Vol 8. No. 1 – March, 2015
African Journal of Computing & ICT

www.ajocict.net

Prediction of Software Maintenance Effort of Object Oriented Metrics Based Commercial Systems

V.K. Yadav
Department of Computer Science and Engineering
S.L.S.E.T.
Kichha Uttarakhand, India
vinodrockcsit@gmail.com
Phone: +919451611568

S. Kumar & M. Mittal
Research Scholar, Department of Computer Science
Gurukula Kangri University
Haridwar, Uttarakhand, India
satendra04cs41@gmail.com
mittal.mohit02@gmail.com
Phones: +918923683462, +918394832967

ABSTRACT

The software systems really advanced and seize with problems on their maintenance. The software maintenance work is presently one in every one of the foremost tough, time-consuming, expensive and costly tasks in the software development life cycle. It's invariably been a vital issue for software project developers. Therefore, it is worthwhile to develop an object oriented system with easy maintenance at design phases. This analysis concentrates the development of a method based on the data mining techniques as K-means and Hierarchical clustering are implemented in MATLAB package on two commercial systems are UIMS (User Interface Management System) QUES (Quality Evaluation System). The maintenance effort is measured by the number of lines changed (addition or a deletion) per class which are already pre defined classes of UIMS and QUES. It is ascertained that the algorithms will be able to decide the cluster with Easy, Medium, and High conditions of maintainable classes of similarity based on object oriented metrics. This paper is most beneficial for the software maker and maintainers to take the necessary steps at design level to design of maintainable object oriented software.

Keywords- Software Metrics; Clustering; K-mean clustering algorithm; Hierarchical clustering

African Journal of Computing & ICT Reference Format:
V.K. Yadav, S. Kumar & M. Mittal. (2015 Prediction of Software Maintenance Effort of Object Oriented Metrics Based Commercial Systems. Afr J. of Comp & ICTs. Vol 8, No. 1. Pp 163-.172.

1. INTRODUCTION

Software maintainability is going to be a seamless challenge for several years to come back. Software maintenance is in view a very necessary and typical or complex section in software life cycle usually constituting 50-70 proportion of total effort allocated to a software system [1]-[2]. The most of researches are working on these prediction approaches [3] but the still need for improvement is always there. The software maintainability is an important aspect to correct errors, enhance features and port to new platforms. In recent years, data mining technology and its ability to deal with huge amounts of data has been considered a suitable solution in assisting software maintenance [4]-[7]. It is accept as true that predicting the maintenance at the design level can facilitate to software designers and maintainers to change the architecture of the software system for higher Performance that will lead to the general reduction of maintenance costs [8].

Keeping this view in mind, the data mining technique of clustering based approach for designing maintainable object oriented systems using the K-means and Hierarchical clustering algorithms are proposed.

2. RELATED WORK

Many papers have shown that clustering is a technique that is used for general application likes pattern recognition, spatial data analysis, image processing and the WWW. It is also used for analyzing the architecture of the software system for better performance and maintenance [3], [5], [9], [14]. Data mining and uncovering hidden patterns have been proposed as a means to support the evolution and assessment of the maintainability of industrial scale software system [10]-[11].

The fault prediction model should be adequate to produce reliable product in accessible time frame and budget, meeting the customer's necessitates. However, numerous faults in the prediction model using clustering algorithms are already available in the literature however still there is need to develop a strong model [12]-[13]. There are many studies for evaluating a system's maintainability and controlling the hassle needed to hold out maintenance activities [14]-[17]. An approach for the evaluation of dynamic clustering was presented in [15], [18], and [19].

The scope of this resolution was evaluating the usefulness of providing dynamic dependencies as input to software clustering algorithms. Various techniques for maintainability consistent with ISO/IEO-9126 have been proposed [20]. Maintainability is characterized by the analyzability, changeability, testability, maintainability compliance. The clustering information extracted from Java source code aiming at capturing program structures and achieving better program understanding methodology have been presented [18].

The value of this work that differentiates it from what presented above, is that here we have a tendency to don't cluster raw software measurement data. Instead, we provide the software designers and maintainers to change the design of the software system for higher performance that leads the general reduction of maintenance by using centroids based clustering method K-means and overlapping based hierarchical clustering are used in this paper.

3. CLUSTERING

Cluster analysis is the process of discovering groups of objects in such a way that data points in same clusters have high intra-class similarity and data points in different clusters have terribly low inter-class similarity. Clustering is a data mining technique to groups datasets primarily based on distance or similarity. Clustering is particularly helpful in issues for unsupervised learning, automatic classification, and typological analysis and clustering is additionally where there is very little previous information available about the data, and software makers must have to take decision of logical and physical storage possibility on these data. These restrictions build this methodology applicable for the investigation of a logical or natural association between two or more things among the data points to make an evaluation concerning their structure [3]. Clustering creation is often performed in a number of ways as follows:

- Partitioning Methods
- Hierarchical Methods
- Density-Based Methods
- Grid-based Methods
- Model-Based Methods

In these methods K-means is one the most popular partitioning clustering algorithms in which each cluster is manipulated by the centroids of the data points (objects) in the cluster. Its main problems are that it is sensitive to noise and to the initial partitioning. As many possible initial partition lead to many different results, the final clustering is influenced by the initial partition.

4. K-MEANS CLUSTERING APPROACHES

The K-means algorithm is centroids based partitioning technique. Here K stands for the number of partitions or clusters. To perform the K-means algorithm we have to take any arbitrary objects as the initial centroids of the first K objects as initial centroids. In this paper we use the euclidean distance between point's r and c in p-dimensional space. The well known metric is euclidean distance, defined as in equation (1):

$$d(r,c)=|x_{r1}-x_{c1}|+|x_{r2}-x_{c2}|+.....|x_{rp}-x_{cp}| \quad (1)$$

Where

$r = (x_{r1}, x_{r2}, ..., x_{rp})$ & $c = (x_{c1}, x_{c2}, ..., x_{cp})$

It uses square error function (terminating condition) to give aggregate dissimilarity of clusters that is defined as follows in equation (2):

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} |p - m_i|^2 \quad (2)$$

$E =$ Summations of the square error in data sets.
$p =$ Points in space representing a given object.
$m_i =$ Mean of cluster c_i.

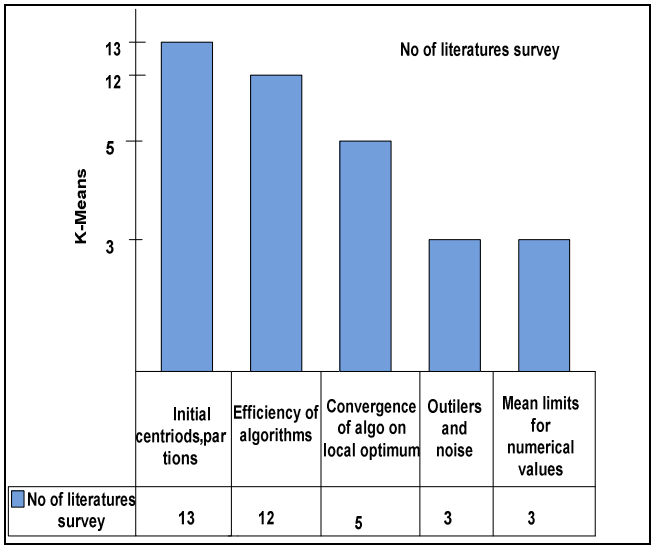


Figure 1: K-means and no. of literatures

Several of the have been proposed in the literature, many authors have tried to exploit the features of K-means. An interesting paper on K-means clustering has been proposed in [21]. According to above figure 1 shows thirteen papers of algorithm's sensitivity to initial conditions: the number of partitions, the initial centroids. The efficiency of the algorithm show by twelve papers. The convergence of algorithm to local optimum rather than a global optimum show by five papers. Three papers show that the K-means is sensitive to outliers and noise. Three papers show the definition of "mean" limits the application only to numerical variables.

5. K-MEANS ALGORITHM FOR SIMULINK

This section presents the K-means clustering algorithm applying it to evaluate a maintenance effort involved on classes within UIMS and QUES object oriented system models. The algorithm used in simulation (in MATLAB) is given below:

Step 1: Initialization
- 1.1 Input the number of classes with their attributes.
- 1.2 Take any random objects as the initial centroids.
- 1.3 Input number of K cluster.

Step 2: Classification
- 2.1 Compute the distance using most popular distance measure is city block distance or Manhattan between classes and randomly choose objects.
- 2.2 Objects are including to the group related to this centroids.

Step 3: Centroids Calculation
- 3.1 For each group generated in the previous step, its centroids are recalculated.

Step 4: Come together or towards the same point's condition
- 4.1 Stopping when reaching a given number of iterations.
- 4.2 Stopping when there is no exchange of objects among groups.

Step 5: If the step 4 is not satisfied then steps 2 to step 4 must be repeated.

Step 6: Produce the group of classes in a given K cluster.

Step 7: Finally, produce the cluster whose maintenance effort are Easy, Medium and High.

6. HIERARCHICAL CLUSTERING APPROACHES

Hierarchical cluster is an agglomerate cluster methodology. As its name suggests, the thought of this methodology is to make a hierarchy of clusters. This method is usually continuing till there's one giant cluster containing all the first information points. Ranked cluster leads to a "tree", showing the link of all of the first points. Hierarchical cluster rule is of 2 types: agglomerate ranked cluster rule dissentious ranked cluster rule each this rule aren't the same as one another. The agglomerate ranked cluster works by grouping the information on the idea of the closed distance live of all the pairwise distance between the info purposes. There are several on the market strategies that distance to contemplate once the teams are fashioned. a number of them are: single linkage, complete linkage, average linkage, Centroids distance, ward's methodology - total of square geometer distance is reduced.
This way we tend to proceed grouping the info till one cluster is made. Currently on the idea of dendogram graph we will calculate what percentage numbers of clusters ought to be really gift.

7. ALGORITHMIC STEPS FOR AGGLOMERATIVE HIERARCHICAL CLUSTERING

Let X = $\{x_1, x_2, x_3, \ldots\ldots\ldots\ldots x_n\}$ be the set of data points.

Step 1: In the first step we have to calculate the distance between every a mated couple of objects within the data set byusing pdist function in MATLAB.

Step 2: In the second step, we bring together of object by using linkage function that is in much closed to each other. The linkage function uses the distance calculation that is formed by pdist function in step 1 to determine the closeness of objects. The objects are bringing together into binary clusters; the previously formed clusters are grouped into of wide range clusters until a hierarchical tree is formed.

Step 3: In the last step, calculate where to cut the hierarchical tree into object's clusters. We use the cluster function of MATLAB to merge the furthest part of the hierarchical tree, and allocate every object below each cut to a single cluster.

8. PROPOSED WORK AND SELECTED OBJECT ORIENTED METRICS

In this paper the k-means clustering data mining techniques are implemented on UIMS class's data and QUES class's data. The UIMS and QUES are used in this inspection, which have been presented in [22]. The UIMS contains 39 classes with 11object oriented metrics and QUES contains 71 classes with 11 object oriented metrics. It observed that the data mining algorithms are able to decide the maintainers to take the necessary action at design level. The description of proposed work and selected object oriented metrics for UIMS and QUES are given in figure 2 and table 1 respectively.

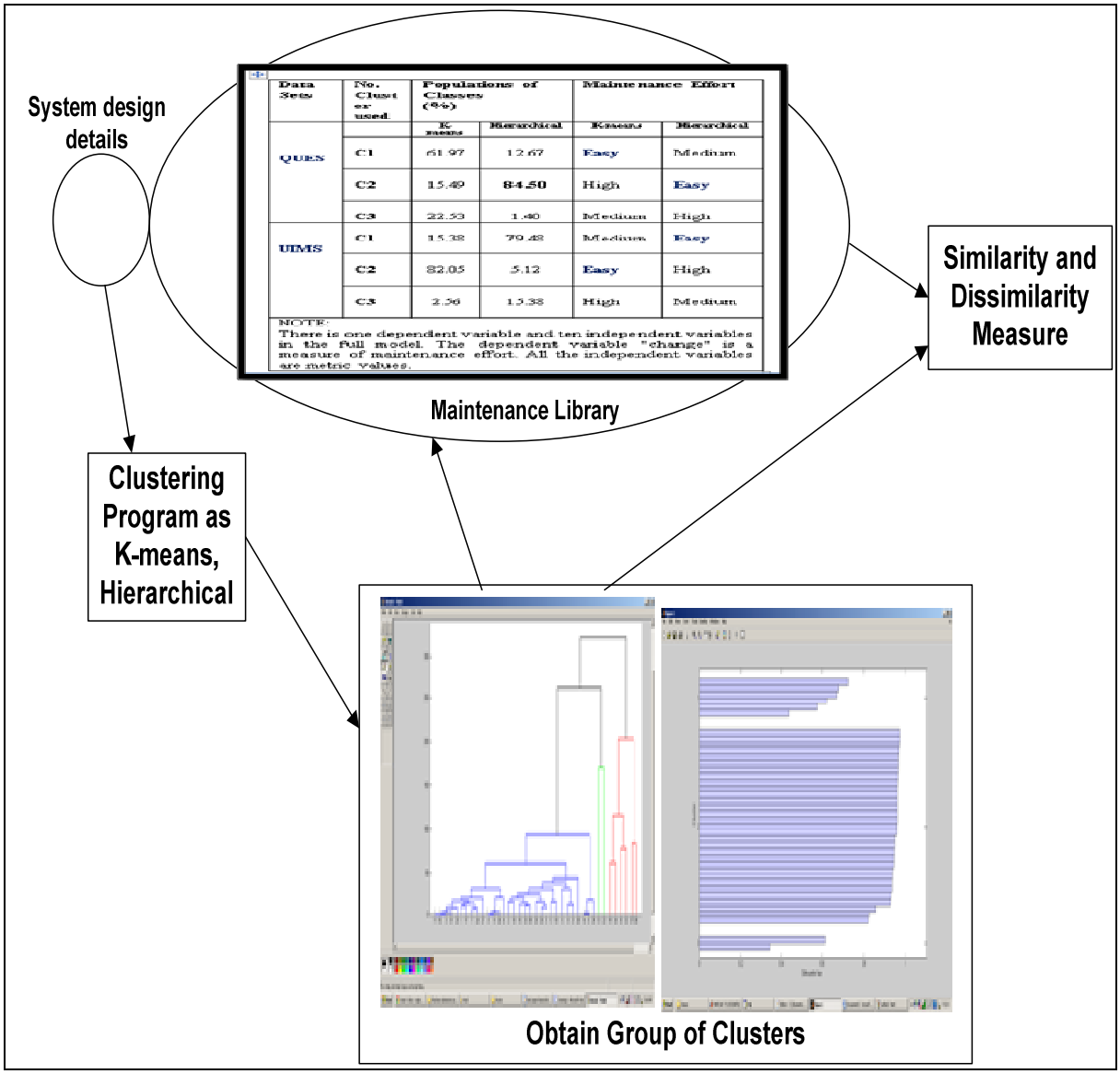


Figure 2: Proposed Steps for QUES and UIMS

Step 1: Take the system design's data of UIMS and QUES.

Step 2: In the second, we use the data mining clustering programme of K-means and Hierarchical on the Step1 in the MATLAB.

Step 3: Having apply clustering programmed we get the group of three clusters. The silhouettes plots and dendogram are used to show the classes belong to which clusters.

Step 4: It is observed that in the fourth step, we obtained the maintenance libraries that will be able to decide the cluster with Easy, Medium, and High conditions of maintainable classes of similarity based on object oriented metrics.

Table 1: Selected Object Oriented Metrics for UIMS and QUES

S/No.	Metrics	Uses
1	DIT	It provides for each class a measure of the inheritance levels from the objects hierarchy top.
2	NOC	It measures the number of the immediate descendants of the class.
3	MPC	It is a count to measure the complexity of message passing among classes.
4	RFC	It is count to the set of all methods that can potentially be invoked in response to all methods accessible within the class hierarchy.
5	LOCM	It measures if a class of the system has all its methods working together in order to achieve a single, well defined purpose.
6	DAC	It counts to measure the coupling complexity caused by ADTs.
7	WMC	It measures the overall complexity of class. It is a sum of all complexities of its methods.
8	NOM	It counts the number of local methods in a pre-defined class.
9	SIZE1	It counts the number of semicolon in pr-defined class.
10	SIZE2	It measures number of attributes and number of methods in class.
11	Change	The maintenance effect is measured by the number of lines changed per class.

.

9. EXPERIMENTAL RESULTS

An experimental test carried, the well known UIMS with 39 classes and 11 different attributes and QUES with 71 classes and 11 different attributes have been taken to measures the performances of these difference classes. We can make a silhouette plot and dendogram using the cluster indices output from K-means and Hierarchical clustering to get an associate plan of how independent the resulting clusters.

The silhouette plot shows how similar each point in one cluster is to points in the other clusters. This measure ranges from +1, showing points that are very away from the nearest clusters, through 0, showing points that are not decidedly in one cluster or another, to -1, showing points that may be assigned to the wrong cluster. The silhouette plot in figure 3 of QUES, we can see that most points in the second and third cluster have a silhouette value, less than 0.6. However, the first cluster contains some points with low silhouette values, and few points with negative values in second and third cluster, indicating that those two clusters are not well separated. Similarly form figure 4 of UIMS, we can see that most points in first, second and third clusters have silhouette value, less than 0.6.

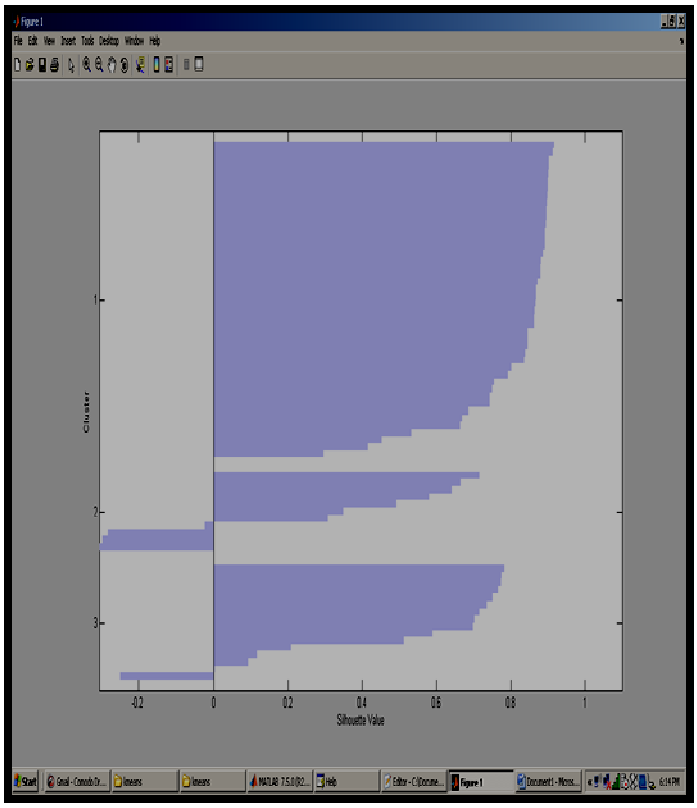


Figure 3: Silhouette Plots of QUES using K means

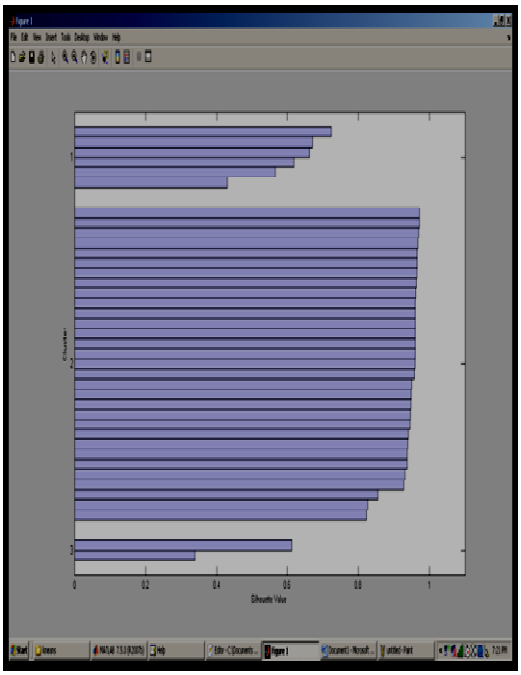
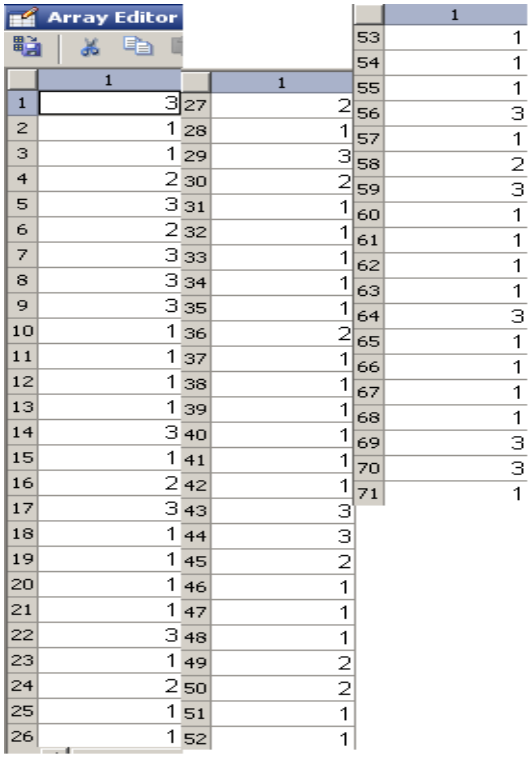


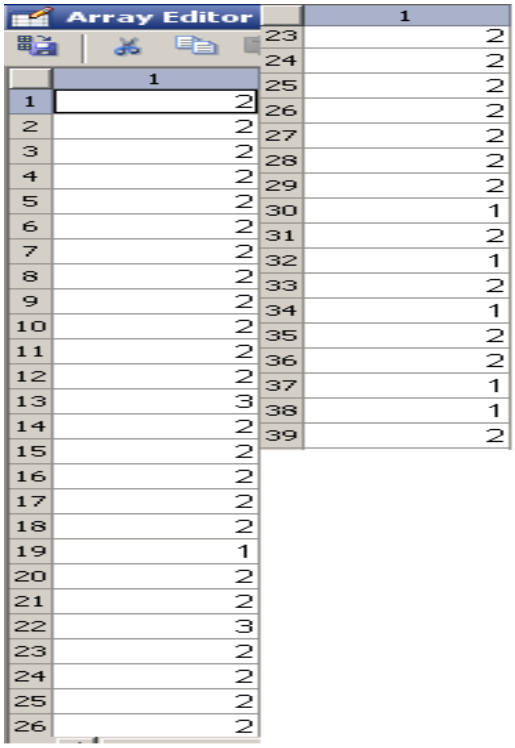
Figure 4: Silhouette Plots of UIMS using K means

The "idx3" in figure 5 and figure 6 of the test results on MATLAB using k-means for the QUES and UIMS. The figure 5 shows that forty four classes are included in cluster1, eleven classes are include in cluster2 and rests of sixteen classes are include in cluster3. According to these results cluster1 has the largest classes as compare to cluster2 and cluster3. This Cluster1 has the largest population of similarity based classes. Therefore it is easier to understand and maintain them. This cluster3 can be considers as "Easy" maintenance effort cluster.



	1		1		1
1	3	27	2	53	1
2	1	28	1	54	1
3	1	29	3	55	1
4	2	30	2	56	3
5	3	31	1	57	1
6	2	32	1	58	2
7	3	33	1	59	3
8	3	34	1	60	1
9	3	35	1	61	1
10	1	36	2	62	1
11	1	37	1	63	1
12	1	38	1	64	3
13	1	39	1	65	1
14	3	40	1	66	1
15	1	41	1	67	1
16	2	42	1	68	1
17	3	43	3	69	3
18	1	44	3	70	3
19	1	45	2	71	1
20	1	46	1		
21	1	47	1		
22	3	48	1		
23	1	49	2		
24	2	50	2		
25	1	51	1		
26	1	52	1		

Figure 5: Classes in 3-Clusters of QUES using K means



	1		1
1	2	23	2
2	2	24	2
3	2	25	2
4	2	26	2
5	2	27	2
6	2	28	2
7	2	29	2
8	2	30	1
9	2	31	2
10	2	32	1
11	2	33	2
12	2	34	1
13	3	35	2
14	2	36	2
15	2	37	1
16	2	38	1
17	2	39	2
18	2		
19	1		
20	2		
21	2		
22	3		
23	2		
24	2		
25	2		
26	2		

Figure 6: Classes in 3-Clusters of UIMS using K means

The above figure shows that six classes are included in cluster1, thirty two classes are include in cluster2 and rests of one class is includes in cluster3. According to these results cluster2 has the largest classes as compare to cluster1 and cluster1. This Cluster2 has the largest population of similarity based classes. Therefore it is easier to understand and maintain them. This cluster2 can be considers as "Easy" maintenance effort cluster.

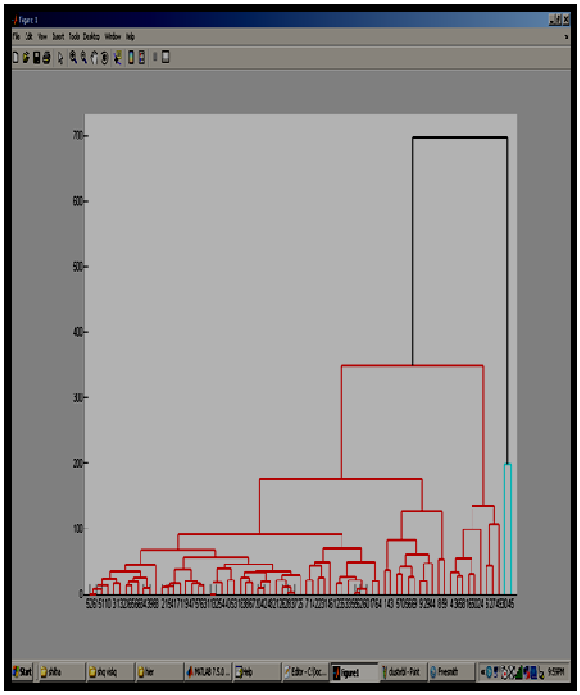


Figure 7: Dendrogram of QUES using Hierarchcal clustering

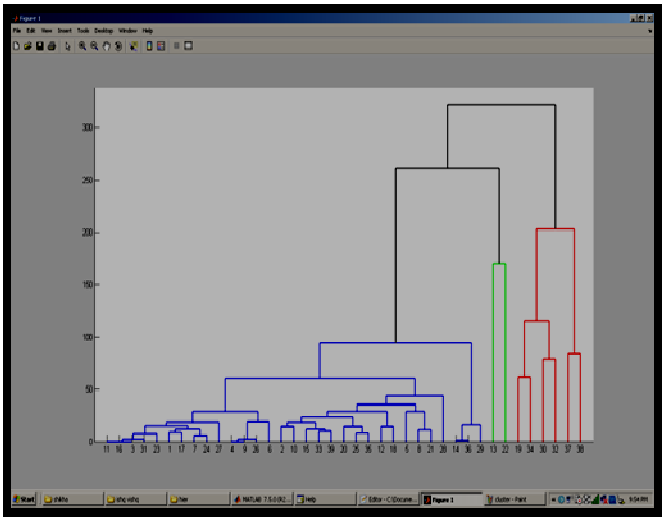
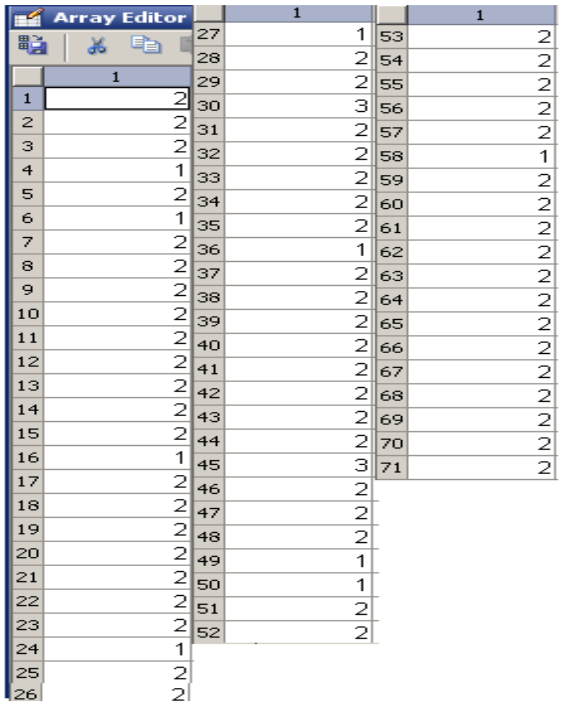


Figure 8: Dendrogram of UIMS using Hierarchical Clustering

The hierarchical, binary cluster tree created by the linkage function is most easily understood when viewed graphically. Statistics Toolbox includes the dendogram function that plots this hierarchical tree information as a graph, as in the figure 7 and figure 8. In the figure7, the numbers along the horizontal axis represent the indices of the objects in the original data set. The links between objects are represented as upside-down U-shaped lines. The height of the U indicates the distance between the objects. For example, the link representing the cluster containing objects 4 and 9 has a height of 0. The link representing the cluster that group's object 26 together with objects 4 and 9 has a height of 1. The height represents the distance linkage computes between objects. Similarly, we can conclude the information about grouping of objects in figure 8.



	1
1	2
2	2
3	2
4	1
5	2
6	1
7	2
8	2
9	2
10	2
11	2
12	2
13	2
14	2
15	2
16	1
17	2
18	2
19	2
20	2
21	2
22	2
23	2
24	1
25	2
26	2

	1
27	1
28	2
29	2
30	3
31	2
32	2
33	2
34	2
35	2
36	1
37	2
38	2
39	2
40	2
41	2
42	2
43	2
44	2
45	3
46	2
47	2
48	2
49	1
50	1
51	2
52	2

	1
53	2
54	2
55	2
56	2
57	2
58	1
59	2
60	2
61	2
62	2
63	2
64	2
65	2
66	2
67	2
68	2
69	2
70	2
71	2

Figure 9: Classes in 3-Clusters of QUES using Hierarchical Clustering

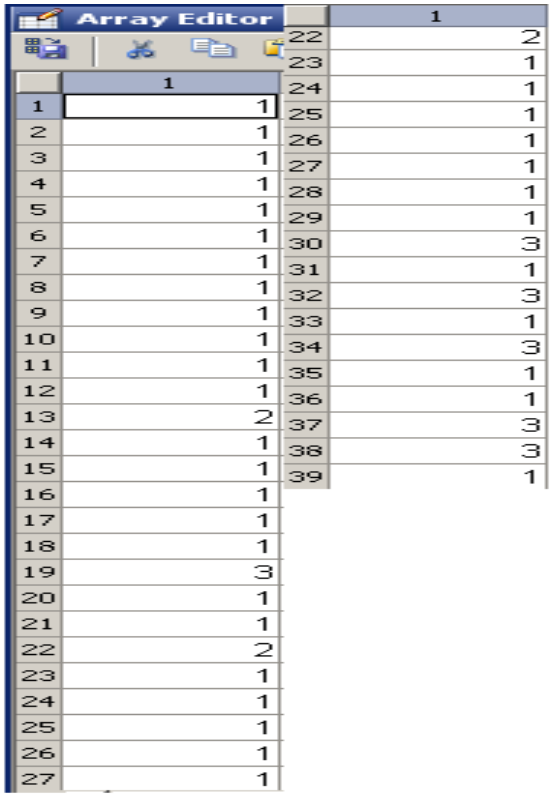


Figure 10: Classes in 3-Clusters of UIMS using Hierarchical Clustering

The "idx3" in figure 9 and figure 10 of the test results on MATLAB using hierarchical clustering for the QUES and UIMS. The figure 9 shows that nine classes are included in cluster1, sixty classes are include in cluster2 and rests of one class is include in cluster3. According to these results cluster2 has the largest classes as compare to cluster1 and cluster3. This Cluster1 has the largest population of similarity based classes. Therefore it is easier to understand and maintain them. This cluster2 can be considers as "Easy" maintenance effort cluster. Similarly, we can conclude the information about classes of each cluster in figure 10.

Table 2: Maintenance effort on QUES and UIMS

Data Sets	No. Cluster used	Populations of Classes (%)		Maintenance Effort	
		K-means	Hierarchical	K-means	Hierarchical
QUES	C1	61.97	12.67	**Easy**	Medium
	C2	15.49	84.50	High	**Easy**
	C3	22.53	1.40	Medium	High
UIMS	C1	15.38	79.48	Medium	**Easy**
	C2	82.05	5.12	**Easy**	High
	C3	2.56	15.38	High	Medium

The final an experiential results of QUES and UIMS in MATLAB are represented in table 2. According to this we can observed the following results on k-means and hierarchical clustering are given below:

10. RESULTS ANALYSIS

1. At times, there is an interpretive advantage to non-hierarchical clusters. For example, assume that if the data are divided into three clusters, units A and B will be in the same cluster. It may often make sense that if the data are divided into say two clusters, A and B will be in different clusters. This result is impossible with a hierarchical method.
2. K-means algorithms are used for large data set while hierarchical algorithms for small data set.
3. Hierarchical algorithms give better result when random dataset are used while k-means provide better result in case of ideal dataset.
4. As the number of records increase the performance of hierarchical algorithm goes decreasing and time for execution increased.
5. As the value of k becomes greater, the accuracy of hierarchical clustering becomes better while k-means have less quality (accuracy). We can see on table 2 the easy maintenance effort of k means algorithm is 82.05 on the other hand the easy maintenance effort of hierarchical clustering algorithm is 84.50. Thus according to this Hierarchical algorithm shows more quality as compared to k-mean algorithm.

6. There is one dependent variable as Change and ten independent variables as DIT, NOC, SIZE1, etc. are used to implement the data mining techniques on QUES and UIMS. The dependent variable "change" is a measure of maintenance effort. It is ascertained that the table 2 shows to decide the clusters with Easy, Medium, and High conditions. It provides the help to the software designers and maintainers to take the proper action at design level. It can also be used by software designers to modify the design of difficult to keep up classes at design level of software.

11. CONCLUSION AND FUTURE SCOPE

The main objective of this paper to development of a methodology based on the K-means and Hierarchical clustering data mining techniques have been implemented on UIMS and QUES class's data with set of selected metrics. This work is small step toward the design of maintainable object oriented software system. There is a future scope for more similar studies may be carried out with large data set of industrial object oriented system. It is planned, in future, to compare this methodology with other data mining clustering technique in terms of performance, noise and outliers and complexity, to increase the significant level of k clusters in K-means. This paper presents the identification of maintainable classes at the design phase of software development.

Thus, overall, it is concluded that it helps to the software designers and maintainers to take the appropriate action at design level. More similar studies must be carried out with large data type of set. It is planned, in future, to compare this methodology with other data mining clustering technique in terms of performance, noise and outliers and complexity, to increase the significant level of k clusters in K-means.

REFERENCES

[1] Pigoski T.M., Practical Software Maintenance: Best Practices for Managing your Software Investment, *Wiley Computer Publishing*, 1996.

[2] Sommerville, Software Engineering, 6th ed., *Harlow, Addison-Wesley*, 2001.

[3] Zhou, Y. and Leung, H., "Predicting Object-Oriented Software Maintainability using Multivariate Adaptive regression Splines", *The Journal of Systems and Software*, 80(2007), pp. 1349-1361.

[4] Anponellis, P., Antorious, D. and others, "A Data Mining Methodology for Evaluating Maintainability According to *ISO/IEC-9126 Software Engineering-Product", Internet.*

[5] Chidamber and Kemerer, C.F., "A Metrics suite for Object Oriented Design," *IEEE Transactions on Software Engineering",* Vol. 20, No.4, pp. 476-493, 1994.

[6] Kanellopoulos, Y. and Others, "K-attractors: A clustering Algorithm for Software Measurement Data Analysis," 19th *IEEE International Conference on Tools with A.I.*

[7] Malviya, A.K. and Dutta, M., "Measuring the Maintainability of Object Oriented Systems", *International Journal of Information & Computing Science*, Vol. 7 and No. 2, pp. 1-12.

[8] Muthana, S., Kontogiannis, k., Ponnambalam, K. and Stacey, B. "A Maintainable Model for Industrial Software Systems Using Design Level Metrics", *IEEE Software,* 2000.

[9] Zhong S., T.M. Khoshgoftar, and N. Seliya, "Analyzing Software Measurement Data with Clustering Techniques", *IEEE Intelligent System*, Vol. 19, No. 2, 2004, pp. 20-27.

[10] Kanellopoulos Y., Makris C. and Tjortjis C., "An Improved Methodology on information Distillation by Mining Program Source Code", to appear at *Elsevier Data & Knowledge Engineering*, 2006.

[11] Kanellopoulos Y., Dimopoulos T., Tjortjis C. and Makris C., "Mining Source Code Elements for Comprehending Object-Oriented Systems and Evaluating Their Maintainability" to appear at the *ACM SIGKDD Explorations* v8.1, Special Issue on Successful Real-World Data Mining Applications, June 2006.

[12] Kaur A., Sandhu P.S. and Brar A.S. (2009), "Early software fault prediction using real time defect data". In the proceedings of Second *international conference on Machine Vision*, Dubai, pp.242-245.

[13] Kaur A., Sandhu P. and Brar A. "An Empirical Approach for software fault prediction", *Fifth International Conference on industrial and Information System,* pp.261-265, 2010.

[14] Bandi R, Vijay K. Vaishnavi, Daniel E. Turk, "Predicting Maintenance Performance Using Object Oriented Design Complexity Metrics", *IEEE Transactions on Software,* Vol. 29, No. 1, January 2003, pp.77-87.

[15] Arisholm E., Lionel C. Briand, Audun Foyen, "Dynamic Coupling Measurement for Object-Oriented Software*", IEEE Transactions on Software Engineering,* Vol. 30, No. 8, August 2004, pp. 491-506.

[16] Sutherland J., "Business Objects in corporate Information Systems," *ACM Computing Survey,* vol. 27, 1995, pp 274-276.

[17] Tan, Y., Mookerjee, V. S. "Comparing Uniform and Flexible Policies for Software Maintenance and Replacement", *IEEE Transactions on Software Engineering*, vol. 31(3), March 2005, pp. 238-255.

[18] Rousidis, D. and Tjortjis, C., "Clustering Data Retrieved from Java Source Code to Support Software maintenance: A Case Study", Proceeding *IEEE ninth European Conference on Software Maintenance and Reengineering,* 2005, pp. 276-279.

[19] Xiao C., V. Tzerpos, "Software Clustering on Dynamic Dependencies", *Proceeding IEEE ninth European Conference on Software Maintenance and Reengineering*, 2005, pp. 124-133.

[20] ISO/IEC 9126-1, *Software Engineering –Product Quality International Standard*, Geneva 2001.

[21] Joaquín Pérez Ortega, Ma. Del Rocío Boone Rojas and María J. Somodevilla García, *Proceedings of the 2nd Workshop on Semantic Web and New Technologies (SemWeb09), Puebla, Mexico*, March 23-24, 2009. Vol-534, pages 83-96.

[22] Li. W., & Henry, S., "Object-Oriented Metrics that Predict Maintainability", *The Journal of Systems and Software,* Vol. 23, pp. 111-122, 1993.