Software Quality Prediction using different Software Metrics

1 Asif Ali  
2 Dr. Kavita Choudhary  
3 Dr. Ashwin Sharma

ABSTRACT
This paper provides mathematical implications of software quality prediction using different software metrics. We have considered object oriented module as the dataset. The data used for the experimentation have class, object, inheritance and dynamic behavior. The data modularity considered for this work is 11-20. The software quality predictions are applied after the chi-square test. If the modules qualify the chi-square test then those modules are selected for software metric analysis. This testing is based on the object oriented parameters like inheritance, class, reference and run time binding. It will provide us the probability distribution values. Then F- measure (FM), Power (PO) and Odd Ratio (OR) are applied for the quality analysis based on the metrics passed. We have presented mathematical implications with all the steps for the calculation and determine the modularity with the help of above software metrics.

KEYWORDS
Software quality, Clustering, Chi-Square Test, FM, PO, OR

INTRODUCTION
To discovering the product particularity is the greatest challenge in today's situation. There are a few examination introductions in this course. Notwithstanding the measure of exertion spent in the configuration and utilization of flaw expectation models, programming deficiency forecast research region still stands incredible difficulties. Lamentably, none of the strategies which are made in couple of years prior fulfills the relevance in the product business because of a few reasons including the absence of programming devices to robotize this expectation prepare, the unwillingness to gather the deficiency information, and the other down to earth issues [11-14]. The customary way which is utilized from the earliest using so as to star point is to gauge programming quality programming measurements and shortcoming information gathered from past framework discharges or comparative undertakings to develop a quality-expectation or quality-order model. At that point designers utilize this model to foresee the issue inclination of programming segments being developed. Past exploration [1] has demonstrated that product quality models in view of programming measurements can yield expectations with helpful exactness. Such models can be used to anticipate the response variable that can either be the class of a portion or a quality component for a fragment. The past is normally suggested as gathering models [2] while the latter is for the most part implied as desire models [3]. The focal point of this paper is on the past, i.e., request models. Often, predicting the amount of imperfections is excessive. It is the entire all the all the more confounding subsequent to, as De Vaux and Hand [4] Stated, 60-95% of the effort of data examination is making use for the cleaning. The scope of examination like information structures and data mining the impact of poor data has been seen as an issue which ought to be tended to by database originators and data customers alike. Redman [5] for instance communicated that poor data quality is an issue which consequences for all pieces of the economy: associations, governments, and the insightful world and their customers”, and Wand and Wang [6] forewarned of the great impact of poor data quality on the ampleness of an affiliation. Information Clustering and Classification are examined in [7][8][9][10]. At that point we apply the testing on article arranged property taking into account chi square test.

LITERATURE SURVEY
In [15] authors have depicted an exploratory examination strategy that addresses two difficulties and that is constructed with bunching and the assistance of a product building master. It is an unsupervised technique since named preparing information is not needed to foresee the deficiency inclination of programming modules. In [16] authors examined about several programming grouping calculations Most of these calculations have been connected to specific programming frameworks with significant achievement. On the other hand, the topic of how to choose a product bunching calculation that is most appropriate for a particular programming framework stays unanswered. In [17] authors presented, a novel correlation technique for programming deteriorations that can be connected to both settled and level disintegrations. The advantages of using this system are exhibited in both explanatory and exploratory design. We likewise contrast with the END structure, the main other existing system for settled decay examination. In [18] authors have utilized subtractive grouping based fluffy surmising framework approach which is utilized for right on time recognition of flaws in the capacity arranged programming frameworks. The execution of the proposed methodology is better if there should be an occurrence of Joined Model. In [19] authors have presented and evaluate

1 Research Scholar, Department of Computer Science & Engineering, Jagannath University Jaipur, Rajasthan
Email: asifali29@gmail.com

2 Associate Professor, Department of Computer science & Engineering, Jagannath University Jaipur, Rajasthan
Email: kavita@jagannathuniversity.org

3 Asst. Professor, Department of Computer science & Engineering, GEC Jhalawar (Raj.), India
Email: ashwini.sharma.in@gmail.com

Copyright ©IJICCT, Vol IV, Issue I(Jan-Jun 2016):ISSN 2347-7202
the thought of bunching calculation likeness. It depends on the idea that calculations with diverse destinations ought not be straightforwardly thought about. Of course, we find that few of the distributed calculations in the writing are not practically identical to one another. In 2012, [20] authors have discussed about bunching which is the unsupervised characterization of examples into gatherings. A bunching calculation parcels information set into a few gatherings such that closeness inside of a gathering is bigger than among gatherings the grouping issue has been tended to in numerous connections and by scientists in numerous orders; this mirrors its expansive request and helpfulness as one of the progressions in exploratory information investigation. In 2012, Saifi et al. [21] study and investigation of information quality is a critical issue which has been tended to as information warehousing, information mining and data frameworks. It has been concurred that poor information quality will affect the nature of consequences of investigations and that it will in this way effect on choices made on the premise of these outcomes. An endeavor to enhance order precision by pre-grouping did not succeed. In any case, screw up rates within clusters from planning sets was unequivocally connected with slip rates within the same gatherings on the test sets. Coupling measures are similarly checked in [22]. In 2014, Shihab et al. [23] recommended a methodology that predicts dangerous changes, showing how Software imperfection expectation can be additionally enveloping and proactive. In 2015, Tripathi et al. [24] created models to anticipate the change inclination of the classes in the item situated framework by investigating the relationship between the article arranged measurements and change inclination. The model proposed is additionally accepted by article arranged open source programming. Their outcomes acquired demonstrates that there is a noteworthy relationship between the article arranged measurements and change inclination. In 2015, Biray et al. [25] proposed a learning-based choice tree model for identifying mistake inclined classes with auxiliary configuration imperfections. The principle curiosity in our methodology is that we consider EFs and change tallies of classes to develop fitting information set for the model's preparation. Efficient data classification has been introduced in [26] in different aspect.

For the testing we have considered object oriented modules. This data set can be tested further by Chi-Square test Algorithm. We have a null hypothesis which states that the observed distribution is not significantly different from the expected distribution and of course use words relevant to that particular problem. The choice guideline for this test will dependably be the place the discriminating worth must be perused from the dissemination table. The main two numbers expected to turn upward this discriminating worth are the level of hugeness and the quantity of degrees of flexibility. The degrees of flexibility for this test will be characterized as the quantity of classifications less 1. This is the way we locate the basic worth for a specific issue: assume that we utilize and have 5 degrees of opportunity (6 classifications). We can find the critical value by looking at the point where the two arrows in the table meet. The test statistic is \( \chi^2 \) where E and O are the expected and observed frequencies per category.

This test is performed according to the following formula shown below:
\[
\chi^2 = \sum \frac{(E - O)^2}{E}
\]

Where E and O are the expected and observed values in each group of parameter’s defined.

If the calculated set shows the p value less than the threshold, the suggested hypothesis is then rejected, and presumes that some element other than chance is working for the deviation. The chi-square test will produce the fittest results. Then in the next step we have used quality metrics. The quality metrics are F-Measure, Odd Ratio and Power. In this section we discuss the above in details.

F- Measure (FM)

The relative fitness in terms of sub groups are compared based on the retrieved values. It can be compared by FM. For comparison we suppose that there are two groups or cluster obtained as the intermediate phase results:

C = the correct vector of bit masks = \{c1, c2, c3...cn\}

K= the vector of bit mask results of some algorithm = \{k1, k2, k3...km\}

Then a substantial matrix should be created based on the two groups for matching in the found K. Here K represents the clusters. Based on the cluster the matrix will match the values and separate it accordingly.

FM is calculated based on precision (PR) and recall (RC) values. It is based on true positive (TP), true negative (TN), false positive (FP) and false negative (FN).

\[
F= \frac{(2 * PR * RC)}{(PR + RC)}
\]

Where:

\[
PR = \frac{TP}{TP + FP}
\]
Software Quality Prediction using different Software Metrics

RC = TP / TP + FN

Odd Ratio (OR)
The OR estimates the association and relevancy between the independent data values. It is utilized as an elucidating measurement, and assumes a vital part in logistic relapse.

OR= 2 * RC / (1 - PR * RC)

The chances proportion is one of a scope of measurements used to evaluate the danger of a specific result (programming estimation) if a sure element (or introduction) is available. The chances proportion is a relative measure of danger, letting us know what amount more probable it is that somebody who is presented to the element under study will build up the result when contrasted with somebody who is not uncovered. Third, chances proportions are a typical method for exhibiting the aftereffects of a meta-examination a measurable investigation for joining the consequences of a few studies, utilized inside efficient surveys.

Power (PO)

Power (PO) is defined as:

PO= ((1-PR)k-(1-RC)k)

At long last in view of the above parameters we can continue to the examination which depends on expense and time. We arrive to enhance the adequacy of use as a product administration. By the above methodology we can locate the better result.

RESULT EVALUATION

For result evaluation we have considered the set between 11-20 intervals. Table 1 shows the record considered in 11-20 intervals. Table 2 shows the predictive value obtained for table 1.

Table 3 shows the result of FM based on the deviation. In the first row of table 4.6 the observed value (O) is 18, expected value (E) is 15 so the deviation D = 3(O-E). Then the qualifying factor can be calculated based on the formula of chi-square test that is P=d2 /15. So the obtained value of P is 0.6 which means it qualifies for the next data processing. Then in the next process we calculate the FM = (2 * PR * RC) / (PR + RC).

PO= ((1-PR)k-(1-RC)k)

The same procedure is applied in case of object, inheritance and dynamic memory allocation (DMA).

Table 3 Result of FM ratio

Table 4 Result of OR ratio

Table 5 Result of PO ratio

Table 6 Results in individual category

Copyright ©IJICCT, Vol IV, Issue 1 (Jan-Jun 2016): ISSN 2347-7202
CONCLUSION
In this paper mathematical explanation has been presented for the quality estimation for object oriented modules. These above phenomena provide us to measure with the quality metrics like F measure, Odd Ratio and Power. This provides an insight for the software metrics calculation and testing. For ease we use the data set of java programs, because in today’s era the live emphasize in software industry is based on object oriented programming.

REFERENCES
Software Quality Prediction using different Software Metrics
