# Analysis of Quality of Software Projects using Data Clustering Techniques

Pushphavathi T.P
Jain University, Bangalore, Research Industry and Incubation Centre, Dayanada Sagar Institute, Bangalore, India

Ramaswamy.V
Bapuji Institute of Technology, Davanagere, India

Suma.V
Research Industry and Incubation Centre, Dayanada Sagar Institute, Bangalore, India

## ABSTRACT
Ever since the evolution of software, prediction of desirable level of product quality which is measured at every phase of development is deemed a continuous and consistent effort. Quality is however viewed in various dimensions which also includes effective defect management. However, predicting the defect pattern within the empirical projects which directs the efficient management of defects in the future projects is always a challenging task in software industry. Clustering technique enables one to mine the defect associated information in order to achieve the above said challenge. Hence, there is dire need to develop software defect prediction model based on unsupervised learning which can help to predict the defect proneness of projects when defect labels for modules do not exist. This paper provides an empirical analysis of defects logged in several projects developed at various software industries using data mining and Fuzzy C-means (FCM) clustering approaches. This approach enables one to predict the characteristics of software projects early in the development phases. It further aids the project manager to plan and control the project activities which aims towards implementation of strategies for improved productivity and sustainability of the company in the industrial market.

## General Terms
Software Engineering, Data Mining, Fuzzy Logic

## Keywords
Software Engineering, Data Mining, Clustering, Fuzzy C means clustering, Metrics, Software Quality, Project Management.

## 1. INTRODUCTION
The population of software projects that are developed since the time of evolution of software to current day is massive indicating wide spectrum of application domains being developed using varied programming languages and in different operating environments. Therefore, development of a software product is an inherently difficult endeavour. Managing such a complex venture demands the skilful ability to estimate all project parameters accurately and reliably. Inaccurate estimations lead to unnecessary efforts in continuously modifying the project plan, preventable delays and in extreme cases the project failures. Quality management involves successful completion of a project under the limited availability of resources.

The journey of research associated with success and failure of software projects gained its popularity since four decades and the most widely cited definition of project success was a project that was completed on time, within budget and met customer requirements or agreed upon business objectives [1]. Further, several case studies involving ample of data were used to analyse the success and failure factors for software projects. However, success and failure of software projects are viewed in various perceptions by diverse project stakeholders indicating the high nature of vagueness involved in analysing accurately the rationales for project to be either successful or not[2].

Fuzzy Logic is an effective soft-computing technique to solve uncertainties due to imprecise inputs, in order to generate linguistic or quantitative outputs. The fuzzy logic model uses the fuzzy logic concepts introduced by Lofti A. Zadeh [5]. Fuzzy logic is a form of multi-valued logic derived from fuzzy set theory to deal with reasoning that is approximate rather than precise. A fuzzy set expresses the degree to which an element belongs to a set. The characteristic function of a fuzzy set is allowed to have values between 0 and 1, which denotes the degree of membership of an element in a given set. They have been employed in various real life applications. Fuzzy logic modelling techniques such as Fuzzy C-means clustering (FCM), fuzzy inferences have been shown to be a useful addition to the existing statistical and machine learning techniques used for modelling software development [6].

Fuzzy Logic has gained popularity in recent history as a sensible technique to achieve improved estimation accuracy of variables in any process. Fuzzy C-means clustering (FCM) is the one of the most recent contributions to the field of Artificial Intelligence (AI) and data clustering. Within project management, these variables range from software resource estimation to resource allocations for the completion of a software project [7].

This research therefore involved application of fuzzy logic on imprecise nature of software projects in determining the success or failure rates. However, the aim of this paper is to estimate the level of project success based on defect count as one of the factors that can transform a project to be either successful or failure. In order to achieve above said goal, this study directed us to apply fuzzy logic and data mining techniques for mining of defect related information from the wide band of projects which are developed at various software industries.

Although the idea of applying data mining techniques on software engineering data has existed since 1990s the idea has

especially attracted a large amount of interest recently within software engineering community [3].

Data Mining is defined as extracting or mining knowledge from large amounts of data. Data mining comprises of two modes of digging data namely predictive data mining and descriptive data mining. Both the modes of data mining are used to determine characteristics of association, classification, clustering, prediction and estimation within data sets. However, predictive method is opted in this research since the intension of the analysis is able to come out with an accurate predictive model for defect estimations to facilitate effective project management.

Organization of the paper is as follows. Section 2 specifies the related work in the domain of fuzzy logic, data mining and software engineering. Section 3 provides research methodology followed during this work. Section 4 provides overview FCM and K-means clustering algorithms. Section 5 indicates experimental results and summary of this part of research is briefed in Section 6.

## 2. RELATED WORK

Many researchers have contributed towards process and product quality improvement for achieving project success. Fuzzy Logic, Data Mining and Software Engineering domains aim towards the realization of aforementioned objective.

Authors in [7] investigated the application of fuzzy logic as a feasible technique for improved estimation accuracy to all the tasks within the project management knowledge areas to ensure higher software project success rates. Success of any project relies heavily on the initial estimation of all project parameters.

Authors in [8] express the use of fuzzy logic for project management may not be the same throughout the development life cycle. However, information available on different level of project development phase and desired precision suggest that it can be used differently depending on the current phase, although a single model can be used for consistency.

Authors in [2] propose a framework to estimate the software project success potential using association rule mining technique. It aims to explore the relationship between the risk dimensions and project outcome. Association rules that take risk dimensions as the condition and the project outcome as the result, project managers can estimate whether a new project will be successful or failure based on its risk factor values as early as possible.

Authors in [9] suggested data mining clustering and classification algorithms to predict the factors influencing for software project success. Authors in [10] showed the power of using a data mining approach in order to indicate the most important factors that lead to quality software development. The added value of visualization provided by different mining model viewers was crucial to the project managers who are not specialists in data mining. Authors in [11] presented different data mining clustering algorithms each algorithm has its unique features.

Authors in [13] found that K-means is first clustering technique also called as Hard C-means clustering [14] as compared to Fuzzy C-means clustering. In this technique each data point belongs to a cluster to a degree specified by a membership grade. As in K-means clustering, Fuzzy C-means clustering relies on minimizing a cost function (or objective function) of dissimilarity measure. It has been applied to a various fields including preprocessing for system models.

Authors in [15] explain practical benefits of clustering approach to the expert who must decide the labels. Instead of inspecting and labeling software projects one at a time, the expert can inspect and label a given cluster as a whole, in order to ease out the tediousness of the labeling task, which is compounded when projects are numerous. when actual labels for the software modules are available, clustering analysis can provide the expert with valuable feedback for improving expert-based labeling in future releases of the given software project or other software projects.

This research therefore focused us towards effectively managing the project using efficient fuzzy logic and data mining clustering techniques on software projects

## 3. RESEARCH METHODOLOGY

Several projects are collected from various software industries to carry out this research. The software industries are CMMI level 4 and 5 certified industries which comprises of both service-based and product-based software developing industries. The projects for investigation purpose are sampled out using random sampling technique from the population of projects developed since 2005 onwards in these industries. The project information is collected from repositories and data centers of each company. All the projects are non critical application oriented projects which further fall under the category of Enterprise Resource Planning and web applications. These projects are developed in Linux operating system and using object supporting languages such as Java, C++.

The empirical software project data considered for this investigation includes input attributes related to defect analysis and one output attribute which indicates whether the project is identified as successful or failure based on defect count. The data set is partitioned into two, where two-thirds of the data is deemed for training and one third for evaluation. The number of clusters into which the data set is to be partitioned is two clusters are defect or defect free. Because of the high number of dimensions in the problem, it depends on performance measures to evaluate the clustering techniques rather than on visual approaches.

The following are the parameters used in the clustering process:

a) Total project time in hours
b) Inspection time scheduled
c) Number of inspectors involved
d) Defect count estimation
e) Number of defects detected
f) Defects actually captured
g) Number of defects not captured
h) Defects due to bad fixes
i) Testing time scheduled
j) Number of testers

The similarity metric used to calculate the similarity between an input value and a cluster center is the Euclidean distance. Since most similarity metrics are sensitive to the large ranges of elements in the input values, each of the input variables must be normalized to lie within the unit interval [0, 1]. Each clustering algorithm is presented with the training data set, and as a result two clusters are produced. The data in the evaluation set is then tested against the found clusters and an analysis of the result is conducted. The following sections present the results of each clustering technique, followed by a comparison of the two techniques. MATLAB is used for these experiments.

## 4. ANALYSIS OF FCM AND K-MEANS CLUSTERING

This part of research aimed towards analyzing the effectiveness of clustering techniques such as FCM and K-means are used to evaluate the efficiency of the effective defect management.

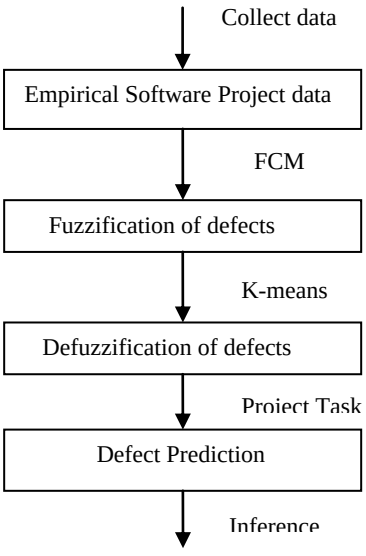


**Fig 1: Comparative analysis of FCM and K-means clustering**

Figure 1 depicts the mining of software project data for evaluation of software project management.Emperical data which is collected, pre processed by clustering techniques in order to analyze the software project tasks such as defect prediction. Due to vagueness in attributes selection at early stage development of software projects leads to wrong prediction of projects outcomes, also need to predict defect distribution pattern using FCM technique. However, K-means clustering is used to predict the impact of defect on projects.

Several methods that are established and a few are proven to be efficient in data mining and fuzzy logic. However, this paper focuses on analysing the efficient methods for estimating the quality of the projects using empirical data analysis. Clustering is applied upon the sampled empirical projects.

### 4.1 Fuzzy C Means Clustering

Fuzzy C-means clustering (FCM) has the basic idea of Hard C-means clustering (HCM). However, the data point in FCM belongs to a cluster having degree of membership grade, while in HCM every data point belongs to a certain cluster or fall in some outlier. Therefore, FCM employs fuzzy partitioning such that a given data point can belong to several groups with the degree of membership specified by membership grades between 0 and 1. However, FCM uses an objective function that is to be minimized while doing partition the data set. The membership matrix U is allowed to have elements with values between 0 and 1. However, the summation of degrees of belongingness of a data point to all clusters is always equal to unity. The algorithm is made up of following steps [5]:

1. Initialize U = U[ij] matrix U[0]
2. K-step Calculate the center vectors C[k] = [Cj] with U(k) where

$$c = \frac{\sum_{i=1}^{N} u_{ij}^{m} . x_i}{\sum_{i=1}^{N} u_{ij}^{m}}$$

3. Update U(k) and U (k+1)

$$u_{ij} = \frac{1}{\sum_{k=1}^{C} \left( \frac{\left\| x_i - c_j \right\|}{\left\| x_i - c_k \right\|} \right)^{\frac{2}{m-1}}}$$

4. If ||U (k+1) – U (k) || < € then Stop, Otherwise return to the second step.

FCM allows for data points to have different degrees of membership to each of the clusters, thus eliminating the effect of hard membership introduced by K-means clustering. This approach employs fuzzy measures as the basis for membership matrix calculation and for cluster centers identification. As it is the case in K-means clustering.

FCM clustering and its different threshold (α- cut) values are employed to classify the projects into different groups based on severity of defects, in order to analyse the impact of defect

in projects and also helpful to predict for pattern of defect distribution. This awareness of accurate defect distribution pattern enables the project manager to accurately estimate the defect management strategies.

## 4.2 K-means Clustering

The grouping is done by minimizing the sum of squares of distances between data and the corresponding cluster centroid.It can be viewed as a greedy algorithm for partitioning the n samples into k clusters so as to minimize the sum of the squared distance to the cluster centers.

The K-means clustering is an algorithm based on finding data clusters in a data set such that an objective function of dissimilarity or distance measure is minimized [13]. In most cases this dissimilarity measure is chosen as the Euclidean distance. The algorithm is made up of following steps [16].The objective function of K-means clustering is shown in equation 1.

$$J = \sum_{j=1}^{k} \sum_{i=1}^{k} \left\| X_i^{(j)} - Cj \right\|^2 \quad (1)$$

Where $\left\| X_i^{(j)} - Cj \right\|^2$ is a chosen distance measure between a data point $X_i^{(j)}$ and the cluster centre $Cj$ is an indicator of the distance of the n data points from their respective cluster centres.

Step 1: Initialize the cluster center by randomly selecting from the data points.

Step 2: Assign each object to the group that has the closest centroid.

Step 3: When all objects have been assigned, recalculate the positions of the K centroids. Stop if either it is below a certain tolerance value or its improvement over previous iteration is below a certain threshold.

Step 4: Repeat Steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated.

The performance of the K-means algorithm depends on the initial positions of the cluster centers, thus it is desirable to run the algorithm several times, each with a different set of initial cluster centers.

K-means clustering is applied on empirical projects based on the assumptions made as explained in section 3. Evaluating the algorithm is realized by testing the accuracy of the evaluation set. The cluster centers are determined randomly according to the k-means algorithm. The evaluation of project defect count values are assigned to their respective clusters according to the distance between each defect value and each of the cluster centers. An error measure is then calculated and the Root Mean Square Error (RMSE) is used for this purpose.An accuracy measure is calculated as the percentage of correctly classified projects.

# 5. EXPERIMENTAL RESULTS

To predict the results, we have used confusion matrix as shown in Table 1. The confusion matrix has four categories: True positives (TP) are the projects correctly classified as defect. False positives (FP) refer to defect-free projects incorrectly labeled as defect. True negatives (TN) are the defect-free projects correctly labeled as such. False negatives (FN) refer to defect projects incorrectly classified as defect-free projects indicating the level of inaccurate defect distribution pattern.

**Table 1. A Confusion Matrix of Prediction Outputs**

| | | **Predicted** | |
|---|---|---|---|
| | | Defect | Defect free |
| **Actual** | Defect | TP | FN |
| | Defect free | FP | TN |

The following evaluation measures are being used to find the results:

• Mean Absolute Error (MAE) is a quantity used to measure actual outcomes.

• Root Mean Square Error (RMSE) which measures the difference between predict and corresponding observed values are each squared and then averaged over the sample

• Accuracy: It indicates proximity of measurement results to the true value, precision to the repeatability or reproducibility of the measurement. The accuracy is the proportion of true results (both true positives and true negatives) in the population. Lower values of MAE and RMSE will give better results.

FCM starts by assigning random values to the membership matrix U, thus several runs have to be conducted to have higher probability of getting good performance. However, the results showed either no variation in performance or accuracy when the algorithm was run for several times. For testing the results, every input value in the evaluation data set is assigned to one of the clusters with a certain degree of membership as done in the training set. However, because the output values we have are crisp values (either 1 or 0), the evaluation set degrees of membership are defuzzified to be tested against the actual outputs. The same performance measures applied in K-means clustering will be used. However, only the effect of the cluster controlling parameter or weighting exponent m is analyzed, since the effect of random initial membership grades has insignificant effect on the final cluster centers.

Table 2 lists the results of the tests with the effect of varying the weighting exponent m. It is observed that very low or very high values for m reduce the accuracy. Additionally, high values tend to increase the time taken by the algorithm to find the clusters. A value of 2.5 will be tolerable, since it has better accuracy and requires with a reduction of number of iterations against the weighting factor.

FCM technique showed no improvement over the K-means clustering. Both showed close accuracy. Further, FCM was found to be slower than K-means because of many fuzzy calculations.

**Table 2. FCM clustering performance results**

| Performance | Weighting exponent m | | | | | |
|---|---|---|---|---|---|---|
| | 1 | 1.5 | 2 | 2.5 | 3 | 5 |
| No.of iterations | 15 | 18 | 25 | 27 | 30 | 35 |
| RMSE | 0.40 | 0.42 | 0.45 | 0.44 | 0.46 | 0.46 |
| Accuracy | 75% | 77% | 78% | 79% | 76% | 75% |

As mentioned in the previous section, K-means clustering works on finding the cluster centers by trying to minimize a objective function, J. It alternates between updating the membership matrix and updating the cluster centers using methods explained above, until no further improvement in the objective function is noticed. Since the algorithm initializes the cluster centers randomly, its performance is affected by those initial cluster centers. Hence, it is suggested for conduction several runs of the algorithm to have better results.

| Performance | Test iterations | | | | | |
|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 |
| No.of iterations | 8 | 10 | 15 | 13 | 7 | 8 |
| RMSE | 0.39 | 0.42 | 0.46 | 0.43 | 0.35 | 0.39 |
| Accuracy | 75% | 76% | 78% | 82% | 70% | 75% |

**Table 3. K-means clustering performance results**

As seen from the results, the best case achieved 82% accuracy and an RMSE of 0.43.This relatively reasonable performance is related to the high dimensionality of the projects.

**Table 4. Comparison of Clustering Performance**

| Performance | RMSE | Accuracy | Time(sec) |
|---|---|---|---|
| FCM | 0.45 | 79% | 2.7 |
| K-means | 0.43 | 82% | 0.9 |

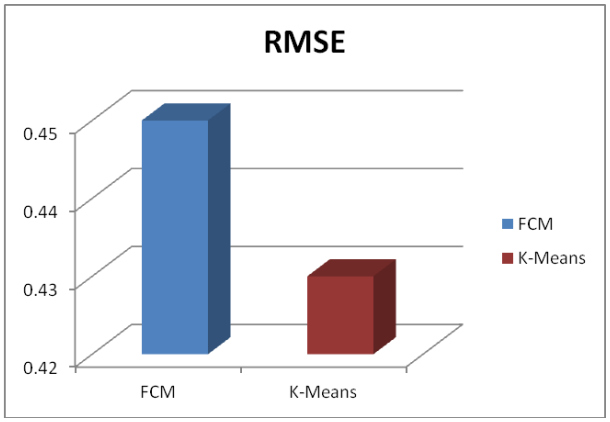


**Fig 2: Performance results of RMSE**

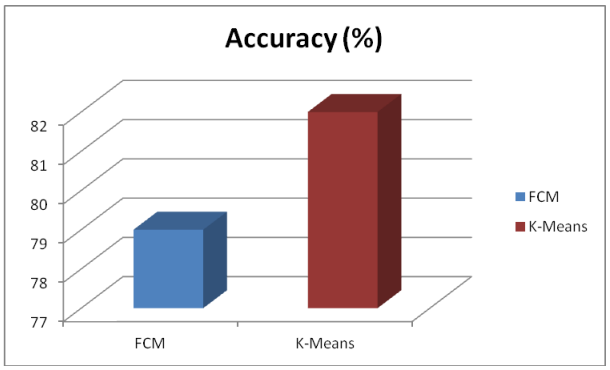


**Fig 3: Performance results of Accuracy**

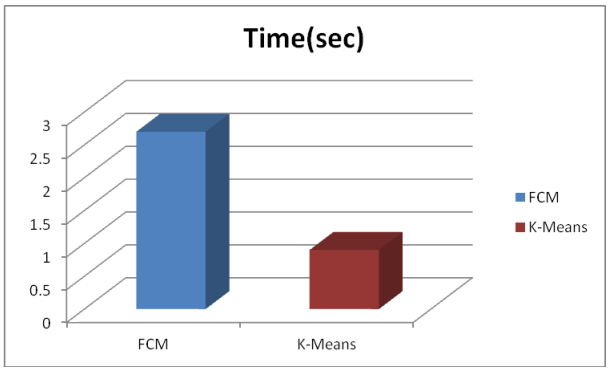


**Fig 4: Performance results of Time**

From the Table 4, it is quite apparent that K means is better when compared to FCM in terms of accuracy and time. As an instance, value of m is 2.5 in the test runs, accuracy of FCM turns out is 79% and K-means with RMSE value is 0.43 turn out to be 82%.The results of RMSE in figure 2, result outcome of Accuracy in figure 3 and figure 4 of Time depicts the same. Further, it is concluded that with increase in attribute list for clustering the projects, K –means has a better accuracy than with FCM. Additionally the processing time decreases in FCM with increase in attribute list when compared to k-means.

## 6. CONCLUSION

Since software has gained its strong impact on all applications, it is imperative to develop projects with accurate estimation to realize high quality production. Since quality has various dimensions, one of the most significant dimensions of quality is being defect free. Thus, it becomes one of the vital activities of the project manager to accurately predict the defect distribution pattern based on empirical analysis.

Fuzzy clustering and data mining are two significant approaches through which information can be elicited from the huge and varied population of software projects for accurate prediction of defect distribution pattern and henceforth the accurate estimation of defects in the subsequent projects.

An empirical analysis is conducted on various projects that were developed across several software industries for accurate prediction of defect distribution pattern. This paper presents a comparative analysis of two significant techniques applied upon the empirical software projects for comprehending their suitability in terms of accuracy and time constraints. The experimental results indicate that with increased list of attributes that are deemed for the prediction purpose specifies that K-means is more accurate and fast in processing the information when compared to FCM. This knowledge of right choice of approach for effective defect distribution pattern enables the project managers for making accurate estimation of defect count in their subsequent projects. It further leads to effective defect management which in turn enhances the software quality.

## 7. ACKNOWLEDGMENTS

The authors would like to sincerely acknowledge all the software personnel who helped us to carry out this research their industries as per the Non Disclosure agreement policies of the respective companies.

## 8. REFERENCES

[1] Keider SP, "Why projects fail", Datamation 20, 1974, pp.53–55

[2] Xiao Hong Shan, GuoRui Jiang, Tiyun Huang, A framework of estimating software project success potential based on association rule mining, 978-1-4244-4639-1/09/$25.00 ©2009 IEEE.

[3] Ahmed E. Hassan , Ahmed E. Hassan, Mining Software Engineering Data ,ICSE '10, May 2-8 2010, Cape Town, South Africa Copyright 2010 ACM 978-1-60558-719-6/10/05 ...$10.00

[4] Lovre Hribar,Denis Duka, Software component quality prediction using KNN and Fuzzy logic,MIPRO 2010,May 24-28,2010,Opatija,Croatia.

[5] Zadeh, L.A., Fuzzy sets, Info and Control, 8,338-353, 1965

[6] J.N.V.R.Swarup Kumar,T.Govinda Rao,Y.Naga Babu S.Chaitanya,K.Subrahmanyam, A Novel Method for Software Effort Estimation Using Inverse Regression as firing Interval in fuzzy logic, 978-1-4244-8679-3/11/$26.00 ©2011 IEEE.

[7] Imran Siwani and Miriam Capretz, APPLICATION OF FUZZY LOGIC FOR IMPROVED SOFTWARE PROJECT MANAGEMENT ESTIMATIONS, 2004 0-7803-8253-6/04/$17.00 0 2004 IEEE.

[8] Andrew R. Gray and Stephen G. MacDonell, Fuzzy Logic for Software Metric Models throughout the Development Life-Cycle, 0-7803-5211 - 4/99/$10.000 1999 IEEE.

[9] Anand Prasad, Juzer Arsiwala, Praval Pratap Singh.Estimation and Improving the Probability of Success of a Software Project by Analyzing the Factors Involved Using Data Mining. 978-1-4244-6936-9/10/$26 2010 IEEE.

[10] A.H.Yousef, A.Gamal, A.Warda, M.Mahmoud,Software Projects Success Factors Identification using Data Mining,1-4244-0272-7/06/$20.00 ©2006 IEEE

[11] WEKA Data Mining Software in Java: http://www.cs.waikato.ac.nz/ml/weka/

[12] Manoel Mendonca , Nancy L. Sunderhaft, Mining Software Engineering Data: A Survey, A DACS State-of-the-Art Report, Mining Software Engineering Data: A Survey

[13] J. A. Hartigan and M. A. Wong, A k-means clustering algorithm, Applied Statistics, 28:100-- 108, 1979.

[14] Jang, J.-S. R., Sun, C.-T., Mizutani, E., Neuro- Fuzzy and Soft Computing – A Computational Approach to Learning and Machine Intelligence, Prentice Hall.

[15] Shi Zhong, Taghi M. Khoshgoftaar, and Naeem Seliya, Analyzing Software Measurement Data with Clustering Techniques, Florida Atlantic University. 1094-7167/04/$20.00 © 2004 IEEE, IEEE INTELLIGENT SYSTEMS.

[16] Deepak Gupta, Vinay Kumar Goel, Harish Mittal, Software Quality Analysis of Unlabeled Program Modules with Fuzzy C-means Clustering Technique. IJMRS's International Journal of Engineering Sciences, Vol. 01, Issue 02, June 2012, ISSN: 2277-9698.