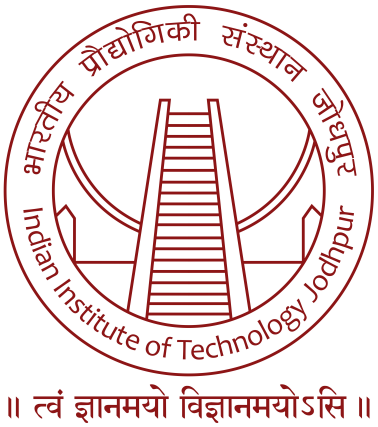
Indian Institute of Technology, Jodhpur



CSL7620 Machine Learning

Assignment 2

Name:- Sahil **Roll No.:-** M21MA210

M.Sc - M.Tech (Data and Computational Sciences)
Department of Mathematics

Submission Date: October 14, 2023

Contents

List of Figures

1 Task:- K means Clustering

Google Collab Link:-
https://colab.research.google.com/drive/1u0dqXh735Gu5G9CIX-_0zknUG-mP7mMn?usp=sharing

1.1 About the given MNIST dataset

The given dataset is the MNIST dataset, primarily used in Machine Learning and Computer Vision. This dataset has a total of $70,000$ images, out of which $60,000$ are contained in the training dataset, and the rest $10,000$ are in the test dataset. It has images of 28×28 pixels consisting of handwritten digits from 0 to 9.
Sample of an image in MNIST Dataset.

Figure 1: Image sample from dataset

1.2 Exploratory Data Analysis

Using the given dataset, we can analyze the following properties:-

- Visualisation of Data (containing images). Sample images are shown:-

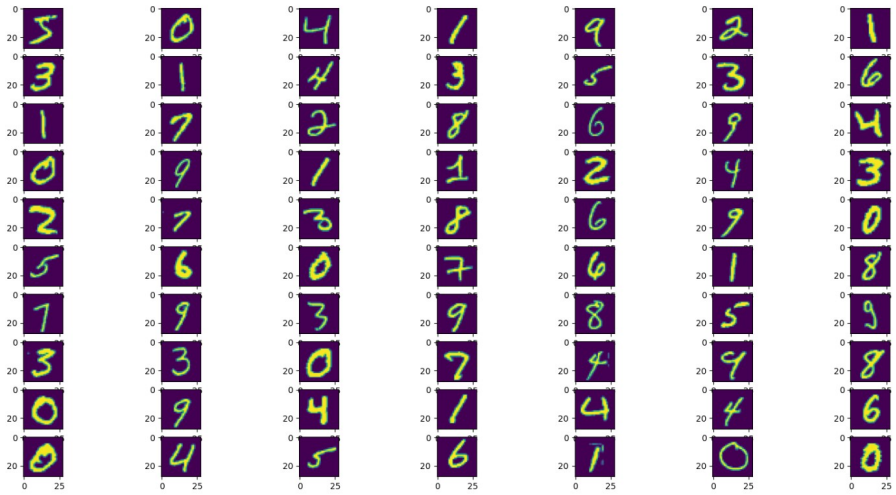


Figure 2: Visualising the images of the given MNIST dataset

- Digit(Label) Count in the training dataset

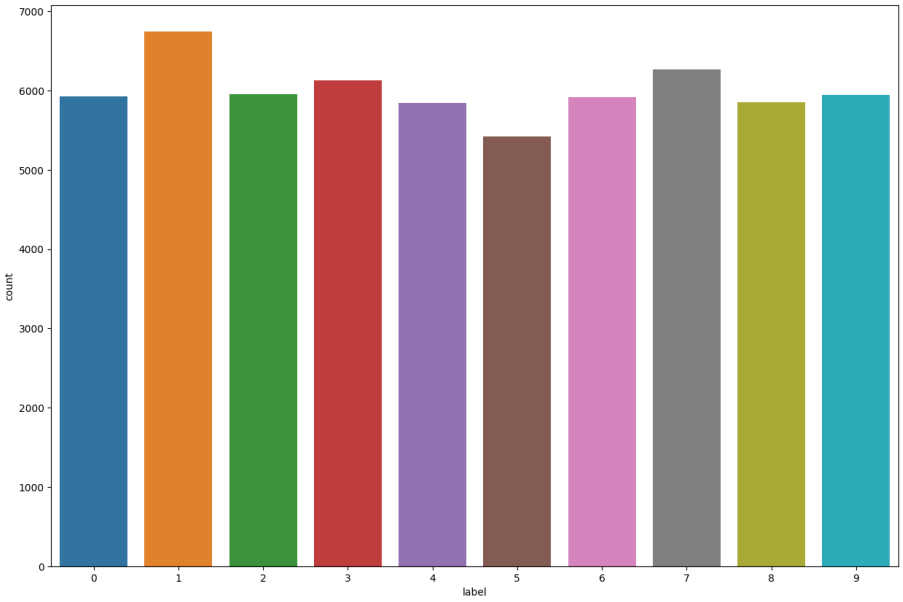


Figure 3: Digit Count in the Training dataset

- Digit(Label) Count in the test dataset

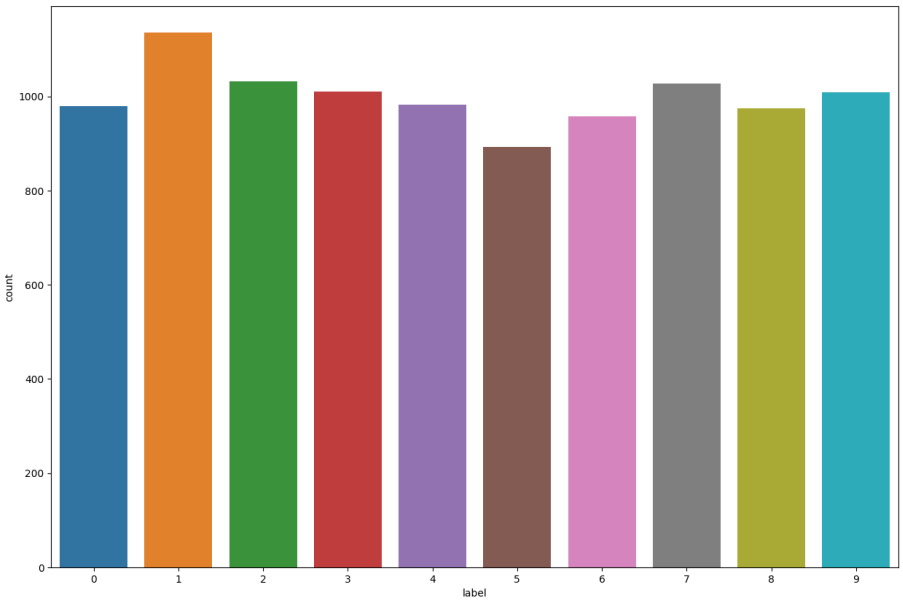


Figure 4: Digit Count in the Test dataset

- Distribution of the pixel intensity among all the images given in the dataset.

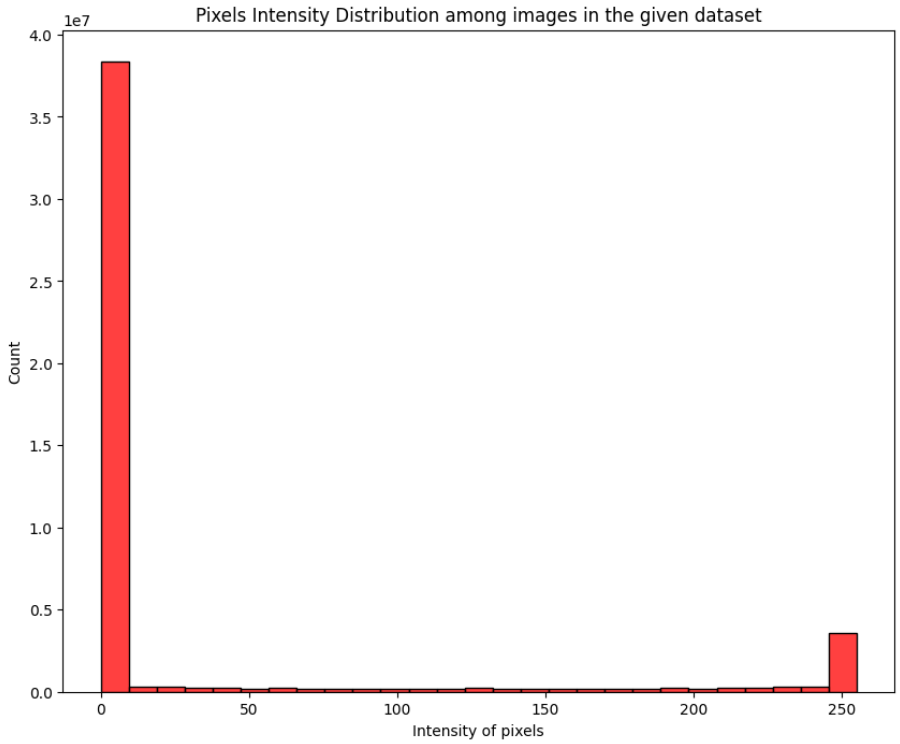


Figure 5: Pixel Intensity Distribution

- Correlation heat map

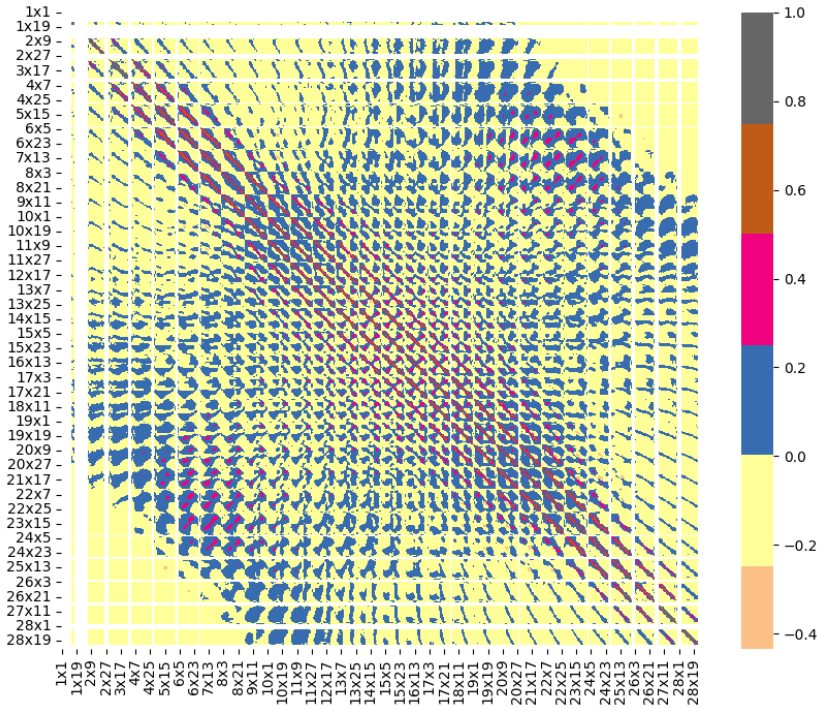


Figure 6: Correlation Heat map

1.3 Performing K means Clustering on MNIST Data-set from scratch using cosine similarity as distance metric instead of Euclidean distance metric

This task has been done on Google Collab Notebook and the python code is attached to the assignment.

Here for this task we used cosine similarity, instead of Euclidean distance metric For any two vectors, v_1 and v_2, the cosine similarity is given as:-

$$\text{Cosine Similarity} = cos(\theta) = \frac{v_1.v_2}{||v_1||||v_2||}$$

In the most general case, we have:-

- If $\theta = 0$, it implies that the vectors overlap and they are similar.
- If $\theta = \frac{\pi}{2}$, it implies that the vectors are dissimilar.

1.4 Visualising the clusters so obtained using K-Means clustering

- Here is the output for K=4 clusters

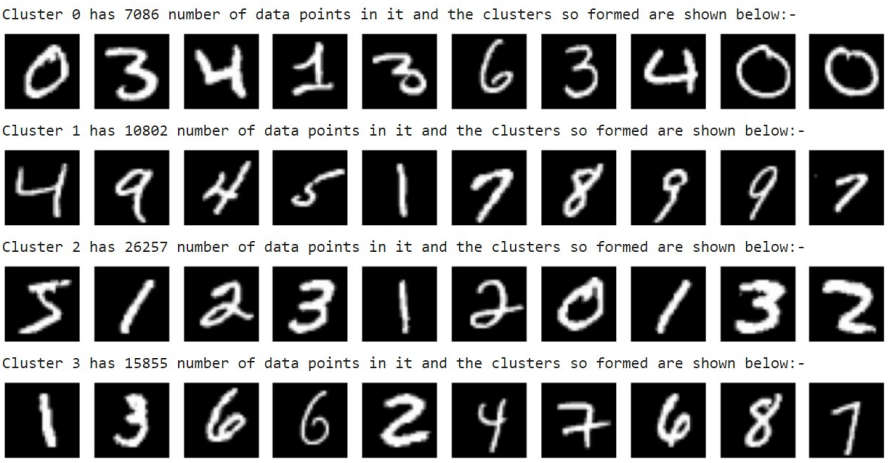


Figure 7: Data in 4 clusters

- Here is the output for K=7 clusters

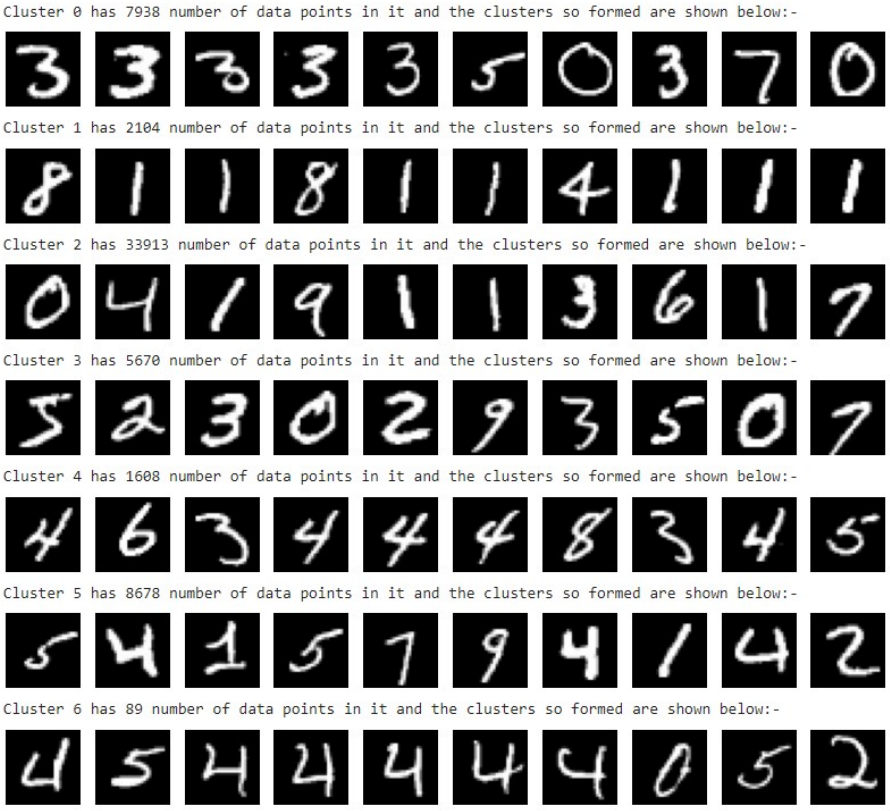


Figure 8: Data in 7 clusters

- Here is the output for K=10 clusters

Figure 9: Data in 10 clusters

1.5 Comment on the cluster characteristics by K-means as done in part(i)

The following are the characteristics of the clusters so formed:-

- The K-means clustering performed hard assignment
- Every cluster has a different size, *i.e* number of data points in each cluster is different.
- The quality of clusters depends on the number of clusters
- Depending on the initialization, K-means take different computational time for different number of clusters.

- It also groups the different digits in a cluster of the particular digit, as it cannot differentiate digits that have approximately similar characteristics as that of members of the cluster. It is because some of the digits look similar to each other. This leads to **Misclassification**
- We can easily observe that K-Means is susceptible to noise and outliers and thus can't handle them easily, it may create a different cluster according to it.
- Due to the hard assignment of the data points, there is randomness of different digits in every cluster
- The worst case can be seen in the case of 4 clusters
- The best case can be seen in the case of 10 cluster
- Please Note: Only a few images corresponding to a cluster are shown, You can check which image is present in which cluster, just by giving its index into the code snippet provided in the attached code. A sample image is shown as below:-

1.6 Python function to find optimal number of clusters for this dataset

This task has been done on Google Collab Notebook and the python code for this task is included in the Code of K-means clustering.
The Elbow curve is attached below which shows that the optimal number of clusters is 10.

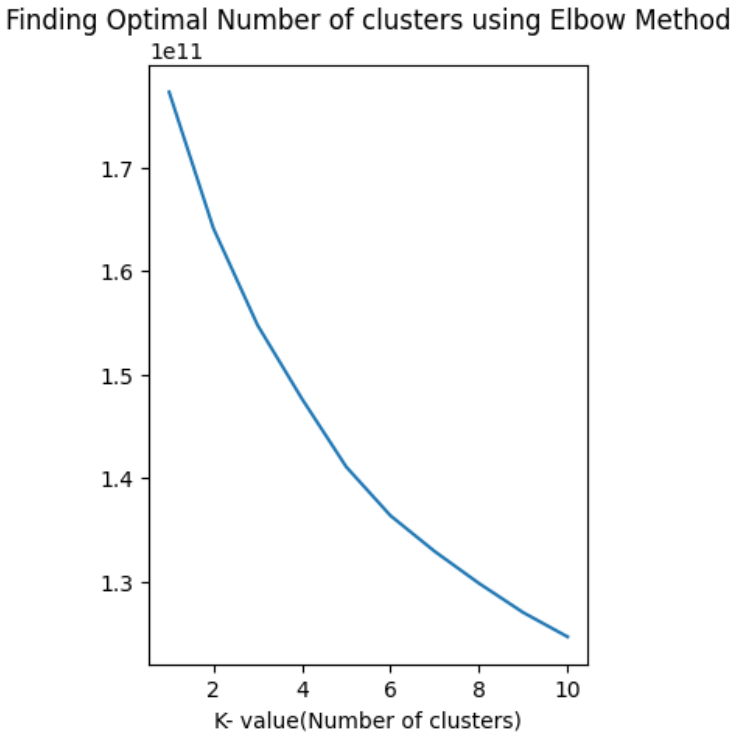


Figure 10: Elbow Curve

2 Task:- PCA from Scratch and GMM clustering

Google Collab Link:-
https://colab.research.google.com/drive/18XgKzS8CLidb-4eHtrv7lueAZH51mbmb?usp=sharing

2.1 Perform PCA on MNIST dataset from scratch and then do GMM Clustering. You can use the library for SVD and GMM

We have to perform PCA using for $32, 64, 128$ components and then we have to do clustering for 4,7,10 clusters.
This task has been done on Google Collab Notebook and the python code is attached in the assignment.

2.2 Visualising the clustered images among different clusters

Below attached are the images of the results obtained using the model for different components and different clusters.

- Number of Components = 32 and Number of Clusters = 4

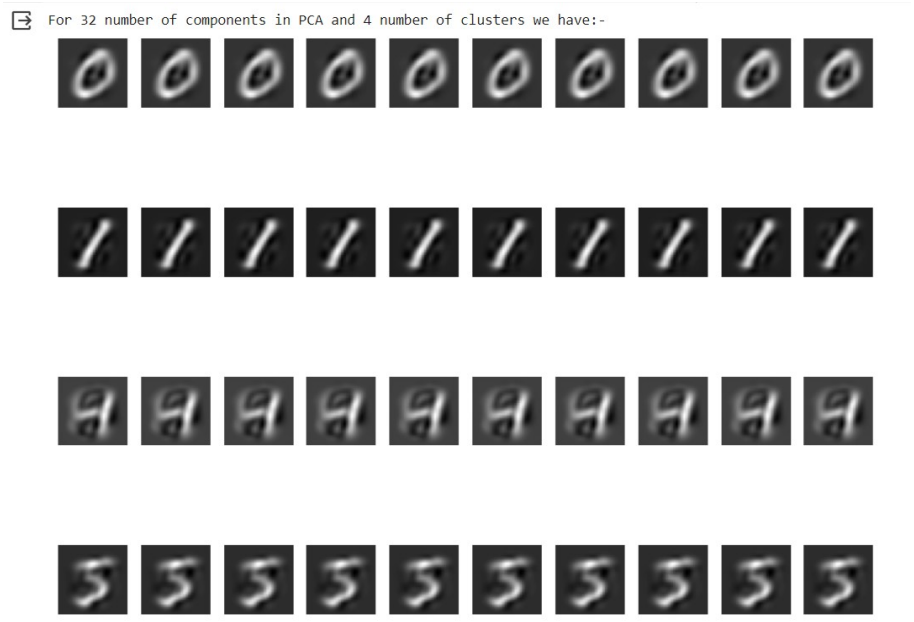


Figure 11: Components = 32, Clusters = 4

- Number of Components = 64 and Number of Clusters = 4

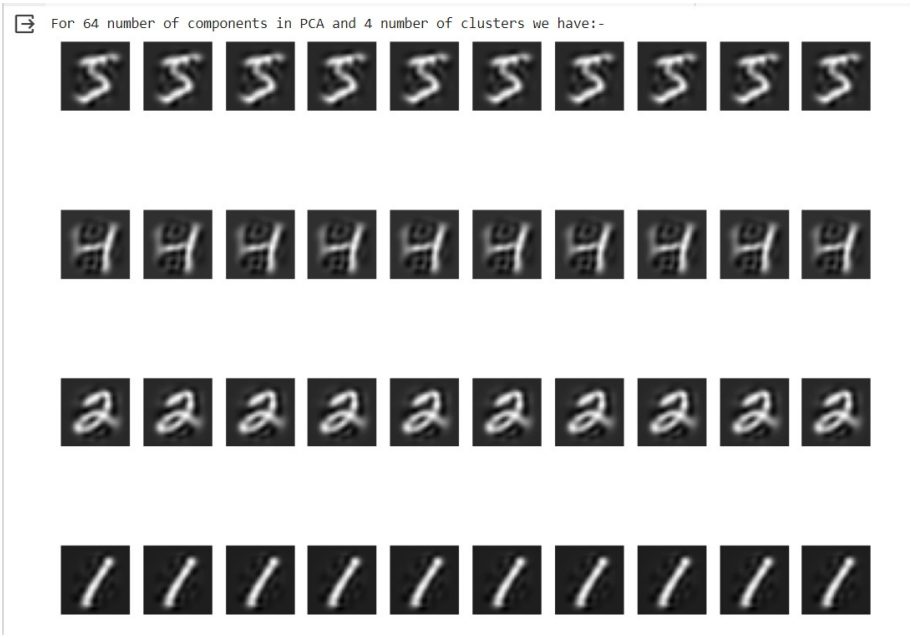


Figure 12: Components = 64, Clusters = 4

- Number of Components = 128 and Number of Clusters = 4

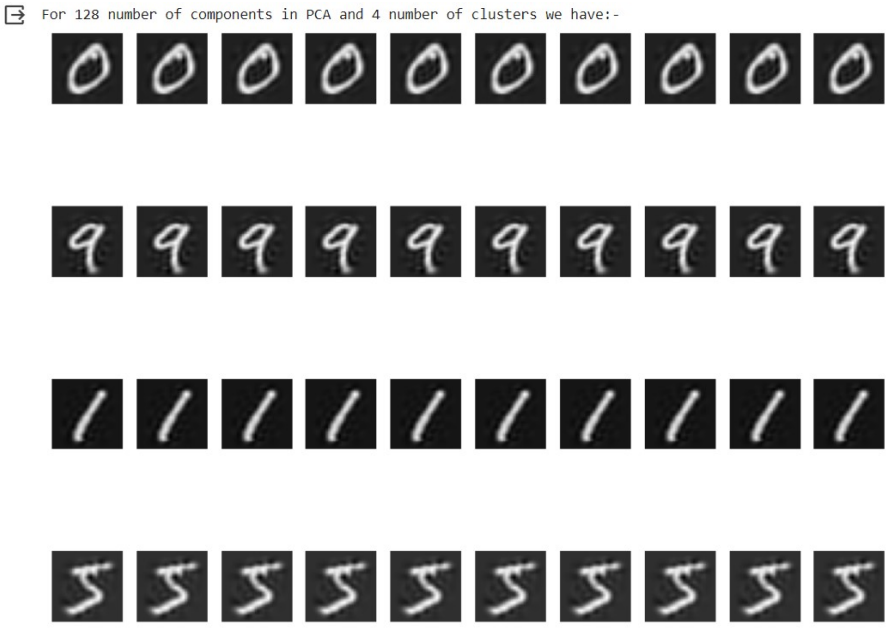


Figure 13: Components = 128, Clusters = 4

- Number of Components = 32 and Number of Clusters = 7

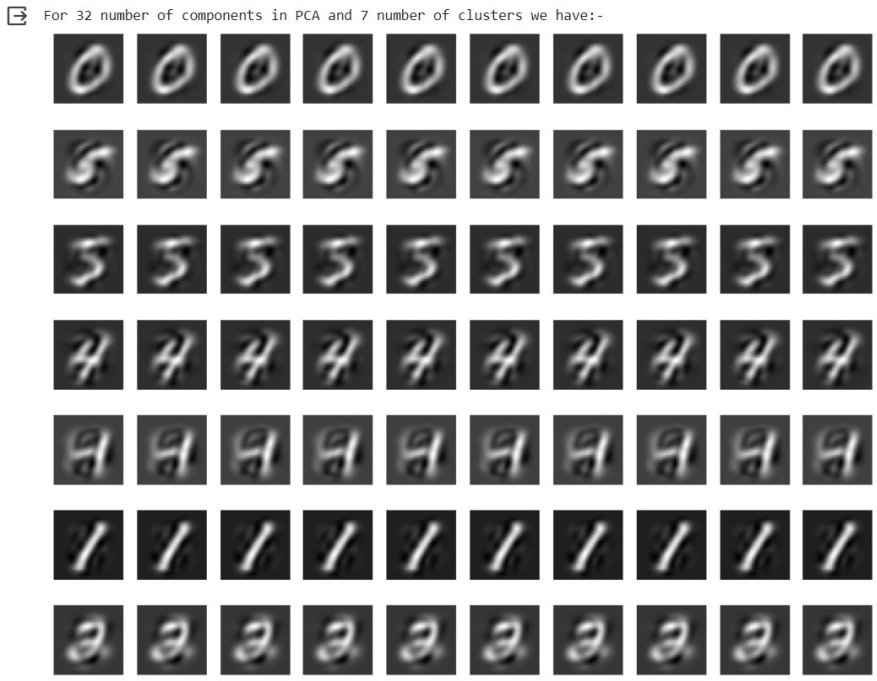


Figure 14: Components = 32, Clusters = 7

- Number of Components = 64 and Number of Clusters = 7

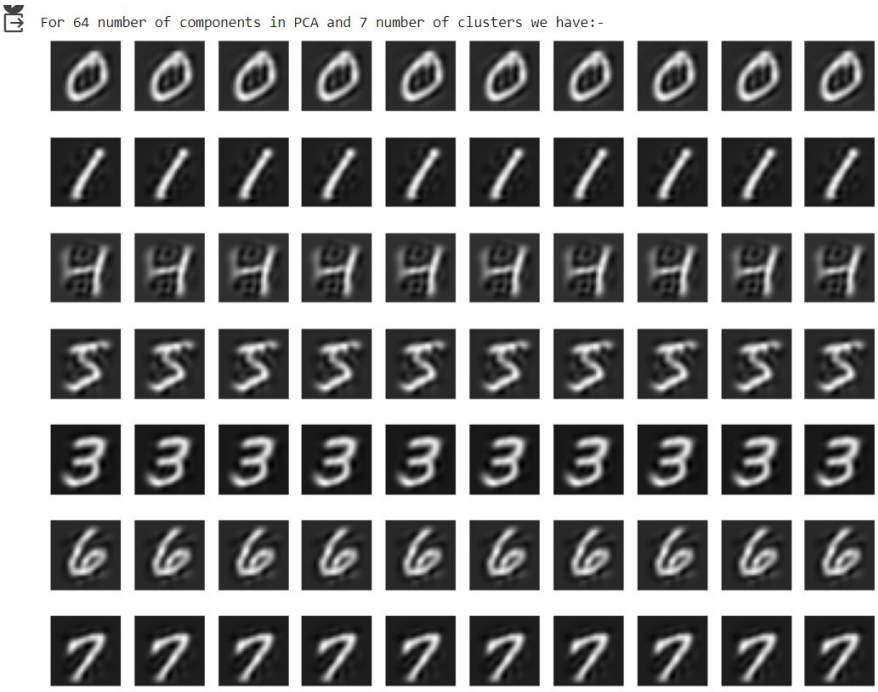


Figure 15: Components = 64, Clusters = 7

- Number of Components = 128 and Number of Clusters = 7

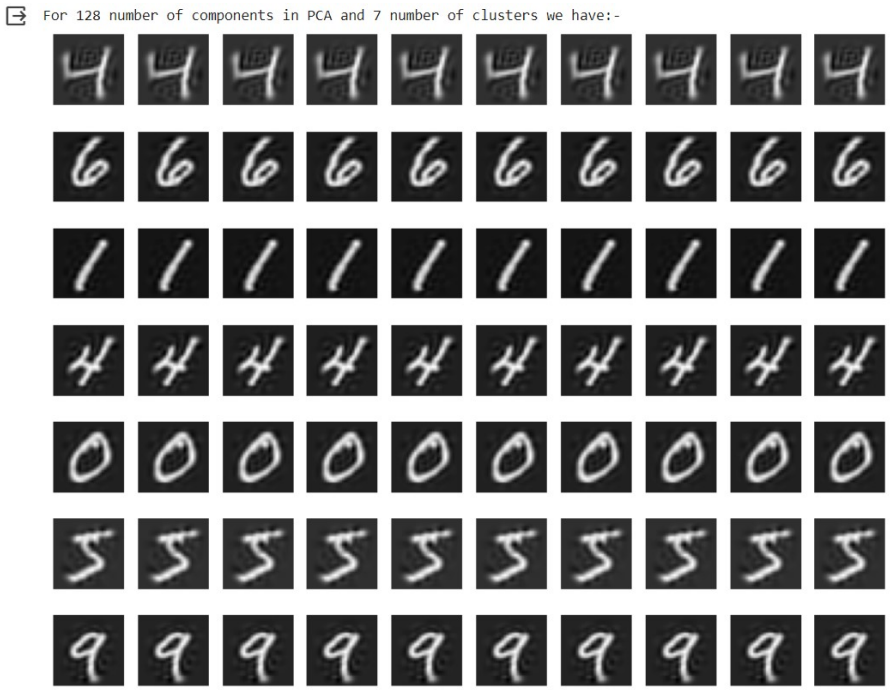


Figure 16: Components = 128, Clusters = 7

- Number of Components = 32 and Number of Clusters = 10

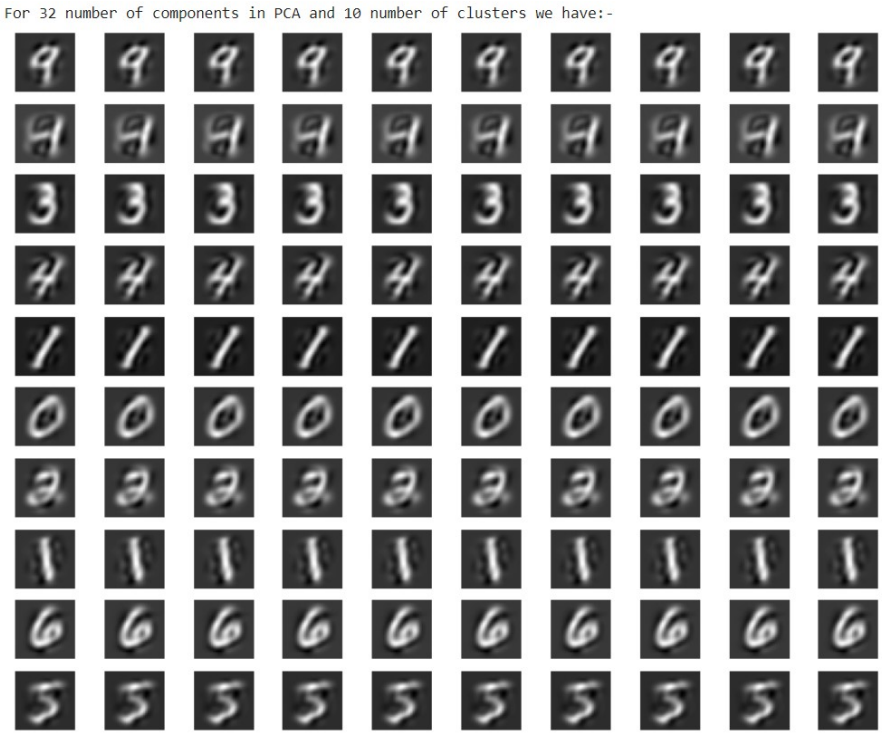


Figure 17: Components = 32, Clusters = 10

- Number of Components = 64 and Number of Clusters = 10

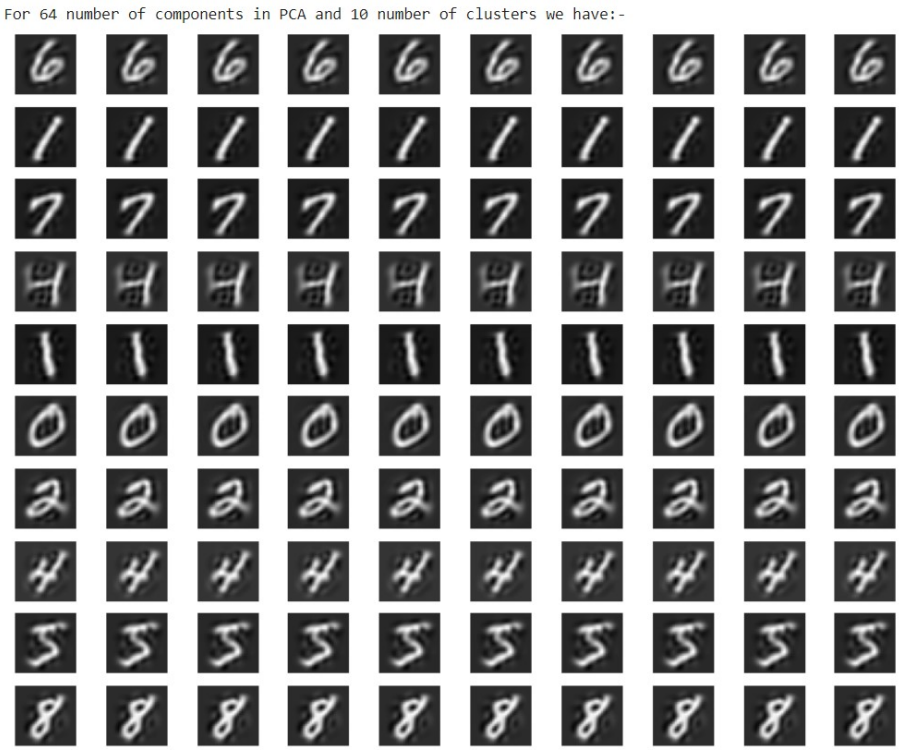


Figure 18: Components = 64, Clusters = 10

- Number of Components = 128 and Number of Clusters = 10

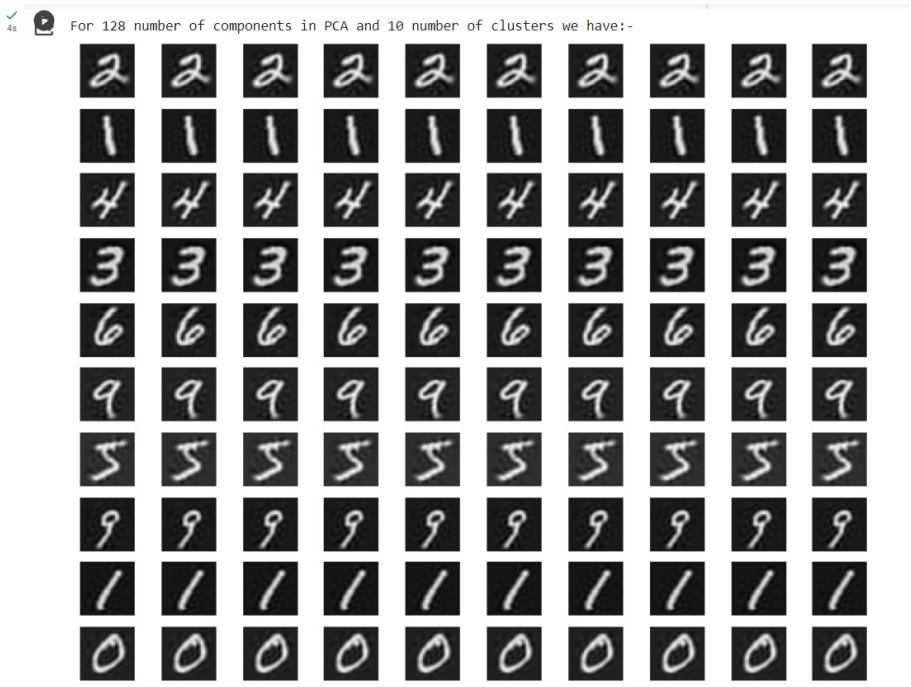


Figure 19: Components = 128, Clusters = 10

2.3 Characteristics of the cluster and comparison with the clusters obtained in Task 1

After the GMM model's performance, the following are the notable characteristics of the clusters obtained as compared to the K-means clusters:-

- GMM performed soft assignment of the clusters by calculating probabilities of belonging of each data point to different clusters and as a result we get the good clustering whereas K-means performs hard assignments.
- The formed clusters are well-defined and have no ambiguity or overlapping pattern. While in K means, overlapping of digits may happen as there may be two digits that share common features and are difficult to classify according to their pixel values.
- Different clusters have different numbers of data points in them, we can say that the shape of clusters so formed are not spherical, they may be elongated, however in case of K-means, the clusters are spherical.
- From the visualizations, we can see that GMM can handle the outliers and noise in the data whereas in the case of K-Means, this was not so.
- It can also be seen that it can handle more complex patterns also as we can see that in K-Means some clusters are not pure as they have impurities or other digits.

2.4 Finding the optimal number of components PCA should choose which covers almost all necessary patterns in the data and will discuss when PCA can fail (BONUS)

- From our model, we have attached the necessary outputs and we can see that the clusters formed after performing **PCA using 128 components** are well-defined and the images are clear enough to identify the digit value it contains. Thus we can say that 128 components should be chosen for this dataset, as they covers almost all necessary patterns in the data. **Thus the optimal number of components are 128.**

- Now we will discuss when PCA can fail:-

  1. It fails when the relationship among the features is non-linear, or we can say that the observed variables are not linearly correlated, as PCA is a linear dimension reduction technique.
  2. It may also fail when the data has not been standardized before applying PCA, it is sensitive to scaling of the data
  3. PCA preserves the direction with max variance and thus we have some information lost in the min variance direction. It might lead to a significant loss of detailed information as in our task we are aimed to recognize the digits, where fine details can also be accounted for.
  4. In our case, the size of the data is large, thus it is computationally expensive.

5. Sometimes, it fails when we try to do classification when data is projected in the direction of principal components, in that case, we use Fischer Linear Discriminant. *Thus we can say that the reduced dimensionality representation obtained using PCA may not be optimal for classifying different classes*
6. PCA is sensitive to outliers as its aim to capture the direction of maximum variance.

3 Resources Used

- https://en.wikipedia.org/wiki/K-means_clustering
- https://en.wikipedia.org/wiki/Mixture_model
- https://www.geeksforgeeks.org/gaussian-mixture-model/
- https://www.geeksforgeeks.org/digital-image-processing-basics/
- https://numpy.org/doc/stable/reference/generated/numpy.linalg.svd.html

********** THE END **********