1.021	1.039	1.042	1.040	0.000 (.322)	0.000 (.965)	0.000 (.647)	0.000 (.941)	1.025	1.045	1.050	1.052	0.130	64.400
LMC	-0.827	-4.180	-5.861	-6.670	0.075 (.000)	0.143 (.000)	0.140 (.000)	0.189 (.000)	0.008	0.010	0.013	0.013	0.002 (.000)	0.001 (.001)	0.002 (.004)	0.000 (.910)	1.002	1.001	1.002	0.998	0.092	63.600
TCC	-0.830	-4.086	-5.804	-6.507	0.003 (.223)	0.008 (.025)	0.013 (.003)	0.008 (.257)	1.003	1.008	1.013	1.008	0.002 (.000)	0.003 (.000)	0.002 (.000)	0.003 (.000)	1.002	1.003	1.002	1.003	0.075	63.000
PACK	-1.130	-5.262	-7.349	-8.178	0.064 (.000)	0.121 (.000)	0.133 (.000)	0.135 (.000)	1.066	1.129	1.143	1.144	0.002 (.000)	0.002 (.000)	0.003 (.000)	0.003 (.000)	1.002	1.002	1.003	1.003	0.177	65.100
NOM	-0.857	-4.134	-5.845	-6.574	0.010 (.009)	0.016 (.000)	0.018 (.000)	0.017 (.002)	1.010	1.016	1.018	1.017	0.003 (.000)	0.003 (.000)	0.003 (.000)	0.004 (.000)	1.003	1.003	1.003	1.004	0.077	63.100
LCOM2	-0.822	-4.041	-5.726	-6.465	0.000 (.079)	0.001 (.012)	0.000 (.042)	0.001 (.012)	1.001	1.001	1.000	1.001	0.003 (.000)	0.004 (.000)	0.004 (.000)	0.004 (.000)	1.003	1.004	1.004	1.004	0.076	63.100
INST	-0.829	-4.074	-5.755	-6.481	0.000 (.861)	0.002 (.406)	0.003 (.411)	0.000 (.987)	1.000	1.002	1.003	1.000	0.003 (.000)	0.004 (.000)	0.005 (.000)	0.005 (.000)	1.003	1.004	1.005	1.005	0.074	63.200
CBO	-0.920	-4.178	-6.072	-6.943	0.034 (.000)	0.041 (.003)	0.077 (.000)	0.089 (.000)	1.021	1.014	1.047	1.056	0.003 (.000)	0.004 (.000)	0.004 (.000)	0.004 (.000)	1.003	1.004	1.004	1.004	0.081	63.300
MAXCC	-0.896	-4.131	-5.760	-6.518	0.031 (.000)	0.030 (.003)	0.016 (.390)	0.018 (.354)	1.032	1.030	1.016	1.019	0.002 (.000)	0.004 (.000)	0.004 (.000)	0.004 (.000)	1.002	1.004	1.004	1.004	0.078	63.200
FOUT	-0.952	-4.220	-5.947	-6.785	0.109 (.000)	0.125 (.000)	0.141 (.000)	0.161 (.000)	1.115	1.134	1.152	1.174	0.002 (.000)	0.004 (.000)	0.004 (.000)	0.004 (.000)	1.002	1.004	1.004	1.004	0.089	63.600
AVCC	-1.072	-4.165	-5.727	-6.537	0.165 (.000)	0.085 (.000)	0.029 (.000)	0.061 (.000)	1.128	0.969	0.799	0.775	0.002 (.000)	0.004 (.000)	0.004 (.000)	0.004 (.000)	1.002	1.003	1.003	1.003	0.084	63.100

Table12:Multinomial Bivariate logistic regression of Eclipse2.1

Metric	Const				Coeff.(p-value)				ΔΨ				size coeff.(p-value)				sizeΔΨ				R^2	% of cases correctly
	Nom	Low	Mid	High	Nominal	Low	Mid	High	Nominal	Low	Mid	High	Nominal	Low	Mid	High	Nominal	Low	Mid	High		classified
NOS	-1.514	-4.659	-6.299	-8.044	.006(.000)	.007(.000)	.009(.000)	.012(.000)	1.003	1.003	1.004	1.004	.000(.695)	.000(.777)	-.001(.702)	-.003(.342)	1.000	1.000	0.999	0.997	0.132	74.100
UWCS	-1.516	-4.668	-6.273	-7.903	-.001(.441)	.000(.997)	.001(.693)	.000(.849)	0.999	1.000	1.001	1.000	.004(.000)	.006(.000)	.006(.000)	.006(.000)	1.004	1.006	1.006	1.006	0.129	74.100
CC	-1.520	-4.665	-6.275	-7.901	.000(.954)	.001(.760)	.002(.433)	.001(.695)	1.000	1.001	1.002	1.001	.004(.000)	.006(.000)	.006(.000)	.006(.000)	1.004	1.006	1.006	1.006	0.129	74.100
RFC	-1.707	-5.184	-7.102	-8.879	.021(.000)	.030(.000)	.035(.000)	.036(.000)	1.021	1.030	1.035	1.036	.000(.949)	.000(.491)	.000(.808)	.000(.975)	1.000	1.000	1.000	1.000	0.177	74.500
EXT	-1.677	-5.189	-7.118	-9.577	.027(.000)	.039(.000)	.045(.000)	.057(.000)	1.027	1.040	1.046	1.058	.001(.003)	.001(.000)	.001(.0003	.000(.688)	1.000	1.001	1.000	0.998	0.187	74.700
MPC	-1.677	-5.189	-7.118	-9.577	.027(.000)	.039(.000)	.045(.000)	.057(.000)	1.027	1.040	1.046	1.058	.001(.003)	.001(.000)	.001(.0003	.000(.688)	1.000	1.001	1.000	0.998	0.187	74.700
LMC	-1.513	-4.725	-6.338	-7.965	.059(.000)	.087(.000)	.094(.000)	.090(.000)	1.061	1.091	1.099	1.094	0.003(.000)	.004(.000)	.004(.000)	.004(.000)	1.003	1.004	1.004	1.004	0.138	74.400
TCC	-1.519	-4.669	-6.328	-7.872	.001(.575)	.004(.258)	.010(.012)	.002(.845)	1.001	1.004	1.010	1.002	.004(.000)	.004(.000)	.004(.000)	.006(.000)	1.004	1.004	1.004	1.006	0.130	74.100
PACK	-1.906	-5.471	-7.188	-9.006	.071(.000)	.096(.000)	.101(.000)	.106(.000)	1.074	1.101	1.106	1.112	.002(.000)	.004(.000)	.004(.000)	.004(.000)	1.002	1.004	1.004	1.004	0.223	75.300
NOM	-1.512	-4.687	-6.322	-7.895	-.003(.343)	.002(476)	.005(.149)	.001(.931)	0.997	1.002	1.005	1.001	.005(.000)	.006(.000)	.006(.000)	.006(.000)	1.005	1.006	1.006	1.006	0.130	74.200
LCOM2	-1.530	-4.681	-6.274	-7.891	.000(.004)	.000(.005)	.000(.019)	.000(.095)	1.000	1.000	1.000	1.000	.005(.000)	.006(.000)	.006(.000)	.006(.000)	1.005	1.006	1.006	1.006	0.130	74.200
INST	-1.519	-4.663	-6.254	-7.902	-.001(.670)	-.001(.610)	-.001(.697)	.000(.965)	0.999	0.999	0.999	1.000	.004(.000)	.006(.000)	.006(.000)	.006(.000)	1.004	1.006	1.006	1.006	0.129	74.200
CBO	-1.519	-4.780	-6.494	-8.494	.001(.996)	.031(.004)	.048(.001)	.076(.000)	1.001	1.031	1.049	1.079	.004(.000)	.006(.000)	.006(.000)	.006(.000)	1.004	1.006	1.006	1.006	0.132	74.100
MAXCC	-1.619	-4.730	-6.307	-7.693	.042(.000)	.029(.003)	.023(.083)	-.023(.720)	1.043	1.029	1.023	0.977	.003(.000)	.005(.000)	.005(.000)	.006(.000)	1.004	1.005	1.006	1.006	0.137	74.100
FOUT	-1.528	-4.737	-6.353	-8.195	.011(.347)	.043(.007)	.050(.023)	.083(.000)	1.011	1.044	1.051	1.087	.004(.000)	.006(.000)	.006(.000)	.006(.000)	1.004	1.006	1.006	1.006	0.131	74.200
AVCC	-1.835	-4.836	-6.439	-7.663	.202(.000)	.129(.012)	.134(.130)	-.045(.866)	1.223	1.138	1.144	0.956	.003(.000)	.005(.000)	.005(.000)	.008(.000)	1.003	1.005	1.005	1.008	0.144	74.100

Table13: Multinomial Bivariate logistic regression for Eclipse3.0

Metric	Const				Coeff.(p-value)				ΔΨ				size coeff.(p-value)				sizeΔΨ				R^2	% of cases correctly classified
	Nom	Low	Mid	High	Nominal	Low	Mid	High	Nominal	Low	Mid	High	Nominal	Low	Mid	High	Nominal	Low	Mid	High		
NOS	-1.462	-5.352	-6.935	-7.898	.001(.624)	.003(.177)	.006(.090)	.002(.347)	1.001	1.003	1.006	1.002	.004(.002)	.004(.008)	.002(.532)	.004(.023)	1.004	1.004	1.002	1.004	0.129	74.100
UWCS	-1.495	-5.412	-6.977	-7.921	.006(.001)	.010(.000)	.011(.000)	.009(.020)	1.007	1.010	1.011	1.009	.004(.000)	.005(.000)	.005(.000)	.005(.000)	1.004	1.005	1.005	1.005	0.132	74.100
CC	-1.475	-5.382	-6.945	-7.901	.005(.006)	.009(.000)	.010(.000)	.008(.028)	1.005	1.009	1.010	1.008	.003(.000)	.003(.000)	.003(.000)	.004(.000)	1.003	1.003	1.003	1.004	0.131	74.200
RFC	-1.597	-5.915	-7.518	-8.417	.017(.000)	.028(.000)	.029(.000)	.026(.000)	1.017	1.028	1.029	1.027	.001(.008)	.001(.069)	.001(.306)	.001(.018)	1.001	1.001	1.001	1.001	0.158	74.200
EXT	-1.549	-5.869	-7.179	-8.631	.019(.000)	.032(.000)	.027(.000)	.034(.000)	1.019	1.032	1.027	1.034	.002(.000)	.002(.000)	.002(.000)	.003(.000)	1.002	1.002	1.002	1.003	0.156	74.200
MPC	-1.549	-5.869	-7.179	-8.631	.019(.000)	.032(.000)	.027(.000)	.034(.000)	1.019	1.032	1.027	1.034	.002(.000)	.002(.000)	.002(.000)	.003(.000)	1.002	1.002	1.002	1.003	0.156	74.200
LMC	-1.457	-5.375	-6.935	-7.931	.032(.000)	.050(.000)	.056(.001)	.052(.005)	1.033	1.052	1.057	1.053	.004(.000)	.004(.000)	.004(.000)	.005(.000)	1.004	1.004	1.004	1.005	0.131	74.100
TCC	-1.464	-5.359	-6.929	-7.885	.002(.371)	.008(.022)	.012(.003)	.005(.354)	1.002	1.008	1.012	1.005	.004(.000)	.004(.000)	.003(.000)	.005(.000)	1.004	1.004	1.003	1.005	0.130	74.200
PACK	-1.738	-6.006	-7.714	-9.152	.053(.000)	.075(.000)	.081(.000)	.090(.000)	1.054	1.078	1.084	1.095	.003(.000)	.004(.000)	.004(.000)	.004(.000)	1.003	1.004	1.004	1.004	0.188	74.800
NOM	-1.490	-5.413	-6.991	-7.902	.009(.005)	.015(.000)	.017(.000)	.010(.274)	1.009	1.016	1.017	1.010	.004(.000)	.005(.000)	.005(.000)	.005(.000)	1.004	1.005	1.005	1.005	0.131	74.100
LCOM2	-1.481	-5.330	-6.889	-7.883	.000(.000)	.000(.000)	.000(.013)	.000(.035)	1.000	1.000	1.000	1.000	.005(.000)	.006(.000)	.006(.000)	.007(.000)	1.005	1.006	1.006	1.007	0.131	74.100
INST	-1.471	-5.369	-6.924	-7.894	.004(.053)	.009(.001)	.010(.001)	.008(.072)	1.004	1.009	1.010	1.008	.004(.000)	.006(.000)	.006(.000)	.006(.000)	1.004	1.006	1.006	1.006	0.130	74.100
CBO	-1.522	-5.525	-7.090	-8.082	.022(.000)	.049(.000)	.051(.001)	.050(.007)	1.022	1.050	1.053	1.051	.004(.000)	.005(.000)	.005(.000)	.006(.000)	1.004	1.006	1.006	1.006	0.132	74.100
MAXCC	-1.538	-5.389	-6.969	-7.927	.034(.000)	.027(.003)	.037(.000)	.025(.077)	1.034	1.027	1.038	1.025	.003(.000)	.005(.000)	.005(.000)	.005(.000)	1.003	1.005	1.005	1.005	0.133	74.100
FOUT	-1.535	-5.469	-7.001	-8.003	.067(.000)	.092(.000)	.087(.000)	.088(.000)	1.070	1.096	1.091	1.092	.004(.000)	.005(.000)	.005(.000)	.006(.000)	1.004	1.005	1.005	1.006	0.135	74.000
AVCC	-1.669	-5.454	-7.282	-8.066	.134(.000)	.095(.084)	.205(.000)	.123(.297)	1.144	1.100	1.227	1.131	.004(.000)	.005(.000)	.005(.000)	.006(.000)	1.004	1.005	1.005	1.006	0.136	74.000

Table14: Binary Multivariate LR model statistics for all three version of Eclipse

Eclipse2.0				Eclipse2.1				Eclipse3.0			
Metric	B	S.E	sig	Metric	B	S.E	sig	Metric	B	S.E	Sig
NOS	0.003	0.001	0.012	UWCS	-0.007	0.004	0.045	AVCC	0.122	0.02	0
PACK	0.064	0.004	0	RFC	0.006	0.002	0.004	UWCS	0.011	0.002	0
CBO	0.019	0.008	0.017	CC	0.01	0.004	0.007	PACK	0.056	0.004	0
FOUT	0.073	0.015	0	MAXCC	0.025	0.007	0	FOUT	0.041	0.013	0.001
AVCC	0.115	0.023	0	NLOC	-0.003	0.001	0.003	TCC	0.007	0.002	0
NLOC	-0.002	0.001	0.011	PACK	0.068	0.004	0	CBO	0.015	0.007	0.023
Constant	-1.443	0.057	0	Const	-1.966	0.047	0	Const	-2.085	0.051	0

Table 15:Multinomial Multivariate LR model statistics for all three version of Eclipse

	Eclipse2.0				Eclipse2.1				Eclipse3.0			
	Metric	B	S.E	Sig	Metric	B	S.E	sig	Metric	B	S.E	Sig
	NOM	0.014	0.006	0	AVCC	0.137	0.03	0	CC	0.011	0.002	0
	AVCC	0.136	0.033	0.027	NOS	0.003	0.002	0.026	MPC	-0.017	0.005	0
	NOS	0.004	0.001	0.004	PACK	0.067	0.005	0	FOUT	0.044	0.013	0.001
Nominal	PACK	0.063	0.005	0	NLOC	-0.003	0.001	0.002	RFC	0.014	0.004	0.002
	CBO	0.018	0.008	0.025	TCC	0.009	0.004	0.021	PACK	0.056	0.004	0
	NLOC	-0.002	0.001	0.014	Constant	-2.167	0.064	0	AVCC	0.115	0.026	0
	FOUT	0.076	0.016	0					Constant	-2.113	0.056	0
	constant	-1.5	0.062	0								
Low	NOS	0.01	0.004	0.018	PACK	0.079	0.008	0	CC	0.019	0.004	0
	PACK	0.115	0.009	0	RFC	0.012	0.004	0.004	LMC	-0.044	0.018	0.016
	NLOC	-0.009	0.003	0.005	CBO	0.037	0.018	0.035	PACK	0.066	0.007	0
	LMC	0.041	0.02	0.044	Constant	-5.881	0.224	0	Constant	-6.541	0.267	0
	MAXCC	0.04	0.014	0.006								
	Constant	-5.217	0.268	0								
Mid	PACK	0.124	0.013	0	PACK	0.071	0.011	0	CBO	0.084	0.025	0.001
	CBO	0.079	0.022	0	RFC	0.024	0.007	0	PACK	0.085	0.009	0
	NLOC	-0.017	0.007	0.02	CBO	0.069	0.022	0.002	AVCC	0.279	0.127	0.028
	TCC	0.039	0.019	0.043	Constant	-8.1	0.513	0	Constant	-8.762	0.501	0
	Constant	-6.861	0.703	0								
High	PACK	0.135	0.018	0	NOM	-0.241	0.121	0.047	CC	0.024	0.007	0.001
	CBO	0.099	0.036	0.007	NOS	0.041	0.017	0.017	PACK	0.089	0.011	0
	LMC	0.112	0.044	0.01	PACK	0.076	0.028	0.007	Constant	-9.991	0.902	0
	FOUT	0.102	0.05	0.042	RFC	0.061	0.021	0.005				
	Constant	-8.983	1.056	0	NLOC	-0.035	0.015	0.018				
					Constant	-7.364	1.334	0				

Table 16:Summary of Hypothesis

Hypothesis	Accepted	Rejected /Null Hypothesis accepted
H1	√	-
H2	-	√
H3	√	-
H4	-	√

Table 17: Evaluation result of the performance of Binary Bivariate LR model for Eclipse2.0

Metric+NLOC	TP	FN	FP	TN	SENSITIVITY	1-SPECIFICITY	AUC	Accuracy
NOS	3833	253	2115	447	0.174	0.062	0.556	64.38
UWCS	3833	253	2124	438	0.171	0.062	0.555	64.245
CC	3829	257	2120	442	0.173	0.063	0.555	64.245
RFC	3739	347	1916	646	0.252	0.085	0.584	65.96
EXT	3738	348	1905	657	0.256	0.085	0.586	66.11
MPC	3738	348	1905	657	0.256	0.085	0.586	66.11
LMC	3814	272	2037	525	0.205	0.067	0.569	65.268
TCC	3832	254	2120	442	0.173	0.062	0.555	64.29
PACK	3661	425	1769	793	0.31	0.104	0.603	66.998
NOM	3832	254	2123	439	0.171	0.062	0.555	64.245
LCOM2	3849	237	2134	428	0.167	0.058	0.555	64.335
INST	3833	253	2120	442	0.173	0.062	0.555	64.305
CBO	3829	257	2098	464	0.181	0.063	0.559	64.576
MAXCC	3800	286	2072	490	0.191	0.079	0.561	64.531
FOUT	3797	289	2040	522	0.204	0.071	0.567	64.967
AVCC	3761	325	2038	524	0.205	0.08	0.562	64.455

Table 18: Evaluation result of the performance of Binary Bivariate LR model Eclipse2.1

Metric+NLOC	TP	TN	FP	FN	SENSITIVITY	1-SPECIFICITY	AUC	Accuracy
NOS	5521	333	1769	174	0.158	0.031	0.564	75.08
UWCS	5529	333	1769	166	0.158	0.029	0.565	75.183
CC	5528	328	1774	167	0.156	0.029	0.563	75.106
RFC	5460	464	1638	235	0.221	0.041	0.59	75.978
EXT	5440	485	1617	255	0.231	0.045	0.593	75.991
MPC	5440	485	1617	255	0.231	0.045	0.593	75.991
LMC	5515	368	1734	180	0.175	0.032	0.572	75.452
TCC	5530	327	1775	165	0.156	0.029	0.563	75.119
PACK	5411	561	1541	284	0.267	0.05	0.609	76.594
NOM	5527	329	1773	168	0.157	0.029	0.564	75.106
LCOM2	5523	331	1771	172	0.157	0.03	0.564	75.08
INST	5529	331	1771	166	0.157	0.029	0.564	75.157
CBO	5527	328	1774	168	0.156	0.029	0.563	75.093
MAXCC	5508	342	1760	187	0.163	0.033	0.565	75.029
FOUT	5529	330	1772	166	0.157	0.029	0.564	75.144
AVCC	5477	357	1745	218	0.17	0.038	0.566	74.824

Table 19: Evaluation result of the performance of Binary Bivariate LR model Eclipse3.0

Metric+NLOC	TP	TN	FP	FN	SENSITIVITY	1-SPECIFICITY	AUC	Accuracy
NOS	7400	465	2413	226	0.162	0.03	0.566	74.876
UWCS	7392	468	2410	234	0.163	0.031	0.566	74.829
CC	7407	457	2421	219	0.159	0.029	0.565	74.867
RFC	7328	547	2331	298	0.19	0.039	0.575	74.971
EXT	7318	555	2323	308	0.193	0.04	0.576	74.952
MPC	7318	555	2323	308	0.193	0.04	0.576	74.952
LMC	7394	473	2405	232	0.164	0.03	0.567	74.895
TCC	7406	464	2414	220	0.161	0.029	0.566	74.924
PACK	7273	659	2219	353	0.229	0.046	0.591	75.514
NOM	7398	465	2413	228	0.162	0.03	0.566	74.857
LCOM2	7390	467	2411	236	0.162	0.031	0.566	74.8
INST	7397	465	2413	229	0.162	0.03	0.566	74.848
CBO	7401	468	2410	225	0.163	0.03	0.567	74.914
MAXCC	7382	482	2369	244	0.167	0.032	0.568	75.06
FOUT	7394	465	2413	232	0.162	0.03	0.566	74.819
AVCC	7369	481	2397	257	0.167	0.034	0.567	74.733

Table 20: Evaluation result of the performance of Multinomial Bivariate LR model for Eclipse2.0

Metric+N	TRUE				FALSE				SENSITIVITY				1-SPECIFICITY				AUC				Accuracy			
LOC	Nom.	low	mid	high	Nom.	low	mid	high	Nom.	low	mid	high	nom	low	mid	high	nom	low	mid	high	Nom.	low	mid	high
NOS	344	3	0	0	281	5	0	0	0.14	0.39	0.58	0.58	0.07	0.09	0.09	0.09	0.54	0.65	0.75	0.75	63.39	58.26	58.21	58.21
UWCS	326	2	0	0	286	2	0	0	0.14	0.38	0.58	0.66	0.07	0.09	0.09	0.09	0.53	0.65	0.75	0.79	63.06	58.18	58.15	58.15
CC	329	2	0	0	280	2	0	0	0.14	0.37	0.54	0.67	0.07	0.09	0.09	0.09	0.54	0.64	0.73	0.79	63.16	58.24	58.21	58.21
RFC	479	3	1	0	409	10	0	0	0.20	0.65	0.71	0.75	0.10	0.13	0.13	0.14	0.55	0.76	0.79	0.81	57.99	50.83	50.80	50.78
EXT	509	2	0	0	410	8	0	0	0.21	0.63	0.67	0.75	0.10	0.13	0.14	0.14	0.56	0.75	0.77	0.81	64.38	56.75	56.72	56.72
MPC	509	2	0	0	410	8	0	0	0.21	0.63	0.67	0.75	0.10	0.13	0.14	0.14	0.56	0.75	0.77	0.81	64.38	56.75	56.72	56.72
LMC	377	3	0	1	315	5	0	0	0.16	0.49	0.58	0.67	0.08	0.10	0.10	0.10	0.54	0.70	0.74	0.79	63.54	57.91	57.87	57.88
TCC	325	2	0	0	283	3	1	0	0.14	0.34	0.50	0.67	0.07	0.09	0.09	0.09	0.53	0.64	0.70	0.79	63.00	58.14	58.11	58.11
PACK	611	9	0	0	490	16	0	0	0.26	0.80	0.75	0.75	0.12	0.16	0.17	0.17	0.57	0.83	0.81	0.80	64.92	55.87	55.73	55.73
NOM	325	2	0	0	282	3	0	0	0.14	0.41	0.54	0.67	0.07	0.09	0.09	0.09	0.53	0.66	0.73	0.79	63.12	58.26	58.23	58.23
LCOM2	313	3	0	0	272	4	0	0	0.13	0.40	0.54	0.67	0.07	0.08	0.09	0.09	0.53	0.66	0.73	0.79	63.06	58.39	58.35	58.35
INST	329	2	0	0	279	2	0	0	0.14	0.37	0.50	0.67	0.07	0.09	0.09	0.09	0.54	0.64	0.71	0.79	63.16	58.24	58.21	58.21
CBO	342	1	0	0	285	3	0	1	0.14	0.39	0.58	0.67	0.07	0.09	0.09	0.09	0.54	0.65	0.75	0.79	63.33	58.20	58.18	58.18
MAXCC	361	1	0	0	312	2	1	0	0.15	0.36	0.58	0.58	0.07	0.10	0.10	0.10	0.54	0.63	0.74	0.74	63.15	57.73	57.72	57.72
FOUT	393	2	0	0	330	3	0	0	0.17	0.45	0.63	0.75	0.08	0.10	0.11	0.11	0.54	0.67	0.76	0.82	63.55	57.67	57.64	57.64
AVCC	462	1	0	0	356	4	0	0	0.17	0.38	0.50	0.58	0.09	0.11	0.11	0.11	0.54	0.64	0.69	0.73	64.02	57.08	57.07	57.07

Table 21: Evaluation result of the performance of Multinomial Bivariate LR model for Eclipse2.1

Metric+N	TRUE				FALSE				SENSITIVITY				1-SPECIFICITY				AUC				Accuracy			
LOC	Nom.	low	mid	high	Nom.	low	mid	high	Nom.	low	mid	high	nom	low	mid	high	nom	low	mid	high	Nom.	low	mid	high
NOS	227	5	0	0	218	10	0	0	0.12	0.42	0.61	0.83	0.04	0.05	0.06	0.06	0.54	0.69	0.78	0.89	73.99	71.14	71.08	71.08
UWCS	229	4	0	0	211	10	0	0	0.12	0.40	0.61	0.67	0.04	0.05	0.06	0.06	0.54	0.67	0.78	0.80	74.07	71.18	71.13	71.13
CC	226	4	0	0	211	9	0	0	0.12	0.40	0.61	0.67	0.04	0.05	0.06	0.06	0.54	0.67	0.78	0.80	74.03	71.18	71.13	71.13
RFC	314	3	1	0	291	11	2	0	0.17	0.56	0.79	833.00	0.05	0.07	0.07	0.08	0.56	0.75	0.86	0.88	74.48	70.49	70.46	70.45
EXT	355	2	0	0	314	12	0	0	0.19	0.56	0.75	0.83	0.06	0.08	0.09	0.09	0.57	0.74	0.84	0.88	74.72	70.19	70.17	70.17
MPC	355	2	0	0	314	12	0	0	0.19	0.56	0.75	0.83	0.06	0.08	0.09	0.09	0.57	0.74	0.84	0.88	74.72	70.19	70.17	70.17
LMC	260	2	0	0	231	9	0	0	0.14	0.46	0.61	0.83	0.04	0.06	0.06	0.06	0.55	0.70	0.77	0.88	74.36	71.05	71.03	71.03
TCC	224	4	0	0	205	9	1	0	0.12	0.39	0.61	0.67	0.04	0.05	0.06	0.06	0.54	0.70	0.78	0.81	74.07	71.25	71.19	71.19
PACK	424	4	0	0	343	12	0	0	0.22	0.60	0.75	1.00	0.06	0.09	0.10	0.10	0.58	0.75	0.83	0.95	75.29	69.90	69.85	69.85
NOM	230	5	0	0	207	11	0	0	0.12	0.40	0.61	0.67	0.04	0.05	0.06	0.06	0.54	0.67	0.78	0.80	74.11	71.22	71.16	71.16
LCOM2	237	5	0	0	210	9	0	0	0.12	0.39	0.61	0.67	0.04	0.05	0.06	0.06	0.54	0.67	0.78	0.80	74.14	71.17	71.10	71.10
INST	232	5	0	0	208	10	0	0	0.12	0.40	0.61	0.67	0.04	0.05	0.06	0.06	0.54	0.67	0.78	0.80	74.12	71.21	71.14	71.14
CBO	227	5	0	0	208	10	0	0	0.12	0.40	0.61	0.67	0.04	0.05	0.06	0.06	0.54	0.67	0.78	0.81	74.07	71.22	71.16	71.16
MAXCC	249	5	0	0	229	9	0	0	0.13	0.39	0.61	0.67	0.04	0.06	0.06	0.06	0.55	0.67	0.77	0.80	74.07	70.94	70.87	70.87
FOUT	232	5	0	0	208	8	0	0	0.12	0.39	0.61	0.67	0.04	0.05	0.06	0.06	0.54	0.67	0.78	0.80	74.13	71.22	71.16	71.16
AVCC	258	6	0	0	239	11	0	0	0.14	0.39	0.61	0.67	0.04	0.06	0.06	0.07	0.55	0.67	0.77	0.80	74.03	70.80	70.72	70.72

Table 22: Evaluation result of the performance of Multinomial Bivariate LR model for Eclipse3.0

Metric+N	TRUE				FALSE				SENSITIVITY				1-SPECIFICITY				AUC				Accuracy			
LOC	Nom.	low	mid	high	Nom.	low	mid	high	Nom.	low	mid	high	nom	low	mid	high	nom	low	mid	high	Nom.	low	mid	high
NOS	369	1	0	0	276	8	0	0	0.14	0.57	0.68	0.78	0.04	0.06	0.06	0.06	0.55	0.76	0.81	0.86	74.11	70.61	70.60	70.60
UWCS	373	3	0	0	283	7	0	0	0.14	0.56	0.68	0.78	0.04	0.06	0.06	0.06	0.55	0.75	0.81	0.86	74.07	70.54	70.52	70.52
CC	366	3	0	0	270	8	0	0	0.13	0.57	0.68	0.78	0.04	0.06	0.06	0.06	0.55	0.76	0.81	0.86	74.13	70.68	70.65	70.65
RFC	444	4	0	0	344	8	0	0	0.16	0.62	0.58	0.78	0.05	0.07	0.08	0.08	0.56	0.78	0.75	0.85	74.19	70.00	69.96	69.96
EXT	450	2	0	0	356	5	0	0	0.16	0.61	0.63	0.78	0.05	0.07	0.08	0.08	0.56	0.77	0.78	0.85	74.16	69.90	69.88	69.88
MPC	450	2	0	0	356	5	0	0	0.16	0.61	0.63	0.78	0.05	0.07	0.08	0.08	0.56	0.77	0.78	0.85	74.16	69.90	69.88	69.88
LMC	382	2	0	0	288	6	0	0	0.14	0.57	0.63	0.78	0.04	0.06	0.06	0.06	0.55	0.76	0.78	0.86	74.12	70.51	70.49	70.49
TCC	369	3	1	0	275	6	0	0	0.14	0.57	0.68	0.78	0.04	0.06	0.06	0.06	0.55	0.76	0.81	0.86	74.11	70.63	70.61	70.60
PACK	558	1	0	0	404	8	0	0	0.20	0.64	0.63	0.89	0.05	0.09	0.09	0.09	0.58	0.78	0.77	0.90	74.75	69.45	69.44	69.44
NOM	372	3	0	0	276	6	0	0	0.14	0.55	0.63	0.78	0.04	0.06	0.06	0.06	0.55	0.75	0.79	0.86	74.11	70.59	70.56	70.56
LCOM2	377	3	0	0	292	5	0	1	0.14	0.58	0.68	0.78	0.04	0.06	0.06	0.06	0.55	0.76	0.81	0.86	74.05	70.49	70.46	70.46
INST	366	4	0	0	281	8	0	0	0.14	0.57	0.68	0.78	0.04	0.06	0.06	0.06	0.55	0.76	0.81	0.86	74.03	70.58	70.54	70.54
CBO	370	2	0	0	278	5	0	0	0.14	0.57	0.63	0.78	0.04	0.06	0.06	0.06	0.55	0.76	0.79	0.86	74.11	70.60	70.58	70.58
MAXCC	385	2	0	0	293	7	0	0	0.14	0.55	0.68	0.78	0.04	0.06	0.06	0.07	0.55	0.75	0.81	0.86	74.10	70.45	70.43	70.43
FOUT	374	2	0	0	290	5	0	0	0.14	0.56	0.63	0.78	0.04	0.06	0.06	0.06	0.55	0.75	0.78	0.86	74.02	70.48	70.46	70.46
AVCC	381	3	0	0	305	8	0	0	0.14	0.58	0.68	0.78	0.04	0.06	0.07	0.07	0.55	0.76	0.81	0.86	73.96	70.36	70.34	70.34

Table 23: Evaluation result of the performance of Binary Multivariate model LR for all 3 version of Eclipse

Version	TP	TN	FP	FN	SENSITIVITY	1-SPECIFICITY	AUC	Accuracy
Eclipse2.0	3649	879	437	1683	0.34	0.11	0.62	68.11
Eclipse2.1	5386	595	309	1507	0.28	0.05	0.61	76.71
Eclipse3.0	7250	727	376	2151	0.25	0.05	0.60	75.94

Table 24: Evaluation result of the performance of Multinomial Multivariate LR model for all 3 version of Eclipse

Version					Mutinomial Multivariate model																			
	TRUE				FALSE				Sensitivity				1-specificty				AUC				Accuracy			
	Nom.	low	mid	high	nom	low	mid	high	nom	low	mid	high	nom	low	mid	high	nom	low	mid	high	nom	low	mid	high
Eclipse 2.0	694.00	12.00	2.00	2.00	495.00	20.00	2.00	0.00	0.29	0.79	0.75	0.75	0.12	0.17	0.18	0.18	0.59	0.81	0.81	0.80	65.91	55.66	55.51	55.51
Eclipse 2.1	453.00	5.00	2.00	3.00	368.00	11.00	1.00	1.00	0.26	0.58	0.75	1.00	0.07	0.10	0.11	0.11	0.59	0.74	0.83	0.97	62.29	55.55	55.51	55.52
Eclipse 3.0	622.00	7.00	3.00	1.00	409.00	10.00	0.00	0.00	0.23	0.64	0.74	0.89	0.06	0.10	0.10	0.10	0.59	0.78	0.83	0.90	64.83	55.58	55.52	55.49