APPLICATION OF MULTIPLE RANDOM CENTROID (MRC) BASED K-MEANS CLUSTERING ALGORITHM IN INSURANCE – A REVIEW ARTICLE

Sundari NallamReddy, Samarandra Behera, Sanjeev Karadagi, Dr. Anantha Desik

Tata Consultancy Services, Hyderabad, India

ABSTRACT:

In this paper we present a review of some applications of cluster analysis in the field of Insurance and allied sciences. Primarily there are two types of clustering techniques used in predictive analytics based on the business problems, Partition-based clustering technique and Hierarchical agglomerative clustering approach. Hierarchical agglomeration based clustering approach is time consuming and complexity increases with increase in number of dimensions. Partition based algorithms contrary to hierarchical tries to divide the search space before arriving at the final clusters. Both methods have its merits and demerits and hence proper knowledge of domain, number of variables and computation prowess is required before deciding on the algorithm.

Insurance industry is rich in data and attributes that can be used for data analytics are varied in nature. Hence, Hierarchical methods are generally not suitable for Insurance. K-means algorithms depending on partition-based clustering techniques are popular and widely used and applied to a variety of domains specifically in Insurance. However, K-means algorithms are extremely sensitive to the choice of initial centroid. Several different initialization approaches were proposed for the K-means algorithm in the last decades due to such sensitivity.

This paper proposes an iterative Multiple Random method for selection of initial cluster centroid in K-means clustering instead of the simple random seed methods. Performance assessment of the proposed initialization method over two different Insurance datasets with different dimensions of distance functions, numbers of observations, groups and clustering complexities are discussed in detail. The proposed algorithm is developed in-house using Java and results are compared with some of the standard available software. Results from two insurance datasets varying in business problems and attributes, clearly indicates that the proposed initialization method is more effective and converges to more accurate clustering results than those of the simple random initialization methods.

1. INTRODUCTION

This is an era of knowledge and information. There is virtually an explosion of information these days and fiercely competitive Insurance industry is definitely among the top user of data driven insights. In the digital journey Insurers are trying to understand competitive landscapes using deep insights of the customers. These insights are helping Insurance organizations to serve customers better and offer the right products at the right time thereby increasing profits. It is well understood by Insurer that, every customers need is different. Customer financial and investment needs are different and varies significantly. So, segment of customers, which is group of customers with similar Insurance need, differs from some other segments for the same Insurer. Insurance organization started their digital journey by trying to understand these subtle differences by applying advance analytics techniques.

Cluster analysis is the mostly used technique to segments the customers in all the business areas. Insurance business predominantly used this technique in underwriting, customer analysis, claim and marketing analytics. The application of hybrid methods i.e. analyzing data by applying predictive techniques inside each segments are relatively new but very powerful methodologies in Insurance. In view of this there is a lot of importance of cluster analysis in the insurance industry.

Clustering is an important unsupervised learning technique where a set of patterns, usually vectors in a multidimensional space, are used for identifying group of similar characteristics. Each group, called cluster, consists of vectors that are similar between themselves and dissimilar to vectors of other groups. However, cluster analysis is considered to be the most popular tool in statistical data analysis which is widely applied in a variety of scientific areas such as data mining, pattern recognition, geographic information systems, information retrieval, microbiology analysis and Insurance & Finance. There are several clustering techniques available and those are organized into the following categories as partitioning methods, hierarchical methods, density-based methods, grid-based methods, model-based methods, methods for high-dimensional data and constraint-based clustering [1]. Partition-based clustering [2] attempts to analyze data and assembles it in a set of groups separate from each other. This paper is limited to discussion about K-Means algorithm which comes under partitioning methods where number of clusters is known a priori. Also, data in Insurance varies a lot and hence parametric data analysis implementation requires a lot of data cleansing and normalization. Segmentation approaches provides clean data on which accurate predictive models can be built. K means algorithm also helps in restricting the search spaces to a user defined value i.e. K, which enables faster data analytics implementation contrary to Hierarchical methods [3] to [7].

This paper has been organized as following. We discuss the K-means clustering algorithm using the two distance function Euclidean & Manhattan, followed by discussion on the importance of Initial Centroid (I- Centroid) and different approaches for initial seed. This is succeeded by a discussion on proposed Multiple Random Centroid (MRC Method) and the algorithm based on it. Computational results based on developed Java program using the two initial seeds methods, comparison with standard software, performance were discussed subsequently followed by conclusion and future work.

2. K-means Clustering

K-means is one of the most famous partition clustering algorithms and considered the top ten most influential data mining algorithms. It is at the same time very simple and quite scalable, as it has linear asymptotic running time with respect to any variable of the problem. K-means clustering is a method of cluster analysis which aims to partition n observations (x1, x2… xn), where each observation is a d-dimensional real vector into k clusters in which each observation belongs to the cluster with the nearest mean. In general K-means is one of the most important and best performances of the clustering algorithms. However, there are some drawbacks for K-means algorithm like sensitivity to the initial cluster centroids which is addressed in this paper. Moreover, when the number of data points is large, it takes enormous time to find the global optimal solution.

K-means has several limitations which are listed below:

Scalability: It scales poorly computationally.

Initial means: The clustering result is extremely sensitive to the initial means. Noise: Noise or outliers deteriorates the quality of the clustering result.

Number of clusters: The number of clusters must be determined before the means clustering begins. Local minima: It always converges to local minima.

K-Means Algorithm:

K-Means is one of the fastest Algorithm which converges very fast, the algorithm begins with random initial centroid and keep reassigning the patterns to clusters based on similarity between the patterns and the cluster centroid until a convergence criterion will met after certain no. of iterations. The K-Means algorithm is popular because it is easy to implement and time complexity is O(n), where „n‟ is the number of patterns. The basic K- Means process as follows

- **Step1** Assign the number of clusters (k parameters in k-means).
- **Step2** K-means selects randomly k cluster centroids.
- **Step3** Assign objects to clusters based on distance function.
- **Step4** When all objects have been assigned, Re-compute new cluster centroids by averaging the observations assigned to a cluster.
- **Step5** Repeat (3-4) until convergence criterion is satisfied.

Pseudo code for K-means algorithm:

Require: $k \geq 2$ and $t \geq 1$, where k = no. of clusters & t = no. of iterations

```
Select initial cluster Centroids, c1, c2,..cl
Repeat
    For each point xj in a dataset do
        Fōr all  do
            Compute the dissimilarity d(xj,ci)
        End for.
        assign point xj to closest cluster Ci;
    End for.
    Fōr all  do
        Up̄date  as the centroid of cluster Ci;
    End for.
```

The implementation of K-means algorithm requires several important factors to be considered like Initial seed, distance function, normalization of the data and convergence/stopping criteria.

Importance of Initial seed:

Starting points in K-means algorithm has significant impact on the results. Choosing different starting point values lead to different clusters with different error values which effects K-means initialization process. Selection of the first centroids, which are far apart and falls into different clusters, has several benefits. It decrease amount of computation and optimize algorithm performance by minimizing the objective function of K-means algorithm, which leads to better results. Generally the process of initializing the centroids with any random seed and computing the clusters without seeing the closeness of these initial centroids to the data may lead to more computations and also may not provide good results. To come out of this challenge researchers generally use different random & heuristic process to compute the initial centroids. After the process of selecting the start point randomly; some calculations are performed to guess whether the point is suitable to be considered as a first initial centroid or not. Such decision is based on the process of computing distances between the selected centroid and other points within the dataset. The proposed algorithm uses "N‟ no. of random points and calculates the SSE (Sum of Squared Error) of random centroids from different points of data like, first, first quartile, median etc. The random point with maximum SSE is considered as the initial centroid to start the algorithm.

3. Proposed Seed Algorithm

1. Set seed
2. For i, 1 to n
3. Assign random seed to cj
4. SSE between the (cj, x) X is the seed from data
5. Consider Cj where SSE is higher.

Distance Functions:

Generally different distance functions used calculate the distances but the propose algorithm we have used Euclidean & Manhattan distances.

Euclidean

The Euclidean distance between two points in the plane with coordinates (x, y) and (a, b) is given by -

$$\text{dist.}((x, y), (a, b)) = \surd(x - a)^2 + (y - b)^2$$

Manhattan

The Manhattan distance (sum of the magnitudes of differences in each dimension) is given as-

$$\text{dist.}((x, y), (a, b)) = |(x - a)| + |(y - b)|$$

The developed program supports both the functions depending on the data and we have compared the results for these distance functions.

Normalization:

The proposed K-means algorithm runs only for numerical data, all the categorical variables that needs to be converted into numerical or binary variables. In addition to it the numerical variable needs to be standardized so as to bring all the variables in proportion to one another. This data standardization is carried out using Z- statistics otherwise distance computation will be difficult as there will be large variation in variables.

$$Z = \frac{\text{Value} - \text{Mean}}{\text{SD}}$$

Where Value is individual value of the variable for each observation, Mean is the average of that variable and

$$SD = \sqrt{\text{Variance}}$$

Stopping /Convergence:

There are different methods used for convergence of the algorithm like the difference between corresponding difference between the centroids to be small and we have used the below process for convergence no (or minimum) re-assignments of data points to different clusters, no (or minimum) change of centroids, or minimum decrease in the sum of squared error (SSE), Ci is the jth cluster, mj is the centroid of cluster Cj (the mean vector of all the data points in Cj), and dist(x, mj) is the distance between data point x and centroid mj.

We also used Sum square of group cluster (SBG), which is the sum of the squared distance of a cluster of a cluster centroid, Ci, to the overall means "C" of all the data points. The higher the total SBG of a clustering the more separated the clusters are from one another. The total SBG directly related to pairwise distance between the centroids.

4. Data Sets

The performance evaluation of the proposed initialization method is applied on two real world insurance datasets. These datasets are deliberately chosen from two entirely different business problem i.e. identifying new customers for Insurance products and Claims in General Insurance (GI) to test the robustness of the proposed algorithm. Testing the proposed algorithm in this way also provided confidence on the generalization of this algorithm to some other business problems. Dataset1is of 21 variables of 4000 records and dataset2 new business insurance file is of 12 variables of 10000 records. Furthermore, the performance of K-means algorithm with the proposed initialization method is evaluated using popular evaluation method such as Sum of Square Errors (SSE). The results of such evaluation are compared with K-means algorithm with other methods, also it is compared against SAS and some free software.

5. Experimental Results

The implementation of the proposed algorithm using multiple random initial centroids is compared with the using simple random initial centroids. Also this implementation is compared against the K-means results of the SAS Enterprise Miner (EM) and open source products. The experiment implementation is summarized in coding K-means algorithm and the process of selecting initial centroids of clusters using Java programming language and thus, the algorithm runs on any platform. Moreover, the algorithm allows the user to load many datasets and various file formats such as csv or sas data etc. Also various options like five initial centroid processes and two distance functions i.e. Euclidean and Manhattan are provided.

Table 1 shows the comparison of K-means performance results using the proposed method, initial centroid, SAS and open source centroids. It is observed that the cluster results are almost similar lines for other methods and SSE of proposed method are better than initial centroid methods. The MRC method cluster results percentage almost matches with the SAS & open source outputs. The SBG values are also better than the initial centroid method.

Table-1: Dataset-1 Percentage of observation in each cluster and SSE, SBG Values

Cluster No	MRC-Method		I-Centroid		SAS	Open Source	
	Euclidean	Manhattan	Euclidean	Manhattan	Euclidean	Euclidean	Manhattan
1	20.61	18.39	7.59	4.42	23.28	23.00	23.00
2	19.80	18.98	37.03	40.52	20.68	23.00	13.00
3	6.09	12.05	25.03	19.91	20.68	19.00	17.00
4	39.84	34.79	24.03	21.04	27.72	14.00	25.00
5	13.66	15.79	6.32	14.11	7.45	29.00	22.00
SSE	21600	21637	24155	22925	NA	6338	17095
SBG	3.7	3.7	5.2	7.1	NA	NA	NA

Table-2 is for validation of the MRC method taking three folds of different random data from the dataset-1 to test the algorithm performance. The results shows almost all the random samples the cluster percentages are similar and even SSE &SBG values are very small. The proposed method is hence found to be robust in terms of efficiency across various datasets.

Table-2: Three fold validation of the proposed algorithm

Cluster No.	First Sample		Second Sample		Third Sample	
	Euclidean	Manhattan	Euclidean	Manhattan	Euclidean	Manhattan
1	41.20	36.60	50.80	12.50	7.20	12.80
2	21.20	19.20	16.30	23.00	36.10	22.30
3	17.10	18.80	11.60	33.50	4.20	15.50
4	8.20	13.80	9.60	9.70	25.50	28.60
5	12.40	11.60	11.70	21.30	27.00	20.80
SSE	4303	4764	5296	5470	4456	5048
SBG	3.6	3.9	6.6	6.4	7.4	3.6

Table-3 depicts the implementation on a large dataset which provided better results considering the cluster % of initial method and also the results are similar to the output of SAS & open source software. The SSE values are far better than initial centroid methods. The proposed algorithm merged the cluster if the no. of data points is zero and in this case the algorithm considered only six clusters even though we run „7‟ cluster solution.

Table-3: Dataset-2 Percentage of observation in each cluster and SSE, SBG Values

Cluster No	MRC-Method		I-Centroid		SAS	Open Source	
	Euclidean	Manhattan	Euclidean	Manhattan	Euclidean	Euclidean	Manhattan
1	28.99	0.22	0.29	0.86	15.34	12.00	21.00
2	39.52	1.24	50.78	18.53	9.88	14.00	9.00
3	12.29	27.88	28.53	63.40	20.00	19.00	14.00
4	9.85	10.18	6.57	3.43	3.07	18.00	15.00
5	2.67	53.90	9.85	10.22	6.71	18.00	24.00
6	6.68	6.58	2.67	1.14	21.78	9.00	5.00
7	NA	NA	1.31	2.42	23.22	9.00	11.00
SSE	14882	15626	16727	18386	NA	7011	13178

Table - 4 provides the result from the validation of the MRC method on Dataset-2 taking three samples of data randomly from the dataset to test the algorithm performance. The results shows almost all the random samples the cluster percentages are similar and even SSE & SBG values are very small. The Euclidean distance cluster5 and SSE are better than Manhattan values. Again, the proposed method outperform in terms of efficiency and robustness across multiple samples.

Table-4: Dataset-2 Threefold validation of the proposed algorithm

Cluster No	First Set		Second Set		Third Set	
	Euclidean	Manhattan	Euclidean	Manhattan	Euclidean	Manhattan
1	10.35	14.70	8.80	6.35	46.70	9.95
2	24.80	11.10	52.50	55.20	6.90	2.05
3	3.40	22.50	2.70	2.00	9.35	0.50
4	6.50	1.35	28.05	26.40	28.10	81.10
5	54.30	43.50	1.10	9.75	1.50	6.00
6	0.65	6.35	6.85	0.30	6.45	0.40
7					1.00	
SSE	3123	2916	319	3128	2821	3320
SBG	24	42	34	44	24	62

6. Conclusion

Clustering is used in many fields such as data mining, knowledge discovery, statistics and machine learning. A good clustering algorithm produces high quality clusters to yield low inter cluster similarity and high intra cluster similarity. This paper presents a new way to select initial centroids using Multiple Random method in K- means algorithm. This initialization method is as fast and as simple as the K-means algorithms, which makes it attractive in practice. There are already so methods in practice for initial centroid selection. The main reason of this method is to make K-means less sensitive to the initialization process and to get consistent results every time algorithm runs. Experimental results demonstrate that the modification appears to give efficient performance when dealing with different insurance real-world datasets, and it is observed that the proposed method has substantially outperformed the standard K-means in terms of values & accuracy.

References

[1] T. Abraham and J. F. Roddick, "Survey of Spatio-Temporal Databases," GeoInformatica, vol. 3, March 1999.

[2] R. Maitra, "Initializing partition-optimization algorithms," IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 6, pp. 144–157, 2009

[3] Pena J., Lozano J. and Larranaga P., “An Empirical comparison of four initialization methods for the K-means algorithm”, Pattern Recognition Letters 20 (1999), 1027-1040.

[4] Ting Su and Jennifer Dy, "A Deterministic Method for Initializing K-means Clustering," Tools with Artificial Intelligence, 2004. ICTAI 2004. 16th IEEE International Conference, pp. 784 - 786, Nov 2004.

[5] S. Kalyani and K.S. Swarup, "Particle swarm optimization based K-means clustering approach for security assessment in power systems," Expert Systems with Applications, vol. 30, pp. 10839–10846, 2011.

[6] Mishra, S. K. and Raghavan, V. V., An empirical study of the performance of heuristic methods for clustering. In Pattern Recognition in Practice, E. S. Gelsema and L. N. Kanal, Eds. 425436, 1994.