CS571 Project Report

SATYAM SINGH T21012

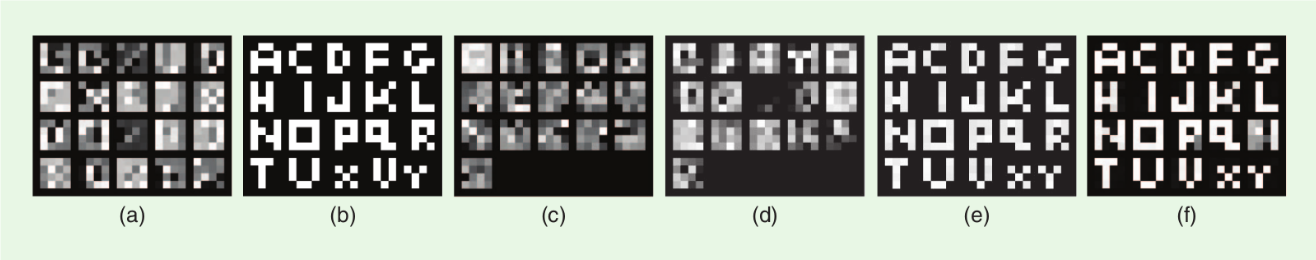
1- Summary

This project emphasis on how an important component of image can be recognised from the noisy set of image with help of various techniques which majorly focuses on decomposition of image which is represented in the form of array. Approach to analysis is that it uses dimensionality reduction and linear algebra concepts to recognize images. This approach is computationally less expensive and easy to implement and thus used in various applications at that time such as image recognition.

PCA (Principal Component Analysis) is a dimensionality reduction technique. It uses Eigenvalues and Eigen Vectors to reduce dimensionality and project a training sample/data on small feature space.

So, basically I have taken sample of images and added Gaussian noise to and tried to recognised my samples with the help of various techniques like PCA,NMF, dictionary learning.

2 –Introduction



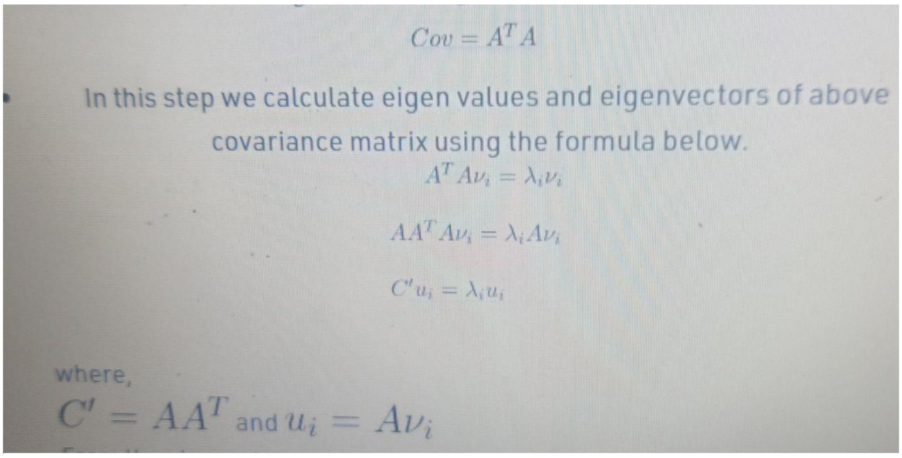
[FIG1] Learning underlying causes from a set of noisy observations of English letters. A subset of 20 noisy 4 × 4 images is shown in (a). These samples have been generated as linear combinations of two letters randomly chosen from the alphabet in (b), and they have been corrupted by additive Gaussian noise. When run of 5,000 such samples, PCA and ICA find the same number of components as the dimension of the signal. Therefore, they cannot find the underlying 20 letters. Sparse coding [4] learns an overcomplete dictionary of 20 components, thus it can separate these causes and find all 20 letters from the original alphabet. K-SVD [5] performs similarly, i.e., it finds almost all of the letters. However, since the implementation of K-SVD [5] uses MP for the sparse approximation step, it converges to a local minimum resulting in some repeated letters in the learned dictionary. (a) Noisy samples; (b) original causes; (c) PCA; (d) ICA; (e) sparse coding; and (f) KSVD.

1.Generate the images as described above.
2.Determine the components using PCA, NMF and dictionary learning. Check corresponding functions sklearn.decomposition. DictionaryLearning, sklearn.decomposition.PCA, sklearn.decomposition.NMF
3.Compare the results and see if the underlying (clean) images can be reproduced.

This problem is useful because PCA is an unsupervised linear dimensionality reduction algorithm to find a more meaningful basis or coordinate system for our data and works based on a covariance matrix to find the strongest features of your samples.

3 –Solution

We have tried to solve this problem with help of decomposition of matrix. We have converted our image into single vector then we found mean of these vectors and subtracted each from individual image vector. Calculated covariance matrix from it and tried to find Eigen values and Eigen vectors



$$Cov = A^T A$$

- In this step we calculate eigen values and eigenvectors of above covariance matrix using the formula below.

$$A^T A\nu_i = \lambda_i \nu_i$$

$$AA^T A\nu_i = \lambda_i A\nu_i$$

$$C' u_i = \lambda_i u_i$$

where,

$C' = AA^T$ and $u_i = A\nu_i$

Now we calculate Eigenvector and Eigenvalues of this reduced covariance matrix
Now we select the eigen vectors of corresponding to the largest eigenvalues .
Then we used the eigenvectors that we got in previous step.
We take the normalized training faces and represent each face vectors in the linear of combination.
We take the coefficient of eigenvectors and represent the training images in the form of a vector of those coefficients.
Given an unknown face y, we need to first pre process the face to make it centered in the image and have the same dimensions as the training image.
We take the vector generated in the above step and subtract it from the training image to get the minimum distance between the training vectors and testing vectors.
Then we try to analyse from the minimum error and predict the correlation of test image with training image.

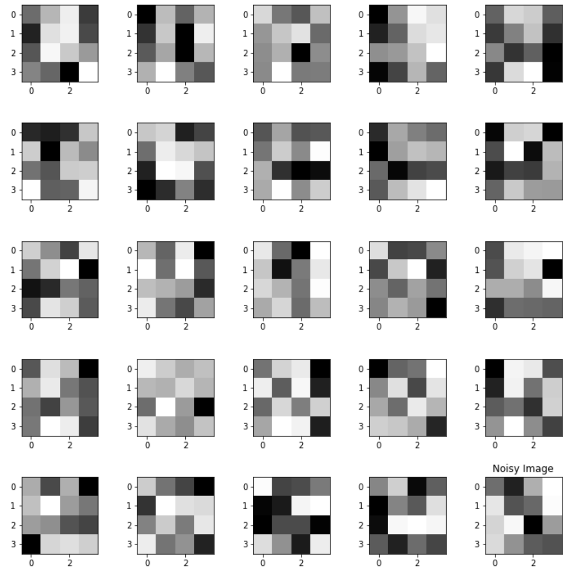
3.1 Assumptions

We have taken non negative parameters of Gaussian noise for NMF.

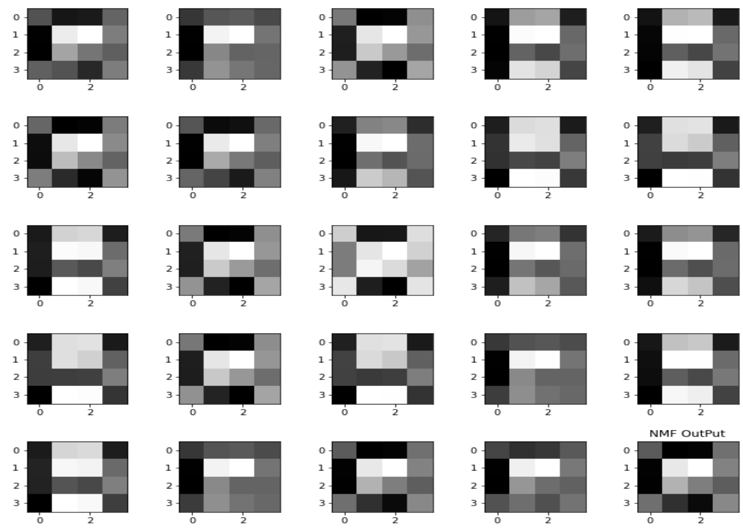
3.2 Algorithms used

1.Generate the images as described above.
2.Determine the components using PCA, NMF and dictionary learning. Check corresponding functions sklearn.decomposition.DictionaryLearning, sklearn.decomposition.PCA, sklearn.decomposition.NMF
3.Compare the results and see if the underlying (clean) images can be reproduced

4 -Results and analysis



Result obtain after PCA ;images are noisy because of reduction of dimensions.



5 –Conclusion

We deduce that the noise added is not removed completely but important attributes of image are retrieved from noisy image and hence we can say that it can be helpful in image recognise if we have huge set of training data then we can easily recognised our test images.

It was difficult to retrieve image for set of images greater than 16 as we have taken image of order (4*4) so the diagonal matrix of eigen values misses out important features.

6 -Project Github page

https://github.com/satyam-sn/CS-571-PROJECT-Generate-the-images-and-Determine-the-components-using-PCA-NMF

7- References

1.Paper "Dictionary Learning", by Tosic et al, IEEE Sig Proc Mag

2. PCA using Python (scikit-learn) by Michael Galarnyk

3. https://www.geeksforgeeks.org/ml-principal-component-analysispca/