International Journal of Information Retrieval Research
Volume 12 • Issue 1

Maxmin Data Range Heuristic-Based Initial Centroid Method of Partitional Clustering for Big Data Mining

Kamlesh Kumar Pandey, Dr. Harisingh Gour Vishwavidyalaya, India
https://orcid.org/0000-0002-7076-5979

Diwakar Shukla, Dr. Harisingh Gour Vishwavidyalaya, India

ABSTRACT

The centroid-based clustering algorithm depends on the number of clusters, initial centroid, distance measures, and statistical approach of central tendencies. The initial centroid initialization algorithm defines convergence speed, computing efficiency, execution time, scalability, memory utilization, and performance issues for big data clustering. Various researchers have proposed the cluster initialization techniques, where some initialization techniques reduce the number of iterations with the lowest cluster quality, and some initialization techniques increase the cluster quality with high iterations. For these reasons, this study proposed the initial centroid initialization based maxmin data range heuristic (MDRH) method for K-means (KM) clustering that reduces the execution times, iterations, and improves quality for big data clustering. The proposed MDRH method has compared against the classical KM and KM++ algorithms with four real datasets. The MDRH method has achieved better effectiveness and efficiency over RS, DB, CH, SC, IS, and CT quantitative measurements.

KEYWORDS

Big Data, Big Data Clustering, Computing Efficiency, Convergence Speed, Initial Centroid Algorithm, K-Means, K-Means++, MDRHK-Means, Scalability

INTRODUCTION

The rapid development of digital technologies had produced enormous amounts of data in a different format at high speed, such as social media. In Sep. 2019 (Viens, 2019), the monthly active users of Facebook was 2.4 billion that sent 41.6 million messages through Messenger in a minute, YouTube has 2 billion users that have watched 4.5 million videos per minute, Instagram has 1 billion users out of which 347,222 users scrolled the Instagram per minute, Twitter has 330 million users where 87500 users tweeted. All these social media describes how much data has been generated by the user in current ages with high speed. Digital technologies have changed the scales, formats, and speed of data production. For these reasons, the nature of the usual data changed to big data. The volume, variety, and velocity characteristics have defined the complex framework of big data. The volume characteristic defines Terabytes and Petabytes scaling, the variety defines heterogeneous data formats generated through the heterogeneous data sources, and the velocity defines the speed of data production and data analysis. Data volume is the base of big data that defines the massive data set. Here, this paper summarizes the volume, variety, and velocity characteristics basis of existing

DOI: 10.4018/IJIRR.289954

research (Hariri et al., 2019; Lee, 2017; Nada Elgendy & Elragal, 2014) as "volume depends upon variety, and variety depends upon velocity."

Recent researchers have suggested other characteristics of the big data as value (Oracle), veracity (IBM), variability (SAS), and visualization. Value is defining the valuable information from massive volume using constant attributes of the big data that describe the decision system (Hariri et al., 2019; Sivarajah et al., 2017). Veracity is determining the quality of the data as trustiness and accuracy during the data analysis, data storing and management, and heterogeneous sources. Variability is defining data structure, meaning, and behavior that changes from time to time due to rapid growth. Veracity is determining the accuracy of the decision-making system (Nada Elgendy & Elragal, 2014; Tabesh et al., 2019) and variability used in sentiment analysis (Gandomi & Haider, 2015; Sivarajah et al., 2017). Visualization characteristic visualizes the knowledge as user expectation, or unstable such as pictorial or graphical such as a table, graph, picture, statically, and so on. This paper summarizes the value, veracity, variability, and visualization characteristics of big data as " Veracity validates the accuracy basis of variety, the value identifies predicted value based on volume and variety, variability presents specific analysis tools based on the volume and variety, and visualization visualized the results and problems based on the volume, variety, and velocity."

Classical data mining algorithms use a centralized data source, but big data mining algorithms use distributed, centralized and a mixture of multiple sources. Multiple sources mining of big data could be grouped into four categories pattern analysis, classification, clustering, and fusion (Wang et al., 2018). The clustering process is the default data mining approach that labels data items without any prior knowledge basis of data similarity (Jain, 2010). For this reason, clustering is known as unsupervised learning. Data similarities define by the distance measures, where data similarities and variance of within-cluster are minimum, the data similarities of between-clusters are maximum. Classical clustering algorithms are facing various challenges due to data volume, variety, and velocity. The data volume is defining the computational cost, speed, efficiency, and scalability challenges of the classical clustering algorithms (Khondoker, 2018; Maheswari & Ramakrishnan, 2019). Big data clustering focuses on scale-up, speed-up, optimizing computation costs, and resources without the effect of cluster quality. The design of the big data clustering is dependent upon the single-machine and multiple-machine execution environment (Khondoker, 2018).

Clustering methods determine the data points and patterns as natural behaviors through the statistical classification techniques and the clustering process used for compression, natural classification, and underlying structure of the data. Data volume consists of data varieties such as structured, unstructured, and semi unstructured. Structured data have specific formats such as table, graph, vector, while unstructured data have no such format for example text, images, audio, video, and so on. The structured data have the semantic relationship between each object which is basic requirements for clustering. During the clustering process, data variety ignores the structure of the unstructured data and uses its feature vector. Unstructured data firstly converted into the pooled feature vector (Jain, 2010) or numerical data vector (Duwairi & Abu-Rahmeh, 2015), and after that become suitable for application of applies the clustering methods. This type of clustering is called multi-way clustering, co-clustering, bimodal clustering, etc. (Jain, 2010). Clustering algorithms are classified into partitional, hierarchical, distribution, probabilistic, model, density, grid, fuzzy, and graph clustering (K.K. Pandey & Shukla, 2019; Kamlesh Kumar Pandey et al., 2020).

The partitional clustering algorithms used in four phases for performing the clustering process and optimization of the relative objective function. The first phase selects the number of cluster K, and the second phase initializes the K cluster centroids basis of initialization methods. During the third phase, the distance measures find the nearest data points through the cluster centroid and data points of the dataset. The last step of the partitional clustering is to update the K centroid through central tendency based statistical approaches. The third and fourth phases use multiple times until the defined objective function has not to be optimized (Jain, 2010). This paper selects the K-Means (KM) to study cluster quality, execution time, speed up, memory utilization, and scalability under big

data mining setup considering the initial centroid initialization. The KM clustering is widely adopted for segmentation, text mining, bioinformatics, wireless sensor networks, financial discipline, data compression, texture segmentation, computer vision, vector quantization, etc (Pandove et al., 2018; Xie et al., 2019).

The result of the KM depends on the initial centroid because poorly initial centroid increases the number of iterations and troubles the local minima. This weakness is compensated by using a better initialization method or repeating the KM several times and extracting better results (Fränti & Sieranoja, 2019). The study contained in (Fränti & Sieranoja, 2018) described a better initialization approach that removes the weakness of well-separated clusters, improves cluster balancing without any number of K, and reduced the number of iterations. The content of (Peña et al., 1999) defines local minima, the number of K clusters and iterations are indirectly related to the initial centroid of the KM clustering. Better initial centroid points reduced the local optima and achieved the nearest optima (Duwairi & Abu-Rahmeh, 2015), improved accuracy (Arthur & Vassilvitskii, 2007), convergence speed (Duwairi & Abu-Rahmeh, 2015), and computing time (Fränti & Sieranoja, 2019).

The objective of this study is to reduce the iterations and execution time of the KM algorithm without effecting the cluster quality through the initial centroid. The first section elaborates big data clustering and fundamental issues of the KM clustering under data volume. Section 2 presents the related works of partitional clustering-based initial centroid initialization techniques and compares them for big data clustering basis of data size, convergence speed, computation efficiency, memory efficiency, scalability, and time complexity. In section 3, describe the KM objective function and proposed the MDRH initial centroid method for partitional big data clustering. The extensive experimental studies between the proposed method, classical KM, and KM++ algorithms with four real datasets respected to clustering objective, convergence speed, computing time in section 4. Finally, section 5 concludes the achievement of work, remarks, and future scope.

CENTROID CLUSTERING-BASED INITIALIZATION TECHNIQUES

Let $X = \left\{x_1,\ x_2,\ldots\ldots\ldots x_n\right\}$ dataset with d dimensional space and the partitioning clustering method clusters the X into K non-overlapping cluster $C = \left\{c_1,\ c_2,\ldots\ldots\ldots c_n\right\}$ basis of K clusters, initial centroid, distance measure, and statistical tools. The objective function of partitional clustering methods is to minimize the squared error of the cluster through the centroid statistical approach. The number of clusters always greater than two during the objective function minimization. If K is equal to one that is known as hard clustering, and K is equal to two, that clustering suffers the NP-hard problems. Partitional clustering algorithm satisfies convex, proportion, omission, and monotone properties (Jain, 2010; Pandove et al., 2018), and offering the drawn cluster are always non-empty (Arora & Chana, 2014).

The KM algorithm minimizes the sum-of-squared errors of numerical and non-numerical data, through grouping into K clusters (Fränti & Sieranoja, 2019). The KM algorithm is the second top algorithm in data mining and depends on the mean central tendency. During the KM clustering, first, select the K according to data labeling requirements. After that, select the K data objects as the initial centroid of the clusters using a random manner. Initialization of the K through initial centroid is known as the initialization step, and thereafter applies the update step through distance measure and mean statistical tool. Update step clustered the data items according to the minimum distance of the centroid and data items, and reassigned the centroid as a mean of the K clusters. In statistics, various distance measures are available for minimization of the SSE that is used inside the KM clustering. The Euclidean distance achieves the optimal distance in the least time with spherical or ball-shaped clusters. The Mahalanobis distance gives the hyperellipsoidal shaped cluster with higher computation time (Jain, 2010), the cosine measure gives spherical KM (Zahra et al., 2015), Gaussian mixture distance model generates a natural cluster (Fränti & Sieranoja, 2019). The update step repeats until the

previous and current means are the same. This stage terminates the clustering process and obtains the optimum objective function. Iteration of the update step depends on the initialization step, where good centroid gives the least iterations with the improvement of cluster results (Fränti & Sieranoja, 2019). The complexity of the KM depends upon the initialization and update steps (HajKacem et al., 2019).

The quality of the KM clustering improves via centroid initialization and updation (distance measures, termination criteria, statistical approach) steps (Celebi et al., 2013). This section investigates the related works of initial centroid methods and analyzed them for big data mining with respect to high volume computation related criteria. Initial centroid methods can use other partitional clustering algorithms such as Fuzzy c-means, K-Modes, K-Median, K-Medoids, and so on because initial centroid methods work independently.

Forgy (Forgy, 1965) proposed a random centroid-based method for the KM algorithm, in which each centroid of K is selected randomly. MacQueen (MacQueen, 1967) proposed a method, where first to select the K data points randomly, and map them into the rest of the data points for selecting the K centroid. The mapping process of the MacQueen method is based on data density. Authors of (Peña et al., 1999) described the random partition method, where the entire dataset is partitioned into K clusters through a random selection of data and started the updating process of the KM algorithm. The Random partition method avoids the worst-case centroid selection than random centroids based methods (Fränti & Sieranoja, 2019).

Kaufman et al. (Gentle et al., 1990) discussed the Kaufman method, where the first centroid of the cluster extracted through successive selection, and the rest of the centroid selected through the maximum distance and heuristic rule. The successive approach gives the most centrally located centroid of the dataset (Fränti & Sieranoja, 2019). According to the literature of (Fränti & Sieranoja, 2019), the random partition and Kaufman method act as a variant of the maxmin heuristic. Hierarchical clustering used the maximin heuristic for choosing the initial centroid that is known non-trivial solutions of random centroids based methods (Duwairi & Abu-Rahmeh, 2015).

Gonzalez (Gonzalez, 1985) proposed the approximation based maximin methods for minimizing the maximum intercluster distance that avoided the worst-case centroid selection of random centroids. The first cluster centroid selected through the arbitrary order, and the remaining cluster centroid selected through the maximum Euclidean distance between data points and previously selected centroid. Mirkin (Mirkin, 2005) discussed MaxMin heuristics, where the first cluster centroid selected through maximum distances, and the remaining cluster centroid selected by the minimum distances of the data points (Steinley & Brusco, 2007).

Bradley and Fayyad (Bradley & Fayyad, 1998) proposed the refinement based Bradley and Fayyad method (BFM), where the entire dataset partitioned into J subsets according to the random manner and these subsets clustered by using the MacQueen process. The contribution (Duwairi & Abu-Rahmeh, 2015) defines the J subsets drawn by the mixture model, joint probability distribution with maxima of probability that generalized the other iterative clustering methods. BFM produced high convergence speed and accuracy, but it doesn't offer a guarantee that the initial points are efficient, and the number of subsets expected. BFM method describes the high time complexity due to the high data points of refinement (Duwairi & Abu-Rahmeh, 2015). The contribution of (Bradley & Fayyad, 1998) defines the results of the BFM and random partition methods are similar in the majority of the dataset.

Arthur et al. (Arthur & Vassilvitskii, 2007) proposed the K-Means++ (KM++) method for increasing the speed and accuracy of the KM algorithm, where the first centroid selected through a uniform random process and other centroid chosen through $d\left(x\right)^2 / \sum_{x \in X} d\left(x\right)^2$ probability. In the probability formulation, $d\left(x\right)^2$ indicated the distance between previously selected centroid to other data points. The KM++ method identified the interpolates between the MacQueen and Maximin methods (Celebi et al., 2013). The Maxmin and KM++ method belongs to the furthest point heuristic

because each step takes the new centroid between the nearest (min) existing centroid and furthest (max) data points(Fränti & Sieranoja, 2019).

Various research contributions identified greedy methods for the optimization of the initial centroid. Some greedy based initial centroid approaches (Fränti & Sieranoja, 2019) are the greedy technique (He et al., 2004), subsampling (Celebi et al., 2013), and repeated strategy (Bradley & Fayyad, 1998). He et al. (He et al., 2004) used the greedy method for optimization of the cluster distance for achieving the cluster objectives through the KM clustering. Celebi et al. (Celebi et al., 2013) used the greedy method as a subsampling process through the KM++ algorithm for avoiding the similarity chances of two centroids. The greedy KM++ algorithm selects the centroid through probabilistically on each iteration after that greedy method selects those centroids which reduce the SSE. Bradley et al. (Bradley & Fayyad, 1998) used the repeat strategy for the refinement of the KM algorithm.

Sorting based initial centroid method select the K centroid after the sorting of the data points using some heuristics such as select first k points, select every N / k points, select K according to random or systematic sampling (Fränti & Sieranoja, 2019). Hartigan et al. (Hartigan & Wong, 1979) sort the data points firstly according to the data distance and select every N / k^{th} data point for the centroid of the clusters. The sorting distance finds through the centroid of the dataset to data points. Astrahan (Astrahan, 1970) suggested the nearest neighbor density-based heuristics rules for centroid initialization, where the select the high-density location data points as first centroid and other centroid selected through decreasing order of the density location data points.

The projection-based initial centroid method selects the K centroid after the projection of data points using some similar heuristics rules of sorting methods that are generally applicable to one dimension of data (Fränti & Sieranoja, 2019). The recent study (Sieranoja & Fränti, 2018) used random projection and two furthest points projection approach for centroid selection, where the random projection approach is much efficient. Random projection method projects the two random data points through the line passing, and after that uses the heuristics rules for centroid selection.

Su et al. (Su & Dy, 2007) suggested the PCA-Part (Principal Component Analysis Partitioning) and Var-Part (Variance Partitioning) methods for minimizing the SSE through the divisive hierarchical approach. The PCA-Part method finds the principal eigenvector through the covariance matrix, and the Var-Part method splits the covariance matrix according to a higher variance-based coordinate axis of data. The PCA-Part method had found better desirable for cluster quality and convergence speed, but its time complexity is higher than Var-Part.

Luxburg (Luxburg, 2010) proposed a density heuristics-based method for centroid selection, where cluster size and density correlated for the SSE minimization. This method draws many independent n size samples and uses the KM algorithm in each sample. Later combine the minimal matching distance cluster until the K clusters are found.

The split based initial centroid method combines all data points into one cluster and splits until the K clusters are found. The splitting process used standard deviation (Franti et al., 1997), bisecting KM (Steinbach et al., 2000), tri-level KM (Yu et al., 2017), and so on.

Pena et al. (Peña et al., 1999) empirically compared the Forgy (Forgy, 1965), MacQueen (MacQueen, 1967), Kaufman (Gentle et al., 1990), and Random Partition (Peña et al., 1999) initialization methods concerning the sensitivity of initial centroid points, cluster quality, and effect of convergence speed. This study found Kaufman and Random Partition methods outperformed as compared to other studied methods and encouraged the robustness and effectiveness of the KM algorithm results. This study described theoretical differences between the studied initial methods, where the Forgy and random partition methods are independent on the instance order, the MacQueen method dependent on instance order, and the Kaufman method used deterministic order. The MacQueen method shows the fastest converging approach of KM, and Kaufman describes the higher convergence speed than other studied methods. Pena et al. suggested the Kaufman method for KM as initial centroid because Kaufman method achieved excellent robustness, effectiveness, and convergence speed.

He et al. (He et al., 2004) reviewed and measured the random centroids, distance optimization, and density estimation for the initial centroid methods basis of the quantitative property. The Forgy (Forgy, 1965) and MacQueen (MacQueen, 1967) methods are categories under the random centroids, the Simple Cluster Seeking (SCS) (He et al., 2004) method and other greedy variant methods are categories under the distance optimization, and the Kaufman method (Gentle et al., 1990) and other Maximin variant methods are categories under the density estimation. This contribution observed that the initial centroid methods abandoned the cluster separation and considered cluster compaction during the KM algorithm optimization. He et al. found the convergence speed, cluster separation, and cluster compactness of random centroids are extremely poor compared to other studied methods. The Greedy and Maxmin methods achieved better convergence speed, cluster separation, and cluster compactness than other studied methods. The computational efficiency of Maximin has been found better than another greedy approach, but it needs more distance calculation.

Steinley et al. (Steinley & Brusco, 2007) examined Astrahan (Astrahan, 1970), BFM (Bradley & Fayyad, 1998), Continuous KM (Faber, 1994), Hand and Krzanowski (HKM) (Hand & Krzanowski, 2005), SPSS [30], Milligan ward (Milligan & Isaac, 1980), Mirkin maxmin heuristics (Mirkin, 2005), Mirkin Intelligent KM (Mirkin, 2005), PCA-part, SAS, Likas global KM (Likas et al., 2003), and Steinley KM (Steinley, 2003) initial centroid techniques for batch KM where some of the centroid techniques were based on agglomerative clustering. The Stanley KM (Steinley, 2003) method performs better than other studied methods based on numbers of clusters and variables, effects of variable and multidimensional, and cluster density factors. Steinley et al. observed the Steinley KM (Steinley, 2003) needs multiple repetitions for achieving the centroid, and agglomerative clustering works better if using the numerous clusters.

Celebi et al. (Celebi et al., 2013) compared the linear time complexity based initial centroid methods as Forgy (Forgy, 1965), MacQueen (MacQueen, 1967), Maximin (Gonzalez, 1985), BFM with J = 10 (Bradley & Fayyad, 1998), KM++ (Arthur & Vassilvitskii, 2007), Greedy KM++ (Celebi et al., 2013; Tou & González, 1974), Var-Part (Su & Dy, 2007), and PCA-Part (Su & Dy, 2007) respect to the cluster quality and speed criteria. Cluster quality criteria measured by the initial SSE, final SSE, normalized rand value, van Dongen, the variance of information, and clustering speed criteria measured by the number of iterations and CPU time. This study concludes that the Forgy, MacQueen, and Maximin methods perform worst in all analyzed criteria with slower convergence, and remaining methods achieved high convergence and effectiveness clustering results. Here, the PCA-Part and Var-Part methods define the high computational complexity and much-complication for implementation due to hierarchical formulation. The BFM and Greedy KM++ methods obtained a high convergence rate and effectiveness in the massive datasets. The PCA-Part and Var-Part methods are known as deterministic and other studied methods known as non-deterministic, where non-deterministic methods outperform respect to minimum statistic and deterministic methods outperform respect to mean and standard deviation statistic. Celebi et al. suggested the KM++ algorithm achieved better all discussed criteria except standard deviation statistics and BFM, Greedy KM++, Var-Part, PCA-Part used for approximate clustering.

Fränti et al. (Fränti & Sieranoja, 2019) studied the Random partition (Peña et al., 1999), Random Centroid (Forgy, 1965; MacQueen, 1967), Maxmin (Gonzalez, 1985), KM ++ (Arthur & Vassilvitskii, 2007), BFM (Bradley & Fayyad, 1998), Sorting (Hartigan & Wong, 1979) , Projection (Sieranoja & Fränti, 2018), Luxburg (Luxburg, 2010), and Split (Franti et al., 1997) initial centroid methods respect to the overlap and number of clusters, dimensions, and unbalance of cluster sizes using CI-values, success rates and iterations criteria. The CI values of the Random partition identified poorly, and Luxburg and Split identified strongly with high computational efficiency. The Luxburg and Split methods have achieved a high success rate and improved the computational space by using fewer iterations, but both methods identified unbalance in some computational criteria. This study found BFM achieved better overlap factor and worst unbalance cluster, the Luxburg method minor affected by a number of cluster criteria, and other studied methods obtained accurate dimensions factors. Fränti

et al. observed the KM ++ and Maxmin algorithms work efficiently in all computational criteria. For these reasons, the Fränti et al. suggested the KM ++ and Maximin initial centroid methods for KM clustering.

The author of (Fahad et al., 2014; Pandey et al., 2020) is determining the clustering algorithms for volume, variety, and velocity characteristics of the big data using some clustering algorithm characteristics. The volume considered dataset size, the sensitivity of outliers/ noisy and high dimensionality handling criteria, the variety considered a type of dataset and cluster shape criteria, and the velocity considered the performance of clustering algorithms such as time complexity, computing efficiency, scalability. The comparative analysis of (Celebi et al., 2013; Fränti & Sieranoja, 2019) discussed recent initial centroid methods of KM clustering and compare them through effectiveness and efficiency related measurement.

Based on the existing research perspective, and comparative analysis of (Celebi et al., 2013; Fränti & Sieranoja, 2019; He et al., 2004; Peña et al., 1999; Steinley & Brusco, 2007) finds which initial centroid methods are suitable for the KM algorithm. Here, this paper examines which initial centroid methods are achieved better performance under the big data environment through high volume data processing capability, convergence speed, time complexity, scalability, computation efficiency, memory efficiency factor. Pros and cons examinations of the initial centroid methods are shown in table 1 for big data clustering through the discussed literature and comparative analysis (Celebi et al., 2013; Fränti & Sieranoja, 2019; He et al., 2004; Peña et al., 1999; Steinley & Brusco, 2007) using random centroid, random partition, repeated heuristics, maxmin/distance optimization, greedy heuristics, sort heuristics, projection heuristics, density heuristics, and split heuristics categories.

Size of Data (DS) (Fahad et al., 2014; Fränti & Sieranoja, 2019)

This parameter identified data processing capability in the massive datasets for achieving the initial centroid. The better initial centroid method presents the high capacity for handling large data size because the size of the data affects the clustering quality and processing time.

Convergence Speed (CS) (He et al., 2004; Peña et al., 1999)

This parameter identified the number of iterations for achieving the centroid. The best initial centroid method presents the higher convergence speed rate and high convergence speed that are obtained by fewer iteration. Here the convergence speed of the initial centroid method affected the convergence speed of the KM algorithm.

Computation Efficiency (CE) (He et al., 2004; Peña et al., 1999)

This parameter identified the quality of centroid. The best initial centroid method presents high computation efficiency because better centroid gives robustness and better quality cluster, minimum SSE, and cluster objectives.

Memory Efficiency (ME) (Celebi et al., 2013; Fränti & Sieranoja, 2019)

This parameter identified the memory space and computational resources during the clustering. The better initial centroid methods used smaller consumption of memory space and computing resources because of the final memory efficiency of the KM algorithm is dependent upon the memory consumption of the initial centroid method.

Scalability (SB) (Fränti & Sieranoja, 2019; Steinley & Brusco, 2007)

This parameter identified the quality of initial centroid methods and it unaffected by data volume, dimensions, attributes, variables, number of clusters, and observation of dataset. The best initial centroid method presents high scalability.

Time Complexity (TC) (Fränti & Sieranoja, 2019; Peña et al., 1999)

This parameter identified the CPU time during the centroid achieved by initial centroid methods. The better initial centroid method presents the lower time complexity because the final time complexity of the KM algorithm depends upon the time complexity of the initial centroid method.

Table 1 identified the random centroid, random partition, and maximin/distance optimization family-based methods achieved better computing performance than other centroid methods. Based on the existing comparative research examination (Celebi et al., 2013; Fränti & Sieranoja, 2019; He et al., 2004; Peña et al., 1999; Steinley & Brusco, 2007), and table 1, the KM++ algorithm performs better than other random centroids, random partition, and other categories methods. The initial centroid of the KM++ algorithm minimized the local optima, CPU time, and increased the convergence speed and cluster quality of the KM algorithm concerning Forgy, MacQueen, random partition, and other centroid methods. According to this outcome, this paper takes KM++ and random centroid based classical KM for comparison to the proposed work.

PROPOSED INITIAL CENTROID INITIALIZATION METHOD

This section describes the clustering objective and presents the Maxmin Data Range Heuristic (MDRH) initial centroid initialization technique for the KM algorithm under the big data mining using the single machine execution. Here, the heuristic term defines the rules of the proposed method. The proposed work increased the convergence speed, speed-up, and removed the worst case of local optima without the effect of cluster quality and objective.

Objective Function

The objective of the KM is minimizing the SSE of all K clusters through the iterative process. Formally, the objective function of the KM defined as Eq 1(Fränti & Sieranoja, 2019; Jain, 2010).

$$SSEJ\left(X,C\right)=\sum_{k=1}^{K}\sum_{x_i\in C_k}x_i-\mu_k^{\,2} \tag{1}$$

Where x_i is the data points, and μ_K is the centroid of C_K cluster. The contains of C_K is restrain the minimum SSE which defined as follows (Xiao & Yu, 2012).

$$C_k=\left\{x_i\in X\mid k=\underset{j\in\{1,2,\ldots K\}}{\arg\min}\,x_i-\mu_j^{\,2}\right\} \tag{2}$$

$$\mu_k=\frac{\sum_{x_i\in C_k}x_i}{\left|C_i\right|} \tag{3}$$

Table 1. Comparative analysis of centroid initial methods for big data

Centroid Algorithms	DS	CS	CE	ME	SB	TC
Random Centroid Methods						
Forgy	Large	High	Low	Better	High	$\ddot{Y}(N)$
MacQueen	Large	Medium	Low	Better	High	$\ddot{Y}(N)$
Faber Continuous KM	Large	Medium	Medium	Better	Medium	$\ddot{Y}(N)$
Random Partition Methods						
Random partition	Large	High	Low	Better	High	$\ddot{Y}(N)$
Steinley KM	Large	Low	High	Medium	High	$\ddot{Y}(\mathrm{K}N)$
Repeat based heuristics Methods						
BFM	Medium	Medium	High	Medium	Medium	$\ddot{Y}(\mathrm{K}N + RK^2)$
HKM	Medium	Medium	High	Medium	Medium	$\ddot{Y}(\mathrm{K}ND + RK^2)$
Maxmin/distance optimization heuristics Methods						
Gonzalez maxmin	Large	Medium	High	Medium	Medium	$\ddot{Y}(\mathrm{K}N)$
Kaufman maxmin	Large	Medium	High	Medium	Medium	$\ddot{Y}(N^2)$
Mirkin maxmin (IKM)	Medium	Low	Medium	Medium	Medium	$\ddot{Y}(\mathrm{K}N)$
KM++	Large	Medium	High	Medium	High	$\ddot{Y}(\mathrm{KN})$
Likas Global KM	Medium	Low	Medium	Medium	Medium	$\ddot{Y}(\mathrm{K}N^2)$
Greedy based heuristics Methods						
SCS	Medium	Medium	High	Medium	Medium	$\ddot{Y}(\mathrm{K}N)$
Greedy KM++	Medium	Medium	High	Medium	Medium	$\ddot{Y}(\mathrm{K}N)$
Sort based heuristics Methods						
Hartigen sort	Large	Medium	Medium	Medium	High	$\ddot{Y}(N\,log\,N)$
Astrahan sort	Large	Medium	Medium	Medium	High	$\ddot{Y}(N^2)$

continued on followng page

Table 1. Continued

Centroid Algorithms	DS	CS	CE	ME	SB	TC
Projection-based heuristics Methods						
Random projection	Medium	Medium	Medium	Medium	Medium	$\ddot{Y}\left(N\log N\right)$
PCA-Part	Medium	Low	High	Medium	Medium	$\ddot{Y}\left(\mathrm{N}d^{2}\mathrm{K}\right)$
Var-Part	Medium	Low	High	Medium	Medium	$\ddot{Y}\left(NKd\right)$
Density-based heuristics Methods						
Luxburg	Large	Medium	High	Medium	Medium	$\ddot{Y}\left(\mathrm{K}N\log N\right)$
Split based heuristics Methods						
Standard deviation split	Medium	Medium	Medium	Medium	High	$\ddot{Y}\left(N\log N\right)$
Bisecting KM	Medium	Medium	Medium	Medium	Medium	$\ddot{Y}\left(N^{2}\right)$
Tri-level KM	Medium	Medium	Medium	Medium	Medium	$\ddot{Y}\left(NKdI\right)$
Milligan ward	Medium	Medium	Medium	Medium	Medium	$\ddot{Y}\left(\mathrm{K}N^{2}\right)$

(N=Size of Data Set, K=Number of Clusters, R= Number of repeats, D= Cluster separation, d= Number of dimensions, I= Number of Iterations)

Unfortunately, the optimal solution of objective function converges to the local minima that are related to the NP-hard (Jain, 2010; Xiao & Yu, 2012). The KM algorithm used the greedy approach through various iterations for obtaining the objective of KM clustering. The proposed method reduced the iterations and abolished the worst case of the local minima during the acquiring of the KM objective.

The research (Arthur & Vassilvitskii, 2007; Celebi et al., 2013; Fränti & Sieranoja, 2019; Peña et al., 1999) observed the optimal initial centroid achieved the minimum SSE and the minimum number of iterations through multiple runs of the KM algorithm. This study identified the first iteration can minimize the SSE then certainly as soon as achieved the minimum SSE in fewer iterations, and it is possible through prior knowledge of the data range and their corresponding maxmin distance. The SSE of the first iteration depends on the initial centroid of the cluster. The proposed method finds the initial centroid of the clusters through the centroid points of the dataset according to the data range. Normalization of the dataset centroid is minimizing the worst case of the local minima and achieving near SSE to the minimum SSE in the first iteration. The objective function of the first iteration is defined as Eq 4 with the condition of Eq 5 based on converges of the KM algorithm (Kalyanakrishnan, 2017).

$$SSEJ\left(X,C\right)\cong \min SSEJ\left(X,C\right) \tag{4}$$

$$SSEJ\left(X,C\right)\geq SSEJ\left(X^{t+1},C^{t+1}\right) \tag{5}$$

Where X^{t+1} is the data member and C^{t+1} is the centroid of the next iteration of the KM, the current iteration of the KM algorithm gave minimum or near SSE as compared to the previous iteration of SSE. Through this concept can minimize the iteration of KM and reduce the KM execution CPU time by selecting the accurate centroid of the dataset.

Proposed Initial Centroid Method

The proposed algorithm is inspired by the stratification theory of the stratified sampling process that improves the computing efficiency and reduces the computational cost of any algorithm. In stratification, entire data objects are partitioned into respective homogenous subsets and after that extract the appropriate knowledge in each subset. To achieve the initial centroid of the cluster, firstly create the K homogenous subsets using the maxmin data range heuristic, and hereafter extract the knowledge as means of each K subset for K initial centroid.

The proposed MDRH method is to use the maxmin data range heuristics for creating K homogeneous data groups. A better maxmin data range heuristic has described the high density and compaction inside each group. The proposed MDRH method uses the three phases for extracting the initial centroid of the KM algorithm. The first phase finds the K centroid of the entire dataset using max and min range heuristic of the data points, the second phase initializing the whole data into K groups using maximin distance heuristic, and the last phase finds mean of K groups. The combination of the three phases is known as the heuristic that defines the rule of finding the initial centroid. The mean of each group used as the initial centroid of the cluster. The complexity of the proposed method is equal to one iteration of the KM algorithm and smaller than the KM++ algorithm.

The final step of the proposed algorithm is validating quality measures of the MDRH method/ algorithm. The quality measures show that any two mean values are equal or zero then the data set has required deep preprocessing. The most observed reasons are that datasets have hidden noise, data has wrongly grouped, the number of clusters is highly or least selected, or any means value finds most negative and so on. The proposed MDRH method shown in algorithm 1 which finds the C_e initial centroid of the cluster, where step 1 to 13 shows the maxmin data range heuristic. After that proceed the standard KM with dataset X, number of K clusters, and initial centroid C_e.

The proposed MDRH method works as universally for any partitional clustering on the replacement of mean statistics as the other center tendency statistics. In this study, the paper is considering the KM algorithm, where center tendency uses the mean of the clusters, that is the reason initial centroid used the mean of the group. The proposed MDRH method to resolve the worst-case performance of KM using mean as an initial centroid.

EXPERIMENTAL ANALYSIS

Any experimental analysis validated any research work and based on the computing environment, dataset, existing algorithm, evaluation criteria, and results. This section computes the computing performance of MDRH based KM based on the cluster internal and efficiency-related measurement.

Experiment Environment and Dataset

The MDRH based K-Means (MDRHKM) algorithm is implemented through the Jupyter notebook framework with python computing tool, Intel I3 processor, 320 GB hard disk, 4 GB of main memory, Windows 7 Operating System. All experiments carried on four real data sets (https://archive.ics.uci.edu/ml/datasets.php) under the single machine execution. The characteristics of the experimental data sets shown in table 2.

Algorithm 1. Maxmin Data Range Heuristic initial centroid initialization (MDRH) method

Input:

1. $\mathrm{X} = \left\{\mathrm{x}_1, \ \mathrm{x}_2, \ \ldots\ldots\ldots \mathrm{x}_\mathrm{n}\right\}$ data points with d dimensions

2. K= Number of cluster/ required number of center

Output:

3. $C_e = \{C_{e1}, C_{e2} \ldots\ldots C_{ek}\}$ of the initial centroid of the cluster

Method:

Find the centroid of the entire dataset

4. $min = min\left(X_i\right), 1 \le i \le d$

5. $max = max\left(X_i\right), 1 \le i \le d$

6. $v = \left(\max - \min / \left(K + 1\right)\right)$

7. $if\, v \ge max$

8. Dataset has high variance noise data point and exit()

9. else

Find the centroid of the K group

10. $c_1 = \min + v$

11. $c_2 = c_1 + v$

12. $c_k = c_{k-1} + v$

13. End if

14. for i=1 to length(X)

Initialized group data member

15. $\mathrm{dis}_{\mathrm{euclidean}}\left(X_i, c_k\right) = \sqrt{\left|X_i - c_k\right|^2}$

16. Assign near distance on the closed $C_k = \left\{\mathrm{C}_1, \ \mathrm{C}_2, \ \ldots\ldots\ldots \mathrm{C}_\mathrm{K}\right\}$ group

17. end for

Find the initial centroid of each cluster

18. $C_e = mean\{C_1, C_2 \ldots\ldots C_K\}$

19. $if\, any mean\left(C_K == C_{K-1}\right) or any\ \left(C_e == 0\right) then$

20. Dataset has highly noise data point and wrongly clustered and restart process to step 4

21. else

22. $return\left(\ C_e = mean\ \{C_1, C_2 \ldots\ldots C_K\}\right)$

Table 2. Characteristics of the experiments data sets

Datasets	Objects	Attributes	Area
Corel Image Feature	68,040	17	Image
Person Activity	1,64,859	8	Life
3D Road Network	4,34,873	4	Computer
Geo-magnetic	58,374	10	Computer

Compared Initial Centroid Algorithms

Based on comparative analysis (Celebi et al., 2013; Fränti & Sieranoja, 2019; He et al., 2004; Peña et al., 1999; Steinley & Brusco, 2007) and examination of table 1, this paper compared the proposed method against the standard random KM (Forgy, 1965; Jain, 2010) and KM++ algorithm (Arthur & Vassilvitskii, 2007; Fränti & Sieranoja, 2019) based on efficiency and effectiveness related measures because both methods achieved better performance in the massive dataset and suitable for the big data clustering.

Evaluation Criteria

Evaluation criteria measure the performance of any proposed work through effectiveness (internal and external measure) and efficiency (speed) criteria. The objective of clustering is validated through internal measurement techniques because internal techniques did not require any external information for clustering validation. In this study, the paper used R square, Davies Bouldin score, Calinski Harabasz score, Silhouette coefficient as internal validation (Aggarwal & Reddy, 2013; Gan et al., 2007), and the number of iterations, CPU time as efficiency (speed) validation (Celebi et al., 2013; Peña et al., 1999; Zahra et al., 2015). Better-resulted value of R square, Calinski Harabasz score, and Silhouette coefficient are each time maximized, and the Davies Bouldin score, number of iterations, and CPU time are each time minimized.

- **R square (RS):** RS is validating the degree of difference between the clusters. This measure finds the ratio of the sum of squares between (SSB) and sum of squares total (SST).

$$RS = \frac{SSB}{SST} \tag{6}$$

- **Davies Bouldin score (DB):** DB measures the average similarity of each cluster and validates the separation between the clusters. The cluster similarity is the ratio of within to between-cluster distances. The DB score validates the clustering algorithm without depending on the number of clusters.

$$DB = \frac{1}{k}\sum_{i=1}^{k} R_i \tag{7}$$

$$R_i = max_{i \neq j} \frac{within_i + within_j}{between_{ij}} \tag{8}$$

$$within_j = \frac{1}{|c_j|}\sum_{i=1}^{|c_j|} \left\| x_i - c_j \right\|^2 \tag{9}$$

$$between_{ij} = \left\| c_i - c_j \right\|^2 \quad (10)$$

In the DB formulation, k is the total number of clusters, $\left|c_j\right|$ defines the total number of data point x_i inside of c_j cluster and c_i is another cluster.

- **Calinski Harabasz score (CH):** CH measures the variance of the cluster and validates the clustering performance through the average sum of squares value of between-and-within the clusters.

$$CH = \frac{\left(n-k\right)tr\left(B\right)}{\left(k-1\right)tr\left(w\right)} \quad (11)$$

$$tr\left(B\right) = \sum_{i=1}^{k} \left|c_i\right| \left(m_i - m\right)^T \left(m_i - m\right) \quad (12)$$

$$tr\left(w\right) = \sum_{i=1}^{k} \sum_{x \in c_i} \left(x - m_i\right)^T \left(x - m_i\right) \quad (13)$$

In the CH formulation, n is the total number of data points, k is the total number of clusters, x is data points inside the c_i cluster, m is mean of the entire dataset, and m_i is the mean of c_i cluster.

- **Silhouette coefficient (SC):** SC measures the similarity within the cluster and validates the clustering performance based on the pairwise difference of the cluster compactness and separation of the clusters.

$$S = \left\{ \sum_{x \in Ci} \frac{b\left(x\right) - a\left(x\right)}{\max\left[b\left(x\right),\ a\left(x\right)\right]} \right\} \quad (14)$$

In this formula *a(x)* is the average distance of *x* to other data points in the same cluster *C*, *b(x)* is the average distance of *x* to other data points in all *Ci* clusters.

- **Number of iterations (IS) :** IS measures the convergence speed of algorithm/model and validates the efficiency by independent of the compiler, implementation style, and CPU architecture. The number of iterations is obtained through the execution of the KM algorithm because the KM algorithm executes multiple times until the requirements have reached.
- **CPU time (CT):** CT measures the total computation time of any algorithm/model. The CT is defined by between the entry EN_T and exit EX_T time of any data mining algorithm.

$$ET = EX_T - EN_T \quad (15)$$

Results and Discussion

The code of the KM, KM++, and MDRHKM algorithms are written in the Jupyter notebook using the python language. Efficiency and effectiveness related results shown in table 3-4 and reported results of each evaluation measure are showing the average value of ten trials. Here is the number of clusters fixed at three for experiments according to the dataset. For this cluster setup, the Corel Image Feature, 3D Road Network, and Geo-magnetic dataset results are closed to the optimal value, and the Person Activity dataset resulted in little afar from accurate value. This paper used the predefined python library function DB, CH, and SC internal measurements and technical code of RS, IS, and CT for clustering algorithm evaluation. The optimal value of each measure is marked as bold in table 3-4, where the optimal values of RS, CH and SC are required maximization, and DB, IS and CT are required minimization. The minimization of IS and CT is showing the speed-up of any clustering algorithms. Comparative analysis of all measurements shown in fig 1-8 using a line chart, where measure value organized as ascending order.

Table 3 summarizes the proposed MDRHKM methods using optimum iterations and CPU time that removes the worst-case efficiency situation of the KM and KM++ algorithms. Table 4 summarizes the proposed MDRHKM method is to improve the cluster quality and remove the worst-case cluster effectiveness of the KM and KM++ algorithms.

Experimental observations of the Corel Image Feature dataset, the values of RS and DB have found similarities inside the KM, KM++, and MDRHKM algorithms. The CH and SC values of the MDRHKM algorithm are found better as compared to the KM and KM++ algorithms, but they are approx equal. The MDRHKM algorithm minimized the IS value as 71.22% and 71.32% and reduced the CT value as 76.43% and 77.98% respect to the KM and KM++ algorithms.

Experimental observations of the 3D road network dataset, the values of RS, DB, CH have found similarities inside the KM, KM++, and MDRHKM algorithms. The SC value of the MDRHKM algorithm is achieved better than KM and KM++ algorithms. The MDRHKM algorithm minimized the IS value as 38.77% and 18.91% and reduced the CT value as 48% and 53.17% respect to KM and KM++ algorithms.

Experimental observations of the Geo-magnetic and Person Activity dataset, the reported values of RS, DB, CH, SC have found better inside the MDRHKM algorithm than KM, and KM++ algorithms, where the performance of the KM++ algorithm is achieved worst. The IS and CT of

Table 3. *Comparative average analysis of efficiency ($means \pm std$) over 10 trials*

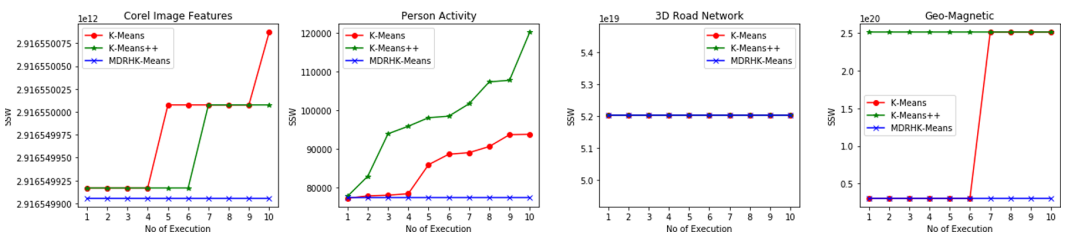
Dataset	Criteria	KM	KM++	MDRHKM
Corel Image Feature	IS	27.8 ± 4.13	27.9 ± 3.31	8 ± 0.0
	CT	172.16 ± 20.67	184.31 ± 55.62	40.57 ± 8.96
Person Activity	IS	3.3 ± 1.33	3.9 ± 1.28	3 ± 0.0
	CT	42.52 ± 18.60	67.91 ± 12.88	31.60 ± 3.55
3D Road Network	IS	9.8 ± 2.48	7.7 ± 1.33	6 ± 0.0
	CT	331.15 ± 131.11	364.58 ± 59.12	170.73 ± 24.10
Geo-magnetic	IS	4.8 ± 2.85	8.1 ± 0.31	2 ± 0.0
	CT	22.33 ± 13.73	30.68 ± 10.87	8.11 ± 0.79

Table 4. *Comparative average analysis of internal measures ($means \pm std$) over 10 trials*

Dataset	Criteria	KM	KM++	MDRHKM
Corel Image Feature	RS	0.88889 ± 0.0	0.88889 ± 0.0	0.88889 ± 0.0
	DB	0.5 ± 0.0	0.5 ± 0.0	0.5 ± 0.0
	CH	272147.83 ± 0.0061	272147.83 ± 0.0049	272147.84 ± 0.0
	SC	0.5898 ± 0.0021	0.58877 ± 0.00277	0.58998 ± 0.0035
Person Activity	RS	0.57139 ± 0.034	0.50572 ± 0.061	0.61089 ± 0.0
	DB	1.22994 ± 0.11	1.24879 ± 0.16	1.08436 ± 0.0
	CH	111005.24 ± 15578.52	86749.70 ± 21843.74	129410.055 ± 0.0
	SC	0.27932 ± 0.3058	0.29581 ± 0.03748	0.3123 ± 0.0021
3D Road Network	RS	0.91605 ± 0.0	0.91605 ± 0.0	0.91605 ± 0.0
	DB	0.39505 ± 0.0	0.39505 ± 0.0	0.39505 ± 0.0
	CH	2372570.77 ± 0.0	2372570.77 ± 0.0	2372570.77 ± 0.0
	SC	0.6809 ± 0.0027	0.68096 ± 0.0031	0.68187 ± 0.0030
Geo-magnetic	RS	0.9372 ± 0.06036	0.86707 ± 0.0	0.98396 ± 0.0
	DB	0.28651 ± 0.09475	0.39661 ± 0.0	0.21312 ± 0.0
	CH	1150346.40 ± 82621.58	190363.67 ± 0.0	1790334.89 ± 0.0
	SC	0.80862 ± 0.11064	0.68096 ± 0.002	0.89567 ± 0.002

the MDRHKM algorithm are found better than KM and KM++ algorithms in both data sets. Inside the Person Activity dataset, the MDRHKM algorithm minimized the IS value as 9.03% and 23.07% and reduced the CT value as 25.68% and 53.46% respect to KM and KM++ algorithms. For the Geo-magnetic data set, the MDRHKM algorithm minimized the IS value as 58.33% and 75.30% and reduced the CT value as 63.69% and 73.57% respect to KM and KM++ algorithms.

Figure 1. Analysis of SSW on each trial



The observation of table 3-4 and above discussion shows the proposed MDRHKM algorithm achieved better clustering effectiveness and efficiency for the massive dataset as compared to the KM and KM++ algorithms. For these reasons, the proposed MDRHKM algorithm eliminated the chance of local minima, and increases the speed up and scalability of the KM and KM++ algorithms. A good clustering algorithm defines the sum of squares within-cluster is all the time minimum, and the sum of squares between clusters is all the time maximum. The internal clustering measurement depends upon the SSW, SSB, and SST that reasons these terms describe the variance, homogeny, compaction, separation, similarity, and dissimilarity of the cluster.

Fig 1 and Fig 2 shows the SSW and SSB of the experimental data sets for using the KM, KM++, and MDRHKM algorithms in each trial. The experiment dataset of 3D Road Network gave similar SSB and SSW in each trial for using KM, KM++, and MDRHKM algorithms. Inside the Corel Image Feature, Person Activity, and Geo-magnetic data set, the proposed MDRHKM algorithm gives minimum SSW and maximum SSB in each trial. Observation of fig 1-2 defines the MDRHKM algorithm is to find better compaction and separation of the cluster and takes a guarantee for all the time finds the best case of SSB and SSW than the KM and KM++ algorithms.

Fig 3-6 shows the IS, CT, DB, and SC scores of the KM, KM++, and MDRHKM algorithms for using experimental data sets in each trial, where the proposed algorithm eliminates the worst IS, CT, DB, and SC scores of the KM and KM++ algorithms. The observation of fig 3-6 defines the proposed MDRHKM algorithm achieved close results to the best case of KM, and KM++ algorithms, where the proposed algorithm minimized the IS, CT, and DB, and maximized the SC in each trial than KM, and KM++ algorithms.

Fig 7-8 shows the RS and CH score of the KM, KM++, and MDRHKM algorithms for using experimental datasets in each trial. The observation of the fig 7-8 defines the proposed MDRHKM algorithm finds better within-similarities and between-dissimilarities of the cluster and finds the best case of RS and CH score.

CONCLUSION

This paper discussed various KM based initial centroid methods concerning the big data characteristics through existing research examination and proposed the initial centroid based MDRH method for resolving the worst-case situation of the KM and KM++ algorithms. The existing initial centroid algorithms do not present the optimal cluster quality and speed due to a random selection of the initial centroid. The proposed method presents the optimal cluster quality, iterations, and execution time because the MDRHKM method used constant iterations through the data range heuristic. The experimental study of this paper is based on the clustering objective and analyzed the cluster quality through the internal effectiveness and efficiency measurement that shows the proposed MDRHKM method achieved better effectiveness and efficiency as compared to the KM and KM++ algorithms. This paper observed during the KM and KM++ algorithm's execution, when the KM and KM++ algorithms have used fewer iterations then their CPU time and cluster quality are reduced, respectively. In these situations, the proposed method always used constant iterations, least CPU time, and higher cluster quality. The results of RS, DB, CH, and SC effectiveness validation are shown the MDRHKM method is eliminates the worst case of clustering results than the KM and KM++ algorithms. The results of IS and CT efficiency validation are shown the MDRHKM method is increased the speed-up, scale-up, convergence speed, and utilized memory resources than the KM and KM++ algorithms. This indication shows the proposed method is straightforwardly scalable under big data mining and reduced the local optima problem. During the experimental analysis, this paper observed high variance, and a high noise dataset does not achieve the accurate centroid through maxmin heuristic. Therefore, some of the groups are found empty for that reason, the initial centroid is not forwarding the KM clustering. Further scope of this research is to resolve the high variance related problem

Figure 2. Analysis of SSB on each trial

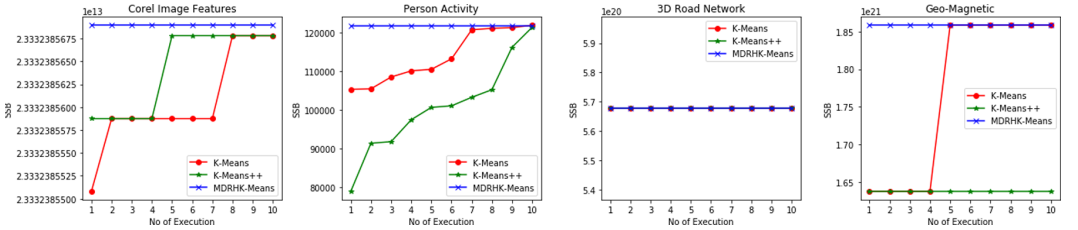


Figure 3. Analysis of IS on each trial

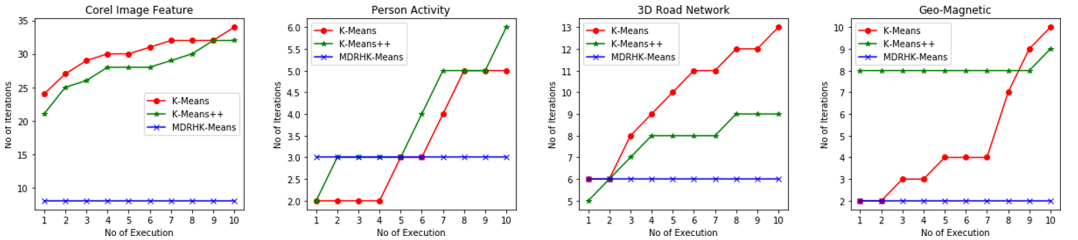


Figure 4. Analysis of CT on each trial

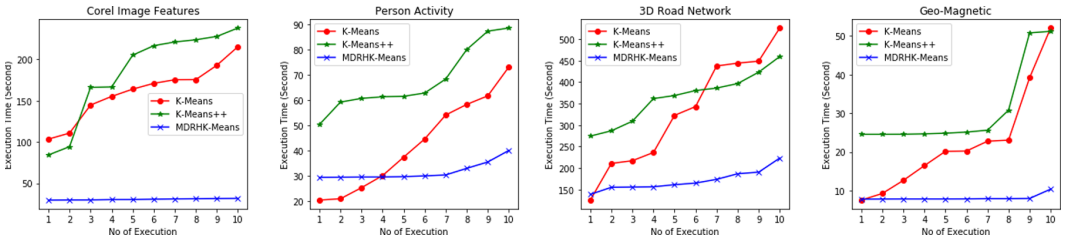


Figure 5. Analysis of DB on each trial

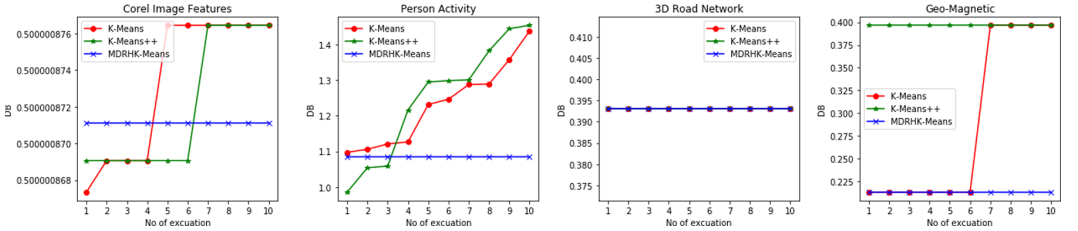


Figure 6. Analysis of SC on each trial

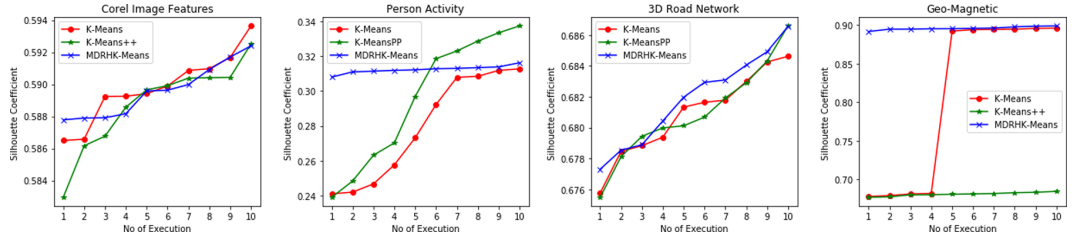


Figure 7. Analysis of RS on each trial

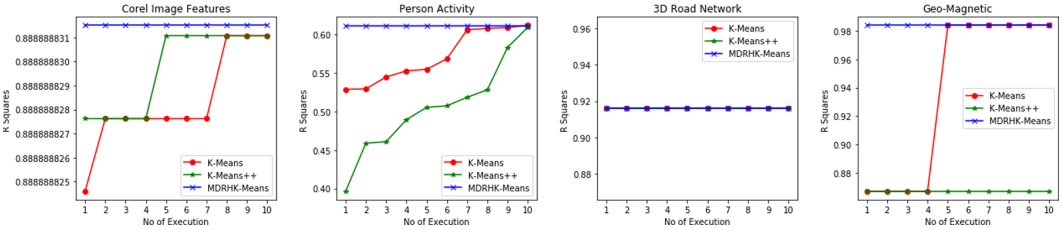
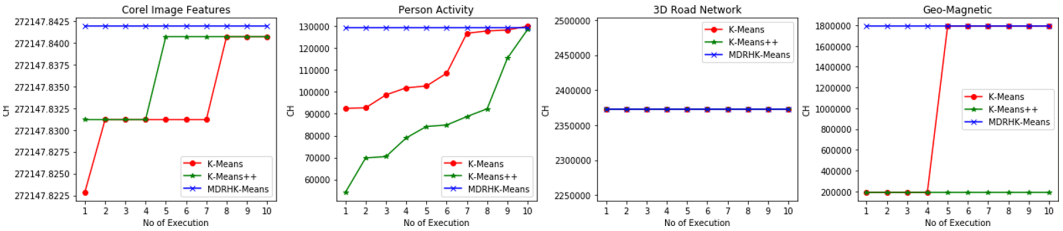


Figure 8. Analysis of CH on each trial



through the multiple machine-based technologies such as Hadoop and Spark using other internal and external measurements.

FUNDING

No Funding

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest.

REFERENCES

Aggarwal, C. C., & Reddy, C. K. (2013). DATA Custering Algorithms and Applications. Academic Press.

Arora, S., & Chana, I. (2014). A survey of clustering techniques for big data analysis. *Proceedings of the 5th International Conference on Confluence 2014: The Next Generation Information Technology Summit*, 59–65. doi:10.1109/CONFLUENCE.2014.6949256

Arthur, D., & Vassilvitskii, S. (2007). k-means++: The Advantages of Careful Seeding. *SODA '07: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, 1027–1035.

Astrahan, M. M. (1970). *speech analysis by clustering, or the hyperphoneme method*. Academic Press.

Bradley, P. S., & Fayyad, U. M. (1998). Refining Initial Points for K-Means Clustering. *15th International Conference on Machine Learning (ICML98)*, 1–9.

Celebi, M. E., Kingravi, H. A., & Vela, P. A. (2013). A comparative study of efficient initialization methods for the k-means clustering algorithm. *Expert Systems with Applications*, *40*(1), 200–210. doi:10.1016/j.eswa.2012.07.021

Duwairi, R., & Abu-Rahmeh, M. (2015). A novel approach for initializing the spherical K-means clustering algorithm. *Simulation Modelling Practice and Theory*, *54*, 49–63. doi:10.1016/j.simpat.2015.03.007

Elgendy & Elragal. (2014). Big Data Analytics: A Literature Review Paper. In P. Perner (Ed.), *ICDM 2014, LNAI 8557* (Vol. 76, pp. 214–227). Springer International Publishing. 10.1007/978-3-319-08976-8_16

Faber, V. (1994). Clustering and the Continuous k-Means Algorithm. *Los Alamos Science*, *22*, 138–144.

Fahad, A., Alshatri, N., Tari, Z., Alamri, A., Khalil, I., Zomaya, A. Y., Foufou, S., & Bouras, A. (2014). A survey of clustering algorithms for big data: Taxonomy and empirical analysis. *IEEE Transactions on Emerging Topics in Computing*, *2*(3), 267–279. doi:10.1109/TETC.2014.2330519

Forgy, E. (1965). Cluster analysis of multivariate data : Efficiency versus interpretability of classificationse. *Biometrics*, *21*, 768–780.

Franti, P., Kaukoranta, T., & Nevalainen, O. (1997). On the splitting method for vector quantization codebook generation. *Optical Engineering (Redondo Beach, Calif.)*, *36*(11), 3043. Advance online publication. doi:10.1117/1.601531

Fränti, P., & Sieranoja, S. (2018). K-means properties on six clustering benchmark datasets. *Applied Intelligence*, *48*(12), 4743–4759. doi:10.1007/s10489-018-1238-7

Fränti, P., & Sieranoja, S. (2019). How much can k-means be improved by using better initialization and repeats? *Pattern Recognition*, *93*, 95–112. doi:10.1016/j.patcog.2019.04.014

Gan, G., Ma, C., & Wu, J. (2007). Data Clustering Theory, Algorithms, and Applications. American Statistical Association and the Society for Industrial and Applied Mathematics.

Gandomi, A., & Haider, M. (2015). Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management*, *35*(2), 137–144. doi:10.1016/j.ijinfomgt.2014.10.007

Gentle, J. E., Kaufman, L., & Rousseuw, P. J. (1990). *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley & Sons. doi:10.2307/2532178

Gonzalez, T. F. (1985). Clustering to minimize the maximum intercluster distance. *Theoretical Computer Science*, *38*(C), 293–306. doi:10.1016/0304-3975(85)90224-5

Hand, D. J., & Krzanowski, W. J. (2005). Optimising k-means clustering results with standard software packages. *Computational Statistics & Data Analysis*, *49*(4), 969–973. doi:10.1016/j.csda.2004.06.017

Hariri, R. H., Fredericks, E. M., & Bowers, K. M. (2019). Uncertainty in big data analytics: Survey, opportunities, and challenges. *Journal of Big Data*, *6*(1), 44. Advance online publication. doi:10.1186/s40537-019-0206-3

Hartigan, J. A., & Wong, M. A. (1979). Algorithm AS 136: A K-Means Clustering Algorithm. *Journal of the Royal Statistical Society. Series C, Applied Statistics*, *28*(1), 100–108. doi:10.2307/2346830

He, J., Lan, M., Tan, C. L., Sung, S. Y., & Low, H. B. (2004). Initialization of cluster refinement algorithms: A review and comparative study. *IEEE International Conference on Neural Networks - Conference Proceedings, 1*(1), 297–302. doi:10.1109/IJCNN.2004.1379917

Jain, A. K. (2010). Data clustering: 50 years beyond K-means. *Pattern Recognition Letters*, *31*(8), 651–666. doi:10.1016/j.patrec.2009.09.011

Kacem, N'Cir, Ben, & Essoussi. (2019)., Clustering Methods for Big Data Analytics. In Unsupervised and Semi-Supervised Learning (pp. 1–23). doi:10.1007/978-3-319-97864-2_1

Kalyanakrishnan, S. (2017). *k-means Clustering*. Academic Press.

Khondoker, M. R. (2018). Big Data Clustering. *Wiley StatsRef: Statistics Reference Online*, 1–10. 10.1002/9781118445112.stat07978

Lee, I. (2017). Big data: Dimensions, evolution, impacts, and challenges. *Business Horizons*, *60*(3), 293–303. doi:10.1016/j.bushor.2017.01.004

Likas, A., Vlassis, N., & Verbeek, , J. (2003). The global k-means clustering algorithm. *Pattern Recognition*, *36*(2), 451–461. doi:10.1016/S0031-3203(02)00060-2

MacQueen, J. (1967). Some methods for classification and analysis of multivariate ob- servations. *Berkeley Symposium on Mathematical Statistics and Probability*, 281–297.

Maheswari, K., & Ramakrishnan, M. (2019). Kernelized Spectral Clustering based Conditional MapReduce function with big data. *International Journal of Computers and Applications*, *0*(0), 1–11. doi:10.1080/12062 12X.2019.1587892

Milligan, G. W., & Isaac, P. D. (1980). The validation of four ultrametric clustering algorithms. *Pattern Recognition*, *12*(2), 41–50. doi:10.1016/0031-3203(80)90001-1

Mirkin, B. (2005). *Clustering: A Data Recovery Approach*. Chapman and Hall/CRC Boris.

Pandey, K. K., Shukla, D., & Milan, R. (2020). A Comprehensive Study of Clustering Algorithms for Big Data Mining with MapReduce Capability. In R. K. Shukla, J. Agrawal, S. Sharma, N. S. Chaudhari, & K. K. Shukla (Eds.), Social Networking and Computational Intelligence, Lecture Notes in Networks and Systems 100 (pp. 427–440). Springer Nature Singapore Pte Ltd. doi:10.1007/978-981-15-2071-6_34 427

Pandey, K. K., & Shukla, D. (2019). A study of clustering taxonomy for big data mining with optimized clustering mapreduce model. *International Journal on Emerging Technologies*, *10*(2).

Pandove, D., Goel, S., & Rani, R. (2018). Systematic review of clustering high-Dimensional and large datasets. *ACM Transactions on Knowledge Discovery from Data*, *12*(2), 1–68. Advance online publication. doi:10.1145/3132088

Peña, J. M., Lozano, J. A., & Larrañaga, P. (1999). An empirical comparison of four initialization methods for the K-Means algorithm. *Pattern Recognition Letters*, *20*(10), 1027–1040. doi:10.1016/S0167-8655(99)00069-0

Sieranoja, S., & Fränti, P. (2018). Random Projection for k-means Clustering. In Artificial Intelligence and Soft Computing (LNCS, volume 10841). Springer. doi:10.1007/978-3-319-91253-0_63

Sivarajah, U., Kamal, M. M., Irani, Z., & Weerakkody, V. (2017). Critical analysis of Big Data challenges and analytical methods. *Journal of Business Research*, *70*, 263–286. doi:10.1016/j.jbusres.2016.08.001

Steinbach, M., Karypis, G., & Kumar, V. (2000). A comparison of document clustering techniques. *KDD Workshop on Text Mining*.

Steinley, D. (2003). Local Optima in K-Means Clustering: What You Don't Know May Hurt You. *Psychological Methods*, *8*(3), 294–304. doi:10.1037/1082-989X.8.3.294 PMID:14596492

Steinley, D., & Brusco, M. J. (2007). Initializing K-means batch clustering: A critical evaluation of several techniques. *Journal of Classification*, *24*(1), 99–121. doi:10.1007/s00357-007-0003-0

Su, T., & Dy, J. G. (2007). In search of deterministic methods for initializing K-means and Gaussian mixture clustering. *Intelligent Data Analysis*, *11*(4), 319–338. doi:10.3233/IDA-2007-11402

Tabesh, P., Mousavidin, E., & Hasani, S. (2019). Implementing big data strategies: A managerial perspective. *Business Horizons*, *62*(3), 347–358. doi:10.1016/j.bushor.2019.02.001

Tou, J. T., & González, R. C. (1974). *Pattern recognition principles*. Addison-Wesley Pub. Co.

Viens, A. (2019). *Visualizing Social Media Use by Generation*. Visual Capitalist. https://www.visualcapitalist.com/visualizing-social-media-use-by-generation

von Luxburg, U. (2010). Clustering Stability: An Overview. *Foundations and Trends in Machine Learning*, *2*(3), 235–274. doi:10.1561/2200000008

Wang, R., Ji, W., Liu, M., Wang, X., Weng, J., Deng, S., Gao, S., & an Yuan, C. (2018). Review on mining data from multiple data sources. *Pattern Recognition Letters*, *109*, 120–128. doi:10.1016/j.patrec.2018.01.013

Xiao, Y., & Yu, J. (2012). Partitive clustering (K-means family). *Wiley Interdisciplinary Reviews. Data Mining and Knowledge Discovery*, *2*(3), 209–225. doi:10.1002/widm.1049

Xie, H., Zhang, L., Lim, C. P., Yu, Y., Liu, C., Liu, H., & Walters, J. (2019). Improving K-means clustering with enhanced Firefly Algorithms. *Applied Soft Computing*, *84*, 105763. doi:10.1016/j.asoc.2019.105763

Yu, A. S., Chu, S., & Wang, C. (2017). Two Improved k-means Algorithms. *Applied Soft Computing*. Advance online publication. doi:10.1016/j.asoc.2017.08.032

Zahra, S., Ghazanfar, M. A., Khalid, A., Azam, M. A., Naeem, U., & Prugel-Bennett, A. (2015). Novel centroid selection approaches for KMeans-clustering based recommender systems. *Information Sciences*, *320*, 156–189. doi:10.1016/j.ins.2015.03.062

Kamlesh Kumar Pandey is pursuing a Ph.D. from Dr. HariSingh Gour Vishwavidyalaya (A Central University), Sagar, India, under the supervision of Prof. Diwakar Shukla. Currently, He is researching the Big Data Clustering algorithms, concerning three dimensions of Big Data. He is the author and co-author of several research papers in international journals and conferences such as IEEE, Springer, and others. He has 8 years of teaching and research experience. He has awarded in Training of Young Scientist in 34th and 35th M.P. Young Scientist Congress.

Diwakar Shukla is presently working as a Head , Department of Computer Science and Applications and Dean ,School of Mathematical and Physical Sciences, Dr. Harisingh Gour Vishwavidyalaya, Sagar, MP, India and has over 30 years' experience regarding teaching and research. He obtained M.Sc. (Stat.), Ph.D. (Stat.) degrees from Banaras Hindu University, Varanasi, UP and served the Devi Ahilya University, Indore, M.P. as a permanent Assistant Professor up to nine years and obtained M.Tech.(Computer Science) degree. He joined Dr. Harisingh Gour Vishwavidyalaya, Sagar as an Associate Professor in Statistics in year 1998. During Ph.D. from BHU, he was junior and senior research fellow of CSIR, New Delhi which he obtained through qualifying All India Fellowship Test of 1983. Till now, he has published more than 130 research papers in national and international journals and participated in more than 40 seminars/conferences at the national level. He also worked as a Professor in Lucknow University, Lucknow, UP., for one year (from 2007 to 2008) . He visited abroad to Sydney (Australia) and Shanghai (China) for conference participation and paper presentation. He has supervised 17 Ph.D. theses in Statistics and Computer Science and seven students are presently enrolled for their doctoral degree under his supervision. He is the author of six books and member of 11 learned bodies of Statistics and Computer Science at the national level. Areas of his research works are Sampling Theory, Graph Theory, Stochastic Modeling, Data mining, Big Data, Operation Research, Computer Network, and Operating Systems.