ISSN No. 0976-5697

Volume 6, No. 8, Nov-Dec 2015

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Object Oriented K-Means clustering using Eigen Decomposition for Student Data

N.Sree Ram
LBRCE, Mylavaram
JNTUK Kakinada
Andhra Pradesh,India

Dr.K.Satya Prasad
Professor, Dept of ECE
JNTUK, Kakinada
Andhra Pradesh,India

Dr.J.V.R.Murthy
Professor, Dept of CSE
JNTUK, Kakinada
Andhra Pradesh,India

Abstract: Data clustering [1] is the process of forming classes or groups of similar data objects. The real time data objects are either multi-dimensional [4] or high dimensional [3]. Grouping these high dimensional data objects requires a lot of computational effort, time and space. To remove these hurdles from clustering of high dimensional data, the proposed work uses Eigen decomposition for dimensionality reduction, and then k-means is implemented through object oriented [2] programming on student's marks data. The experimental results show how Eigen value decomposition and object oriented implementation brings novelty to clustering process**.**

Keywords: Eigen decomposition; k-means; object oriented implementation; clustering; high dimensional data

I. INTRODUCTION

Clustering [1] high dimensional data is a challenging issue in data and knowledge engineering. Generally the real time data objects are high dimensional. Typical clustering techniques are available to process these high dimensional data, but implementation of these techniques don't provide modularity, data security, code reusability and require much more computational effort. In order to bring novelty to k-means, this proposed work uses Eigen decomposition for dimensionality reduction, and then k-means is implemented as object oriented system. Dimensionality reduction [10] is one of the important tasks in data pre-processing and uses different kinds of matric decomposition techniques such as cholesky decomposition, LUdecompostion, QR decomposition, single valued decomposition, and Eigen Decomposition. In this paper, Eigen decomposition is used for dimensionality reduction, which will be explained in the second section of this paper. In order to incorporate modularity, reusability, and data security, k-means was implemented with object oriented paradigm, which will be clearly discussed in section 3 of this paper. In Object Oriented implementation[11], the entire system is implemented as a set of objects, classes and methods section 4 provides information about object oriented k-means using Eigen decomposition. Section 5 of this paper shows implementation details of the system in object oriented paradigm and experimental results of proposed algorithm whereas section 6 provides conclusion.

II. RELATED WORK

A. *Eigen Decomposition*

Clustering of high dimensional data [3] requires dimensionality reduction as pre-processing of data. Typical dimensionality reduction techniques are available, which can project data from high dimensional space to low dimensional space. This proposed work uses Eigen Decomposition based dimensionality reduction technique known as Principal Component Analysis [11] (PCA). The Eigen decomposition is also known as spectral decomposition and it can be applied to a square matrix with distinct Eigen vectors i.e.

$$\mathbf{A=VDV^{-1}} \quad (1)$$

Where D is diagonal matrix forms from Eigen values of A, and the columns of V are the corresponding to Eigen vectors of A. The classical Dimensionality reduction method Principal Component Analysis works based on Eigen decomposition. PCA [4][5] was invented in 1901 by Karal Pearson and is the simplest of the true Eigen vector based multivariate analysis. When a multivariate data set is represented as a set of coordinates in a high dimensional data space, PCA can generate the low dimensional data. In PCA first the given high dimensional data is represented as a set of coordinates called as data matrix. Secondly calculation is made on the mean of the each dimension to generate mean variant matrix and generate the covariance matrix of the data matrix by using the following formula

$$\mathbf{Cov}(X, Y)=\Sigma(X_i - X)(Y_i - Y)/ N =\Sigma x_i y_i / N \quad (2)$$

Where N is the number of scores in each set of data, X is the mean of the N scores in the first data set X_i is the ith

raw score in the first set of scores, x_i is the *i*th deviation score in the first set of scores, *Y* is the mean of the *N* scores in the second data set, Y_i is the *i*th raw score in the second set of scores y_i is the *i*th deviation score in the second set of scores and Cov(*X*, *Y*) is the covariance of corresponding scores in the two sets of data. Covariance matrix is always square matrix. Now, calculate the Eigen values from the covariance matrix using formula

$$(A - \lambda I)\mathbf{v} = 0, \quad (3)$$

Where I is an identity square matrix. Now generate the Eigen vectors of each Eigen value and arrange the vectors as per the order of Eigen values. Finally transpose the Eigen vector matrix, extract highest significant dimensions and multiply this with mean variant matrix of data matrix to get feature vector. These dimensions are called Principal Components of the data matrix.

Feature vector=Mean adjusted data* Row feature vectors.

The entire process is explained as an algorithm as follows

Algorithm1: PCA or Eigen Decomposition

Input: Data matrix

Output: Feature vector

Method:

Step 1: Represent the given data set as coordinates matrix

Step 2: Calculate the mean of each dimension in the data set using ∑xi/m

Step 3: Generate mean adjusted matrix for data matrix

Step 4: Generate the covariance of data matrix using ∑(X - mean(X))(Y-mean(Y))/n

Step 5: Find out the Eigen value of covariance matrix i.e. $(A - \lambda I)\mathbf{v} = 0,$ where I is identity matrix

Step 6: Generate the Eigen vectors of respective Eigen values of covariance matrix.

Step 7: Arrange the Eigen vectors from high significant Eigen value to low significant Eigen value.

Step 8: Transpose the Eigen vector matrix

Step 9: Extract the high significant dimensions and multiply with mean adjusted data matrix i.e. Feature Vector=Eigen vector matrix*Mean adjusted matrix.

Step 10: The product of two matrices in above step is the feature vectors and also known as Principal Components.

Thus Eigen decomposition projects data from high dimensional space to low dimensional space.

B. K-Means

K-means [1][7] is one of the most well-known partitioning clustering techniques, which assigns each object to exactly one cluster. The objects in one cluster are similar to each other and the objects in one cluster are dissimilar to objects in other cluster. It uses proximity measures to identify the similarity and dissimilarity between the objects. The most popular proximity measure is Euclidian Distance, which measures the geometric distance between the objects. in this first any k of the objects need to pick up as centroids of k clusters. Now calculate distance between each centroid and other objects in data set. Based on the distance the objects are allocated to clusters respectively and then calculate mean of each cluster. Now replace centroid of each cluster with this mean. Repeat this process until no change in further clusters. The entire process is represented as an algorithm.

Algorithm 2: K-Means

Input: k specifies number of clusters, D data set and N specifies number of objects in data set.

Output: K clusters

Method:

Step 1: Randomly pickup k centroids from data set D

Step 2: Calculate the distance from each centroid to remaining all other objects and construct distance matrix with size K*N

Step 3: Assign each object to exactly one cluster which is having minimum distance.

Step 4: Calculate the mean of each cluster

Step 5: Replace centroids with this mean. Now mean of cluster will act as centroid

Step 6: Repeat the steps from step 2 until no change in clusters.

C. Object Oriented Paradigm

Object Oriented Paradigm is fashion of implementing system by following Object Oriented Principles and concepts. In this the entire system is implemented as a collection of objects. An object is a real time entity which is having some set of characteristics and specific behaviour. The characteristics of an object specify the state of an object and the behaviour of an object change the state of an object. In this one object can interact with other objects in the system by using message passing technique. The three basic object oriented principles are encapsulation, inheritance and polymorphism. Encapsulation is the process of combining both characteristics and behaviour of object as single object. Inheritance is a process of defining a new object by inheriting features and behaviour from an existing object. Polymorphism is a process of defining different behaviours with the same name. There are variety of programming languages with object oriented Paradigm [1] such as smalltalk, C++ and Java. The object oriented programming languages provide a basic construct to define an object called as class. The class can be

used to combine both behaviour and characteristics of an object. The characteristics of an object is defines as member data and behaviour can define through methods. Data abstraction is one of the important features available in this paradigm which provides security to data by means of methods. Inheritance and polymorphism provide modularity and reusability respectively

III. OBJECT ORIENTED K-MEANS

Most of the researches implemented k-means algorithm in conventional procedural paradigm, which doesn't provide data security, modularity and code reusability. In order to incorporate modularity and code reusability the proposed work is implemented K-Means in Object Oriented paradigm. In this work entire k-means is defined as a single object, which consists of dataset, distance matrix, k number of clusters and centroids as member data and distance calculation and cluster forming as methods. The entire process is explained as algorithm as follows

Algorithm 3: Object Oriented k-Means

Input: Data set, Distance matrix, and K (Number of clusters)

Output: K clusters

Method:

Step 1: create class and named it as k-means

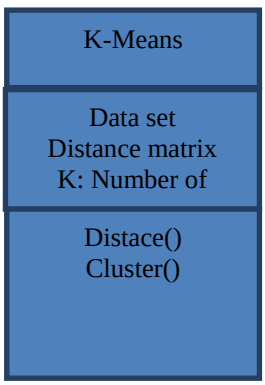
Step 2: Define Data set, Distance matrix and k (number of clusters) as member data i.e. properties of an object

Step3: Implement the methods distance calculation and cluster forming to define behaviour of object

Step 4: Finally create object to k-means class

The object oriented k-means can be modelled by using class diagram as follows

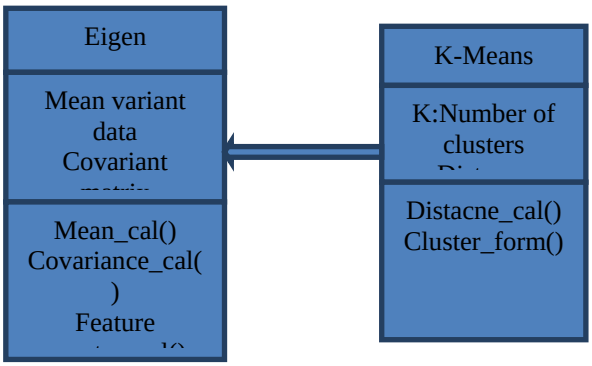
Fig.1. Class Diagram for k-means



IV. OBJECT ORIENTED K-MEANS USING EIGEN DECOMPOSITION

To enable the object oriented k- means to cluster multi-dimensional data objects the proposed work used Eigen decomposition technique for Dimensionality reduction. First Eigen decomposition was applied on dataset which will generate feature vector. This Eigen decomposition is implemented as Eigen object. Now apply Object oriented K-means on feature vector instead of directly applying on original data set. Object oriented k-means is implemented as another object and there exists generalization relationship between Eigen object and k-means object which can be modeled as follow using class diagram.

Fig.2. Class Diagram for Object oriented K-Means using Eigen Decomposition



V. IMPLEMENTATION AND EXPERAMENTAL RESULTS

This proposed work is implemented by using object oriented programming language Java [6] and the student marks data is stored in Oracle data base. It used JDBC Type 4 driver [6] to access data set from database. The data set in data base is five dimensional and is shown as follows

Table I. student data

H.T.No	Marks in DBMS	Marks in SE	Marks in Java	Marks in Data Mining
1176050001	75	80	73	69
1176050002	74	78	69	65
1176050003	54	58	72	81
……	……	…..	….	…..
1176050012	92	81	76	78

The student data objects are clustered based on their marks so the four dimensions of the data representing the marks obtained by the student are taken as task relevant data Eigen decomposition is applied to reduce the dimensions. During this process 4*4 covariance matrix was generated as shown below.

Table II. Covariance matrix

231.8792	365.2727	94.4242	65.0606
365.2727	163.2954	82.2272	56.91
94.4242	82.2272	77.6728	52.4242
65.0606	56.91	52.4242	66.7880

From this covariance matrix the feature vector was generated by applying Eigen decomposition. The feature vector resulted from this work is shown as follows

Table III. Feature vector

-0.2274	**5..22015**
-4.3601	**9.6141**
-25.2179	**-14.7251**
10.7626	**4.8654**
-20.8115	**-5.9421**
35.8651	**-9.8587**
32.3133	**-6.4433**
-4.9100	**6.6012**
-13.8488	**3.3918**
5.1934	**6.0616**
-31.2323	**-0.1777**
15.5242	**2.9602**

The result came from the above implementation is shown as output screens as follows

Fig. 3: output screen showing initial results

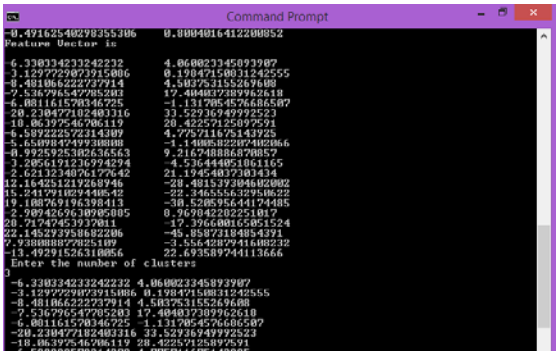
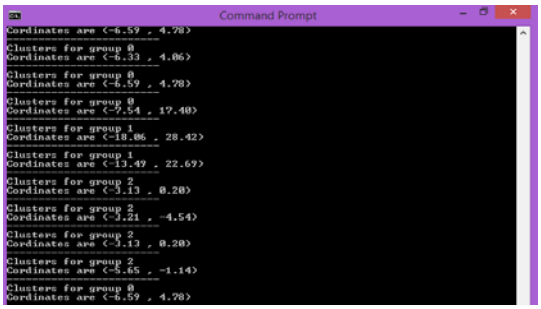


Fig. 4 Output screen showing final results



VI. CONCLUSION

In this paper we presented Eigen decomposition as dimensionality reduction and k-means applied on the feature vector resulted from Eigen decomposition. The implementation of entire work is done in object oriented paradigm, So that the data security is provided by restricting the data accessibility except the methods in that object. Due to the usage of object oriented principles it could be possible to bring modularity and reusability. In order to accelerate the performance of the algorithm in future it will implement in multi core environment by using multi-threading.

VII. REFERENCES

[1] A.K.Jain., M.N. Murthy., & P.J.*Flynn Data Clustering : Review* ACM Computing Surveys, Vol.31, No.3, September 1999.

[2] Anja Struyf., Mia Hubert Peter., & J. Rousseeuw *Clustering in an Object-Oriented nvironment* Journal of Stastical software ISSN 1548-7660

[3]] Brian Mc Williams.,& Giovanni Motana Subspace clustering of high dimensional Data: a predictive approach" Data Mining and Knowledge Discovery published online: 5 May 2014, Spinger

[4]] Cacilia Zirn., & Heiner Stuckenschmidt *Multi Dimensional topic alalysis in political texts* Data & Knowledge Engineering 90(2014)38-53, Elsevier

[5] Hailin Li. Asynchronism-based prinicipal component analusis for time series data mining Expert system Applications 41(2014) 2842-2850 ELSEVIER

[6] Herbert Schildt. *Java 2* The complete Reference Fifth edition Tata McGraw-Hill

[7] Jiawei Han., & Micheline Kamber *Data Mining Concepts* Second Edition ELSEVIER

[8] M.S.B.PhridviRaj, C.V. GuruRao *Data mining-past, present and future- a typical survey on data streams* The 7th International Conference INTER-ENG2013 ScienceDirect ELSEVIER

[9] Pang-Ning Tan., Vipin Kumar ., & Michael Steinbach *Introduction to Data Mining* PEARSON

[10]] Piotr Pawliczek., & Witold *Dzwine Interactive Data Mining by using multi dimensional scaling*SviVerse ScienceDirect 2013 International Conference Procedia Computer Science ELSEVIER

[11] Principal Component Analysis Agilent Technologies, Inc, 2005 Main 866.744.7638

[12] S.Jiang, J.Ferreira.,& M.C.Gonzales Clustering daily patterns human activity in the city Data Mining and Knowledge Discovery DOI 10.1007/s10618-012-0264-z Published online: 20 April 2012 Springer

[13] Srecko Natek., & Moti Zwilling Student data mining solution –knowledge management system related to higher education institutions Expert system Applications 41(2014) 6400-6407 ELSEVIER

[14] Xuan Hong Dang, & James Bailey Generating multiple alternative clusterings via globally optimal subspaces Data Mining and Knowledge Discovery published online: 6 April 2013 , Spinger