SciEn
Journal of Engineering Advancements Vol. 02(01) 2021, pp 01-06 https://doi.org/10.38032/jea.2021.01.001

Improved Mean Shift Algorithm for Maximizing Clustering Accuracy

Chinmay Bepery[1,], Shaneworn Bhadra[1], Md. Mahbubur Rahman[1], Mihir Kanti Sarkar[2] and Mohammad Jamal Hossain[1]*

[1]Department of Computer Science and Information Technology, Patuakhali Science and Technology University, Patuakhali-8602, Bangladesh
[2]Ministry of Housing and Public Works, Bangladesh

Received: November 26, 2020, Revised: December 29, 2020, Accepted: December 31, 2020, Available Online: January 02, 2021

ABSTRACT

Clustering is a machine learning method that can group similar data points. Mean Shift (MS) is a fixed window-based clustering algorithm, which calculates the number of clusters automatically but cannot guarantee the convergence of the algorithm. The main drawback of the Mean Shift Algorithm is that the algorithm requires to set a stopping criterion (threshold point) otherwise all clusters move towards one cluster and fixed bandwidth is used here. It cannot define the upper bound of iteration numbers and need to set the iteration numbers. This paper proposed a new Mean Shift Algorithm, called Improved Mean Shift (IMS) algorithm, which overcomes the all defined pitfalls of Mean Shift Algorithm. The IMS process KD-tree data structure was used to sort the dataset and all data points as initial cluster centroids without a random selection of initial centroids. In each iteration, it shifts the variable bandwidth sliding window to the actual data point nearest to the mean using k-nearest neighbours (kNN) algorithm and finds the number of clusters automatically. Also, this paper handles the missing values using Mean Imputation (MI). The IMS algorithm produces better results than the Mean Shift Algorithm on both synthetic and real datasets.

Keywords: Clustering; Mean Shift; KD-tree; kNN; Mean Imputation.

1. Introduction

Data mining becomes a very popular decision support technique where we can extract hidden, unknown and valuable knowledge among the large amount of data [1]. Clustering must be a significant application of data mining that ensures the grouping of data points where similar groups contain mostly similar types of data points and different groups contain highly dissimilar types of data points. Nowadays, the speedily rising computer technology produced many large volume and highly dimensional datasets [2]. Clustering is an essential part of data analysis that can ensure the partition of data points where similar objects should be in the same cluster [3].

Mean Shift (MS) is an iterative, non-parametric fixed bandwidth clustering method used widely in many applications. Fukunaga and Hostetler proposed this algorithm in 1975 that introduced for locating dense areas of data points using sliding window [4]. The main advantage of Mean Shift (MS) method is that it does not want the previous knowledge about the number of clusters and does not constrain the clusters shape. Missing values in dataset become a great problem in the real world applications. Some methods are used to deal with this problem. Missing values in data are mainly caused by equipment failures, system errors, human errors, and so on [5]. The methods used for handling missing values are divided into two categories. First one is case deletion method: In this method, we need to delete all data points with missing values. If there are less instances with missing values, we can delete them, but if there are more missing values and delete them, the dataset becomes small and impacts the results [6]. The second one is missing data imputation technique: Here, we replace the missing values with the distribution's known value. Using this method, the IMS Algorithm can work better as like as a complete dataset because each missing value is replaced with known values. In this paper we handle missing values by Mean Imputation (MI) method where we have replaced the missing data values with the mean of all the instances in the dataset.

This paper focused on the improvement in the quality and accuracy of the Mean Shift Algorithm. Our proposed Improved Mean Shift method can define the upper bound of the iteration number, but this characteristic is missing in MS Method. In contrast to the MS algorithm, the proposed IMS does not require to set a stopping criterion (threshold point) and the number of iterations. Also, the Improved Mean Shift Algorithm provide the guarantee of convergence. We performed many experiments with Improved Mean Shift Algorithm (IMS) on synthetic and real datasets. Our proposed algorithm gives better clustering result on the selected datasets than Mean Shift Algorithm.

2. Related Works

Several methods have been proposed over the last few years to improve the quality and accuracy of the Mean Shift Algorithm. Most of the author used fixed bandwidth, and no one handles missing values problem. Also, no one defines the upper bound of the iteration number. Chunxia Xiao et al. [7] proposed an Efficient Mean-shift Clustering technique that used a reduced feature-space to improve the result. The reduced feature-space represents an adaptive clustering result of the original dataset using adaptive KD-tree structure in high dimensional feature space. But fixed bandwidth is used here, and for this reason, it's very complicated to get an optimal size bandwidth for the dataset of different size and dimension to get better clustering result. Bogdan Georgescu et al. [8] proposed a new technique called locality-sensitive-hashing (LSH) algorithm to minimize the computational complexity of adaptive Mean Shift process, but

*Corresponding Author Email Address: chinmay.cse@gmail.com

here we need to use pilot learning technique to discover the optimal parameters of the dataset. Vo Thi Ngoc Chau et al. [9] proposed a new clustering algorithm that has two parts. The first part is used to resolve the incompleteness of education data and the second part proposed a Mean Shift-based clustering approach using the nearest prototype strategy called MMS_nps. The main limitation is that it cannot automatically evaluate the bandwidth value h based on datasets' inherent characteristics. Dorin Comaniciu et al. [10] planned a Modified Mean Shift Algorithm for solving the automatic bandwidth problem (variable bandwidth). It can find an optimal bandwidth for dataset of different size and dimension to get better clustering result. But the convergence of the algorithm is not proven and need to set iteration number for clustering purpose. Loai AbdAllah et al. [11] proposed a new Mean Shift clustering technique that can handle missing values problem of datasets. They take a weighted distance function called MD_E distance with Mean Shift Algorithm instead of Euclidian distance to compute the distance between two points with missing attribute values. But need to set stopping criterion.

3. Improved Mean Shift Algorithm

In the case of the Mean Shift Algorithm, we need to set a stopping criterion (threshold point) and define the iteration number's upper bound. Also, it uses fixed bandwidth and cannot guarantee the convergence of algorithm. Our proposed Improved Mean Shift Algorithm (IMS) can solve these problems. IMS works with some following steps given below.

3.1 Handling missing values by Mean Imputation (MI)

Presence of missing values in the dataset are very common problem in real-world applications. If there are less instances with missing values, we can delete them. But if there are more instances with missing values and delete them, the dataset becomes small and the characteristics of datasets become change. Due to this missing value the performance and accuracy of algorithm decrease heavily. The Mean Imputation (MI) method is used here to replace the missing values to solve this problem.

Mean Imputation (MI): Replace the missing values with the mean of all the instances in each column.

3.2 KD-tree for Data Partition

KD-tree is a binary search tree where the data points are organized in K-dimensional feature space [12]. KD-tree is used in this paper in order to store and represent the dataset in a data structure. A non-leaf node in KD-tree divides the feature space into two parts where points in the left of this space are defined by left subtree of that node and points to the right of the space by the right subtree [13]. Suppose we have two Dimensional data (x,y) showing in Table 1.

Table 1 2D Dataset for KD-tree.

Data points	x	y
Data_point1	0.67	0.97
Data_point2	0.33	0.76
Data_point3	0.40	0.68
Data_point4	0.12	0.56
Data_point5	0.60	0.30
Data_point6	0.28	0.72
Data_point7	0.83	0.73

Firstly, we divide the data points into two parts by comparing each x value with root of x. Root(x) = Max(x) + Min(x)/2= (0.83 + 0.12)/2= 0.48. Next label we compare dividing two groups y values with root of y. Root(y) = Max(y) + Min(y)/2. Repeat this until fulfill the condition. Every node has three things such as (1). Dimension, (2). Value and (3). Tightest bounding box.

Table 2 Tight bounds for node _1 and node_ 2.

Tight bounds	x	y
Node _1	0.11 <= x <= 0.42	0.53 <= y <= 0.75
Node_ 2	0.54 <= x<= 0.96	0.29 <= y <= 0.93

Table 2 shows the tightest bounds area for node_1 and node_2. Similarly, divide the data structure into more parts on the basis of dimensions until each leaf node holds maximum two data points.

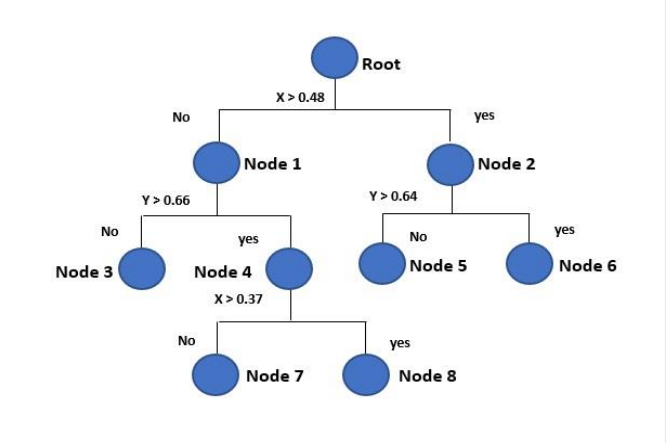

Fig. 1 KD-tree for TABLE I dataset.

Fig. 1 displays the visual representation of sorted data for Table 1. At first, the label compares x value with the root of x. Next label compares y value with the root of y. Repeat this pattern until each leaf node holds maximum two points [14].

3.3 Improved Mean Shift Clustering

In Improved Mean Shift (IMS) algorithm, we take all data points as initial cluster centres and set variable bandwidth sliding window in each data points. In each iteration, the sliding window is moved towards higher density areas by moving the initial centers to the actual data point nearest to the mean using the KNN algorithm. Multiple sliding windows overlap when they have same mean, and then data points are clustered according to the sliding window in which they reside.

KNN algorithm is used to find the nearest data point. It first loads the data points and set the number of K. Euclidean distance is used here for distance measurement purpose. Next, we need to sort the distance and get our expected nearest expected data point.

The kernel density function is used here with Improved Mean Shift (IMS) algorithm.

Given m data objects q_j, j = 1,…,m on a d-dimensional space R_d. For m number of data points, we have m initial cluster From Fig. 3 shows that in every iteration, the sliding window

shift to new centroids by moving the initial centres to the actual data point nearest to the mean using the kNN algorithm, inside the sliding window. Multiple sliding windows overlap when they have the same mean. Finally, the data points are clustered properly by the help of the sliding window.

centroids, and the algorithm become converges at most (m-1) iterations. So there is no need to set the number of iterations. Our proposed IMS algorithm can provide the upper bound of the number iterations (i.e. m − 1) for each data point. The multivariate kernel density estimation obtained with kernel K(q) and window radius $h_j \equiv h(q_j)$ is

$$f(q_j)=\frac{1}{mh_j^{\,d}}\sum_{j=1}^{m-1}K\left(\frac{q-q_j}{h_j}\right) \quad (1)$$

For radially symmetric kernels, Kernel k(q) satisfying

$$K(q)=c_{k,d^k}(||q||^2) \quad (2)$$

where $c_{k,d}$ is defined as normalization constant that guarantees $K(q) \propto 1$ and modes of K(q) are pointing at $\nabla f(q)=0$.

The gradient of density estimator (1) is

$$\nabla f\left(q_j\right)=\frac{2c_{k,d}}{mh_j^{\,d+2}}\sum_{j=1}^{m-1}(q_j-q)g\left(\left\|\frac{q-q_j}{h_j}\right\|\right)^2$$

$$=\frac{2c_{k,d}}{mh_j^{\,d+2}}\left[\sum_{j=1}^{m-1}g\left(\left\|\frac{q-q_j}{h_j}\right\|\right)^2\right]\left[\frac{\sum_{j=1}^{m-1}q_j g\left(\left\|\frac{q-q_j}{h_j}\right\|\right)^2}{\sum_{j=1}^{m-1}g\left(\left\|\frac{q-q_j}{h_j}\right\|\right)^2}-q\right] \quad (3)$$

where $g(p)=-k'(p)$. The first term of $\nabla f\left(q_j\right)$ is proportional to the density estimated at q with kernel $G(q) = c_{g,d^g}(\|q\|^2)$ and the second term is

$$m_{h_j}(q_j)=\frac{\sum_{j=1}^{m-1}q_j g\left(\left\|\frac{q-q_j}{h_j}\right\|\right)^2}{\sum_{j=1}^{m-1}g\left(\left\|\frac{q-q_j}{h_j}\right\|\right)^2}-q \quad (4)$$

defined as Improved Mean Shift with variable bandwidth hj and number of iteration (m-1). Improved Mean Shift vector always moves toward the direction of the maximum dense area.

So, the Improved Mean Shift can be obtained by

- evaluating Improved Mean Shift vector $m_{h_j}(q^t)$
- translation of sliding window $q^{t+1}=q^t+m_{h_j}(q^t)$

That provides the guarantee of convergence of the algorithm where $\nabla f\left(q_j\right)=0$.

Improved Mean Shift (IMS) mode finding process is illustrated in Fig. 2 and Fig. 3. From Fig. 2, we know that the Improved Mean Shift clustering algorithm is also a practical application of the mode finding procedure: In Improved Mean Shift clustering algorithm we first take the dataset as weighted matrix and handle the missing values using Mean imputation method. Then take all data points as initial cluster centres. Next, we should set a variable bandwidth sliding window in each data points for clustering purpose. There is no need to set the iteration numbers. For handling outliers, we also set a condition that can solve the outlier problem.

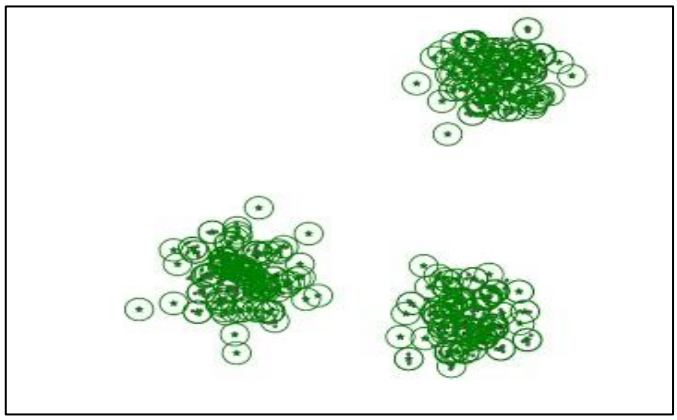

Fig. 2 Improved Mean Shift clustering procedure (take all data points as initial cluster centres).

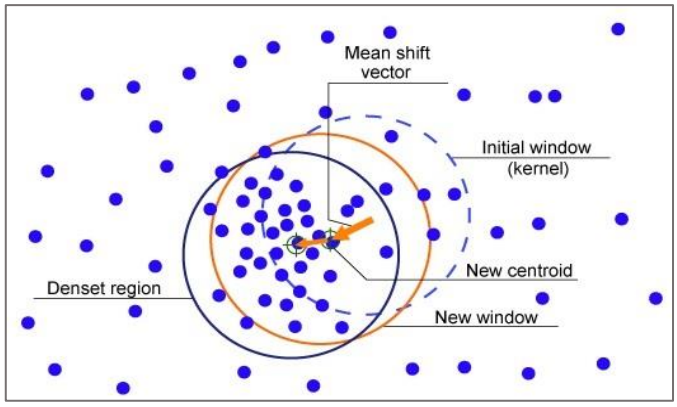

Fig. 3 Improved Mean Shift clustering procedure (shifting window).

This paper used the Gaussian KD-tree algorithm to speed up the IMS clustering process for large data sets. KD-tree algorithm partitions the datasets based on feature space in a top-down way. It begins from a root cell, and recursively split a root into two child cells adaptively along with a dimension that is alternated at successive tree levels [7].

3.4 Overview of the Algorithm

The proposed Improved Mean Shift Algorithm (IMS) works with the following steps:

Input:

A high dimensional dataset $Q=\{q_1.....q_n\}, n \geq 2$ on a d-dimensional space, Variable bandwidth h_i where i=1…n and Profile function g(q).

Output:

Clustering results $R_{1.....}R_k$, where k defined the number of clusters.

Steps:

Initialize the dataset as weighted matrix.
Use Mean Imputation for handling missing values.
Sort the dataset by KD-tree data structure.
Take all data points as initial cluster centroids.
Set variable bandwidth sliding windows in each data point.
Calculate the mean of instances lying inside the window.

Find the actual data points nearest to the mean using KNN algorithm and shift the window to that points.

Repeat till convergence and gain k number of Clusters.

Eliminate cluster that contains less than a minimum number of data points based on the condition for handling outliers.

4. Experiments

In the experimental area, we show the calculated results of our proposed IMS algorithm on synthetic and real datasets to measure IMS's performance compared with the Mean Shift Algorithm.

4.1 Datasets

The IMS algorithm operates on both synthetic and real datasets to measure the clustering output. The synthetic datasets are made by using Gaussian distribution [15]. Here we have used ten synthetic datasets of different size. The synthetic datasets are shown in Table 3. Dataset are characterized by instance number q, cluster number k and the feature number n.

Table 3 Characteristics of synthetic datasets.

Serial Number	Synthetic Dataset	Instance (q)	Feature (n)	Cluster (k)
1	S_data1	1000	2	3
2	S_data2	3000	2	3
3	S_data3	5000	2	3
4	S_data4	10000	2	5
5	S_data5	1500	5	3
6	S_data6	2500	10	5
7	S_data7	3500	15	10
8	S_data8	4500	15	10
9	S_data9	5500	15	10
10	S_data10	6500	20	10

Next phase, we select five real-world datasets downloaded from the UCI machine learning repository [15]. The real-world datasets are seen in Table 4.

Table 4 Characteristics of real datasets.

Serial Number	Real Datasets	Instance (q)	Feature (n)	Cluster (k)
1	Wine	178	13	3
2	Iris	150	4	3
3	Seed	210	7	3
4	Glass	214	10	6
5	Mammo	961	6	2

4.2 Experimental Settings and Evaluation Methods

To measure the clustering accuracy of IMS algorithm, we used two types of evaluation techniques in this paper. They are Purity and Rand Index defined as defined below:

Purity: Purity is a popular estimation technique that calculates the percentage of correctly classified objects. The purity range is between 0 and 1. Purity is defined as follows:

$$f(x)=\frac{1}{P}\sum_{j=1}^{n} max_n|Q_j \cap R_n| \quad (5)$$

Here (a). P defines the number of objects in datasets, (b). n defines cluster numbers, (c). Q_j defines the set of instances in cluster j, (d). R_n defines a set of objects in class n that has the highest number of intersections with cluster j, among all the clusters.

Rand Index: Rand Index is another popular evaluation technique where a set of m elements $Q=\{Q_1,\ldots..,Q_n\}$ divide into two partitions R and S to compare $R=\{R_1,\ldots..R_u\}$with u subsets and $S=\{S_1,\ldots..,S_v\}$with v subsets, define as following:

- e : defines the number of pairs in set S that are same for both R and S subsets.
- f : defines the number of pairs in set S that are different for both R and S subsets.
- g: defines the number of pairs in set S that are same in R subset and different in S subset.
- h: defines the number of pairs in set S that are different in R subset and same in S subset.

$$R=\frac{e+f}{e+f+g+h}=\frac{e+f}{\binom{m}{k}} \quad (6)$$

Intuitively e+f defines the number of agreements and g+h defines number of disagreements between R and S.

4.3 Experimental Results and Analysis

This section discusses and compares the experimental results between our proposed Improved Mean Shift (IMS) and Mean Shift Algorithm (MS). We take ten synthetic datasets and five real datasets to present the increment in clustering result using the IMS. In the MS algorithm, we need to set the number of iteration and a stopping criterion (a threshold point ϵ) otherwise all clusters may move toward one cluster. It also used fixed bandwidth, so it is tough to find an optimal sized bandwidth for different dataset to get better clustering result.

So, the convergence of Mean Shift is not proven. But in IMS, we use all data points as initial cluster centres, and if there are m data points, we need at most m-1 iterations to fulfil convergence criterion. Also, no need to establish a stopping criterion and iteration numbers and also handle outliers. Variable bandwidth, KD-tree and kNN algorithm are also used with IMS for better accuracy with better clustering result.

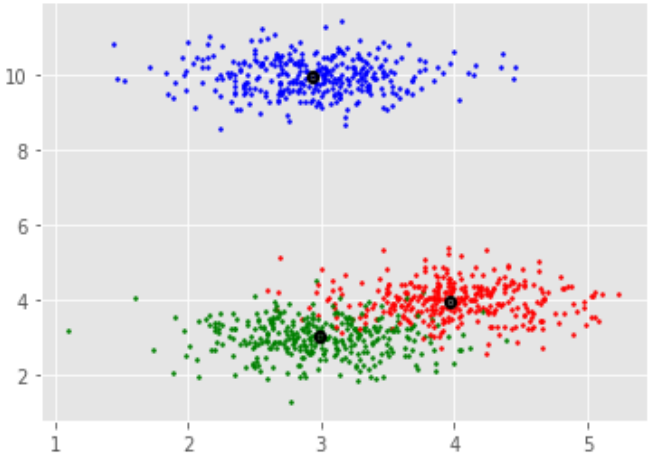

Fig. 4 Plotting of a well-distributed dataset with Mean Shift.

Fig. 4 shows the clustering results on a synthetic dataset named S_dataset1 using Mean Shift. S_dataset1 is a well-decorated dataset has 800 instances in 2D space with three clusters. We used fixed bandwidth h=1000, stopping threshold point ϵ =0.01 and number of iterations 100 for running Mean Shift Algorithm. After completing all iteration, the Mean Shift Algorithm can determine three clusters, but two clusters become ambiguous. On the contrary, in improved Mean Shift we need at most (800-1) iterations to clustered S_dataset1, and for handling outliers, we need to set a condition defined as an estimated cluster that holds more than 5 data points otherwise delete that cluster. Variable bandwidth is used here that can select bandwidth automatically.

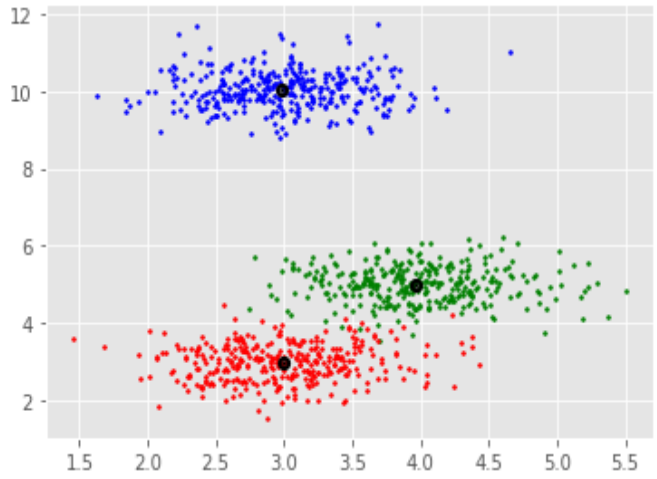

Fig. 5 Plotting of a well-distributed dataset with Improved Mean Shift.

Fig. 5 show that Improved Mean Shift can cluster the data points more correctly than the Mean Shift Algorithm. So, we can agree that IMS is far better clustering algorithm than Mean Shift.

Table 5 Comparison of clustering accuracy between Mean Shift and improved Mean Shift on synthetic datasets.

Synthetic Data	**Purity**		**Rand Index**	
	MS	***IMS***	***MS***	***IMS***
S_data1	0.791	0.802	0.789	0.810
S_data2	0.820	0.831	0.822	0.839
S_data3	0.858	0.858	0.849	0.851
S_data4	0.715	0.748	0.720	0.739
S_data5	0.775	0.789	0.768	0.780
S_data6	0.798	0.811	0.729	0.792
S_data7	0.765	0.796	0.704	0.751
S_data8	0.718	0.823	0.767	0.834
S_data9	0.709	0.799	0.818	0.851
S_data10	0.712	0.770	0.695	0.743

Table 5 presents the accuracy-test, including purity and rand index for ten synthetic datasets using both Mean Shift and Improved Mean Shift Algorithm. Here we see that S_data3 shows the almost similar result with IMS and MS. But the others show better results in accuracy test using Improved Mean Shift Algorithm than Mean Shift Algorithm. Following these explanations, it clears that our proposed IMS is far better than MS on accuracy Comparison. Python Spyder (ana) is used here for coding purpose.

Table 6 Comparison of clustering accuracy between Mean Shift and improved Mean Shift on real datasets.

Real Data	**Purity**		**Rand Index**	
	MS	***IMS***	***MS***	***IMS***
Wine	0.701	0.704	0.710	0.719
Iris	0.755	0.791	0.713	0.741
Seeds	0.696	0.719	0.633	0.674
Glass	0.661	0.679	0.512	0.552
Mammo	0.511	0.602	0.636	0.683

Table 6 presents the accuracy-test, including purity and rand index for five real datasets using both MS and IMS algorithm. Viewing these observations, we can say that our proposed IMS is far better than the Mean Shift Algorithm for these given five real datasets based on accuracy. Fig. 6 shows the comparison plot of accuracy between MS and IMS.

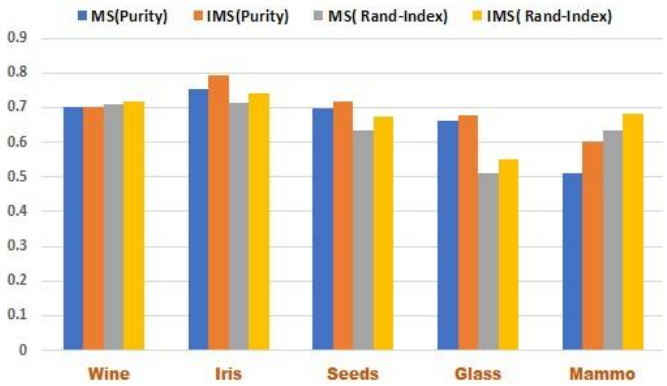

Fig. 6 Visual representation of accuracy between MS and IMS.

5. Conclusion and Future Work

Mean Shift uses fixed bandwidth, and for this reason, it is tough to find an optimal sized bandwidth for different dataset to get better clustering result. Also, it cannot guarantee the convergence of the algorithm and requires to set a stopping criterion named threshold point and the upper bound of the iteration number. But our proposed Improved Mean Shift Algorithm can solve all these problems. IMS uses all data points as initial cluster centres, and if there are n data points, we need at most m-1 iterations to being convergence. Also, set a stopping criterion is not needed here. It also eliminates cluster that contains less than a minimum number of data points based on the condition for handling outliers. Variable bandwidth sliding window, KD-tree and KNN algorithm are also used in IMS for better accuracy and better clustering results. The time complexity of our IMS is more than MS, but accuracy is much higher. So, in the future, we try to reduce our IMS algorithm's time complexity and try it on more complex datasets.

References

[1] Jiawei, H. and Micheline, K., 2005. Data Mining: Concepts and Techniques [M] Beijing. Mechanical Industry Press.

[2] Han, J. and Kamber, M., 2001."Data Mining Concepts and Techniques, Morgan Kaufmann Publishers," San Francisco, CA, pp. 335-391.

[3] Bindra, K. and Mishra, A., 2017, September. A detailed study of clustering algorithms. In *2017 6th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)* (pp. 371-376). IEEE.

[4] Fukunaga, K. and Hostetler, L., 1975. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on information theory*, *21*(1), pp.32-40.

[5] AbdAllah, L. and Shimshoni, I., A distance function for data with missing values and its applications on knn and kmeans algorithms. *Submitted to Int. J. Advances in Data Analysis and Classification*.

[6] Zhang, S., Qin, Z., Ling, C.X. and Sheng, S., 2005. "Missing is useful": missing values in cost-sensitive decision trees. *IEEE transactions on knowledge and data engineering*, *17*(12), pp.1689-1693.

[7] Xiao, C. and Liu, M., 2010. Efficient mean-shift clustering using gaussian kd-tree. In *Computer Graphics Forum* Vol. 29, No. 7, pp. 2065-2073.

[8] Georgescu, B., Shimshoni, I. and Meer, P., 2003, October. Mean shift based clustering in high dimensions: A texture classification example. In *null* (p. 456). IEEE.

[9] Chau, V.T.N., Loc, P.H. and Tran, V.T.N., 2015, November. A robust mean shift-based approach to effectively clustering incomplete educational data. In *2015 International Conference on Advanced Computing and Applications (ACOMP)* (pp. 12-19). IEEE.

[10] Comaniciu, D., Ramesh, V. and Meer, P., 2001, July. The variable bandwidth mean shift and data-driven scale selection. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001* (Vol. 1, pp. 438-445). IEEE.

[11] AbdAllah, L. and Shimshoni, I., 2014, September. Mean shift clustering algorithm for data with missing values. In *International Conference on Data Warehousing and Knowledge Discovery* (pp. 426-438). Springer, Cham.

[12] Masud, M.A., Rahman, M.M., Bhadra, S. and Saha, S., 2019, December. Improved k-means Algorithm using Density Estimation. In *2019 International Conference on Sustainable Technologies for Industry 4.0 (STI)* (pp. 1-6). IEEE.

[13] Singh, G., 2017. "Introductory guide to Information Retrieval using kNN and KDTree," *analytics vidhya.*

[14] Adams, A., Gelfand, N., Dolson, J. and Levoy, M., 2009. Gaussian kd-trees for fast high-dimensional filtering. In *ACM SIGGRAPH 2009 papers* (pp. 1-12).

[15] Masud, M.A., Huang, J.Z., Zhong, M., Fu, X. and Mahmud, M.S., 2018, November. Slice_OP: Selecting Initial Cluster Centers Using Observation Points. In *International Conference on Advanced Data Mining and Applications* (pp. 17-30). Springer, Cham.