Homework Assignment #7

21600004

Kang Seok-Un

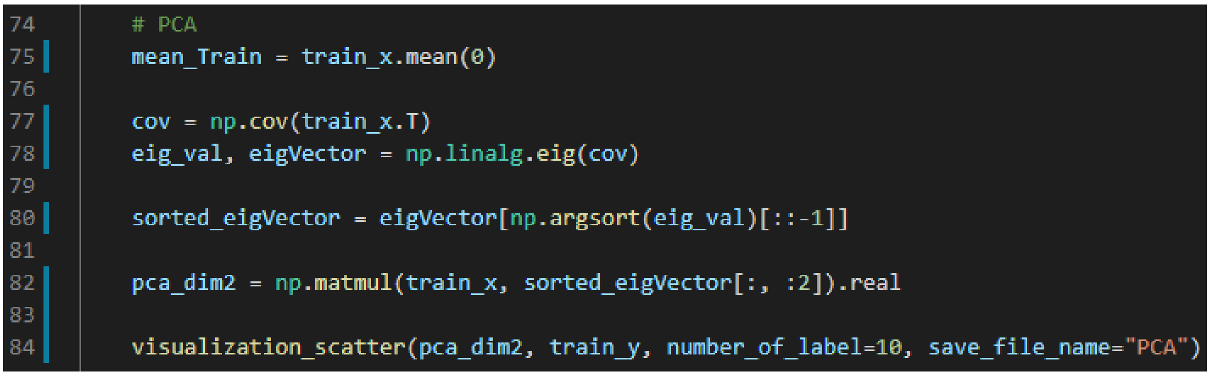
Introduction

For this assignment, I will compare the PCA and the t-SNE using the MNIST Dataset, a well-known public dataset. The dimension is reduced dimension of the MNIST dataset from 784-dim to 2-dim by using PCA and t-SNE. Afterward, I will try to compare PCA and t-SNE by plotting.

Experiments

1. PCA

To implement PCA, the same code as the code used in the last hw6 was used as shown in figure 1.

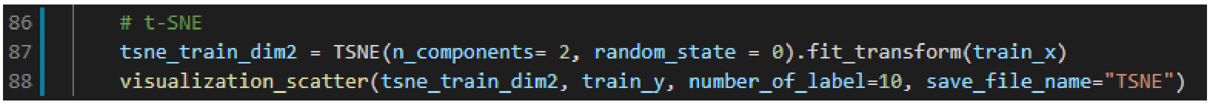


```
    # PCA
    mean_Train = train_x.mean(0)

    cov = np.cov(train_x.T)
    eig_val, eigVector = np.linalg.eig(cov)

    sorted_eigVector = eigVector[np.argsort(eig_val)[::-1]]

    pca_dim2 = np.matmul(train_x, sorted_eigVector[:, :2]).real

    visualization_scatter(pca_dim2, train_y, number_of_label=10, save_file_name="PCA")
```

Figure 1. Implemented Code for PCA

2. t-SNE

Unlike previous assignments, t-SNE was implemented using a library called "sklearn". The implementation codes are shown in Figure 2.



```
    # t-SNE
    tsne_train_dim2 = TSNE(n_components= 2, random_state = 0).fit_transform(train_x)
    visualization_scatter(tsne_train_dim2, train_y, number_of_label=10, save_file_name="TSNE")
```

Figure 2. Implemented Code of t-SNE

Results

Figure 3 shows the result of dimension reduction from 784-dim to 2-dim using PCA. Figure 4 shows the result of dimension reduction from 784-dim to 2-dim using t-SNE.

The first part that felt the difference while performing PCA and t-SNE is the execution speed. The t-SNE took a very long time compared to PCA. According to a research, the time complexity of t-SNE is $O(n^2)$, where n is the number of input data.

Since t-SNE has a huge time complexity, it would be one way to reduce the dimension of the input data to the proper dimension which can be explained original input dataset through the PCA and then plot the data through the t-SNE.

The next impressive part was how much it was distinguished for each class when comparing Figures 3 and 4. In the case of PCA, there were many overlapping parts rather than clustered the areas of each class. However, the t-SNE results of Figure 4 showed clearer boundaries for each class. But also not all data are clustered for each class, and it can be seen that there are also several data present in wrong classes.

PCA uses a linear method when projecting high dimension data into low dimension.t-SNE is a non-linear method that takes similarity between data in high dimension to low dimension when projecting high dimension data into low dimension.

Due to the above two differences, it is judged that there is a difference in the result of the plotting.

Conclusion

Unlike previous tasks, t-SNE was implemented using an open-source library. Because of this, unlike PCA and LDA, I did not implement t-SNE on my own, so the theoretical aspect of t-SNE is understood, but it is not determined whether it is really understood.

However, through this assignment, I learned that t-SNE looks clearer than PCA when visualizing high-dimensional datasets.

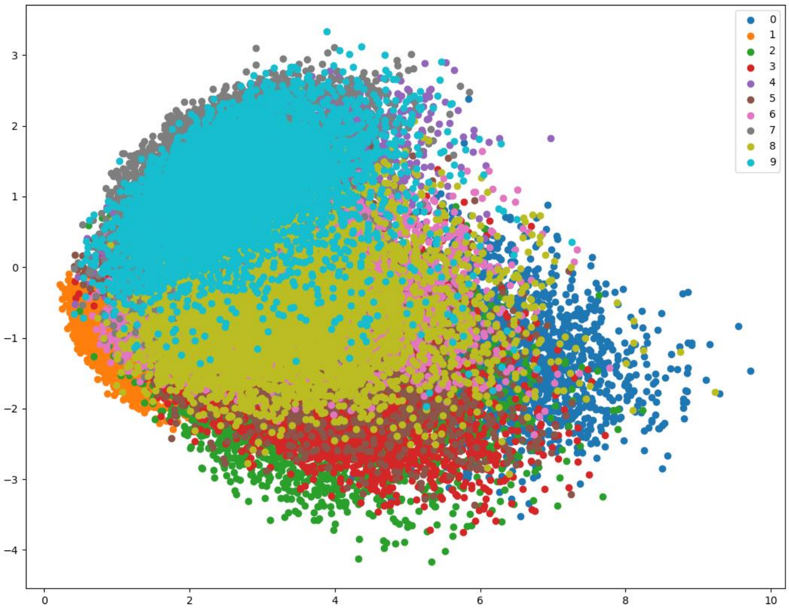


Figure 3. Plot Image of PCA which projected into 2-dim

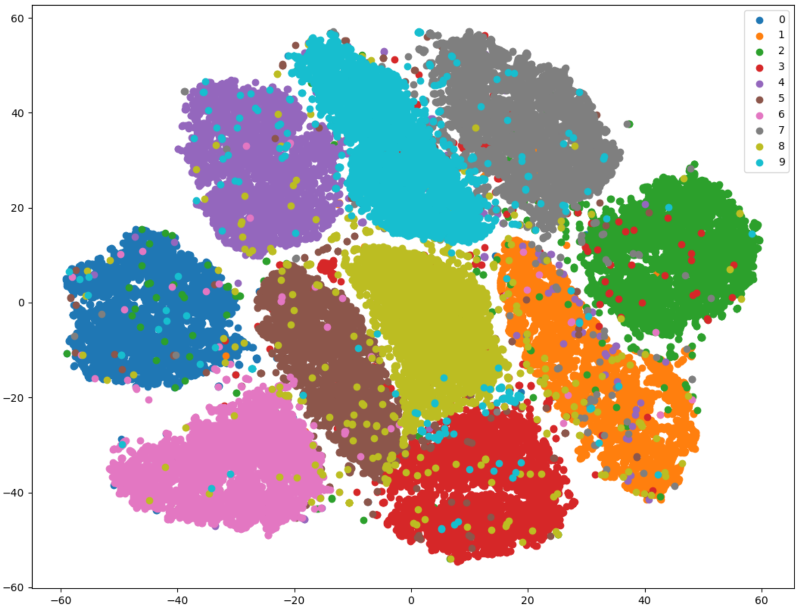


Figure 4. Plot Image of t-SNE which projected into 2-dim