MATEMATIKA, 2013, Volume 29, Number 1c, 253-267

An Efficient Dimension Reduction Technique for Basic K-Means Clustering Algorithm

[1]Dauda Usman and [2]Ismail Mohamad
[1,2]Department of Mathematical Sciences, Faculty of Science, Universiti Teknologi Malaysia
81310, UTM Johor Bahru, Johor Darul Ta'azim, Malaysia
e-mail: [1]dauusman@gmail.com, [2]ismailm@utm.my
*Corresponding author: (dauusman@gmail.com)

Abstract K-means clustering is being widely studied problem in a variety of application domains. The computational complexity of the basic k-means is very high, the number of distance calculations also increases with the increase of the dimensionality of the data. Several algorithms have been proposed to improve the performance of the basic k-means. Here we investigate the behavior of the basic k-means clustering algorithm and two alternatives to it, we have analyzed the performances of three different standardization methods. Equivalently, we prove that z-score and principal components are the best pre-processing methods that will simplify the analysis and visualize the multidimensional dataset. The analyzed result revealed that the z-score outperform min-max and decimal scaling also principal component analysis picks up the dimensions with the largest variances. Our results also provide effective ways to solve the k-means clustering problems.

Keywords Decimal Scaling; K-Means Clustering; Min-Max; Principal Component Analysis; Standardization; z-score.

2010 Mathematics Subject Classification 62H30, 68T10

1 Introduction

The purpose of clustering is to reveal the natural structure inherent by datasets and extracting useful information from noisy data. The main objective in cluster analysis is to group objects that are similar in one cluster and separate objects that are dissimilar by assigning them to different clusters. One of the most popular clustering methods is k-means clustering algorithm [1, 2]. It classifies objects to a pre-defined number of clusters, which is given by the user (assume k clusters). The idea is to choose random cluster center points, one for each cluster. These center points are preferred to be as far as possible from each other. In this algorithm mostly Euclidean distance is used to find the distance between data points and centroids [3].

The computational complexity of the basic k-means clustering algorithm is very high. In addition the number of distance calculations increases tremendously with the increase of the dimensionality of the data. When the dimensionality increases normally, just a small number of dimensions are relevant to certain clusters, but data in the insignificant dimensions may possibly produce very much noise and conceal the real clusters to be discovered. Moreover when dimensionality increases, data normally turn out to be increasingly sparse, due to which data points positioned at various dimensions can be viewed as all equally distanced and also the distance measure, which, basically for cluster analysis, turns into meaningless. However, if there are some features, with a large size or great variability, these kinds of features will strongly affect the clustering result. In this case, dimensionality reduction and data standardization are important preprocessing task to scale or control the variability of the datasets.

Preprocessing [3] is actually essential before using any data exploration algorithms to enhance the results' performance. Standardization of the dataset is among the preprocessing processes in data

exploration, in which the attribute data are scaled to fall in a small specified range. Standardization before clustering is specifically needed for distance metric, like the Euclidian distance that are sensitive to variations within the magnitude or scales from the attributes. In actual applications, due to the variations in selection of the attribute's value, one attribute might overpower another one. Standardization prevents outweighing features having a large number over features with smaller numbers. The aim would be to equalize the dimensions or magnitude and also the variability of those features. But what kind of standardization is suitable for k-means clustering algorithm. In this article, we take the advantage to compare the three different standardization methods on a basic k-means clustering algorithm.

However, dimensional datasets are sometimes transformed into lower dimension by applying principal component analysis [4] (or singular value decomposition) whereby coherent patterns could be detected more easily. This type of unsupervised dimension reduction is commonly employed in tremendously broad areas which includes meteorology, image processing, genomic analysis, and information retrieval. It is also well-known that principal component analysis can be used to project dimensional data into a reduced dimensional subspace and then k-means will then be applied to the subspace [5]. In other instances, data are embedded in a low dimensional space just like the eigenspace from the graph Laplacian, and k-means will then be employed [6].

A very important reason for principal component analysis based dimension reduction is that, it picks up the dimensions with the largest variances. Mathematically, this is equivalent to finding the best low rank approximation (in *L2 norm*) of the data applying singular value decomposition [7].

The result also provides best ways to address the basic k-means clustering problem. K-means technique employs *k* prototypes, the centroids of clusters to characterize the data. These are determined by minimizing error sum of squares.

1.1 *K*-means Clustering Algorithm

A conventional procedure for *k*-means clustering is straightforward. Getting started we can decide the amount of groups k and that we presume a centroid or center of those groups. Immediately consider any kind of random items as initial centroids or a first k items within the series which can also function as an initial centroids.

After that the k-means technique will perform the 3 stages listed here before convergence. Iterate until constant (= zero item move group):

1. Decide the centroid coordinate.
2. Decide the length of every item to the centroids.
3. Cluster the item according to minimal length.

1.2 Principal Component Analysis

The principal component analysis can be looked at mathematically as the transformation of the linear orthogonal of the data to a different coordinate so that the largest variance of any of the data projections lies on the first coordinate (known as the first principal coordinate), the next largest on the second coordinate, and so on. It transforms a numerous possibly correlated variables into a compact quantity of uncorrelated variables called principal components. principal component analysis is a statistical technique for determining key variables in a high dimensional dataset which accounts for differences in the observations and is very important for analysis and visualization where information is very little lacking.

1.3 Principal Component

Principal components can be determined by the Eigen value decomposition of a data sets correlation matrix/ covariance matrix or singular value decomposition of the data matrix, normally after mean centering the data for every feature. Covariance matrix is preferred when the variances of features are extremely large on comparison to correlation. It will be best to choose the type of correlation once the features are of various types. Likewise singular value decomposition method is employed for statistical precisions.

2 Some Related Works

Several efforts have been made by different researchers to enhance the performance as well as the efficiency of the basic k-means algorithm. Principal component analysis by [8] and [9] is known as an unsupervised Feature Reduction technique meant for projecting huge dimensional data into a new reduced dimensional representation of the data that explains as much of the variance within the data as possible with minimum error reconstruction.

Chris and Xiaofeng [10] Proved that principal components remain the continuous approaches to the discrete cluster membership indicators for k-means clustering and also, proved that the subspace spanned through the cluster centroids are given by the spectral expansion of the data covariance matrix truncated at *k-1* terms. The effect signifies that unsupervised dimension reduction is directly related to unsupervised learning. In dimension reduction, the effect gives new insights to the observed usefulness of principal component analysis based data reductions, beyond the traditional noise-reduction justification. Mapping data points into a higher dimensional space by means of kernels indicates that the solution for kernel k-means provided by kernel principal component analysis. On learning, the results suggest effective techniques for k-means clustering. In [11], principal component analysis is used to reduce the dimensionality of the data set and then the k-means algorithm is used in the principal component analysis subspaces. Executing principal component analysis is the same as carrying out singular value decomposition (singular value decomposition) on the covariance matrix of the data. Karthikeyani and Thangavel [12] employed the singular value decomposition technique to determine arbitrarily oriented subspaces with very good clustering.

Karthikeyani and Thangavel [12] extended k-means clustering algorithm by applying global normalization before performing the clustering on distributed datasets, without necessarily downloading all the data into a single site. The performance of proposed normalization based distributed k-means clustering algorithm was compared against distributed k-means clustering algorithm and normalization based centralized k-means clustering algorithm. The quality of clustering was also compared by three normalization procedures, the min-max, z-score and decimal scaling for the proposed distributed clustering algorithm. The comparative analysis shows that the distributed clustering results depend on the type of normalization procedure. Alshalabi *et al.,* [13] designed an experiment to test the effectiveness of different normalization methods for accuracy and simplicity. The experiment results suggested choosing the z-score normalization as the method that will give much better accuracy.

3 Materials and Methods

Let $Y = \{X_1, X_2, ..., X_n\}$ imply the d-dimensional raw data set.

Then the data matrix is an $n \times d$ matrix given by:

$$X_1, X_2, ..., X_n = \begin{pmatrix} a_{11} & \cdots & a_{1d} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nd} \end{pmatrix}. \quad (1)$$

3.1 *Z*-score

The *z*-score is a form of standardization used for transforming normal variants to standard score form. Given a set of raw data Y, the *z*-score standardization formula is defined as

$$x_{ij} = Z\left(x_{ij}\right) = \frac{x_{ij} - \overline{x}_j}{\sigma_j} \quad (2)$$

where $\overline{x}_j$ and σ_j are the sample mean and standard deviation of the *j*th attribute, respectively. The transformed variable will have a mean of 0 and a variance of 1. The location and scale information of the original variable has been lost [14]. One important restriction of the *z*-score standardization Z is that it must be applied in global standardization and not in within-cluster standardization [15].

3.2 Min-Max

Min-Max standardization is the process of taking data measured in its engineering units and transforming it to a value between 0.0 and 1.0. Whereby the lowest (min) value is set to 0.0 and the highest (max) value is set to 1.0. This provides an easy way to compare values that are measured using different scales or different units of measure. The standardized value is defined as:

$$MM(X_{ij}) = \frac{X_{ij} - X_{\min}}{X_{\max} - X_{\min}} \quad (3)$$

3.3 Decimal Scaling

Standardization by decimal scaling: standardizes by moving the decimal point of values of feature *X*. The number of decimal points moved depends on the maximum absolute value of *X*. A modified value *DS(X)* corresponding to *X* is obtained using:

$$DS(X_{ij}) = \frac{X_{ij}}{10^c} \quad (4)$$

where *c* is the smallest integer such that $max\ [|DS(X_{ij})|] < 1$

3.4 Principal Component Analysis

Let **v** = $\left(v_1, v_2, ..., v_d\right)'$ be a vector of *d* random variables, where $'$ is the transpose operation. The first step is to find a linear function $a_1'v$ of the elements of **v** that maximizes the variance, where a_1 is a *d*-dimensional vector $\left(a_{11}, a_{12}, ..., a_{1d}\right)'$ so

$$a_1'v = \sum_{i=1}^{n} a_{1i}v_i. \tag{5}$$

after finding $a_1'v, a_2'v, ..., a_{j-i}'v$, we look for a linear function $a_j'v$ that is uncorrelated with $a_1'v, a_2'v, ..., a_{j-i}'v$ and has maximum variance. Then we will find such linear functions after *d* steps. The jth derived variable $a_j'v$ is the jth PC. In general, most of the variation in **v** will be accounted for by the first few PCs.

To find the form of the PCs, we need to know the covariance matrix $\sum$ of **v**. In most realistic cases, the covariance matrix $\sum$ is unknown, and it will be replaced by a sample covariance matrix. That is for *j = 1, 2, . . . , d,* it can be shown that the jth PC is: $z_j = a_j'v$, where a_j is an eigenvector of $\sum$ correspond with the jth main eigenvalue λ_j.

In fact, in the first step, $z = a_1'v$ can be found by solving the following optimization problem:

Maximize $\text{var}\left(a_1'v\right)$ subject to $a_1'a = 1$,

Where $\text{var}\left(a_1'v\right)$ is computed as

$\text{var}\left(a_1'v\right) = a_1'\Sigma a_1$.

To solve the above optimization problem, the technique of Lagrange multipliers can be used. Let λ be a Lagrange multiplier. We want to maximize

$$a_1'\Sigma a_1 - \lambda\left(a_1'a - 1\right). \tag{6}$$

Differentiating Equation (6) with respect to a_1, we have

$$\Sigma a_1 - \lambda a_1 = 0,$$

or

$$\left(\Sigma - \lambda I_d\right)a_1 = 0,$$

where I_d is the $d \times d$ identity matrix.

Thus λ is an eigenvalue of Σ and a_1 is the corresponding eigenvector. Since

$$a_1' \Sigma a_1 = a_1' \lambda a_1 = \lambda ,$$

a_1 is the eigenvector corresponding with the main eigenvalue of Σ. In fact, it can be shown that the jth PC is $a_j' v$, where a_j is an eigenvector of Σ corresponding to its jth largest eigenvalue λ_j [4].

3.5 Singular Value Decomposition

Let $D = \{x_1, x_2, ..., x_n\}$ be a numerical data set in a *d*-dimensional space. Then *D* can be represented by an $n \times d$ matrix *X* as

$$X = \left(x_{ij}\right)_{n \times d} ,$$

where x_{ij} is the j-component value of x_i .

Let $\bar{\mu} = \left(\bar{\mu}_1, \bar{\mu}_2, ..., \bar{\mu}_d\right)$ be the column mean of *X,*

$$\bar{\mu}_j = \frac{1}{n} \sum_{i=1}^{n} x_{ij}, j = 1, 2, ..., d$$

and let e_n be a column vector of length n with all elements equal to one. Then SVD expresses $X - e_n \bar{\mu}$ as

$$X - e_n \bar{\mu} = USV^T \tag{7}$$

where *U* is an $n \times n$ column orthonormal matrix, i.e., $U^T U = I$ is an identity matrix, *S* is an $n \times d$ diagonal matrix containing the singular values, and *V* is a $d \times d$ unitary matrix, i.e.,

$V^H V = I$, where V^H is the conjugate transpose of *V*. The columns of the matrix *V* are the eigenvectors of the covariance matrix *C* of *X;* precisely,

$$C = \frac{1}{n} X^T X - \bar{\mu}^T \bar{\mu} = V \Lambda V^T \tag{8}$$

Since *C* is a $d \times d$ positive semi definite matrix, it has *d* nonnegative eigenvalues and *d* orthonormal eigenvectors. Without loss of generality, let the eigenvalues of *C* be ordered in decreasing order: $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_d$. Let σ_j *(j = 1, 2, . . . , d)* be the standard deviation of the jth column of *X,* i.e.,

$$\sigma_j = \left(\frac{1}{n} \sum_{i=1}^{n} (x_{ij} - \bar{\mu}_j)^2 \right)^{\frac{1}{2}}.$$

The trace Σ of C is invariant under rotation, i.e.,

$$\Sigma = \sum_{j=1}^{d} \sigma_j^2 = \sum_{j=1}^{d} \lambda_j .$$

Noting that $e_n^T X = n\bar{\mu}$ and $e_n^T e_n = n$ from Equations (7) and (8), we have

$$\begin{aligned} VS^T SV^T &= VS^T U^T USV^T \\ &= \left(X - e_n\bar{\mu}\right)^T \left(X - e_n\bar{\mu}\right) \\ &= X^T X - \bar{\mu}^T e_n^T X - X^T e_n\bar{\mu} + \bar{\mu}^T e_n^T e_n\bar{\mu} \\ &= X^T X - n\bar{\mu}^T\bar{\mu} \\ &= nV\Lambda V^T \end{aligned} \tag{9}$$

Since V is an orthonormal matrix, from Equation (9), the singular values are related to the eigenvalues by:

$$S_j^2 = n\lambda_j, j = 1, 2, ..., d .$$

The eigenvectors constitute the PCs of X, and uncorrelated features will be obtained by the transformation $Y = \left(X - e_n\bar{\mu}\right)V$. PCA selects the features with the highest eigenvalues.

3.6 K-means Clustering

Provided some series involving observations $(x_1, x_2, ..., x_n)$, in which each observation is known as a d-dimensional real vector, k-means clustering is designed to partition an n observations to k units $(k = n)$ $S = S_1, S_2, ..., S_k$ as a way to reduce the within-cluster sum of squares (WCSS):

$$\arg_S \min \sum_{i=1}^{k} \sum_{x_j \in S_i} / / x_j - \mu_i / /^2 \tag{10}$$

at which μ_i stands out as the mean for items within S_i.

4 Results and Discussions

The presence of noise in a dataset can easily be filtered out by applying data standardization and principal component analysis, since such a treatment was specifically designed to denoise both smaller and larger numerical sets of data.

In this section, we evaluate and examine the performances of three different standardization methods and principal component analysis approaches on a basic k-means clustering algorithm with infectious diseases dataset having 14 data objects and 8 attributes as shown in Table 1. The accuracy of the clustering are also evaluated, whereby accuracy is measured by the intra-cluster distance, that is a

distance between the data vectors in a cluster and the centroid of the cluster, the smaller the sum of the distances is, the better the accuracy of clustering and the error sum of squares.

Table 1 The original datasets with 14 data objects and 8 attributes

	X1	X2	X3	X4	X5	X6	X7	X8
Day 1	9	6	7	5	3	6	2	3
Day 2	16	5	5	11	4	5	1	1
Day 3	6	7	6	2	8	7	2	2
Day 4	7	3	2	2	6	3	2	2
Day 5	10	12	3	5	6	12	5	5
Day 6	13	5	13	8	10	5	4	4
Day 7	2	3	2	3	8	3	1	3
Day 8	3	2	3	3	9	2	3	3
Day 9	17	3	19	3	4	3	3	3
Day 10	8	7	1	1	5	2	1	1
Day 11	7	3	7	1	8	3	1	1
Day 12	15	9	5	5	13	9	5	5
Day 13	13	2	3	2	5	3	2	1
Day 14	6	1	7	5	4	2	1	2

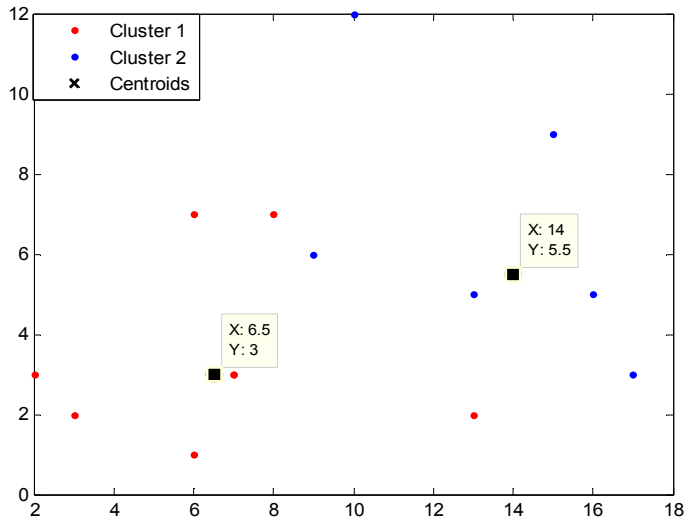

Figure 1 Basic k-means algorithm

Figure 1 presents the result of the basic k-means algorithm using the original dataset having 14 data objects and 8 attributes as shown in Table 1, with error sum of squares equal 206.00

Table 2 The variances cumulative percentages

	Variances	Percentage of Variances	Cumulative Percentage of Variances
PC1	37.9134	45.0055	45.0055
PC2	22.7305	26.9825	71.9880
PC3	10.4788	12.4390	84.4270
PC4	5.9049	7.0095	91.4365
PC5	5.2974	6.2884	97.7249
PC6	1.1322	1.3439	99.0688
PC7	0.6689	0.7940	99.8628
PC8	0.1156	0.1372	100.0000

Table 2 presents the variances, the percentage of the variances and cumulative percentage which corresponds to the principal components.

Table 3 Reduced principal components

PC1	PC2	PC3	PC4
0.6953	0.1289	0.4595	0.2590
0.1081	0.5791	-0.0802	-0.3898
0.6367	-0.4670	-0.5110	-0.2626
0.2402	0.1402	0.2707	0.3439
-0.0204	0.2346	-0.5728	0.7141
0.1554	0.5445	-0.1467	-0.2851
0.1093	0.1810	-0.1979	0.0498
0.0730	0.1614	-0.2435	0.0016

Table 3 presents the reduced principal components that have variances greater than mean variance. But the number of principal components found is the same with the number of the original data set, here we present only the eighty percent to be considered for further analysis.

Table 4 The reduced data set with 14 data objects and 4 attributes.

	X1	X2	X3	X4
Day 1	1.0256	0.1354	1.2886	-3.4995
Day 2	5.5207	1.4200	7.4900	2.2379
Day 3	-2.3292	1.9301	-3.2384	-2.1515
Day 4	-5.1939	-1.0366	1.3179	0.4293
Day 5	1.1679	10.4433	-0.3684	-3.9448
Day 6	8.2323	-0.6884	-3.5493	2.7680
Day 7	-8.5072	-1.0916	-1.9002	0.8582
Day 8	-7.2406	-1.9567	-2.6934	2.3434
Day 9	13.0461	-7.6722	-1.7983	-2.4772
Day 10	-5.2604	0.6139	2.8580	-1.4327

Day 11	-2.4738	-3.3848	-2.2117	0.1494
Day 12	4.9839	8.4253	-2.4217	3.8489
Day 13	-0.5462	-1.7050	4.4606	1.3948
Day 14	-2.4252	-5.4327	0.7662	-0.5242

Table 4 presents the transformed data set having 14 data objects and 4 attributes which are generated using the reduced principal component analysis and the original data set shown in Table 3 and 1 respectively.

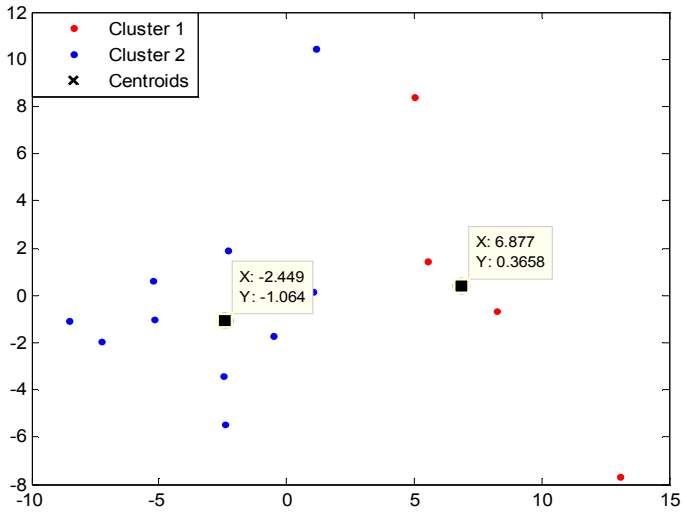

Figure 2 K-means applying principal component analysis

Figure 2 presents the result of the k-means algorithm applying a principal component analysis to the original dataset. The reduced datasets containing 14 data objects and 4 attributes as shown in Table 4, with error sum of squares equal 147.68

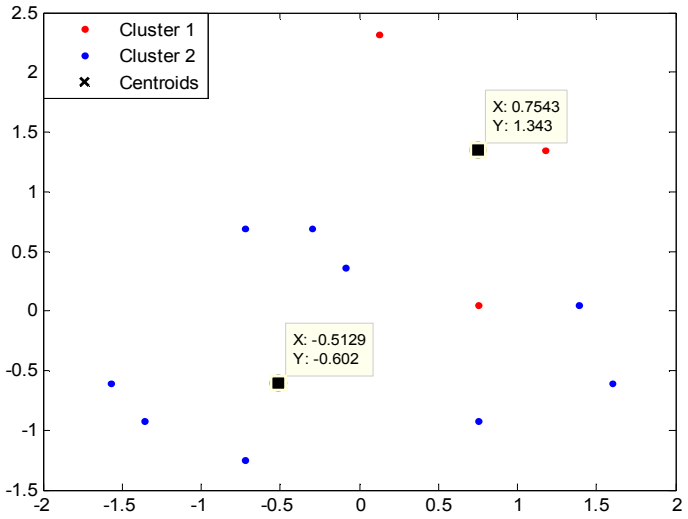

Figure 3 K-means with z-score standardized dataset

Figure 3 presents the result of the k-means algorithm using the standardized dataset with z-score method, having 14 data objects and 8 attributes with error sum of squares equal 65.17

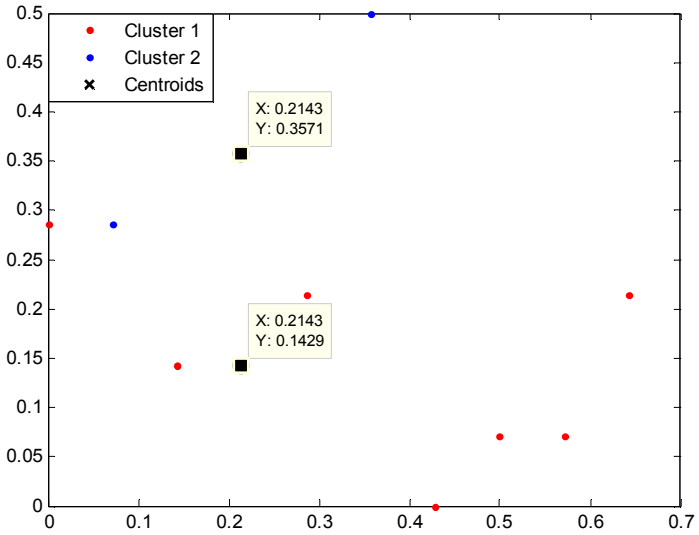

Figure 4 K-means with min-max standardized dataset

Figure 4 presents the result of the k-means algorithm using the rescale dataset with min-max data standardization method, having 14 data objects and 8 attributes with error sum of squares equal 10.94

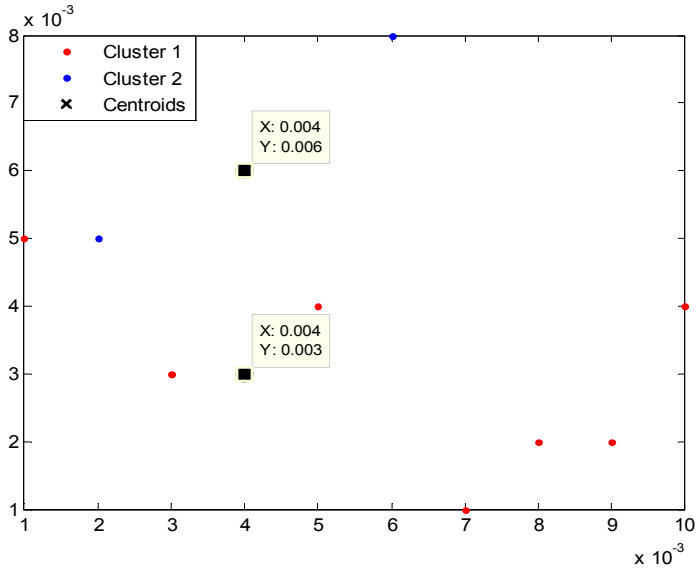

Figure 5 K-means with decimal scaling standardized dataset

Figure 5 presents the result of the k-means algorithm using the rescale dataset with decimal scaling data standardization method, having 14 data objects and 8 attributes with error sum of squares equal 0.197, converted to 197.00

Table 5 The variances cumulative percentages

	Variances	Percentage of Variances	Cumulative Percentage of Variances
PC1	3.6476	45.5953	45.5953
PC2	1.7497	21.8707	67.4660
PC3	1.0900	13.6248	81.0908
PC4	0.6704	8.3800	89.4708
PC5	0.4658	5.8228	95.2936
PC6	0.2425	3.0317	98.3253
PC7	0.0861	1.0766	99.4019
PC8	0.0478	0.5981	100.0000

Table 5 presents the variances, the percentage of the variances and cumulative percentage which corresponds to the principal components.

Table 6 Reduced PCs with variances greater than mean variance.

PC1	PC2	PC3	PC4
0.2398	0.5687	-0.0831	0.1394
0.4136	-0.1814	-0.4261	0.2683
0.0983	0.5683	0.4766	0.2956
0.2164	0.4178	-0.2873	-0.7842
0.2525	-0.3467	0.5175	-0.4163
0.4630	-0.1105	-0.3298	0.1333
0.4796	-0.0392	0.2431	0.1177
0.4548	-0.1103	0.2571	-0.0369

Table 6 presents the reduced principal components that have variances greater than mean variance. But the number of principal components found is the same with the number of the original dataset, here we present only the eighty percent to be considered for further analysis.

Table 7 The reduced dataset with 14 data objects and 4 attributes.

	X1	X2	X3	X4
Day 1	0.1360	0.5295	-0.9484	0.4306
Day 2	-0.2681	2.1891	-1.9860	-1.4727
Day 3	0.1445	-1.0275	-0.2006	0.5407
Day 4	-1.2233	-0.7455	0.0178	0.0953
Day 5	3.7568	-1.0019	-1.4293	0.7553
Day 6	2.0177	1.2608	1.3165	-0.9561
Day 7	-1.2263	-1.4947	0.3862	-0.7351
Day 8	-0.6925	-1.3390	1.2353	-0.7616
Day 9	0.1819	2.7332	1.3918	1.4944
Day 10	-1.6357	-0.8606	-0.9725	0.7398
Day 11	-1.6768	-0.4499	0.6264	0.3290
Day 12	3.8091	-0.7409	0.7115	-0.4053
Day 13	-1.4481	0.3528	-0.2187	0.4200
Day 14	-1.8753	0.5946	0.0699	-0.4744

Table 7 presents the transformed data set having 14 data objects and 4 attributes which are generated using the reduced principal component analysis and z-score standardized dataset.

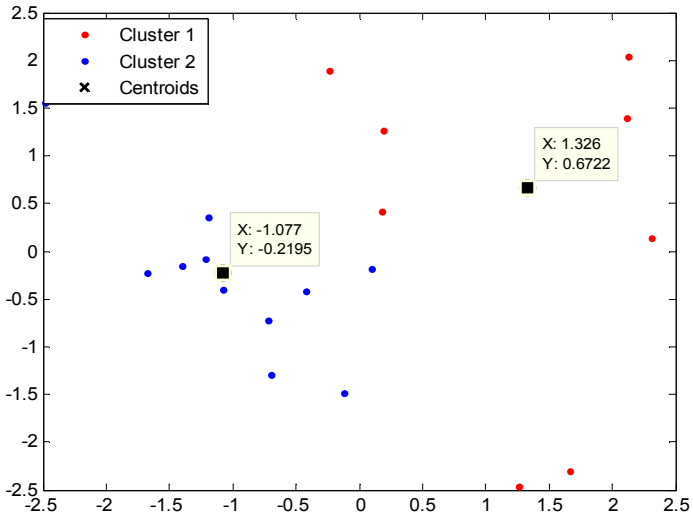

Figure 6 K-means applying standardization and principal component analysis

Figure 6 presents the result of the k-means algorithm applying z-score standardization method and principal component analysis to the original dataset. The reduced datasets containing 14 data objects and 4 attributes shown in Table 7 with error sum of squares equal 46.39.

Table 8 Summary of the results for cluster formations

	Cluster 1 points out	Cluster 2 points out	ESSs
Basic *K*-means	1	1	206.00
K-means with PCA	0	0	147.68
K-means with Z-score	0	0	65.18
K-means with Min-Max	2	1	10.94
K-means with Decimal Scaling	3	1	197.00
K-means with z-score and PCA	0	0	46.39

Table 8 shows the number of points that are out of cluster formations for both cluster 1 and cluster 2. The total error sum of squares for all the methods employed, indicating the accuracy of the z-score standardization method and principal component analysis.

5 Conclusion

In this work, two pre-processing methods and unsupervised clustering are studied for low dimensional datasets to avoid clustering with redundant data so as to improve the quality of clustering techniques. The methods were tested with infectious diseases datasets and analysis the performance of cluster values using the within cluster scatter for cluster i, which has to be as low as possible and also the separation between the i^{th} and the j^{th} cluster, which ideally has to be as large as possible shown in Figure 4 with the error sum

of squares equal 46.39 which is the minimum among all the cluster formations. This also shows the efficiency and effectiveness of the techniques.

References

[1] Kohei, A. and Ali R. B. Hierarchical K-Means: An algorithm for centroids initialization for K-Means. *Reports of the Faculty of Science and Engineering, Saga University*. 2007. 36: 25-31.

[2] De, K. R. and Bhattacharya, A. "Divisive Correlation Clustering Algorithm (DCCA) for grouping of genes: Detecting varying patterns in expression profiles," *Bioinformatics*. 2008. 24: 1359-1366.

[3] Luai, A. S., Zyad, S. and Basel, K. (2006). Data mining: A preprocessing engine, *Journal of Computer Science*, 2(9): 735-739.

[4] Jolliffe, I. *Principal Component Analysis, 2nd edition*. Springer Series in Statistics. 2002. New York: Springer-Verlag.

[5] Zha, H., Ding, C., Gu, M., He, X., and Simon, H. Spectral relaxation for K-means clustering. *Advances in Neural Information Processing Systems (NIPS'01)* 14. 2002. 1057-1064.

[6] Ng, A., Jordan, M. and Weiss, Y. On spectral clustering: analysis and an algorithm. *Proc. Neural Info. Processing Systems*. (NIPS 2001). 2001.

[7] Eckart, C., and Young, G. The approximation of one matrix by another of lower rank psychometrika. 1936. 1: 183-187.

[8] Valarmathie, P., Srinath, M and Dinakaran, K. An increased performance of clustering high dimensional data through dimensionality reduction technique. *Journal of Theoretical and Applied Information Technology*. 2009. 13: 271-273.

[9] Yan, J., Zhang, B., Liu, N., Yan S., Cheng Q., Fan W., Yang, Q., Xi., W. and Chen Z. Effective and efficient dimensionality reduction for large scale and streaming data preprocessing, *IEEE transactions on Knowledge and Data Engineering*. 2006. 18(3): 320-333.

[10] Chris, D. and Xiaofeng, H. K-means clustering via principal component analysis. *Proc. of the 21st International Conference on Machine Learning*. 2006. Banff, Canada.

[11] Ding, C. and He, X. *K*-means clustering via principal component analysis. *Twenty-first International Conference on Machine Learning*. 2004. New York: ACM Press.

[12] Karthikeyani, V. N. and Thangavel, K. Impact of normalization in distributed K-means clustering. *International Journal of Soft Computing*. 2009. 4(4): 168-172.

[13] Alshalabi, L., Shaaban, Z. and Kasasbeh, B.. Data mining: A preprocessing engine. *Journal of Computer Science*. 2006. 2(9): 735-739.

[14] Jain, A. and Dubes, R. *Algorithms for Clustering Data*. 1988. Prentice Hall.

[15] Milligan, G. and M. Cooper. A study of standardization of variables in cluster analysis. *Journal of Classification*. 1988. 5: 181-204.