International Journal of Mathematical Archive-2(11), 2011, Page: 2480-2486
Available online through www.ijma.info ISSN 2229 – 5046

A LINEAR TRANSFORMATION FOR DIMENSIONALITY REDUCTION IN HIGH DIMENSIONAL DATASETS USING PRINCIPLE COMPONENT ANALYSIS

D. NAPOLEON
Assistant Professor, Department of Computer Science, School of Computer Science and Engineering
Bharathiar University, Coimbatore-641046, India
E-mail: mekaranapoleon@yahoo.co.in

S. SATHYA
Research scholar, Department of Computer Science, School of Computer Science and Engineering
Bharathiar University, Coimbatore-641046, India
E-mail: selvarajsathya72@gmail.com

M. PRANEESH*
Research scholar, Department of Computer Science, School of Computer Science and Engineering
Bharathiar University, Coimbatore-641046, India
E-mail: raja.praneesh@gmail.com

M. SIVASUBRAMANI
Research scholar, Department of Computer Science, School of Computer Science and Engineering
Bharathiar University, Coimbatore-641046, India
E-mail: sivasu4all@gamil.com

(Received on: 31-10-11; Accepted on: 12-11-11)

ABSTRACT

Data mining is the use of automated data analysis techniques to uncover previously undetected relationships among data items. Due to the mega high dimensionality nature of datasets, data dimension reduction has drawn special attention for such type of data analysis. Data Reduction can be viewed as preprocessing step which removes distracting variance from the datasets so that clustering, classifiers can estimators perform better. In this paper principal component analysis, a linear transformation is used for dimensionality reduction and clustering with K-Medoids algorithm is applied and shows the results.

Key words: *Principal component analysis, dimensional reduction, K-Medoids clustering.*

I. INTRODUCTION:

Data Mining refers to the mining or discovery of new information in terms of patterns or rules from vast amounts of data. Data mining is a process that takes data as input and outputs knowledge. One of the earliest and most cited definitions of the data mining process, which highlights some of its distinctive characteristics, is provided by Fayyad, Piatetsky-Shapiro and Smyth (1996), who define it as "the nontrivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data."Some popular and widely used data mining clustering techniques such as hierarchical, K-Means and K-Medoids clustering techniques are statistical techniques and can be applied on high dimensional datasets [2]. A good survey on clustering methods is found in Xu *et al*. (2005).High dimensional data are often transformed into lower dimensional data via the principal component analysis (PCA)(Jolliffe, 2002) (or singular value decomposition) where coherent patterns can be detected more clearly [4]. Such unsupervised dimension reduction is used in very broad areas such as meteorology, image processing, genomic analysis, and information retrieval [3].

Dimension reduction is the process of reducing the number of random variables under consideration, and can be divided into feature selection and feature extraction [1]. As dimensionality increases, query performance in the index structures degrades. Dimensionality reduction algorithms are the only known solution that supports scalable object retrieval and satisfies precision of query results [14]. Feature transforms the data in the high-dimensional space to a space of fewer dimensions [3].The data transformation may be linear, as in principal component analysis (PCA), but any nonlinear dimensionality reduction techniques also exist [9]. In general, handling high dimensional data using clustering techniques

Corresponding author: M. PRANEESH*, *E-mail: raja.praneesh@gmail.com

obviously a difficult task in terms of higher number of variables involved. In order to improve the efficiency the noisy and outlier data may be removed and minimize the execution time, we have to reduce the no. of variables in the original data set. To do so, we can choose dimensionality reduction methods such as principal component analysis (PCA), Singular value decomposition (SVD), and factor analysis (FA). Among this, PCA is preferred to our analysis and the results of PCA are applied to a popular model based clustering technique [6].

Principal component analysis (PCA) is a widely used statistical technique for unsupervised dimension reduction. K-Means clustering is a commonly used data clustering for unsupervised learning tasks. Here we prove that principal components are the continuous solutions to the discrete cluster membership indicators for K-Means clustering [7].The main linear technique for dimensionality reduction, principal component analysis, performs a linear mapping of the data to a lower dimensional space in such a way, that the variance of the data in the low-dimensional representation is maximized. In practice, the correlation matrix of the data is constructed and the eigenvectors on this matrix are computed. The eigenvectors that correspond to the largest eigenvalues (the principal components) can now be used to reconstruct a large fraction of the variance of the original data. Moreover, the first few eigenvectors can often be interpreted in terms of the large-scale physical behavior of the system. The original space (with dimension of the number of points) has been reduced (with data loss, but hopefully retaining the most important variance) to the space spanned by a few eigenvectors.

Many applications need to use unsupervised techniques where there is no previous knowledge about patterns inside samples and its grouping, so clustering can be useful. Clustering is grouping samples base on their similarity as samples in different groups should be dissimilar. Both similarity and dissimilarity need to be elucidated in clear way. High dimensionality is one of the major causes in data complexity. Technology makes it possible to automatically obtain a huge amount of measurements. However, they often do not precisely identify the relevance of the measured features to the specific phenomena of interest. Data observations with thousands of features or more are now common, such as profiles clustering in recommender systems, personality similarity, genomic data, financial data, web document data and sensor data. However, high-dimensional data poses different challenges for clustering algorithms that require specialized solutions. Recently, some researchers have given solutions on high-dimensional problem. Our main objective is proposing a framework to combine relational definition of clustering with dimension reduction method to overcome aforesaid difficulties and improving efficiency and accuracy in K-Means algorithm to apply in high dimensional datasets. Kmeans clustering algorithm is applied to reduced datasets which is done by principal component analysis dimension reduction method [15].

II. METHODOLOGIES:

A. Clustering:

Cluster analysis is one of the major data analysis methods widely used for many practical applications in emerging areas[17].Clustering is the process of finding groups of objects such that the objects in a group will be similar (or related) to one another and different from (or unrelated to) the objects in other groups. A good clustering method will produce high quality clusters with high intra-cluster similarity and low inter-cluster similarity [8]. The quality of a clustering result depends on both the similarity measure used by the method and its implementation and also by its ability to discover some or all of the hidden patterns [16].

B. K-Medoids Clustering:

Clustering is the process of grouping a set of objects into classes or clusters so that objects within a cluster have similarity in comparison to one another, but are dissimilar to objects in other clusters (Han et al 2001). K-Means clustering (MacQueen, 1967) and Partitioning Around Medoids (PAM) (Kaufman and Rousseeuw, 1990) are well known techniques for performing non- hierarchical clustering[12].Unfortunately, K-Means clustering is sensitive to the outliers and a set of objects closest to a centroid may be empty, in which case centroids cannot be updated. For this reason, K-Medoids clustering are sometimes used, where representative objects called medoids are considered instead of centroids. [18] Because it uses the most centrally located object in a cluster, it is less sensitive to outliers compared with the K-Means clustering. Among many algorithms for K-Medoids clustering, Partitioning Around Medoids (PAM) proposed by Kaufman and Rousseeuw (1990) is known to be most powerful [11]. However, PAM also has a drawback that it works inefficiently for large data sets due to its complexity (Han et al, 2001). This is main motivation of this paper. We are interested in developing a new K-Medoids clustering method that should be fast and efficient [5].The most common realisation of *k*-medoid clustering is the Partitioning Around Medoids (PAM) algorithm and is as follows:

- Initialize: randomly select *k* of the *n* data points as the medoids
- Associate each data point to the closest medoid. ("closest" here is defined using any valid distance metric, most commonlyEuclidean distance, Manhattan distance or Minkowski distance)
- For each medoid *m*
- For each non-medoid data point *o*
- Swap *m* and *o* and compute the total cost of the configuration
- Select the configuration with the lowest cost.
- Repeat steps 2 to 5 until there is no change in the medoid.

C. Principal Component Analysis:

Principal component analysis (PCA) involves a mathematical procedure that transforms a number of possibly correlated variables into a smaller number of uncorrelated variables called principal components. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible. Depending on the field of application, it is also named the discrete KarhunenLoève transform (KLT), the Hostelling transform or proper orthogonal decomposition (POD).PCA was invented in 1901 by Karl Pearson.[1] Now it is mostly used as a tool in exploratory data analysis and for making predictive models. PCA involves the calculation of the eigenvalue decomposition of a data covariance matrix or singular value decomposition of a data matrix, usually after mean centering the data for each attribute. The results of a PCA are usually discussed in terms of component scores and loadings (Shaw, 2003).

PCA is the simplest of the true eigenvector-based multivariate analyses [13]. Often, its operation can be thought of as revealing the internal structure of the data in a way which best explains the variance in the data. If a multivariate dataset is visualized as a set of coordinates in a high-dimensional data space (1 axis per variable), PCA supplies the user with a lower-dimensional picture, a "shadow" of this object when viewed from its (in some sense) most informative viewpoint.PCA is closely related to factor analysis; indeed, some statistical packages deliberately conflate the two techniques. True factor analysis makes different assumptions about the underlying structure and solves eigenvectors of a slightly different matrix.

D. Principal Components (PCs):

Technically, a principal component can be defined as a linear combination of optimally weighted observed variables which maximize the variance of the linear combination and which have zero covariance with the previous PCs. The first component extracted in a principal component analysis accounts for a maximal amount of total variance in the observed variables. The second component extracted will account for a maximal amount of variance in the data set that was not accounted for by the first component and it will be uncorrelated with the first component. The remaining components that are extracted in the analysis display the same two characteristics: each component accounts for a maximal amount of variance in the observed variables that was not accounted for by the preceding components, and is uncorrelated with all of the preceding components.

When the principal component analysis will complete, the resulting components will display varying degrees of correlation with the observed variables, but are completely uncorrelated with one another. PCs are calculated using the Eigen value decomposition of a data covariance matrix/ correlation matrix or singular value decomposition of a data matrix, usually after mean centering the data for each attribute. Covariance matrix is preferred when the variances of variables are very high compared to correlation. It would be better to choose the type of correlation when the variables are of different types. Similarly the SVD method is used for numerical accuracy [10].After finding principal components reduced dataset is applied to K-Means clustering.

E. Dataset Description:

We conduct our experiments on a Spectf data set which data is gathered from uci web site. This web site is for finding suitable partners who are very similar from point of personality's view for a person. Several constraints were placed on the selection of these instances from a larger database. The dataset describes diagnosing of cardiac Single Proton Emission Computed Tomography (SPECT) images. Each of the patients is classified into two categories: normal and abnormal. The database of 267 SPECT image sets (patients) was processed to extract features that summarize the original SPECT images. As a result, 44 continuous feature pattern was created for each patient with class distribution. Data are organized in a table with 45columns for attributes of people and 267 rows which are for samples. In class distribution, class value 1 is interpreted as "tested positive for diabetes". There is no Missing Attribute Values available in this dataset.

III. RESULTS:

A. Experimental Setup:

In all experiments we use MATLAB software as a powerful tool to compute clusters and windows XP with Pentium 2.1 GHZ. Reduced datasets done by principal component analysis reduction method is applied to kmeans clustering. As a similarity metric, Euclidean distance has been used in kmeans algorithm.

The steps of the Dimensionality Reduction K-Medoids clustering algorithm are as follows:

Algorithm: K-Medoids clustering algorithm
Input: $X = \{d1, d2, \ldots\ldots, dn\}$ // set of n data items.
Output: A set of *k* clusters

// Phase-1: Apply PCA to reduce the dimension of the breast cancer data set

- Organize the dataset in a matrix X.
- Normalize the data set using Z-score.
- Calculate the singular value decomposition of the data matrix. $X = UDV^T$
- Calculate the variance using the diagonal elements of D.
- Sort variances in decreasing order.
- Choose the p principal components from V with largest variances.
- Form the transformation matrix W consisting of those p PCs.
- Find the reduced projected dataset Y in a new coordinate axis by applying W to X.

//Phase-2: Apply the K-Medoids clustering with Reduced Datasets.

The most common realization of k-medoid clustering is the Partitioning Around Medoids (PAM) algorithm and is as follows:

- Initialize: randomly select k of the n data points as the medoids
- Associate each data point to the closest medoid. ("closest" here is defined using any valid distance metric, most commonly Euclidean distance , Manhattan distance or Minkowski distance)
- For each medoid m
- For each non-medoid data point o
- Swap m and o and compute the total cost of the configuration
- Select the configuration with the lowest cost.
- Repeat steps 2 to 5 until there is no change in the medoid.

B. Experimental Results:

Specify original dataset is reduced using principal component analysis reduction method. Dataset consists of 267 instances and 45 attributes. Here the Sum of Squared Error (SSE), representing distances between data points and their cluster centers have used to measure the clustering quality.

Step 1: Normalizing the original data set:

Using the Normalization process, the initial data values are scaled so as to fall within a small-specified range. An attribute value V of an attribute A is normalized to V' using Z-Score as follows:

$$V' = (V\text{-mean}(A))/\text{std}(A)$$

It performs two things i.e. data centering, which reduces the square mean error of approximating the input data and data scaling, which standardizes the variables to have unit variance before the analysis takes place. This normalization prevents certain features to dominate the analysis because of their large numerical values.

Step 2: Calculating the PCs using Singular Value Decomposition of the normalized data matrix:

The number of PCs obtained is same with the number of original variables. To eliminate the weaker components from this PC set we have calculated the corresponding variance, percentage of variance and cumulative variances in percentage, which is shown in Table II. Then we have considered the PCs having variances less than the mean variance, ignoring the others. The reduced PCs are shown in Table I. Only Sample 20 instances of 529 observations is shown inTableII. The variance in percentage is evaluated using formula

$$\textbf{Var in per} = \frac{\textbf{Var of Pcs}}{\textbf{Total Var}} \times 100$$

The cumulative variance in percentage first value is same as percentage in variance, second value is summation of cumulative variance in percentage and variance in percentage. Similarly other values of cumulative variance are calculated.

Step 3: Finding the reduced data set using the reduced PCs:

The transformation matrix with reduced PCs is formed and this transformation matrix is applied to the normalized data set to produce the new reduced projected dataset, which can be used for further data analysis. We have also applied the PCA on three biological dataset and the reduced no. of attributes obtained for each dataset is shown in Table I.

Step 4: Reduced datasets are applied to kmeans algorithm:

The clustering results shown in Figure I by applying the standard K-Means clustering[18] to the reduced breast cancer dataset. The SSE value obtained and the time taken in ms for reduced breast cancer datasets with original K-Means is given in Table III.

TABLE- I

THE REDUCED DATASETS CONTAINNING 10 ATTRIBUTES WITH 10 INSTANCES

	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10
Data 1	0.275688	-3.52539	-0.15349	1.388551	1.229862	-0.94105	0.002889	-0.67157	-0.45675	0.385471
Data 2	3.208734	-1.25667	-1.18851	-0.05256	0.481033	1.329619	-0.18058	-0.20481	-0.36396	0.419439
Data 3	0.240142	3.031383	-0.38468	-1.70234	1.274549	-0.09329	0.353809	-0.86682	-0.0008	-0.00403
Data 4	0.546438	-1.8004	0.684152	1.987308	-0.74852	-0.61741	-0.58168	0.510881	-0.76904	0.238715
Data 5	-0.35261	0.635868	1.561233	0.016159	1.214766	-0.10369	-1.44954	2.407921	0.943994	2.371395
Data 6	1.700141	-1.84373	0.908034	0.207914	-0.25251	-1.12076	-0.27103	0.498094	-0.65858	-0.41683
Data 7	2.678608	-0.17232	-0.43418	0.785827	0.13861	-0.58924	0.051273	0.072056	0.49082	1.292398
Data 8	0.662313	-0.00466	-1.96551	-1.10087	1.668906	0.3418	1.715326	-0.03901	-0.04487	1.871606
Data 9	0.922855	-0.80656	1.288229	0.366478	0.451446	-1.34816	-1.33913	2.369941	-0.99638	0.58964
Data 10	-1.32448	2.111219	2.428503	0.56695	0.382047	-1.15456	0.181062	0.355317	-0.212	-1.92419

TABLE- II

THE VARIANCES, VARIANCES IN PERCENTAGES, AND CUMMULATIVE VARIANCES IN PERCENTAGES CORRESPONDING TO PC

	Variance	Variance in percentage	Cumulative variance in percentage
PC1	14.1965	31.5477	31.5477
PC2	5.5955	12.4345	43.9823
PC3	4.2683	9.48538	53.4678
PC4	2.4869	5.52643	58.9943
PC5	1.7480	3.88442	62.8787
PC6	1.6115	3.58120	66.4599
PC7	1.3840	3.07546	69.5354
PC8	1.3518	3.00389	72.5393
PC9	1.2635	2.80768	75.3469
PC10	1.1941	2.65351	78.0005
PC11	0.9159	2.03528	80.0358
PC12	0.8561	1.90234	81.9381
PC13	0.8040	1.78670	83.7248
PC14	0.7586	1.68568	85.4105
PC15	0.6627	1.47257	86.8831
PC16	0.5381	1.19572	88.0788
PC17	0.5247	1.16591	89.2447
PC18	0.4942	1.09829	90.3430
PC19	0.4186	0.93024	91.2733
PC20	0.3251	0.72233	91.9956
PC21	0.3220	0.71563	92.7112
PC22	0.2971	0.66014	93.3714
PC23	0.2831	0.62916	94.0006
PC24	0.2551	0.56697	94.5675
PC25	0.2327	0.51712	95.0846
PC26	0.2221	0.49350	95.5781
PC27	0.1916	0.42581	96.0039
PC28	0.1836	0.40790	96.4119
PC29	0.1719	0.38197	96.7938
PC30	0.1575	0.35008	97.1439
PC31	0.1483	0.32962	97.4735
PC32	0.1345	0.29893	97.7725
PC33	0.1296	0.28795	98.0604
PC34	0.1135	0.25216	98.3126
PC35	0.1056	0.23474	98.5473
PC36	0.1008	0.22394	98.7713
PC37	0.0867	0.19256	98.9639
PC38	0.0825	0.18327	99.1471
PC39	0.0776	0.17247	99.3196
PC40	0.0767	0.17049	99.4901
PC41	0.0666	0.14792	99.4901
PC42	0.0490	0.10883	99.7469
PC43	0.0452	0.10035	99.8472
PC44	0.0378	0.08405	99.9313
PC45	0.0309	0.06869	100

TABLE- III
SHOWS RESULTS OF KMEDOIDS WITH NUMBER OF CLUSTERS, SSE AND EXECUTION TIME

K - Medoids			
Dataset	No of Clusters	SSE	Execution Time(in ms)
Spectf Reduced Dataset	1	8403	613
	2	7290	628
	3	6769	709
	4	5355	789
	5	3388	874

The above results show that the K-Medoids algorithm provides sum of squared error distance and Execution time of corresponding clusters. Figure I shows graph of SSE and Number of clusters. In this figure, when number of clusters increases, sum of squared error distance values decreases. Figure II shows number of clusters increases, Execution time increases.

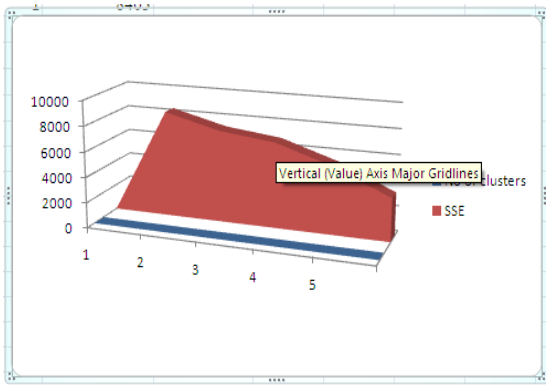

Fig. I: Shows SSE and number of clusters

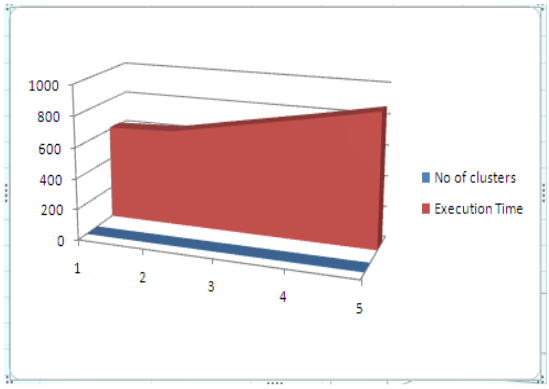

Fig. II: Shows execution time and number of clusters

I V. CONCLUSION:

In this paper a dimensionality reduction through PCA, is applied to K-Medoids algorithm. Using Dimension reduction of principal component analysis, original pima Indian diabetes dataset is compact to reduced data set which was partitioned in to k clusters in such a way that the sum of the total clustering errors for all clusters was reduced as much as possible while inter distances between clusters are maintained to be as large as possible. The experimental results show that principal component analysis is used to reduce attributes and reduced dataset is applied to K-Medoids clustering. Evolving some dimensional reduction methods like canonopies and wavelet transforms can be used for high dimensional datasets is suggested as future work.

V. REFERENCES:

[1] Chao Shi and Chen Lihui, 2005. Feature dimension reduction for microarray data analysis using locally linear embedding, *3rd Asia Pacific Bioinformatics Conference,* pp. 211-217.

[2] Davy Michael and Luz Saturnine, 2007. Dimensionality reduction for active learning with nearest neighbor classifier in text categorization problems, *Sixth International Conference on Machine Learning and Applications*, pp. 292-297

[3] Maaten L.J.P., Postma E.O. and Herik H.J. van den, 2007. Dimensionality reduction: A comparative review", *Tech. rep.University of Maastricht*.

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

[5] Hae-Sang Park*, Jong-Seok Lee and Chi-Hyuck Jun, "A K-Means-like Algorithm for K-Medoids Clustering and Its Performance", Department of Industrial and Management Engineering, POSTECH San 31 Hyoja-dong, Pohang 790-784, S. Korea

[6] IEEE I.T Jolliffe, "*Principal Component Analysis*", Springer, second edition.

[7] Chris Ding and Xiaofeng He, "K-Means Clustering via Principal Component Analysis", In proceedings of the 21st *International Conference on Machine Learning*, Banff, Canada, 2004

[8] Xu R. and Wunsch D., 2005. Survey of clustering algorithms, *IEEE Trans. Neural Networks*, Vol. 16, No. 3, pp. 645-678.

[9] Yan Jun, Zhang Benyu, Liu Ning, Yan Shuicheng, Cheng Qiansheng, Fan Weiguo, Yang Qiang, Xi Wensi, and Chen Zheng,2006. Effective and efficient dimensionality reduction for large-scale and streaming data preprocessing, *IEEE transactions on Knowledge and Data Engineering*, Vol. 18, No. 3, pp. 320-333.

[10] Yeung Ka Yee and Ruzzo Walter L., 2000. "An empirical study on principal component analysis for clustering gene expression Data", *Tech. Report, University of Washington.*

[11] Q Zhang – 2005, "A New and Efficient *K -Medoid* Algorithm for Spatial Clustering".

[12] HS Park - 2009, "A simple and fast algorithm for *K-Medoids* clustering".

[13] *Principal Component Analysis and Effective* ***K-Means Clustering*** *by C Ding*

[14] Wray Buntine, 2008, "**K-Means Clustering** and PCA" National ICT Australia

[15] Constrained K-Means Clustering with Background Knowledge by Kiri Wagsta_ Claire Cardie

[16] Wagsta_, K., & Cardie, C. (2000). Clustering with instance-level constraints. Proceedings of the Seventeenth International Conference on Machine Learning (pp. 1103{1110). Palo Alto, CA: Morgan Kaufmann.

[17] Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. Journal of the American Statistical Association, 66, 846-850.

[18] T Velmurugan - 2010, Computational Complexity between K-Means and K-Medoids Clustering Algorithms for Normal and Uniform Distributions of Data Points".

****

