



An Effective Method of Dimensionality Reduction for High Dimensional Datasets Using PCA

D.Napoleon*

Assistant Professor,
School of Computer Science and Engineering,
Bharathiar University, Coimbatore, India
mekaranapoleon@yahoo.co.in

S.Pavalakodi

M.Phil Scholar,
School of Computer Science and Engineering,
Bharathiar University, Coimbatore, India
pavalakodi@gmail.com

Abstract: - Data mining employs a variety of traditional statistical methods such as cluster analysis, discriminate analysis, logistic regression, and time series forecasting. Due to the mega high dimensionality nature of datasets, data dimension reduction has drawn special attention for such type of data analysis. Feature extraction 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-means algorithm is applied and shows the results.

Keywords: - Principal component analysis, dimensional reduction, kmeans 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 and k-means 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]. It is also common that PCA is used to project data to a lower dimensional subspace and K-means is then applied in the subspace (Zha *et al.*, 2002)[15]. In other cases, data are embedded in a low-dimensional space such as the eigenspace of the graph Laplacian, and K-means is then applied (Ng *et al.*, 2001)[13]. The main basis of PCA-based dimension reduction is that PCA 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 via the singular value decomposition (SVD) (Eckart

& Young, 1936). However, this noise reduction property alone is inadequate to explain the effectiveness of PCA[7].

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 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.

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-Means Clustering Algorithm

K-means is a commonly used partitioning based clustering technique that tries to find a user specified number of clusters (k), which are represented by their centroids, by minimizing the square error function [5]. Although K-means is simple and can be used for a wide variety of data types. The K-means algorithm is one of the partitioning based, nonhierarchical clustering methods. Given a set of numeric objects X and an integer number k , the K-means algorithm searches for a partition of X into k clusters that minimizes the within groups sum of squared errors. The K-means algorithm starts by initializing the k cluster centers[12]. The input data points are then allocated to one of the existing clusters according to the square of the Euclidean distance from the clusters, choosing the closest. The mean (centroid) of each cluster is then computed so as to update the cluster center[11]. This update occurs as a result of the change in the membership of each cluster. The processes of re-assigning the input vectors and the update of the cluster centers is repeated until no more change in the value of any of the cluster centers.

The steps of the K-means algorithm are written below:

Step: 1 Initialization: choose randomly K input vectors (data points) to initialize the clusters.

Nearest-neighbor search: for each input vector, find the cluster center that is closest, and assign that input vector to the corresponding cluster.

Step: 2 Mean update: update the cluster centers in each cluster using the mean (centroid) of the input vectors assigned to that cluster

Step: 3 Stopping rule: repeat steps 2 and 3 until no more change in the value of the means.

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. 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 (Pc)

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 breast cancer 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. Based on 8 pages of psychiatric questions personality of people for different aspects is extracted. Each group of questions is related to one dimension of personality. To trust of user some questions is considered and caused reliability of answers are increased. Data are organized in a table with 90 columns for

attributes of people and 704 rows which are for samples. There are missing values in this table because some questions have not been answered, so we replaced them with 0. On the other hand we need to calculate length of each vector base on its dimensions for further process. All attributes value in this table is ordinal and we arranged them with value from 1 to 5, therefore normalizing has not been done. There is not any correlation among attributes and it concretes an orthogonal space for using Euclidean distance. All samples are included same number of attributes.

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 Feature Reduction k-means clustering algorithm are as follows.

Input: $X = \{d1, d2, \dots, dn\}$ // 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

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

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

- [a] Initialization: choose randomly K input vectors (data points) to initialize the clusters.
- [b] Nearest-neighbor search: for each input vector, find the cluster center that is closest, and assign that input vector to the corresponding cluster.
- [c] Mean update: update the cluster centers in each cluster using the mean (centroid) of the input vectors assigned to that cluster.
- [d] Stopping rule: repeat steps 2 and 3 until no more change in the value of the means.

B. Experimental Results

Breast cancer original dataset is reduced using principal component analysis reduction method. Dataset consists of 569 instances and 30 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 I**. Then we have considered the PCs having variances less than the mean variance, ignoring the others. The reduced PCs are shown in **Table II**. Only Sample 20 instances of 529 observations is shown in table2. The variance in percentage is evaluated using formula

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

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 is 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 II**.

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 Variances, Variances in Percentages, and Cumulative Variances in Percentages Corresponding to PCs

	Variance	Variance in percentage	Cumulative variance in percentage
PC1	13.28161	13.28161	13.28161
PC2	5.691355	18.97118	63.24321
PC3	2.817949	9.393163	72.63637
PC4	1.98064	6.602135	79.23851
PC5	1.648731	5.495768	84.73427
PC6	1.207357	4.024522	88.7588
PC7	0.67522	2.250734	91.00953
PC8	0.476617	1.588724	92.59825
PC9	0.416895	1.389649	93.9879
PC10	0.350693	1.168978	95.15688
PC11	0.293916	0.979719	96.1366
PC12	0.261161	0.870538	97.00714
PC13	0.241357	0.804525	97.81166
PC14	0.15701	0.523366	98.33503
PC15	0.094135	0.313783	98.64881
PC16	0.079863	0.266209	98.91502
PC17	0.059399	0.197997	99.11302
PC18	0.052619	0.175396	99.28841
PC19	0.049478	0.164925	99.45334
PC20	0.031159	0.103865	99.5572
PC21	0.029973	0.09991	99.65711
PC22	0.027439	0.091465	99.74858
PC23	0.024341	0.081136	99.82971
PC24	0.018055	0.060183	99.8899
PC25	0.015481	0.051604	99.9415
PC26	0.008178	0.027259	99.96876
PC27	0.0069	0.023002	99.99176
PC28	0.001589	0.005298	99.99706
PC29	0.000749	0.002496	99.99956
PC30	0.000133	0.000443	100

Table II: The Reduced Datasets Containing 6 Attributes With 20 Instances

	V1	V2	V3	V4	V5	V6
Data1	-9.18475	-1.9469	-1.1222	3.6305	1.1941	1.4102
Data2	-2.38570	3.7649	-0.5288	1.1173	-0.6212	0.0286
Data3	-5.72886	1.0742	-0.5513	0.9113	0.1769	0.5410
Data4	-7.11669	-10.266	-3.2299	0.1524	2.9563	3.0507
Data5	-3.93184	1.9464	1.3885	2.9381	0.5463	-1.2254
Data6	-2.37815	-3.9465	-2.9323	0.9402	1.0351	-0.4506
Data7	-2.23692	2.6877	-1.6385	0.1492	-0.0403	-0.1288
Data8	-2.14141	-2.3382	-0.8712	-0.1269	1.4262	-1.2559
Data9	-3.17213	-3.3888	-3.1172	-0.6008	1.5210	0.5591
Data10	-6.34616	-7.7204	-4.3381	-3.3722	-1.7088	-0.7233
Data11	0.80970	2.6569	0.4884	-1.6711	-0.2756	0.1272
Data12	-2.64876	0.0665	-1.5251	0.0512	-0.3317	0.7642
Data13	-8.17784	-2.6986	5.7252	-1.1113	-1.0426	2.5923
Data14	-0.34183	0.9674	1.7157	-0.5945	-0.4676	1.0068
Data15	-4.33857	-4.8568	-2.8136	-1.4533	-1.2889	-0.3494
Data16	-4.07207	-2.9744	-3.1225	-2.4559	0.4080	0.4953
Data17	-0.22985	1.5634	-0.8018	-0.6500	0.4943	-0.7615
Data18	-4.41413	-1.4174	-2.2683	-0.1861	1.4226	-0.7518
Data19	-4.94435	4.1107	-0.3145	-0.0881	0.0567	-1.1367
Data20	0.1888	-0.5928	1.5949	0.4418	-0.0486	-0.0486

Table III: Shows Results of Kmeans with Number of Clusters, SSE and Execution Time

<i>K - Means</i>			
<i>Dataset</i>	<i>No of Clusters</i>	<i>SSE</i>	<i>Execution Time(in ms)</i>
Breast Cancer Reduced Dataset	1	12604	0.513
	2	9513	0.631
	3	8075	0.689
	4	7641	0.777
	5	1959	0.862

The above results show that the kmeans 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.

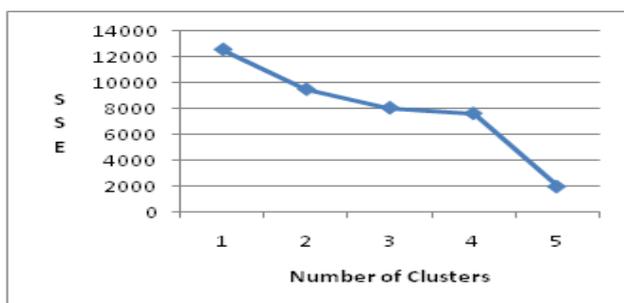


Figure 1: Shows SSE and Number of Clusters

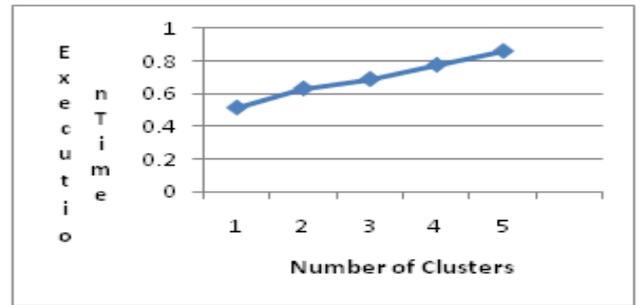


Figure2: Shows Execution Time and Number of Clusters

IV. CONCLUSION

In this paper a dimensionality reduction through PCA, is applied to kmeans algorithm. Using Dimension reduction of principal component analysis, original breast cancer 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 kmeans clustering. Evolving some dimensional reduction methods like canopies 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] RM Suresh, K Dinakaran, P Valarmathie, “Model based modified k-means clustering for microarray data”, International Conference on Information Management and Engineering, Vol.13, pp 271-273, 2009,
- [6] IEEEL.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] 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.
- [15] Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. Journal of the American Statistical Association, 66, 846-850.
- [16] Bradley, P. S., Bennett, K. P., & Demiriz, A. (2000). Constrained k-means clustering (Technical Report MSR-TR-2000-65). Microsoft Research, Redmond.