Comparative Study and Performance Analysis of Clustering Algorithms

Kamalpreet Kaur Jassar
Research Scholar
BBSBEC, Deptt. of CSE,
Fatehgarh Sahib, Punjab, India

Kanwalvir Singh Dhindsa, PhD
Associate Professor
BBSBEC, Deptt. of CSE,
Fatehgarh Sahib, Punjab, India

ABSTRACT
Spatial clustering is a process of grouping a set of spatial objects into groups, these groups are called clusters. Objects within a one cluster show a high degree of similarity, whereas the objects in another cluster are as much non-similar as possible. Clustering is a very well known technique of data mining which is mostly used method of analyzing and describing the data. It is one of the techniques to deal with the large geographical datasets. Clustering is the mostly used method of data mining. SOM and k-means are two classical methods for clustering. This paper illustrates the approach of clustering: Kohonen SOM and K-Means have been discussed and compared using different parameters on same dataset. After comparing these methods effectively, results of the experiments suggest that Self-Organizing Maps (SOM) is more robust to outlier than the k-means method. In this paper, experiments have been performed to compare the performances of clustering algorithms.

Keywords
Spatial Clustering, Clustering Algorithms, SOM, K-Means, PCA, etc.

1. INTRODUCTION
A self-organizing map (SOM) is a kind of artificial neural network that is trained using unsupervised learning to produce a low dimensional typically two dimensional as output. It is discretized representation of the input space of the training samples, called a map. Self-organizing maps are different than other artificial neural networks in the sense that they use a neighbourhood function to preserve the topological properties of the input space. The main set back of this technique, however, is that the number of output nodes is predefined and only the adjacent nodes are taken as neighbourhood [8]. SOM is a clustering method because it organizes the data in clusters (cells of map) such as the instances in the same cell are similar, and the instances in different cells are different. In this point of view, SOM gives comparable results to state-of-the art clustering algorithm such as K-Means [11]. SOM is also considered as data visualization technique because it allows us to visualize data in a low dimensional representation space (basically in 2D).

The Kohonen SOM algorithm is a very powerful tool to analyze the data [21]. It was originally designed to model organized connections between biological neural networks. It was also immediately considered as effective algorithm to realize vectorial quantization as well as pertinent classification with some properties for visualization [20]. Self-Organizing Maps (SOMs) have been used in GIScience both for clustering georeferenced data and also for the spatialization of various non-geographic datasets. The original SOM does not take into account the particular role that geographic location has in most problems involving the clustering of geo-referenced data. In the original SOM algorithm, all variables are treated equally. When clustering geo-referenced data, spatial location is particularly important, since objects that are geographically far away should not be clustered together, even if they are similar in all other aspects. Although the term "Self-Organizing Map" could be applied to number of different approaches, we shall use it as a synonym of Kohonen's Self Organizing Map, or SOM for short, also known as Kohonen Neural Networks. These maps are primarily used as visualization and analysis tools for high dimensional data, but they have been used for clustering, dimensionality reduction, classification, sampling, vector quantization and data-mining.

The basic idea of a SOM is to map the data patterns onto n-dimensional grid of neurons or units. That grid forms the output space, as opposed to the input space and where the data patterns are. This mapping tries to preserve topological relations, i.e., patterns that are close in the input space will be mapped to units that are close in the output space, and vice-versa. So as to allow an easy visualization, the output space is usually 1 or 2 D.

2. RELATED WORK
Aneetha and Bose [8] proposed a modified Self Organizing Map algorithm which initially starts with null network and grows with the original data space as initial weight vector, updating neighbourhood rules and learning rate dynamically in order to overcome the fixed architecture and random weight vector assignment of simple SOM. New nodes are created using distance threshold parameter and their neighbourhood is identified using connection strength and its learning rule and the weight vector updation is carried out for neighbourhood nodes. The k-means clustering algorithm is employed for grouping similar nodes of Modified SOM into k clusters using similar measures.

A comparative study between SOM algorithm and k-means algorithms is given by Toor and Singh [11] and they have found that Kohonen SOM gives the better performance as compare to K-Means with minimum error rate or high accuracy, minimum computation time on same data set and parameters.

Different data clustering algorithms has been studied and compared by Abbas [1]. These are compared according to the factors like size of dataset, type of dataset, number of clusters and tool used. The algorithms considered for investigation are k-means algorithm, self organizing map algorithm, hierarchical clustering algorithm and expectation maximization algorithm. Conclusions extracted from comparative study of these algorithms belong to the performance, quality and accuracy of algorithms.

Dhingra et. al. [20] suggests that Kohonen SOM gives better performance as compare to K-means. The performance of these two algorithms is measured on the basis of different parameters. Finally, it can be stated that when tested in a completely equal working conditions, Kohonen SOM can be considered as an appropriate clustering algorithm for high dimensional data set.

3. EXPERIMENTAL SETUP

3.1 Data Mining Tool -TANAGRA

TANAGRA is free DATA MINING software for academic and research purposes [25]. It proposes several data mining methods from exploratory data analysis, statistical learning, machine learning and databases area. TANAGRA is more powerful, it contains some supervised learning but also other paradigms such as clustering, factorial analysis, parametric and nonparametric statistics, association rule, feature selection and construction algorithms.

TANAGRA is an “open source project” as every researcher can access to the source code, and add his own algorithms.

The main purpose of Tanagra project is to give researchers and students an easy-to-use data mining software, conforming to the present norms of the software development in this domain (especially in the design of its GUI and the way to use it), and allowing to analyze either real or synthetic data. The second purpose of TANAGRA is to propose to researchers an architecture allowing them to easily add their own data mining methods, to compare their performances [27]. TANAGRA acts more as an experimental platform in order to let them go to the essential of their work, dispensing them to deal with the unpleasant part in the programmation of this kind of tools: the data management. The third and last purpose, in direction of novice developers, consists in diffusing a possible methodology for building this kind of software. They should take advantage of free access to source code, to look how this sort of software is built, the problems to avoid, the main steps of the project, and which tools and code libraries to use for. In this way, Tanagra can be considered as a pedagogical tool for learning programming techniques.

3.2 Dataset Description

For performing the comparison analysis we need the past project datasets. In this research the researcher considered following data set which is available on data repositories. These repositories are very helpful for the researchers. This data has been directly applied in the data mining tools and predict the result.

Source: KDD cup 99: Shuttle Data set.

<http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>

Dataset Information: There are approximately 9 attributes all of which are numeric, 58000 instances and 1 target class. These instances are considered or testing. Some of them will be used as an input attributes (continue) and other are used as an output attributes. This data set was generated originally to extract comprehensible rules for determining the conditions under which an autolanding would be preferable to manual control of a spacecraft. Some of the attributes from the dataset are given in the Table 1. Here Var1 to Var9 represents variables which have numeric values. The last column is the class with the following 7 levels: Rad.Flow, Fpv.Close, Fpv.Open, High, Bypass, Bpv.Close, Bpv.Open.

Table 1. Dataset description

Var 1	**Var 2**	**Var 3**	**Var 4**	**Var 5**	**Var 6**	**Var 7**	**Var 8**	**Var 9**	**Class**
50	21	77	0	28	0	27	48	22	Fpv close
55	0	92	0	0	26	36	92	56	high
53	0	82	0	52	-5	29	30	2	Rad flow
37	0	76	0	28	18	40	48	8	Rad flow
37	0	79	0	34	-26	43	46	2	Rad flow
85	0	88	-4	6	1	3	83	80	By pass
56	0	81	0	-4	11	25	86	62	high
55	-1	95	-3	54	-4	40	41	2	Red flow
53	8	77	0	28	0	23	48	24	high
37	0	101	-7	28	0	64	73	8	Rad flow
37	0	78	-2	12	0	42	65	24	Rad flow
45	0	84	0	46	20	38	37	0	Rad flow
38	2	77	0	38	7	39	38	0	Rad flow
37	0	78	0	-2	5	41	81	40	Rad flow
41	0	100	0	38	-8	59	61	2	Rad flow
41	0	89	1	38	-16	48	50	2	Rad flow
47	0	85	-2	46	-4	38	39	0	Rad flow

4 METHODOLOGY

4.1 Kohonen SOM’s Approach

In Kohonen SOM (Kohonen, 1982) discussed spatially continuous input space in which our input vectors live. The aim is to map from this to a low dimensional spatially discrete output space, the topology of which is formed by arranging a set of neurons in a grid. SOM provides such a nonlinear transformation called a feature map [11].

The stages of the SOM algorithm can be summarised as follows:

1. Initialization – Choose random values for the initial weight vectors wi.
2. Sampling – Draw a sample training input vector x from the input space.
3. Matching – Find the winning neuron I(x) with weight vector closest to input vector.
4. Updating – Apply the weight update equation

   $$\mathbf{|Wi\text{-}X| \leq |WK\text{-}X| \quad \forall k}$$

5. Continuation – keep returning to step 2 until the feature map stops changing.

Given the winning node i, the weight update is

$\omega k(new) = \omega k(old) + \Delta\omega k(n)$

where $\Delta\omega k(n)$ represesnts the change in weight**.**

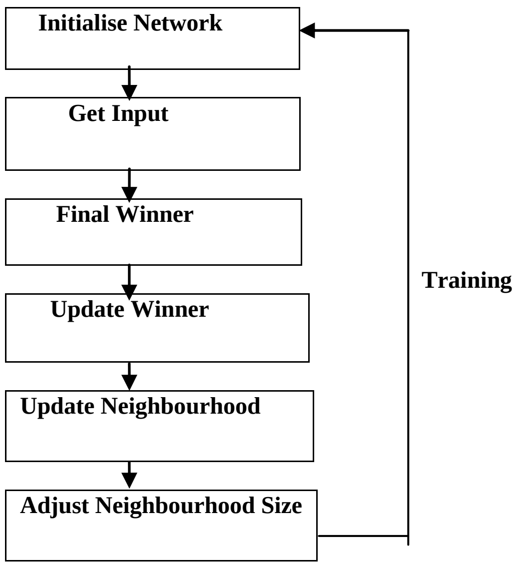


Fig. 1 Working of SOM [11]

4.2 K-Mean's Approach

The K-means (Mac Queen, 1967) is among the commonly used partitioning based clustering method that tries to find a specified number of clusters (k). For any given set of numeric dataset **X** and an integer number k, the K-means algorithm searches for a partition of X into k clusters that minimizes the within groups sum of squared errors. The K-means algorithm starts by initializing the k cluster centers with **n** samples. The input data points are then allocated to one of the existing clusters according to the square of the Euclidean distance from the clusters, choosing the closest. The mean (centroid **C**) of each cluster is then computed so as to update the cluster center. This update occurs as a result of the change in the membership of each cluster. The processes of re-assigning the input vectors and the update of the cluster centers is repeated until no more change in the value of any of the cluster centers.

$$J(X,C)=\sum_{t=0}^{n} \min \left(\|Xj-\mu i\|^2 \right) \text{ where, } \mu i \varepsilon C)$$

The steps of the K-means algorithm are written below:

1. Initialization- Choose randomly K input vectors (data points) to initialize the clusters.
2. Searching- Find the cluster center that is closest for each input vector and assign that input vector to the corresponding cluster.
3. Updation- Update the cluster centers in each cluster using the mean (centroid) of the input vectors assigned to that cluster.
4. Continuation - Repeat steps 2 and 3 until no more change in the value of the means.

5 RESULTS AND DISCUSSION

5.1 Implementation

A new file has been created and imported in Tanagra. The imported file should be in XLS format. Another way to import a file is directly load a XLS file from the system.

Table 2. Input Information

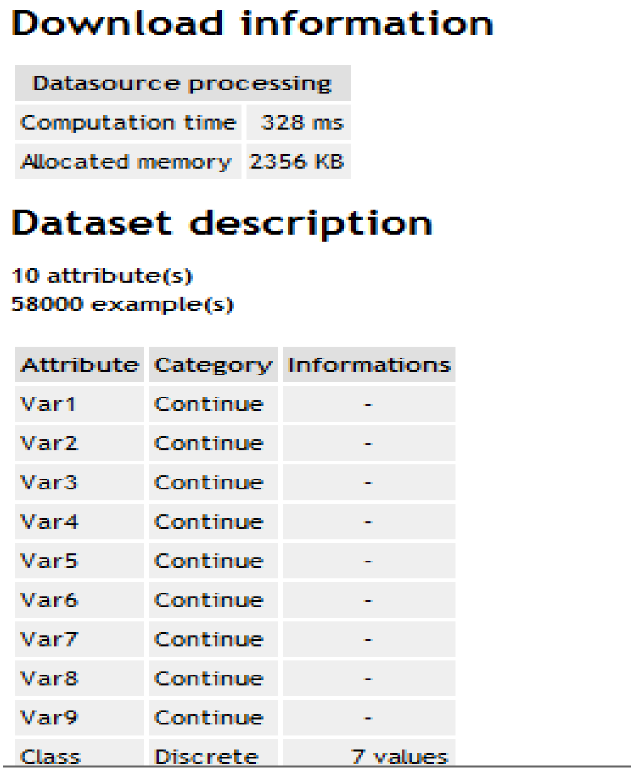
S.No.	Clustering Method	Input	Target
1	Kohonen SOM	One or More Continuous	None
2	K-Means	One or More Continuous	None

Above Table 2 shows the input information for dataset that has been imported. Both algorithms accept continuous attributes instead of discrete attributes as input parameter. There can be one or more the one input attributes.

Table 3. Datasource Processing

S.No.	Parameters	Values
1	Computational Time	296ms
2	Allocated Memory	2356kb

Above Table 3 gives complete information about dataset that has been imported. Here the total computational time of data processing is 296 ms. The total space required to store that data input is its memory which is said to be allocated memory i.e. 2356kb. Below Fig.2 illustrates the same information.



Download information

Datasource processing	
Computation time	328 ms
Allocated memory	2356 KB

Dataset description

10 attribute(s)
58000 example(s)

Attribute	Category	Informations
Var1	Continue	-
Var2	Continue	-
Var3	Continue	-
Var4	Continue	-
Var5	Continue	-
Var6	Continue	-
Var7	Continue	-
Var8	Continue	-
Var9	Continue	-
Class	Discrete	7 values

Fig. 2 Dataset Description

Figure 2 shows the dataset description that there are 10 attributes and 58000 examples in the dataset. Category and information about attribute is given so that one can choose attributes according to requirement. Here for implementing clustering algorithms, continuous attributes have been used as input.

SOM parameters	
Columns	2
Rows	4
Distance normalization	variance
Learning rate	0.05
Seed random generator	Standard

Fig. 3 SOM Parameters

Above Fig. 3 shows the initial setting for implementing SOM algorithm. Total 8 clusters has been generated by taking 4 rows and 2 columns.

MAP Topology

	1	2
1	17132	2436
2	5359	6132
3	6995	18974
4	949	23

MAP Quality

Ratio explained	0.5544

Cluster centroids

Attribute	Cluster n°1	Cluster n°2	Cluster n°3	Cluster n°4	Cluster n°5	Cluster n°6	Cluster n°7	Cluster n°8
Var1	39.485349	80.203612	38.921627	49.628017	54.424303	49.199009	102.977871	50.260870
Var2	-1.341758	0.385057	-6.323754	0.046804	0.253896	0.063455	54.363541	-2.086957
Var3	80.750934	84.663383	101.273745	79.483040	97.149964	81.679667	104.685985	76.826087
Var4	0.182933	-0.067734	1.184176	0.312622	0.107362	0.006430	15.630137	-516.391304
Var5	34.462059	-21.722496	29.700690	-1.429550	48.763117	47.816907	70.533193	30.086957
Var6	1.903864	-30.899425	-0.377682	-0.082192	0.434024	1.558765	119.661749	-335.391304
Var7	41.332127	4.422414	62.250793	29.796477	42.652609	32.446769	1.641728	26.391304

Fig. 4 Implemented Kohonen SOM algorithm and map topology is generated with 8 clusters. Map Quality is 0.5544

Kohonen SOM algorithm is applied by choosing the algorithm from clustering tab. Map topology has been formed with 4 rows and 2 columns and generate 8 clusters. Map Quality as shown above is 0.5544, which represents error rate of the implemented algorithm. Cluster centroids are also formed for each cluster.

Eigen values

Matrix trace	9.000000
Average	1.000000

Axis	Eigen value	Difference	Proportion (%)	Histogram	Cumulative (%)
1	3.118810	1.513828	34.65 %		34.65 %
2	1.604982	0.295464	17.83 %		52.49 %
3	1.309518	0.255022	14.55 %		67.04 %
4	1.054496	0.067289	11.72 %		78.75 %
5	0.987207	0.063926	10.97 %		89.72 %
6	0.923281	0.922551	10.26 %		99.98 %
7	0.000730	0.000109	0.01 %		99.99 %
8	0.000621	0.000266	0.01 %		100.00 %
9	0.000354	-	0.00 %		100.00 %
Tot.	9.000000	-	-	-	-

Fig. 5 Implemented PCA on Kohonen SOM and evaluate Eigen Values

In above Fig. 5 a visualization tool named PCA is used to evaluate eigen values by choosing from factorial analysis tab. PCA (Principal Component Analysis) is a technique which enables to visualize a dataset in a lower dimension without loss of information. Basically, the PCA computes new attributes (says factors and axis) which are linear transformations of input attributes.

K-Means parameters	
Clusters	8
Max Iteration	5
Trials	5
Distance normalization	variance
Average computation	McQueen
Seed random generator	Standard

Fig. 6 K-Means Parameters

Above Fig. 6 shows the initial setting for implementing K-Means algorithm. Number of clusters produced are given in advance. Because k-means algorithm is very sensible to initial setting.

R-Square for each attempt

Number of trials	5
Trial	R-square
1	0.552908
2	0.572644
3	0.538656
4	0.524741
5	0.565060

Cluster centroids

Attribute	Cluster n°1	Cluster n°2	Cluster n°3	Cluster n°4	Cluster n°5	Cluster n°6	Cluster n°7	Cluster n°8
Var1	80.212469	37.070616	54.649063	41.311065	50.957929	54.670368	103.693703	39.328809
Var2	0.419196	-1.865550	0.037638	-0.964540	0.409655	-0.099282	58.295624	-6.445039
Var3	84.663249	79.525035	97.048497	81.355263	81.781796	79.486392	104.937033	102.009750
Var4	-1.257998	1.443048	0.120649	-0.389403	0.162831	0.768996	0.306297	2.246224
Var5	-21.727646	11.824456	48.786076	39.631274	49.780601	-1.409536	69.705443	31.455171
Var6	-27.603774	0.858568	0.467420	3.342969	4.800927	2.361855	15.591249	-0.454789
Var7	4.413043	42.662478	42.331449	40.052327	30.792408	24.642064	1.175027	62.596062
Var8	107.467596	67.786754	48.376525	41.669579	32.031834	81.431526	35.186766	70.433378

Fig. 7 Implemented K-Means algorithm with 8 clusters. Map Quality is 0.5726

K-means algorithm is applied by choosing the algorithm from clustering tab to produce 8 clusters. R-Square as shown above is 0.5726, which represents error rate of the implemented algorithm. Cluster centroids are also formed for each cluster.

Eigen values

Matrix trace	9.000000
Average	1.000000

Axis	Eigen value	Difference	Proportion (%)	Histogram	Cumulative (%)
1	3.118810	1.513828	34.65 %		34.65 %
2	1.604982	0.295464	17.83 %		52.49 %
3	1.309518	0.255022	14.55 %		67.04 %
4	1.054496	0.067289	11.72 %		78.75 %
5	0.987207	0.063926	10.97 %		89.72 %
6	0.923281	0.922551	10.26 %		99.98 %
7	0.000730	0.000109	0.01 %		99.99 %
8	0.000621	0.000266	0.01 %		100.00 %
9	0.000354	-	0.00 %		100.00 %
Tot.	9.000000	-	-	-	-

Fig. 8 Implemented PCA (Principal Component Analysis) on K-Means and evaluate Eigen Values

PCA is a technique which enables to visualize a dataset in a lower dimension without loss of information. Basically, the PCA computes new attributes (says factors and axis) which are linear transformations of input attributes.

5.2 Results

Both algorithms have been implemented by taking same number of clusters (8 clusters) and same number of iterations (5 iterations). Number of iterations must be less than the number of samples (i.e. clusters). Map topology also remains same for both algorithms. Both algorithms has been compared by using differ parameters. These algorithms have been implemented on the same dataset to analyse their performances. After implementation of these algorithms, the following results have been obtained:

Table 4. Comparative results of both algorithms

S.No.	Parameters	Kohonen SOM	K-Means
1	**No. of Clusters**	8	8
2	**Map Topology**	8	8
3	**Iterations**	5	5
4	**Error Rate**	0.5544	0.5726
5	**Computational Time**	953 ms	5578ms
6	**Accuracy**	High	Low
7	**Time Complexity**	Low $O(S)^2$	High O(nkl)
8	**Space Complexity**	High $O(N)^2$	Low O(k+n)
9	**Execution Time**	Fast	Slow

From above table the following results have been concluded.

- Kohonen SOM comparatively gives less error rate (55%) than K-means (57%). Error rate lies between 0 and 1.
- Computation time taken by the Kohonen SOM is very less as compare to the time taken by K-Means on the same data set.
- Results produced by SOM are more accurate than k-means.
- Time complexity for SOM is low and space complexity is high. These both parameters indicate the better algorithm.
- While implementing Kohonen SOM algorithm, it takes less time to execute as compared with K-means algorithm.

6 CONCLUSION AND FUTURE SCOPE

This comparative study suggests that Kohonen SOM gives better performance as compare to K-means. The performance of both algorithms is measured on the basis of few parameters like number of clusters, map topology, error rate, accuracy, computation time, complexity and execution time. Finally, it can be stated that when tested in a completely equal working conditions, Kohonen SOM can be considered better clustering algorithm as compared to K-Means. Future work can focus on how to reduce the time complexity without compromising cluster quality and optimality. More experiments will be conducted with natural datasets with different features.

7 REFERENCES

[1] Abbas, O.A., 2008. Comparison between Data Clustering Algorithms, The International Arab Journal of Information Technology, Vol. 5, No. 3, pp. 320-325.

[2] Sumathi, N., Geetha, R. and Bama, S.S., 2008. Spatial Data Mining- Techniques Trends and Its Applications, Journal of Computer Applications, Vol.1, No.4, pp. 28-30.

[3] Halkidi, M., Batistakis, Y. and Vazirgiannis, M., 2001. On Clustering Validation Techniques, Journal of Intelligent Information Systems, Vol.17, pp. 107–145.

[4] Sisodia, D., Singh L., Sisodia S. and Saxena, K., 2012. Clustering Techniques: A Brief Survey of Different Clustering Algorithms, International Journal of Latest Trends in Engineering and Technology (IJLTET), Vol. 1, Issue 3, pp. 82-87.

[5] S, Sundararajan and S, Karthikeyan, 2012. A Study On Spatial Data Clustering Algorithms In Data Mining, International Journal Of Engineering And Computer Science, Vol. 1 Issue 1, pp. 37-41.

[6] Ng, R.T. and Han, J., 2002. Clarans: A Method For Clustering Objects For Spatial Data Mining, IEEE Transactions On Knowledge And Data Engineering, Vol.14, No.5, pp. 1003-1016.

[7] Bacao, F., Lobo, V. and Painho, M., 2005. Self-organizing Maps as Substitutes for K-Means Clustering, Springer-Verlag, ICCS 2005, LNCS 3516, pp. 476 – 483.

[8] Aneetha, A.S. and Bose, S., 2012. The combined approach for anomaly detection using neural networks and clustering techniques, Computer Science & Engineering, International Journal (CSEIJ), Vol. 2, No. 4, pp. 37-46.

[9] Hemalatha, M. and Saranya, N.N., 2011. A Recent Survey on Knowledge Discovery in Spatial Data Mining, International Journal of Computer Science Issues, Vol. 8, Issue 3, No. 2, pp. 473-479.

[10] Boomija, M.D., 2008. Comparison of Partition Based Clustering Algorithms, Journal of Computer Applications, Vol.1, No.4, pp. 18-21.

[11] Toor, A.K. and Singh, A., 2013. Analysis of Clustering Algorithms Based on Number of Clusters, Error Rate, Computation Time and Map Topology on Large Data Set, International Journal of Emerging Trends & Technology in Computer Science, Vol. 2, Issue 6, pp. 94-98.

[12] Liao, S., Chu, P. and Hsiao, P., 2012. Data mining techniques and applications, Science Direct Expert Systems with Applications, Vol.39, Issue 12, pp. 11303-11311.

[13] Mingoti, S.A. and Lima, J.O., 2006. Comparing SOM neural network with Fuzzy c-means, K-means and traditional hierarchical clustering algorithms, European Journal of Operational Research, pp. 1742–1759.

[14] Ravikumar, S. and Shanmugam, A., 2012. Comparison of SOM Algorithm and K-Means Clustering Algorithm in Image Segmentation, International Journal of Computer Applications, Vol. 46, No. 22, pp. 21-25.

[15] Johal, H.S., Singh, B., Singh, H., Nagpal, A. and Virdi, H.S., 2012. Using Kohonen-SOM & K-Means Clustering Techniques to Analyze QoS Parameters of RSVP, Proceedings of the World Congress on Engineering and Computer Science, Vol. 1, pp. 431-436.

[16] Agrawal, R., Mehta, M., Shafer, J., Srikant, R., Arning, A. and Bollinger, T., 1996. The Quest Data Mining System, Proceedings of 1996 International Conference on Data Mining and Knowledge Discovery (KDD'96), Portland, Oregon, pp. 244-249.

[17] Subitha, N. and Padmapriya, A., 2013. Clustering Algorithm for Spatial Data Mining: An Overview, International Journal of Computer Applications, Vol. 68, No.10, pp. 28-33.

[18] Sharma, N., Bajpai, A. and Litoriya, R., 2012. Comparison the various clustering algorithms of weka tools, International Journal of Emerging Technology and Advanced Engineering, Vol. 2, Issue 5, pp. 73-80.

[19] Bacao, F., Lobo, V. and Painho, M., 2004. Clustering census data: comparing the performance of self-organising maps and k-means algorithms.

[20] Dhingra, S., Gilhotra, R. and Ranishanker, 2013. Comparative Analysis of Kohonen-SOM and K-Means data mining algorithms based on Academic Activities, International Journal of Computers & Technology, Vol. 6, No. 1, pp. 237-241.

[21] Singh, P., 2014. An Efficient Concept-Based Mining Model for Analysis Partitioning Clustering, International Journal of Recent Technology and Engineering, Vol. 2, Issue 6, pp. 1-3.

[22] Sharma, K. and Dhiman, R., 2013, Implementation and Evaluation of K-Means, Kohonen-SOM, and HAC Data Mining Algorithms base on Clustering, International Journal of Computer Science Engineering & Information Technology Research (IJCSEITR), Vol. 3, Issue 1, pp. 165-174.

[23] Birdi, M., Gangwar, R.C. and Singh, G., 2014. A Data Mining Clustering Approach for Traffic Accident Analysis of National Highway-1, International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 4, Issue 10, pp. 44- 47.

[24] Shukla, S. and Naganna, S., 2014. A Review on k-means data clustering approach, International Journal of Information & Computation Technology. Vol. 4, No. 1,

[25] Tanagra-A free data mining tool for teaching and research,http://eric.univlyon2.fr/~ricco/tanagra/en/tanagra.html

[26] Ai, H. and Li, W., 2012. K-means initial clustering center optimal algorithm based on estimating density and refining initial, IEEE, pp. 603-606.

[27] Nalawade, K.M. and Gaiwad, R.G., 2013. Clutsering Algorithm With Reference To TANAGRA, Indian Journal of Applied Research, Vol. 3, Issue 7, pp. 591-593.