Volume 5, Issue 5
October 2016

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

Linear and Non Linear Principle Component Analysis

[1] Raghvendra Singh [2] Vandana Gupta, [3] S. K. Tiwari

[1,3] School of Studies in Mathematics, Vikram University, Ujjain, Madhya Pradesh, INDIA
[2] Department of Mathematics, Govt. Kalidas Girls College, Ujjain, Madhya Pradesh, INDIA

E-mail: [1] raghvendraknp16@gmail.com

ABSTRACT

Principle Components Analysis (PCA) is powerful methods used for dimension reduction in various field of engineering. PCA selects principle axis in such a way the sum of the Euclidean distance from the axis is minimized. PCA works well for linearly distributed data, however it fails in case of non-linear distribution of data. Non-Linear Principle Components Analysis (NLPCA) has a capability to deal with non-linear data. The non-linear mapping can be done using Artificial Neural Network. This paper, discusses the PCA and NLPCA in details. We also discuss the Artificial Neural Network (ANN) implementation of NLPCA. The structure of circular PCA is also detailed. Finally, results are shown for both PCA and Circular PCA for linear and circular data and performance is evaluated in terms of MSE.

Keywords: *PCA, Non-Linear PCA , Circular PCA*

1. INTRODUCTION

From mathematics to engineering principal component analysis (PCA) has been used for dimension reduction. This method is very powerful and reduces computational complexity. PCA is simplified form of Fisher's method. Nonlinear principal component analysis (PCA) [1, 2, 3] is a nonlinear generalization of standard PCA. With the fact that PCA is limited to linear components, nonlinear PCA sums up the principal components from straight lines to curves and subsequently explains the inborn data structure by curved subspaces. Identifying and making the description of nonlinear structures is particularly vital for making the analysis of the time series. Hence, Nonlinear PCA is now and then being used to examine the flow of various natural procedures [4, 5]. Be that as it may, the validation of the model complexity of nonlinear PCA is not an easy assignment [6]. Over-fitting could be the result of the regularly predetermined number of accessible samples; besides, in the case of nonlinear PCA over-fitting can likewise happen due to the intrinsic data geometry, which can't be overcome by enhancing the sample number.

The goal is to locate a model with medium complexity. The word nonlinear PCA (NLPCA) is frequently referred to the auto-associative neural network method. In the case of [7] linear subspaces of PCA are supplanted by manifolds and in [8] a neural network methodology is utilized for the purpose of nonlinear mapping. This particular work is concentrated on the auto-associative neural network system to deal with nonlinear PCA and its model validation issue.

2. PRINCIPAL COMPONENT ANALYSIS

With the assumption of linearity PCA can be defined as to re expressing the original data as a linear combination of its basis vectors. Let **X** and **Y** be $m \times n$ matrices related by a linear transformation **P**. **X** is the original recorded data set and **Y** is a re-representation of that data set.

$$\mathbf{PX} = \mathbf{Y} \quad (1)$$

- p_i are the rows of P
- x_i are the columns of X
- y_i are the columns of Y.

Equation 1 represents a change of basis and thus can have many interpretations.

1. P is a matrix that transforms X into Y.
2. Geometrically, P is a rotation and a stretch which again transforms X into Y.
3. The rows of P, $\{p_1, \ldots, p_m\}$, are a set of new basis vectors which expresses the columns of X.

The goal is summarized as follows. .Find some orthonormal matrix P where

$Y = PX$ such that $S_Y \equiv \frac{1}{n-1} YY^T$ is diagonalized.

The rows of P are the principal components of X.

$$S_Y = \frac{1}{n-1} YY^T = \frac{1}{n-1} (PX)(PX)^T \quad (2)$$

$$S_Y = \frac{1}{n-1} PXX^T P^Y = \frac{1}{n-1} PAP^Y \quad (3)$$

Note that we defined a new matrix $A \equiv XX^T$, where A is symmetric. Our aim is to recognize a symmetric matrix (A) is diagonalized by an orthogonal matrix of its eigenvectors. For a symmetric matrix using diagonalization theorem we have:

$$A = EDE^T \quad (4)$$

where D is a diagonal matrix and E is a matrix of eigenvectors of A arranged as columns. We consider a matrix P to be a matrix whose each row is an eigenvector of XX^T.

Volume 5, Issue 5
October 2016

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

By considering, $P \equiv E^T$. Equation 4 modifies as $A = P^T DP$ .thus equation 3 simplifies to.

$$S_Y = \frac{1}{n-1} PXX^T P^T = \frac{1}{n-1} PAP^T \quad (5)$$

$$S_Y = \frac{1}{n-1} P(P^T DP)P^T = \frac{1}{n-1} (PP^T)D(PP^T) \quad (6)$$

$$S_Y = \frac{1}{n-1} (I)D(I) = \frac{1}{n-1} D \quad (7)$$

In general computing of PCA of a data set *X* considers the subtraction of mean of each measurement type and finally computing the eigenvectors of XX^T .

2.1 Limitations of PCA

The main limitations of the PCA are as follows:
The directions with largest variance are assumed to be of most interest.
We only consider orthogonal transformations (rotations) of the original variables. (Kernel PCA is an extension of PCA that allows non-linear mappings).
PCA is based only on the mean vector and the covariance matrix of the data. Some distributions (e.g. multivariate normal) are completely characterized by this, but others are not.
Dimension reduction can only be achieved if the original variables were correlated. If the original variables were uncorrelated, PCA does nothing, except for ordering them according to their variance.

PCA is not scale invariant.
1. Frontal-view of the image is necessary.
2. PCA produces incorrect results when it is over-fitted.
3. Training of images is computationally hard.
4. Threshold is decided heuristically.

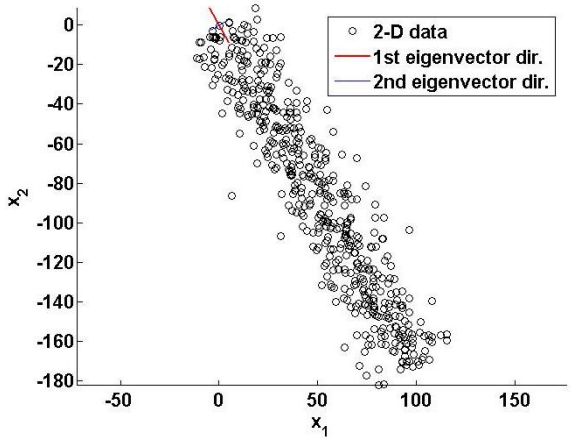


Figure 1: Raw 2D data distribution

In figure 1, raw data distribution is shown, where x_1 is linearly distributed data added with normally distributed noise components, while x_2 is normally distributed random data. Two eigenvectors directions are shown.

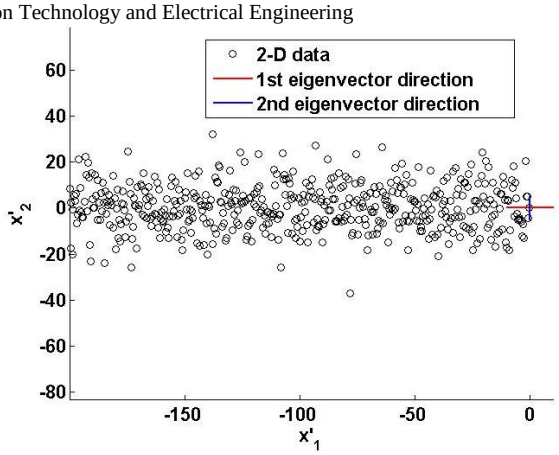


Figure 2: Rotated 2D data distribution

In figure 2, rotated raw data distribution is shown, which is rotated in 2nd eigenvectors direction.

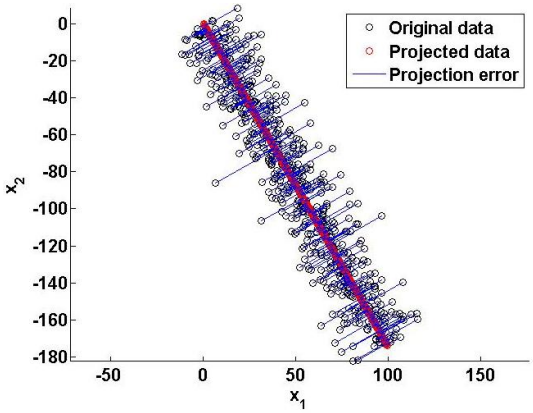


Figure 3: Projection on the primary eigenvector

In figure 3, original data, projected data using PCA is shown, and for randomly generated noise projection error is also shown. PCA selects principal axis such that the maximum variance can be covered, however it completely track the original data, thus with linear data PCA works perfectly fine.

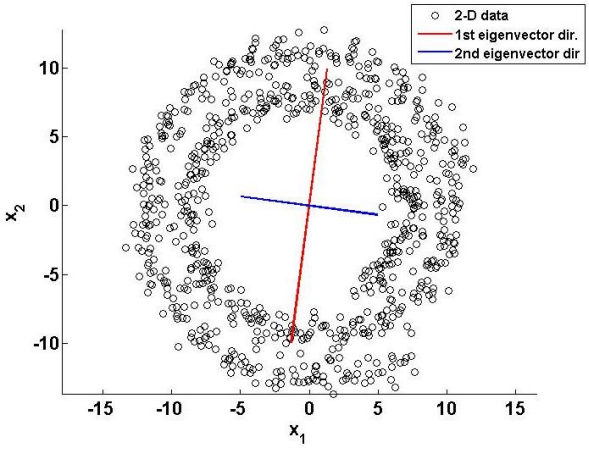


Figure 4: Circular 2D Data distribution

Volume 5, Issue 5
October 2016

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

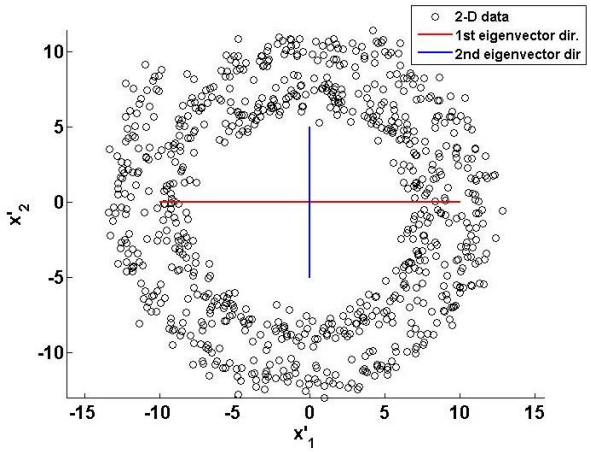


Figure 5: Rotated Circular 2D Data distribution

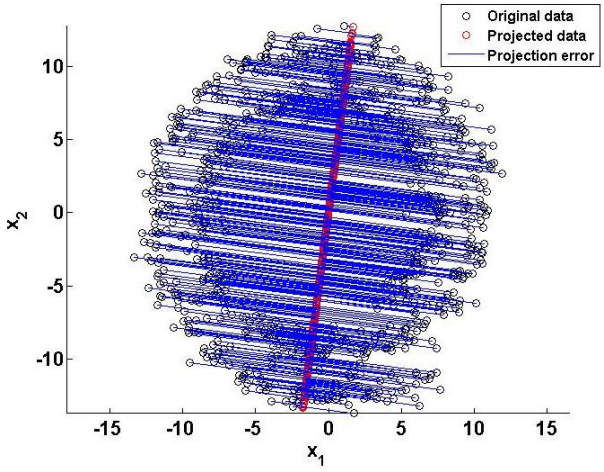


Figure 6: Projection on the primary eigenvector

In figure 4, raw circular data is plotted with normally distributed noise, and two principal axes is also shown. In figure 5, rotated circular data is shown. In finally figure 6, principal components along with projection error is shown. It is clear from the figure that PCA fails to follow circular data. However the variance is very high 35.1826.

3. THE NONLINEAR PCA MODEL

We can represent these data as *m* points in an *n*-dimensional space. In conventional PCA, the first principal component corresponds to the direction of a line running along the principal axis of the resulting cloud of points. There are two equivalent perspectives on the role that the principal axis serves. The first is that the projection of the *m* points on to this line has the maximum dispersion or variance. In other words, the component scores of the first principal component has the most variance. The other perspective is that the average distance of any point from this line is minimized in relation to all possible lines (Figure 7). Nonlinear PCA adopts exactly the same principles but allows for curvilinear lines. In brief, a curve is fitted to the data in *n*-space such that the average distance of the data points from this principal curve is minimized (Figure 8). This heuristic description highlights the intimate relationship between nonlinear PCA and the identification of principal curves or surfaces [9-11] In the case of linear PCA, the principal axes are determined analytically using the eigenvector solution of the $n \times n$ covariance matrix. In nonlinear PCA there is no closed-form solution and iterative techniques are generally employed. These iterative approaches are usually best framed in terms of simple neural networks using gradient ascent or descent on the weights of the connections within the network.

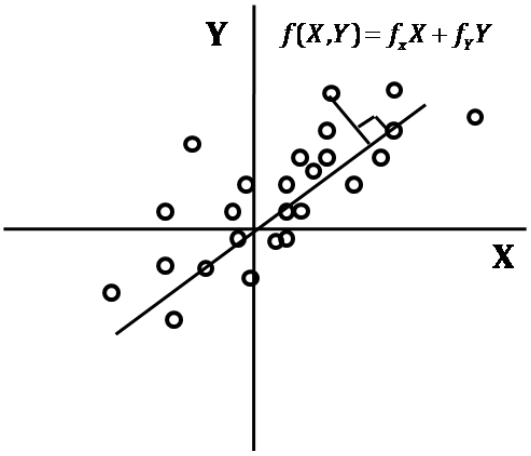


Figure 7: Idea of Linear PCA

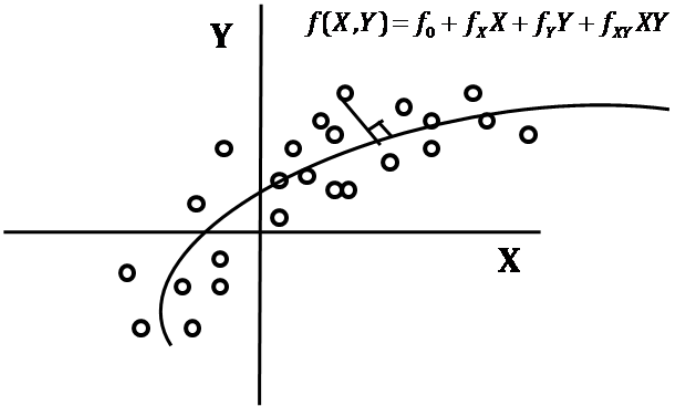


Figure 8: Idea of Non-Linear PCA

It is possible to perform the Nonlinear PCA (NLPCA) with the help of a multi-layer perceptron (MLP) of an auto-associative topology which is also termed as bottleneck, auto-encoder, replicator or sand-glass type network, refer Figure 9.

An identity mapping is performed by the auto-associative network. The result $\hat{x}$ is made to approximate the input x by reducing the squared reconstruction fault $E = \|x - \hat{x}\|^2$.

We can consider that the network have two parts: the extraction function $\Phi_{ext} : X \to Z$ is represented by the first part while on the other hand the inverse function the generation or reconstruction function $\Phi_{gen} : Z \to \hat{X}$ by the second part. A concealed layer in both parts makes it possible for the network to carry out functions of nonlinear mapping. With the use of extra units in the component layer in the middle, we can extend the network to extract component more than one. With the help of a hierarchical nonlinear PCA [12], we can achieve ordered components.

For the method which is proposed for validation, nonlinear PCA should be adapted for the estimation of the missing data. This could be performed with the help of an inverse nonlinear PCA model [13]. This PCA model optimises the generation function by the means of only the second part of the auto

Volume 5, Issue 5
October 2016

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

associative neural network. Due to the fact that the extraction mapping $X \to Z$ is not available, estimation of both the weights w and also the inputs z have to be made which represent nonlinear components' values. The optimisation of both w and z could be done simultaneously in order to reduce the reconstruction error, as illustrated in [13].

We can control the complexity of a model by a weight-decay penalty term [13] included to the error function $E_{total} = E + v\left(\sum_i w_i^2\right)$, we are the network weights. We can vary the impact of the weight-decay term by changing the coefficient v and therefore the model complexity is modified which represents the flexibility of the component curves in nonlinear PCA.

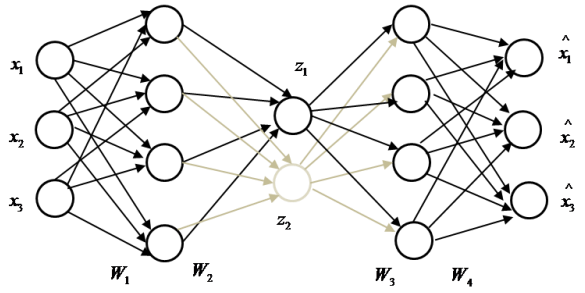


Figure 9: Projection on the primary eigenvector

4. CIRCULAR PCA (CPCA)

Kirby and Miranda [5] introduced a circular unit at the component layer in order to describe a potential circular data structure by a closed curve. As illustrated in Figure 10, a circular unit is a pair of networks units *p* and *q* whose output values z_p and z_r are constrained to lie on a unit circle

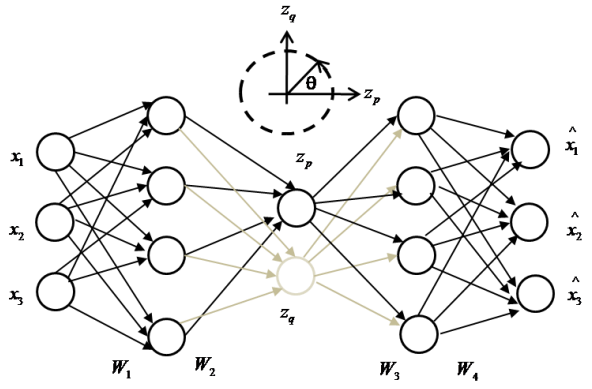


Figure 10: Projection on the primary eigenvector

$$z_p^2 + z_q^2 = 1 \qquad (8)$$

Hence, the values of both units can be represented by a single angular variable θ

$$z_p = \cos(\theta) \text{ and } z_q = \sin(\theta)$$

Here, the forward propagation through the network is described as: Initially, approximately equal to standard units, both units are weighted sums of their inputs z_m provided by the values of all units m in the former layer.

$$a_p = \sum_m w_{pm} z_m \text{ and } a_q = \sum_m w_{qm} z_m \qquad (9)$$

The weights w_{pm} and w_{qm} are of matrix W_2. Biases are not explicitly considered, although, they can be added by putting an additional input with activation set to one.

The sums a_p and a_q are at this point corrected by the radical value $r = \sqrt{a_p^2 + a_q^2}$.

In order to get circularly constraint unit results z_p and z_q

$$z_p = \frac{a_p}{r} \qquad z_q = \frac{a_q}{r} \qquad (10)$$

For backward propagation, we require the derivative of the error function

$$E = \frac{1}{2}\sum_n^N \sum_i^d \left[\hat{x}_i^n - x_i^n\right]^2 \qquad (11)$$

Considering all network weights w. the dimensionality d of the data is provided by the number of observed variables. N represents the samples number. In order to simplify matters, initially we have to consider the error e of a simple $x, e = \frac{1}{2}\sum_i^d \left[\hat{x}_1 - x_i\right]^2$ with $x = \left(x_1, ., ., ., .x^d\right)$. We can extend the resulting derivative with respect to the whole error E provided by the sum over all n samples, $E = \sum_n e^n$.

We are aware with the fact that the derivative of weights of matrices W_1,W_2 ,W_3 and W_4 are obtained by standard back-propagation, the derivatives of the weights w_{pm} and w_{qm} of matrix W_2, which establish connection of units *m* of the second layer with the units *p* and *q* of the component layer, are acquired as follows: Initially, we require the partial derivatives of e with respect to z_p and z_q:

$$\tilde{\sigma}_p = \frac{\partial_e}{\partial z_p} = \sum_j w_{jp}\sigma_j \quad \text{and} \quad \tilde{\sigma}_q = \frac{\partial_e}{\partial z_q} = \sum_j w_{jp}\sigma_j \qquad (12)$$

Here σ_j represents the partial derivatives $\frac{\partial_e}{\partial a_j}$ of units j in the 4th layer. The partial derivatives of e that are required in respect of a_p and a_q of the circular unit pair are

$$\sigma_p = \frac{\partial_e}{\partial a_p} = \left(\tilde{\sigma}_p z_q - \tilde{\sigma}_q z_p\right)\frac{z_q}{r^3} \text{ and}$$

$$\sigma_q = \frac{\partial_e}{\partial a_q} = \left(\tilde{\sigma}_q z_p - \tilde{\sigma}_p z_q\right)\frac{z_p}{r^3} .$$

The final back-propagation formulas for all n samples are

Volume 5, Issue 5
October 2016

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

$$\frac{\partial E}{\partial w_{pm}} = \sum_n \sigma_p^n z_m^n \quad \text{and} \quad \frac{\partial E}{\partial w_{qm}} = \sum_n \sigma_p^n z_m^n \qquad (13)$$

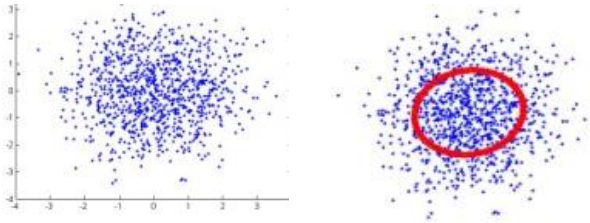
RESULTS

In this part results for circular data is presented using ANN based NLPCA or more precisely Circular PCA.

$$x_1 = \sin t + \eta$$

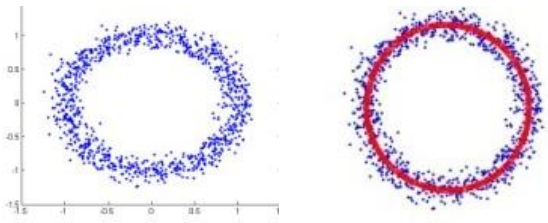
$$x_2 = \cos t + \eta \qquad (14)$$

The data ***x*** lie on a one-dimensional manifold (a circular loop) embedded in two dimensions, plus Gaussian noise η of standard deviation σ varied from 0.001 to 1. 1,000 samples ***x*** were generated from a uniformly distributed factor t over the range $[-\pi, \pi]$, t represents the angle. The weight decay is of 0.001. Network architecture is 3-4-2-4-3 specifies a network of five layers having three units in the input and output layer, four units in both hidden layers, and two units in the component layer, as illustrated in Figure 8. In figure 11(a), raw data is generated which is corrupted by additive Gaussian noise of standard deviation of 1. In figure 11(b) data approximated by CPCA is shown, due to large noise it cannot map all the points in a circle. The obtained Mean Square Error (MSE) is 0.9873.



(a) Raw Data **(b) Circular Map**

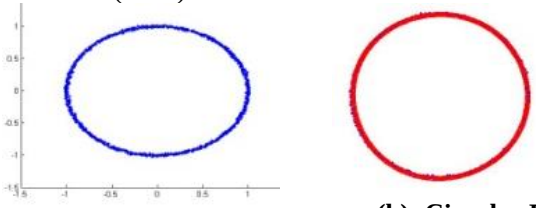
Figure 11: CPCA with Gaussian Noise with variance 1



(a) Raw Data **(b) Circular Map**

Figure 12: CPCA with Gaussian Noise with variance 0.1

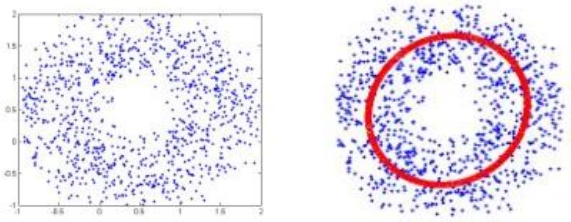
In figure 12(a), raw data is generated which is corrupted by additive Gaussian noise of standard deviation of 0.1. In figure 12(b) data approximated by CPCA is shown, due to large noise it cannot map all the points in a circle. The obtained Mean Square Error (MSE) is 0.0974.



(a) Raw Data **(b) Circular Data**

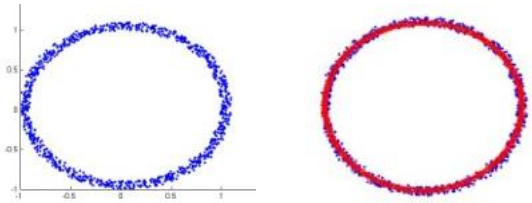
Figure 13: CPCA with Gaussian Noise with variance 0.01

In figure 13(a), raw data is generated which is corrupted by additive Gaussian noise of standard deviation of 0.1. In figure 13(b) data approximated by CPCA is shown, due to large noise it cannot map all the points in a circle. The obtained Mean Square Error (MSE) is 0.0097.



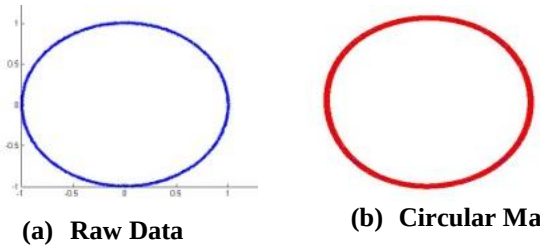
(a) Raw Data **(b) Circular Map**

Figure 14: CPCA with Random Noise with variance 1



(a) Raw Data **(b) Circular Map**

Figure 15: CPCA with Random Noise with variance 0.1



(a) Raw Data **(b) Circular Map**

Figure 16: CPCA with Random Noise with variance 0.01

In figure 14(a), raw data is generated which is corrupted by additive random noise with peak amplitude varying from 0 to 1. In figure 14(b) data approximated by CPCA is shown, due to large noise it cannot map all the points in a circle. The obtained Mean Square Error (MSE) is 0.5679.

In figure 15(a), raw data is generated which is corrupted by additive random noise with peak amplitude varying from 0 to 0.1. In figure 15(b) data approximated by CPCA is shown, due to large noise it cannot map all the points in a circle. The obtained Mean Square Error (MSE) is 0.0573.

In figure 16(a), raw data is generated which is corrupted by additive random noise with peak amplitude varying from 0 to 0.01. In figure 16(b) data approximated by CPCA is shown, due to large noise it cannot map all the points in a circle. The obtained Mean Square Error (MSE) is 0.0058.

In figure 17, MSE vs. number of points is plotted for both Gaussian and random noise. The MSE for random noise is lesser in comparison to Gaussian noise. The effect of Gaussian noise on MSE is predominant till 500 points, thereafter it starts to settle down and become somewhat constant for 1000 number of points. The effect of random noise is settle down after 200 points, and becomes nearly constant after 1000 points.

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

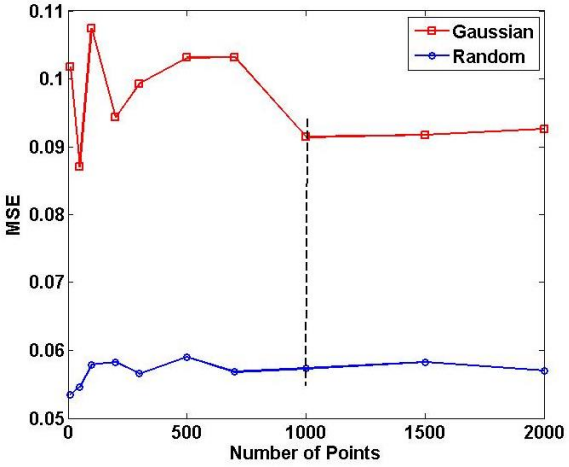


Figure 17: MSE vs. Number of Points

5. CONCLUSIONS

This paper discusses a technique to deal with non-linear data in case of dimension reduction. It is discussed that how non-linear PCA can be realized using ANN. For nonlinear circulator data CPCA has been detailed. Using CPCA mapping of circular data corrupted with Gaussian and random noise is estimated and MSE is obtained. The experiments clearly reveals that the circular PCA exactly matches the circular data when noise is lesser, it is also found that the MSE error is more for Gaussian noise as compared to random noise, however the fluctuation in MSE settle down to a nearly constant value if number of points are 1000.

REFRENCES

[1] Kramer, M.A.: Nonlinear principal component analysis using auto-associative neural networks. AIChE Journal, **37**(2), 233–243 (1991)

[2] DeMers, D., Cottrell, G.W.: Nonlinear dimensionality reduction. In: Hanson, D., Cowan, J., Giles, L., eds.: Advances in Neural Information Processing Systems 5, San Mateo, CA, Morgan Kaufmann, 580–587 (1993)

[3] Hecht-Nielsen, R.: Replicator neural networks for universal optimal source coding. Science, **269**, 1860–1863 (1995)

[4] Malthouse, E.C.: Limitations of nonlinear PCA as performed with generic neural networks. IEEE Transactions on Neural Networks, **9**(1), 165–173 (1998)

[5] Kirby, M.J., Miranda, R.: Circular nodes in neural networks. Neural Computation, **8**(2), 390–402 (1996)

[6] Herman, A.: Nonlinear principal component analysis of the tidal dynamics in a shallow sea. Geophysical Research Letters, 34, L02608 (2007)

[7] Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science, 313 (5786), 504–507 (2006)

[8] Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. Science, 290 (5500), 2323–2326 (2000)

[9] D. DeMers and G. W. Cottrell. Nonlinear dimensionality reduction. In D. Hanson, J. Cowan, and L. Giles, editors, Advances in Neural Information Processing Systems 5, pages 580{587, San Mateo, CA, 1993. Morgan Kaufmann.

[10] B.-W. Lu and L. Pandolfo. Quasi-objective nonlinear principal component analysis. Neural Networks, 24(2):159{170, 2011. doi: 10.1016/j.neunet.2010.10.001.

[11] A. Ilin and T. Raiko. Practical approaches to principal component analysis in the presence of missing values. Journal of Machine Learning Research, 11:1957-2000, 2010.

[12] Cichocki, A., Amari, S.: Adaptive Blind Signal and Image Processing: Learning Algorithms and Applications. Wiley, New York (2003)

[13]. Scholz, M.: Approaches to analyse and interpret biological profile data. PhD thesis, University of Potsdam, Germany (2006) URN: urn:nbn:de:kobv:517-opus-7839, URL: http://opus.kobv.de/ubp/volltexte/2006/783/.