FICA: Fast Incremental Clustering Algorithm

Omar Kettani
Scientific Institute, Geophysics Laboratory
Mohamed V- University
Rabat, Morocco

Faical Ramdani
Scientific Institute, Geophysics Laboratory
Mohamed V- University
Rabat, Morocco

ABSTRACT
In this study a simple deterministic clustering method, called FICA (*Fast Incremental Clustering Algorithm*) is proposed. Its initialization phase consists to run the Katsavounidis, Kuo & Zhang (KKZ) seed procedure, and its incremental step consists simply to assign each data point to its nearest cluster, then the centroid of the last modified cluster is updated. The proposed approach has a lower computational time complexity than the famous k-means algorithm. We evaluated its performance by applying on various benchmark datasets and compare with a related deterministic clustering method: KKZ_ k-means (k-means initialized by KKZ). Experimental results have demonstrated that the proposed approach is effective in producing consistent clustering results in term of average Silhouette index.

General Terms
Data Mining, Algorithms.

Keywords
Clustering, k-means, KKZ, Silhouette.

1. INTRODUCTION
In Data Mining, Clustering is the process of grouping similar data into sets called clusters, so that the objects in the same cluster are more similar to each other and more different from the objects in the other clusters [1]. This optimization problem is known to be NP-hard, even when the clustering process deals with only two clusters [2]. Therefore, many heuristics have been proposed, in order to find near optimal clustering solution in reasonable computational time. In the present study, yet another clustering approach which has the advantage of low computational complexity, is suggested.

In the next section, some related work are briefly discussed. Then the proposed algorithm and its computational complexity are described in Section 3. In section 4, this clustering approach is applied to some standard data sets and compared with a related deterministic clustering method, KKZ_k-means (k-means initialized by KKZ). Finally, conclusion of the paper is summarized in Section 5.

2. RELATED WORK
Given a set of n data points (objects) $X = \{x_1, \ldots, x_n\}$ in R^d and an integer k, the clustering problem consists to determine a partition $(C_j)_{1 \le j \le k}$ of X , in order to minimize the following Sum of Square Error (SSE) function:

$$SSE = \sum_{i=1}^{k} \sum_{x_j \in C_i} \| x_j - c_i \|^2 \qquad (1)$$

$$c_i = (1/|C_i|) \sum_{x_j \in C_i} x_j \qquad (2)$$

where $\| . \|^2$ denote the Euclidean norm , and

denote the centroid of cluster C_i whose cardinality is $|C_i|$.

Among the many existing clustering methods, the k-means algorithm [3][4] is the most popular clustering approach, for its efficiency and simplicity. However, one of its major drawback is its sensitivity to initial seeds. Several methods have been proposed to overcome this issue, among them, the Katsavounidis, Kuo & Zhang (KKZ) seed procedure [5], which consists to pick the data point with the highest Euclidean norm as the first center. Then, it chooses the next center to be the point that is farthest from the nearest seed in the set chosen so far. This approach has a computational time complexity in O(knd). In the present paper, an alternative approach to the k-means algorithm is proposed: Its initialization phase consists to run the KKZ seed procedure, and its incremental step consists simply to assign each data point to its nearest cluster, then the centroid of the last modified cluster is updated.

3. PROPOSED APPROACH

3.1 Pseudo-code
The pseudo-code of the proposed FICA approach is shown below:

Input: A data set X whose cardinality is n and an integer k

Output: k cluster C_j

1. $c_1 \leftarrow \text{Arg}(\text{Max}(\| x_h \|))$
   $1 \le h \le n$

2. **For** j=2:k **do**
   $m \leftarrow \text{Arg}(\text{Max}(\text{Min}(\| x_i - c_h \|)))$
   $1 \le i \le n \quad 1 \le h \le j-1$
   $c_j \leftarrow x_m$

   end For

3. **For** i=1:n **do**
   $j \leftarrow \text{Arg}(\text{Min}(\| x_i - c_h \|))$
   $1 \le h \le k$

   $C_j \leftarrow C_j \cup \{x_i\}$

   $c_j \leftarrow \text{mean}(C_j)$

   end For

4. EXPERIMENTAL RESULTS

Algorithm validation is conducted using different data sets from the UCI Machine Learning Repository [6]. We evaluated its performance by applying on several benchmark datasets and compare with KKZ_ k-means. In preprocessing step, the data were normalized.

Silhouette index [7] which measures the cohesion based on the distance between all the points in the same cluster and the separation based on the nearest neighbor distance, was used in these experiments in order to evaluate clustering accuracy. Given observation i, let a_i be the average distance from point i to all other points in same cluster and d (i, j) represents the average distance from point i to all points in any other cluster j. Finally, let b_i denotes the minimum of these average distances d (i, j). The silhouette width for the i-th observation is:

$silh(i) = (b_i - a_i)/max(a_i , b_i)$

The average silhouette width can be find by averaging silh(i) over all observations:

$$silh = \sum_{i=1}^{n} silh(i)/n \qquad (3)$$

The silhouette width silh(i) ranges from -1 to 1. If an observation has a value close to 1, then the data point is closer to its own cluster than a neighboring one. If it has a silhouette width close to -1, then it is not very well clustered. Kaufman and Rousseeuw [7] use the average silhouette width to estimate the number of clusters in a data set by using the partition with two or more clusters that yields the largest average silhouette width.

FICA was compared with a related deterministic clustering method:KKZ_k-means (k-means initialized by KKZ).

Experimental results are reported in table 1 and figure 1, and some clustering results are depicted in figure 2 to 5.

Table 1. Experimental results of KKZ_k-means and FICA application on different datasets in term of average Silhouette value.

Data set	**k**	**KKZ_k-means**	**FICA**
Iris	3	0.7542	**0.8121**
Ruspini	4	0.9086	**0.9097**
Aggregation	7	0.6536	**0.7856**
Compound	6	0.6484	**0.6540**
Pathbased	3	**0.7316**	0.6464
Spiral	3	0.5206	**0.5234**
D31	31	0.5881	**0.8135**
R15	15	0.5966	**0.7715**
Jain	2	0.6719	**0.9078**
Flame	2	0.5347	**0.8760**
Dim32	16	0.7472	**0.9961**
Dim64	16	**0.9985**	0.9984
Dim128	16	**0.9991**	**0.9991**
Dim256	16	**0.9996**	**0.9996**
Dim512	16	**0.9998**	**0.9998**
dim2	9	**0.7818**	0.6898
dim3	9	0.3968	**0.9463**
dim4	9	0.6949	**0.9749**
dim5	9	0.4492	**0.9918**
dim6	9	0.5947	**0.9557**
dim7	9	0.5651	**0.9553**
dim8	9	0.4599	**0.9938**
dim9	9	0.4155	**0.9260**
dim10	9	0.3738	**0.9115**
dim11	9	0.4696	**0.9041**
dim12	9	0.5060	**0.9140**
dim13	9	0.8106	**0.9304**
dim14	9	0.4533	**0.9258**
dim15	9	0.7210	**0.9087**
a1	20	0.5542	**0.6051**
a2	35	0.5970	**0.5981**
a3	50	0.5752	**0.6871**
tk.4.8k	6	0.6051	**0.6771**

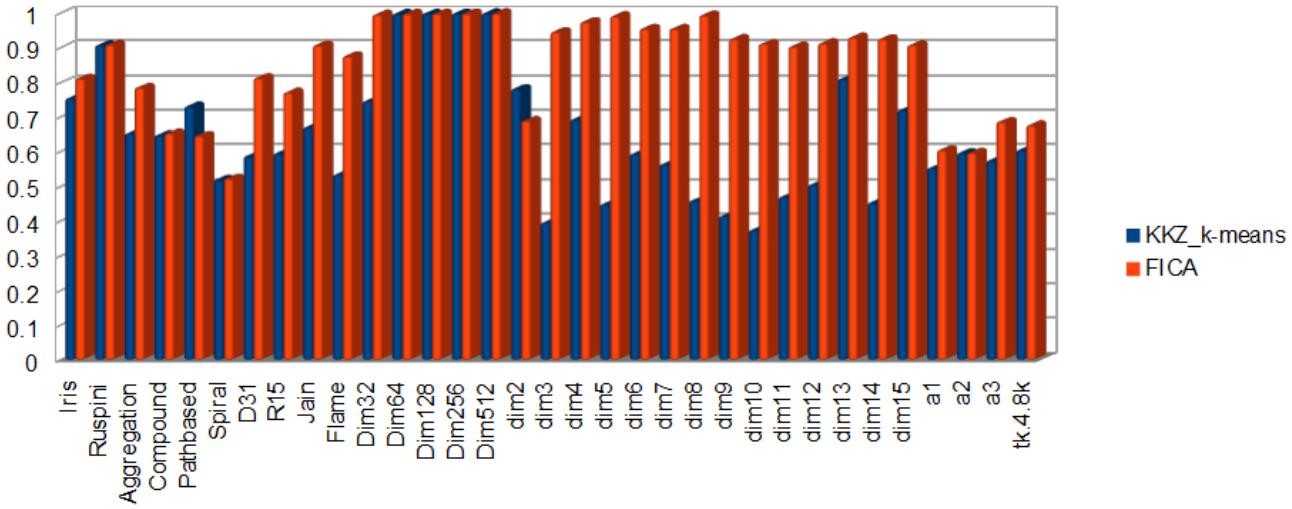

Fig 1: Chart of average Silhouette index for FICA and KKZ_k-means applied on different datasets.

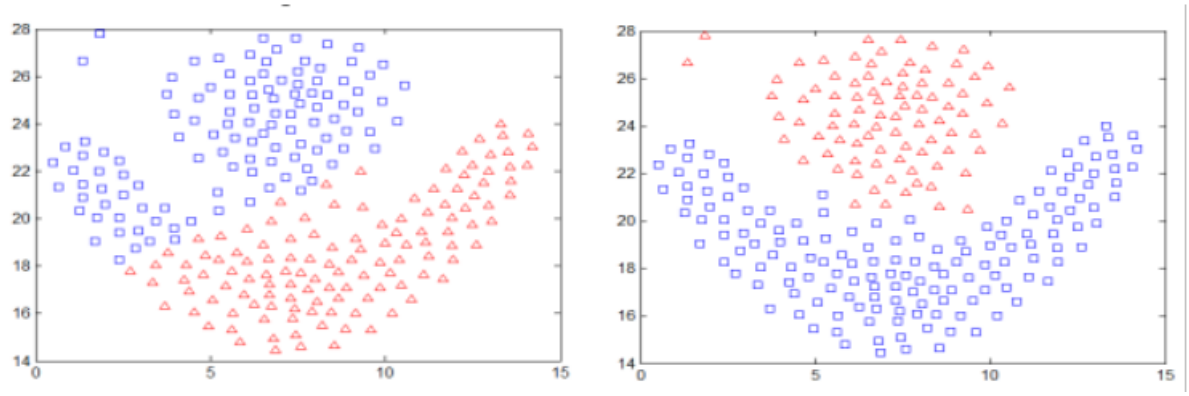

Fig 2: Clustering results of Flame dataset using KKZ_k-means (left) and FICA (right)

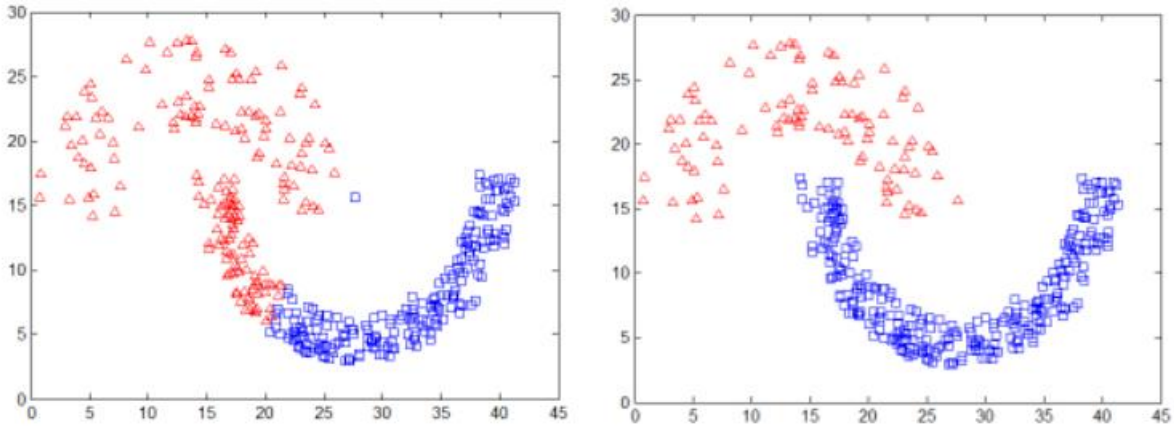

Fig 3: Clustering results of Jain dataset using KKZ_k-means (left) and FICA (right)

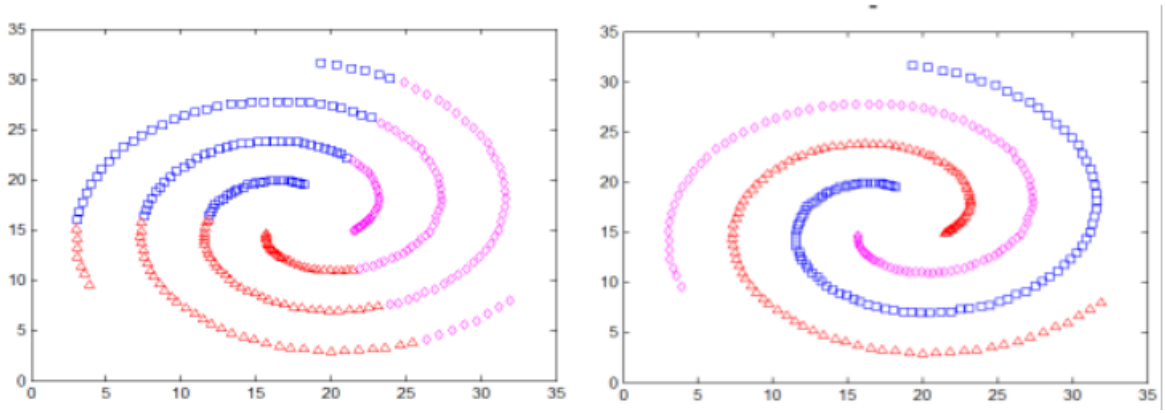

Fig 4: Clustering results of Spiral dataset using KKZ_k-means (left) and FICA (right)

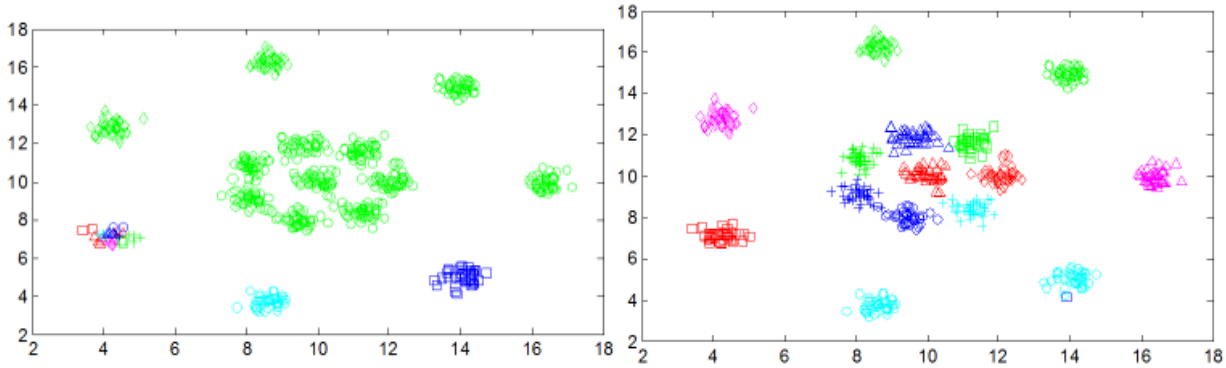

Fig 5: Clustering results of R15 dataset using KKZ_k-means (left) and FICA (right)

5. CONCLUSION

In this study, a fast incremental clustering algorithm was suggested. Experimental results have demonstrated that it is effective in finding consistent clustering results. This approach is an alternative to the k-means algorithm for producing better clustering with less computational time.

Future work includes testing a preprocessing step which consists to remove outliers from the input dataset, in order to produce more accurate clustering results. Another possible improvement will consist to consider a parallelization of this method, for faster clustering.

6. REFERENCES

[1] Ankerst, M., M. Breunig, H.P. Kriegel and J. Sander, 1999.OPTICS: Ordering points to identify the clustering structure. Proceeding of ACM SIGMOD International Conference Management of Data Mining, May 31-June 3, ACM Press, Philadelphia, United States, pp: 49-60.

[2] Aloise, D.; Deshpande, A.; Hansen, P.; Popat, P. (2009). "NP-hardness of Euclidean sum-of-squares clustering". Machine Learning 75: 245–249. doi:10.1007/s10994-009-5103-0.

[3] Lloyd, S.P., 1982. Least square quantization in PCM. IEEE Trans. Inform. Theor., 28: 129-136.

[4] MacQueen, J.B., 1967. Some Method for Classification and Analysis of Multivariate Observations, Proceeding of the Berkeley Symposium on Mathematical Statistics and Probability, (MSP'67), Berkeley, University of California Press, pp: 281-297.

[5] Katsavounidis, I., C.C.J. Kuo and Z. Zhen, 1994. A new initialization technique for generalized Lloyd iteration. IEEE. Sig. Process. Lett., 1: 144-146.

[6] [Asuncion, A. and Newman, D.J. (2007). UCI Machine Learning Repository [http://www.ics.uci.edu/~mlearn/MLRepository.html] Irvine, CA: University of California, School of Information and Computer Science.

[7] L. Kaufman and P. J. Rousseeuw. Finding groups in Data: "an Introduction to Cluster Analysis". Wiley, 1990.