Research Journal of Applied Sciences, Engineering and Technology 7(4): 771-777, 2014
DOI:10.19026/rjaset.7.315
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: April 04, 2013 Accepted: April 22, 2013 Published: January 27, 2014

Research Article

Outlier Removal Approach as a Continuous Process in Basic *K*-Means Clustering Algorithm

[1]Dauda Usman and [2]Ismail Bin Mohamad
[1, 2]Department of Mathematical Sciences, Faculty of Science, Universiti Teknologi Malaysia, 81310, UTM Johor Bahru, Johor Darul Ta'azim, Malaysia

Abstract: Clustering technique is used to put similar data items in a same group. K-mean clustering is a commonly used approach in clustering technique which is based on initial centroids selected randomly. However, the existing method does not consider the data preprocessing which is an important task before executing the clustering among the different database. This study proposes a new approach of k-mean clustering algorithm. Experimental analysis shows that the proposed method performs well on infectious disease data set when compare with the conventional k-means clustering method.

Keywords: Infectious diseases, *k*-means clustering, principal component analysis, principal components, standardization

INTRODUCTION

Data analysis techniques are necessary on studying actually increasing huge range of large sizing data. Regarding the same edge, cluster analysis (Hastie *et al*., 2001) tries to pass through data easily to achieve 1st structure experience by dividing data items straight into disjoint classes in a way that data items owned by identical cluster are the same whereas data items owned by another clusters tend to be different. Among the significant well known as well as effective clustering techniques is known as the K-means technique (Hartigan and Wang, 1979) utilizing prototypes (centroids) so as to signify clusters through perfecting the error sum squared operation. (The specifics report for K-means as well as relevant techniques has been provided in (Jain and Dubes, 1988).

The computational difficulty with traditional K-means algorithm is extremely large, specifically with regard to huge data units. Moreover the amount of distance computations rises greatly with the increase with the dimensionality of the data. When the dimensionality increases usually, just a few dimensions are highly relevant to specific clusters, however data on the unimportant dimensions may possibly generate extremely very much noise and also conceal the true clusters that will possibly be observed. Furthermore whenever dimensionality elevates, data normally turn out to be extremely short, data elements positioned on separate measurements may be regarded virtually all equally distanced as well as the distance amount, that, primarily for grouping exploration, turns into useless. Therefore, feature reduction or just dimensionality lessening is the central data-preprocessing approach regarding cluster analysis for datasets which has a huge number of features.

However, huge dimensional data are sometimes enhanced into reduce dimensional data through Principal Component Analysis (PCA) (Jolliffe, 2002) (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 additionally well-known that PCA can be used to project data into a reduced dimensional subspace and then K-means will then be applied to the subspace (Zha *et al*., 2002). 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 (Ng *et al*., 2001).

A very important reason for PCA reliant dimension lowering is that often it holds the dimensions considering the main variances. This is the same with locating the optimal low rank approximation (in L2 norm) for the data employing the SVD (Eckart and Young, 1936). Also, the dimension lowering property on its own is actually inadequate in order to elucidate the potency of PCA.

On this study, we take a look at the link concerning both of these frequently used approaches and also a data standardization process. We show that principal component analysis and standardization approaches are basically the continuous solution for the cluster

Corresponding Author: Dauda Usman, Department of Mathematical Sciences, Faculty of Science, Universiti Teknologi Malaysia, 81310, UTM Johor Bahru, Johor Darul Ta'azim, Malaysia

membership indicators on the *K*-means clustering technique, i.e., the PCA dimension reduction automatically executes data clustering in line with the K-means objective function. This gives an essential justified reason of PCA-based data reduction.

The result also provides best ways to address the *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.

***K*-means clustering algorithm:** A conventional procedure for *k*-means clustering is straightforward. Getting started we can decide 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):

- Decide the centroid coordinate
- Decide the length of every item to the centroids
- Cluster the item according to minimal length

Principal component analysis: PCA 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 lie 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. PCA 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.

Principal component: Principal components can be determined by the Eigen value decomposition of a data sets correlation matrix/covariance matrix or SVD 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 SVD method is employed for statistical precisions.

LITERATURE REVIEW

Many efforts have been made by researchers to enhance the performance as well as efficiency of the traditional k-means algorithm. Principal Component Analysis by Valarmathie *et al*. (2009) and Yan *et al*. (2006) 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 (2006) 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 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 PCA-based data reductions, beyond the traditional noise-reduction justification. Mapping data points right into a higher dimensional space by means of kernels, indicates that solution for Kernel *K*-means provided by Kernel PCA. In learning, final results suggest effective techniques for *K*-means clustering. In (Ding and He, 2004), PCA is used to reduce the dimensionality of the data set and then the k-means algorithm is used in the PCA subspaces. Executing PCA is the same as carrying out Singular Value Decomposition (SVD) on the covariance matrix of the data. Karthikeyani and Thangavel (2009) Employs the SVD technique to determine arbitrarily oriented subspaces with very good clustering.

Karthikeyani and Thangavel (2009) 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*. (2006) designed an experiment to test the effect of different normalization methods on accuracy and simplicity. The experiment results suggested choosing the z-score normalization as the method that will give much better accuracy.

Removal of the weaker principal components: The transformation on the data set to the new principal component axis provides the number of PCs same as the number in the initial features. Although for various data sets, the first few PCs mention most of the variances and so the others can easily be eliminated with minimum loss of information.

MATERIALS AND METHODS

Let $Y = \{X_1, X_2, \ldots, X_n\}$ imply the d-dimensional raw data set.
Then the data matrix is an $n \times d$ matrix given by:

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

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:

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

where, $\bar{x}_j$ 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 (Jain and Dubes, 1988). One important restriction of the *z*-score standardization Z is that it must be applied in global standardization and not in within-cluster standardization (Milligan and Cooper, 1988).

Principal component analysis: Let $v = (v_1, v_2, ..., v_d)'$ 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 α_1 is a *d*-dimensional vector $(a_{11}\ a_{12}, \dots, a_{1d})'$ so:

$$a_1'v = \sum_{i=1}^{n} a_{1i}v_i. \quad (3)$$

After finding $a_1'v, a_2'v, \dots, a_{j-1}'v$, we look for a linear function $a_1'v$ that is uncorrelated with $a_1'v, a_2'v, \dots, a_{j-1}'v$ and has maximum variance. Then we will find such linear functions after *d* steps. The j^{th} derived variable $\acute{\alpha}_j$v is the j^{th} 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 Σ of v. In most realistic cases, the covariance matrix Σ 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 j^{th} PC is: $z = a_1'v$, where a_j is an eigenvector of Σ correspond with the j^{th} main eigenvalue λ_j.

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

Maximize var ($\acute{\alpha}_1$v) subject to $\acute{\alpha}_1 a = 1$,
where, var ($\acute{\alpha}_1$v) is computed as:
var ($\acute{\alpha}_j$ v) = $\acute{\alpha}_j\ \Sigma\ a_1$

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

$$a_1'\Sigma a_1 - \lambda\left(a_1'a - 1\right) \quad (4)$$

Differentiating Eq. (4) 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 j^{th} PC is $a_1'v$, where a_j is an eigenvector of Σ corresponding to its j^{th} largest eigenvalue λ_j (Jolliffe, 2002).

Singular value decomposition: Let $D = \{x_1, x_2, \dots, x_n\}$ 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 $\bar{\mu} = (\bar{\mu}_1, \bar{\mu}_2, \dots, \bar{\mu}_d)$ 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:

$$X - e_n\bar{\mu} = USV^T \quad (5)$$

where, *U* is an $n \times n$ column orthonormal matrix, i.e., $U^TU = I$ 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., $V^HV = I$, 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:

$$C = \frac{1}{n}X^TX - \bar{\mu}^T\bar{\mu} = V\Lambda V^T \quad (6)$$

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: $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d$. Let σ_j (j = 1, 2, …, d) be the standard deviation of the j^{th} 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 $e^T_n X = n\bar{\mu}$ and $e^T_n e_n = n$ from Eq. (5) and (6), we have:

$$VS^T SV^T = VS^T U^T USV^T$$

$$= (X - e_n\bar{\mu})^T (X - e_n\bar{\mu})$$

$$= X^T X - \bar{\mu}^T e_n^T X - X^T e_n \bar{\mu} + \bar{\mu}^T e_n^T e_n \bar{\mu}$$

$$= X^T X - n\bar{\mu}^T \bar{\mu}$$

$$= nV\Lambda V^T \quad (7)$$

Since V is an orthonormal matrix, from Eq. (7), the singular values are related to the eigenvalues by:

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

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

K-means clustering: Provided some series involving observations ($x_1, x_2, \ldots, x_n$), 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 ($k = n$) $S = S_1, S_2, \ldots, S_k$ as a way to reduce the Within-Cluster Sum of Squares (WCSS):

$$\arg_S \min \sum_{i=1}^{k} \sum_{x_j \in S_i} // x_j - \mu_i //^2 \quad (8)$$

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

RESULTS AND DISCUSSION

The presence of noise in a large amount of data is easily filtered out by the normalization and PCA/SVD preprocessing stages, especially since such a treatment was specifically designed to denoise large numerical values while preserving edges.

In this section, we examine as well as evaluate the tasks for the approaches below: conventional k-means with the original dataset, k-means with normalized dataset, k-means with PCA/SVD dataset and k-means with normalized and PCA/SVD dataset seeing as methods of response to the goal intent behind the *k*-means technique. The level of a particular clustering are as well be evaluated, whereby level is analyzed with the error sum of squares for the intra-cluster range, that is a range among data vectors in a group as well as the centroid for the group, the lesser the sum of the

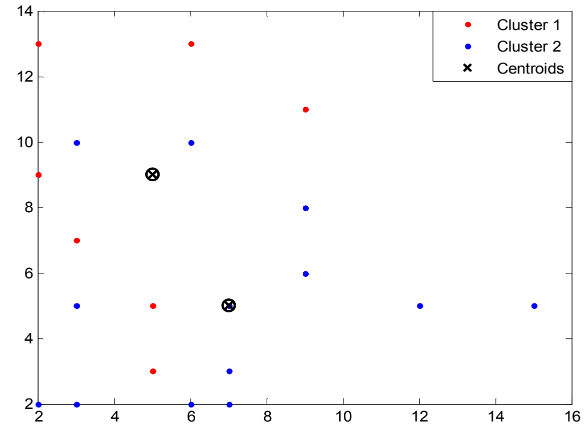

Fig. 1: Basic K-means algorithm

differences is, the better the accuracy of clustering and the error sum of squares.

Figure 1 presents the result of the basic K-means algorithm using the original dataset having 20 data objects and 7 attributes as shown in Table 1. Two points attached to cluster 1 and four points attached to cluster 2 are out of the cluster formation with the error sum of squares equal 211.21.

The number of PCs found is in fact same with the actual number of initial features. To remove the weakened components out of the PC set we worked out the corresponding variance, its percentage and cumulative percentage, shown in Table 2 and 6. There after we considered the PCs with variances lower than the mean variance, disregarding others. The lessened PCs are shown in Table 3 and 7.

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

Figure 2 explained the pareto plot of for the variances percentages against the principal component for the original dataset having 20 data objects and 7 variables.

The improve matrix using lessened PCs has been made this also transformed matrix is simply employed on the initial dataset to generate a different lessened estimated dataset, that will be utilized for the remaining data exploration and also reduced dataset containing 4 attributes is also shown in Table 4.

Figure 3 presents the result of the *K*-means algorithm applying principal component analysis to the original dataset. The reduced datasets containing 20 data objects and 4 attributes as shown in Table 4 and all the points attached to both cluster 1 and 2 are within the cluster formation with the error sum of squares equal 143.14.

Figure 4 presents the result of the *K*-means algorithm using the rescale dataset with z-score standardization method, having 20 data objects and 7 attributes as shown in Table 5. All the points

Table 1: The original datasets with 20 data objects and 7 attributes

	X1	X2	X3	X4	X5	X6	X7
Day 1	3	6	7	1	2	1	5
Day 2	4	5	5	3	1	2	1
Day 3	8	7	6	2	2	3	2
Day 4	6	3	2	1	1	1	2
Day 5	6	12	3	3	3	2	5
Day 6	10	5	13	1	1	2	4
Day 7	8	3	2	3	2	1	3
Day 8	9	2	3	7	2	4	3
Day 9	4	3	2	1	2	1	3
Day 10	5	7	1	2	1	2	1
Day 11	8	3	7	1	1	3	1
Day 12	13	9	5	4	3	2	5
Day 13	11	3	4	3	1	1	5
Day 14	8	2	1	9	2	1	2
Day 15	7	3	1	2	1	2	3
Day 16	12	11	3	4	2	1	4
Day 17	9	4	1	7	1	3	2
Day 18	18	3	2	2	1	1	1
Day 19	12	8	3	8	1	2	1
Day 20	7	5	7	4	2	1	3

Table 2: The variances cumulative percentages

	Variances	Percentage of variances	Cumulative percentage of variances
PC1	17.0108	30.2768	30.2768
PC2	14.5370	25.8738	56.1506
PC3	11.8918	21.1658	77.3164
PC4	6.2813	11.1799	88.4963
PC5	4.5518	8.1016	96.5979
PC6	1.3865	2.4678	99.0657
PC7	0.5249	0.9343	100.0000

Table 3: Reduced PCs with variances greater than mean variance

PC1	PC2	PC3	PC4
-0.4098	-0.7136	0.2094	-0.4792
0.7357	-0.3791	-0.2958	-0.2609
-0.1232	0.0638	0.4822	0.1758
-0.3600	-0.2979	-0.3529	0.4073
0.2261	0.1115	0.6276	-0.2034
-0.2945	0.4878	-0.3377	-0.6552
0.0902	0.0620	-0.0611	0.1868

Table 4: The reduced data set with 20 data objects and 4 attributes

	X1	X2	X3	X4
Day 1	-3.4812	-3.0173	-2.1682	0.1004
Day 2	-1.9385	-2.6762	-0.3191	3.3180
Day 3	-4.1721	-0.5174	-0.5050	2.4220
Day 4	1.7915	-1.7384	-4.5129	-0.8758
Day 5	-0.7461	6.6411	-2.4782	-4.1123
Day 6	-3.2393	-2.1040	3.4439	-4.6208
Day 7	1.1282	2.0127	-3.6750	0.2451
Day 8	-4.4465	2.6318	-1.9869	-0.3308
Day 9	-0.2875	2.6619	7.8169	2.3143
Day 10	3.1641	2.3946	2.3333	0.7282
Day 11	-6.4781	-8.3654	1.2773	-0.0041
Day 12	4.6517	1.2276	-0.8176	3.1338
Day 13	-2.4837	2.6939	2.3137	-2.9135
Day 14	-3.8746	6.3476	-3.0006	-1.0623
Day 15	8.1750	-0.2952	-4.3566	0.6056
Day 16	3.6607	-5.2778	0.3021	-1.3892
Day 17	-0.4212	-4.4088	-2.7642	-0.6398
Day 18	8.4253	-2.7439	2.3826	-2.2940
Day 19	-1.7364	2.4879	-0.3124	5.4735
Day 20	2.3086	2.0453	7.0267	-0.0983

attached to both cluster 1 and 2 are within the cluster formation with the error sum of squares equal 65.57.

Table 6 presents the variances, the percentage of the variances and cumulative percentage which corresponds to the principal components.

The improve matrix using lessened PCs (Table 7) manufactured this also transformed matrix simply employed on a standardized dataset so as to generate different lessened estimated dataset, that will be utilized for the remaining data exploration and the lessened dataset containing 4 attributes shown in Table 8.

Figure 5 presents the result of the K-means algorithm applying standardization and principal component analysis to the original dataset. The reduced datasets containing 20 data objects and 4 attributes as shown in Table 8 and all the points attached to both cluster 1 and 2 are within the cluster formation with the error sum of squares equal 51.26.

Table 5: The standardized dataset with 20 data objects and 7 attributes

	X1	X2	X3	X4	X5	X6	X7
Day 1	0.8425	-0.0820	-0.1378	1.6390	-0.5523	0.2773	0.1442
Day 2	0.2713	-0.3554	0.2559	0.8677	-0.9332	-1.0276	-0.8173
Day 3	0.2713	-1.1756	-0.5316	0.8677	-0.5523	-0.3751	0.1442
Day 4	-0.0143	1.0116	-0.9253	0.4821	-0.9332	0.2773	0.1442
Day 5	-0.8711	-0.3554	-0.5316	-1.0605	-0.1714	2.2345	-0.8173
Day 6	1.6993	-0.3554	0.6497	-0.6749	0.5904	0.6035	-0.8173
Day 7	-0.8711	1.0116	1.4372	0.8677	-0.9332	1.2559	1.1058
Day 8	-0.0143	-1.1756	-0.5316	0.0964	-0.9332	0.9297	0.1442
Day 9	-0.2999	-0.9022	2.6184	-1.0605	1.3522	-0.7014	1.1058
Day 10	-0.8711	0.1914	0.6497	-1.0605	0.5904	-0.3751	-0.8173
Day 11	2.5560	-0.3554	1.0434	1.6390	-0.9332	-0.7014	-0.8173
Day 12	-1.1566	0.7382	0.2559	-0.2892	-0.1714	-0.7014	3.0289
Day 13	0.2713	-0.9022	-0.1378	-0.6749	0.9713	0.9297	-0.8173
Day 14	-0.8711	-1.1756	-0.5316	0.0964	-0.5523	1.9083	0.1442
Day 15	-1.1566	1.8317	-1.3191	-1.0605	-0.9332	-0.7014	1.1058
Day 16	0.8425	1.2850	-0.5316	-0.2892	0.2095	-1.0276	-0.8173
Day 17	0.8425	0.4648	-1.3191	0.8677	-0.5523	-0.3751	-0.8173
Day 18	-0.0143	1.8317	-0.9253	-1.0605	2.1141	-1.0276	0.1442
Day 19	-1.1566	-1.1756	-0.5316	1.2534	0.2095	-0.7014	0.1442
Day 20	-0.2999	-0.3554	1.0434	-1.4462	2.1141	-0.7014	-0.8173

Table 6: The variances cumulative percentages

	Variances	Percentage of variances	Cumulative percentage of variances
PC1	1.9368	27.6685	27.6685
PC2	1.6162	23.0892	50.7577
PC3	1.3526	19.3232	70.0809
PC4	1.1089	15.8412	85.9221
PC5	0.5407	7.7248	93.6469
PC6	0.2552	3.6451	97.2920
PC7	0.1896	2.7080	100.0000

Table 7: Reduced PCs with variances greater than mean variance

PC1	PC2	PC3	PC4
-0.3938	0.4602	-0.3783	0.0452
0.2573	-0.3475	-0.5667	-0.1679
0.1739	0.3771	0.2291	0.6772
-0.6084	-0.1400	-0.1310	0.3352
0.5300	0.4492	-0.0348	-0.0901
-0.1839	-0.1011	0.6771	-0.3459
0.2523	-0.5419	0.0803	0.5206

Table 8: The reduced dataset with 20 data objects and 4 attributes

	X1	X2	X3	X4
Day 1	-1.6813	-0.2195	-0.3000	0.5369
Day 2	-1.1936	0.3510	-0.6851	0.5501
Day 3	-1.2170	-0.0769	0.1050	0.3951
Day 4	-0.6975	-1.2999	-0.6112	-0.5723
Day 5	0.0965	-0.1893	2.0014	-1.8781
Day 6	-0.2414	1.8921	0.1182	-0.3372
Day 7	-0.1213	-1.4772	0.9434	1.2802
Day 8	-1.0771	-0.4033	1.2108	-0.2933
Day 9	2.1110	1.3905	0.9302	2.2522
Day 10	1.3261	0.6722	0.1687	-0.3360
Day 11	-2.4857	1.5585	-1.2492	1.3324
Day 12	1.6681	-2.2993	-0.1100	1.7351
Day 13	0.1853	1.2661	0.9955	-0.9904
Day 14	-0.7177	-0.7254	2.1843	-0.7049
Day 15	1.2560	-2.4654	-1.1173	-0.7063
Day 16	0.1761	0.4221	-1.8994	-0.7236
Day 17	-1.3994	-0.1600	-1.2983	-0.8884
Day 18	2.3070	0.1316	-1.8635	-1.0502
Day 19	-0.4255	-0.4128	0.3473	0.5040
Day 20	2.1311	2.0448	0.1292	-0.1054

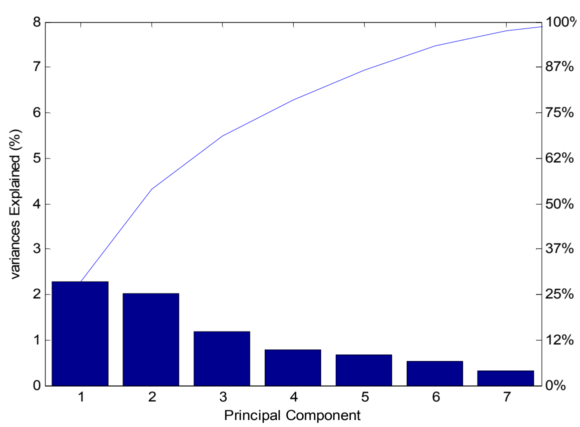

Fig. 2: Pareto plot of variances and principal components

CONCLUSION

We have proposed a novel hybrid numerical algorithm that draws on the speed and simplicity of *k*-

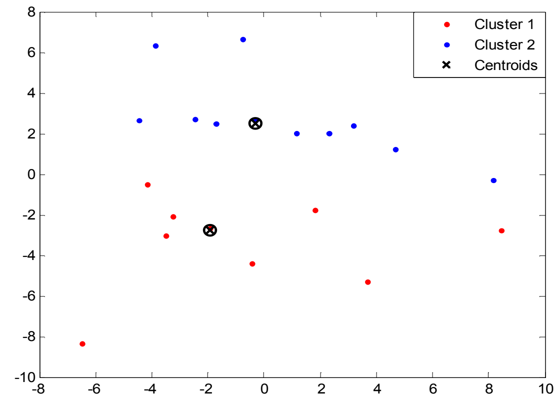

Fig. 3: K-means with PCA/SVD

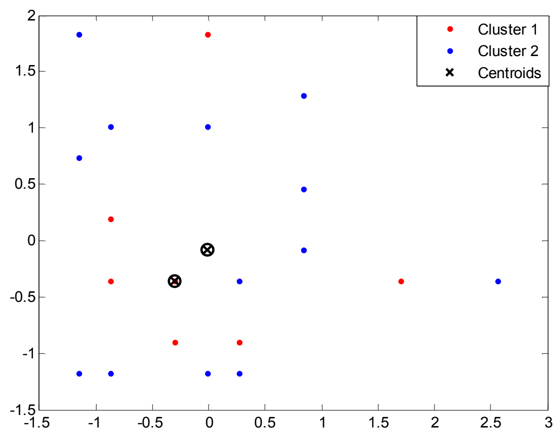

Fig. 4: K-means algorithm with standardized dataset

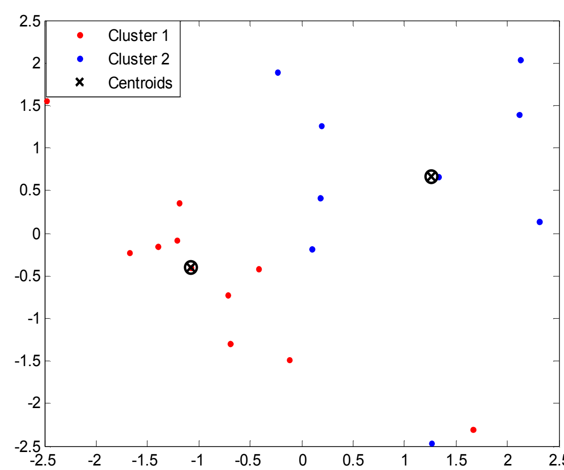

Fig. 5: K-means with rescaled and PCA/SVD datasets

means procedures. The result of the cluster analysis shown in Fig. 1 to 5 by using the basic k-means algorithm with the original data set, *k*-means clustering algorithm applying principal component analysis to the original dataset, *k*-means clustering algorithm with the standardized data set and proposed *k*-means clustering algorithm to the reduced data set respectively, shows the continuity solutions of the *k*-means clustering technique and guarantees the time reduction for clustering as a result of smaller number of features. Also in comparison the results of the analysis obtained by the standard k-means algorithm with the proposed k-means algorithm the sum of squares error are 211.21, 143.14, 65.57 and 51.26 respectively. This also shows the reliability as well as efficiency of the presented k-means technique.

REFERENCES

Alshalabi, L., Z. Shaaban and B. Kasasbeh. 2006. Data mining: A preprocessing engine. J. Comput. Sci., 2(9): 735-739.

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

Ding, C. and X.X. He, 2004. *K*-means clustering via principal component analysis. Proceeding of the 21st International Conference on Machine Learning. ACM Press, New York.

Eckart, C. and G. Young, 1936. The approximation of one matrix by another of lower rank. Psychometrika, 1: 211-218.

Hartigan, J. and M. Wang, 1979. A K-means clustering algorithm. Appl. Stat., 28:100-108.

Hastie, T., R. Tibshirani and J. Friedman, 2001. Elements of Statistical Learning. Springer Verlag, New York.

Jain, A. and R. Dubes, 1988. Algorithms for Clustering Data. Prentice Hall, New York.

Jolliffe, I., 2002. Principal Component Analysis. 2nd Edn., Springer Series in Statistics. Springer-Verlag, New York.

Karthikeyani, V.N. and K. Thangavel, 2009. Impact of normalization in distributed k-means clustering. Int. J. Soft Comput., 4(4): 168-172.

Milligan, G. and M. Cooper, 1988. A study of standardization of variables in cluster analysis. J. Classif., 5: 181-204.

Ng, A., M. Jordan and Y. Weiss, 2001. On spectral clustering: Analysis and an algorithm. Proceeding of the Neural Information Processing Systems (NIPS 2001).

Valarmathie, P., M. Srinath and K. Dinakaran, 2009. An increased performance of clustering high dimensional data through dimensionality reduction technique. J. Theor. Appl. Inform. Technol., 13: 271-273.

Yan, J., B. Zhang, , N. Liu, S. Yan, Q. Cheng, W. Fan, Q. Yang, W. Xi and Z. Chen, 2006. Effective and efficient dimensionality reduction for large scale and streaming data preprocessing. IEEE T. Knowl. Data Eng., 18(3): 320-333.

Zha, H., C. Ding, M. Gu, X. He and H. Simon, 2002. Spectral relaxation for K-means clustering. Neu. Inf. Pro. Syst., 14: 1057-1064.