

Methodological Analysis of Principal Component Analysis (PCA) Method

Liton Chandra Paul¹, Abdulla Al Suman², Nahid Sultan³

^{1,2,3}Department of Electronics & Telecommunication Engineering,
Rajshahi University of Engineering & Technology, Rajshahi-6204, Bangladesh.

Abstract

Principal Components Analysis (PCA) is a practical and standard statistical tool in modern data analysis that has found application in different areas such as face recognition, image compression and neuroscience. It has been called one of the most precious results from applied linear algebra. PCA is a straightforward, non-parametric method for extracting pertinent information from confusing data sets. It presents a roadmap for how to reduce a complex data set to a lower dimension to disclose the hidden, simplified structures that often underlie it. This paper mainly addresses the Methodological Analysis of Principal Component Analysis (PCA) Method. PCA is a statistical approach used for reducing the number of variables which is most widely used in face recognition. In PCA, every image in the training set is represented as a linear combination of weighted eigenvectors called eigenfaces. These eigenvectors are obtained from covariance matrix of a training image set. The weights are found out after selecting a set of most relevant Eigenfaces. Recognition is performed by projecting a test image onto the subspace spanned by the eigenfaces and then classification is done by measuring minimum Euclidean distance. In this paper we present a comprehensive discussion of PCA and also simulate it on some data sets using MATLAB.

KEYWORDS: Principal component, Covariance matrix, Eigenvalue, Eigenvector, PCA.

Introduction

Principal component analysis is also called “Hotelling transform” or “Karhunen-loeve (KL) Method”. Principle Component Analysis (PCA) is one of the most frequently used multivariate data analysis. Principle Component Analysis can be considered as a projection method which projects observations from a p -dimensional space with p variables to a k -dimensional space (where $k < p$) so as to conserve the maximum amount of information (information is measured here through the total variance of the scatter plots) from the initial dimensions. If the information associated with the first 2 or 3 axes represents a sufficient percentage of the total variability of the scatter plot, the observations will be able to be represented on a 2- 3-dimensional chart, thus making interpretation much easier [1]. 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 [2].

2. Objectives of Principal Component Analysis

- To discover or to reduce the dimensionality of the data set.
- To identify new meaningful underlying variables.

The mathematical technique used in PCA is called eigen analysis: we solve for the eigenvalues and eigenvectors of a square symmetric matrix with sums of squares and cross products. The eigenvector associated with the largest eigenvalue has the same direction as the first principal component. The eigenvector associated with the second largest eigenvalue determines the direction of the second principal component. The sum of the eigenvalues equals the trace of the square matrix and the maximum number of eigenvectors equals the number of rows (or columns) of this matrix [2].

3. Characteristics of Principal Components

The first component extracted in a principal component analysis accounts for a maximal amount of total variance in the observed variables. Under typical conditions, this means that the first component will be correlated with at least some of the observed variables. It may be correlated with many.

The second component extracted will have two important characteristics. First, this component will account for a maximal amount of variance in the data set that was not accounted for by the first component. Again under typical conditions, this means that the second component will be correlated with some of the observed variables that did not display strong correlations with component 1.

The second characteristic of the second component is that it will be uncorrelated with the first component. Literally, if you were to compute the correlation between components 1 and 2, that correlation would be zero.

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. A principal component analysis proceeds in this fashion, with each new component accounting for progressively smaller and smaller amounts of variance (this is why only the first few components are usually retained and interpreted). When the analysis is complete, the resulting components will display varying degrees of correlation with the observed variables, but are completely uncorrelated with one another [3].

4. Principal Component Analysis Is Not Factor Analysis

Principal component analysis is sometimes confused with factor analysis and this is understandable, because there are many important similarities between the two procedures.

Both are variable reduction methods that can be used to identify groups of observed variables that tend to hang together empirically. Both procedures can be performed with the SAS System's FACTOR procedure and they sometimes even provide very similar results. Nonetheless, there are some important conceptual differences between principal component analysis and factor analysis that should be understood at the outset. Perhaps the most important deals with the assumption of an underlying causal structure: factor analysis assumes that the co-variation in the observed variables is due to the presence of one or more latent variables (factors) that exert causal influence on these observed variables. An example of such a causal structure is presented in Figure 5.1. The ovals in Figure 1 represent the latent (unmeasured) factors of "satisfaction with Supervision" and "satisfaction with pay." These factors are latent in the sense that they are assumed to actually exist in the employee's belief systems, but cannot be measured directly. However, they do exert an influence on the employee's responses to the seven items that constitute the job satisfaction questionnaire described earlier (these seven items are represented as the squares labelled V1-V7 in the figure). It can be seen that the "supervision" factor exerts influence on items V1-V4 (the supervision questions), while the "pay" factor exerts influence on items V5-V7 (the pay items).

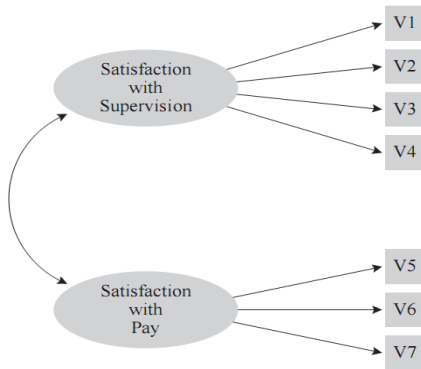


Figure-1: Example of the Underlying Causal Structure that is assumed in Factor Analysis

Researchers use factor analysis when they believe that certain latent factors exist that exert causal influence on the observed variables they are studying. Exploratory factor analysis helps the researcher identify the number and nature of these latent factors. In contrast, principal component analysis makes no assumption about an underlying causal model. Principal component analysis is simply a variable reduction procedure that (typically) results in a relatively small number of components that account for most of the variance in a set of observed variables [3].

In summary, both factor analysis and principal component analysis have important roles to play in social science research, but their conceptual foundations are quite distinct.

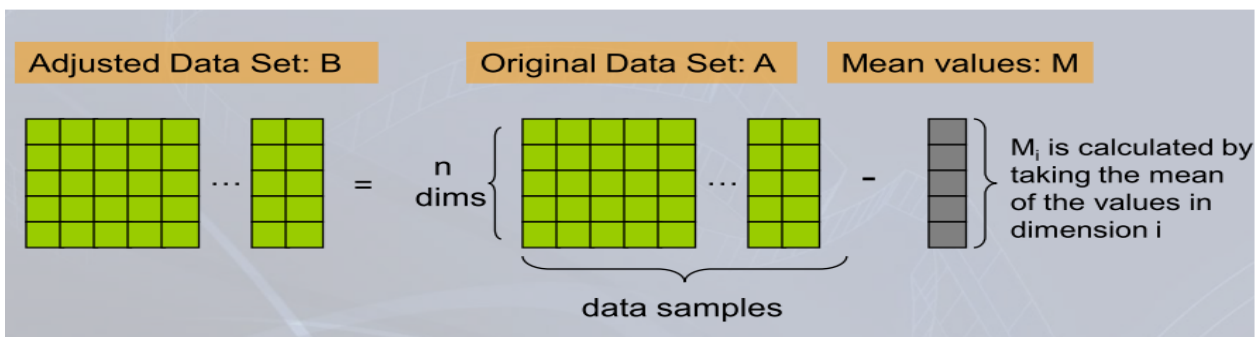
5. Principal Component Analysis Method

Step 1: Get some data

Let us consider a simple arbitrary three dimensional data set. The reason why I have chosen this is so that I can provide plots of the data to show what the PCA analysis is doing at each step. The data I have used is found in Figure 5.2, along with a plot of that data.

Step 2: Subtract the mean

For PCA analysis, you have to subtract the mean from each of the data dimensions. The mean subtracted is the average across each dimension. So, all the x values have \bar{x} (the mean of the values x of all the data points) subtracted, and all the y values have \bar{y} subtracted from them and all the z values have \bar{z} subtracted from them. This produces a dataset whose mean is zero.



For example,

Original Data Set, A			Mean, M			Adjusted Data Set, B		
x	y	z	\bar{x}	\bar{y}	\bar{z}	x	y	z
269.8000	38.9000	50.5000	269.9733	38.9067	50.4800	-0.1733	-0.0067	0.0200
272.4000	39.5000	50.0000				2.4267	0.5933	-0.4800
270.0000	38.9000	50.5000				0.0267	-0.0067	0.0200
272.0000	39.3000	50.2000				2.0267	0.3933	-0.2800
269.8000	38.9000	50.5000				-0.1733	-0.0067	0.0200
269.8000	38.9000	50.5000				-0.1733	-0.0067	0.0200
268.2000	38.6000	50.2000				-1.7733	-0.3067	-0.2800
268.2000	38.6000	50.8000				-1.7733	-0.3067	0.3200
267.0000	38.2000	51.1000				-2.9733	-0.7067	0.6200
267.8000	38.4000	51.0000				-2.1733	-0.5067	0.5200
273.6000	39.6000	50.0000				3.6267	0.6933	-0.4800
271.2000	39.1000	50.4000				1.2267	0.1933	-0.0800
269.8000	38.9000	50.5000				-0.1733	-0.0067	0.0200
270.0000	38.9000	50.5000				0.0267	-0.0067	0.0200
270.0000	38.9000	50.5000				0.0267	-0.0067	0.0200

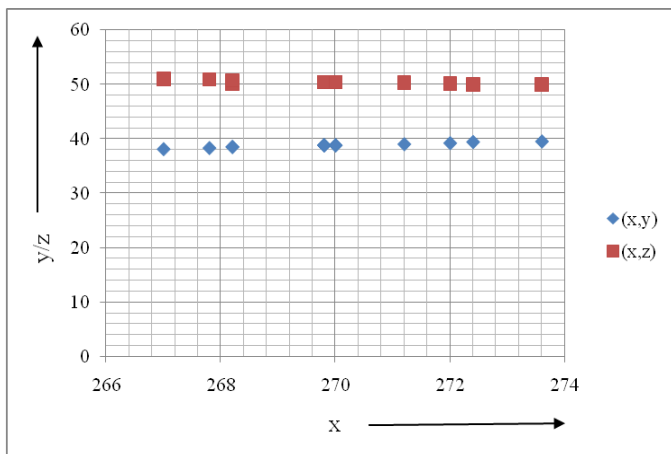


Figure-5.2: Plot of Original Data Set

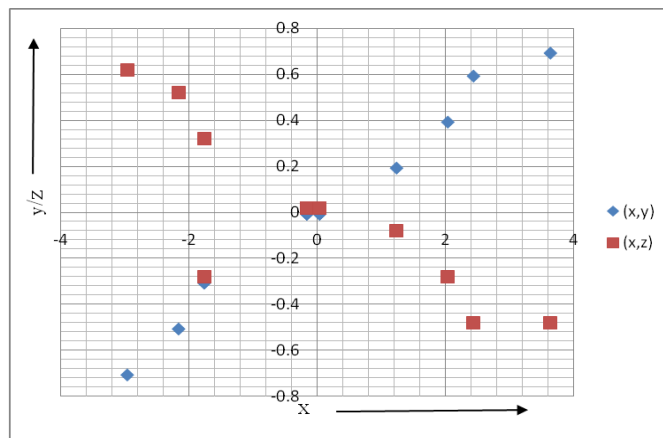


Figure-5.3: Plot of New Adjusted Data Set

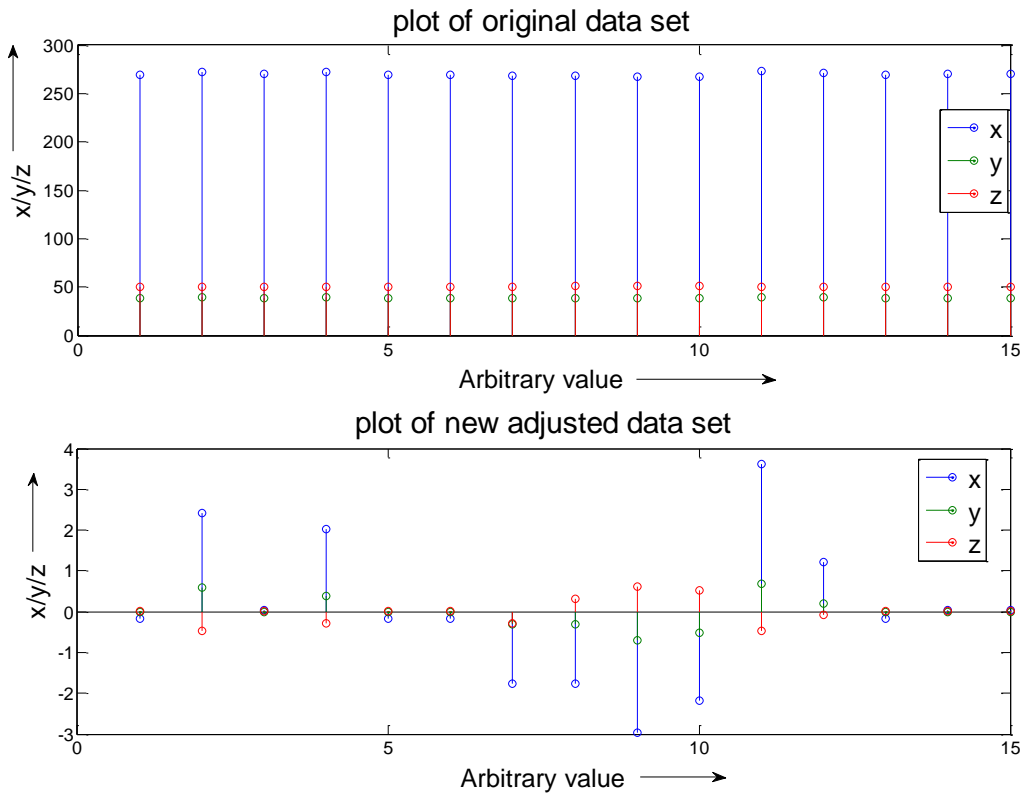


Figure-5.4: Plot of Original Data Set and New Adjusted Data Set along an arbitrary axis

Step 3: Calculate the covariance matrix

Since the data set is 3 three dimensional, so the covariance matrix will be 3×3 as discussed in section 3.2.2. Now, the covariance matrix of new adjusted data set is:

Eigenvectors

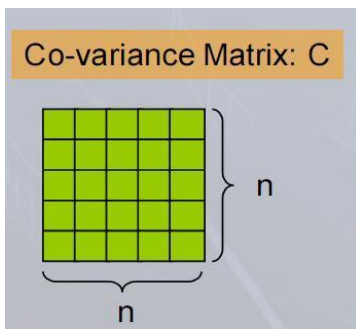
$$V = \begin{bmatrix} 0.1710 & 0.1806 & -0.9686 \\ -0.9648 & -0.1688 & -0.2019 \\ -0.2000 & 0.9690 & 0.1454 \end{bmatrix}$$

Where,

Eigenvalue= 0.0016 for eigenvector $\begin{bmatrix} 0.1710 \\ -0.9648 \\ -0.2000 \end{bmatrix}$

Eigenvalue= 0.0016 for eigenvector $\begin{bmatrix} 0.1806 \\ -0.1688 \\ 0.9690 \end{bmatrix}$

Eigenvalue= 0.0016 for eigenvector $\begin{bmatrix} -0.9686 \\ -0.2019 \\ 0.1454 \end{bmatrix}$



covariance matrix, $c = \begin{bmatrix} 3.1878 & 0.6630 & -0.4734 \\ 0.6630 & 0.1407 & -0.1041 \\ -0.4734 & -0.1041 & 0.0989 \end{bmatrix}$

Step 4: Calculate the eigenvectors and eigenvalues of the covariance matrix

Since the covariance matrix is square, we can calculate the eigenvectors and eigenvalues for this matrix. These are rather important, as they tell us useful information about our data. The eigenvectors and eigenvalues are:
 Eigenvalues = (0.0016, 0.0288, 3.3970)

It is important to notice that these eigenvectors are both unit eigenvectors ie. Their lengths are both 1. This is very important for PCA, but luckily, most mathematics packages, when asked for eigenvectors, will give you unit eigenvectors.

Step 5: Choosing components and forming a feature vector

Here is where the notion of data compression and reduced dimensionality comes into it. If you look at the eigenvectors and eigenvalues from the previous section, you will notice that the eigenvalues are different values. In fact, it turns out that the eigenvector with the highest eigenvalue is the principle component of the data set.

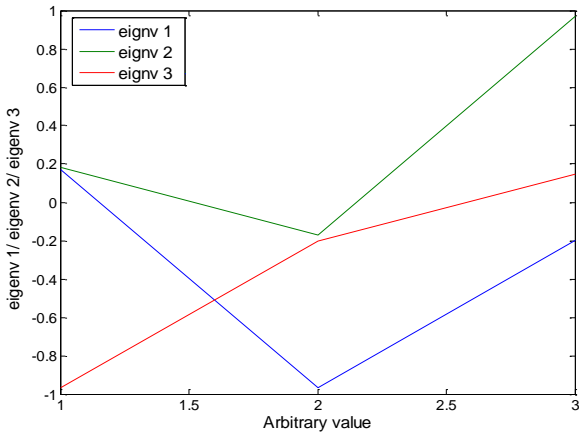
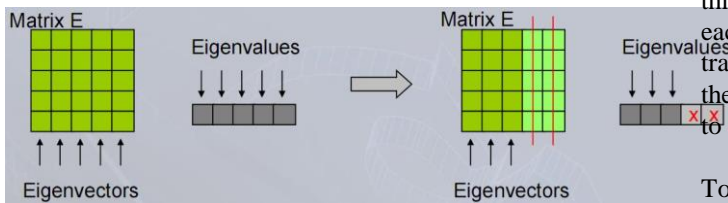


Figure-5.5: Plot of eigenvectors

In our example, the eigenvector with the largest eigenvalue was the one that pointed down the last of the data. It is the most significant relationship between the data dimensions. In general, once eigenvectors are found from the covariance matrix, the next step is to order them by eigenvalue, highest to lowest. This gives you the components in order of significance.

Feature Vector = (eigenvector 1, eigenvector 2 ...
 eigenvector n)

Now, if you like, you can decide to ignore the components of lesser significance. You do lose some information, but if the eigenvalues are small, you don't lose much. If you leave out some components, the final data set will have less dimensions than the original.



To be precise, if you originally have dimensions in your data, and so you calculate eigenvectors and eigenvalues, and then you choose only the first eigenvectors, then the final data set has only dimensions. What needs to be done now is you need to form a feature vector, which is just a fancy name for a matrix of vectors. This is constructed by taking the eigenvectors that you want to keep from the list of eigenvectors, and forming a matrix with these eigenvectors in the columns. For our given example set of data, and the fact that we have 3 eigenvectors, we have three choices. We can either form a feature vector with all of the eigenvectors as:

$$\begin{bmatrix} -0.9686 & 0.1806 & 0.1710 \\ -0.2019 & -0.1688 & -0.9648 \\ 0.1454 & 0.9690 & -0.2000 \end{bmatrix}$$

Or, we can choose first two eigenvectors i.e. first two most significant components as :

$$\begin{bmatrix} -0.9686 & 0.1806 \\ -0.2019 & -0.1688 \\ 0.1454 & 0.9690 \end{bmatrix}$$

Or, we can choose only most significant component i.e. principal component:

$$\begin{bmatrix} -0.9686 \\ -0.2019 \\ 0.1454 \end{bmatrix}$$

Step 6: Deriving the new data set

In order to find out new final data set, we have chosen the components (eigenvectors) that we wish to keep in our data and formed a feature vector, we simply take the transpose of the vector and multiply it on the left of the original data set, transposed.

$$\text{Final Data Set} = \text{Row Feature Vector} \times \text{Row Adjusted Data Set}$$

$$= \text{Feature Vector T} \times \text{Adjusted Data Set T}$$

Where, Row Feature Vector is the matrix with the eigenvectors in the columns transposed so that the eigenvectors are now in the rows, with the most significant eigenvector at the top, and Row Adjusted Data Set is the mean-adjusted data transposed, i.e. the data items are in each column, with each row holding a separate dimension. What will this give us? It will give us the original data solely in terms of the vectors we chose. Our original data set had three dimensional, so our data was in terms of them.

If these axes are perpendicular, then the expression is the most efficient. This was why it was important that eigenvectors are always perpendicular to each other. We have changed our data from being in terms of the axes x, y and z, and now they are in terms of our 3 eigenvectors. In the case of when the new data set has reduced dimensionality, i.e. we have left some of the eigenvectors out, the new data is only in terms of the vectors that we decided to keep. To show this on our data, I have done the final transformation with each of the possible feature vectors. I have taken the transpose of the result in each case to bring the data back to the nice table-like format. I have also plotted the final points to show how they relate to the components.

To show this on our data, I have done the final transformation with each of the possible feature vectors. I have taken the transpose of the result in each case to bring the data back to the nice table-like format. I have also plotted the final points to show how they relate to the components. In the case of keeping both eigenvectors for the transformation, we get the data and the plot found in Figure 5.6 and Figure 5.7. This plot is basically the original data, rotated so that the eigenvectors are the axes. This is understandable since we have lost no information in this decomposition.

$$\text{Transformed Data Set} = \text{Row Final Data Set} \\ = \text{Final Data Set T}$$

Step 7: Getting the old data back

This step is most important to recover the original data set. If we took all the eigenvectors in our transformation will we get exactly the original data back. If we have reduced the number of eigenvectors in the final transformation, then the retrieved data has lost some information. Now, Recall the equation of the final data set that is used in step 6:

$$\text{Final Data Set} = \text{Row Feature Vector} \times \text{Row Adjusted Data Set}$$

$$= \text{Feature Vector T} \times \text{Adjusted Data Set}$$

$$\text{Now, Row Adjusted Data Set} = \text{Final Data Set} \times \text{Row Feature Vector}^{-1}$$

The calculated transformed data set is given below:

Transformed Data Set		
x	y	z
0.1721	-0.0108	-0.0272
-2.5401	-0.1270	-0.0615
-0.0216	0.0253	0.0070
-2.0832	0.0283	0.0231
0.1721	-0.0108	-0.0272
0.1721	-0.0108	-0.0272
1.7388	-0.5398	0.0487
1.8261	0.0416	-0.0713
3.1128	0.1831	0.0494
2.2830	0.1969	0.0132
-3.7226	0.0728	0.0473
-1.2388	0.1114	0.0393
0.1721	-0.0108	-0.0272
-0.0216	0.0253	0.0070
-0.0216	0.0253	0.0070

feature vector, it turns out that the inverse of our feature vector is actually equal to the transpose of our feature vector. This is only true because the elements of the matrix are all the unit eigenvectors of our data set. This makes the return trip to our data easier, because the equation becomes:

$$\text{Row Adjusted Data Set} = \text{Final Data Set} \times \text{Row Feature Vector T}$$

But, to get the actual original data back, we need to add on the mean of that original data (remember we subtracted it right at the start).

$$\text{So, Original Data Set} = [\text{Final Data Set} \times \text{Row Feature Vector T}] + \text{Original Mean}$$

This formula also applies to when you do not have all the eigenvectors in the feature vector. So even when you leave out some eigenvectors, the above equation still makes the correct transform. I will not perform the data re-creation using the complete feature vector, because the result is exactly the data we started with. However, I will do it with the reduced feature vector to show you how information has been lost. Figure 5.8 and 5.9 show this plot. Compare it to the original data plot in Figure 5.2 and Figure 5.4.

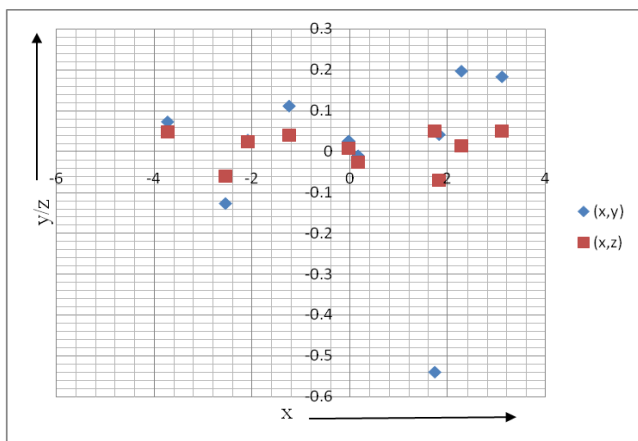


Figure-5.6: Plot of Transformed Data set

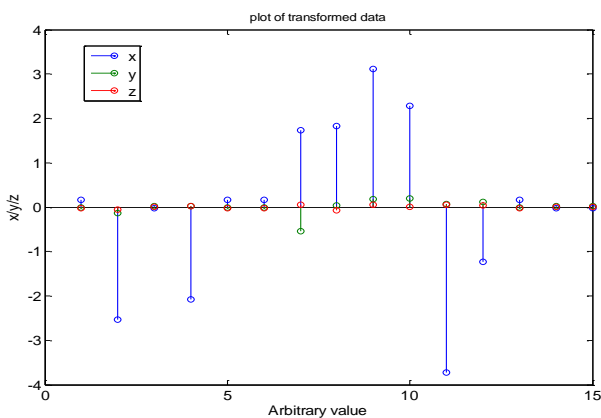


Figure-5.7: Plot of Transformed Data set along an arbitrary axis

For single eigenvector,

$$\text{Reduced feature vector} = \begin{bmatrix} -0.9686 \\ -0.2019 \\ 0.1454 \end{bmatrix}$$

And, Reduced transform data

X
0.1721
-2.5401
-0.0216
-2.0832
0.1721
0.1721
1.7388
1.8261
3.1128
2.2830
-3.7226
-1.2388
0.1721
-0.0216
-0.0216

Where, Row Feature Vector-1 is the inverse of Row Feature Vector. However, when we take all the eigenvectors in our

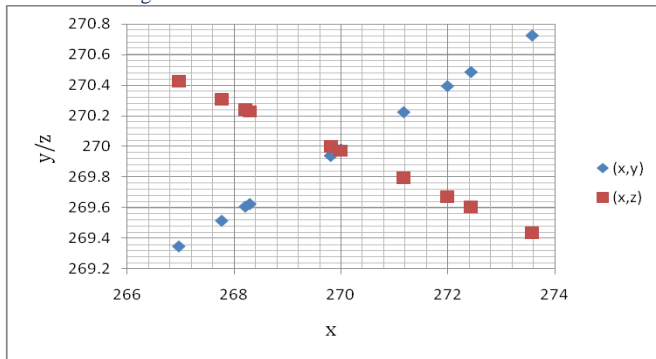


Figure-5.8: Plot of Reconstructed Data Set using only a single eigenvector

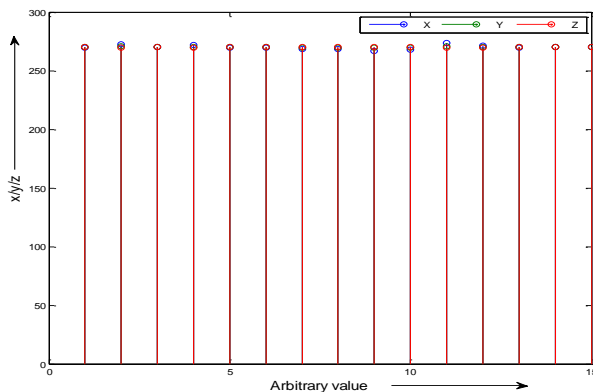


Figure-5.9: Plot of Reconstructed Data Set along an arbitrary axis using only a single eigenvector

CONCLUSIONS

The problem of dimension reduction is introduced as a way to overcome the curse of the dimensionality when dealing with vector data in high-dimensional spaces and as a modelling tool for such data. It is defined as the search for a low-dimensional manifold that embeds the high-dimensional data. A methodological analysis of dimension reduction problems is performed in this paper.

REFERENCES

- [1]. <http://www.xlstat.com/en/products-solutions/feature/principal-component-analysis>
- [2]. http://www.fon.hum.uva.nl/praat/manual/Principal_compone nt_analysis
- [3]. Hatcher, L. & Stepanski, E. A step by step approach to using the SAS system for univariate and multivariate statistics. Cary, NC: SAS Institute Inc, 1994.
- [4]. Lindsay I Smith, A tutorial on Principal Component Analysis, February 26, 2002.
- [5]. Prof. Y. Vijaya Lata , Chandra Kiran Bharadwaj Tungathurthi , H. Ram Mohan Rao, Dr. A. Govardhan , Dr. L. P. Reddy , “ Facial Recognition using Eigenfaces by PCA”. International Journal of Recent Trends in Engineering, Vol. 1, No. 1, May 2009.
- [6]. M. Turk, A. Pentland: Face Recognition using Eigenfaces, Conference on Computer Vision and Pattern Recognition, 3 – 6 June 1991, Maui, HI , USA, pp. 586 – 591.
- [7]. Prof. Y. Vijaya Lata , Chandra Kiran Bharadwaj Tungathurthi , H. Ram Mohan Rao , Dr. A. Govardhan , Dr. L. P. Reddy Department of Computer Science and

Engineering, Gokaraju Rangaraju Institute of Engg&Tech, Jawaharlal Nehru Tech. University, Facial Recognition using Eigenfaces by PCA.

[8] Mrunal S. Bewoor, Assistant Professor, Dept Of Computer Engineering, Emerging Trends in Computer Science and Information Technology -2012(ETCSIT2012), Face Recognition using Principle Component Analysis, Proceedings published in International Journal of Computer Applications® (IJCA).

[9] Sheraz Jadoon ,Face Recognition Approach, School and Health Society, Kristainstad University College, SE- 291 88 Kristainstad, Sweden.

[10] Kailash Chander, Poonam, ” Face Recognition: Today’s Security Tool”, International Journal of Computational Engineering and Management IJCEM, Vol. 15 Issue 2, March 2012.



First Author: Liton Chandra Paul completed his B.Sc. in Electronics & Telecommunication Engineering (ETE) from Rajshahi University of Engineering & Technology (RUET), Bangladesh, in 2012. Currently, he is a lecturer of Electronics and Communication Engineering department at the University of Information Technology & Sciences (UITS), Bangladesh. He has four international journals, two international conference papers and one national conference paper. His research interests include Mobile communication, Wireless & Satellite Communication, Image Processing, Antenna, Digital Signal Processing, Digital Communication, Data Communication etc.



Second Author: Abdulla Al Suman received his B.Sc. in Electronics & Telecommunication Engineering (ETE) from Rajshahi University of Engineering & Technology (RUET), Bangladesh, in 2010. He is pursuing his M.Sc. in EEE from RUET. Currently, he is a lecturer of ETE at the RUET. He has 2 int’l journal and 2 int’l conference paper. His research interests include Wireless Communication, Image Processing, Digital Signal Processing, Information Theory, and Digital Communication.



Third Author: Nahid Sultan received his B.Sc. in Electronics and Telecommunication Engineering from Rajshahi University of Engineering & Technology, Bangladesh. His research interests include various micro strip patch antennas, Digital Communication, Data Communication, Image processing etc.