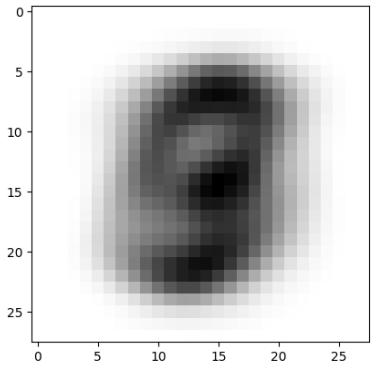
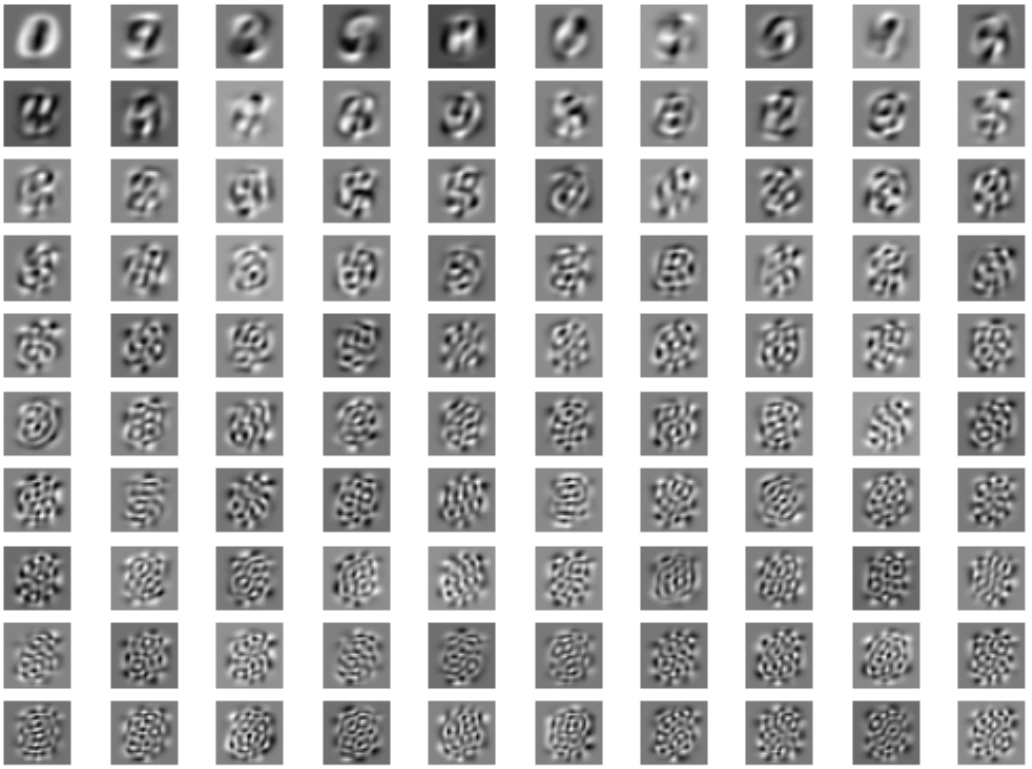
CMPE 481 ASSIGNMENT 3

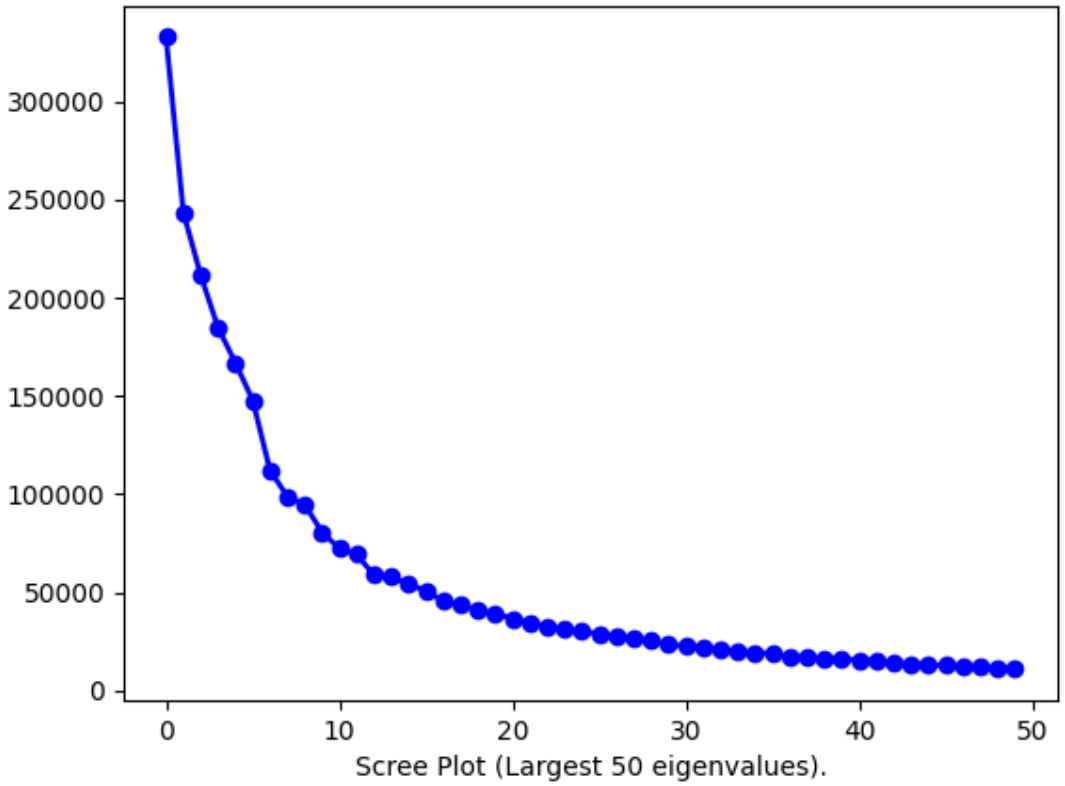
ADALET VEYİS TURGUT - 2017400210

1. Sample digits

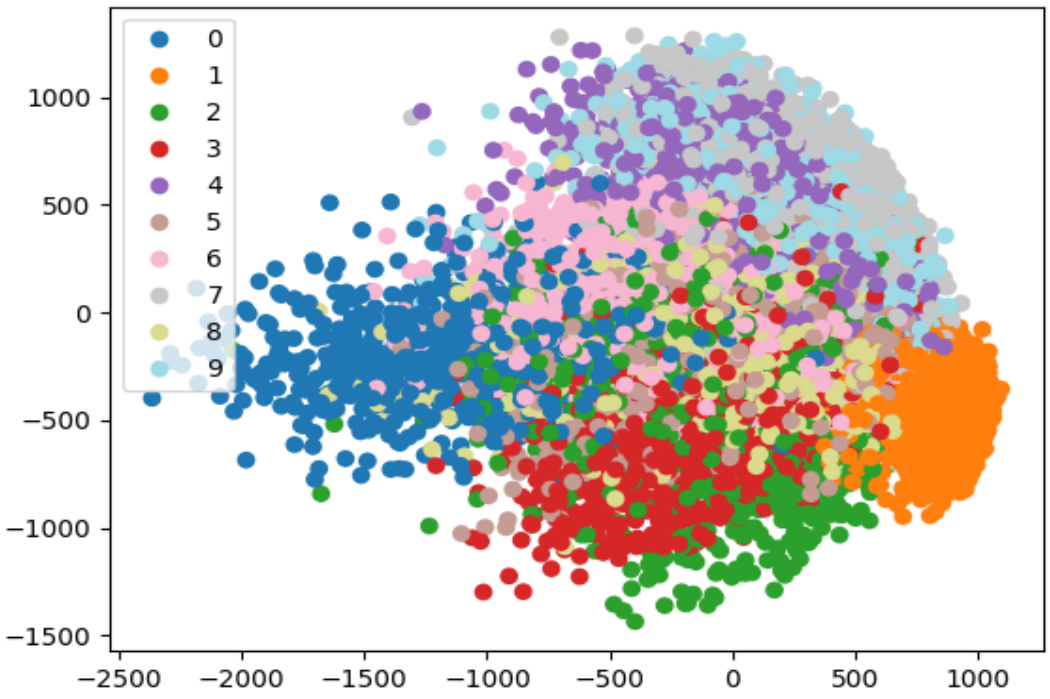
2. Mean image, largest eigenvectors and eigenvalues of training set







3. Dimensionality reduced to 2



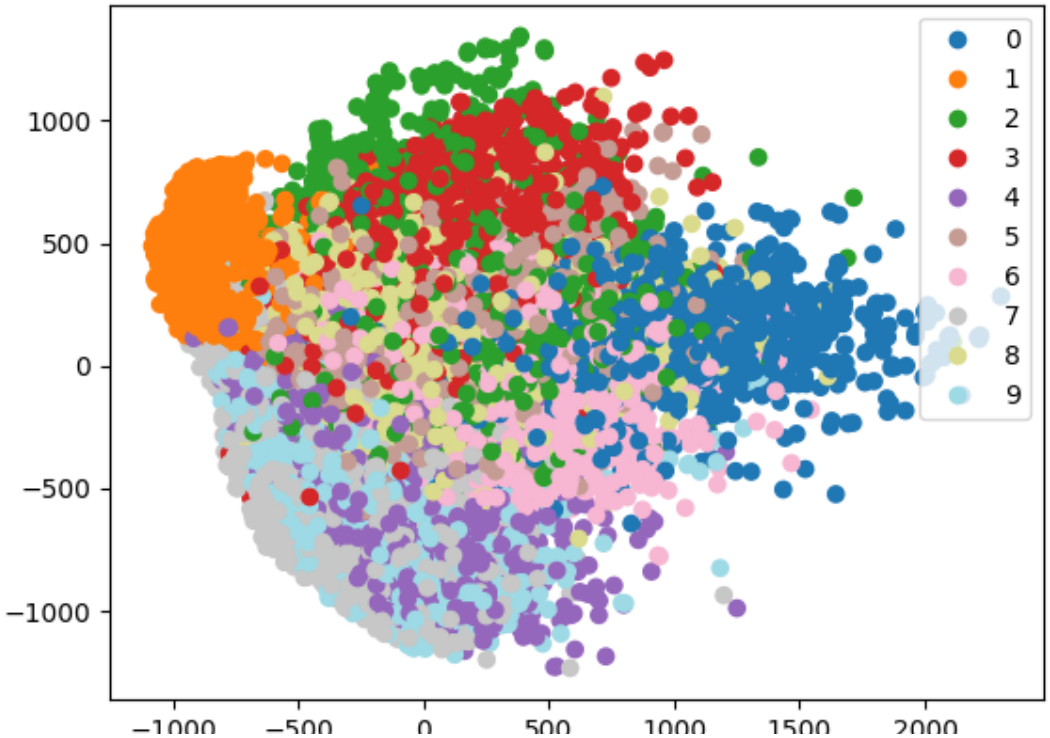
My output and sklearn library's output(initialized with 'PCA' option) is very similar, it's like taking symmetry with respect to the origin. But t-SNE output initialized with 'random' option is very different from mine, but in a better way.

My result is chaotic, it's hard to interpret, but I'll try: I see that 0 and 1 are distinct from other digits, thus well interpretable. 2 and 3 overlapped; 4,7 and 9 overlapped which makes it hard to interpret. Also, I would expect 0,6,8,9 to overlap since they have circles in common.

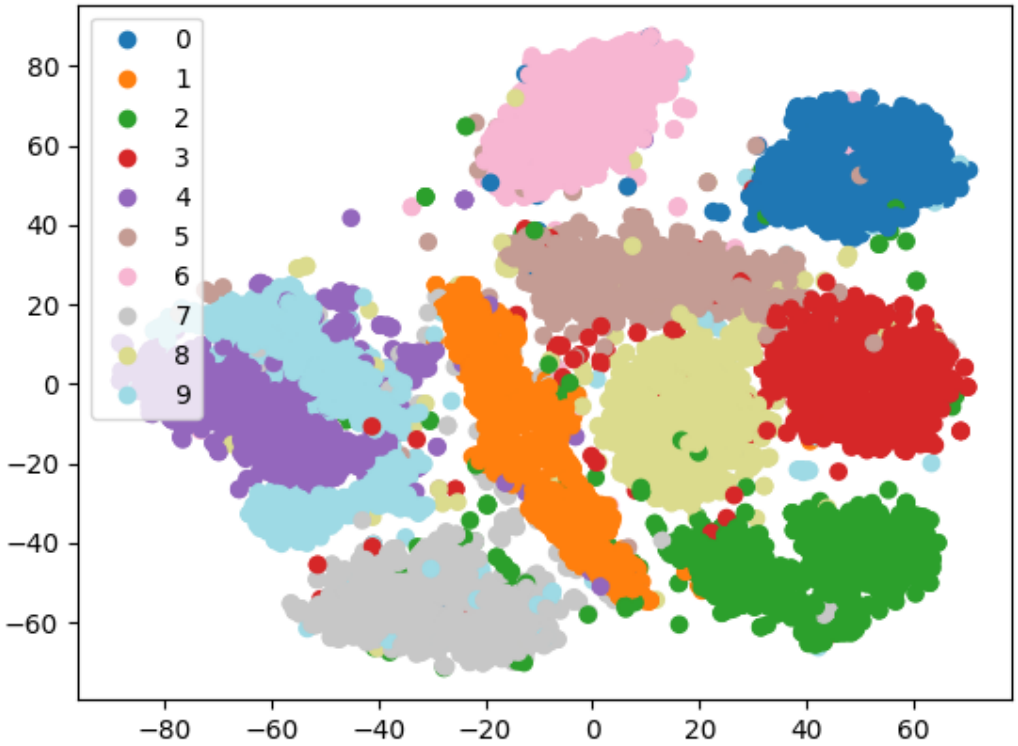
In randomly initialized t-SNE, only 4 and 9 overlapped, other digits are well separated. Result is easy to interpret.

4. t-SNE approach

initialized with ‘pca’ option



initialized with ‘random’ option



5. Fundamentals of the t-SNE approach and comparison to the PCA

5.1. Fundamentals of t-SNE

The main idea behind t-SNE is embedding the points to lower dimensions by minimizing the loss of the neighborhood. Basically in t-SNE, we randomly put the points to the coordinate system, moving these points one at a time until we cluster them. Where should we move a point? We move the point to the direction where its original cluster neighbors are located and the opposite of the direction where other points(points from other clusters) are located. So, we can say that points with the same label will attract, points with different labels will repel the point like a magnet.

To begin with, we first determine the similarity of each point. Let's take a point and call it the main point. We measure the distances between the main point and all other points, and plot it on a gaussian normal curve. The reason we use a gaussian normal curve is that distant points will have very low similarity and closer points will have high similarity. Then, we scale the distances between the main point and the other points such that they add up to 1.

For each point, we do the similarity calculation above and construct a similarity matrix. Diagonal entries are set to 0. If you have realized we have two different scores for a pair of points (since width of the normal plot is very dependent on the neighbors of a point), we take their averages while calculating the direction of movement in the algorithm.

After calculating the similarity matrix, we randomly plot the data to the coordinate system. For every point, we measure the distances between a point and all others and plot it on a "t-distribution" curve. Just like the previous step, we scale the similarity scores such that they add up to 1 and construct a t-similarity matrix. (I made up that name).

The ultimate aim of the algorithm is trying to make the t-similarity matrix similar to the similarity matrix. By moving points little by little, we change the t-similarity matrix and make it more similar to the similarity matrix.

The reason we use t-distribution is that without it clusters would be stacked in the middle since height of the t distribution is smaller than gaussian normal distribution.

5.2. PCA vs t-SNE

- PCA tries to preserve the global structure of data whereas t-SNE tries to preserve the local structure of data.
- PCA is a linear dimensionality reduction algorithm whereas t-SNE non-linear dimensionality reduction algorithm.

- PCA does not involve hyperparameters whereas t-SNE uses hyperparameters like learning rate, number of steps.
- PCA gets highly affected by outliers, t-SNE can handle outliers.
- PCA is a deterministic algorithm, whereas t-SNE is a randomized algorithm.
- PCA works by rotating the vectors for preserving variance, whereas t-SNE tries to minimize the distance between points on a gaussian normal plot.

Advantages of PCA

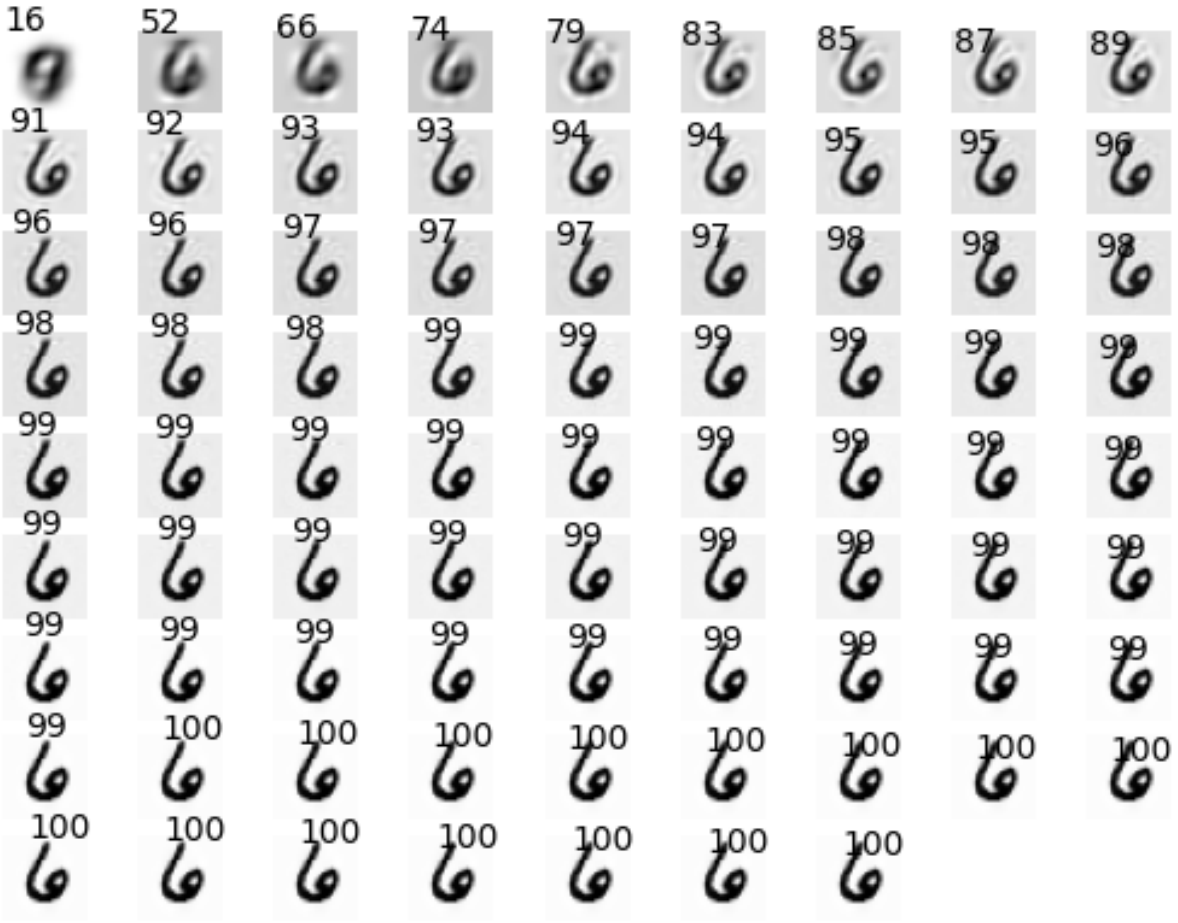
- Since it's deterministic, there is one outcome. On the other hand, two persons may get different outcomes from the same t-SNE visualization.
- Since it preserves local neighborhoods, it's hard to capture global trends in t-SNE. In contrast, interpreting global trends is much easier on PCA.
- Also, since t-SNE computes pairwise scores, it's a much more complex and intense algorithm than PCA.

Advantages of t-SNE

- Since PCA is linear, it is not able to interpret complex polynomial relationships between points, but t-SNE can successfully capture these relationships.
- Since L2 norms are very high for outliers (which PCA tries to minimize) due to square exponent, outliers will drive the PCA components. On the other hand, t-SNE handles outliers.

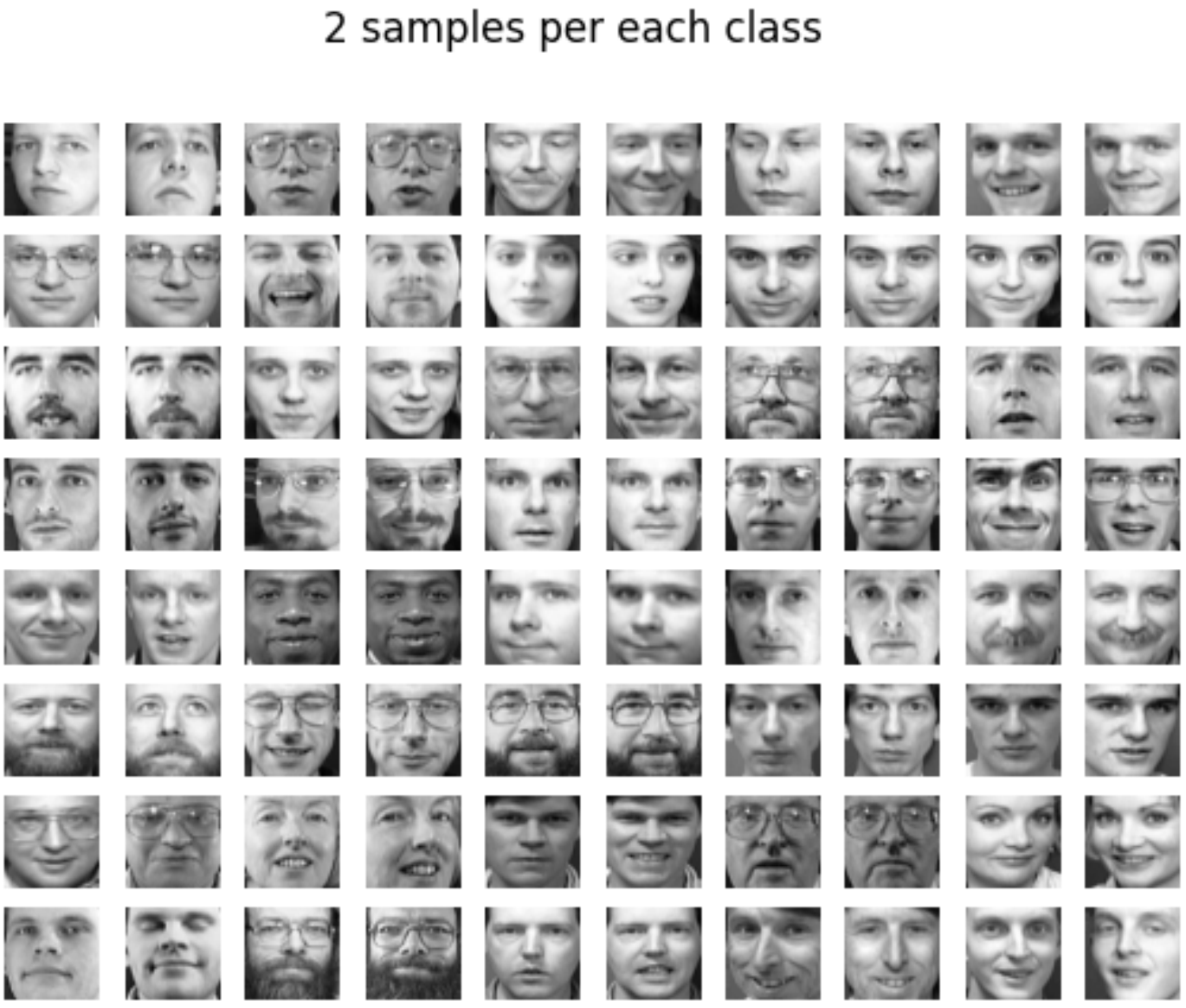
6. Reconstruction

Note: I didn’t reconstruct another non-digit image.

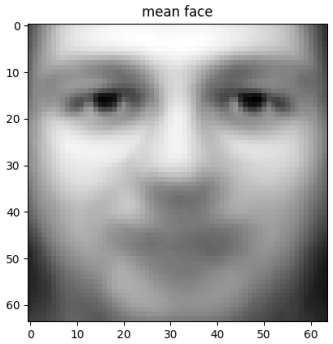


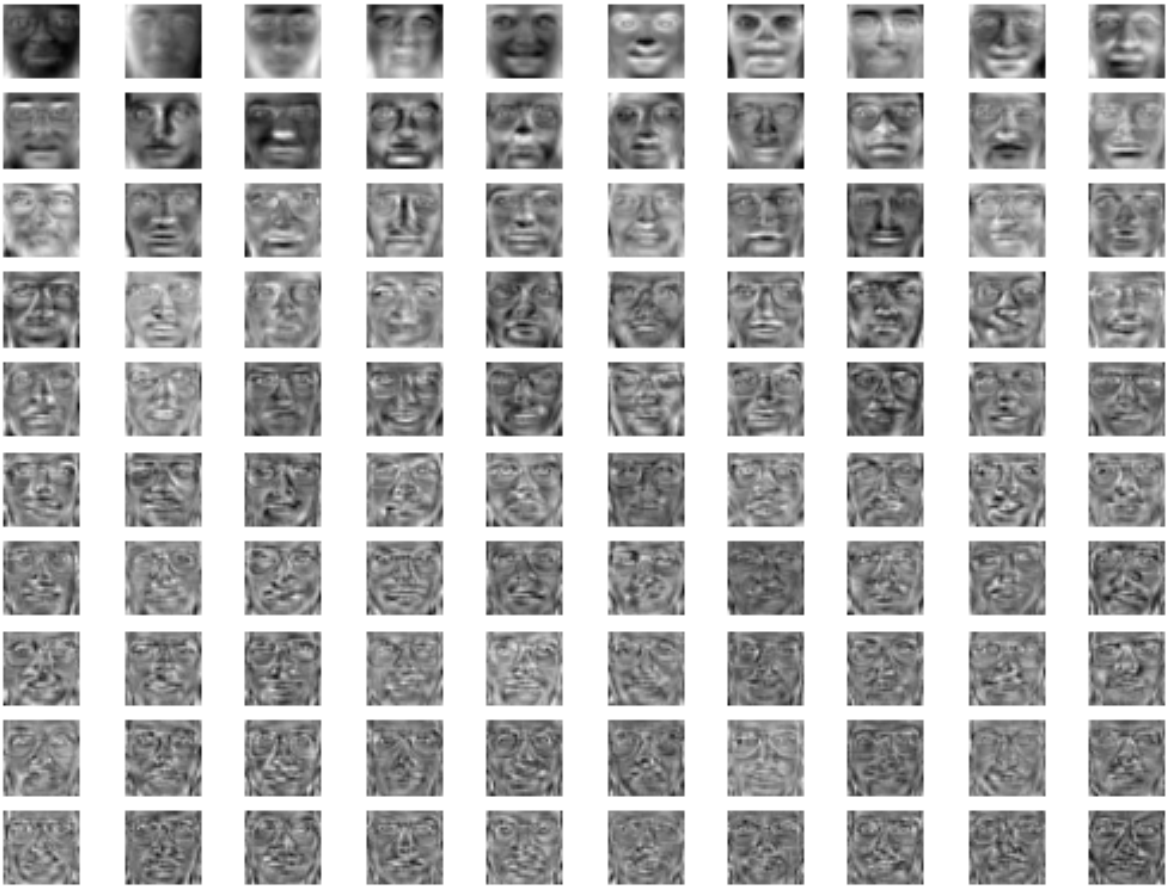
7. Human Face

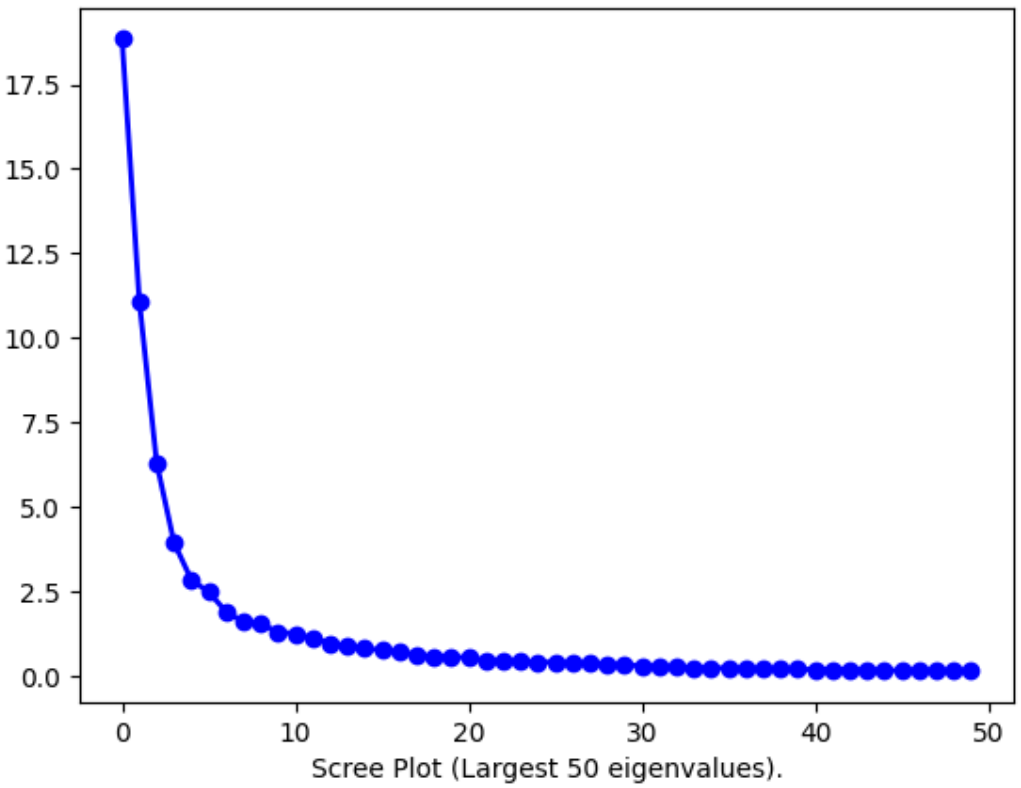
7.1. Sample faces



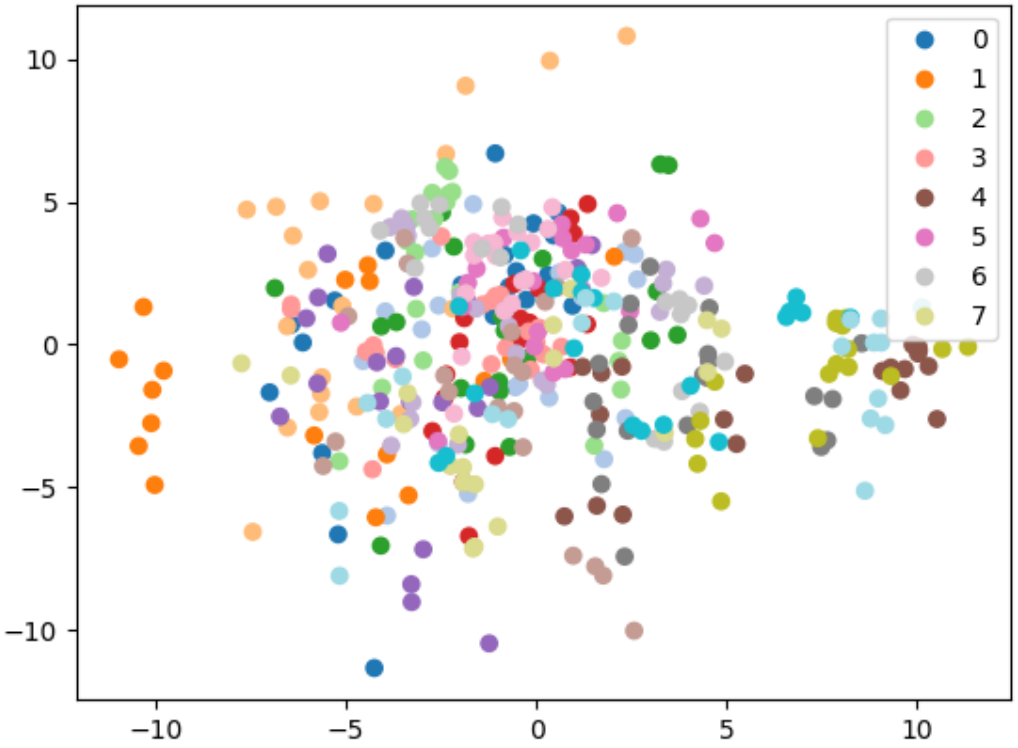
7.2. Mean image, largest eigenvectors and eigenvalues of training set



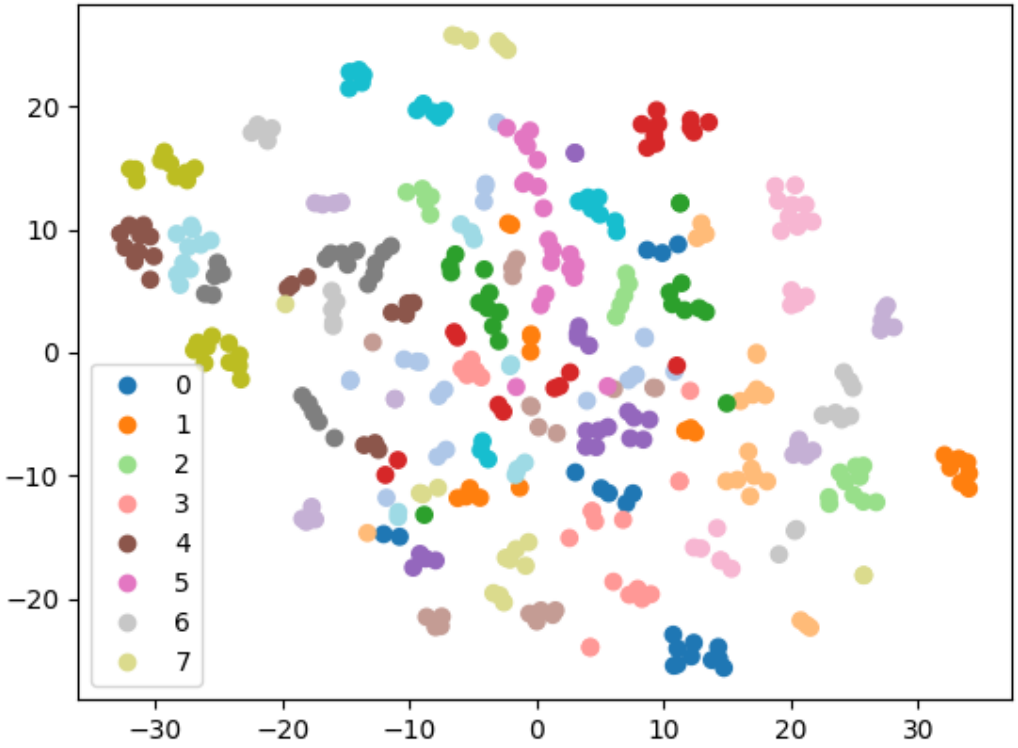




7.3. Dimensionality reduced to 2



7.4. t-SNE approach

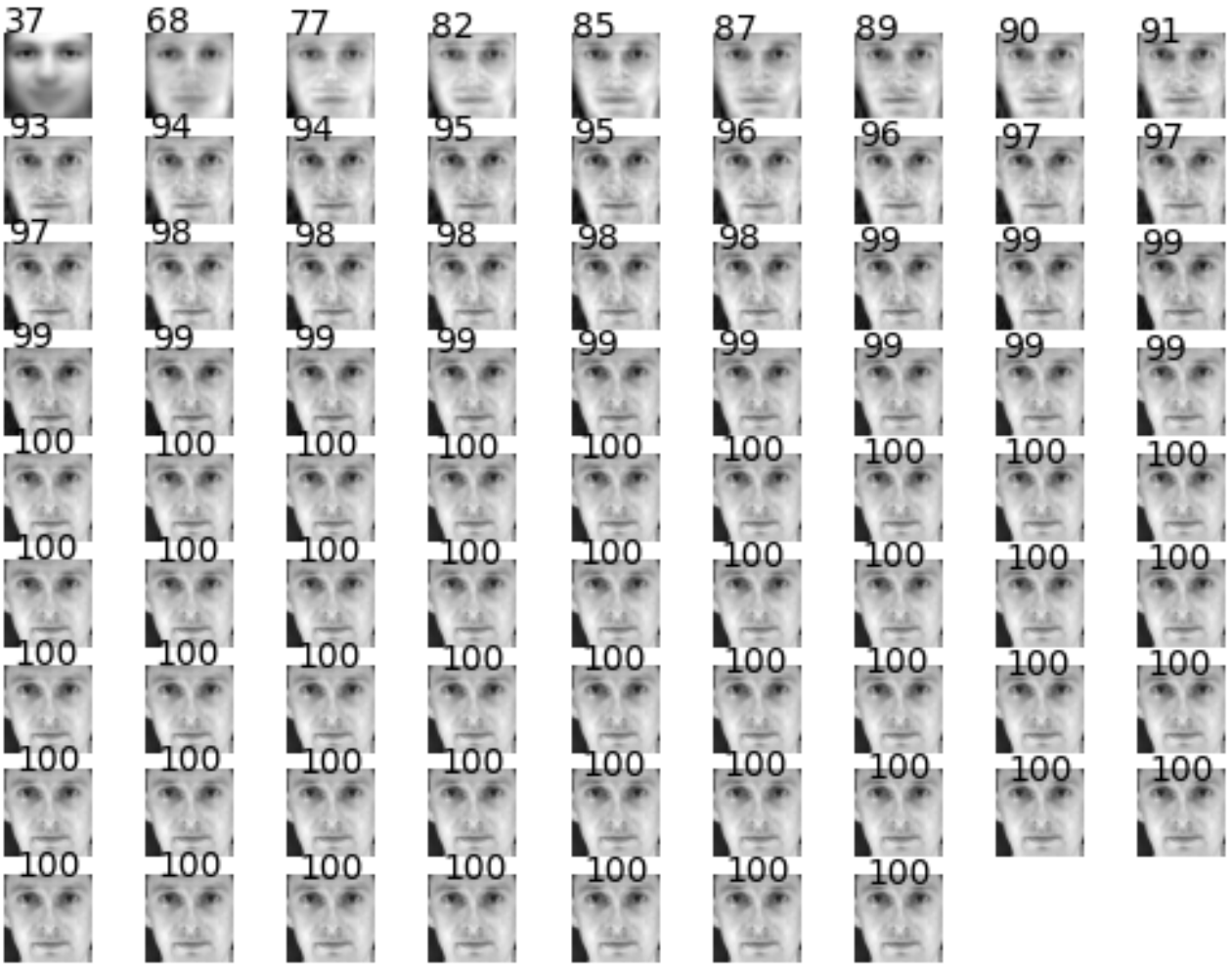


t-SNE clustered data way better than PCA.

Note: Even though legend maps colors from 0 to 7, there are 40 classes.

7.6. Reconstruction

To differentiate the person, 85 percent explained variance ratio is enough, I believe, which corresponds to 42 dimensions.



8. References

- https://lvdmaaten.github.io/publications/papers/JMLR_2008.pdf
- StatQuest: t-SNE, Clearly Explained | Youtube
- https://www.geeksforgeeks.org/difference-between-pca-vs-t-sne/
- https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html