# A Review of Test Case Prioritization using a novel Density based K-means Clustering

[1]Ishadeep Kaur Luthra, [2]Harsimranjit Kaur
[1]Mtech Scholar, [2]Assistant Professor
[1]Department of CSE
[1]Chandigarh Engineering college, Mohali, India

***Abstract* - Regression testing is used to validate the modified software, but unfortunately regression testing is very time consuming and cost inefficient as engineers has to perform the whole test again and again. In this case prioritization technique is used in regression testing. Different methodologies has been used till date for prioritization regression testing such as Genetic algorithm, Greedy search algorithm, particle swarm optimizer and many more. Many new method has also been used along with prioritization is clustering approach. It divides the test cases according to their properties and form different clusters. Prioritization is done on the basis of clusters formed. Hence, this paper reviews the different methods used in prioritization regression testing and what can be used next to obtain results that are cost and time efficient and give maximum result.**

***Index Terms* - Component, formatting, style, styling, insert.**

## I. INTRODUCTION

Regression Testing is one of the type of software testing whose motive is to modify software for better result and confirms that no damage is made to the software. The maintenance process through Regression testing is expensive. The method of using it is by running some or all the test cases created. And to test the latest modifications made in the software with comparison to the previous version of software. But this method is very time consuming and cost consuming. Thus to reduce the cost, time and resources. Test case prioritization using clustering based algorithm is adopted in Regression testing. Selecting the Test case on the bases of priority using density based clustering method. The process of minimizing a test cases based on prioritization from the large density of test cases is known as "Test case prioritization using clustering algorithm".

While consider a clustering approach,[13] it could simplify test case prioritization processes by dividing test cases into groups that have common properties. It has been conjectured that if test cases have common properties, then test cases within the same group may have similar fault detection ability. If this conjecture is correct, engineers may be able to manage regression testing activities more efficiently by using test case prioritization techniques that can utilize clustering approaches. For instance, if an organization does not have enough time to run all the test cases, by running a limited number of test cases from each cluster, they still could have a better chance to catch more faults than otherwise.

Thus, in this paper, we investigate whether the use of a clustering approach can help to improve the effectiveness of test case prioritization techniques. In this work, we review different papers which have worked on Test case prioritization Regression testing and Test case prioritization using clustering in Regression testing, on the basis of code coverage, code complexity and history data on real faults.

To investigate the effectiveness of our approach, we have designed and performed empirical studies using an industrial software product K-MEANS density based clustering algorithm[14]. In density-based clustering algorithms, which are designed to discover clusters of arbitrary shape in databases with noise, a cluster is defined as a high-density region partitioned by low density regions in data space. DBSCAN (Density Based Spatial Clustering of Applications with Noise) is a typical density-based clustering algorithm.

## II. RESEARCH HISTORY

As software evolves, software engineers perform regression testing on it to validate new features and detect whether corrections and enhancements have introduced new faults into previously tested code. In practice, engineers often reuse all of the existing test cases to test the modified version of the software system; however, this retest-all approach can be expensive.

Researchers have studied various methods for improving the cost-effectiveness of regression testing. Regression test selection techniques reduce testing costs by selecting, from the existing test suite, a subset of test cases to execute on the modified software system. Two recent surveys provide an overview of regression test selection techniques. Test suite minimization techniques try to reduce the size of test suites by identifying and eliminating redundant test cases. Both of these techniques reduce costs by reducing testing time and maintenance effort, but unless they are safe they can omit test cases that would otherwise have detected faults.

To solve the problem of test case prioritization various algorithm such as Search Algorithm and Meta heuristic algorithm are used. Manika et al[11] proposed 3 phase approach to solve test case prioritization. In the first phase they removed redundant test caes by simple matrix operations. In second phase test cases are selected from test suits such that selected test cases represent the minimal set which covers all faults and also at the minimum execution time. For this they are using multi objective swarm optimization (MOPSO) which optimize fault coverage and execution time. In the third case priority to test cases are obtained from the second phase. Priority is calculated on the ratio of fault coverage to execution time of test cases. Higher the value of ratio, higher the priority. Carlson et al[13] implemented new prioritization technique that incorporates clustering approach which utilize code coverage, code complexity, and history data on Real faults. They have designed and conducted empirical studies using an industrial software product, Microsoft Dynamic Ax, which contain real faults. R.Krishnamoorthi et al[15] aims to improve the effectiveness of Regression Testing by ordering the Test Cases so that the most beneficial are executed first. The technique of genetic algorithm is used. The proposed technique prioritizes subsequences of original test suites so that the new suites which is in run within a time constrained execution environment, will have a superior rate of fault detection compared with randomly prioritize test suites. Mathusamy et al[16] prioritize the test cases depending on the business impact, importance and frequently used functionalities. Further it gives an improved rate of fault identification, when test suites cannot run to completion. Mumtaz et al[14] proposed to overcome the drawbacks of DBSCAN[18] and kmeans[17] clustering algorithm. This handles clusters of circulary distributed data points and slightly overlapped clusters. Chen et al[19] uses classification algorithm for early fault detection rate and to guide the scheduled process based on code change information.

*Examples of prioritization Regression Testing*

| Projects | Algorithms used |
|---|---|
| Test Case Prioritization | • Particle swam Optimizer[2],[11]<br>• Genetic Algorithm[4],[5]<br>• Classification Algorithm[19]<br>• Greedy Search technique for Set covering[6] |
| Test Case Prioritization using Clustering | • Hierarchical Clustering method[13] |

Fundamental Building Blocks of Priority testing using Clustering Algorithm

- Underlying relationships in massive data about software systems
- Data correlations and patterns
- Test harnesses
- Bug reports
- Clustering approach

## III. PROBLEM FORMULATION

The problem of test case prioritization has gained significant attention over the last few years as software testing forms a major section of the whole software development process. The cost of software development is directly dependent on the testing effort. The thesis aims to reduce this cost by prioritizing test cases and running the tests for the selective test cases as per the available time and manpower. There are a number of test cases available which can consume a lot of time and effort. A selective number of test cases needs to be selected which would be otherwise used for the same purpose. The priorities of the test cases needs to be decided on the basis of several parameters. The parameters for the test case prioritization needs to be chosen and a model needs to be developed which would set priority among the test cases. First of all a data set needs to be generated which would be utilized for our proposed algorithm testing. Then the dataset needs to be preprocessed for outlier removal and redundancy removal. Then a technique for clustering of the test cases needs to be developed which would be utilized for the above mentioned problem.

## IV. PROPOSED METHODOLOGY

It is proposed to use a novel density based K-means algorithm for test case prioritization for regression testing. The naive k-means algorithm partitions the dataset into ‘k’ subsets such that all records, from now on referred to as points, in a given subset "belong" to the same center. Also the points in a given subset are closer to that center than to any other center. The algorithm keeps track of the centroids of the subsets, and proceeds in simple iterations. The initial partitioning is randomly generated, that is, we randomly initialize the centroids to some points in the region of the space.

The k-means needs to perform a large number of "nearest-neighbour" queries for the points in the dataset. If the data is‘d’ dimensional and there are ‘N’ points in the dataset, the cost of a single iteration is O(kdN). As one would have to run several iterations, it is generally not feasible to run the naïve k-means algorithm for large number of points. Sometimes the convergence of the centroids (i.e. C(i) and C(i+1) being identical) takes several iterations. Also in the last several iterations, the centroids move very little. As running the expensive iterations so many more times might not be efficient, we need a measure of convergence of the centroids so that we stop the iterations when the convergence criteria is met. Distortion is the most widely accepted measure.

Clustering error measures the same criterion and is sometimes used instead of distortion. In fact the points minimizes the distortion for the points in the cluster. Also when another cluster center is closer to a point than its current cluster center, moving the cluster from its current cluster to the other can reduce the distortion further. The above two steps are precisely the steps done

by the k-means cluster. Thus k-means reduces distortion in every step locally. The k-means algorithm terminates at a solution that is locally optimal for the distortion function. Hence, a natural choice as a convergence criterion is distortion. Among other measures of convergence used by other researchers, we can measure the sum of Euclidean distance of the new centroids from the old centroids.

**REFERENCES**

[1] Z. Li, M. Harman, and R. M. Hierons, "Search Algorithms for Regression Test Case Prioritization," IEEE Trans. Software Eng., pp.225-23 7 Apr.2007
[2] K. H. S. Hla, Y. Choi, J. S. Park, "Applying Particle Swann Optimization to Prioritizing TestCases for Embedded Real Time Software Retesting, "Proceedings of the IE E E 8th InternationalConference on Computer and Information Technology Workshops, pp. 527-532. 2008
[3] Y. Singh, A Kaur, B.Suri,"Test case prioritization using ant colony optimization," ACM SiGSOFJ Soflware Engineering Notes, Vol.35 No.4, pages 1 -7, July 20 1 0.
[4] J. Wang, Y. Zhuang, C. Jianyun, "Test Case Prioritization Technique based on Genetic Algorithm", International Conference on internet Computing and information Services, Hong Kong pp. 1 73 - 175. 2011
[5] S. Sabharwal, R. Sibal, C. Sharma, "A genetic algorithm based approach For prioritization oftest case scenarios in static testing," Proceedings of the 2nd International Conference on computer and Communication Technology, IEEE Xplore Press, Allahabad, pp: 304-309. Sept.15- 1, 2011
[6] S. Tallam, N. Gupta, "A concept analysis inspired greedy algorithm for test suite minimization, "Proceedings of the 6th ACM SiGPLAN-SiGSOFf workshop on Program analysis for soflware tools and engineering. ACM, Lisbon, Portugal, 2005.
[7] G Rothermel, R Untch, C Chu and M Harrold, 'Test case prioritization: An empirical study , in Soflware Maintenance, 1999. (ICSM' 99) proceedings. IEEE International conference, o n pages 1 79- 1 88 IEEE, 1 999.
[8] S. Elbaum, A Malishevsky, and G Rothermel "Prioritizing test cases for regression testing", Proc. The 2000 ACM SiGSOFJ international Symposium on Soflware Testing and Analysis, Portland, Oregon, USA, Aug-2000, 102-112.
[9] S. Elbaum, A. Malishevsky and G. Rothermel, "Test case prioritization: A family of empirical studies", IEEE Transactions o n Soflware E ngineering, vol. 28(2), 2002, pp. 1 59- 1 82.
[10] W. Wong, J W. Wong, J. Horgan, S. London and H.Agrawal, "A Study of Effective Regression Testing in Practice", In Proc. of the Eighth Inti. Symp. on Softw. Inti. Symp. onSoflw. Rei. E ngr., pages 230-238, Nov. 1 997.
[11] Manika Tyagi, Sona Malhotra, "Test Case Prioritization using Multi Objective Particle Swarm Optimizer", Department of CSE U.I.E.T., Kurukshetra University
[12] AP. Mudgal, "A Proposed Model for Minimization of test suite" Journal of nature inspired computing, vol. 1, No. 2, PP. 34-37, 2013.
[13] R.Carlson, Hyunsook Do, Anne Denton “A Clustering Approach to Improving Test Case Priotization: An Industrial Case Study”.
[14] K.Mumtaz and Dr. K. Duraiswamy, “A Novel Density Based improved k-means Clustering Algorithm- Dbkmeans” IJCSE, Vol 02, No. 02, 2010
[15] R.Krishnamoorthi and S.A.Sahaaya Arul Mary “Regression Test Suite Prioritization using Genetic Algorithms” Department of Computer Science and Engineering, Bharathidasan Institute of Technology, Anna University, Trichy-24, India.
[16] Thillaikarasi Muthusamy and Dr. Seetharaman.K “Effectiveness of test case prioritization techniques based on Regression testing” International Journal of Software Engineering & Applications (IJSEA), Vol.5, No.6, November 2014
[17] Kaufman L. and Rousseeuw P. J., Finding Groups in Data: AnIntroduction to Cluster Analysis, John Wiley & Sons, 1990.
[18] Ester M., Kriegel H., Sander J., Xiaowei Xu, “A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise”, KDD’96, Portland, OR, pp.226-231, 1996.
[19] Xiang Chen, Zhaofei Tan, Jian Xia, Pengfei He “Optimizing Test Case Execution Schedule using Classifiers ” JOURNAL OF SOFTWARE, VOL. 9, NO. 10, OCTOBER 2014