GE 461: Introduction to Data Science

Spring 2024

Project: Dimensionality Reduction and Visualization

Some sample images and their labels from the dataset:

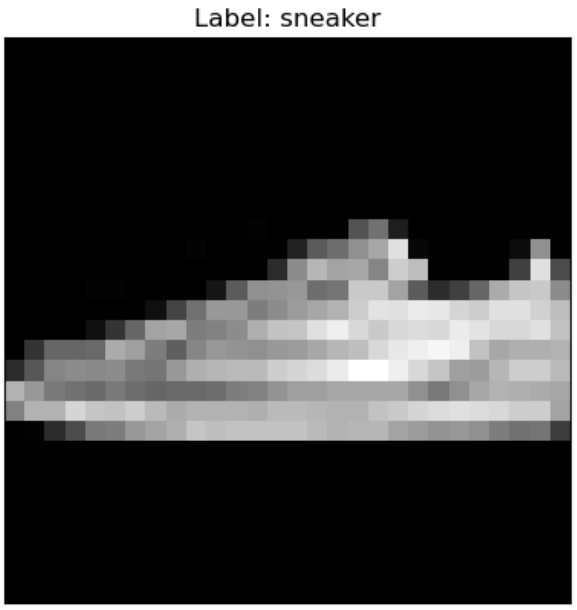

Figure 1: A sample sneaker image.

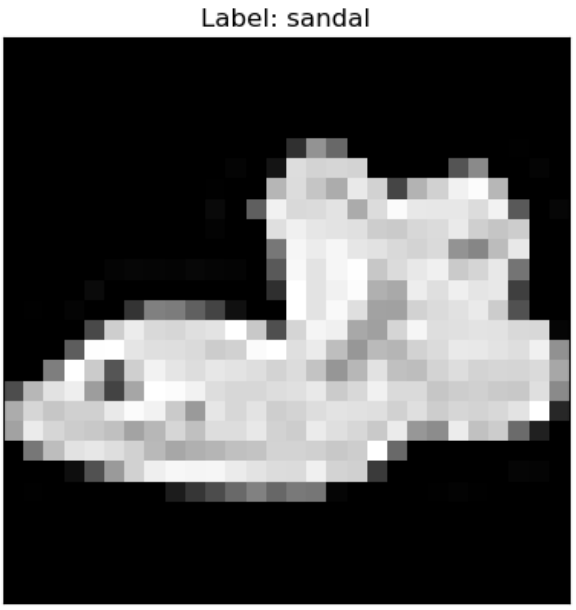

Figure 2: A sample sandal image.

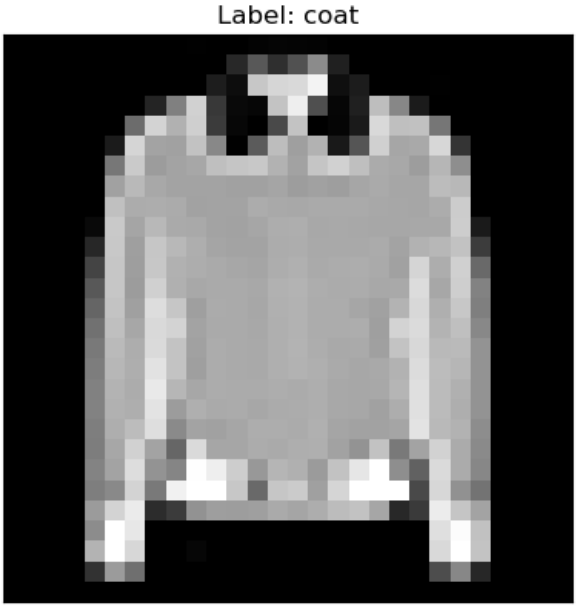

Figure 3: A sample coat image.

Question 1

2.

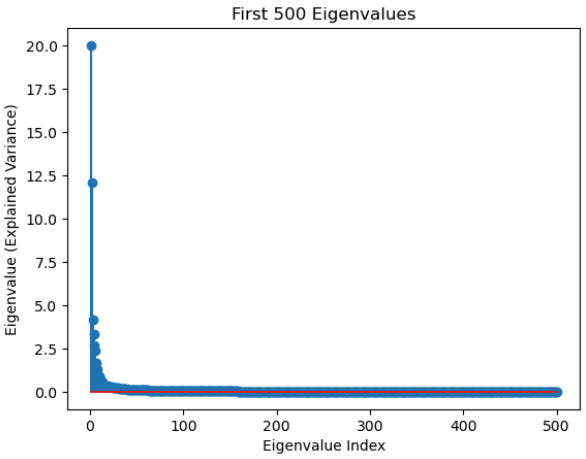

Figure 4: Stem plot of the first **500** eigenvalues.

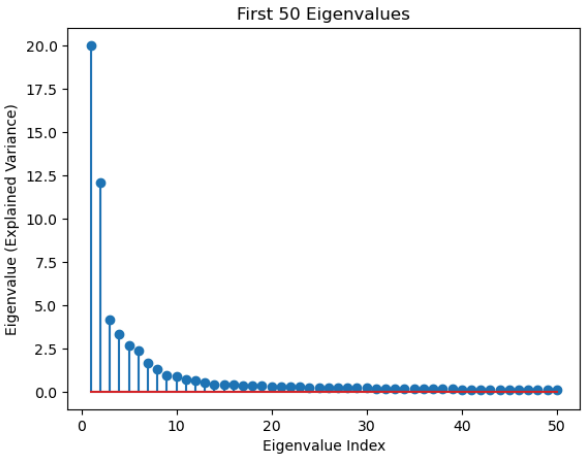

Figure 5: Stem plot of the first **50** eigenvalues.

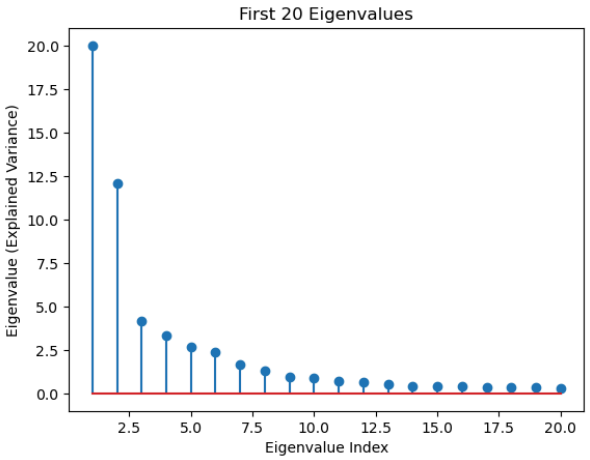

Figure 6: Stem plot of the first **20** eigenvalues.

Just by examining the three plots above, I would choose **the first 11 principal components (PC)**. After the 11th PC, the explained variance seems to have little significance.

3.

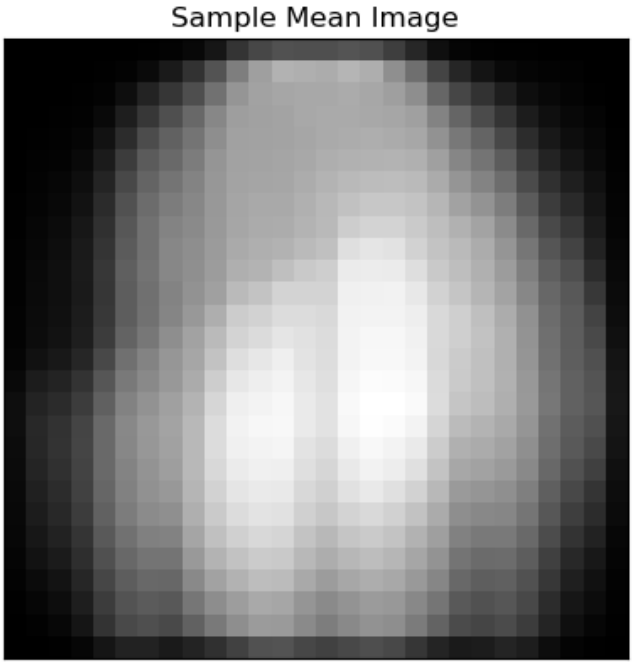

Figure 7: Sample mean image of the dataset.

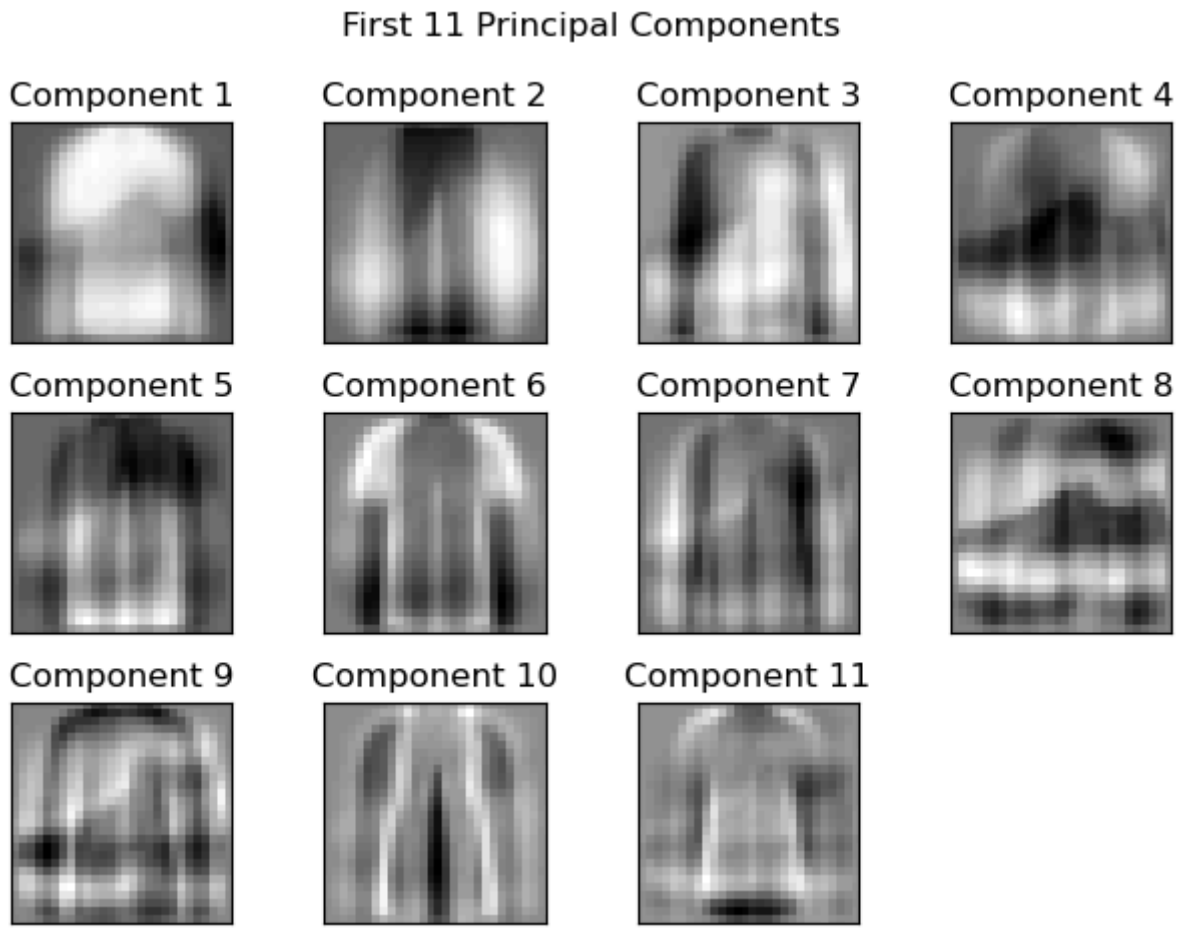

Figure 8: First 11 PCs.

I would have expected the pixel-wise mean of all training set images to be unrecognizable. This is because no separation was made with respect to the classes. After computations, I seem to be mostly correct. Although there seems to be some hints of some classes.

PCA is a method that finds orthogonal basis vectors that maximize the variance in the data. I would have expected PCA to separate two classes into each PC. However, there seems to be more than two classes in each PC. PCA seems to have separated multiple classes into different values in the same image. The separation in grayscale values get worse as the PC number increases. There also seems to be more categories in each PC as the PC number increases.

5.

The following plots are from evaluations of quadratic gaussian classifiers trained on sets with various dimensionalities obtained using PCA. **400** sets from 1 to 400 dimensions are tested.

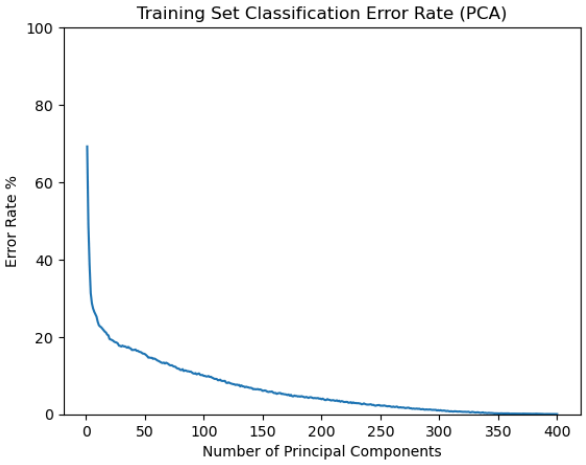

Figure 9: Error rate for the training set for various number of PCs.

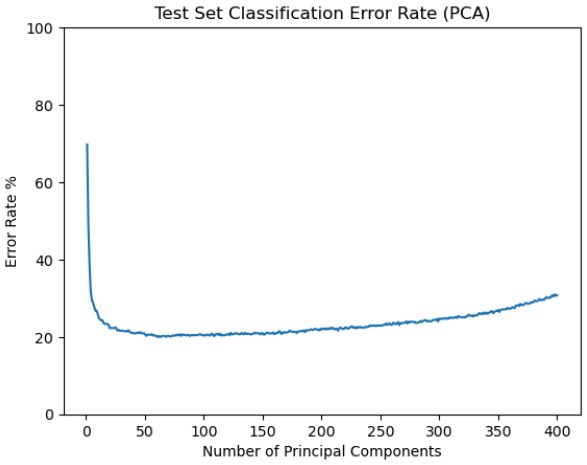

Figure 10: Error rate for the test set for various number of PCs.

Training set error rate decreases monotonically, while the test set error rate starts rising after approximately **65 PCs.** The minimum error rate achievable on the test set is approximately **20%**. After the optimum number of PCs, the model starts overfitting the training set and the generalization gets worse.

Question 2

The following plots are from evaluations of quadratic gaussian classifiers trained on sets with various dimensionalities obtained using random projections. **400** sets from 1 to 400 dimensions are tested.

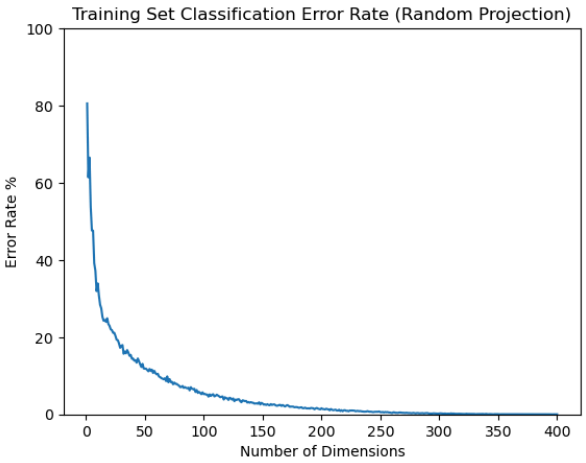

Figure 11: Error rate for the training set for various number of dimensions.

Figure 12: Error rate for the test set for various number of dimensions.

Training set error rate decreases monotonically, while the test set error rate starts rising after approximately **80 dimensions.** The minimum error rate achievable on the test set is approximately **22%**. After the optimum number of dimensions, the model starts overfitting the training set and the generalization gets worse.

Comparing random projection to PCA, **PCA performs slightly better with slightly less dimensions**. Additionally, the error rate vs number of dimensions plots of PCA are smoother while random projection has more noise.

Question 3

1.

Two dimensional Isomap projections with varying number of neighbors hyperparameter values:

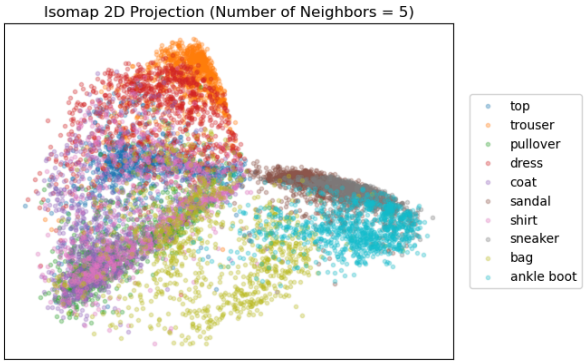

Figure 13: Projection of the whole dataset to 2 dimensions using the Isomap algorithm with the number of neighbors = **5**.

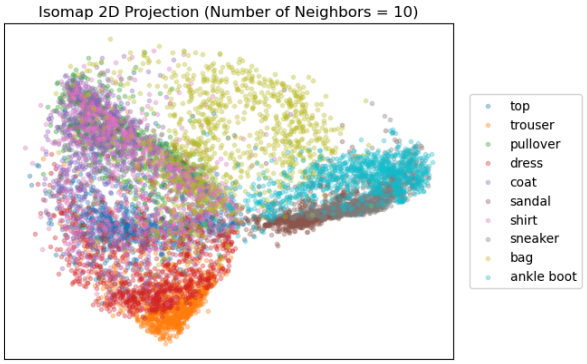

Figure 14: Projection of the whole dataset to 2 dimensions using the Isomap algorithm with the number of neighbors = **10**.

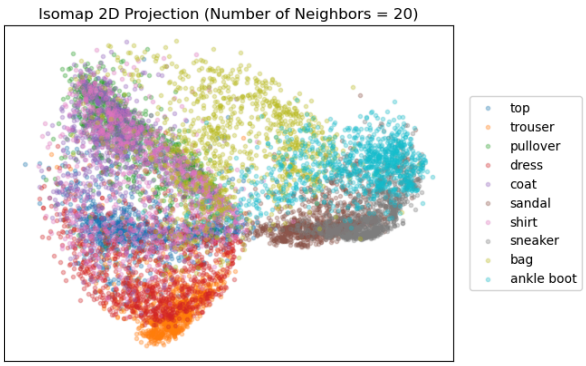

Figure 15: Projection of the whole dataset to 2 dimensions using the Isomap algorithm with the number of neighbors = **20**.

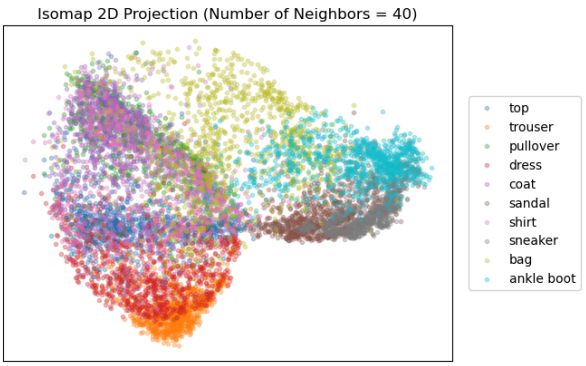

Figure 16: Projection of the whole dataset to 2 dimensions using the Isomap algorithm with the number of neighbors = **40**.

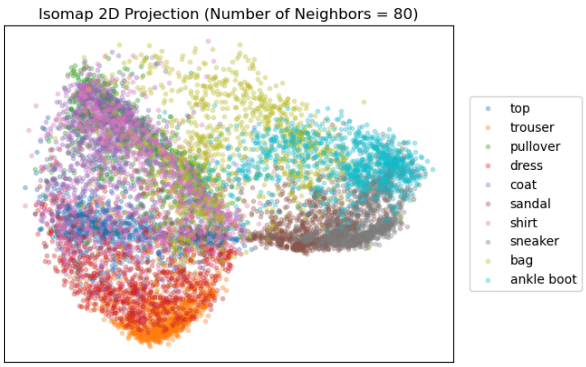

Figure 17: Projection of the whole dataset to 2 dimensions using the Isomap algorithm with the number of neighbors = **80**.

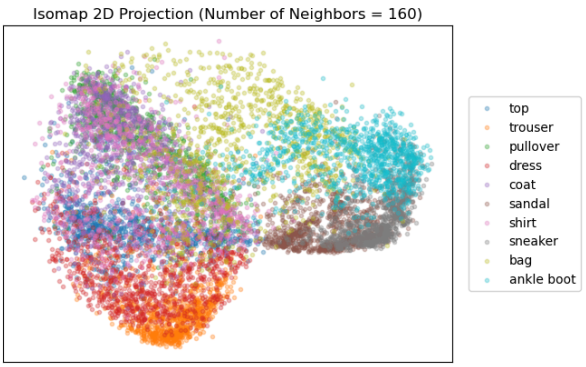

Figure 18: Projection of the whole dataset to 2 dimensions using the Isomap algorithm with the number of neighbors = **160**.

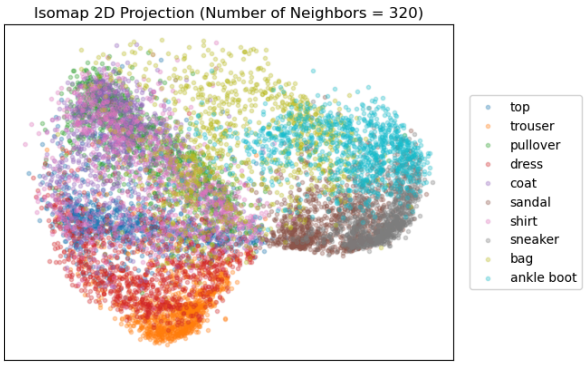

Figure 19: Projection of the whole dataset to 2 dimensions using the Isomap algorithm with the number of neighbors = **320**.

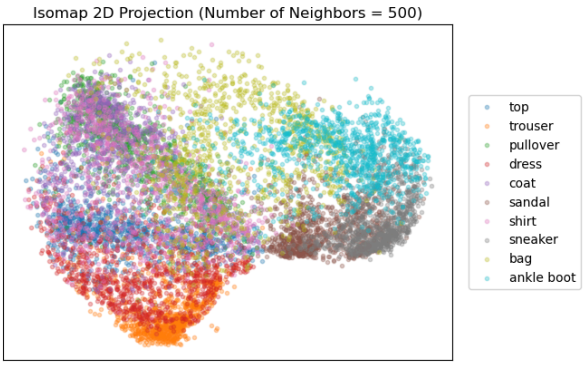

Figure 20: Projection of the whole dataset to 2 dimensions using the Isomap algorithm with the number of neighbors = **500**.

There does not seem to be a significant difference between various number of neighbors parameters. The only number of neighbors parameter value that was odd (5) resulted in a projection that is mirrored with respect to the horizontal axis compared to values that were even.

3.

The following plots are from evaluations of quadratic gaussian classifiers trained on sets with various dimensionalities obtained using Isomap. **400** equally spaced sets from 1 to 400 dimensions are tested. The number of neighbors parameter is set to **11**.

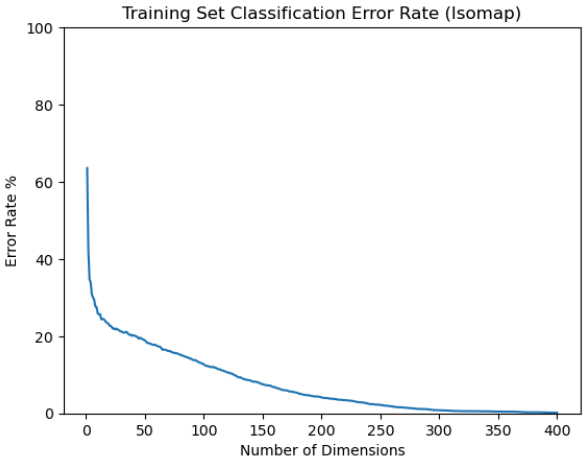

Figure 21: Error rate for the training set for various number of dimensions.

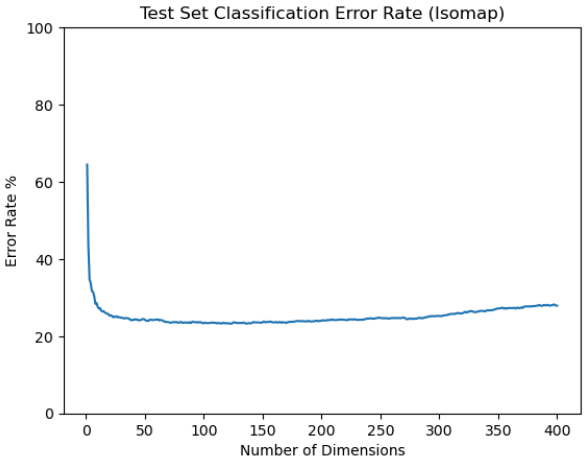

Figure 22: Error rate for the test set for various number of dimensions.

Training set error rate decreases monotonically, while the test set error rate plateaus after approximately **65 dimensions** and starts rising after approximately **175 dimensions.** The minimum error rate achievable on the test set is approximately **23%**. After the optimum number of dimensions, the model starts overfitting the training set and the generalization gets worse.

The Isomap method is more memory and processing power intensive compared to PCA and random projection methods. Furthermore, optimization of the number of neighbors hyperparameter will result in lower error rates.

In the context of the limitation mentioned above, the Isomap method results in worse error rates compared to PCA and random projection. However, the test set error rate starts out much lower and starts increasing after a much larger number of dimensions compared to other methods.

Question 4

The following plots represent t-SNE mappings with different **perplexity** values. Every mapping below is computed with learning rate 625/3, maximum number of iterations 1000, and the same random state (same seed).

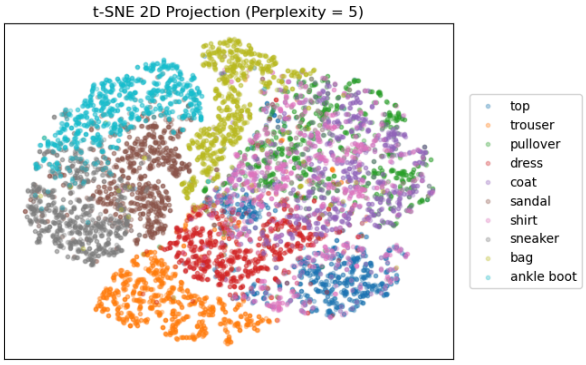

Figure 23: Two dimensional t-SNE projection with perplexity = **5**.

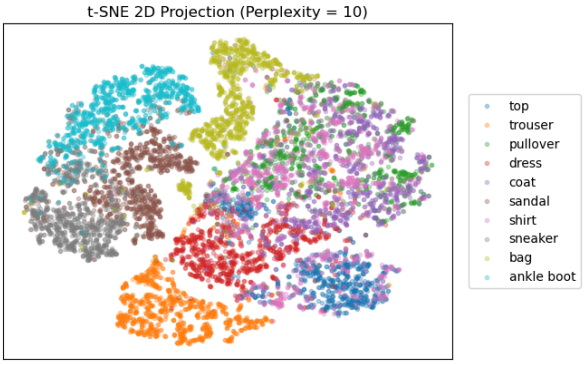

Figure 24: Two dimensional t-SNE projection with perplexity = **10**.

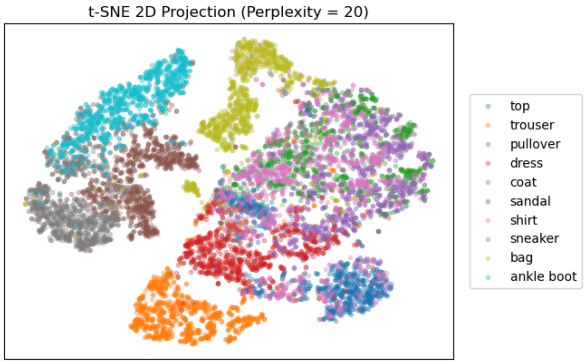

Figure 25: Two dimensional t-SNE projection with perplexity = **20**.

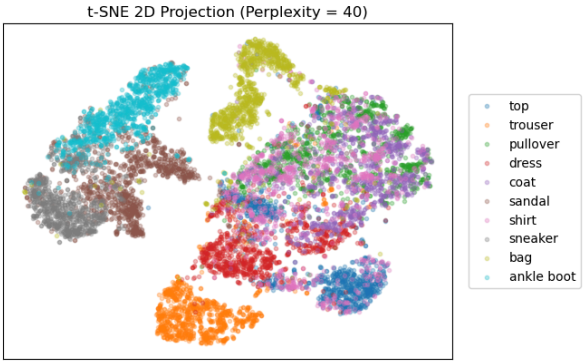

Figure 26: Two dimensional t-SNE projection with perplexity = **40**.

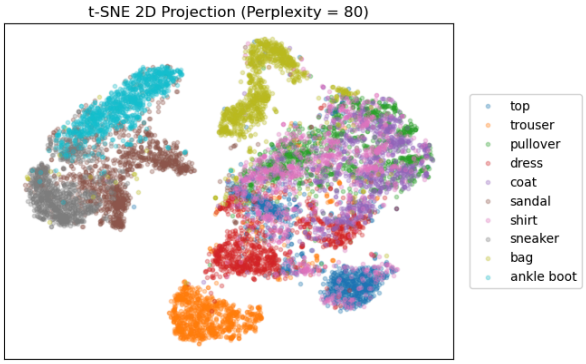

Figure 27: Two dimensional t-SNE projection with perplexity = **80**.

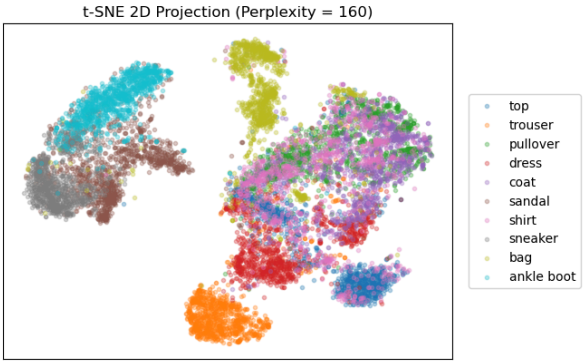

Figure 28: Two dimensional t-SNE projection with perplexity = **160**.

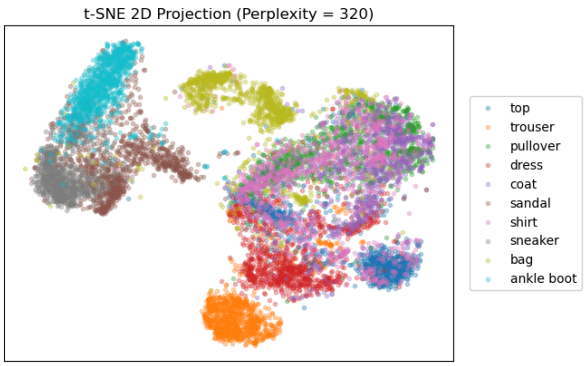

Figure 29: Two dimensional t-SNE projection with perplexity = **320**.

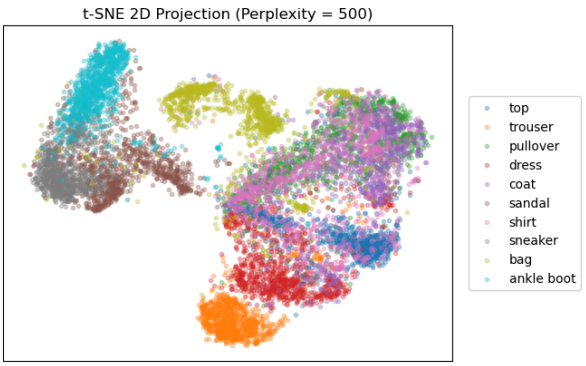

Figure 30: Two dimensional t-SNE projection with perplexity = **500**.

The clusters seem to separate as perplexity increases up to a certain point. Then cluster shapes start changing.

Tools Used

The code for this project was written in Python and many native and external modules were used. The **pathlib** module was used to get the paths of the dataset and label files [1]. **NumPy** was used for importing and working with the dataset [2]. **Matplotlib** was used to draw all of the figures in this report [3]. Finally, **scikit-learn** was used for all machine learning tasks [4].

References

[1] “pathlib — Object-oriented filesystem paths — Python 3.9.4 documentation,” docs.python.org. https://docs.python.org/3/library/pathlib.html

[2] Numpy, “NumPy,” *Numpy.org*, 2009. https://numpy.org/

[3] Matplotlib, “Matplotlib: Python plotting — Matplotlib 3.1.1 documentation,” Matplotlib.org, 2012. https://matplotlib.org/

[4] “scikit-learn: machine learning in Python,” *Scikit-learn.org*, 2019. https://scikit-learn.org/stable/