**COMP257 Unsupervised Learning**

**Section 003**

**Sirada Thoungvitayasutee 301292632**

**Question 1**

1. Output
   1. Each digit

A number written in black ink

Description automatically generated

* 1. 2D projection of PCS

A chart with orange dots

Description automatically generated

* 1. Explained Variance Ratio

A black text on a white background

Description automatically generated

* 1. 1D Hyperplane

A graph with a line

Description automatically generatedA graph with a line

Description automatically generated

* 1. Original shape and reduced shape of MNIST dataset

A close up of words

Description automatically generated

* 1. Original and compressed digit form

A white background with black dots

Description automatically generated

A white background with black dots

Description automatically generated

A white background with black dots

Description automatically generated

A white background with black dots

Description automatically generated

A white background with black and white clouds

Description automatically generated

A white background with black dots

Description automatically generated

A white background with black and white clouds

Description automatically generated

A white background with black dots

Description automatically generated

A white background with black dots

Description automatically generated

A white background with black dots

Description automatically generated

1. Code Explanation:

In the exercise, I began by loading the MNIST dataset and exploring each digit. Then I applied PCA to reduce the dataset to a 2-dimensional space and show the result. After this, I reduced the dimension to 1-dimensional and showed it onto a hyperplane. Finally, I applied incremental PCA to reduce the dataset to 154 dimensions and showed the result of the digits after this reduction.

1. Lesson Learned:

I learned that to plot projections onto a 1D hyperplane, I can use np.zeros\_like to zero out other dimensions and focus just one. Also, when displaying the compressed form of a digit, I had to apply the inverse transform back to the original form first. This is because the data is reduced to 154 dimensions and it cannot be reshaped in to the (28,28) for plotting.

**Question 2**

1. Output
   1. Generated Swiss Roll result

A graph of a graph of a graph

Description automatically generated with medium confidence

* 1. kPCA results for Linear Kernel

A graph of a graph

Description automatically generated with medium confidence

* 1. kPCA results for RBF Kernel

A diagram of a number of dots

Description automatically generated

* 1. kPCA results for Sigmoid Kernel

A colorful circle with numbers

Description automatically generated with medium confidence

* 1. Best parameters



* 1. Results from GridSearchCV

A graph with a line graph

Description automatically generated

1. Code Explanation:

I began by generating the Swiss Roll dataset by using make\_swiss\_roll from sklearn, and then visualized the result to show how the Swiss Roll structure looks in 3D. After that, I applied KernelPCA with three different kernels - linear, RBF and sigmoid - and plotted their 2D projections. Moreover, I applied logistic regression, achieving the accuracy of 85.8%. Finally I used GridSearchCV to optimize the model and plot the results.

1. Lesson Learned:

I learned that since the dataset consists of continuous data, applying logistic regression for classification required me to convert the data into discrete classes. To do this, I divided the data into three classes based on its percentile.