Malaysian Journal of Mathematical Sciences 14(1): 41–62 (2020)

MALAYSIAN JOURNAL OF MATHEMATICAL SCIENCES

Journal homepage: http://einspem.upm.edu.my/journal

# New Approaches to Normalization Techniques to Enhance K-Means Clustering Algorithm

Dalatu, P. I. *[1,3] and Midi, H.[1,2]

[1]*Department of of Mathematics, Faculty of Science, Universiti Putra Malaysia, Malaysia*
[2]*Institute for Mathematical Research, Universiti Putra Malaysia, Malaysia*
[3]*Department of of Mathematics, Faculty of Science, Adamawa State University, Mubi-Nigeria*

*E-mail: dalatup@gmail.com*
* *Corresponding author*

*Received: 25 December 2017*
*Accepted: 25 October 2019*

## ABSTRACT

Clustering is fundamentally one of the leading origin of basic data mining tools, which makes researchers believe the normal grouping of attributes in datasets. The main aim of clustering is to ascertain similarities and arrangements with a large dataset by partitioning data into clusters. It is important to note that distance measures like Euclidean distance, should not be used without normalization of datasets. The limitation of using both Min-Max (MM) and Decimal Scaling (DS) normalization methods are that the minimum and maximum values may be out-of-samples when dataset are unknown. Therefore, we proposed two new normalization approaches to overcome attributes with initially large magnitudes from overweighing attributes with initially smaller magnitudes. The two new normalization approaches are called New Approach to Min-Max (NAMM) and New Approach to Decimal Scaling (NADS). To evaluate the performance of our proposed approaches, simulation study and real

data applications are considered. However, the two proposed approaches have shown good performance compared to the existing methods, by achieving nearly maximum points in the average external validity measures, recorded lower computing time and clustering the object points to almost all their cluster centers. Consequently, from the results obtained, it can be noted that the NAMM and NADS approaches yielded better performance in the data preprocessing methods, which down weight the magnitudes of large values.

**Keywords:** Normalization, k-means, simulation, clustering.

# 1. Introduction

Clustering is an unsupervised arrangement method with leading objective of separation, where points in the same cluster are alike, and points belong to different clusters differ importantly, with regard to their attributes (Mohd et al. (2012) and Krishnasamy et al. (2014)). The technique has the generalization such that points in a cluster are minimizing intra-cluster sameness and maximizing the inter-cluster dissimilarity, with regard to their attributes (refer Nazeer and Sebastian (2009)).

Clustering is fundamentally one of the leading origin of basic data mining tools, which makes researchers believe the normal grouping of attributes in datasets in Goil et al. (1999), El Agha and Ashour (2012) and Rokach and Maimon (2008). Therefore, the quantity of data being gathered in some business and scientific fields, may be exposed to data analysis for the comprehensive truths finding interested and past undiagnosed arrangement. Clustering approaches most particularly for large scale data with also large number of attributes are attaining established achievement of undertaking for knowledge discovery and to achieve data mining in databases with much effectiveness and efficiency (refer Goil et al. (1999)).

Clustering is often applied as the beginning of first steps in data analysis. It functions as an assessment to discover natural clusters in data sets to identify theoretical patterns that might live in, without having any primary ideas on the features of data in Mohd et al. (2012). The major aim of clustering is to ascertain similarities and arrangements within a large dataset by partitioning data into clusters (refer to Suarez-Alvarez et al. (2012)). However, it is acknowledged that data are taken as unlabeled and clustering is usually determined as the most important unsupervised learning assignment (refer to Patel and Mehta (2011) and Suarez-Alvarez et al. (2012)).

Our scope of this study is particularly with partitional clustering algorithms, which is considered completely for the clusters and at the same time as a partition of the data, that is arrangement of the data objects into non-overlapping subsets. According to Christopher et al. (2008), there is an important variation between hard and soft clustering algorithms. Hard clustering computes a hard task-where each object is a member of exactly one cluster. While soft cluster task algorithm- an object‘s task is being determined by distributing over entire clusters. It is known that in soft task, an object has fractional relationship in various clusters.

The K-Means algorithm was first developed by MacQueen (1967) and the algorithm was later highly-developed and expanded by Lloyd (1982), is the renowned and fast-breaking approach in partitioning cluster algorithms (refer to Mohd et al. (2012) and El Agha and Ashour (2012)). The k-means algorithm process is well automated and is less economical to calculate. Considering its inexpensiveness in terms of economical charges, it is achievable to analyse very large sample on a digital computer (MacQueen (1967)).

Founded on its quality and easiness, the K-Means algorithm has been applied in many areas. The algorithm although is very easy and strong in clustering large data sets, the method suffers from some setbacks **Duwairi and Abu-Rahmeh, 2015**. The number of clusters have to be known before hand when applying most of the real world data sets in Rokach and Maimon (2008). It has to undergo the issues of random selection of initial cluster centers (centroids), which may be sensitive to the algorithm in Barakbah and Kiyoki (2009). Nonetheless, the algorithm cannot achieve global optimum results (refer to Reddy et al. (2012) and Rokach and Maimon (2008)). The K-Means algorithm repeatedly converge to a local minimum. The issue of local minimum is being established on the initial cluster centers. Also, the problem of exploratory global minimum is NP (nondeterministic polynomial time)-complete (in Oyelade et al. (2010)). Usually, k-means algorithm continually updates cluster centers until local minimum is achieved. It is observed that in literature, one of the weaknesses of K-Means clustering algorithm is that when unnormalized dataset is used, it is often that the outcome performance may not reach global optimum (Han et al. (2011)).

Data preprocessing methods commonly used raw data to make the data clean, noise free, and consistent (in Patel and Mehta (2011)). Data normalization tasks is to standardize raw data by changing it into classified interval through linear transformation in order to produce good quality clusters and improve the accuracy of clustering algorithms. A normalized dataset observe to produce better outcomes during the actual clustering process by Patel and Mehta (2011). This prevents out weighing features having a large number upon features with smaller numbers. The main aim is to equalize the magnitude and also, prevent the much inconsistency in those features (in Mohamad and Usman (2013)).

However, the features are expected to have no dimension as the numerical values of the intervals of dimensional features rest on the units of measurements and selection of units of measurements may seriously change the outcomes of clustering. It is important to note that distance measures like Euclidean distance, should not be used without normalization of datasets (in Aksoy and

Haralick (2001)), because dissimilarity measure influences differences in the magnitudes of the input variables (Milligan and Cooper (1988)). To date, there is no generally definite algorithm for normalizing the datasets, therefore, the user has right to choose desired algorithm (in Visalakshi and Thangavel (2009)).

In this study we were motivated by a problem pointed out in Visalakshi and Thangavel (2009), that up to this present time, there is no specifically certain rule for normalizing the datasets, however, the researcher has open options to select whichever approach he wishes to apply. In addition, Wu et al. (2009), stated that, the importance of normalizing validation measures has not been completely accepted. Also, it was stated by Aksoy and Haralick (2001), that is essential to take into cognizance that distance measures like Euclidean distance should not be applied without preprocessing.

Furthermore, according to Nayak et al. (2014) and Ogasawara et al. (2010) recently, that the main limitation of using both Min-Max and Decimal Scaling normalization method is that the minimum and maximum values may be out-of-sample when data set are unknown (according to Kotsiantis and Pintelas (2004)), samples may have unknown attribute values, which do not have group attributes associated with the majority of samples). This issue may occur most especially in data sample like time series forecast data set as it may be applied to Equations 2 and 3. Supported by Tan et al. (2005) and Hand et al. (2001), these techniques may lead to important information loss and to a concentration of values on certain part of the normalized range. Yet, Patel and Mehta (2011), Visalakshi and Thangavel (2009) and Sola and Sevilla (1997), stated that it may imply more computational effort and also loss of quality in the learning approaches.

Therefore, the aim of this study is to improve the conventional normalization techniques in quest for definite algorithm and higher quality clusters from a standard K-Means algorithm in clustering analysis. This paper is organized as follows: Section 2 presents materials and methods; the conventional and proposed methods. Section 3 gives the results and discussion Section 4 finally, some concluding remarks were given.

# 2. Materials and Methods

## 2.1 Conventional Methods

In the literature, there are a number of conventional techniques in normalization and standardization, but the most common used ones are min-max, decimal scaling, and Z-score methods. However, for our study, we are going to limit ourselves to the two normalization approaches as min-max and decimal scaling. Furthermore, we also would like to investigate the performance of K-Means clustering algorithm that evaluates dataset without normalization, which often being practice by practitioners.

### 2.1.1 K-Means Clustering Algorithm

The K-means clustering algorithm consist of four steps, which are iterated until convergence (Mohamad and Usman (2013)). The iteration will stop when the clusters produced are stable, which means there are no more movement of objects crossing any group. The K-Means algorithms are enlisted by MacQueen (1967), Lloyd (1982) and Shirkhorshidi et al. (2015) as follows: The K-Means clustering algorithm is broadly used in data mining to group data with similar features together. Assumed n data points, the algorithm distributes them into k groups in three stages: (1) evaluate the distances between data points with each of k clusters and assign the data to the nearest cluster, (2) calculate the center of each cluster, (3) update the clusters repeatedly until the k clusters change no more or stabilized. The aim of the algorithm is to minimize the cost function. The cost function (in Khan (2012)):

$$J = \sum_{i=1}^{n} \sum_{j=1}^{k} \parallel x_i - c_j \parallel^2 \tag{1}$$

where, $\parallel x_i - c_j \parallel^2$ is an arbitrary distance measure between a data point $x_i$ and the cluster center $c_j$ is assigned to the distance of the n data points from their individual centers.

The algorithm consists of the following steps (Khan (2012)):

1. Initialize the centers at random;
2. Assign data points to their respective clusters having the nearest mean;

3. Compute new centers as means of the clusters assigned in step 2;
4. Repeat steps 2 and 3 until no change is made in the centers.

It creates a partition of the objects into groups from which the metric to be minimize can be calculated, after data normalization as given below in Equation 2, 3, 4, and 5.

### 2.1.2 Min-Max(MM)

The normalization executes linear transformation on the original data. The min-max represents lowest and highest values for an attribute $j$, with range given as $(0, 1)$. The normalized value represents $v_i$, of $j$ to $v_i'$ defined and computed as in Jayalakshmi and Santhakumaran (2011):

$$v_i' = \frac{v_i - min_j}{max_j - min_j} \tag{2}$$

where, $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, n$ is attribute values. It is being stated in Han et al. (2011), that min-max normalization conserves the relation amongst the primary data values.

### 2.1.3 Decimal Scaling(DS)

It normalizes the dataset by moving the decimal point of values of attribute $j$. The number of decimal points moved depends on the maximum absolute value of $j$. The value, $v_i$ of $j$ is normalized to $v_i'$ and computed as in Han et al. (2011):

$$v_i' = \frac{v_i}{10^j} \tag{3}$$

where $j$ is the smallest integer (the integer $j$ is equal to the maximum numbers of digits; example, 986, $j = 3$).

## 2.2 Proposed Methods

In this section we will discuss the two proposed normalization approaches. The two proposed approaches are based on the min-max of in Jayalakshmi and

Santhakumaran (2011) and decimal scaling of by Han et al. (2011). Therefore, in Jayalakshmi and Santhakumaran (2011) and Han et al. (2011) claimed that by normalizing the data preserves the relationship among the original data values. Data transformation such as normalization may improve the accuracy and efficiency of mining algorithms like K-Means algorithm. They further emphasized that for distance-based methods, normalization helps to prevent attributes with initially large ranges from overweighting attributes with initially smaller ranges.

Therefore, on some limitations and weaknesses of normalization of by Jayalakshmi and Santhakumaran (2011) and Han et al. (2011), that up to this present time there is no specifically certain rule for normalizing the datasets; however, the researcher has open options to select whichever approach he(she) wishes to apply (refer to Visalakshi and Thangavel (2009)). While, by Wu et al. (2009) supported that the importance of normalizing validation measures has not been completely accepted. Also, by Han et al. (2011), noted that normalization can change the original data quite a bit especially when using decimal scaling.

Furthermore, in Liu (2011) and Jain et al. (2005), have identified one of the weakness of using both the min-max and decimal scaling in data transformation. They stated that both of the methods will have overflow problem, this makes the two methods lack robustness. However, inZumel and Mount (2013) and Jain et al. (2005), suggested that, in order to remedy this problem in decimal scaling method, one may use $\log_{10}\max(x_i)$. While in min-max method, and Liu (2011) and Milligan (1989), suggested to down weighing the method so that irrelevant variables approach near zero.

Therefore, we were motivated by lack of robustness of the two methods and we adopted the ideas suggested by Zumel and Mount (2013) and Jain et al. (2005), for decimal scaling and (Liu (2011) and Milligan (1989)) for min-max methods to improve the methods of min-max (Jayalakshmi and Santhakumaran (2011)) and decimal scaling (Han et al. (2011)).

The two new approaches to normalization techniques are tagged as NAMM (New Approach to Min-Max) and NADS (New Approach to Decimal Scaling). The proposed approaches are summarized as follows.

### 2.2.1 New Approach to Min-Max (NAMM)

The new approach to min-max, adopted ideas from Liu (2011) and Milligan (1989). It has its parameters as minimum and maximum, which are being

represented by the lowest and highest values respectively and the attributes having range as $(0, 1)$. The normalized value is calculated using the ideas from Equation 2 with changes made in the denominator by introducing $c$ variable as follows:

$$v_i' = \frac{v_i - min_j}{(max_j - min_j)^c} \tag{4}$$

where $c = 2$ is a constant raised to the power of the denominator. Therefore, any integer greater than two will make the variable values approximately zero.

The constant $c$ is an integer used, in order to down weight (Liu (2011) and Milligan (1989) maximum variable values dominating the variability of the magnitudes.

### 2.2.2 New Approach to Decimal Scaling (NADS)

The new approach to decimal scaling is formulated following the ideas of Zumel and Mount (2013), but with a slight modifications where normalization is done by replacing the decimal point of values of feature $j$ with that of $c+1$. The number of decimal points moved depends on the maximum absolute value of the attributes by Mohamad and Usman (2013). The new approach to decimal scaling is calculated using the ideas from Equation 3 with the introduction of $c+1$ to power 10 replacing the maximum absolute integer value with absolute real value using logarithm base 10 as follows:

$$v_i' = \frac{v_i}{10^{(c+1)}} \tag{5}$$

where $c = log_{10} max(x_i)$; if evaluated without adding 1 to $c$, all the variable values will be slightly greater than 1, which is out of bound for the upper range. Therefore, it is calculated based on the following conditions and rules:

1. We first compute the largest absolute value in each row using logarithm base 10 and plus 1 each.
2. Then, divide the original row value by 10 raised to this computed value to obtain the normalized value.

However, it is important to mention that after the transformation of data by min-max, decimal scaling and the two proposed methods, the following steps are carried out to compare the performance of the proposed methods and the existing methods:

1. Perform the K-Means clustering (with unnormalized data).
2. Then, perform the K-Means clustering with the classical and the proposed normalization methods.
3. Some external measures such as Purity (in Hernandez-Torruco et al. (2014)), Fowlkes-Mallow Index (in Velardi et al. (2012), Rand Index (Noorbehbahani et al. (2015)), F-Measure Score, Jaccard Index, F-Measure ($\beta$ varied) (refer to Velardi et al. (2012)), Geometric Means (Tomar and Agarwal (2015)), Precision (Kou et al. (2014) and Rokach and Maimon (2008)), Specificity (Velardi et al. (2012)), Accuracy (Tomar and Agarwal (2015)), Sensitivity (Velardi et al. (2012)) and the computing time (minutes) are recorded.

# 3. Results and Discussion

## 3.1 Simulation Study

In this section, Monte Carlo simulation study is presented to compare the performance of some existing methods such as Conventional K-Means by Shirkhorshidi et al. (2015) (not transformed), Min-Max (Jayalakshmi and Santhakumaran (2011)), and Decimal Scaling (Han et al. (2011)), with our proposed methods NAMM (New Approach to Min-Max) and NADS (New Approach to Decimal Scaling). Following Shirkhorshidi et al. (2015), Jayalakshmi and Santhakumaran (2011) and Han et al. (2011), two variables $(x_1, x_2)$ and four variables $(x_1, x_2, x_3, x_4)$ are generated such that each of the exploratory variables $(x_1, x_2)$ and $x_1, x_2, x_3, x_4$ are simulated from uniform distribution with range $[-10, 10]$. The variables are clustered into three class (luster or group) as cluster 1, cluster 2 and cluster 3. We consider a sample of size $(50, 100, 160)$. The conventional distance functions, K-Means clustering algorithm, Min-Max (MM), Decimal Scaling (DS) and the proposed New Approach to Min-Max (NAMM) and New Approact to Decimal Scaling (NADS) were then applied to the data.

Some external validity measures such as: Purity, Fowlkes-Mallow Index, Rand Index, F-Measure Score, Jaccard Index, F-Measure ($\beta$ varied), Geometric Means, Precision, Specificity, Accuracy, Sensitivity, and the computing time (minutes) are recorded. In each of the experimental runs, there are 1000 replications. The performance of the five methods are evaluated based on average external validity measures for each methods , computational timing (minutes) and having three levels of cluster as; cluster 1, cluster 2, and cluster 3.

n = 50, $(x_2, x_2)$

Table 1: Av. Ext. Validity Measures, Computing Time and Max. Clusters

| Method | Convt. | MM | DS | NAMM | NADS |
|---|---|---|---|---|---|
| Purity | 0.8444 | 0.8333 | 0.8556 | 0.8788 | 0.8878 |
| Fow. M.I. | 0.7357 | 0.7447 | 0.8168 | 0.8781 | 0.8885 |
| Rand Index | 0.7567 | 0.7655 | 0.8304 | 0.8852 | 0.8997 |
| F.M.Score | 0.7444 | 0.7355 | 0.7553 | 0.8778 | 0.8871 |
| Jaccard I. | 0.7879 | 0.7890 | 0.7913 | 0.8569 | 0.8607 |
| F.M. Varied | 0.7358 | 0.7367 | 0.7554 | 0.8778 | 0.8788 |
| G-Means | 0.7589 | 0.7567 | 0.7657 | 0.8831 | 0.8907 |
| Precision | 0.7380 | 0.7400 | 0.8608 | 0.8792 | 0.8738 |
| Specificity | 0.7678 | 0.7689 | 0.7778 | 0.8889 | 0.8900 |
| Accuracy | 0.7568 | 0.7668 | 0.7704 | 0.8852 | 0.8865 |
| Sensitivity | 0.7444 | 0.7434 | 0.7556 | 0.8778 | 0.8795 |
| Average | 0.7610 | 0.7619 | 0.7941 | 0.8790 | 0.8839 |
| Comput. time(min) | 32 | 32 | 28 | 23 | 22 |
| Clust.1(max15) | 11 | 11 | 10 | 13 | 14 |
| Clust.2(max15) | 12 | 11 | 11 | 14 | 13 |
| Clust.3(max20) | 11 | 12 | 14 | 15 | 16 |

n = 50, $(x_1, x_2, x_3, x_4)$

Table 2: Av. Ext. Validity Measures, Computing Time and Max. Clusters

| Method | Convt. | MM | DS | NAMM | NADS |
|---|---|---|---|---|---|
| Purity | 0.8444 | 0.8333 | 0.8556 | 0.8988 | 0.9078 |
| Fow. M.I. | 0.8457 | 0.8547 | 0.8668 | 0.8981 | 0.8885 |
| Rand Index | 0.8567 | 0.8655 | 0.8704 | 0.8905 | 0.9007 |
| F.M.Score | 0.8444 | 0.8355 | 0.8553 | 0.8977 | 0.8871 |
| Jaccard I. | 0.7879 | 0.7890 | 0.8163 | 0.8569 | 0.8607 |
| F.M. Varied | 0.8358 | 0.8367 | 0.8554 | 0.8778 | 0.8788 |
| G-Means | 0.8589 | 0.8567 | 0.8657 | 0.8831 | 0.9078 |
| Precision | 0.8380 | 0.8400 | 0.8608 | 0.8792 | 0.8708 |
| Specificity | 0.8678 | 0.8689 | 0.8778 | 0.8889 | 0.8900 |
| Accuracy | 0.8568 | 0.8668 | 0.8704 | 0.8852 | 0.8865 |
| Sensitivity | 0.8444 | 0.8434 | 0.8556 | 0.8778 | 0.8795 |
| Average | 0.8437 | 0.8446 | 0.8591 | 0.8849 | 0.8871 |
| Comput. time(min) | 38 | 38 | 36 | 34 | 33 |
| Clust.1(max15) | 11 | 10 | 12 | 13 | 12 |
| Clust.2(max15) | 10 | 11 | 11 | 11 | 11 |
| Clust.3(max20) | 13 | 13 | 14 | 16 | 17 |

n = 100, $(x_1, x_2)$

Table 3: Av. Ext. Validity Measures, Computing Time and Max. Clusters

| Method | Convt. | MM | DS | NAMM | NADS |
|---|---|---|---|---|---|
| Purity | 0.8666 | 0.8555 | 0.8778 | 0.8900 | 0.8990 |
| Fow. M.I. | 0.8579 | 0.8669 | 0.8789 | 0.8993 | 1.0000 |
| Rand Index | 0.8789 | 0.8877 | 0.8792 | 0.9000 | 1.0000 |
| F.M.Score | 0.8666 | 0.8577 | 0.8775 | 1.0000 | 1.0000 |
| Jaccard I. | 0.7890 | 0.7900 | 0.8185 | 0.8780 | 0.8829 |
| F.M. Varied | 0.8578 | 0.8589 | 0.8776 | 0.8991 | 0.8911 |
| G-Means | 0.8713 | 0.8789 | 0.8879 | 0.8931 | 0.8956 |
| Precision | 0.8590 | 0.8422 | 0.8830 | 0.9000 | 0.8930 |
| Specificity | 0.8890 | 0.8890 | 1.0000 | 1.0000 | 1.0000 |
| Accuracy | 0.8788 | 0.8889 | 0.8972 | 1.0000 | 1.0000 |
| Sensitivity | 0.8666 | 0.8656 | 0.8778 | 1.0000 | 1.0000 |
| Average | 0.8620 | 0.8619 | 0.8869 | 0.9327 | 0.9511 |
| Comput. time(min) | 41 | 41 | 39 | 37 | 34 |
| Clust.1(max30) | 24 | 23 | 23 | 24 | 27 |
| Clust.2(max30) | 22 | 22 | 21 | 24 | 26 |
| Clust.3(max40) | 24 | 25 | 33 | 37 | 37 |

n = 100, $(x_1, x_2, x_3, x_4)$

Table 4: Av. Ext. Validity Measures, Computing Time, and Max. Clusters

| Method | Convt. | MM | DS | NAMM | NADS |
|---|---|---|---|---|---|
| Purity | 0.8511 | 0.8511 | 0.8511 | 0.8722 | 0.8711 |
| Fow.M.I | 0.8312 | 0.8412 | 0.8522 | 0.8734 | 0.8625 |
| Rand I. | 0.8514 | 0.8414 | 0.8734 | 0.8945 | 0.9000 |
| F.M.Score | 0.8412 | 0.8413 | 0.8511 | 0.8822 | 0.8733 |
| Jaccard I. | 0.7745 | 0.8695 | 0.8851 | 0.9052 | 0.9070 |
| F.M.Varied | 0.8330 | 0.8420 | 0.9520 | 0.9535 | 0.9545 |
| G-Means | 0.8563 | 0.8573 | 0.9000 | 0.9775 | 0.9745 |
| Precision | 0.8340 | 0.8440 | 0.8540 | 0.9667 | 0.9548 |
| Specificity | 0.8565 | 0.8745 | 0.8900 | 0.9000 | 0.9000 |
| Accuracy | 0.8523 | 0.8523 | 0.8734 | 0.9745 | 0.9000 |
| Sensitivity | 0.8333 | 0.8411 | 0.8511 | 0.9622 | 0.9612 |
| Average | 0.8377 | 0.8505 | 0.8758 | 0.9238 | 0.9144 |
| Comput. time(min) | 42 | 43 | 40 | 37 | 37 |
| Clust.1(max30) | 23 | 23 | 22 | 23 | 24 |
| Clust.2(max30) | 21 | 22 | 22 | 24 | 24 |
| Clust.3(max40) | 24 | 25 | 30 | 36 | 35 |

$$\text{n} = 160,\ (x_1, x_2)$$

Table 5: Av. Ext. Validity Measures, Computing Time, and Max. Clusters

| Method | Convt. | MM | DS | NAMM | NADS |
|---|---|---|---|---|---|
| Purity | 0.9411 | 0.9411 | 0.9411 | 0.9622 | 0.9511 |
| Fow.M.I | 0.9201 | 0.9301 | 0.9411 | 0.9534 | 0.9625 |
| Rand I. | 0.9403 | 0.9503 | 0.9323 | 0.9745 | 1.0000 |
| F.M.Score | 0.9630 | 0.9301 | 0.9400 | 0.9622 | 0.9533 |
| Jaccard I. | 0.8634 | 0.8834 | 0.8930 | 0.9052 | 0.9070 |
| F.M.Varied | 0.9220 | 0.9310 | 0.9410 | 0.9535 | 0.9535 |
| G-Means | 0.9452 | 0.9462 | 0.9500 | 0.9775 | 0.9734 |
| Precision | 0.9240 | 0.9330 | 0.9430 | 0.9557 | 0.9503 |
| Specificity | 0.9454 | 0.9734 | 0.9600 | 1.0000 | 1.0000 |
| Accuracy | 0.9412 | 0.9412 | 0.9423 | 0.9745 | 0.9800 |
| Sensitivity | 0.9201 | 0.9301 | 0.9400 | 0.9622 | 0.9601 |
| Average | 0.9296 | 0.9354 | 0.9385 | 0.9619 | 0.9628 |
| Comput. time(min) | 60 | 57 | 57 | 53 | 53 |
| Clust.1(max50) | 35 | 41 | 40 | 43 | 44 |
| Clust.2(max50) | 36 | 37 | 39 | 40 | 40 |
| Clust.3(max60) | 50 | 51 | 50 | 53 | 52 |

$$\text{n} = 160,\ (x_1, x_2, x_3, x_4)$$

Table 6: Av. Ext. Validity Measures, Computing Time, and Max. Clusters

| Method | Convt. | MM | DS | NAMM | NADS |
|---|---|---|---|---|---|
| Purity | 0.8733 | 0.9433 | 0.9433 | 0.9844 | 0.9733 |
| Fow.M.I | 0.8534 | 0.9334 | 0.9444 | 0.9756 | 0.9847 |
| Rand I. | 0.8736 | 0.9436 | 0.9656 | 0.9967 | 1.0000 |
| F.M.Score | 0.8630 | 0.9334 | 0.9433 | 0.9844 | 0.9755 |
| Jaccard I. | 0.8367 | 0.8977 | 0.8973 | 0.9274 | 0.9290 |
| F.M.Varied | 0.8550 | 0.9340 | 0.9440 | 0.9757 | 0.9767 |
| G-Means | 0.8785 | 0.9495 | 0.9800 | 0.9997 | 0.9967 |
| Precision | 0.8561 | 0.9361 | 0.9461 | 0.9779 | 0.9705 |
| Specificity | 0.8787 | 0.9667 | 0.9700 | 1.0000 | 1.0000 |
| Accuracy | 0.8745 | 0.9445 | 0.9656 | 0.9967 | 1.0000 |
| Sensitivity | 0.8534 | 0.9334 | 0.9433 | 0.9844 | 0.9834 |
| Average | 0.8633 | 0.9378 | 0.9494 | 0.9821 | 0.9809 |
| Comput. time(min) | 81 | 68 | 67 | 62 | 62 |
| Clust.1(max50) | 40 | 43 | 42 | 44 | 43 |
| Clust.2(max50) | 35 | 37 | 38 | 42 | 44 |
| Clust.3(max60) | 40 | 49 | 50 | 54 | 53 |

Tables $(1-6)$, exhibit the average values of 1000 replications of the external validity measures, computational timing (minutes), and maximum number of samples in each clusters. A good normalization method is the one that has average external validity measure closer to 1 or (1) at maximum, minimum computation time, and has the correct number of samples (as prior assigned above) in each clusters.

It can be clearly observed from Tables $(1-6)$, that the two proposed methods; NAMM and NADS had shown the maximum average performance of external validity measures, recorded the lowest computational time and clustered the maximum clusters in each groups. For example in Table 6, with sample size $(n = 160; x_1, x_2, x_3, x_3)$; the average external validity measures for NAMM and NADS are 0.9823 and 0.9811, respectively, which is closer to 1 and computing time (minutes) for both are 62 and 62, respectively. The total clustered out of 160 for NAMM and NADS are 140 $(44, 42, 54)$ and 140 $(43, 44, 53)$, respectively. However, the conventional method (without transformation) gave the smallest value. For its average external validity measures is 0.8626, computing time is 81 and total clustered is 117 $(40, 37, 40$. This indicates that the performance of NAMM and NADS are more accurate and efficient compared to the existing methods. It is evident that the conventional method without transformation give the poor results. Therefore, based on this simulated data results, the two proposed methods may be used especially in distance-based data preprocessing clustering analysis methods in many sectors of real lfe situations.

In order to see the effect of outliers on the performance of our proposed methods, the data are contaminated with 5% and 10% outliers. Here, we considered three different sample size $(n = 50, 100, 160)$, with two $(x_1, x_2)$ and four $(x_1, x_2, x_3, x_4)$ attribute variables each. Each of the variables are generated from uniform distribution $[-10, 10]$. The contaminated data (outliers) is generated from uniform distribution $[15, 16]$. The data is contaminated by replacing certain percentage of good observations with outliers. The average external validity measures and computational times are then recorded in Table 7.

Table 7 exhibits clearly the advantage of data preprocessing procedure before any clustering analysis especially distance-based. It can be observed that the proposed methods in the presence of outliers still were able to perform well based on the maximum average performance of external validity measures, and lower computing time for each methods. This finding has shown evidently that our proposed methods are better compared to the existing methods even in the presence outliers.

$$\mathrm{n} = (50, 100, 160)$$

Table 7: Average External Validity Measures and Computing Time

| n | Cont. | Method | $x_1, x_2$ | | $x_1, x_2, x_3, x_4$ | |
|---|---|---|---|---|---|---|
| | | | Av.Ext. | C.Time | Av.Ext. | C.Time |
| 50 | 5% | Conv. | 0.6434 | 57 | 0.6191 | 60 |
| | | MM | 0.7051 | 56 | 0.6933 | 58 |
| | | DS | 0.7149 | 55 | 0.7016 | 59 |
| | | **NAMM** | **0.7628** | **49** | **0.7340** | **52** |
| | | **NADS** | **0.7703** | **48** | **0.7421** | **51** |
| | 10% | Conv. | 0.5811 | 67 | 0.5674 | 70 |
| | | MM | 0.6479 | 63 | 0.6211 | 64 |
| | | DS | 0.6532 | 61 | 0.6353 | 63 |
| | | **NAMM** | **0.6991** | **56** | **0.6774** | **58** |
| | | **NADS** | **0.7002** | **55** | **0.6834** | **57** |
| 100 | 5% | Con. | 0.7010 | 68 | 0.6819 | 72 |
| | | MM | 0.7274 | 65 | 0.7163 | 67 |
| | | DS | 0.7277 | 65 | 0.7181 | 67 |
| | | **NAMM** | **0.7632** | **61** | **0.7318** | **63** |
| | | **NADS** | **0.7594** | **61** | **0.7310** | **63** |
| | 10% | Conv. | 0.6175 | 73 | 0.6021 | 75 |
| | | MM | 0.6490 | 71 | 0.6341 | 73 |
| | | DS | 0.6487 | 71 | 0.6327 | 73 |
| | | **NAMM** | **0.6908** | **66** | **0.7029** | **68** |
| | | **NADS** | **0.6917** | **66** | **0.7015** | **68** |
| 160 | 5% | Conv. | 0.6001 | 74 | 0.5872 | 76 |
| | | MM | 0.6285 | 71 | 0.6174 | 73 |
| | | DS | 0.6310 | 71 | 0.6200 | 73 |
| | | **NAMM** | **0.6874** | **66** | **0.6631** | **69** |
| | | **NADS** | **0.6822** | **65** | **0.6602** | **69** |
| | 10% | Conv. | 0.5460 | 77 | 0.5168 | 80 |
| | | MM | 0.5847 | 72 | 0.5633 | 76 |
| | | DS | 0.5921 | 72 | 0.5670 | 76 |
| | | **NAMM** | **0.6572** | **68** | **0.6214** | **70** |
| | | **NADS** | **0.6695** | **67** | **0.6304** | **69** |

## 3.2 Real Data Applications

In this section, the Iris and Hayes-Roth datasets are considered to verify the performance of our proposed methods:

### 3.2.1 Iris dataset

The iris dataset was applied by many researchers such as in Galili (2012), Jayalakshmi and Santhakumaran (2011), **Benson-Putnins et al., (2011)** and Han et al. (2011). The dataset contains 3 classes of 150 instances each , where each class refers to a type of iris plant. It comprises the following attributes information: (1) Sepal length in cm, (2) Sepal width in cm, (3) Petal length in cm, and (4) Petal width in cm. The classes are listed as follows: (1) iris Setosa, (2) iris Verisiclor, and (3) iris Virginica (refer to Bache and Lichman (2013)).

### 3.2.2 Hayes-Roth dataset

The Hayes-Roth dataset was used by many researchers such as Han et al. (2011), Jayalakshmi and Santhakumaran (2011) and Ryu and Eick (2005). The dataset contains 3 classes of 160 instances each, with 4 attributes namely: (1) hobby, (2) age, (3) educational and (4) marital status (Bache and Lichman (2013)).

Tables 8 and 9 presents the average performance of external validity measures and computing time under each distance functions. Generally, on the average, all the two datasets indicated that the two proposed approaches had shown impressive performance based on higher maximum average external validity measures and lower computing time. It is noted that, based on the two datasets applied; real numbers used in iris dataset gives higher quality performance in the external validity measures compared to integer numbers used in Hayes-Roth dataset.

Table 8: Average External Validity Measures and Computing Time under each Distance Functions, Iris Dataset

| **Methods** | Convt. | Min-Max | D.Scaling | NAMM | NADS |
|---|---|---|---|---|---|
| Purity | 0.9072 | 0.9079 | 0.9072 | 0.9305 | 0.9319 |
| F.Mallow Index | 0.9188 | 0.9157 | 0.9174 | 0.9239 | 0.9305 |
| Rand Index | 0.9206 | 0.9176 | 0.9233 | 0.9398 | 0.9413 |
| F.M(F-Score) | 0.9003 | 0.9077 | 0.9052 | 0.9173 | 0.9219 |
| Jaccard Index | 0.8933 | 0.8967 | 0.8933 | 0.9201 | 0.9308 |
| F.M(F-$\beta$varied) | 0.9016 | 0.9051 | 0.9022 | 0.9109 | 0.9145 |
| G-Means | 0.9045 | 0.9137 | 0.9211 | 0.9287 | 0.9291 |
| Precision | 0.9038 | 0.9164 | 0.9182 | 0.9211 | 0.9252 |
| Specificity | 0.9219 | 0.9395 | 0.9347 | 0.9401 | 0.9472 |
| Accuracy | 0.9137 | 0.9218 | 0.9161 | 0.9214 | 0.9291 |
| Sensitivity | 0.9067 | 0.9081 | 0.9069 | 0.9378 | 0.9393 |
| Average | 0.9084 | 0.9137 | 0.9132 | 0.9265 | 0.9310 |
| Comput.Time(Min.) | 48 | 47 | 47 | 44 | 43 |
| Clust.1(max150) | 138 | 136 | 136 | 140 | 140 |
| Clust.2(max150) | 138 | 137 | 138 | 138 | 138 |
| Clust.3(max150) | 91 | 141 | 140 | 141 | 140 |

Table 9: Average External Validity Measures and Computing Time under each Distance Functions, Hayes-Roth Dataset

| **Methods** | Convent. | Min-Max | D.Scaling | NAMM | NADS |
|---|---|---|---|---|---|
| Purity | 0.4250 | 0.4267 | 0.4309 | 0.4485 | 0.4497 |
| F.Mallow Index | 0.4129 | 0.4211 | 0.4256 | 0.4371 | 0.4384 |
| Rand Index | 0.4375 | 0.4383 | 0.4450 | 0.4575 | 0.4603 |
| F.M(F-Score) | 0.4261 | 0.4275 | 0.4280 | 0.4305 | 0.4391 |
| Jaccard Index | 0.4132 | 0.4266 | 0.4259 | 0.4391 | 0.4395 |
| F.M(F-$\beta$varied) | 0.4152 | 0.4218 | 0.4247 | 0.4294 | 0.4300 |
| G-Means | 0.5463 | 0.5477 | 0.5483 | 0.5545 | 0.5561 |
| Precision | 0.4133 | 0.4139 | 0.4141 | 0.4206 | 0.4219 |
| Specificity | 0.6171 | 0.6185 | 0.6188 | 0.6211 | 0.6229 |
| Accuracy | 0.5037 | 0.5045 | 0.5133 | 0.5251 | 0.5290 |
| Sensitivity | 0.4355 | 0.4386 | 0.4393 | 0.4408 | 0.4453 |
| Average | 0.4587 | 0.4623 | 0.4649 | 0.4731 | 0.4757 |
| Comput.Time(Min.) | 49 | 48 | 48 | 46 | 46 |
| Clust.1(max160) | 99 | 99 | 99 | 99 | 100 |
| Clust.2(max160) | 87 | 88 | 88 | 89 | 89 |
| Clust.3(max160) | 73 | 74 | 74 | 75 | 76 |

# 4. Conclusion

In this paper, we proposed two normalization approaches to overcome attributes with initially large range from overweighting attributes with initially smaller ranges. The new normalization approaches are called new approach to min-max (NAMM) and new approach to decimal-scaling (NADS). The proposed approaches are based on normalizing the data to preserves the relationship among the original data values and also, may improve the accuracy and efficiency of mining algorithms like the K-Means algorithm.

The results indicate that the conventional K-Means without normalization has the least performance. This is due to the fact that distance measures like Euclidean distance, should not be applied without normalization of datasets. Although, the two proposed approaches have good performance; evidently, by achieving nearly maximum points in the external validity measures and clustering the object points to almost all their cluster centers and recorded lower computing time.

From the results, it can be concluded that the NAMM and NADS approaches are much better in the data preprocessing methods; which down weight the magnitudes of larger values.

# Acknowledgements

The authors thank the anonymous referee for their constructive comments and suggestions. The authors also acknowledge the financial support from Universiti Putra Malaysia through Geran Putra.

# References

Aksoy, S. and Haralick, R. M. (2001). Feature normalization and likelihood-based similarity measures for image retrieval. *Pattern recognition letters*, 5(22):563–582.

Bache, K. and Lichman, M. (2013). Uci machine learning repository. *Pattern recognition letters*. http://archive.ics.uci.edu/ml.

Barakbah, A. R. and Kiyoki, Y. (2009). A pillar algorithm for k-means optimization by distance maximization for initial centroid designation. *Computational Intelligence and Data Mining 2009, IEEE Symposium.*

Christopher, D. M., Prabhakar, R., and Hinrich, S. (2008). *Introduction to information retrieval: Introduction to information retrieval.* Cambridge, England: Cambridge University Press.

El Agha, M. and Ashour, W. M. (2012). Efficient and fast initialization algorithm for k means clustering. *I.J. Intelligent Systems and Applications*, 1:21–31.

Galili, T. (2012). Dendextend: an r package for visualizing, adjusting and comparing trees of hierarchical clustering. *Bioinformatics*, 31(22):3718–3720.

Goil, S., Nagesh, H., and Choudhary, A. (1999). Mafia: Efficient and scalable subspace clustering for very large data sets. *Technical Report No. CPDC-TR-9906-010*, pages 1–20. North Western University: Center for Parallel and Distributed Computing.

Han, J., Kamber, M., and Pei, J. (2011). *Data mining: concepts and techniques.* USA: Morgan Kaufmann Publisher. 3rd Edition.

Hand, D. J., Mannila, H., and Smyth, P. (2001). *Principles of data mining.* London, England: The MIT Press.

Hernandez-Torruco, J., Canul-Reich, J., Frausto-SolÃs, J., and Mendez-Castillo, J. J. (2014). Feature selection for better identification of subtypes of guillain-barre syndrome. *Computational and mathematical methods in medicine*, 2014:1–9. Article ID 432109, http://dx.doi.org/10.1155/2014/432109.

Jain, A., Nandakumar, K., and Ross, A. (2005). Score normalization in multimodal biometric systems. *Pattern Recognition*, 38(12):2270–2285.

Jayalakshmi, T. and Santhakumaran, A. (2011). Statistical normalization and back propagation for classification. *International Journal of Computer Theory and Engineering*, 3(1):1793–8201.

Khan, F. (2012). An initial seed selection algorithm for k-meams clustering of georeferenced data to improve replicability of cluster assignments for mapping application. *Applied Soft Computing*, 12(11):3698–3700.

Kotsiantis, S. and Pintelas, P. (2004). Recent advances in clustering: A brief survey. *WSEAS Transactions on Information Science and Applications*, 1(1):73–81.

Kou, G., Peng, Y., and Wang, G. (2014). Evaluation of clustering algorithms for financial risk analysis using mcdm methods. *Information Sciences*, 275:1–12.

Krishnasamy, G., Kulkarni, A. J., and Paramesran, R. (2014). A hybrid approach for data clustering based on modified cohort intelligence and k-means. *Expert Systems with Applications*, 41(13):6009–6016.

Liu, Z. (2011). A method of svm with normalization in intrusion detection. *Procedia Environmental Sciences*, 11:256–262.

Lloyd, S. (1982). Least squares quantization in pcm. *Least squares quantization in PCM*, 28(2):129–137.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, 1(14):281–297.

Milligan, G. W. (1989). A validation study of a variable weighting algorithm for cluster analysis. *Journal of Classification*, 6(1):53–71.

Milligan, G. W. and Cooper, M. C. (1988). A study of standardization of variables in cluster analysis. *Journal of Classification*, 5(2):181–204.

Mohamad, I. B. and Usman, D. (2013). Standardization and its effects on k-means clustering algorithm. *Research Journal of Applied Sciences, Engineering and Technology*, 6(17):3299–3303.

Mohd, W. M. B. W., Beg, A. H., Herawan, T., and Rabbi, K. F. (2012). An improved parameter less data clustering technique based on maximum distance of data and lioyd k-means algorithm. *Procedia Technology*, 1:367–371.

Nayak, S., Misra, B., and Behera, H. (2014). Impact of data normalization on stock index forecasting. *International Journal of Computational Information Systems and Industrial Management Applications*, 6:357–369.

Nazeer, K. A. and Sebastian, M. P. (2009). Improving the accuracy and efficiency of the k-means clustering algorithm. *In Proceedings of the world congress on engineering*, 1:1–3.

Noorbehbahani, F., Mousavi, S. R., and Mirzaei, A. (2015). An incremental mixed data clustering method using a new distance measure. *Soft Computing*, 19(3):731–743.

Ogasawara, E., Martinez, L. C., De Oliveira, D., ZimbrÃ£o, G., Pappa, G. L., and Mattoso, M. (2010). Adaptive normalization: A novel data normalization approach for non-stationary time series. *In the 2010 International Joint Conference on Neural Networks*, pages 1–8.

Oyelade, O. J., Oladipupo, O. O., and Obagbuwa, I. C. (2010). Application of k means clustering algorithm for prediction of students academic performance. *arXiv preprint arXiv:1002.2425*.

Patel, V. R. and Mehta, R. G. (2011). Impact of outlier removal and normalization approach in modified k-means clustering algorithm. *International Journal of Computer Science Issues*, 8(5):331.

Reddy, D., Jana, P. K., and Member, I. S. (2012). Initialization for k-means clustering using voronoi diagram. *Procedia Technology*, 4:395–400.

Rokach, L. and Maimon, O. Z. (2008). *Data mining with decision trees: Theory and applications*. Singapore: World Scientific Publishing Co. Pte. Ltd.

Ryu, T. W. and Eick, C. F. (2005). A database clustering methodology and tool. *Information Sciences*, 171(1):29–59.

Shirkhorshidi, A. S., Aghabozorgi, S., and Wah, T. Y. (2015). A comparison study on similarity and dissimilarity measures in clustering continuous data. *PloS one*, 10(12):e014405. https://doi.org/10.1371/journal.pone.0144059.

Sola, J. and Sevilla, J. (1997). Importance of input data normalization for the application of neural networks to complex industrial problems. *IEEE Transactions on Nuclear Science*, 44(3):1464–1468.

Suarez-Alvarez, M. M., Pham, D. T., Prostov, M. Y., and Prostov, Y. I. (2012). Statistical approach to normalization of feature vectors and clustering of mixed datasets. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 468(2145):2630–2651.

Tan, P., Steinbach, M., and Kumar, V. (2005). *Introduction to data mining*. England: Pearson Education Limited.

Tomar, D. and Agarwal, S. (2015). Hybrid feature selection based weighted least squares twin support vector machine approach for diagnosing breast cancer, hepatitis, and diabetes. *Advances in Artificial Neural Systems*, 2015:1–10. http://dx.doi.org/10.1155/2015/265637.

Velardi, P., Navigli, R., Faralli, S., and Ruiz-Martinez, J. M. (2012). A new method for evaluating automatically learned terminological. *In LREC*, pages 1498–1504.

Visalakshi, N. K. and Thangavel, K. (2009). Impact of normalization in distributed k-means clustering. *International Journal of Soft Computing*, 4(4):168–172.

Wu, J., Xiong, H., and Chen, J. (2009). Adapting the right measures for k-means clustering. *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 877–886.

Zumel, N. and Mount, J. (2013). Log transformations for skewed and wide distributions. *Retrieved 18 March 2014*.