Single Performance And Cluster Evaluator (space) for clustering algorithms comparison

Fiaz Ahmed[*], Nazia Perveen[**]

[*] Department of Computer Science University of Gujrat Pakistan
[**] Department of Computer Science Virtual University of Pakistan

DOI: 10.29322/IJSRP.9.07.2019.p9185
http://dx.doi.org/10.29322/IJSRP.9.07.2019.p9185

Abstract- Clustering is renowned data mining technique used in exploratory Data Analysis which is based on unsupervised learning. Many clustering algorithms have been developed so for, and are applied in variety of disciplines based on nature of problem .The selection of suitable algorithm is necessary for accuracy, nature and complexity of data and practical representation of results.In this paper, we present SPACE algorithm which stands for**"Single Performance And Cluster Evaluator"**. This paper also reviews the clustering algorithms and provides foundation to the data analyst to further explore the power of clustering. SPACE algorithm is novel technique which provides solid analysis and visualizes the results efficiently.

Index Terms- space, cluster, kmeans, hierarchical, algorithm, Density-based.

I. Introduction

Clustering is unsupervised learning technique which partition the data in unknown groups based on similarity of instances. The data instances with maximum similar features are placed in the same cluster and similarity must be practically meaningful. The basic steps in clustering are selection of representative features, selection of appropriate algorithms, evaluation of results and explanation and representation of results. Clustering helps to detect outliers and manage the data in groups which can be analyzed from different perspectives and help in decision making in different aspects of life. For example clustering helps to know:

I. Group the customers with similar buying habits?
II. Show the documents with relevant contents?
III. Show the customers with extra-large transactions (help in fraud detection).outlier detection.
IV. Identify the users with similar browsing history.(help in personalized services in www.)
V. Lot of more examples.

Clustering is unsupervised classification, we can't say that particular object will go into a particular group (cluster) rather the attributes are used to check the similarity of objects.

1.1. Things Before Clustering.

There are important points which must be in mind before stepping into clustering. Few of them are as follows:

1.1.1. **1.1.1. Data Preprocessing**

Although different clustering algorithms can cope with curse of dimensionality and other starting drawbacks but the quality of clustering results widely effected by the pre-preprocessing techniques. Therefore, the data analyst must be familiar with data preprocessing techniques like, scaling,handling missing values, dimensionality reduction ,normalization and handling categorical data.Different data preprocessing for clustering are given in (1)(2).

1.1.2. **1.1.2. Similarity measure**

The composition of data helps to understand itsnature. All the attributes of objects may not be relevant for proper clustering results. Therefore, select relevant features .For Quantitative data features we use different distance functions to measure the similarity among object. The common distance functions are Minkowsky,Euclidean and pearson correlation distance functions(3). For Qualitative data attributes we use different similarity functions instead of distance functions(4) .Therefore ,the data analyst must be familiar with distance functions and similarity function .These functions are summarized in(5).

1.1.3. **1.1.3. Handling parameters.**

Most of the clustering algorithms need some parameters to set while working with them.therefore, data analyst should adjust them frequently to get appropriate results.Although many algorithms exists which are parameterless like APSCAN.a new trend for parameter less clustering is given in (6).

1.1.4. **Evaluation indicators.**

The evaluation of clustering algorthms is very important for effective results. There are two main approaches which are internal evaluation and external evaluation. Both of these can not completely judge the quality of clustering and are only informative to identify bad clusters(7).the common internal evaluation indicators are,Davies-Bouldin index,Dunn index and Silhouette coefficient.whereas external evaluation indicators are purity,random-measure , F-measure and confusion-matrix .the comprehensive disscussion of these can be found in (8).

1.1.5. **Clustering software (Tool)**

The selection of data clustering software plays a vital role to succeed in clusting analysis because the user has to ultimately run the clustering algorithm in any software. There are lot of open source and premium data analysis tools which apply clusterig algorithms.They are based on different underlying plateforms

(OS,languages,libraries ,etc) and interface (command-line,GUI).The Tools with command-line interface like R , scikit-learn etc. provide more power and flexibilty to the Data Analyst as compared to the GUI Tools like Weka,SPSS etc.The use of any clustering software should not effect the quality of clustering results. Almost all data analysis tools are backed by large community of developers, data analysts,framework supporters ,tutorials and forums.They are updating ,resolving issues , developing and refining algorithms and interfaces. Therefore ,the the user must join the relevent community of that software to keep up with updates.table[2] presents famous tools with download links.The large scale overview of data analysis tools can be found in (9)

Clustering Tools with download links.		
Data Analysis Tool	**Download page**	**Features**
Weka	https://www.cs.waikato.ac.nz/ml/weka/downloading.html	• By Univesity of waikaito For Machine Learning and knowledge analysis under GNU under general purpose Licence. • Require Java runtime environment. • Documentation-url: https://www.cs.waikato.ac.nz/ml/weka/documentation.html • dataset from UCI repository are included in download package. • GUI interface.
R	https://www.r-project.org/	• R is language and intigrated suite of software for data manipulation ,data handling and graphic display. • Under GNU general Public License. • Complete and flexible environment to apply statistical techniques. • Few R packages are available with R distribution and rest of the package are available through CRAN family of internet sites. • Command-Line as well as GUI interfaces. • The best Feature is R help.
CLuto	http://glaros.dtc.umn.edu/gkhome/cluto/cluto/download	Cluto is copyright by unversity of Minnesota and can by used for educational and research purpose. The documents need to be converted in cluto formate by using perl script given on the site.
SPSS	https://www.ibm.com/analytics/data-science/predictive-analytics/spss-trials	Proprieter software by IBM . Trial version is available. GUI interface and easy to use.
Scikit learn	http://scikit-learn.org/stable/install.html	Open source under BSD License. Built on Python,numpy,scipy and matplotlib. Simple and powerful machine learning tool.

II. LITERATURE REVIEW

Clustering alogrithms can be divided into two broad categories which are classic (traditional)and Modern ones. Traditional algorithms are further divided based on the techniques used to create cluster such as partition-based, Hierarchical, Grid-Based, Density-Based, Model-Based and fuzy-theory based. Modern clusteringalgorithms use novel techniques and mix of the traditional algorithms to achieve better results.in this paper we briefly review the traditional clustering algorithms.

1.2. Partition-Based Algorithms

Partition-Based-Algorithms partition the dataset into k clusters by using the mean , median or Mode as centroid. They randomly select some objects (call them centroid) and then compute the distance of other objects from these objects.Then again select new centroid and compute distance of other objects to form K clusters. In each iteration we try to optimize some objective function by iterative minimizing the distance of points from centroid.

K-means is the famous partitioning algorithm it works in very simple way. It require the k as input parameter. The algorithms is :

K-Means: For input k and D where k is number of clusters and D is data set.

I. At random select k points as cluster centers call them centroids.
II. compute distance of all points from these centroids.
III. Assign each point to that centroid which is at minimum distance. Each point will be the part of one of k clusters .
IV. Compute the mean of each cluster and this mean will be the new centroid .
V. Repeat the step 2-4.
VI. Stop until no change in mean values.

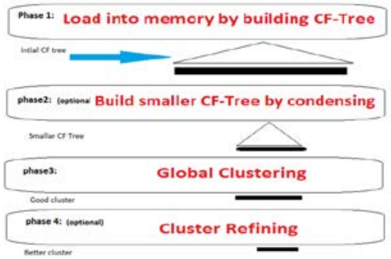
Output: k clusters.

Different variations of k-means were developed and they differ in their distance function or the centroid selection as given in(10).Simple k-means minimize the error-function.In(11)Author present the different variations of k-means from its origin to date.

PAM uses Mode instead of mean to reduce the effect of noise (mean is more effected by the biased data), and replace the cluster center(medoid) with non-selected object and try to decrease the objective function as given in(4).PAM has drawback of computational complexity due to selecting of the entire data set. CLARA is another algorithm which, unlike PAM, select a random sample for clustering and has improved performance than PAM. CLARANS is improved version of CLARA which is based on randomized search. If you compare PAM, CLARA and CLARANS on some dataset it will be clear that the later will outperform the other twoas compared in(12).

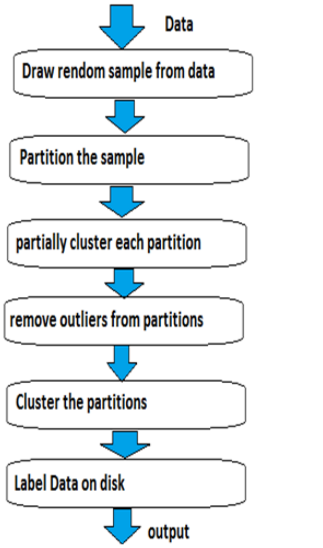
1.3. Hierarchical-Clustering

These methods decompose the dataset hierarchically and form a tree like structure called dendrogram. Tree (dendrogram) is built either from bottom up (agglomerative) fashion or from top to bottom (divisive) manner. There are different measures to split or merge two clusters. The famous criteria is to compare distance between cluster means. Termination condition of merging or splitting, updating the distance matrix and failing to detect arbitrary shaped clusters are the main issues of hierarchical methods.Birch is example of hierarchical clustering.An other algorithm called chameleon is given in(13).BIRCH stands for **B**alance **I**terative **R**educing and **C**lustering using **H**ierarchies' .BIRCH. Uses the CF-Tree data structure for clustering the large data set as given in(14) .figure-05


Phase 1: Load into memory by building CF-Tree
Intial CF tree
phase2: (optional Build smaller CF-Tree by condensing
Smaller CF Tree
phase3: Global Clustering
Good cluster
phase 4: (optional) Cluster Refining
Better cluster

Birch Algorithm figure.05

CURE stands for **C**lustering **U**sing **Re**presentative .Ituses few representative points instead of a single centroid for clusters .This algorithm can handle outliers very efficiently and can detect clusters with different shapes (Non-Convex) and sizes .If you carefully look at the algorithm and its working you will agree that it can be describe as given in (15).figure.06.


Data
Draw rendom sample from data
Partition the sample
partially cluster each partition
remove outliers from partitions
Cluster the partitions
Label Data on disk
output

Cure algorithm .figure.06

1.4. Grid-Based Clustering

Grid-base algorithms form Grid of input for clustering .For example CLIQUE and STING form grid and rectangular boxes (cell) of input and merge the adjacent high density cells. STING stands for StatisticalInformation Grid-base clustering (16). STING algorithms works as :

STING Algorithm:

I. Pick a layer to start .
II. Find confidence for each cell of layer for query relevance.
III. If this is bottom layer go to 5 otherwise 4.
IV. Go to next level of hierarchy and repeat step 2 for the relevant cell of higher level.
V. If query is satisfied go to 7 else 6.
VI. Retrieve data from relevant cells for further processing and return result. Go to step 8.
VII. Return the regions of relevant cells.
VIII. Stop.

1.5. Density-Based-Clustering:

These algorithms separate dense regions of objects from low density regions and form cluster of arbitrary shapes depending upon the density distribution of data objects(17).Example is DBSCAN.Density based algorithm to handle noise is given in (18).

According to (17) DBSCAN(**DB** for Density-Based **S** For Spatial **C** for Clustering **A** for Applications **N** For Noise) is Density-Based and can construct cluster with any shape depending upon data and this algorithm requires less parameters .The Parameters required for DBSCAN are Eps and Minpts. These are the terms which can be described as:

Eps can be called as radius of the cluster. The points inside this distance are called Eps-neighborhood (**Eps short for epsilon**) of a point.Minpts stands for Minimum points with in Eps-Range (neighborhood).Formal definitions can be found in (19).

DBSCAN Algorithm:

Step1. Select P point arbitrarily.

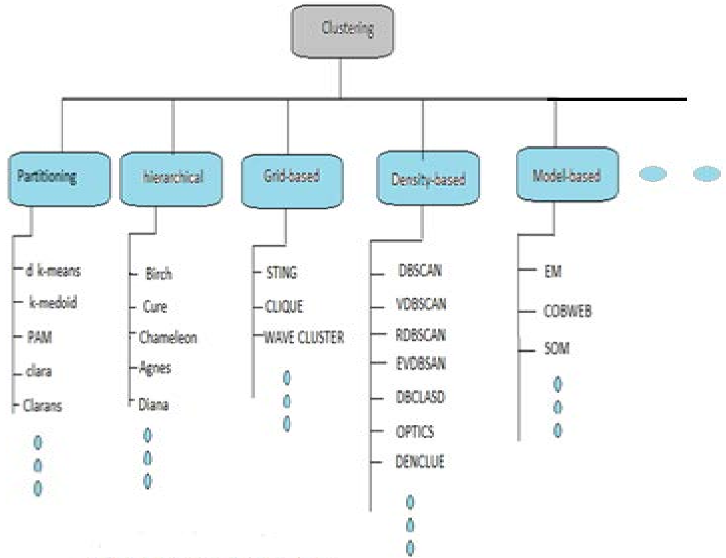
Step2. Get all Density-Reachable pointsfromp within Eps and Minpts range.
Step.3 Cluster will be formed if p is the Core-Point otherwise visit the next point.
Step.4. continue until all the points have been processed.

<u>RDBC-Algorithm:</u>

RDBC-Algorithm (**R** For Recursive **D** for Density **B** for Based **C** for Clustering) .it is also called Recursive-DBSCAN .It is improvement to the original DBSCAN. It separates the core points from the data set and apply DBSCAN on core points. Than it assigns the remaining points to the clusters formed from core points .

1.6. Model-Based-Clustering.

These algorithms use hypothetical Modelsand try to adjust the data so that it can fit to some model. The model-based clustering algorithms assume that the data were generated from a model and try to discover original model from data .the common model-based clustering algorithm is Expectation-maximization (EM). The detail comparison of model-based clustering algorithm can be found in (20). Summary of classic clustering algorithms.



Different clustering Algorithms.fig.03

1.7. Modern Algorithms:

Modern clustering algorithms use more sophisticated approaches to form clusters .they mostly use mix of traditional algorithms with new techniques .we can divide the modern algorithms based on approach use as we did in classic one. Kernel-based, swarm intelligence-based, quantum Theory-based, ensemble-based, Graph theory-based are major categories to consider.They also use optimization techniques which individually or in combination with other algorithms are used.Ant-Colony-Optimization(ACO) and Particle-Swarm-Optimization (PSO)are very famous techniques which have been applied in recent years in different areas of computer science.

III. SINGLE PERFORMANCE AND CLUSTER EVALUATOR (SPACE)

As discussed in literature review section of this paper clustering algorithms are different in terms of input parameters, size of input ,way of working and presentation of results. SPACE algorithm provides the data analyst a single plateform to apply and test the result of famous clustering algorithms .it takes the dataset and clustering algorithms along with parameters as input ,and show the results on graph with different colors, for visual evidence . It also append the result as new column with original data set. The algorithm is quite flexible and efficient for measuring the performance of each algorithms. It also keeps record of all the cluster centers and can print them, if needed.

1.8. SPACE Algorithm Formal Definition

```
Input:
    I.   clustering algorithm   # like kmean, dbscan etc.
    II.  Dataset                # the actual dataset X to apply clustering.
    III. Arguments              # parameters like k, eps etc.
    IV.  Kwds                   # additional arguments
Start :
On input: space(X,algorithm,args,kwds)
    1. Variable declaration:
    2. Apply clustering algorithm
    3. Print results
    4. Plot results on graph with time consumed
    5. append the results with original dataset and save in global result
    6. return global result
End
```

1.9. Methodology and workplan

As discussed in introduction section of the paper ,that preprocessing and selection of clustering tool is necessary part of clustering. We preferred to use sickit learn in python language due to its flexibility and open source availability.we used jupyter notebook environment to code and test in python language.

3.3. Sample code in python

```
# here is the sample code of space algorithm please proper indent the code .this can be #found on my github account or email to get notebook.
def space(X, algorithm, args,kwds):
# start time stamps
start_time = time.time()
algo=algorithm(*args,**kwds)
algo.fit_predict(X)
labels=algo.labels_                        # show the labels of   cluster
results=pd.Series(labels)                  # save results to global variable
global result
        # print parameters                     print(algo)
clusters=labels.max()+1              # total clusters ,labels start from 0
cluster_values=Counter(labels)          # instances assigned
end_time = time.time()

#################################################
# show number of clusters and elements of each cluster
print("Total cluster are",clusters)
print("cluster wise instance are : \n (0 for c1 ,1 for c2 and so on):",cluster_values)
#####################################################
# this portion is to show cluster centers (centroids) and to skip for spectral and #agglomerative algorithms
if(algorithm==cluster.SpectralClustering)or (algorithm==cluster.AgglomerativeClustering)or (algorithm==cluster.Birch)or (algorithm==cluster.DBSCAN):
 pass   # do not have clusters  hence pass.
else:          # as spectral and agglomerative dont have centers
centers=algo.cluster_centers_         # centeroids
#print("clusters centers are ",centers) uncomment if we needed
#######################################################
# ploting portion to show clusters with different colors on scatter plot
# assign different colors to clusters
palette=sns.color_palette('deep', np.unique(algo.labels_).max()+1)
colors = [palette[x] if x >= 0 else (0.0, 0.0, 0.0) for x in algo.labels_]
# 4rth and 5th columns of X for clustering
plt.scatter(X.iloc[:,4],X.iloc[:,5],c=colors,cmap='rainbow')
plt.title('Clustersfoundby{}'.format(str(algorithm.__name__)),fontsize=24)
# print the total time taken by the algorithm to form clusters
plt.text(0.5, 0.9, 'Clustering took: {0:6.3f} s'.format(end_time - start_time), fontsize=14)
# prints total cluster count on graph
plt.text(0.1,0.1,"totalclustersFoundby{clust}are:{num}".format(clust=str(algorithm.__name__)
,num=clusters,fontsize=14))
plt.show()
########## add new column with algorithm name to see the results
result =pd.concat([X,results.rename(str(algorithm.__name__))],axis=1)
return result
#here is one way how to call and save result manually
space(X, cluster.KMeans,(), {'n_clusters':6})
result.to_csv("D:/Data/Results/2017_results.csv")
```

3.4. Datasets

The data used for clustering was taken from kaggle ,sickit learn datasets and UCI Repository (21).The dataset can be downloaded from the link given in (11)(22).

Table -1 Data sets

The dataset	Source	Type	Attributes	Instances	Missing values
Word happiness	kaggle	numeric	10	155	none

Survey_ result stackoverflow	kaggle	Numeric, categorical	38	98855	yes
xclara	Sickit learn	numeric	2	3000	none
Wholesale customers data	UCI	numeric	8	440	none
The iris	UCI	numeric	5	150	none
Diabetic	UCI	Numeric	50	101766	none
Weather	UCI	Numeric	5	14	none
CPU-vendor	UCI	Nominal	8	209	1
Online retail	UCI	numeric	8	541909	none

3.5. Clustering algorithms Tested.

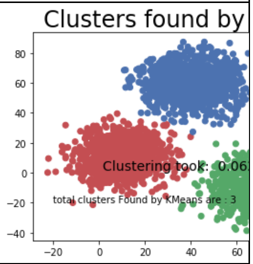
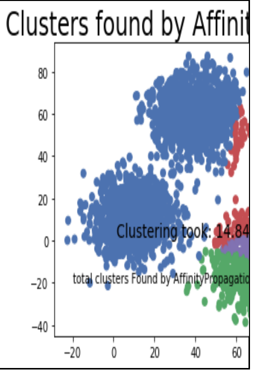
The following algorithms were applied and used in testing.

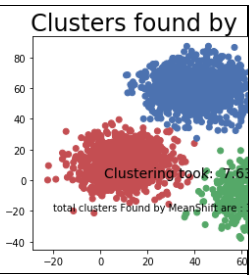
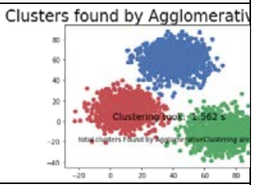
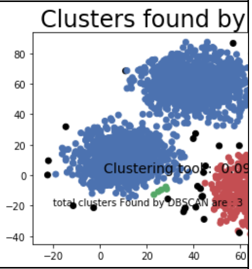
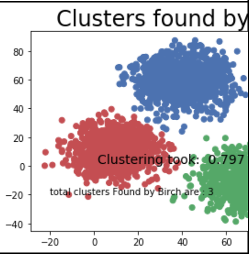
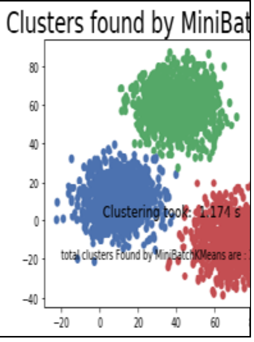
1. KMeans
2. MeanShift.
3. MiniBatchKMeans
4. AffinityPropagation
5. SpectralClustering
6. AgglomerativeClustering
7. DBSCAN
8. Birch

3.6 Results.

The following table shows the results of clustering algorithms applied on xclara data set with known 3 clusters the dataset have two columns with names V1 and V2 .the parameters were adjusted to bring the cluster count to 3 (if possible for comparison purpose).

Table -2 Results

Sr.no	**Algorithm Name**	**Parameter set**	**Clusters**	**Performance (seconds)**	**Plot image**
1	kmeans	n_cluster=3	3	0.062	
2	Affinity propogation	Pref =-80, damping= 0.95	4 (failed to form=3)	14.842	

3	Mea n Shift	Clust er all=true	3	7.633	
4	Spect ral clustering	n_cl uster=3	N/ A	N/A	Took too long to return ,tried different parameters {As TMs is undecidable}
5	Aglo meratve clustering	n_cl uster=3	3	1.453	
6	DBSC AN	Eps= 5.2	3 (outlier =35)	0.094	
7	Birch	n_cl uster=3	3	1.703	
8	Mini Batch Kmeans	n_cl uster=3	3	1.174	

3.7. Discussion.

The total eight algorithms with different parameters were given as input to the SPACE algorithm as input alongwith dataset. From Table 2 and figure 4 below it is clear that affinity propagation is not suitable for this dataset.

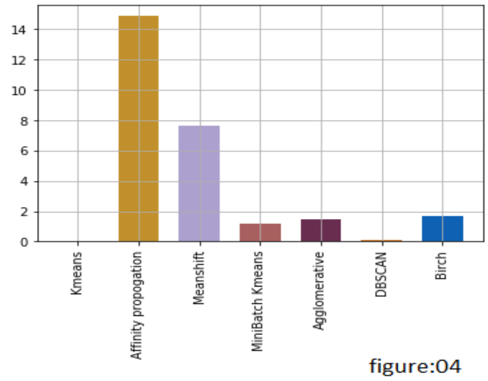


figure:04

running time on xclara dataset

Out of eight algorithms, six algorithms successfully presented the actual 3 big clusters in dataset. This does not mean that we cannot find more clusters with in these three rather we can do, and we actually found 6 or more but these does not have much dissimilarity. The same algorithms were tested on remaining datasets and the presentation of all the results can be found on my official GitHub page or can be requested by email. The performance of algorithms becomes very important when the dataset is very large. As the resource requirement and efficiency varies from application to application therefore, you should carefully select clustering algorithm which best suit your needs.

3.8. Further Explore

As discussed in **“Sample code in Python “**section of this paper the SPACE algorithm form a new column with algorithm name and append with actual dataset and this dataset can be saved in file on disk. This helps to explore and elaborate the results in more details as given in top 10 rows of appended results table.

Table-3 Appended results

	Channel	Region	Fresh	Milk	Grocery	Frozen	Detergents_Paper	Delicassen	KMeans
0	2	3	12669	9656	7561	214	2674	1338	0
1	2	3	7057	9810	9568	1762	3293	1776	0
2	2	3	6353	8808	7684	2405	3516	7844	0
3	1	3	13265	1196	4221	6404	507	1788	0
4	2	3	22615	5410	7198	3915	1777	5185	1
5	2	3	9413	8259	5126	666	1795	1451	0
6	2	3	12126	3199	6975	480	3140	545	0
7	2	3	7579	4956	9426	1669	3321	2566	0
8	1	3	5963	3648	6192	425	1716	750	0
9	2	3	6006	11093	18881	1159	7425	2098	0
10	2	3	3366	5403	12974	4400	5977	1744	0

For any dataset, when I changed the distance function the running time was also changed. The running time increases with the increase in instances. When I applied the same algorithm on different datasets with the same number of instances the number of clusters were different in case of density based algorithms. Partitioning algorithms like k-mean are efficient for fast clustering applications.

IV. Conclusion

With our experimental results and discussions it is obvious that the choice of appropriate algorithm is very important before applying clustering to any problem domain .The choice of clustering algorithm also depends on data.
We also should kept in mind that:

1. There is no Universal clustering algorithm which can solve all problems.
2. The time taken to form clusters, the number of clusters, the shape and size of cluster, ability to handle large dataset, robustness, the input parameters, use of distance or partitioning function, handling the type of data and number of dimensions are the major characteristics which decide the success of an algorithm.
3. Due to the advancement in technologies and use of computing in all aspects of life, there is always room to improve the clustering algorithms and find new ones.
4. There is overlap of categories of algorithms and novel algorithms are hybrid of the present algorithms.

Compliance with Ethical Standard:
Conflict of Interest:

1. Author 1(fiaz ahmed) declares that he has no conflict of intrest.
2. Author 1(fiaz ahmed) declares that he has no conflict of intrest.

Ethical approval: This article does not contain any studies with human participants or animals performed by any of the authors.

Funding: No funding was obtained from any one.

References

[1] 1. Facilitating data preprocessing by a generic framework: a proposal for clustering. kathrine krichner, jelena zec. s.l. : artificial intelligence reviews, 2015.

[2] 2. *A Data Preprocessing Method Applied to Cluster Analysis on Stock Data by Kmeans.* zhigang Xiong, Zhongneng zhang. s.l. : international conference on intelligent control and computer applications (ICCA 2016), 2016.

[3] 3. *Clustering Techniques and the Similarity Measures used.* Jasmine irani, Nitin pis,Madhura Phatak. 7, s.l. : internal journal of computer applications, 2016, Vol. 134.

[4] 4. Leonard Kaufman, Peter J. Rousseeuw.*Finding Groups in Data: An Introduction to Cluster Analysis.* s.l. : Wiley Series in Probability and Statistics, 2008.

[5] 5. *A comprehensive survey of clustering algorithm.* dongkuan Xu, yingjie Tian. s.l. : springer-verlag, 12 August 2015.

[6] 6. *Towards parameter-independent data clustering and image segmentation.* jian Hou, weixue liu. pattern recognition, s.l. : ELSEVIER, december 2016, Vol. 60.

[7] 7. https://nlp.stanford.edu/IR-book/html/htmledition/evaluation-of-clustering-1.html. [Online]

[8] 8. *Evaluating and Analyzing Clusters in Data mining using different algorithms.* N.sonil chowdhary, sri lakshmi prasanna. 2, s.l. : international journal of computer science and mobile computing., february2014, Vol. 3.

[9] 9. https://www.predictiveanalyticstoday.com/top-data-analysis-software/. [Online]

[10] 10. Charu.C Aggerwal ,Chandan .K Reddy. *Data Clustering:Algorithms and Applications.* s.l. : CRC press,2013, 2013.

[11] 11. *Data clustering: 50 years beyond K-means.* jain, Anil k. s.l. : International Conference in Pattern Recognition (ICPR), 2010. Vol. 31.

[12] 12. *CLARANS: A Method for clustering objects for spatial data mining. "IEEE vol.14. no.5,2002.* Han, Raymond T. Ng and Jiawei. 5, s.l. : IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, 2002, Vol. 14.

[13] 13. chameleon: A hierarchical clustering algorithm using dynamic modeling. karypis, Han and kuma. s.l. : IEEE , 1999.

[14] 14. Tian zhang, Ramakrishnan and livny. *Birch:An efficient data clustering method for large databases.* s.l. : ACM SIGMOD, 1996.

[15] 15. sudipto, Rastogi. CURE: an efficient clustering algorithm for large databases. s.l. : ACM SIGMOD conference, 1998.

[16] 16. wang, yang and muntz,. *STING:A statistical information grid approach to spatial data mining* . s.l. : technical report no.970006 computer science department UCLA, 1997.

[17] 17. *A Review on Density based Clustering Algorithms for Very Large Datasets.* Lovely sharma, prof.k.Ramya. 12, s.l. : International Journal of Emerging Technology and Advanced Engineering, 2013, Vol. 3.

[18] 18. *A density-based algorithm for discovering clusters in large spatial databases with noise.* Martin Ester, Hans-Peter Kriegel, Jiirg Sander, Xiaowei X. s.l. : AAAI press, 1996. second international conference on knowledge discovery and data mining. pp. 226-231.

[19] 19. *A survey on density based clustering algorithms for mining large spatial databases.* M.parimala, Dephne,senthilkumar. s.l. : international journal of advance science and technology, 2011, Vol. 31.

[20] 20. Marina Meila, hackerman. *An experimental comparison of model-based clustering algorithms.* s.l. : Microsoft Research Redmond WA 98052 USA, 2001.

[21] 21. https://archive.ics.uci.edu/ml/index.html. *UCI machine learning repository.* [Online]

[22] 22. kaggle. https://www.kaggle.com/datasets. [Online]

Authors

First Author – Fiaz Ahmed, Department of Computer Science University of Gujrat Pakistan, email : fiazmianwal@gmail.com
Second Author – Nazia Perveen, Department of Computer Science Virtual University of Pakistan, email : naziamianwal@gmail.com, Phone: +92-346-6469074